EXPERT INSIGHT

Cybersecurity Threats, Malware Trends, and Strategies

Mitigate exploits, malware, phishing, and other social engineering attacks

Tim Rains

Packt

Cybersecurity Threats, Malware Trends, and Strategies

Mitigate exploits, malware, phishing, and other social engineering attacks

Tim Rains

Packt

BIRMINGHAM - MUMBAI

Cybersecurity Threats, Malware Trends, and Strategies

Producer: Tushar Gupta
Acquisition Editor – Peer reviews: Divya Mudaliar
Content Development Editor: James Robinson-Prior
Technical Editor: Karan Sonawane
Project Editor: Janice Gonsalves
Copy Editor: Safis Editing
Proofreader: Safis Editing
Indexer: Rekha Nair
Presentation Designer: Sandip Tadge

First published: May 2020

Production reference: 1280520

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham B3 2PB, UK.

ISBN 978-1-80020-601-4

www.packt.com

packt.com

Subscribe to our online digital library for full access to over 7,000 books and videos, as well as industry leading tools to help you plan your personal development and advance your career. For more information, please visit our website.

Why subscribe?

- Spend less time learning and more time coding with practical eBooks and Videos from over 4,000 industry professionals
- Learn better with Skill Plans built especially for you
- Get a free eBook or video every month
- Fully searchable for easy access to vital information
- Copy and paste, print, and bookmark content

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at www.Packt.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at customercare@packtpub.com for more details.

At www.Packt.com, you can also read a collection of free technical articles, sign up for a range of free newsletters, and receive exclusive discounts and offers on Packt books and eBooks.

Contributors

About the author

Tim Rains worked at Microsoft for the better part of two decades where he held a number of roles, including Global Chief Security Advisor, Director of Security, Identity and Enterprise Mobility, Director of Trustworthy Computing, and was a founding technical leader of Microsoft's customer facing Security Incident Response team.

Currently, Tim works at Amazon Web Services as Regional Leader, Security and Compliance Business Acceleration for Europe, the Middle East and Africa.

Tim lives with his wife Brenda and their two sons Tristan and Liam in London, England.

I'd like to thank my wife Brenda for her encouragement, assistance, and patience, without which this book would not have been written. Yuri Diogenes, thank you for your encouragement. Thank you Dr. ir Johannes Drooghaag for being our Technical Reviewer and to the entire team at Packt.

About the reviewer

Dr. ir Johannes Drooghaag is an Executive Consultant, Trainer, Author and Thought Leader for Cybersecurity, Sustainability and Privacy. After 30 years in leading roles at international corporations, Dr. ir Johannes Drooghaag founded his consultancy under his own name, which is based on the *Spearhead Management Model* he developed.

In his work, he focuses on building bridges between human potential and technological capabilities. This means teaching highly skilled technical leaders the soft skills of leadership and the mindset of leaders and growing the technical savviness of business leaders to be able to make the right decisions about risks and mitigations.

Dr. ir Johannes Drooghaag advocates for kids to be involved in Cybersecurity and Privacy Awareness as early as possible with free programs like *Internet Safety for Kids (and their Parents)*. His workshop *Cyber Security for Road Warriors (and Couch Potatoes)* focuses on frequent travelers and remote workers and has made thousands of people aware of the cyber risks surrounding them and the steps they can take to avoid becoming a victim.

Table of Contents

Preface

Imagine you are in a submarine, submerged miles below the surface surrounded by dark, freezing water. The hull of the submarine is under constant immense pressure from all directions. A single mistake in the design, construction or operation of the submarine spells disaster for it and its entire crew.

This is analogous to the challenge **Chief Information Security Officers** (**CISOs**) and their teams face today. Their organizations are surrounded on the internet by attackers that are constantly probing for ways to penetrate and compromise their organization's IT infrastructure. The people in their organizations receive wave after wave of social engineering attacks designed to trick them into making poor trust decisions that will undermine the controls that their security teams have implemented. The specters of ransomware and data breaches continue to haunt CISOs, **Chief Information Officers** (**CIOs**) and **Chief Technology Officers** (**CTOs**) of the most sophisticated organizations in the world.

After conducting hundreds of incident response investigations and publishing thousands of pages of threat intelligence, I have had the opportunity to learn from and advise literally thousands of businesses and public sector organizations all over the world. I wrote this book to share some of the insights and lessons I've learned during this extraordinary journey.

The views and opinions expressed in this book are my own and not those of my past or present employers.

Who this book is for?

This book is for CISOs, aspiring CISOs, senior managers in the office of the CISO, CIOs, CTOs and other roles who have meaningful responsibility for the cybersecurity of their organizations.

What this book covers

Chapter 1, *Ingredients for a Successful Cybersecurity Strategy*, provides a detailed look at the ingredients that are necessary for a successful cybersecurity program.

Chapter 2, *Using Vulnerability Trends to Reduce Risk and Costs*, provides a unique
20-year view of vulnerabilities, using vulnerability disclosure data from the National Vulnerability Database. This will help the reader more accurately evaluate the efficacy of cybersecurity strategies discussed in later chapters.

Chapter 3, *The Evolution of the Threat Landscape – Malware*, provides a unique data-driven perspective of how malware has evolved around the world over a 10 year period. This helps the reader understand the types of malware threats they face and which malware threats are most, and least, prevalent.

Chapter 4, *Internet-Based Threats*, examines some of the way's attackers have been using the internet and how these methods have evolved over time. This chapter dives into phishing attacks, drive-by download attacks and malware hosting sites.

Chapter 5, *Cybersecurity Strategies*, discusses the major cybersecurity strategies employed in the industry for the past 20 years or so. This chapter introduces the Cybersecurity Fundamentals Scoring System, which enables the reader to estimate an efficacy score for any cybersecurity strategy.

Chapter 6, *Strategy Implementation*, provides an example of how one of the best cybersecurity strategies identified can be implemented. This chapter illustrates how an Attack-Centric Strategy, namely the Intrusion Kill Chain, can be implemented.

Chapter 7, *Measuring Performance and Effectiveness*, looks at the challenge that CISOs and security teams have always had and how to measure the effectiveness of their cybersecurity program. This chapter examines how to measure the performance and effectiveness of a cybersecurity strategy.

Chapter 8, *The Cloud – A Modern Approach to Security and Compliance*, provides an overview of how the cloud is a great cybersecurity talent amplifier. This chapter looks at how the cloud can mitigate the ways enterprises typically get compromised. Additionally, this chapter dives into how security teams can use encryption and key management to protect data in the cloud.

To get the most out of this book

- You'll already understand basic **Information Technology** (**IT**) concepts and have some experience using, implementing, and/or operating IT systems and applications.
- Experience managing enterprise IT and/or cybersecurity teams will be helpful, but is not strictly required.
- You'll bring a healthy appetite to learn about some of the aspects of cybersecurity that you might not have been exposed to in the past.

Conventions used

The following conventions are used in the book:

A block of code is set as follows:

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "Example:user123",
        "arn": "arn:aws:sts::Example:assumed-role/Admin/user123",
        "accountId": "Example-ID",
}
```

Bold: Indicates a new term or an important word.

Download the color images

We also provide a PDF file that has color images of the screenshots/diagrams used in this book. You can download it here: `https://static.packt-cdn.com/downloads/9781800206014_ColorImages.pdf`.

Get in touch

Feedback from our readers is always welcome.

General feedback: Email `feedback@packtpub.com`, and mention the book's title in the subject of your message. If you have questions about any aspect of this book, please email us at `questions@packtpub.com`.

Errata: Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you have found a mistake in this book we would be grateful if you would report this to us. Please visit, `http://www.packtpub.com/submit-errata`, selecting your book, clicking on the Errata Submission Form link, and entering the details.

Piracy: If you come across any illegal copies of our works in any form on the Internet, we would be grateful if you would provide us with the location address or website name. Please contact us at `copyright@packtpub.com` with a link to the material.

If you are interested in becoming an author: If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, please visit .

Reviews

Please leave a review. Once you have read and used this book, why not leave a review on the site that you purchased it from? Potential readers can then see and use your unbiased opinion to make purchase decisions, we at Packt can understand what you think about our products, and our authors can see your feedback on their book. Thank you!

For more information about Packt, please visit `packtpub.com`.

1

Ingredients for a Successful Cybersecurity Strategy

There's no doubt that enterprises today, more than ever, need effective cybersecurity strategies. However a sound strategy is not in and of itself a guarantee of success. There are several ingredients that are necessary for a cybersecurity program to be successful. This chapter will describe what a cybersecurity strategy looks like and each of the necessary ingredients for success in detail.

Throughout this chapter, we'll cover the following topics:

- Defining the term *cybersecurity strategy*
- Common ways organizations become compromised, and how the mitigation of these are fundamental to effective cybersecurity
- Understanding the difference between an attacker's motivation and their tactics
- Additional guidance on formulating a successful cybersecurity strategy

Let's begin with a fundamental question that we'll need to answer before discussing cybersecurity strategy in any detail: what do we actually mean when we say "cybersecurity strategy"?

What is a cybersecurity strategy?

Organizations that have a super-strong security culture, essentially have cybersecurity baked into them. For everyone else, there's strategy. In my experience, the terms "strategy" and "tactics" are poorly understood in the business world. One person's strategy is another person's tactics. I once worked with a Corporate Vice President who would tell me that I was talking about tactics when I was explaining our strategy. Throughout my career, I've been in meetings where people have talked past each other because one person is discussing strategies and the other is discussing tactics.

Additionally, security and compliance professionals sometimes use the term "strategy" when they are referring to frameworks, models, or standards. There are lots of these in the industry and many organizations use them. For example, ISO standards, NIST standards, OWASP Top 10, CIS Benchmarks, STRIDE, risk management frameworks, SOC 2, PCI, HIPAA, the Cloud Security Alliance Cloud Controls Matrix, the AWS Cloud Adoption Framework Security Perspective, AWS Well-Architected Security Pillar, and many more. All of these can be helpful tools for organizations seeking to improve their security postures, comply with regulations, and demonstrate that they meet industry standards.

I'm not proposing a new dictionary definition of the term "strategy," but I do want to explain what I mean when I'm discussing cybersecurity strategies in this book. In my view, there are at least two critical inputs to a cybersecurity strategy:

1. Each organization's high-value assets
2. The specific requirements, threats, and risks that apply to each organization, informed by the industry they are in, the place(s) in the world where they do business, and the people associated with each organization

High Value Assets (**HVAs**) are also known as "crown jewels." There are many definitions for these terms. But when I use them, I mean the organization will fail or be severely disrupted if the asset's confidentiality, integrity, or availability is compromised. HVAs are rarely the computers that the organization's information workers use. Yet I've seen so many organizations focus on the security of desktop systems as if they were HVAs.Given the importance of HVAs, it would be easy to focus on them to the exclusion of lower-value assets. But keep in mind that attackers often use lower-value assets as an entry point to attack HVAs. For example, those old development and test environments that were never decommissioned properly, typically, aren't HVAs. But they are often found to be a source of compromise.

One of the first things a CISO needs to do when they get the job is to identify the organization's HVAs. This might be more challenging than it sounds as the crown jewels might not be obvious to people that don't possess expertise specifically related to the business they are supporting. Interviewing members of the C-suite and members of the board of directors can help to identify assets that would truly cause the business to fail or be severely disrupted.

Working backward from the organization's objectives can also help identify its HVAs. As CISOs do this analysis, they should be prepared for some nuances that weren't initially obvious. For example, could the business still meet its objectives without power, water, heating, air conditioning, and life-safety systems? Depending on the business and the type of building(s) it uses, if elevators aren't available, is there any point letting employees and customers through the front door? Customers might be willing to walk up a few flights of stairs, but would they be willing to walk up 40 flights of stairs if that was necessary? Probably not.

If this disruption was sustained for days, weeks, or months, how long could the business survive? Where are the control systems for these functions? And when was the last time the security posture of these systems was assessed? Identifying an organization's HVAs doesn't mean that CISOs can ignore everything else. Understanding which assets are truly HVAs and which aren't helps CISOs prioritize their limited resources and focus on avoiding extinction events for the organization.

Once the CISO has identified their organization's crown jewels, the next step is to ensure that the C-suite and board of directors understand and agree with that list. This clarity will be very helpful when the time comes to request more resources or different resources than the organization has leveraged in the past. When the organization needs to make hard decisions about reductions in resources, clarity around HVAs will help make risk-based decisions. The time and effort spent getting the senior stakeholder community on the same page will make the CISO's life easier moving forward.

The second critical input to a cybersecurity strategy is the specific requirements, threats, and risks that apply to the organization, informed by the industry they are in, the place(s) in the world where they do business, and the people associated with it. This input helps further scope the requirements of the cybersecurity program. For example, the industry and/or location where they do business might have regulatory compliance requirements that they need to observe, or they could face stiff fines or get their business license revoked. Keep in mind that most organizations can't identify all possible threats and risks to them. That would require omniscience and is a natural limitation of a risk-based approach.

After publishing thousands of pages of threat intelligence when I worked at Microsoft (Microsoft Corporation, 2007-2016), I can tell you that there are global threats that have the potential to impact everyone, but there are also industry-specific threats and regional threats. Using credible threat intelligence to inform the strategy will help CISOs prioritize capabilities and controls, which is especially helpful if they don't have unlimited resources. Trying to protect everything as if it's of the same value to the organization is a recipe for failure. CISOs have to make trade-offs, and it's better if they do this knowing the specific threats that really apply to the industry and region of the world where they do business. This doesn't mean CISOs can ignore all other threats, but identifying the highest-risk threats to their organization's crown jewels will help them focus resources in the most important places.

I have dedicated three chapters in this book to help you understand the threat landscape and how it has evolved over the last 20 years. *Chapter 2*, *Using Vulnerability Trends to Reduce Risk and Costs*, dives deep into vulnerability management and will show you how vulnerability disclosures have trended over the past two decades. *Chapter 3*, *The Evolution of the Threat Landscape – Malware*, focuses on how malware has evolved over the last 20 years. *Chapter 4*, *Internet-Based Threats*, examines internet-based threats that every organization should seek to mitigate.

Without the two inputs I've described here, CISOs are left implementing "best practices" and industry standards that are based on someone else's threat model. Again, these can be helpful in moving organizations in the right direction, but they typically aren't based on the HVAs of individual organizations and the specific threats they care about. Using best practices and industry standards that aren't informed by these two inputs will make it more likely that there will be critical gaps.

At this point, you might be wondering what a cybersecurity strategy looks like. The following diagram represents a cybersecurity strategy. HVAs are central and are supported by the other parts of the strategy. The cybersecurity fundamentals include the foundational capabilities that support a successful security program, such as vulnerability management and identity management, among others.

Advanced cybersecurity capabilities are investments that organizations should make as they become very proficient at the fundamentals. If your organization isn't really good at the fundamentals, then don't bother investing in advanced cybersecurity capabilities, as attackers won't need to do anything "advanced" to successfully compromise the environment and subvert those advanced capabilities.

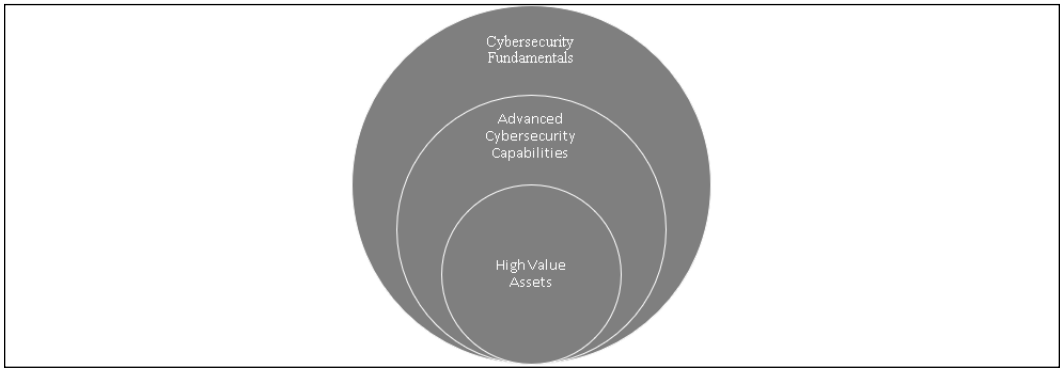


Figure 1.1: An illustrative example of a cybersecurity strategy

Now that we have a good idea of what cybersecurity strategy entails, let's examine what I consider to be a critical ingredient of cybersecurity strategies: the common ways that organizations are compromised.

How organizations get initially compromised and the cybersecurity fundamentals

The foundation of the strategy is what I call the "cybersecurity fundamentals." A solid foundation is required for a successful strategy. The cybersecurity fundamentals are based on the threat intelligence I mentioned earlier. After performing hundreds of incident response investigations and studying Microsoft's threat intelligence for over a decade, I can tell you with confidence that there are only five ways that organizations get *initially* compromised. After the initial compromise, there are many, many **tactics, techniques, and procedures** (**TTPs**) that attackers can use to move laterally, steal credentials, compromise infrastructure, remain persistent, steal information, and destroy data and infrastructure. Some of these have been around for decades and some are new and novel.

The five ways that organizations get initially compromised are what I call the "cybersecurity usual suspects":

1. Unpatched vulnerabilities
2. Security misconfigurations
3. Weak, leaked, and stolen credentials
4. Social engineering
5. Insider threats

The cybersecurity fundamentals are the part of the strategy that focuses on mitigating the cybersecurity usual suspects. Let's look at each one of these in more detail, starting with the exploitation of unpatched vulnerabilities.

Unpatched vulnerabilities

A vulnerability is a flaw in software or hardware design and/or the underlying programming code that allows an attacker to make the affected system do something that wasn't intended. The most severe vulnerabilities allow attackers to take complete control of the affected system, running arbitrary code of their choice. Less severe vulnerabilities lead to systems disclosing data in ways that weren't intended or denying service to legitimate users. In *Chapter 2*, *Using Vulnerability Trends to Reduce Risk and Costs*, I provide a deep dive into vulnerability management and some of the key vulnerability disclosure trends over the past 20 years. I'll save that in-depth discussion for the next chapter, but I will provide some more context here.

Attackers have been using vulnerabilities to compromise systems at scale since at least the days of Code Red and Nimda in 2001. In 2003, SQL Slammer and MSBlaster successfully disrupted the internet and compromised hundreds of thousands of systems worldwide by exploiting unpatched vulnerabilities in Microsoft Windows operating systems. In the years following these attacks, a cottage industry developed an ongoing effort to help enterprise organizations, those with the most complex environments, inventory their IT systems, identify vulnerabilities in them, deploy mitigations for vulnerabilities, and patch them. At the end of 2019, there were over 122,000 vulnerabilities disclosed in software and hardware products from across the industry, on record, in the National Vulnerability Database (National Vulnerability Database, n.d.). As you'll read in *Chapter 2*, *Using Vulnerability Trends to Reduce Risk and Costs*, the number of vulnerabilities disclosed across the industry surged between 2016 and 2020, reaching levels never seen before.

An economy has evolved around the supply and demand for vulnerabilities and exploits, with a varied list of participants including vendors, attackers, defenders, various commercial entities, governments, and others. The number of participants in this economy and their relative sophistication make it harder for organizations to protect themselves from the exploitation of vulnerabilities in their IT environment by pressurizing the associated risks. Using unpatched vulnerabilities are a mainstay of attackers' toolkits.

Organizations that are highly efficient and competent at vulnerability management make it much harder for attackers to successfully attack them.

A well-run vulnerability management program is a fundamental component and a critical requirement of a cybersecurity strategy. Without it, organizations' cybersecurity efforts will fail regardless of the other investments they make. It's important enough to reiterate this point. Unpatched vulnerabilities in operating systems, and the underlying platform components that advanced cybersecurity capabilities rely on, enable attackers to completely undermine the effectiveness of these investments. Failing to efficiently address ongoing vulnerability disclosures in the "trusted computing base" that your systems rely on renders it untrustworthy.

An accurate inventory of all IT assets is critical for a vulnerability management program. Organizations that can't perform accurate and timely inventories of all their IT assets, scan all IT assets for vulnerabilities, and efficiently mitigate and/or patch those vulnerabilities, shouldn't bother making other investments until this is addressed. If your organization falls into this category, please reread the preface section of this book and recall the submarine analogy I introduced. If the CISO and vulnerability management program managers rely on their organization's IT group or other internal partners to provide IT asset inventories, those inventories need to be complete – not just inventories of systems they want to comply with.

Assets that don't show up in inventories won't get scanned or patched and will become the weak link in the security chain you are trying to create. Very often, this is at odds with the uptime objectives that IT organizations are measured against, because patching vulnerabilities increases the number of system reboots and, subsequently, decreases uptime even if everything goes smoothly. My advice in scenarios where asset inventories are provided by parties other than the vulnerability management program itself is to trust but verify. Spend the extra effort and budget to continually check asset inventories against reality. This includes those official and unofficial development and test environments that have been responsible for so many breaches in the industry over the years.

If the sources of asset inventories resist this requirement or fail to provide accurate, timely inventories, this represents the type of risk that the board of directors should be informed of. Providing them with a view of the estimated percentage of total asset inventory currently not managed by your vulnerability management program should result in the sources of asset inventories reprioritizing their work and the disruption of a dangerous status quo. I'll discuss vulnerability management in more detail in *Chapter 2*, *Using Vulnerability Trends to Reduce Risk and Costs*, of this book. I'll also discuss vulnerability management in *Chapter 8*, *The Cloud – A Modern Approach to Security and Compliance*, on cloud computing.

The cloud can render the old-fashioned methods of inventorying, scanning, and patching security vulnerabilities obsolete.

Of course, one challenge with the approach I just described is environments that have embraced **Bring Your Own Device** (**BYOD**) policies that allow information workers to use their personal mobile devices to access and process enterprise data. The underlying question is whether enterprise vulnerability management teams should inventory and manage personal devices? This debate is one reason why many security professionals originally dubbed BYOD as "Bring Your Own Disaster." Different organizations take different approaches when answering this question. Some organizations give employees corporate-owned and fully managed mobile devices, while others require personal devices to enroll in enterprise mobile device management programs. I've also seen a more passive management model, where users are required to have a access pin on their devices and aren't allowed to connect to their employers' networks if the latest mobile operating system version isn't installed on their devices. Some organizations use **Network Access Control** (**NAC**) or **Network Access Protection** (**NAP**) technologies to help enforce policies related to the health of systems connecting to their network. Minimizing the number of unpatched systems allowed to connect to enterprise networks is a best practice, but can be challenging to accomplish depending on corporate cultures and mobile device policies. Collecting data that helps security teams understand the risk that mobile devices pose to their environments is very helpful for a rationalized risk-based approach.

Next, we'll consider security misconfigurations. Like unpatched vulnerabilities, security misconfigurations can potentially enable attackers to take a range of actions on a system including disrupting its operation, stealing information, lowering security settings or disabling security features, seizing control of it, and using it to attack other systems.

Security misconfigurations

Security misconfigurations can be present in a system as the default setting, like a preset key or password that is the same on every system manufactured by a vendor. Security misconfigurations can also be introduced gradually as a system's configuration changes incrementally as it's managed over time.

After performing hundreds of incident response investigations while I was on the customer-facing incident response team at Microsoft, I can tell you that a significant percentage of systems get initially compromised through security misconfigurations.

This is especially true of internet-facing systems such as web servers, firewalls, and other systems found in enterprise **demilitarized zones** (**DMZs**). Once a misconfiguration enables an attacker to control a system in a DMZ or use it to send authenticated commands on the attacker's behalf (such as a server-side request forgery attack), the attacker aspires to use the system to gain access to other systems in the DMZ and ultimately get access to systems inside the internal firewall of the organization. This has been a common pattern in attackers' playbooks for 20 years or more.

Security misconfigurations have also plagued endpoint devices, such as PCs, smartphones, and **Internet of Things** (**IoT**) devices. The infrastructures that these endpoints connect to, such as wireless access points, are also frequently probed by attackers for common misconfigurations. Security misconfigurations have also been an issue in **industrial control systems** (**ICS**). For example, one scenario with ICS that has burned security teams in the past is "fall back to last known status," which can override more recent security configuration changes in favor of former, less secure settings. Hardcoded credentials and vulnerable default configurations have long haunted manufacturers of all sorts of software and hardware across the industry.

A well-run vulnerability management program typically includes identifying security misconfigurations as part of its scope. Many of the same vulnerability scanners and tools that are used to identify and patch security vulnerabilities are also capable of identifying security misconfigurations and providing guidance on how to address them. Again, organizations should forego big investments in advanced cybersecurity capabilities if they aren't already very proficient at identifying and mitigating security misconfigurations in their environment. There's no point in spending a bunch of money and effort looking for the **advanced persistent threat** (**APT**) in an environment if attackers can use decades-old lists of hardcoded passwords, which are available on the internet, to successfully compromise and move around the environment. Even if CISOs found such attackers in their IT environment, they would be powerless to exorcise them with unmanaged common security misconfigurations present.

Some of the biggest breaches in history were a result of an initial compromise through a combination of unpatched vulnerabilities and security misconfigurations. Both can be managed through a well-run vulnerability management program. This is a non-optional discipline in any cybersecurity strategy that should be resourced accordingly. Don't forget, you can't manage what you don't measure; complete, accurate, and timely IT asset inventories are critical for vulnerability management programs. Trust but verify asset inventories, always. It's worth keeping in mind that the cloud provides several advantages over the old on-premises IT world. I'll discuss this in detail in *Chapter 8*, *The Cloud – A Modern Approach to Security and Compliance*, in this book.

Security misconfigurations can be present by default with new hardware and software, or can creep in over time. Another ongoing threat that requires constant attention is that of compromised credentials. Organizations must constantly and proactively work to mitigate this threat vector.

Weak, leaked, and stolen credentials

Compromised IT environments due to weak, leaked, or stolen credentials are common. There are several ways that credentials get leaked and stolen, including social engineering such as phishing, malware that does keystroke logging or steals credentials from operating systems and browsers, and compromised systems that cache, store, and/or process credentials. Sometimes, developers put projects on publicly available code-sharing sites that have secrets such as keys and passwords forgotten in the code. Old development and test environments that are abandoned but still running will ultimately yield credentials to attackers after not being patched over time.

Massive lists of stolen and leaked credentials have been discovered on the internet over the years. In addition to these lists, the availability of high-performance computing clusters and GPU-based password cracking tools have rendered passwords, by themselves, ineffective to protect resources and accounts. Once passwords have been leaked or stolen, they can be potentially leveraged for unauthorized access to systems, in "reuse" attacks and for privilege escalation. The usefulness of passwords, by themselves, to protect enterprise resources has long passed. Subsequently, using **multi-factor authentication** (**MFA**) is a requirement for enterprises and consumers alike. Using MFA can mitigate stolen and leaked credentials in many, but not all, scenarios. Using MFA, even if attackers possess a valid username and password for an account, they won't get access to the account if attackers don't also possess the other factors required for authentication. Other factors that can be used for authentication include digital certificates, one-time passwords and pins generated on dedicated hardware or a smartphone app, a call to a preregistered landline or mobile phone, and more.

MFA isn't a silver bullet for weak, leaked, or stolen passwords, but it's super helpful in many scenarios. There have been some successful attacks on some MFA methods. For example, SIM-swapping attacks to intercept pin codes sent to preregister mobile phones via SMS. Another real limitation of MFA is that it isn't ubiquitous in enterprise IT environments. Organizations with decades of legacy applications that use old-fashioned authentication and authorization methods are less likely to fully mitigate the risk with MFA. Even if the latest systems and cloud-based services require MFA, chances are there are more legacy applications that cannot utilize it easily.

A picture of an iceberg comes to mind here. Several CISOs that I've talked to have experienced this limitation firsthand during penetration tests that exposed the limitations of MFA in their environments. Still, MFA should be widely adopted as it successfully mitigates many attack scenarios where weak, leaked, and stolen passwords are involved. It should be required for new systems being adopted and the risks posed by the old systems without it should be carefully considered and mitigated where possible. There are several vendors that specialize in such mitigations.

When an on-premises enterprise environment is initially compromised, attackers use leaked or stolen credentials to perform reconnaissance and to look for other credentials that have been cached in the environment. They are especially on the lookout for administrator credentials that could give them unlimited access to resources in the compromised environment. Typically, within seconds of the initial compromise, attackers try to access the victim organization's user account directory service, such as Microsoft **Active Directory** (**AD**), to dump all the credentials in the directory. The more credentials they can use to move and stay persistent, the harder it will be to expel them from the environment – they can persist indefinitely. Attackers will try to steal user account databases. If attackers successfully get all the credentials from their directory service, then recovery really is aspirational.

Once attackers have stolen hashed credentials, the weakest of these credentials can be cracked in offline attacks in a matter of hours. The longer, uncommon, and truly complex passwords will get cracked last. There have been raging debates for decades about the efficacy of passwords versus passphrases, as well as appropriate character lengths, character sets, password lockout policies, password expiration policies, and the like. Guidance for passwords has changed over the years as threats and risks have changed and new data has become available. Some of the people I worked with on Microsoft's Identity Protection team published password guidance based on the data from 10 million credential attacks per day that they see on their enterprise and consumer identity systems. "Microsoft Password Guidance" (Hicock, 2016) is recommended reading.

When credentials are leaked or stolen from an organization, it doesn't take attackers long to run them through scripts that try to log in to financial institutions, e-commerce sites, social networking sites, and other sites in the hopes that the credentials were reused somewhere. Reusing passwords across accounts is a terrible practice. Simply put, credentials that provide access to more than one account have a higher ROI for attackers than those that don't. Sets of compromised credentials that can provide access to corporate resources and information, as well as social networks that can also serve as a rich source of information and potential victims, are valuable.

Using unique passwords for every account and using MFA everywhere can mitigate this risk. If you have too many accounts to assign unique passwords to, then use a password vault to make life easier. There are numerous commercially available products for consumers and enterprises.

Identity has always been the hardest part of cybersecurity. Identity governance and management deserves its own book. I offer a very incomplete list of recommendations to help manage the risk of weak, leaked, and stolen credentials:

- MFA can be very effective – use it everywhere you can. Microsoft published a great blog post about the effectiveness of MFA called "Your Pa$$word Doesn't Matter" (Weinert, 2019) that is recommend reading.
- You should know if your organization is leaking credentials and how old those leaked credentials are. Using a service that collects leaked and stolen credentials, and looks for your organization's credentials being sold and traded online, can give you a little peace of mind that you aren't missing something obvious. Getting some idea as to the age of these credentials can help decide if password resets are necessary and the number of people potentially impacted.
- Privileged Access Management solutions can detect pass-the-hash, pass-the-ticket, and Golden Ticket attacks, as well as attackers' lateral movement and reconnaissance in your infrastructure:
    - Many of these solutions also offer password vaulting, credential brokering, and specialized analytics. Some of these solutions can be noisy and prone to false positives, but still, they can help you to manage and detect weak, leaked, and stolen credentials.
- In cloud-based environments, identity and access management (IAM) controls are the most powerful controls you have. Taking advantage of all the power that IAM controls offer can help you to protect and detect resources in the cloud. But this is one control set area that can proliferate into an unmanageable mess quickly. Extra thoughtful planning around your organization's IAM strategy will pay huge security dividends.

I will discuss identity a little more in *Chapter 5*, *Cybersecurity Strategies* of this book.

An important aspect of protecting credentials involves educating information workers within an organization to be aware of social engineering attacks in which attackers may attempt to steal credentials through methods such as phishing. This is not the only way in which social engineering is used to compromise systems, however. We'll cover social engineering in a little more detail next.

Social engineering

Of the cybersecurity usual suspects, social engineering is the most widely used method. Simply put, social engineering is tricking users into making poor trust decisions. Examples of poor trust decisions include lowering the security posture of a system by changing its settings without understanding the possible outcomes of doing so or installing malware on a system. Attackers rely on the naivety of their victims in social engineering attacks.

The volume of social engineering attacks is orders of magnitudes larger than other types of attacks. For example, the volume of email phishing attacks Microsoft reported for July 2019 was 0.85% of the more than 470 billion email messages that flowed through Office 365 that month (Microsoft Corporation, n.d.). That's 4 billion phishing emails that all relied on social engineering, detected in a single month. Similarly, Trojans, a category of malware that relies on social engineering to be successful, has been the most prevalent category of malware in the world continuously for the last decade. I'll discuss this category of malware and many others, in detail, in *Chapter 3*, *The Evolution of the Threat Landscape – Malware*.

Given the massive volume of social engineering attacks, and their historical record of success, mitigating these attacks really isn't optional for enterprises. A fundamental component of an enterprise cybersecurity strategy is a mitigation strategy for social engineering attacks. Put another way, not including social engineering attacks in your cybersecurity strategy would mean ignoring the top way that organizations get initially compromised by volume.

Social engineering attacks are typically perpetrated by attackers external to organizations, to which users must be prepared through appropriate education and training. Another challenging threat to defend against is one from within. The final potential route of compromise, which we'll discuss next, is that of the insider threat.

Insider threats

When discussing insider threats with CISOs and security teams, I find it useful to break them down into three different categories, listed here from most likely to least likely:

1. Users and administrators that make mistakes or poor trust decisions that lead to bad security outcomes.
2. The lone wolf insider or a very small group of individuals that use their privileged access to steal information or otherwise negatively impact the confidentiality, integrity, or availability of the organization's information technology and/or data.

3. The mass conspiracy where multiple insiders work together to overcome the separation of duties that distributes the span of security control. I've found that enterprises typically bring this category up in discussions about risks in managed service provider environments and the cloud.

Mitigating insider threats is an important aspect of cybersecurity and is something that should be fundamental to any enterprise-wide strategy. Enforcing meaningful separation of duties and embracing the principle of least privilege are helpful, as are monitoring and auditing.

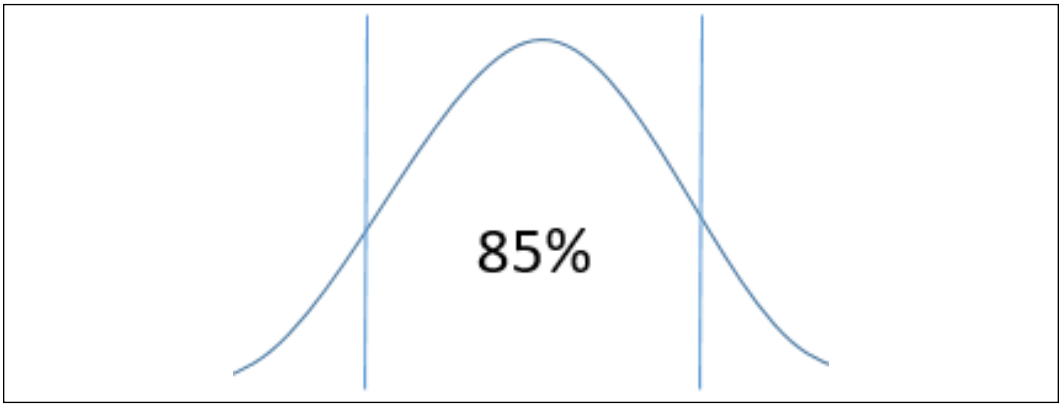
I became a big fan of deception technology after seeing how it can be used to mitigate insider threats. There are a few different approaches to deception technology, but the basic concept is to present attackers with a system, potentially with publicly known vulnerabilities or common security misconfigurations that, when interacted with, alerts defenders to the presence of attackers. This approach can help alert defenders to the presence of external attackers and insider threats. I've heard some security professionals refer to it as a "canary in the coal mine" for IT environments. Implementing deception technology with as few people involved as possible and keeping the program confidential can be helpful in exposing at least two of the three categories of insider threats that I have outlined.

Those are the five ways organizations get initially compromised. Defending against these five vectors of attack is fundamental to effective cybersecurity.

Focus on the cybersecurity fundamentals

To have a successful cybersecurity program, organizations need to get very good at continuously mitigating all five of these types of threats. This competency forms the foundation of a sound cybersecurity strategy. Other cybersecurity-related investments will potentially have diminishing returns if the foundation of the strategy is not solid.

After an attacker uses one or more of these five ways to initially compromise an organization, then they might employ a plethora of novel and advanced TTPs. Organizations that focus on the cybersecurity fundamentals make it much harder for attackers to be successful; that is, by focusing on the inside 85% of the bell curve below which the cybersecurity fundamentals sit, instead of the activities in the outlying 7.5% on either end of the curve, security teams will be much more successful. Unfortunately, the allure of hunting advanced persistent threats can take resources away from the less sexy, but critical, work in the middle of the curve.



A bell curve illustrating that most security teams should spend their time on the cybersecurity fundamentals

If there really are only five ways that organizations get initially compromised, why does there seem to be so much confusion in the industry on proper priorities for cybersecurity programs? I think there are a bunch of factors contributing to the confusion. One reason for the confusion is the way that attacks, security incidents, and data breaches have been reported in popular media outlets sometimes confuses attackers' tactics with their motivations. This can lead organizations to make the wrong security prioritization decisions.

Understanding the difference between the attacker's motivations and tactics

One of the reasons I've found so many organizations lack focus and competency around the cybersecurity fundamentals is the way big data breaches have been reported in the news over the last decade. Stories that claim an attack was the "most advanced attack seen to date" or the work of "a nation state" seem to be common. But when you take a closer look at these attacks, the victim organization was always initially compromised by attackers using one or more of the five ways I outlined in this chapter.

There are attackers that operate in the open because they don't believe there are consequences for their illicit activities, based on their location and legal jurisdiction. But this is the exception to the rule that they will obfuscate their true personal identities. Claims that an attack was the work of a nation state or an APT group are typically based on circumstantial evidence. Rapidly changing networks of social media accounts and news outlets spreading false information exasperate the challenge of attribution.

Attributing an attack to an individual or group can be extremely hard. This is because the internet is based on a suite of protocols that was developed over 35 years ago.

The engineers that developed these immensely scalable and sophisticated protocols never envisioned a future world where an entire multi-billion-dollar-a-year industry would be based on the discoveries of new security vulnerabilities, malware research, social engineering protection, and nation state actors. TCP/IP version 4, the basis of the internet, was never designed to help investigators perform attribution for attacks that leverage vast networks of compromised distributed systems around the world. Comparing code fragments from two malware samples to determine if the same attackers developed both is not a reliable way to perform attribution, especially when the attackers know this is a common technique. Finding "patient zero," where the compromise started, in large environments that have been compromised for months or years, using data from compromised systems, can't be done with complete confidence.

But still, many cybersecurity professionals use this type of data to surmise the attackers' motivations and identities. Attacker motivations include:

- **Notoriety**: The attacker wants to prove they are smarter than the big high-tech companies and their victims.
- **Profit**: As I'll discuss in *Chapter 3*, *The Evolution of the Threat Landscape – Malware*, after the successful worm attacks in 2003, malware began to evolve to support a profit motive that continues to the present day.
- **Economic espionage**: For example, alleged activities by groups in China to steal valuable intellectual property from western nations to give their own industries a competitive and economic advantage.
- **Military espionage**: A motivation as old as governments themselves, where governments want to understand the military capabilities of their adversaries.
- **Hacktavism**: Attacks against organizations and institutions based on disagreements on political or philosophical issues.
- **Influencing elections**: Using cultural manipulation and information warfare to help nations achieve foreign policy objectives.
- **Many others**: Watch any James Bond movie where the **Special Executive for Counterintelligence, Terrorism, Revenge, and Extortion** (**SPECTRE**) is part of the plot.

If most organizations can't really know who is attacking them, then they can't really understand what the attacker's motivation is. If CISOs don't know what's motivating the attacker, how do they know what a proportional response is? Who should help the victim organization with the response to the attack – local authorities, the military, an international coalition?

Still, I have talked to organizations whose cybersecurity strategies rely heavily on attribution. After performing hundreds of incident response investigations for Microsoft's customers, I find the assumption that timely attribution can be done with any confidence to be overly optimistic. For most organizations, relying on accurate attribution to inform their cybersecurity strategy or to help make incident response decisions is pure fantasy. But I believe you can, with 99.9% certainty, predict the tactics the attackers will use when they try to initially compromise an IT environment. This is what organizations should invest in – the cybersecurity fundamentals.

Having a cybersecurity strategy is a great step in the right direction. But by itself, it represents good intentions, not a commitment by the organization. In the next section, we'll take a look at what else needs to be done in order to successfully implement an effective cybersecurity strategy.

Other ingredients for a successful strategy

There is a bunch of management-related work that needs to be done to ensure the CISO, the security team, and the rest of the organization can effectively execute a cybersecurity strategy. This section outlines some of the ingredients that give a strategy the best chance of success.

CISOs that tell the businesses they support, "No, you can't do that," are no longer in high demand. Security teams must align with their organizations' business objectives, or they won't be successful.

Business objective alignment

I've met many CISOs that were struggling in their roles. Some of them simply weren't properly supported by their organizations. It's easy to find groups of executives that think cybersecurity threats are overblown and everything their CISO does is a tax on what they are trying to accomplish. To these folks, cybersecurity is just another initiative that should stand in line behind them for resources. After all, the company won't get to that next big revenue milestone via a cost center, right?

Working with executives that don't understand the cybersecurity threats their organization faces and really don't have the time to pay attention isn't uncommon. Most CISOs must work with other executives to get things done, even if those executives don't realize they have a shared destiny with the CISO; when the CISO fails, they all fail. But the best CISOs I've met tend to thrive in such environments.

Whether a CISO works in an environment like the one I described, or they are lucky enough to work with people that care if they are successful, to be successful, CISOs need to align with the business they support. CISOs that don't understand and embrace the objectives of the organizations they support generate friction. There is only so much friction senior leaders are willing to tolerate before they demand change. Deeply understanding the business and how it works gives enlightened CISOs the knowledge and credibility required to truly support their organizations. Put another way, "purist" CISOs that try to protect data in isolation of the people, business processes, and technologies that their organization relies on to succeed are only doing part of the job they were hired to do.

A cybersecurity strategy will only be successful if it truly supports the business. Developing a strategy that helps mitigate the risks that the security team cares most about might give the team the satisfaction that they have a buttoned-up plan that will make it difficult for attackers to be successful. But if that strategy also makes it difficult for the business to be competitive and agile, then the security team must do better.

The best way to prove to your C-suite peers that you are there to help them is to learn about the parts of the business they manage, what their priorities are, and earn their trust. None of this is going to happen in your **security operations center** (**SOC**), so you are going to have to spend time in their world, whether that's on a factory floor, in a warehouse, on a truck, or in an office. Walk a mile in their shoes and they'll have an easier time following your counsel and advocating for you when it's important.

Lastly, remember it's the CISO's job to communicate, manage, and mitigate risk to the business, not to decide what the organization's risk appetite is. The board of directors and senior management have been managing risk for the organization since it was founded. They've been managing all sorts of risks including financial risks, economic risks, HR risks, legal risks, and many others. Cybersecurity risks might be the newest type of risk they've been forced to manage, but if the CISO can learn to communicate cybersecurity risks in the same way that the other parts of the business do, the business will do the right thing for their customers and shareholders or they will pay the price – but that's the business' decision, not the CISO's.

That said, accountability, liability, and empowerment go hand-in-hand. Many CISOs face the harsh reality that they are made accountable for mitigating risks accepted by the business, but are not empowered to make the necessary changes or implement countermeasures. Simply put, a CISO's job is a hard one. This might help explain why CISO tenures are typically so short compared to those of other executives.

Having a clear and shared vision on where cybersecurity fits into an organization's wider business strategy is not only important within the upper echelons of an organization; the organization as a whole should have a clear stance on their vision, mission, and imperatives for their cybersecurity program. We'll take a look at this next.

Cybersecurity vision, mission, and imperatives

Taking the time to develop and document a vision, mission statement, and imperatives for the cybersecurity program can be helpful to CISOs. A shared vision that communicates what the future optimal state looks like for the organization from a cybersecurity perspective can be a powerful tool to develop a supportive corporate culture. It can inspire confidence in the cybersecurity team and the future of the organization. It can also generate excitement and goodwill toward the security team that will be helpful in the course of their work.

Similarly, a well-written mission statement can become a positive cultural mantra for organizations. A good mission statement can communicate what the security team is trying to accomplish while simultaneously demonstrating how the security team is aligned with the business, its customers, and shareholders. The mission statement will help communicate the security team's objectives as it meets and works with other parts of the organization.

Finally, business imperatives are the major goals that the cybersecurity team will undertake over a 2- or 3-year period. These goals should be ambitious enough that they can't be achieved in a single fiscal year. Imperatives support the strategy and are aligned with the broader business objectives. When the strategy isn't aligned with broader business objectives, this can show up as an imperative that is out of place – a square peg in a round hole. Why would be the business support a big multi-year goal that isn't aligned with its objectives? This should be a message to the CISO to realign the strategy and rethink the imperatives. These multi-year goals become the basis for the projects that the cybersecurity group embarks on. An imperative might be accomplished by a single project or might require multiple projects. Remember a project has a defined start date, end date, and budget.

Don't confuse this with a program that doesn't necessarily have an end date and could be funded perpetually. Programs can and should contribute to the group's imperatives.

Developing a vision, mission statement, and imperatives for the cybersecurity program isn't always easy or straightforward. The vision cannot be actioned without the support of stakeholders outside of the cybersecurity group, and convincing them of the value of the program can be time-consuming. The future rewards from this work, for the CISO and the cybersecurity group as a whole, typically make the effort worthwhile. We'll briefly discuss securing this support next, as one of our important ingredients to a successful cybersecurity strategy.

Senior executive and board support

Ensuring that the senior executives and the board of directors understand and support the organization's cybersecurity strategy is an important step for a successful security program. If the senior executives understand the strategy and had a hand in developing it and approved it, they should show more ownership and support it moving forward. But if they don't have a connection to the strategy, then the activities that are executed to support it will be potentially disruptive and unwelcome. They won't understand why changes are being made or why the governance model behaves the way it does.

Two of the important questions CISOs should ask when they are interviewing for a new CISO job is who the role reports to and how often the CISO will be meeting with the board of directors or the Board Audit Committee? If the CISO isn't meeting with the board quarterly or twice per year, that's a red flag. It might be that the role that the CISO reports to, meets with the board instead. But unless that role is steeped in the strategy and the daily operations, they should be sharing or delegating the job of meeting with the board to the CISO. This gives the CISO firsthand experience of discussing priorities with the board. It also allows board members to get their updates directly from the CISO and ask them their questions directly. I'd be very hesitant to take a CISO job where the role didn't meet directly with the board at least a couple of times per year.

This experience is important and demonstrates that the CISO is a legitimate member of the organization's C-suite. If the CISO doesn't have the opportunity to ask the board for help with their peers, including the CEO, that's one more reason their peers don't really need to support them. Adding a management layer between the CISO and board can be a tactic that senior management uses to delay, influence, or deter the CISO from making progress with their security program. It can also provide shelter to CISOs that don't have the business acumen or corporate maturity to interact directly with the board.

But if the executive management team is truly supportive of the CISO and the cybersecurity strategy, they should welcome the opportunity for the CISO to get the help they need as quickly as possible without instituting more bureaucracy. Besides, the executive team should already know what the CISO is going to tell the board if they are taking their responsibilities seriously. Of course, history has taught us that this is not always the case where cybersecurity is concerned.

If the CISO is successful at getting the board on board with the cybersecurity strategy, this will make it easier for the board to understand why the security team is doing what they are doing. It will also make it easier for the CISO to elicit help when needed and report results against the strategy. I don't claim this is an easy thing to do. The first couple of times I met with boards of directors was like meeting the characters in an Agatha Christie novel or from the game of Clue. The board members I've met have all been very accomplished professionally. Some are humble about their accomplishments, while others assert their accomplishments to influence others. There always seems to be at least one board member who claims to have cybersecurity experience, who wants to ask tough questions, and give the CISO advice on cybersecurity. But if the CISO can effectively communicate a data-driven view of results against the cybersecurity strategy, the same strategy that the board approved, these conversations can be very helpful for all stakeholders. Additionally, results from internal and external audits typically provide boards with some confidence that the CISO is doing their job effectively.

After talking with executives at literally thousands of organizations around the world about cybersecurity, I can tell you that there are real differences in how much risk organizations are willing to accept. In addition to gaining support from senior executives and the board, it is important to have a good understanding of their appetite for risk, as we'll discuss next, since this could significantly impact cybersecurity strategy.

Understand the risk appetite

Some organizations are in hypercompetitive industries where innovation, speed, and agility are top priorities; these organizations tend to be willing to accept more risk when faced with security and compliance decisions that will potentially slow them down or otherwise impede their ability to compete. For these companies, if they don't take calculated risks, they won't be in business long enough to make decisions in the future. Other organizations I've talked to are very risk-averse. That doesn't mean they necessarily move slowly, but they demand more certainty when making decisions.

They are willing to take the time to really understand factors and nuances in risk-based decisions in an effort to make the best possible decision for their organization. Of course, there are also organizations in the spectrum between these two examples.

CISOs that understand the risk appetite of the senior management in their organizations can help them make faster, better decisions. I've seen many CISOs over the years decide to play the role of "the adult in the room" and try to dictate how much risk the organization should accept. In most cases, this isn't the CISO's job. Providing context and data to help the business make informed risk-based decisions is a function CISOs should provide. Sometimes, they also have to educate executives and board members who do not understand cybersecurity risks. But I find it useful to always keep in mind that, in established organizations, executive suites were managing many types of risks for the organization long before cybersecurity risks became relevant to them. Note, this could be different for start-ups or in organizations where the CISO also has deep expertise in the business they support; in these scenarios, the CISO might be expected to make risk decisions for the organization more directly. But in all cases, understanding how much risk the organization is willing to accept in the normal course of business is important for CISOs.

The organization's appetite for risk will show up in their governance model and governance practices. In many cases, organizations that accept more risk in order to move faster will streamline their governance practices to minimize friction and blockages. Organizations that want to take a meticulous approach to decision making will typically implement more governance controls to ensure decisions travel fully through the appropriate processes. For this reason, it's important that CISOs validate their understanding of their organizations' risk appetite instead of making assumptions about it. This is where their knowledge of the business and their peers' priorities will help.

In addition to a knowledge of business priorities, it's important to have a realistic idea of the organization's current capabilities and technical talent. We'll discuss that next.

Realistic view of current cybersecurity capabilities and technical talent

Many of the CISOs I know aspire to have a world-class cybersecurity team designing, implementing, and operating sophisticated and effective controls. However, being honest with themselves about their current state of affairs is the best starting point.

The entire industry has been suffering from an acute shortage of cybersecurity talent for over a decade. This problem is getting worse as more and more organizations come to the realization that they need to take cybersecurity seriously or suffer potential non-compliance penalties and negative reputational consequences. Assessing the talent that a security team currently has helps CISOs, as well as CIOs, identify critical gaps in expertise. For example, if a security team is understaffed in a critical area such as vulnerability management or incident response, CIOs and CISOs need to know this sooner than rather than later. If you have people that are untrained on some of the hardware, software, or processes that they are responsible for or are expected to use, identifying those gaps is the first step in addressing them. It also helps CIOs and CISOs identify professional growth areas for the people on the security team and spot potential future leaders. Cross-pollinating staff across teams or functions will help develop them in ways that will potentially be useful in the future.

The key is for CIOs and CISOs to be as realistic in their assessments as they can be so that they have a grounded view of the talent in the organization. Don't let aspirations of greatness paint an inaccurate picture of the talent the organization has. This will make it easier to prioritize the type of talent required and give the organization's recruiters a better chance of attracting the right new talent.

Cartography, or doing an inventory of your current cybersecurity capabilities, is another important exercise. The results will inform the development of the cybersecurity imperatives that I discussed earlier, as well as helping to identify critical gaps in capabilities. It can also help identify over-investment in capabilities. For example, it's discovered that the organizations procured three identity management systems and only one of them is actually deployed. This is occurring while the organization doesn't have enough vulnerability scanners to do a competent job of scanning and patching the infrastructure in a reasonable amount of time.

In most big, complex IT environments, this won't be an easy task. It might turn out to be relatively easy to get a list of entitlements from the procurement department or a deployed software inventory from IT. But knowing that a particular appliance, piece of software, or suite of capabilities has been deployed only answers part of the question the CISO needs answered. Really understanding the maturity of the deployment and operation of those capabilities is just as important but is typically much harder to determine. Just because an identity management product is in production doesn't mean all of its capabilities have been implemented or enabled, that the product is being actively managed, and the data it produces is being consumed by anyone.

Discovering these details can be challenging, and measuring their impact on your strategy might be too difficult to realistically contemplate. But without these details, you might not be able to accurately identify gaps in protection, detection, and response capabilities, and areas where over-investment has occurred.

If CIOs and CISOs can get an accurate view of the current cybersecurity talent and capabilities they have, it makes it much easier and less expensive for them to effectively manage cybersecurity programs for their organizations.

In my experience, there can be a lot of conflict and friction in organizations when cybersecurity teams and compliance teams do not work well together. Let's explore this dynamic next.

Compliance program and control framework alignment

I've seen cybersecurity and compliance teams conflict with one another over control frameworks and configurations. When this happens, there tends to be a disconnect between the cybersecurity strategy and the compliance strategy within the organization. For example, the CISO might decide that the cybersecurity team is going to embrace ISO as a control framework that they measure themselves against. If the compliance team is measuring compliance with NIST standards, this can result in conversation after conversation about control frameworks and configurations. Some organizations work out these differences quickly and efficiently, while other organizations struggle to harmonize these efforts.

A common area for misalignment between cybersecurity and compliance teams is when controls in an internal standard and an industry standard differ. Internal standards are typically informed by the specific risks and controls that are most applicable to each organization. But differences between an internal standard and an industry standard can happen when the internal standard is newer than the industry standard or vice versa. For example, the industry standard states that an account lockout policy must be set to a maximum of 5 incorrect password entries. But the cybersecurity team knows that this control is "security theatre" in an environment that enforces a strong password policy and especially on systems that have MFA enabled. But in order to meet the industry standard, they might be forced to turn on the account lockout policy, thus enabling attackers to lock accounts out any time they want to with a denial of service attack.

I've seen compliance professionals argue with CISOs on the efficacy of such dated control standards, who are simply trying to successfully comply with an industry standard without considering that they are actually increasing risk for the entire organization. I've even seen some of these compliance professionals, in the course of their work, claim that they can accept risk on behalf of the entire organization where such decisions are concerned – which is rarely, if ever, the case.

It should be recognized and acknowledged that both compliance and security are important to organizations. Compliance is driven by the regulation of liability, and security is driven by prevention, detection, and response. CISOs should foster normalization and the alignment of applied frameworks for security and compliance. Compliance professionals need to recognize that any organization that places compliance as a higher priority will eventually be compromised.

The cybersecurity group and the compliance group should work together to find ways that they can meet standards while also protecting, detecting, and responding to modern-day threats. These different, but overlapping, disciplines should be coordinated with the common goal of helping to manage risk for the organization. As I mentioned earlier, the cybersecurity strategy should be informed by the organization's high-value assets and the specific risks they care about. The compliance team is the second line of defense designed to ensure the cybersecurity team is doing their job effectively by comparing their controls against internal, industry, and/or regulated standards. But they need to be prepared to assess the efficacy of controls where there are differences or where they conflict, instead of blindly demanding a standard be adhered to.

Typically, the decision to accept more risk by meeting a dated industry standard, for example, should be made by a risk management board or broader internal stakeholder community instead of by a single individual or group. Internal and external audit teams are the third line of defense that help to keep both the cybersecurity team and the compliance team honest by auditing the results of their work. No one wins when these teams fight over control frameworks and standards, especially when the frameworks or standards in question are based on someone else's threat model, as is almost always the case with industry and regulated standards.

Some organizations try to solve this problem by making the CISO report to the compliance organization. I always feel sorry for CISOs that I meet that report to compliance or audit leadership. This isn't a criticism of compliance or audit professionals or leadership in any way. Simply put, cybersecurity and compliance are different disciplines.

Compliance focuses on demonstrating that the organization is successfully meeting internal, industry, and/or regulated standards. Cybersecurity focuses on protecting, detecting, and responding to modern-day cybersecurity threats. Together, they help the organization manage risk. I'm going to discuss "compliance as a cybersecurity strategy," in detail, in in *Chapter 5*, *Cybersecurity Strategies*. Next, however, we'll talk about the importance of cybersecurity and IT maintaining a happy and productive relationship with one another.

An effective relationship between cybersecurity and IT

In my experience, CISOs that have a good working relationship with their business' IT organization are typically happier and more effective in their job. An ineffective relationship with IT can make a CISO's life miserable. It's also true that CISOs can make the jobs of CIOs and VPs of IT disciplines frustrating. I've met so many CISOs that have suboptimal working relationships with their organization's IT departments. I've seen many cybersecurity groups and IT organizations interact like oil and water, when the only way to be successful is to work together. After all, they have a shared destiny. So, what's the problem? Well, simply put, in many cases, change is hard. It is easy for CIOs to interpret the rise of CISOs as a by-product of their own shortcomings, whether this is accurate or not. CISOs represent change and many of them are change leaders.

Moreover, I think this dynamic can develop for at least a few reasons. The way that these groups are organized can be one of them. The two most common ways I've seen cybersecurity groups integrated, who are typically newer than IT organizations in large, mature organizations, are as follows:

- The CISO reports to IT and shares IT resources to get work done.
- The CISO reports outside of IT, to the CEO, the board of directors, legal, compliance, or the CFO. There are two flavors of this model:
    1. The CISO has their own cybersecurity resources, but needs IT resources to get work done.
    2. The CISO has their own cybersecurity and IT resources and can get work done independently of IT.

The scenario where the CISO reports into the IT organization, historically, has been very common. But this reporting line has been evolving over time. Today, I estimate that less than 50% of the CISOs I meet report into IT. One of the reasons for this change in reporting lines is that, all too often, CIOs prioritize IT priorities over cybersecurity.

Cybersecurity is treated like any other IT project in that it must queue up with other IT projects and compete with them for resources to get things done. Frustrated CISOs would either be successful in convincing their boss that cybersecurity wasn't just another IT project, or they were forced to escalate. There are no winners with such escalations, least of all the CISO. In many cases, the CISO gets left with a CIO that resents them and sees them as a tax on the IT organization.

It took years for many CIOs to realize that every IT project has security requirements. Deprioritizing or slowing down cybersecurity initiatives means that every IT project that has a dependency on cybersecurity capabilities will either be delayed or will need an exception to sidestep these requirements. The latter tends to be much more common than the former. When CEOs and other executives began losing their jobs and directors on boards were being held accountable because of data breaches, many organizations were counseled by outside consultants to have their CISOs report to the CEO or directly to the board of directors. This way, cybersecurity would not be deprioritized without the most senior people being involved in making those risk decisions.

A new challenge is introduced in the scenario where the CISO reports outside of IT to the CEO, the board of directors, or another part of the company. Where is the CISO going to get the IT staff required to get things done? When the CISO reported into IT, it was easier to get access to IT resources, even if they had to queue up. For CISOs that sit outside the IT organization, they only have a few options. They can get resources from IT and become their customer, or they must hire their own IT resources. Becoming a customer of IT sounds like it could make things easier for CISOs, but only when they have a good relationship with IT that leads to positive results. Otherwise, it might not be sufficiently different from the model where the CISO reports into IT. As expedient as hiring their own resources sounds, there are challenges with this approach. For example, change control can become more complex because IT isn't the only group of people that can make changes in the environment. Many times, this results in IT engineers watching cybersecurity engineers making changes in their shared environment and vice versa. Using twice as many resources to ensure things get done in a timely manner is one way to approach this problem. But most organizations can find better uses for their resources.

I've seen a better approach in action. When CISOs, CIOs, and CTOs have mutual respect for each other's charter and support each other, the work is easier, and things get done more efficiently.

Instead of a relationship defined by resource contention or assertions of authority, CISOs need to have good, effective working relationships with their IT departments to ensure they can do their jobs. Building such relationships isn't always easy, or even possible, but I believe this is a critical ingredient for a successful cybersecurity strategy. Ideally, these relationships blossom into a security culture that the entire organization benefits from.

On the topic of culture, the last ingredient for a successful cybersecurity strategy is a strong security culture. This culture involves everybody in the organization understanding their role in helping to maintain a good security posture to protect the organization from compromise. Let's talk about it in a little more detail in the next and final section of this chapter.

Security culture

Someone famous recently said, "Culture eats strategy for breakfast." I agree wholeheartedly. Organizations that are successful in integrating security into their corporate culture are in a much better position to protect, detect, and respond to modern-day threats. For example, when everyone in the organization understands what a social engineering attack looks like and is on the lookout for such attacks, it makes the cybersecurity team's job much easier and gives them a greater chance of success. Contrast this with work environments where employees are constantly getting successfully phished and vulnerabilities are constantly being exploited because employees are double-clicking on attachments in emails from unknown senders. In these environments, the cybersecurity team is spending a lot of their time and effort reacting to threats that have been realized. A strong security culture helps reduce exposure to threats, decrease detection and response times, and thus reduce the associated damage and costs.

Culture transcends training. It's one thing for employees to receive one-time or annual security training for compliance purposes, but is quite another thing for the concepts and calls to action that employees learn in training to be constantly sustained and reinforced by all employees and the work environment itself. This shouldn't be limited to front-line information workers. Developers, operations staff, and IT infrastructure staff all benefit from a culture where security is included. A security culture can help employees make better decisions in the absence of governance or clear guidance.

One note on the gamification of cybersecurity training: I've seen good results when organizations shift some of their cybersecurity training away from reading and videos into more interactive experiences.

I've facilitated "game days" focused on helping organizations learn about threat modeling and cloud security. To be completely honest, I was more than a little skeptical about using this approach. But I've seen many groups of executives and security teams embrace it and provide glowing feedback that now I'm a big fan of gamification for training purposes.

CISOs have a better chance of success when everyone in their organizations helps them. I encourage CISOs, with the help of other executives, to invest some of their time in fostering a security culture, as it will most certainly pay dividends.

Chapter summary

I covered a lot of ground in this chapter. But the context I provided here will be helpful for readers throughout the rest of this book. In this chapter, I introduced the cybersecurity fundamentals, the cybersecurity usual suspects, High Value Assets (HVAs), and other concepts, that I will relentlessly refer to throughout the rest of this book.

What is a cybersecurity strategy? There are at least two critical inputs to a cybersecurity strategy: your organization's HVAs, and the specific requirements, threats, and risks that apply to your organization, informed by the industry you are in, the place(s) in the world where you do business, and the people associated with the organization. If an HVA's confidentiality, integrity, or availability is compromised, the organization will fail or be severely disrupted. Therefore, identifying HVAs and prioritizing protection, detection, and response for them is critical. This does not give security teams permission to completely ignore other assets. Clarity on HVAs helps security teams prioritize, and to avoid extinction events.

There are only five ways that organizations get initially compromised, I call them the cybersecurity usual suspects. They include, unpatched vulnerabilities, security misconfigurations, weak, leaked, and stolen credentials, social engineering, and insider threat. Organizations that are very proficient at managing the cybersecurity fundamentals, make it much harder for attackers to be successful. After the initial compromise of an IT environment, there are many tactics, techniques, and procedures (TTPs) that attackers can use to achieve their illicit goals. Advanced cybersecurity capabilities can help security teams detect the use of TTPs and reduce response and recovery times. Don't confuse an attacker's motivations with their tactics. Since accurate attribution for attacks is so difficult to accomplish, it's unlikely most organizations will be able to determine who is attacking them and what their motivation is.

Whether the attacker is a purveyor of commodity malware or a nation state, the ways they will try to initially compromise their victims' IT environments are limited to the cybersecurity usual suspects. Being very proficient at the cybersecurity fundamentals makes it much harder for attackers, whether they are a nation state trying to steal intellectual property or an extortionist.

A cybersecurity strategy is required for success, but it is not sufficient by itself. Ingredients for a successful strategy include:

- Business objective alignment
- Cybersecurity vision, mission, and imperatives
- Senior executive and board support
- Understand the organization's risk appetite
- A realistic view of current cybersecurity capabilities and technical talent
- Compliance program and control framework alignment
- An effective relationship between cybersecurity and IT
- Security culture

Now that all this context has been introduced, I'll build on it in the chapters that follow. In the next few chapters, I'll explore how the threat landscape has evolved. I believe that CISOs can make better decisions when they understand how threats have changed over time. The three categories of threats that I'll dive into are the ones that CISOs have asked me about most frequently: vulnerabilities, malware, and internet-based threats like phishing and drive-by download attacks.

References

1. Hicock, R. (2016). *Microsoft Password Guidance*. Retrieved from Microsoft Corporation Web site: `https://www.microsoft.com/en-us/research/wp-content/uploads/2016/06/Microsoft_Password_Guidance-1.pdf`
2. Microsoft Corporation. (2007-2016). Microsoft Security Intelligence Report. Retrieved from `www.microsoft.com/sir`
3. Microsoft Corporation. (n.d.). Microsoft Security Intelligence Report. Retrieved from `https://www.microsoft.com/securityinsights/Phishing`
4. National Vulnerability Database. (n.d.). Retrieved from `https://nvd.nist.gov/vuln`
5. Weinert, A. (July 9, 2019). *Your Pa$$word doesn't matter*. Retrieved from `https://techcommunity.microsoft.com/t5/azure-active-directory-identity/your-pa-word-doesn-t-matter/ba-p/731984`

2

Using Vulnerability Trends to Reduce Risk and Costs

Vulnerabilities represent risk and expense to all organizations. Vendors who are serious about reducing both risk and costs for their customers focus on reducing the number of vulnerabilities in their products and work on ways to make it hard and expensive for attackers to exploit their customers, thereby driving down attackers' return on investment. Identifying the vendors and the products that have been successful in doing this can be time-consuming and difficult.

In this chapter, I will provide you with valuable background information and an in-depth analysis of how some of the industry's leaders have managed vulnerabilities in their products over the last two decades, focusing on operating systems and web browsers. I introduce a vulnerability improvement framework that can help you to identify vendors and products that have been reducing risks and costs for their customers. This data and analysis can inform your vulnerability management strategy.

Throughout this chapter, we'll cover the following topics:

- A primer on vulnerability management
- Introducing a vulnerability management improvement framework
- Examining vulnerability disclosure trends for select vendors, operating systems, and web browsers
- Guidance on vulnerability management programs

Let's begin by looking at what vulnerability management is.

Introduction

Over the past 20 years, organizations have been challenged to manage a continual volume of new vulnerabilities in software and hardware. Attackers and malware constantly attempt to exploit unpatched vulnerabilities on systems in every industry and in every part of the world. Vulnerabilities are currency for many interested groups, including security researchers, the vulnerability management industry, governments, various commercial organizations, and, of course, attackers and purveyors of malware. These groups have different motivations and goals, but they all value new vulnerabilities, with some willing to pay handsomely for them.

I had a front row seat at ground zero for the tumultuous period where worms and other malware first started exploiting vulnerabilities in Microsoft software at scale. After working on the enterprise network support team at Microsoft for a few years, I was asked to help build a new customer-facing security incident response team. I accepted that job on Thursday, January 23, 2003. Two days later, on Saturday, January 25th, SQL Slammer hit the internet, disrupting networks worldwide. That Saturday morning, I got into my car to drive to the office but had to stop for gas. Both the cash machine and the pumps at the gas station were offline due to "network issues". At that point, I realized just how widespread and serious that attack was. Then, one day in August 2003, MSBlaster disrupted the internet to an even greater extent than SQL Slammer had. Then, over the course of the following year, MSBlaster variants followed, as well as MyDoom, Sasser, and other widespread malware attacks. It turns out that *millions* of people were willing to double-click on an email attachment labeled "MyDoom".

Most of these attacks used unpatched vulnerabilities in Microsoft products to infect systems and propagate. This all happened before Windows Update existed, or any of the tools that are available today for servicing software. Because Microsoft had to release multiple security updates to address the underlying vulnerabilities in the components that MSBlaster used, many IT departments began a long-term pattern of behavior, delaying patching systems to avoid patching the same components repeatedly and rebooting systems repeatedly. Most internet connected Windows-based systems were not running anti-virus software in those days either, and many of those that did, did not have the latest signatures installed. Working on a customer-facing security incident response team, supporting security updates, and helping enterprise customers with malware infections and hackers, was a very tough job in those days—you needed thick skin. Subsequently, I learned a lot about malware, vulnerabilities, and exploits in this role.

Later in my career at Microsoft, I managed marketing communications for the **Microsoft Security Response Center** (**MSRC**), the Microsoft **Security Development Lifecycle** (**SDL**), and the **Microsoft Malware Protection Center** (**MMPC**). The MSRC is the group at Microsoft that manages the incoming vulnerability reports and attack reports. The MMPC is what they called Microsoft's anti-virus research and response lab back then. The SDL is a development methodology that was instituted at Microsoft in the years that followed these devastating worm attacks. I learned a lot about vulnerabilities, exploits, malware, and attackers in the 8 or 9 years I worked in this organization, called Trustworthy Computing.

I often get asked if things are better today than they were 5 or 10 years ago. This chapter is dedicated to answering this question and providing some insights into how things have changed from a vulnerability management perspective. I also want to provide you with a way to identify vendors and products that have been reducing risk and costs for their customers.

Vulnerability Management Primer

Before we dive into the vulnerability disclosure trends for the past couple of decades, let me provide you with a quick primer on vulnerability management so that it's easier to understand the data and analysis I provide, and how some vulnerability management teams use such data.

The **National Vulnerability Database** (**NVD**) is used to track publicly disclosed vulnerabilities in all sorts of software and hardware products across the entire industry. The NVD is a publicly available database that can be accessed at `https://nvd.nist.gov`.

In this context, a vulnerability is defined as:

> "*A weakness in the computational logic (e.g., code) found in software and hardware components that, when exploited, results in a negative impact on confidentiality, integrity, or availability. Mitigation of the vulnerabilities in this context typically involves coding changes, but could also include specification changes or even specification deprecations (e.g., the removal of affected protocols or functionality in their entirety).*"
>
> *– (NIST, n.d.)*

When a vulnerability is discovered in a software or hardware product and reported to the vendor that owns the vulnerable product or service, the vulnerability will ultimately be assigned a **Common Vulnerability and Exposures** (**CVE**) identifier at some point.

The exact date when a CVE identifier is assigned to a vulnerability is a function of many different factors, to which an entire chapter in this book could be dedicated. In fact, I co-wrote a Microsoft white paper on this topic called *Software Vulnerability Management at Microsoft*, which described why it could take a relatively long time to release security updates for Microsoft products. It appears that this paper has disappeared from the Microsoft Download Center with the sands of time. However, the following are some of the factors explaining why it can take a long time between a vendor receiving a report of a vulnerability and releasing a security update for it:

- **Identifying the bug**: Some bugs only show up under special conditions or in the largest IT environments. It can take time for the vendor to reproduce the bug and triage it. Additionally, the reported vulnerability might exist in other products and services that use the same or similar components. All of these products and services need to be fixed simultaneously so that the vendor doesn't inadvertently produce a zero-day vulnerability in its own product line. I'll discuss zero-day vulnerabilities later in this chapter.
- **Identifying all variants**: Fixing the reported bug might be straightforward and easy. However, finding all the variations of the issue and fixing them too is important as it will prevent the need to re-release security updates or to release multiple updates to address vulnerabilities in the same component. This can be the activity that takes the most time when fixing vulnerabilities.
- **Code reviews**: Making sure the updated code actually fixes the vulnerability and doesn't introduce more bugs and vulnerabilities is important and sometimes time-consuming.
- **Functional testing**: This ensures that the fix doesn't impact the functionality of the product—customers don't appreciate it when this happens.
- **Application compatibility testing**: In the case of an operating system or web browser, vendors might need to test thousands of applications, drivers, and other components to ensure they don't break their ecosystem when they release the security update. For example, the integration testing matrix for Windows is huge, including thousands of the most common applications that run on the platform.
- **Release testing**: Make sure the distribution and installation of the security update works as expected and doesn't make systems unbootable or unstable.

It is important to realize that the date that a CVE identifier is assigned to a vulnerability isn't necessarily related to the date that the vendor releases an update that addresses the underlying vulnerability; that is, these dates can be different. The allure of notoriety that comes with announcing the discovery of a new vulnerability leads some security researchers to release details publicly before vendors can fix the flaws. The typical best-case scenario is when the public disclosure of a vulnerability occurs on the same date that the vendor releases a security update that addresses the vulnerability. This reduces the window of opportunity for attackers to exploit the vulnerability to the time it takes organizations to test and deploy the update in their IT environments.

An example of a CVE identifier is CVE-2018-8653. As you can tell from the CVE identifier, the number 8653 was assigned to the vulnerability it was associated with in 2018. When we look up this CVE identifier in the NVD, we can get access to a lot more detail about the vulnerability it's associated with. For example, some details include the type of vulnerability, the date the CVE was published, the date the CVE was last updated, the severity score for the vulnerability, whether the vulnerability can be accessed remotely, and its potential impact on confidentiality, integrity, and availability.

> *It might also contain a summary description of the vulnerability, like this example: "A remote code execution vulnerability exists in the way that the scripting engine handles objects in memory in Internet Explorer, aka "Scripting Engine Memory Corruption Vulnerability." This affects Internet Explorer 9, Internet Explorer 10, and Internet Explorer 11. This CVE ID is unique from CVE-2018-8643."*
>
> *– (NIST)*

Risk is the combination of probability and impact. In the context of vulnerabilities, risk is the combination of the probability that a vulnerability can be successfully exploited and the impact on the system if it is exploited. A CVE's score represents this risk calculation for the vulnerability. The **Common Vulnerability Scoring System** (**CVSS**) is used to estimate the risk for each vulnerability in the NVD. To calculate the risk, the CVSS uses "exploitability metrics", such as the attack vector, attack complexity, privileges required, and user interaction (NIST, n.d.). To calculate an estimate of the impact on a system if a vulnerability is successfully exploited, the CVSS uses "impact metrics", such as the expected impact on confidentiality, integrity, and availability (NIST, n.d.).

Notice that both the exploitability metrics and impact metrics are provided in the CVE details that I mentioned earlier. The CVSS uses these details in some simple mathematical calculations to produce a base score for each vulnerability (Wikipedia).

Vulnerability management professionals can further refine the base scores for vulnerabilities by using metrics in a temporal metric group and an environmental group.

The temporal metric group reflects the fact that the base score can change over time as new information becomes available; for example, when proof of concept code for a vulnerability becomes publicly available. Environmental metrics can be used to reduce the score of a CVE because of the existence of mitigating factors or controls in a specific IT environment. For example, the impact of a vulnerability might be blunted because a mitigation for the vulnerability had already been deployed by the organization in their previous efforts to harden their IT environment. The vulnerability disclosure trends that I discuss in this chapter are all based on the base scores for CVEs.

The CVSS has evolved over time—there have been three versions to date. The ratings for the latest version, version 3, are represented in the following diagram (NIST, n.d.). NVD CVSS calculators for CVSS v2 and v3 are available to help organizations calculate vulnerability scores using temporal and environmental metrics (NIST, n.d.).

The scores can be converted into ratings such as low, medium, high, and critical to make it easier to manage than using granular numeric scores (NIST, n.d.).

Rating	CVSS Score
Low	0.1 - 3.9
Medium	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

Table 2.1: Rating descriptions for ranges of CVSS scores.

Vulnerabilities with higher scores have higher probabilities of exploitation and/or greater impacts on systems when exploited. Put another way, the higher the score, the higher the risk. This is why many vulnerability management teams use these scores and ratings to determine how quickly to test and deploy security updates and/or mitigations for vulnerabilities in their environment, once the vulnerabilities have been publicly disclosed.

Another important term to understand is "zero-day" vulnerability. A zero-day vulnerability is a vulnerability that has been publicly disclosed before the vendor that is responsible for it has released a security update to address it. These vulnerabilities are the most valuable of all vulnerabilities, with attackers and governments willing to pay relatively large sums for them (potentially a million dollars or more for a working exploit).

The worst-case scenario for vulnerability management teams is a critical rated zero-day vulnerability in software or hardware they have in their environment. This means the risk of exploitation could be super high and that the security update that could prevent exploitation of the vulnerability is not publicly available. Zero-day vulnerabilities aren't as rare as you might think. Data that Microsoft released recently indicates that of the CVEs that were known to be exploited in Microsoft products in 2017, the first time they were exploited, 100% were zero-day vulnerabilities and, in 2018, 83% were zero-day vulnerabilities (Matt Miller, 2019).

Here is a fun fact for you. I created a large, sensational news cycle in 2013 when I coined the term "zero day forever" in a blog post I wrote on Microsoft's official security blog. I was referring to any vulnerability found in Windows XP after official support for it ended. In this scenario, any vulnerability found in Windows XP after the end of support would be a zero day forever, as Microsoft would not provide ongoing security updates for it.

Let me explain this in a little more detail. Attackers can wait for new security updates to be released for currently supported versions of Windows, like Windows 10. Then they reverse engineer these updates to find the vulnerability that each update addresses. Then, they check whether those vulnerabilities are also present in Windows XP. If they are, and Microsoft won't release security updates for them, then attackers have zero-day vulnerabilities for Windows XP forever. To this day, you can search for the terms "zero day forever" and find many news articles quoting me. I became the poster boy for the end of life of Windows XP because of that news cycle.

Over the years, I have talked to thousands of CISOs and vulnerability managers about the practices they use to manage vulnerabilities for their organizations. The four most common groups of thought on the best way to manage vulnerabilities in large, complex enterprise environments are as follows:

- **Prioritize critical rated vulnerabilities**: When updates or mitigations for critical rated vulnerabilities become available, they are tested and deployed immediately. Lower rated vulnerabilities are tested and deployed during regularly scheduled IT maintenance in order to minimize system reboots and disruption to business. These organizations are mitigating the highest risk vulnerabilities as quickly as possible and are willing to accept significant risk in order to avoid constantly disrupting their environments with security update deployments.
- **Prioritize high and critical rated vulnerabilities**: When high and critical rated vulnerabilities are publicly disclosed, their policy dictates that they will patch critical vulnerabilities or deploy available

mitigations within 24 hours and high rated vulnerabilities within a month. Vulnerabilities with lower scores will be patched as part of their regular IT maintenance cycle to minimize system reboots and disruption to business.

- **No prioritization – just patch everything**: Some organizations have come to the conclusion that given the continuous and growing volume of vulnerability disclosures that they are forced to manage, the effort they put into analyzing CVE scores and prioritizing updates isn't worthwhile. Instead, they simply test and deploy all updates on essentially the same schedule. This schedule might be monthly, quarterly, or, for those organizations with healthy risk appetites, semi-annually. These organizations focus on being really efficient at deploying security updates regardless of their severity ratings.
- **Delay deployment**: For organizations that are acutely sensitive to IT disruptions, who have been disrupted by poor quality security updates in the past, delaying the deployment of security updates has become an unfortunate practice. In other words, these organizations accept the risk related to all publicly known, unpatched vulnerabilities in the products they use for a period of months to ensure that security updates from their vendors aren't re-released due to quality issues. These organizations have decided that the cure is potentially worse than the disease; that is, disruption from poor quality security updates poses the same or higher risk to them than all potential attackers in the world. The organizations that subscribe to this school of thought tend to bundle and deploy months' worth of updates. The appetite for risk among these organizations is high, to say the least.

To the uninitiated, these approaches and the trade-offs seem might not make much sense. The primary pain point that deploying vulnerabilities creates, besides the expense, is disruption to the business. For example, historically, most updates for Windows operating systems required reboots. When systems get rebooted, the downtime incurred is counted against the uptime goals that most IT organizations are committed to. Rebooting a single server might not seem material, but the time it takes to reboot hundreds or thousands of servers starts to add up. Keep in mind that organizations trying to maintain 99.999% (5 "9s") uptime can only afford to have 5 minutes and 15 seconds of downtime per year. That's 26.3 seconds of downtime per month. Servers in enterprise data centers, especially database and storage servers, can easily take more than 5 minutes to reboot when they are healthy. Additionally, when a server is rebooted, this is a prime time for issues to surface that require troubleshooting, thereby exacerbating the downtime. The worst-case scenario is when a security update itself causes a problem. The time it takes to uninstall the update and

reboot yet again, on hundreds or thousands of systems, leaving them in a vulnerable state, also negatively impacts uptime.

Patching and rebooting systems can be expensive, especially for organizations that perform supervised patching in off hours, which can require overtime and weekend wages. The concept of the conventional maintenance window is no longer valid, as many businesses are global and operate across borders, 24 hours per day, seven days per week. A thoughtful approach to scheduled and layered patching, keeping the majority of infrastructure available while patching and rebooting a minority, has become common.

Reboots are the top reason that organizations decide to accept some risk by patching quarterly or semi-annually, so much so that the MSRC that I worked closely with for over a decade used to try to minimize the number of security updates that required system reboots to every second month. To do this, when possible, they would try to release all the updates that required a reboot one month and then release updates that didn't require reboots the next month. When this plan worked, organizations that were patching systems every month could at least avoid rebooting systems every second month. But the "out of band" updates, which were unplanned updates, seemed to spoil these plans frequently.

When you see how vulnerability disclosures have trended over time, the trade-offs that organizations make between risk of exploitation and uptime might make more sense. Running servers in the cloud can dramatically change this equation—I'll cover this in more detail in *Chapter 8, The Cloud – A Modern Approach to Security and Compliance*.

There are many other aspects and details of the NVD, CVE, and CVSS that I didn't cover here, but I've provided enough of a primer that you'll be able to appreciate the vulnerability disclosure trends that I provide next.

Vulnerability Disclosure Data Sources

Before we dig into the vulnerability disclosure data, let me tell you where the data comes from and provide some caveats regarding the validity and reliability of the data. There are two primary sources of data that I used for this chapter:

1. The NVD: `https://nvd.nist.gov/vuln/search`
2. CVE Details: `https://www.cvedetails.com/`

The NVD is the *de facto* authoritative source of vulnerability disclosures for the industry, but that doesn't mean the data in the NVD is perfect, nor is the CVSS. I attended a session at the Black Hat USA conference in 2013 called "Buying into the Bias: Why Vulnerability Statistics Suck" (Brian Martian, 2013).

This session covered numerous biases in CVE data. This talk is still available online and I recommend watching it so that you understand some of the limitations of the CVE data that I discuss in this chapter. CVE Details is a great website that saved me a lot of time collecting and analyzing CVE data. CVE Details inherits the limitations of the NVD because it uses data from the NVD. It's worth reading how CVE Details works and its limitations (CVE Details). Since the data and analysis that I provide in this chapter is based on the NVD and CVE Details, they inherit these limitations and biases.

Given that the two primary sources of data that I used for the analysis in this chapter have stated limitations, I can state with confidence that my analysis is not entirely accurate or complete. Also, vulnerability data changes over time as the NVD is updated constantly. My analysis is based on a snapshot of the CVE data taken months ago that is no longer up to date or accurate. I'm providing this analysis to illustrate how vulnerability disclosures were trending over time, but I make no warranty about this data – use it at your own risk.

Industry Vulnerability Disclosure Trends

First, let's look at the vulnerability disclosures in each year since the NVD was started in 1999. The total number of vulnerabilities assigned a CVE identifier between 1999 and 2019 was 122,774. As *Figure 2.1* illustrates, there was a large increase in disclosures between 2016 and 2018. There was a 128% increase in disclosures between 2016 and 2017, and a 157% increase between 2016 and 2018. Put another way, in 2016, vulnerability management teams were managing 18 new vulnerabilities per day on average. That number increased to 40 vulnerabilities per day in 2017 and 45 per day in 2018, on average.

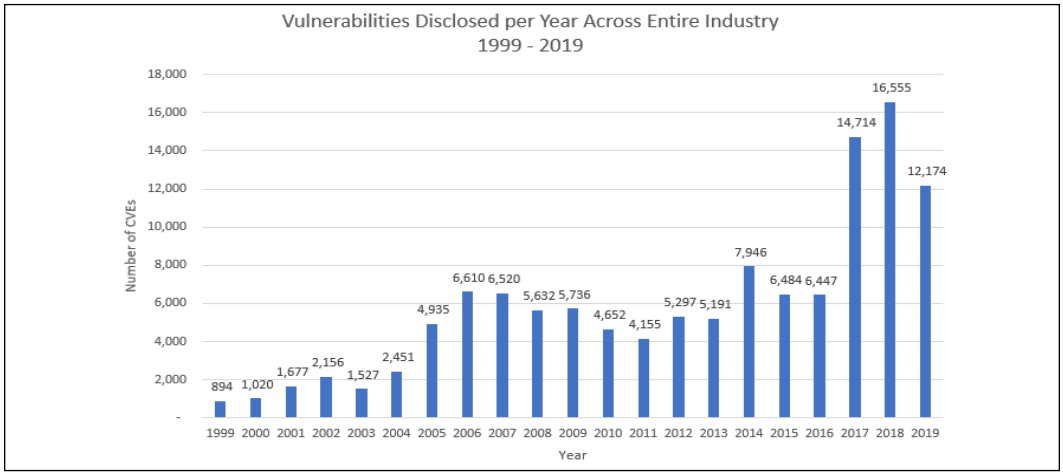


Figure 2.1: Vulnerabilities disclosed across the industry per year (1999–2019)

You might be wondering what factors contributed to such a large increase in vulnerability disclosures. The primary factor was likely a change made to how CVE identifiers are assigned to vulnerabilities in the NVD. During this time, the CVE anointed and authorized what they call "CVE Numbering Authorities (CNAs)" to assign CVE identifiers to new vulnerabilities (Common Vulnerabilities and Exposures, n.d.). According to MITRE, who manages the CVE process that populates the NVD with data:

> *"CVE Numbering Authorities (CNAs) are organizations from around the world that are authorized to assign CVE IDs to vulnerabilities affecting products within their distinct, agreed-upon scope, for inclusion in first-time public announcements of new vulnerabilities. These CVE IDs are provided to researchers, vulnerability disclosers, and information technology vendors.*
>
> *Participation in this program is voluntary, and the benefits of participation include the ability to publicly disclose a vulnerability with an already assigned CVE ID, the ability to control the disclosure of vulnerability information without pre-publishing, and notification of vulnerabilities in products within a CNA's scope by researchers who request a CVE ID from them."*
>
> *– MITRE*

The advent of CNAs means that there are many more organizations assigning CVE identifiers after 2016. As of January 1, 2020, there were 110 organizations from 21 countries participating as CNAs. The names and locations of the CNAs are available at `https://cve.mitre.org/cve/cna.html`. Clearly, this change has made the process of assigning CVE identifiers more efficient, thus leading to the large increase in vulnerability disclosures in 2017 and 2018. 2019 ended with fewer vulnerabilities than 2018 and 2017, but still significantly more than 2016.

There are other factors that have led to higher volumes of vulnerability disclosures. For example, there are more people and organizations doing vulnerability research than ever before and they have better tools than in the past. Finding new vulnerabilities is big business and a lot of people are eager to get a piece of that pie. Additionally, new types of hardware and software are rapidly joining the computer ecosystem in the form of **Internet of Things** (**IoT**) devices. The great gold rush to get meaningful market share in this massive new market space has led the industry to make all the same mistakes that software and hardware manufacturers made over the past 20 years.

I talked to some manufacturers about the security development plans for their IoT product lines several years ago, and it was evident they planned to do very little. Developing IoT devices that lack updating mechanisms takes the industry back in time, to when personal computers couldn't update themselves, but on a much, much larger scale. Consumers simply are not willing to pay more for better security and manufacturers are unwilling to invest the time, budget, and effort into aspects of development that do not drive demand. If the last 3 years are any indication, this increased volume of vulnerability disclosures appears to be the new normal for the industry, leading to much more risk and more work to manage.

The distribution of the severity of these CVEs is illustrated in *Figure 2.2*. There are more CVEs rated high severity (CVSS scores between 7 and 8) and medium severity (CVSS scores between 4 and 5) than CVEs with other ratings. The weighted average CVSS score is 6.6. More than a third of all vulnerabilities (44,107) are rated critical or high. For organizations that have vulnerability management policies dictating the emergency deployment of all critical rated vulnerabilities and the monthly deployment of CVEs rated high, that's potentially more than 15,000 emergency deployments and over 25,000 monthly patch deployments over a 20-year period. This is one reason why some organizations decide not to prioritize security updates based on severity—there are too many high and critical severity vulnerabilities to make managing them differently than lower-rated vulnerabilities an effective use of time. Many of these organizations focus on becoming really efficient at testing and deploying security updates in their environment so that they can deploy all updates as quickly as possible without disrupting the business, regardless of their severity.

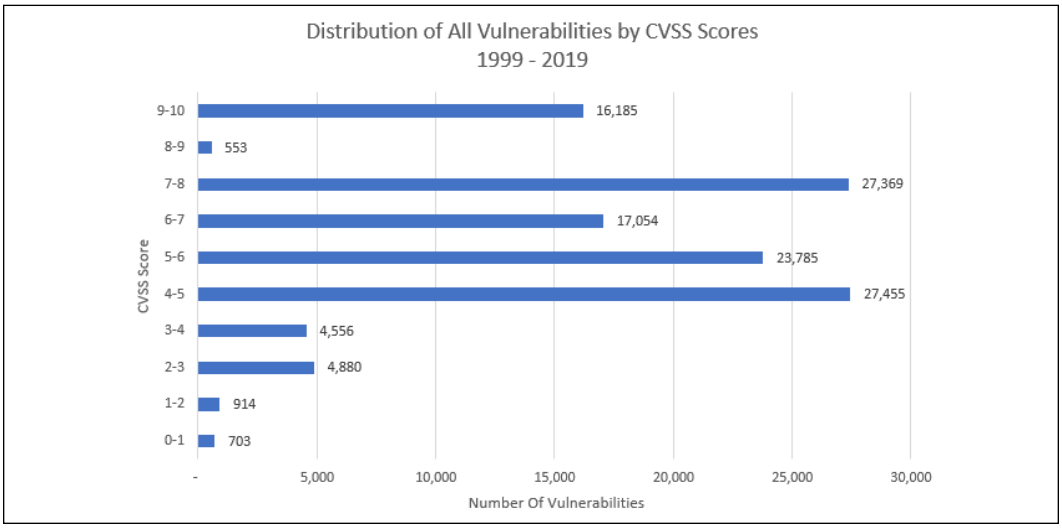


Figure 2.2: CVSS scores by severity (1999–2019)

The vendors and Linux distributions that had the most CVEs according to CVE Details' Top 50 Vendor List (CVE Details, 2020) on January 1 2020 are listed in Figure 2.3. This list shouldn't be all that surprising as some vendors in this list are also the top vendors when it comes to the number of products they have had in the market over the last 20 years. The more code you write, the more potential for vulnerabilities there is, especially in the years prior to 2003 when the big worm attacks (SQL Slammer, MS Blaster, and suchlike) were perpetrated.

After 2004, industry leaders like the ones on this list started paying more attention to security vulnerabilities in the wake of those attacks. I'll discuss malware more in *Chapter 3*, *The Evolution of the Threat Landscape – Malware*. Additionally, operating system and web browser vendors have had a disproportionate amount of attention and focus on their products because of their ubiquity. A new critical or high rated vulnerability in an operating system or browser is worth considerably more than a vulnerability in an obscure application.

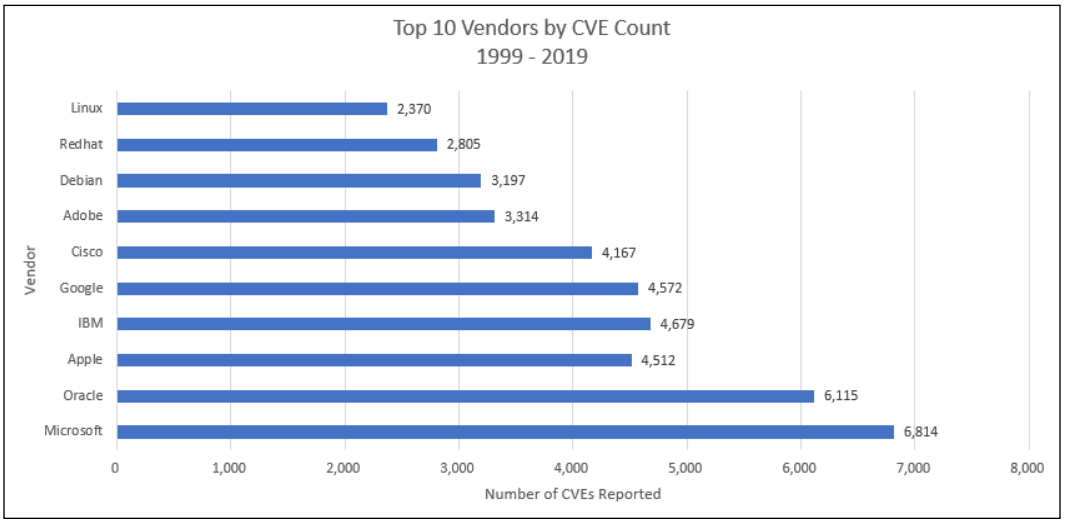


Figure 2.3: Top 10 vendors/distributions with the most CVE counts (1999–2019)

At this point, you might be wondering what type of products these vulnerabilities are in. Categorizing the top 25 products with the most CVEs into operating systems, web browsers, and applications, *Figure 2.4* illustrates the breakdown. In the top 25 products with the most CVEs, there are more CVEs impacting operating systems than browsers and applications combined.

But interestingly, as the number of products is expanded from 25 to 50, this distribution starts to shift quickly, with 5 percent of the total CVEs shifting from the operating system category to applications. I suspect that as the number of products included in this analysis increases, applications would eventually have more CVEs than the other categories, if for no other reason than the fact that there are many, many more applications than operating systems or browsers, despite all the focus operating systems have received over the years. Also keep in mind that the impact of a vulnerability in a popular development library, such as JRE or Microsoft .NET, can be magnified because of the millions of applications that use it.

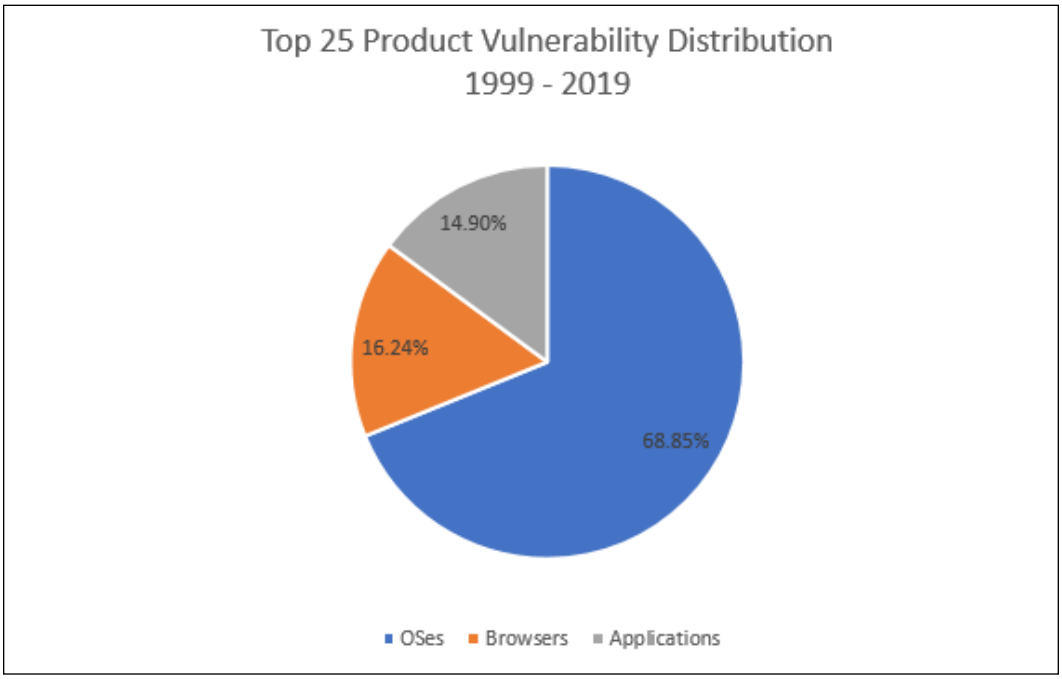


Figure 2.4: Vulnerabilities in the 25 products with the most CVEs categorized by product type (1999–2019)

The specific products that these vulnerabilities were reported in are illustrated in the following list (CVE Details, n.d.). This list will give you an idea of the number of vulnerabilities that many popular software products have and how much effort vulnerability management teams might spend managing them.

Rank	Product	Vendor	Product Type	Vulnerabilities Disclosed
1	Debian Linux	Debian	OS	3067
2	Android	Google	OS	2563
3	Linux Kernel	Linux	OS	2357
4	Mac Os X	Apple	OS	2212
5	Ubuntu Linux	Canonical	OS	2007
6	Firefox	Mozilla	Browser	1873
7	Chrome	Google	Browser	1858
8	Iphone Os	Apple	OS	1655
9	Windows Server 2008	Microsoft	OS	1421
10	Windows 7	Microsoft	OS	1283
11	Acrobat Reader Dc	Adobe	Application	1182
12	Acrobat Dc	Adobe	Application	1182
13	Windows 10	Microsoft	OS	1111
14	Flash Player	Adobe	Application	1078
15	Windows Server 2012	Microsoft	OS	1050
16	Enterprise Linux Server	Redhat	OS	1050
17	Enterprise Linux Desktop	Redhat	OS	1039
18	Internet Explorer	Microsoft	Browser	1030
19	Safari	Apple	Browser	1029
20	Windows 8.1	Microsoft	OS	978
21	Acrobat	Adobe	Application	949
22	Enterprise Linux Workstation	Redhat	OS	941
23	Thunderbird	Mozilla	Application	921
24	Opensuse	Opensuse	OS	918
25	Windows Server 2016	Microsoft	OS	889

Table 2.2: The top 25 products with the most CVEs (1999–2019)

Back in 2003, when the big worm attacks on Microsoft Windows happened, many of the organizations I talked to at the time believed that only Microsoft software had vulnerabilities, and other vendors' software was perfect. This, even though thousands of CVEs were being assigned each year before and after 2003 for software from many vendors.

A decade and a half later, I haven't run into many organizations that still believe this myth, as their vulnerability management teams are dealing with vulnerabilities in all software and hardware. Note that there are only two Microsoft products in the top 10 list.

But this data is not perfect and counting the total number of vulnerabilities in this manner does not necessarily tell us which of these vendors and products have improved over the years or whether the industry has improved its security development practices as a whole. Let's explore these aspects more next.

Reducing Risk and Costs – Measuring Vendor and Product Improvement

How can you reduce the risk and costs associated with security vulnerabilities? By using vendors that have been successful at reducing the number of vulnerabilities in their products, you are potentially reducing the time, effort, and costs related to your vulnerability management program. Additionally, if you choose vendors that have also invested in reducing attackers' return on investment by making exploitation of vulnerabilities in their products hard or impossible, you'll also be reducing your risk and costs. I'll now provide you with a framework that you can use to identify such vendors and products.

In the wake of the big worm attacks in 2003, Microsoft started developing the Microsoft SDL (Microsoft, n.d.). Microsoft continues to use the SDL to this day. I managed marketing communications for the SDL for several years, so I had the opportunity to learn a lot about this approach to development. The stated goals of the SDL are to decrease the number and severity of vulnerabilities in Microsoft software.

The SDL also seeks to make vulnerabilities that are found in software after development harder or impossible to exploit. It became clear that even if Microsoft was somehow able to produce vulnerability-free products, the applications, drivers and third-party components running on Windows or in web browsers would still render systems vulnerable. Subsequently, Microsoft shared some versions of the SDL and some SDL tools with the broader industry for free. It also baked some aspects of the SDL into Visual Studio development tools.

I'm going to use the goals of the SDL as an informal "vulnerability improvement framework" to get an idea of whether the risk (probability and impact) of using a vendor or a specific product has increased or decreased over time. This framework has three criteria:

1. Is the total number of vulnerabilities trending up or down?
2. Is the severity of those vulnerabilities trending up or down?
3. Is the access complexity of those vulnerabilities trending up or down?

Why does this seemingly simple framework make sense? Let's walk through it. Is the total number of vulnerabilities trending up or down? Vendors should be working to reduce the number of vulnerabilities in their products over time. An aspirational goal for all vendors should be to have zero vulnerabilities in their products. But this isn't realistic as humans write code and they make mistakes that lead to vulnerabilities. However, over time, vendors should be able to show their customers that they have found ways to reduce vulnerabilities in their products in order to reduce risk for their customers.

Is the severity of those vulnerabilities trending up or down? Given that there will be some security vulnerabilities in products, vendors should work to reduce the severity of those vulnerabilities. Reducing the severity of vulnerabilities reduces the number of those emergency security update deployments I mentioned earlier in the chapter. It also gives vulnerability management teams more time to test and deploy vulnerabilities, which reduces disruptions to the businesses they support. More specifically, the number of critical and high severity CVEs should be minimized as these pose the greatest risk to systems.

Is the access complexity of those vulnerabilities trending up or down? Again, if there are vulnerabilities in products, making those vulnerabilities as hard as possible or impossible to exploit should be something vendors focus on. Access complexity or attack complexity (depending on the version of CVSS being used) is a measure of how easy or hard it is to exploit a vulnerability. CVSS v2 provides an estimate of access complexity as low, medium or high, while CVSS v3 uses attack complexity as either high or low. The concept is the same—the higher the access complexity or attack complexity, the harder it is for the attacker to exploit the vulnerability.

Using these measures, we want to see vendors making the vulnerabilities in their products consistently hard to exploit. We want to see the number of high access complexity CVEs (those with the lowest risk) trending up over time, and low complexity vulnerabilities (those with the highest risk) trending down or zero. Put another way, we want the share of high complexity CVEs to increase.

To summarize this vulnerability improvement framework, I'm going to measure:

- CVE count per year
- The number of critical rated and high rated CVEs per year. These are CVEs with scores of between 7 and 10
- The number of CVEs per year with low access complexity or attack complexity

When I apply this framework to vendors, who can have hundreds or thousands of products, I'll use the last five years' worth of CVE data. I think 5 years is a long enough period to determine whether a vendor's efforts to manage vulnerabilities for their products has been successful. When I apply this framework to an individual product, such as an operating system or web browser, I'll use the last 3 years (2016-2018) of CVE data so that we see the most recent trend. Note that one limitation of this approach is that it won't be helpful in cases where vendors and/or their products are new and there isn't enough data to evaluate.

Now that we have a framework to measure whether vulnerability disclosures are improving over time, I'll apply this framework to two decades of historical CVE data for some select vendors, operating systems, and web browsers to get a better idea of the state of popular software in the industry. Just to add an element of suspense and tension, like you'd find in a Mark Russinovich cybersecurity thriller novel, I'll reveal Microsoft's CVE data last!

Oracle Vulnerability Trends

Since Oracle is #2 in the top 10 list of vendors with the most CVEs, let's start with them. There are CVEs for Oracle products dating back to 1999. *Figure 2.5* illustrates the number of CVEs published each year for Oracle products between 1999 and 2018.

During this period, 5,560 CVEs were assigned, of which 1,062 were rated as critical or high and 3,190 CVEs had low access complexity. There were 489 CVEs disclosed in 2019, making a grand total of 6,112 CVEs in Oracle products between 1999 and 2019 (CVE Details, n.d.).

Note that Oracle acquired numerous technology companies and new technologies during this period, including MySQL and Sun Microsystems. Acquisitions of new technologies can lead to significant changes in CVE numbers for vendors. It can take time for acquiring vendors to get the products they obtain into shape to meet or exceed their standards. In Oracle's case, some of the technologies they acquired turned out to have the most CVEs of any of the products in their large portfolio; these include MySQL, JRE and JDK (CVE Details, n.d.).

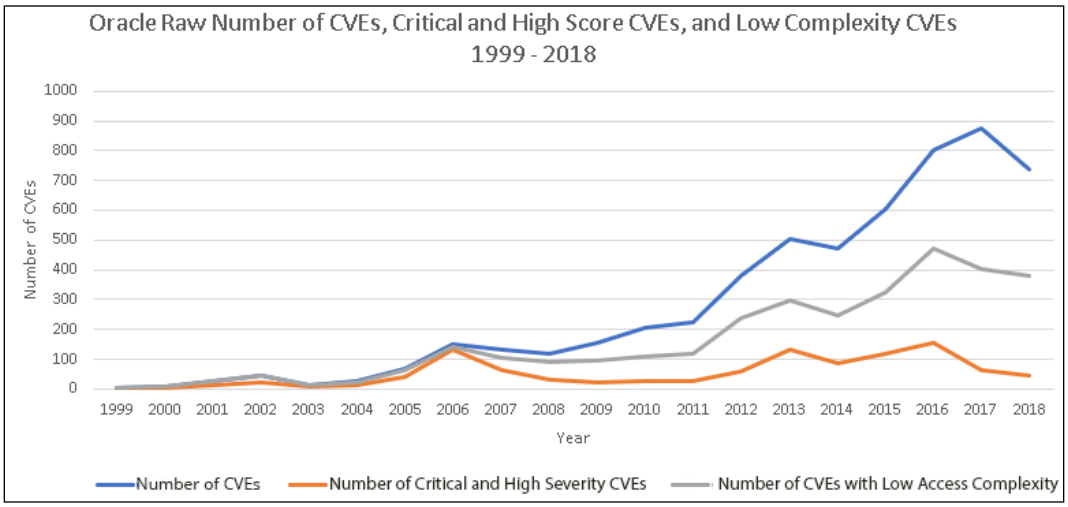


Figure 2.5: Number of CVEs, critical and high CVEs, and low complexity CVEs in Oracle products (1999–2018)

Taking a view of just the last five full years, starting at the beginning of 2014 and ending at the end of 2018, the number of CVEs increased by 56%. There was a 54% increase in the number of CVEs with low access complexity or attack complexity. However, the number of critical and high score (with scores of between 7 and 10) CVEs decreased by 48% during this same period. This is impressive given the big increase in the number of vulnerabilities during this time. This positive change is illustrated by *Figure 2.6* this illustrates the number of critical and high severity CVEs as a percentage of the total CVEs for each year between 1999 and 2018. It also shows us CVEs with low access complexity as a percentage of all CVEs during the same period.

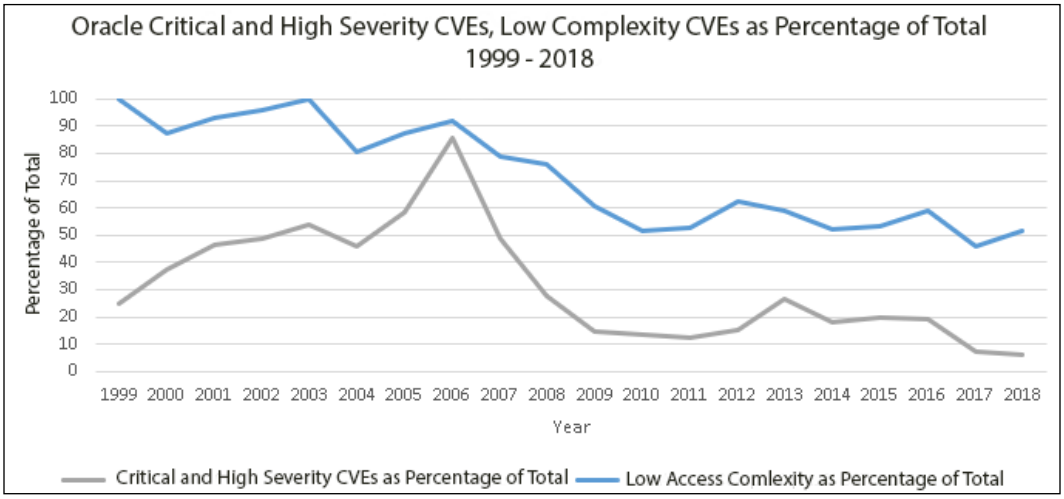


Figure 2.6: Critical and high severity rated CVEs and low complexity CVEs in Oracle products as a percentage of total (1999–2018)

Long-term trends like this don't happen by accident. Oracle likely implemented some changes in people (such as security development training), processes, and/or technology that helped them reduce the risk for their customers. Older products that reach end of life can also help improve the overall picture. Oracle likely also made progress addressing vulnerabilities in many of the technologies it had acquired over the years. There's still a relatively high volume of vulnerabilities that vulnerability management teams need to deal with, but lower severity vulnerabilities are helpful as I discussed earlier.

According to CVE Details, the Oracle products that contributed the most to the total number of CVEs between 1999 and 2018 included MySQL, JRE, JDK, Database Server, and Solaris.

Apple Vulnerability Trends

Next on the list of vendors with the highest number of CVEs is Apple. Between 1999 and 2018, there were 4,277 CVEs assigned to Apple products; of these CVEs, 1,611 had critical or high scores, and 1,524 had access complexity that was described as low (CVE Details, n.d.). There were 229 CVEs disclosed in Apple products in 2019 for a total of 4,507 CVEs between 1999 and 2019 (CVE Details, n.d.). As you can see from *Figure 2.7* there have been big increases and decreases in the number of CVEs in Apple products since 2013.

Looking at just the 5 years between 2014 and the end of 2018, comparing the start and end of this period, there was a 39% reduction in the number of CVEs, a 30% reduction in CVEs with CVSS scores of 7 and higher, and a 65% reduction in CVEs with low access complexity. However, vulnerability management teams had their work cut out for them in 2015 and 2017 when there were the largest increases in CVE numbers in Apple's history.

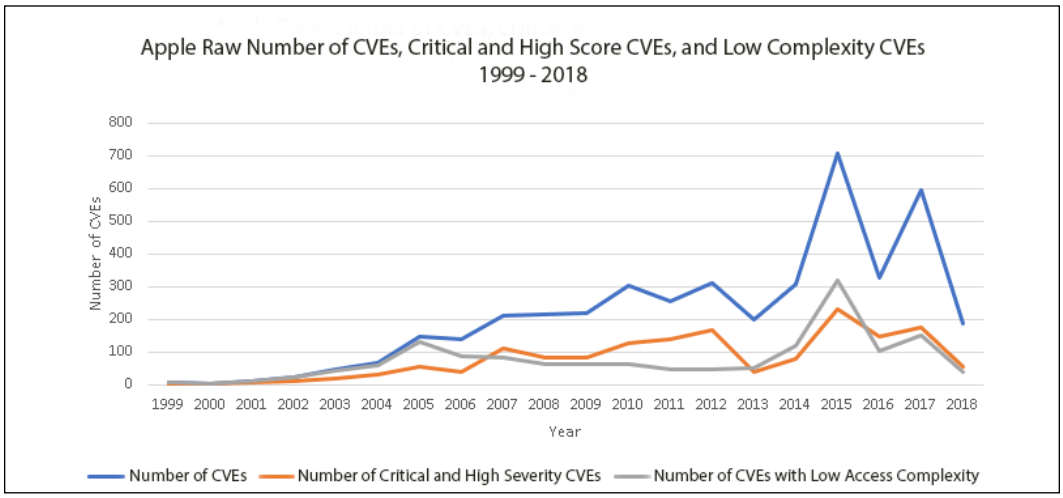


Figure 2.7: Number of CVEs, critical and high CVEs, and low complexity CVEs in Apple products (1999–2018)

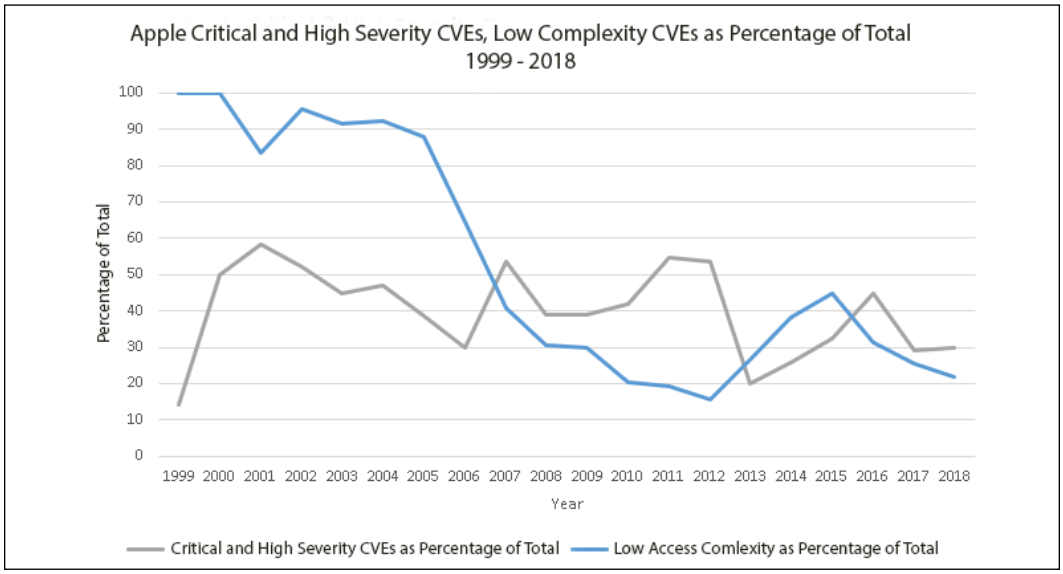


Figure 2.8: Critical and high severity rated CVEs and low complexity CVEs in Apple products as a percentage of total (1999–2018)

The Apple products that contributed the most CVEs to Apple's total, according to CVE Details, include macOS, iOS, Safari, macOS Server, iTunes, and watchOS (CVE Details, n.d.).

IBM Vulnerability Trends

IBM is ranked fourth on the list of vendors with the most vulnerabilities, with just slightly fewer CVEs than Apple between 1999 and 2018, with 4,224 (CVE Details, n.d.), incredibly, a difference of only 53 CVEs over a 19-year period between these two vendors. But Big Blue had nearly half the CVEs rated critical or high compared to Apple. However, IBM had significantly more CVEs with low access complexity compared to Apple.

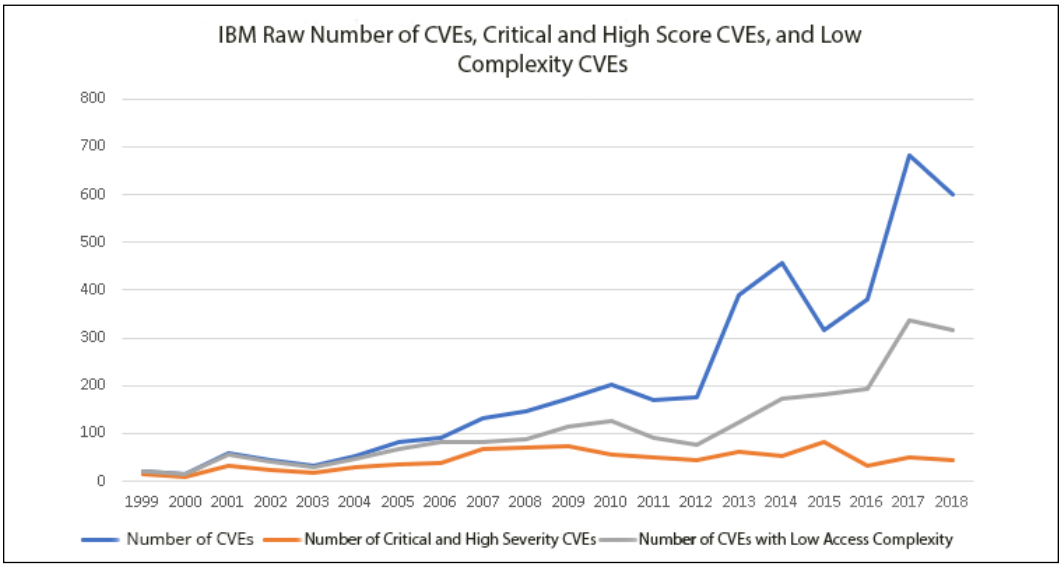


Figure 2.9: Number of CVEs, critical and high score CVEs and low complexity CVEs in IBM products (1999–2018)

Focusing on just the last 5 years between 2014 and the end of 2018, IBM saw a 32% increase in the number of CVEs. There was a 17% decrease in the number of critical and high score CVEs, while there was an 82% increase in CVEs with low access complexity. That decrease in critical and high rated vulnerabilities during a time when CVEs increased by almost a third is positive and noteworthy.

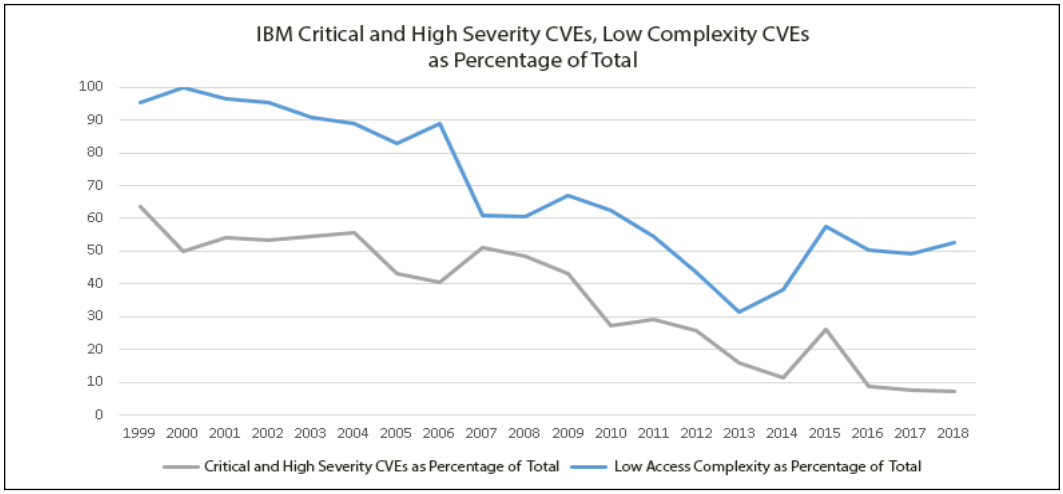


Figure 2.10: Critical and high severity rated CVEs and low complexity CVEs in IBM products as a percentage of total (1999–2018)

The products that contributed the most to IBM's CVE count were AIX, WebSphere Application Server, DB2, Rational Quality Manager, Maximo Asset Management, Rational Collaborative Lifecycle Management and WebSphere Portal (CVE Details, n.d.).

Google Vulnerability Trends

Rounding out the top five vendors with the most CVEs is Google. Google is different from the other vendors on the top 5 list. The first year that a vulnerability was published in the NVD for a Google product was 2002, not 1999 like the rest of them. Google is a younger company than the others on the list.

During the period between 2002 and 2018, there were 3,959 CVEs attributed to Google products. Of these CVEs, 2,078 were rated critical or high score (CVE Details, n.d.). That's more than double the number of critical and high score vulnerabilities versus IBM and Oracle, and significantly more than Apple. Google has more critical and high severity vulnerabilities than any vendor in the top five list, with the exception of Microsoft. 1,982 of the CVEs assigned to Google products during this period had low access complexity (CVE Details, n.d.).

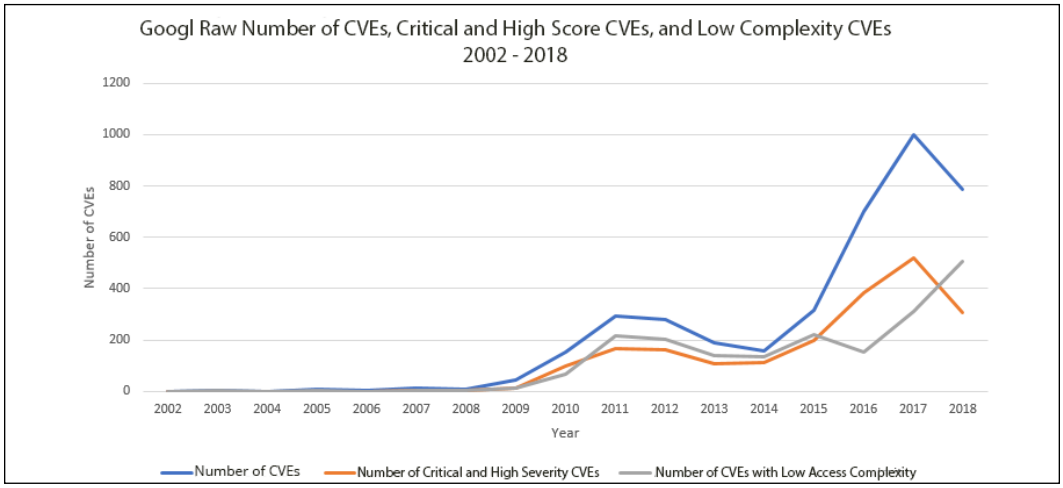


Figure 2.11: The number of CVEs, critical and high CVEs and low complexity CVEs in Google products (2002–2018)

Looking at the trend in the 5 years between 2014 and the end of 2018, there was a 398% increase in CVEs assigned to Google products; during this same period there was a 168% increase in CVEs rated critical or high and a 276% increase in low complexity CVEs (CVE Details, n.d.). The number of CVEs in 2017 reached 1,001, according to CVE Details (CVE Details, n.d.), a feat that none of the top 5 vendors has ever achieved.

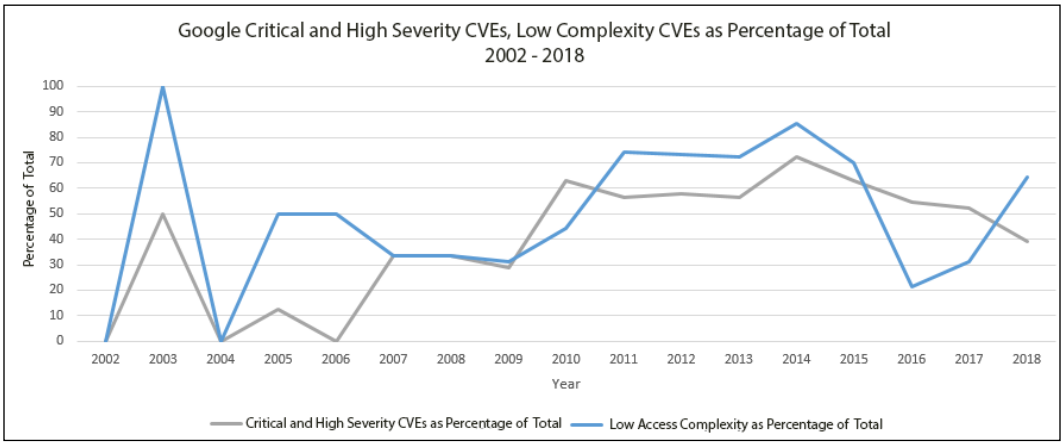


Figure 2.12: Critical and high severity rated CVEs and low complexity CVEs in Google products as a percentage of total (2002–2018)

According to CVE Details, the Google products that contributed the most to Google's overall CVE count included Android and Chrome (CVE Details, n.d.).

Microsoft Vulnerability Trends

Now it's time to look at how Microsoft has been managing vulnerabilities in their products. They top the list of vendors with the most CVEs, with 6,075 between 1999 and the end of 2018 (CVE Details, n.d.).

Of the aforementioned 6,075 CVEs, 3,635 were rated critical or high, and 2,326 CVEs had low access/attack complexity (CVE Details, n.d.). Of the 5 vendors we examined, Microsoft had the highest total number of vulnerabilities, the highest number of vulnerabilities with CVSS scores of 7 and higher, and the most CVEs with low access complexity.

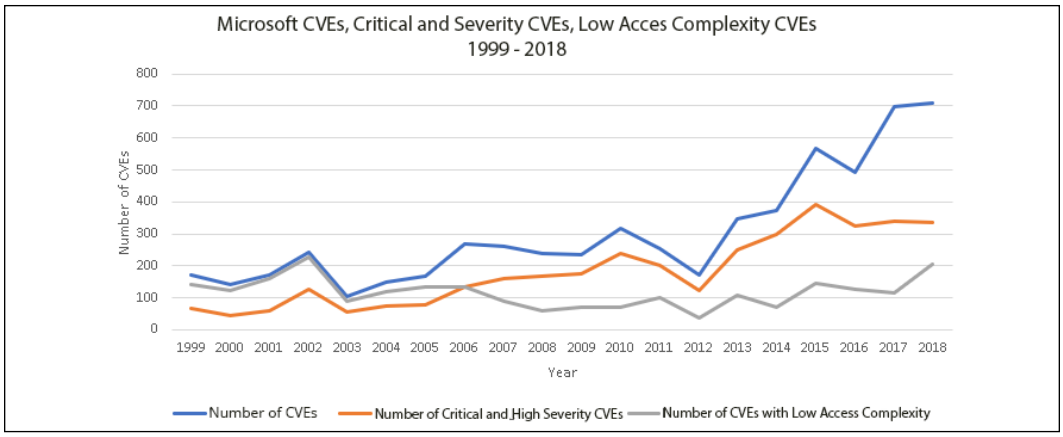


Figure 2.13: The number of CVEs, critical and high CVEs and low complexity CVEs in Microsoft products (1999–2018)

Focusing on the 5 years between 2014 and the end of 2018, there was a 90% increase in CVEs assigned to Microsoft products. There was a 14% increase in critical and high score vulnerabilities and a 193% increase in low access complexity CVEs. If there is a silver lining, it's that Microsoft has made it significantly harder to exploit vulnerabilities over the long term. Microsoft released compelling new data recently on the exploitability of their products that is worth a look to get a more complete picture (Matt Miller, 2019).

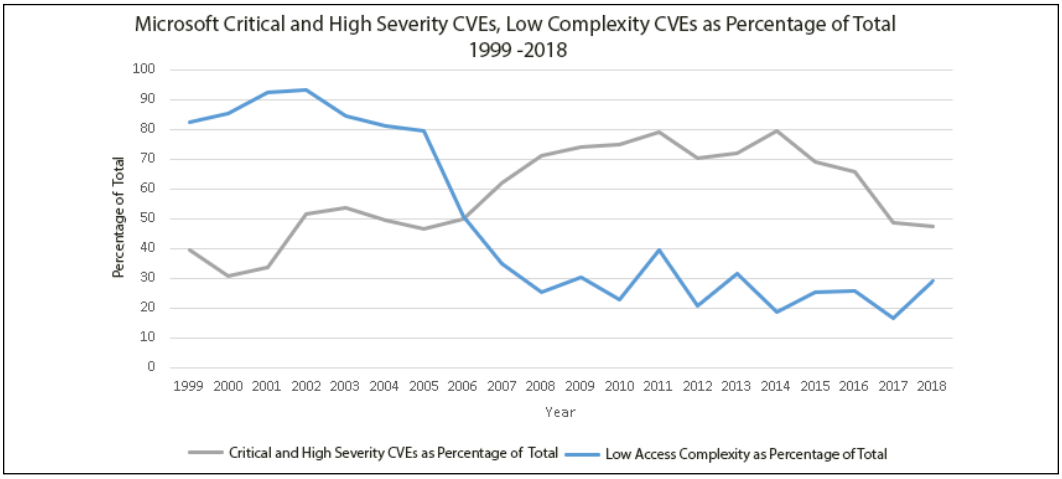


Figure 2.14: Critical and high severity rated CVEs and low complexity CVEs in Microsoft products as a percentage of total (1999–2018)

The products that contributed the most to Microsoft's overall CVE count include Windows Server 2008, Windows 7, Windows 10, Internet Explorer, Windows Server 2012, Windows 8.1, and Windows Vista (CVE Details, n.d.). Some operating systems on this list were among the most popular operating systems in the world, at one time or another, especially among consumers. This makes Microsoft's efforts to minimize vulnerabilities in these products especially important. I'll discuss vulnerability disclosure trends for operating systems and web browsers later in this chapter.

Vendor Vulnerability Trend Summary

All the vendors we examined in this chapter have seen dramatic increases in the number of vulnerabilities in their products over time. The volume of vulnerability disclosures in the 2003–2004 timeframe seems quaint compared to the volumes we have seen over the past 3 years. Big increases in the number of vulnerabilities can make it more challenging to reduce the severity and increase the access complexity of CVEs.

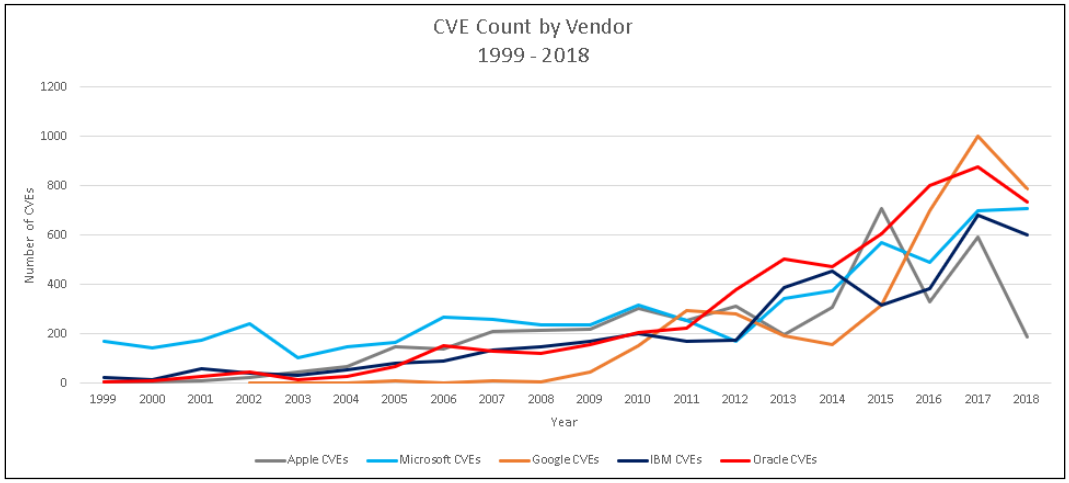


Figure 2.15: CVE count for the top five vendors (1999–2018)

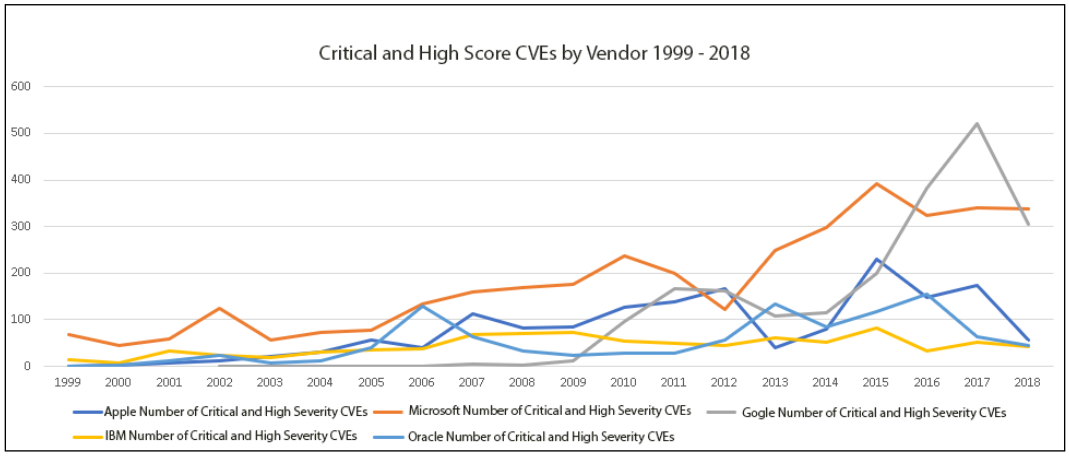


Figure 2.16: The counts of critical and high rated severity CVEs for the top five vendors (1999–2018)

Only one of the industry leaders we examined has achieved all three of the goals we defined earlier for our informal vulnerability improvement framework. Focusing on the last five full years for which I currently have data (2014–2018), Apple successfully reduced the number of CVEs, the number of critical and high severity CVEs and the number of CVEs with low access complexity. Congratulations Apple!

Vendor	Fewer CVEs?	Fewer CVEs with CVSS Score 7-10?	Fewer Low Access Complexity CVEs?
Apple	Yes	Yes	Yes
Google	No	No	No
IBM	No	Yes	No
Microsoft	No	No	No
Oracle	No	Yes	No

Table 2.3: The results from applying the vulnerability improvement framework (2014–2018)

It's super challenging to drive these metrics in the right direction across potentially hundreds of products for years at a time. Let's examine how individual products have performed over time. Next, we'll look at select operating systems and web browsers.

Operating System Vulnerability Trends

Operating systems have garnered a lot of attention from security researchers over the past couple of decades. A working exploit for a zero-day vulnerability in a popular desktop or mobile operating system is potentially worth hundreds of thousands of dollars or more. Let's look at the vulnerability disclosure trends for operating systems and look closely at a few of the products that have the highest vulnerability counts.

Figure 2.17 illustrates the operating systems that had the most unique vulnerabilities between 1999 and 2019, according to CVE Details (CVE Details, n.d.). The list contains desktop, server, and mobile operating systems from an array of vendors including Apple, Google, Linux, Microsoft, and others:

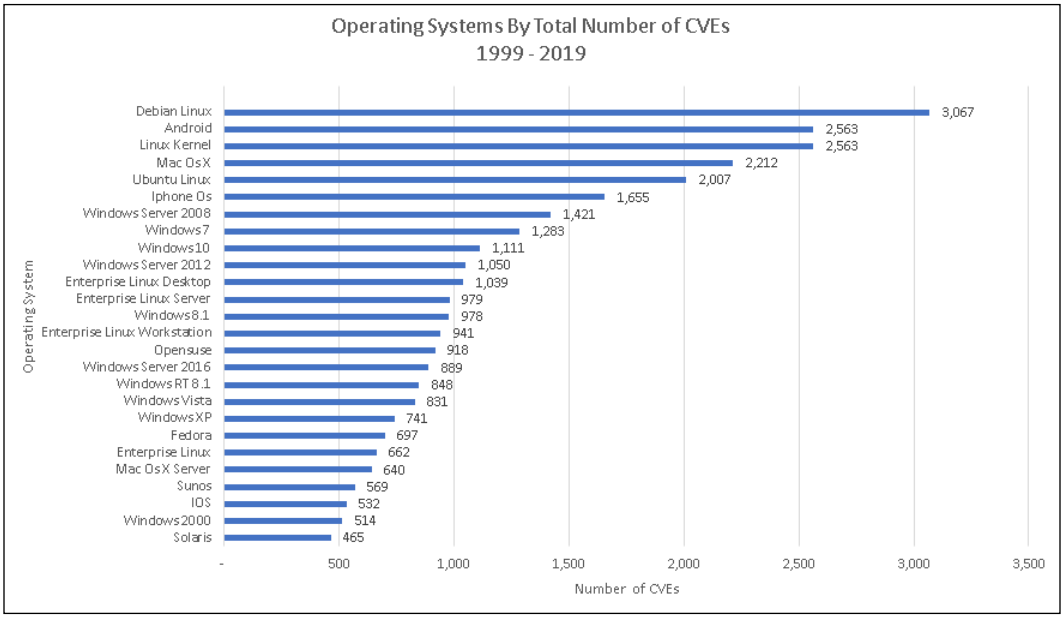


Figure 2.17: Operating systems with the most unique vulnerabilities by total number of CVE counts (1999–2019)

Microsoft Operating System Vulnerability Trends

Since we covered Microsoft last in the previous section, I'll start with their operating systems here. After working on that customer-facing incident response team at Microsoft that I mentioned earlier, I had the opportunity to work in the Core Operating System Division at Microsoft. I was a program manager on the Windows Networking team. I helped ship Windows Vista, Windows Server 2008, and some service packs. Believe it or not, shipping Windows was an even harder job than that customer facing incident response role. But that is a topic for a different book.

Let's look at a subset of both client and server Microsoft operating systems. *Figure 2.18* illustrates the number of CVEs per year for Windows XP, Windows Server 2012, Windows 7, Windows Server 2016, and Windows 10.

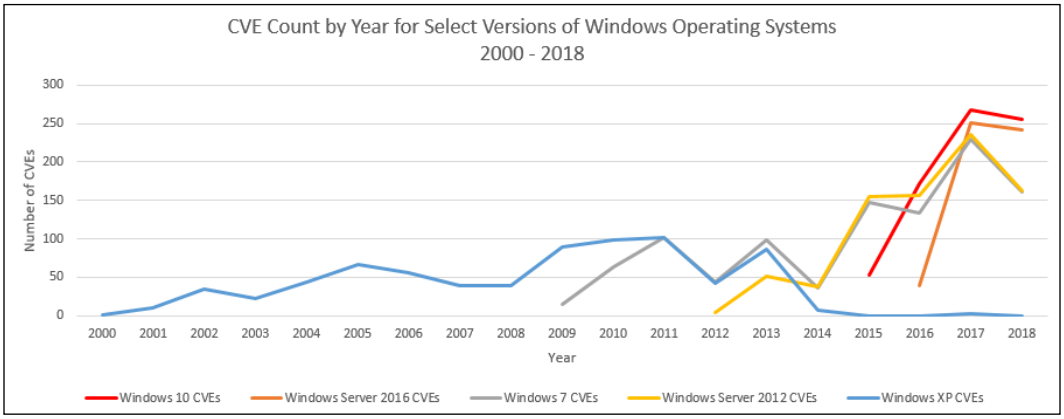


Figure 2.18: CVE count for select versions of Microsoft Windows (2000–2018)

Figure 2.18 gives us some insight into how things have changed with vulnerability disclosures over time. It shows us how much more aggressively vulnerabilities have been disclosed in the last 4 or 5 years compared with earlier periods. For example, in the 20 years that vulnerability disclosures were reported in Windows XP, a total of 741 CVEs were disclosed (CVE Details, n.d.); that's 37 CVEs per year on average. Windows 10, Microsoft's latest client operating system, exceeded that CVE count with 748 CVEs in just 4 years. That's 187 vulnerability disclosures per year on average. This represents a 405% increase in CVEs disclosed on average per year.

Server operating systems have also seen an increasingly aggressive vulnerability discovery rate. A total of 802 vulnerabilities were disclosed in Windows Server 2012 in the 7 years between 2012, when it was released, and 2018 (CVE Details, n.d.); that's 114 CVEs per year on average. But that average jumps to 177 CVEs per year for Windows Server 2016, which represents a 55% increase.

Given that the newest operating systems, Windows 10 and Windows Server 2016, shouldn't have any of the vulnerabilities that were fixed in previous operating systems before they shipped and they have had the benefit of being developed with newer tools and better trained developers, the pace of disclosures is incredible. However, with other operating systems reaching end of life, and Windows 10 being the only new client version of Windows, it is likely getting more attention from security researchers than any other Windows operating system version ever.

Let's now take a deeper look at some of these versions of Windows and apply our vulnerability improvement framework to them.

Windows XP Vulnerability Trends

Windows XP no longer received support as of April 2014, but there were 3 CVEs disclosed in 2017 and 1 in 2019, which is why the graph in figure 2.19 has a long tail (CVE Details, n.d.). Although the number of critical and high severity CVEs in Windows XP did drop from their highs in 2011 by the time support ended in early 2014, the number of CVEs with low access complexity remained relatively high. I don't think we can apply our vulnerability improvement framework to the last few years of Windows XP's life since the last year, in particular, was distorted by a gold rush to find and keep new zero-day vulnerabilities that Microsoft would presumably never fix. These vulnerabilities would be very valuable as long as they were kept secret.

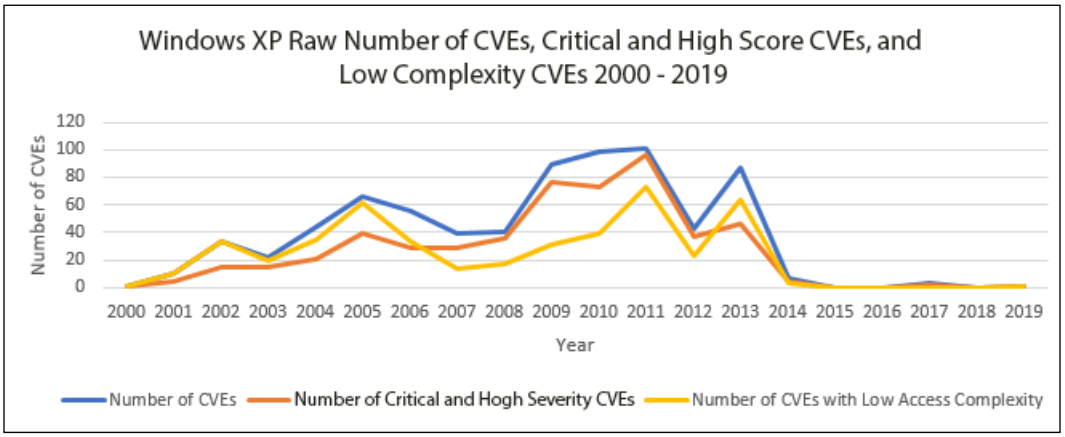


Figure 2.19: The number of CVEs, critical and high rated severity CVEs and low complexity CVEs in Microsoft Windows XP (2000–2019)

Why did Microsoft release security updates for Windows XP after it went out of support? It's that "zero day forever" concept I mentioned earlier. Facing new, critical, potentially worm-able vulnerabilities in Windows XP, Microsoft made the decision to offer security updates for Windows XP after the official support lifetime ended.

The alternative was potentially thousands or millions of compromised and infected "zombie" Windows XP systems constantly attacking the rest of the internet. Microsoft made the right decision releasing updates for Windows XP after its end of life given how many enterprises, governments, and consumers still use it.

Figure 2.20 illustrates the critical and high severity CVEs and low complexity CVEs as a percentage of the total number of CVEs in Windows XP. The erratic pattern in 2017 and 2019 is a result of very few CVEs disclosed in those years (3 in 2017 and 1 in 2019) (CVE Details, n.d.).

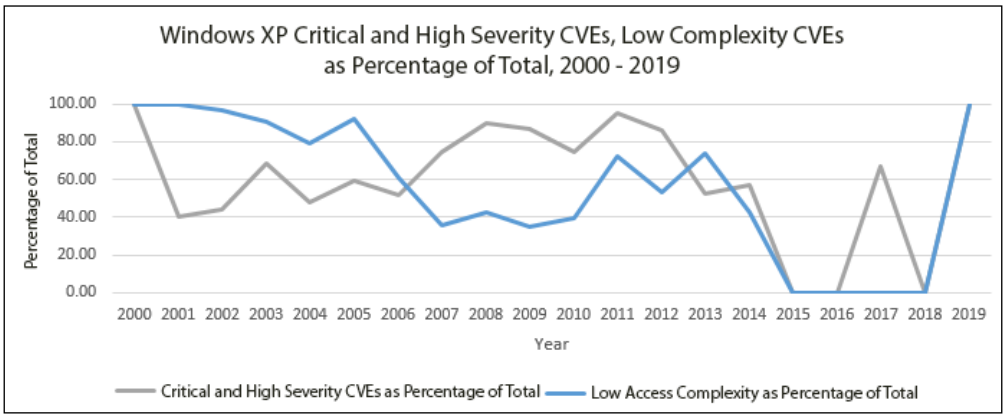


Figure 2.20: Critical and high severity rated CVEs and low complexity CVEs in Microsoft Windows XP as a percentage of all Microsoft Windows XP CVEs (2000–2019)

Windows 7 Vulnerability Trends

Next, let's examine the data for the very popular Windows 7 operating system. Windows 7 went out of support on January 14, 2020 (Microsoft Corporation, 2020). Windows 7 was released in July 2009, after the poorly received Windows Vista. Everyone loved Windows 7 compared to Windows Vista. Additionally, Windows 7 enjoyed a "honeymoon" when it was released from a CVE disclosure perspective as it took a couple of years for CVE disclosures to ramp up, and in recent years, they have increased significantly.

Windows 7 had 1,031 CVEs disclosed between 2009 and 2018. On average, that's 103 vulnerability disclosures per year (CVE Details, n.d.). That's not as high as Windows 10's average annual CVE disclosure rate, but is nearly 3 times the average number of CVEs disclosed in Windows XP per year. Windows 7 had 57 critical or high rated vulnerabilities per year on average.

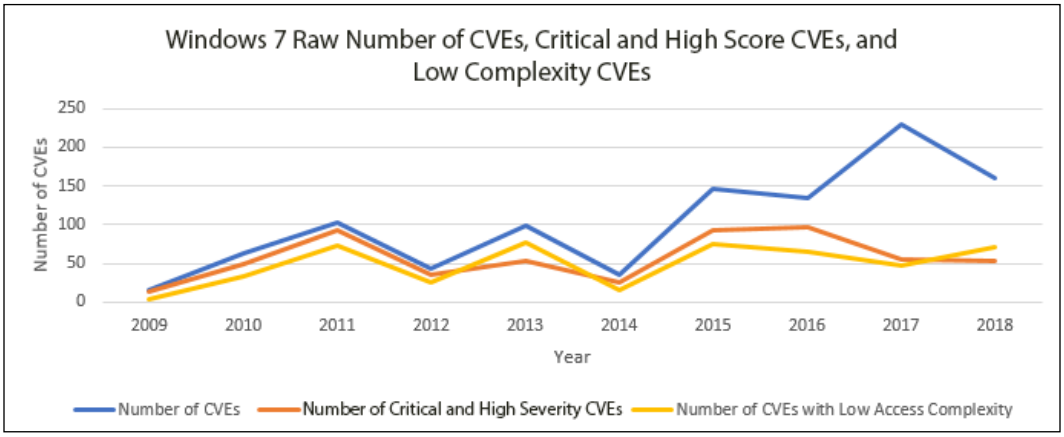


Figure 2.21: The number of CVEs, critical and high rated severity CVEs and low complexity CVEs in Microsoft Windows 7 (2009–2018)

If we focus on just the last 3 years between 2016 and 2018 (a period for which we have data for several Windows versions for comparison purposes), the number of CVEs increased by 20% from the beginning of 2016 and the end of 2018, while the number of critical and high severity CVEs decreased by 44%, and the number of low complexity CVEs increased by 8% (CVE Details, n.d.). A significant decrease in vulnerability severity is helpful to vulnerability management teams, but this doesn't achieve the goals of our vulnerability improvement framework for this 3-year period.

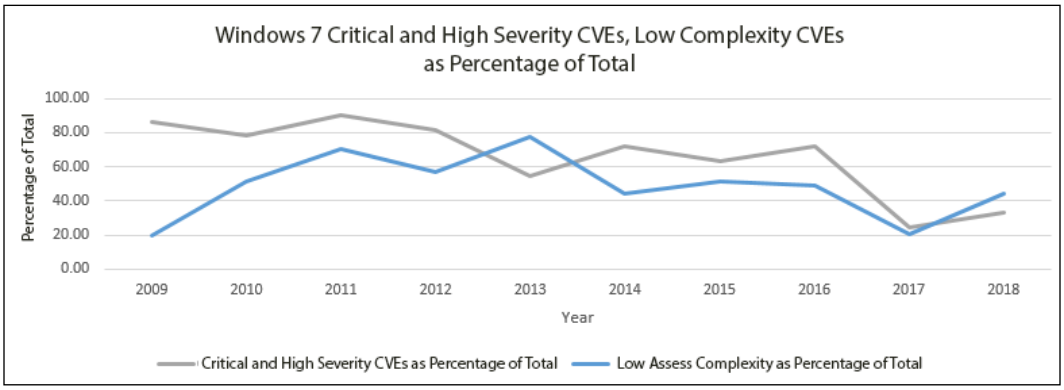


Figure 2.22: Critical and high severity rated CVEs and low complexity CVEs in Microsoft Windows 7 as a percentage of all Microsoft Windows 7 CVEs (2009-2018)

Windows Server 2012 and 2016 Vulnerability Trends

Let's now look at a couple of Windows Server SKUs – Windows Server 2012 and 2016. Windows Server 2012 was released in September 2012. Windows Server 2016 was released in September 2016, so we don't have a full year's worth of data for 2016. This will skew the results of our framework because it will appear that our metrics all had large increases compared to 2016.

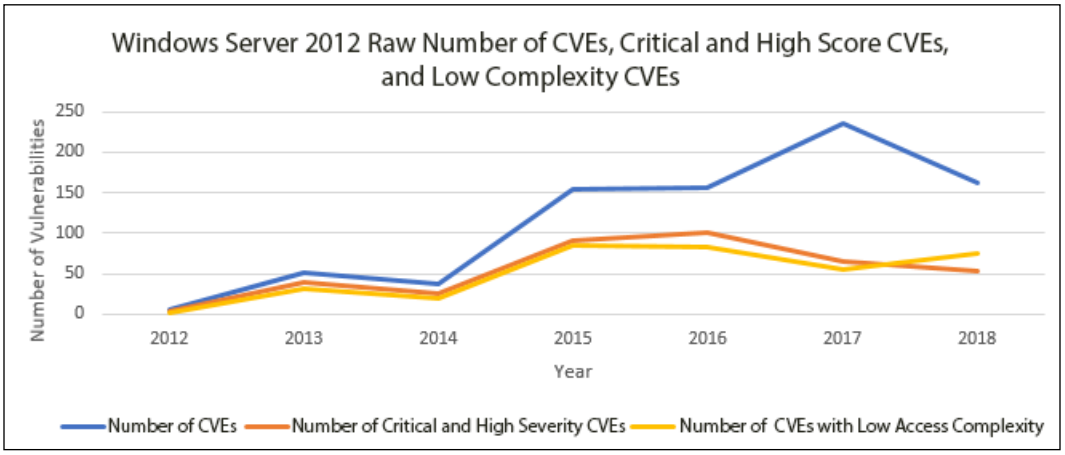


Figure 2.23: The number of CVEs, critical and high rated severity CVEs and low complexity CVEs in Microsoft Windows Server 2012 (2012–2018)

By the end of 2018, Windows Server 2012 had 802 CVEs in the NVD. Across the 7 years in *Figure 2.23*, on average, there were 115 CVEs per year, of which 54 CVEs were rated critical or high (CVE Details, n.d.). For the period between 2016 and the end of 2018, Windows Server 2012's CVE count increased by 4%, while critical and high severity CVEs decreased by 47%, and low complexity CVEs decreased by 10%. It comes very close to achieving the goals of our vulnerability improvement framework. So close!

Unfortunately, the story isn't as straightforward for Windows Server 2016. We simply do not have enough full year data to see how vulnerability disclosures are trending. There is a huge increase (518%) in CVE disclosures between 2016 and 2018, but that's only because we only have one quarter's data for 2016. However, the number of disclosures between 2017 and 2018 is essentially the same (251 and 241, respectively).

Windows Server 2012 had 235 disclosures in 2018 and 162 in 2018 (CVE Details, n.d.). That's an average of 199 CVEs per year for those 2 years, where Windows Server 2016's average was 246 for 2 full years. However, 2 years' worth of data simply isn't enough data; we need to wait for more data in order to understand how Windows Server 2016 is performing.

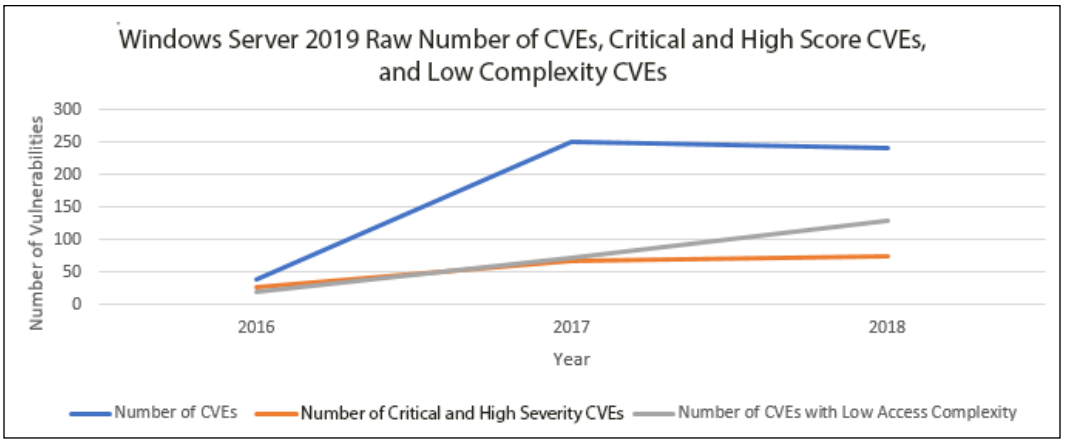


Figure 2.24: The number of CVEs, critical and high rated severity CVEs, and low complexity CVEs in Microsoft Windows Server 2016, (2016–2018)

Windows 10 Vulnerability Trends

The final Windows operating system I'll examine here was called "the most secure version of Windows ever" (err…by me (Ribeiro, n.d.)), Windows 10. This version of Windows was released in July 2015. At the time of writing, I had a full three years' worth of data from 2016, 2017 and 2018. By the end of 2018, Windows 10 had a total of 748 CVEs in the NVD; on average, 187 CVEs per year and 76 critical and high severity vulnerabilities per year (CVE Details, n.d.).

During this 3-year period the number of CVEs in Windows 10 increased by 48%, while the number of critical and high score CVEs decreased by 25% and the number of low access complexity CVEs increased by 48%.

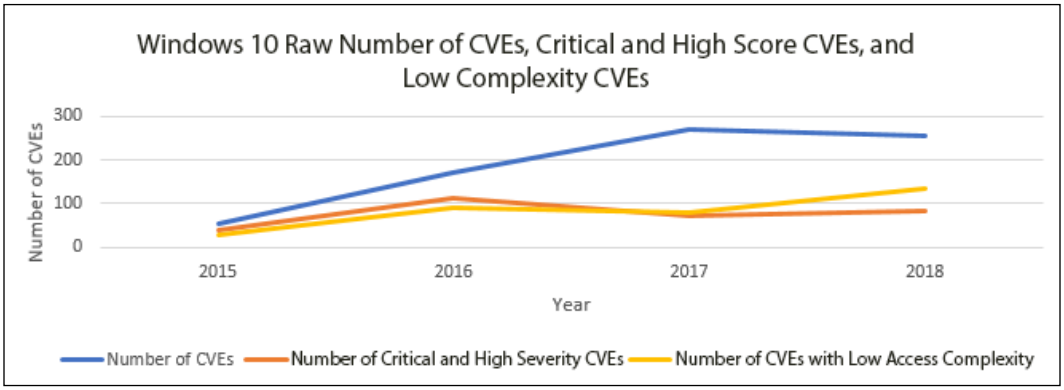


Figure 2.25: The number of CVEs, critical and high rated severity CVEs and low complexity CVEs in Microsoft Windows 10 (2015–2018)

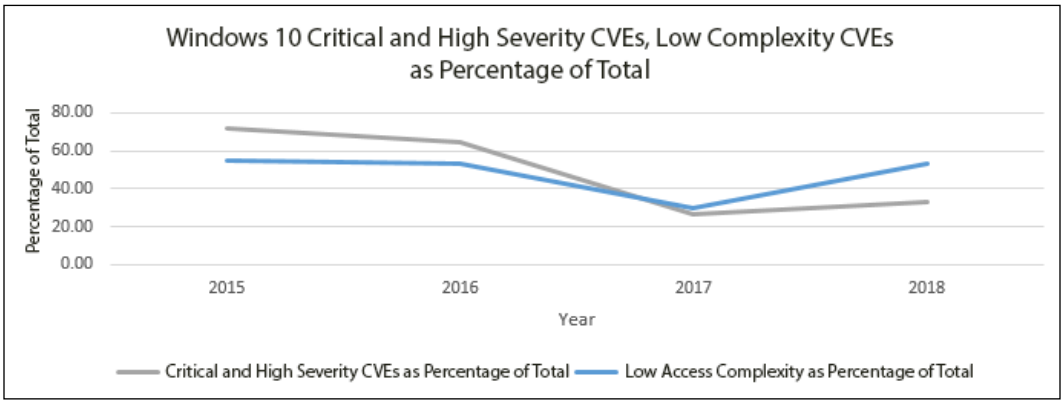


Figure 2.26: Critical and high severity rated CVEs and low complexity CVEs in Microsoft Windows 10 as a percentage of all Microsoft Windows 10 CVEs (2015–2018)

2019 ended with 357 CVEs in Windows 10, a 33% increase from 2018, and the highest number of CVEs than any year since it was released (CVE Details, n.d.). One important factor this data doesn't reflect is that Microsoft has become very good at quickly patching hundreds of millions of systems around the world. This is very helpful in reducing risk for their customers. Let's now examine whether some other popular operating systems managed to meet our criteria.

Linux Kernel Vulnerability Trends

According to CVE Details, at the time of writing, Debian Linux and Linux Kernel have the highest numbers of CVEs of all the products they track. Let's examine the CVE trends for Linux Kernel. The cumulative total number of CVEs from 1999 to 2018 is 2,163, or about 108 CVEs per year on average (CVE Details, n.d.). This is 3 times the annual average of Windows XP, just under the annual average for Windows Server 2012 (114), and well under the annual average for Windows Server 2016 (177). There were 37 critical and high rated CVEs in the Linux Kernel per year on average.

Looking at the same three-year period between 2016 and the end of 2018, we can see from the following graph in figure 2.28, that there was a large increase in CVE disclosures between 2016 and 2017. This is consistent with the trend we saw for the entire industry that I discussed earlier in the chapter. This appears to be a short-term increase for Linux Kernel. 2019 ended with 170 CVEs in Linux Kernel, down from 177 in 2018 (CVE Details, n.d.).

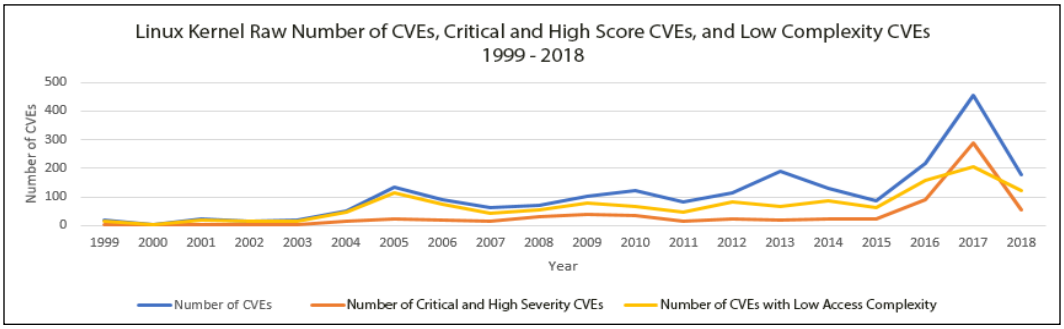


Figure 2.27: The number of CVEs, critical and high rated severity CVEs and low complexity CVEs in Linux Kernel (1999–2018)

Between 2016 and the end of 2018, the number of CVEs decreased by 18%, while the number of CVEs with scores of 7 and higher decreased by 38%. During the same period, the number of low complexity CVEs decreased by 21%. Linux Kernel appears to have achieved the goals of our vulnerability improvement framework. Wonderful!

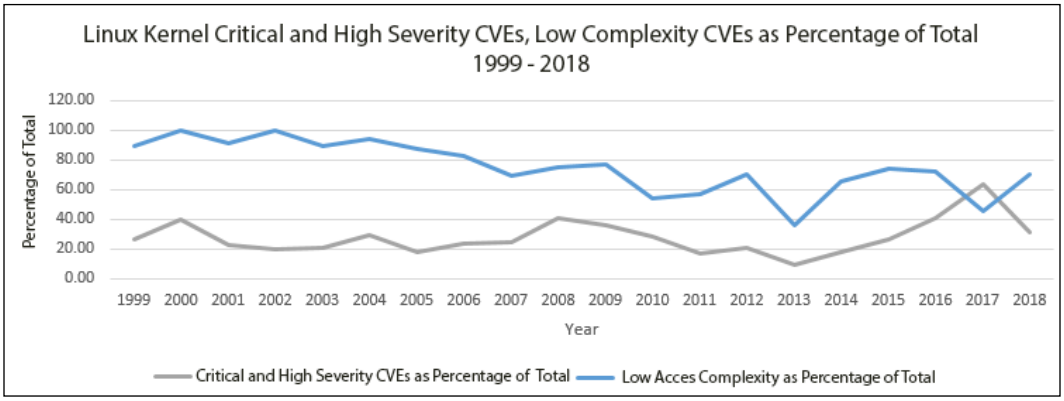


Figure 2.28: Critical and high severity rated CVEs and low complexity CVEs in Linux Kernel as a percentage of all Linux Kernel CVEs (1999–2018)

Google Android Vulnerability Trends

Let's look at Android, a mobile operating system manufactured by Google. Android's initial release date was in September 2008 and CVEs for Android start showing up in the NVD in 2009. On average, there were 215 CVEs filed for Android per year, with 129 CVEs per year rated critical or high severity; Android only had 43 CVEs in the 6 years spanning 2009 and 2014 (CVE Details, n.d.). The volume of CVEs in Android started to increase significantly in 2015 and has increased since then.

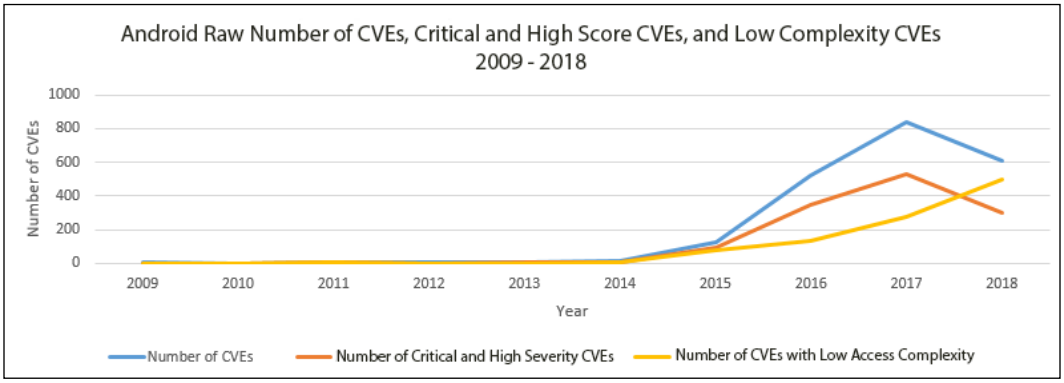


Figure 2.29: The number of CVEs, critical and high rated severity CVEs and low complexity CVEs in Google Android (2009–2018)

In the 3 years between 2016 and the end of 2018, the number of CVEs in Android increased by 16%, while the number of critical and high score CVEs decreased by 14%, but the number of low complexity CVEs increased by 285%.

The total number of CVEs filed for Android between 2009 and the end of 2018 was 2,147 according to CVE Details (CVE Details, n.d.).

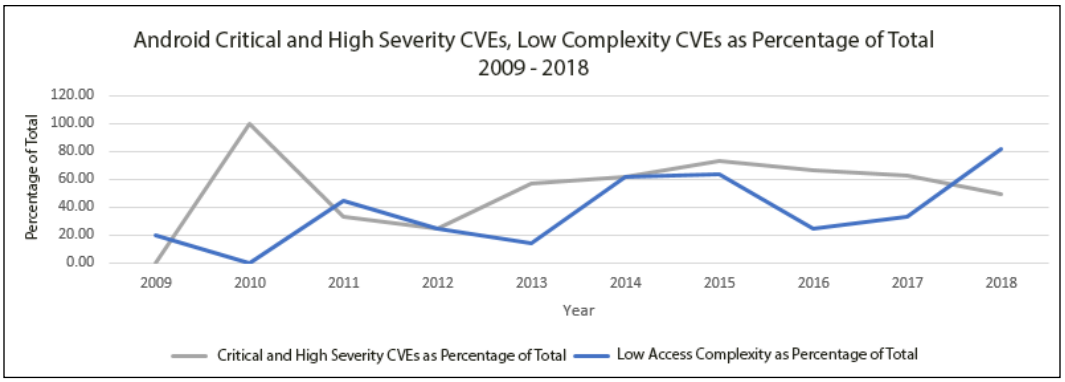


Figure 2.30: Critical and high severity rated CVEs and low complexity CVEs in Google Android as a percentage of all Google Android CVEs during (2009–2018)

Apple macOS Vulnerability Trends

The final operating system I'll examine here is Apple's macOS. Between 1999 and 2018, 2,094 CVEs were entered into the NVD for macOS (CVE Details, n.d.). That's 105 CVEs per year on average, with about 43 critical and high severity CVEs per year. This is very similar to Linux Kernel's average of 108 CVEs per year. You can see from *Figure 2.31* that there was a large increase in CVEs in 2015.

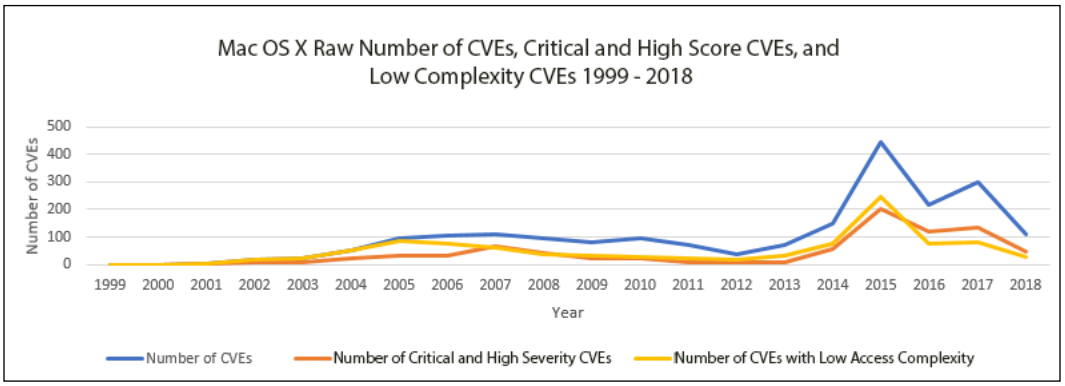


Figure 2.31: Number of CVEs, critical and high rated severity CVEs, and low complexity CVEs in macOS (1999–2018)

During the period spanning from the start of 2016 to the end of 2018, the number of CVEs for MacOS X declined by 49%. The number of critical and high severity CVEs decreased by 59%. Low access complexity CVEs decreased by 66%. MacOS X achieved the objectives of our vulnerability improvement framework. Well done again, Apple!

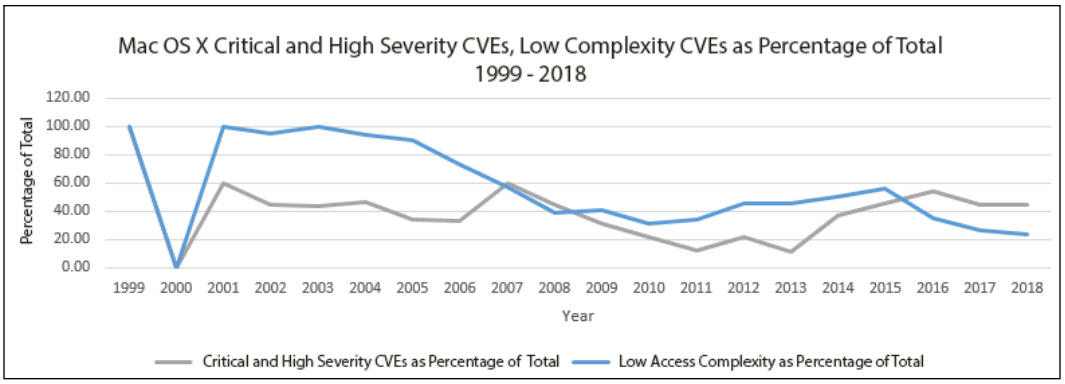


Figure 2.32: Critical and high severity rated CVEs and low complexity CVEs in macOS as a percentage total of all CVEs (1999–2018)

Operating Systems Vulnerability Trend Summary

The operating systems we examined in this chapter are among the most popular operating systems in history. When I applied our vulnerability improvement framework to the vulnerability disclosure data for these operating systems, the results were mixed.

None of the Microsoft operating systems I examined met the criteria set in our vulnerability improvement framework. Windows Server 2012 came very close, but CVEs for it did increase by 4% during the period I examined. Adjusting the timeframe might lead to a different conclusion, but all the operating systems' CVE trends I examined were for the same period. Microsoft has released exploitation data that shows that the exploitability of vulnerabilities in their products is very low due to all the memory safety features and other mitigations they've implemented in Windows (Matt Miller, 2019). This is bittersweet for vulnerability management teams because although the vast majority of vulnerabilities cannot be successfully exploited, they still need to be patched. However, in terms of mitigating the exploitation of unpatched vulnerabilities, it's good to know Microsoft has layered in so many effective mitigations for their customers.

Google Android did not meet the goals in the vulnerability improvement framework during the 2016–2018 timeframe. There was a small increase in CVEs and a 285% increase in low complexity CVEs during this period. (CVE Details, n.d.)

macOS and Linux Kernel did meet the criteria of the vulnerability improvement framework, and these vendors should be congratulated and rewarded for their achievement of reducing risk for their customers.

OS	Fewer CVEs?	Fewer CVEs with CVSS Score 7-10?	Fewer Low Access Complexity CVEs?
Microsoft Windows Server 2012	No	Yes	Yes
Microsoft Windows 7	No	Yes	No
Microsoft Windows 10	No	Yes	No
Linux Kernel	Yes	Yes	Yes
Android	No	Yes	No
Mac OS X	Yes	Yes	Yes

Table 2.4: Application results for the vulnerability improvement framework (2016–2018)

In Table 2.5, I am providing you with an interesting summary of the CVE data for the operating systems I have examined. The Linux Kernel and Apple macOS stand out from the others on the list due to the relatively low average number of critical and high severity CVEs per year.

OS	Total Number of CVEs at the end of 2018	Average Number of CVEs/Year	Average Number of Critical & High CVEs/Year
Microsoft Windows 10	748	187	76
Microsoft Windows Server 2012	802	115	54
Microsoft Windows 7	1031	103	57
Mac OS X	2094	105	43
Android	2147	215	129
Linux Kernel	2163	108	37

Table 2.5: Operating systems' vital statistics (1999– 2018)

Before I examine web browsers, I want to point out one of the limitations of the data I presented in this section. While I was able to split out CVE data for each individual version of Windows, I didn't do that for macOS releases. Similarly, I didn't dig into the granular details of different Linux distributions to examine data for custom kernels and third-party applications. Comparing an individual version of Windows, such as Windows 7, for example, with all macOS releases isn't like comparing apples with apples, if you can forgive the pun. More research is required to uncover trends for specific non-Windows operating system releases.

The trend data for individual operating system releases could be quite different from the results for all releases as a group. However, the data I did present still illustrates something more fundamental than trends for specific operating system versions, many of which are out of support. It illustrates how the development and test processes of these operating system vendors have performed over a period of many years. Put another way, it illustrates what vendors' security standards look like and whether they've been able to improve continuously over time. From this, we can draw conclusions about which of these vendors is adept at potentially reducing the costs of vulnerability management for enterprises, while also reducing risks for them.

Next let's look at vulnerability trends in web browsers, which also get a lot of scrutiny from security researchers around the world.

Web Browser Vulnerability Trends

Web browsers attract a lot of attention from security researchers and attackers alike. This is because they are hard to live without. Everyone uses at least one browser on desktops, mobile devices and servers. Operating systems' development teams can bake layers of security features into their products, but web browsers tend to bring threats right through all those host-based firewalls and other security layers. Web browsers have been notoriously difficult to secure and, as you'll see from the data in this section, there has been a steady volume of vulnerabilities over the years in all popular browsers.

Just an additional warning about the web browser data that I share with you in this section. Of all the NVD, CVE and CVSS data that I analyzed for this chapter, I have the least confidence in the accuracy of this data. This is because, over time, different product names were used for CVEs in the NVD, making it challenging to ensure I have a complete data set. For example, some CVEs for Internet Explorer were labeled as "IE" instead. I did my best to find all the variations using nicknames that I could, but I can't guarantee that the data is complete and accurate.

The number of CVEs between 1999 and April 2019 is illustrated in *Figure 2.33* (CVE Details, n.d.).

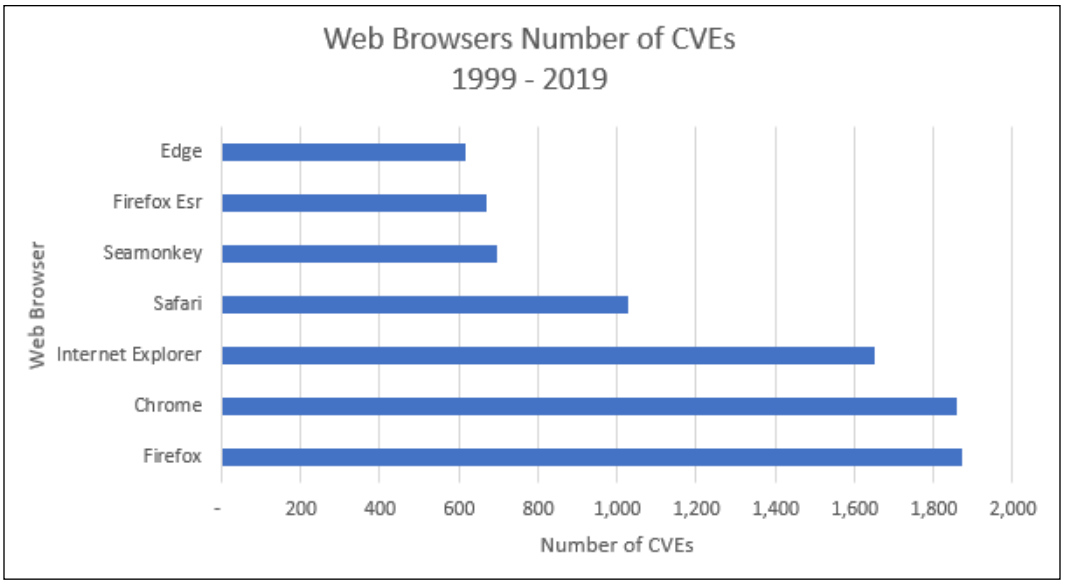


Figure 2.33: Total number of CVEs in popular web browsers (1999–2019)

I'll dig into the data and apply our vulnerability improvement framework to a few of these products to give you an idea of how these vendors have been managing vulnerabilities in some of the world's most popular web browsers.

Internet Explorer Vulnerability Trends

Let's start by examining Microsoft **Internet Explorer** (**IE**). IE has been around for many years with different versions getting released for different operating systems. I was able to find 1,597 CVEs for Internet Explorer between 1999 and 2018 (CVE Details, n.d.). This is an average of 80 vulnerabilities per year and 57 critical and high severity CVEs per year.

Figure 2.34 illustrates the number of CVEs, the number of critical and high rated CVEs, and the number of low complexity CVEs for each year between 1999 and 2018. You can see a big increase in the number of CVEs, and the number of critical and high score CVEs during the period 2012–2017.

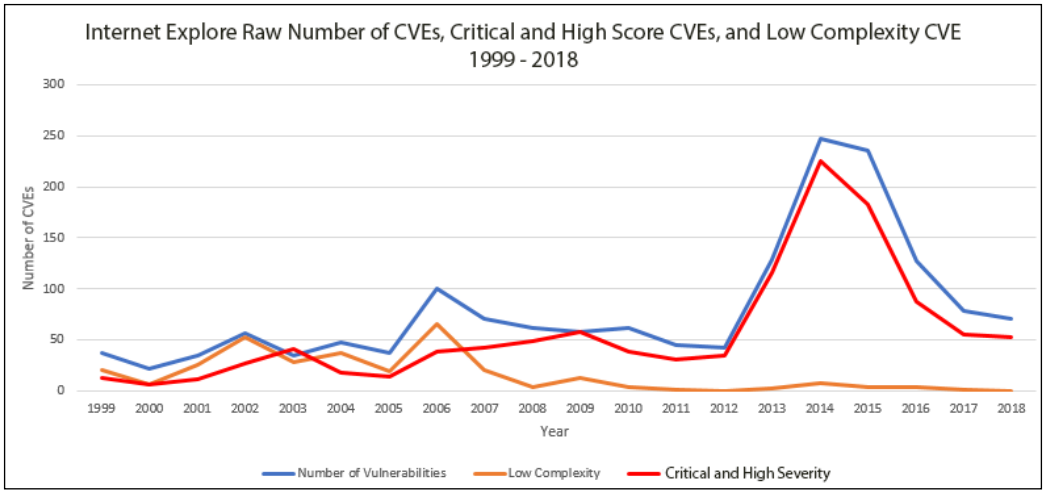


Figure 2:34: The number of CVEs, critical and high severity CVEs and low complexity CVEs in IE (1999–2018)

A noteworthy data point, illustrated by figure 2.37, is just how many critical rated CVEs have been found in IE over the years. Remember that many organizations will initiate and perform an emergency update process for every critical rated vulnerability that is disclosed, because the risk is so high. Of the 1,597 CVEs in IE, 768 of them, that's 48%, were rated critical (CVE Details, n.d.). The years that saw the largest number of these CVEs were 2013, 2014, and 2015. Microsoft moved to a quarterly security update release model where they release cumulative security updates instead of individual updates in order to minimize the disruption all of these CVEs would otherwise cause.

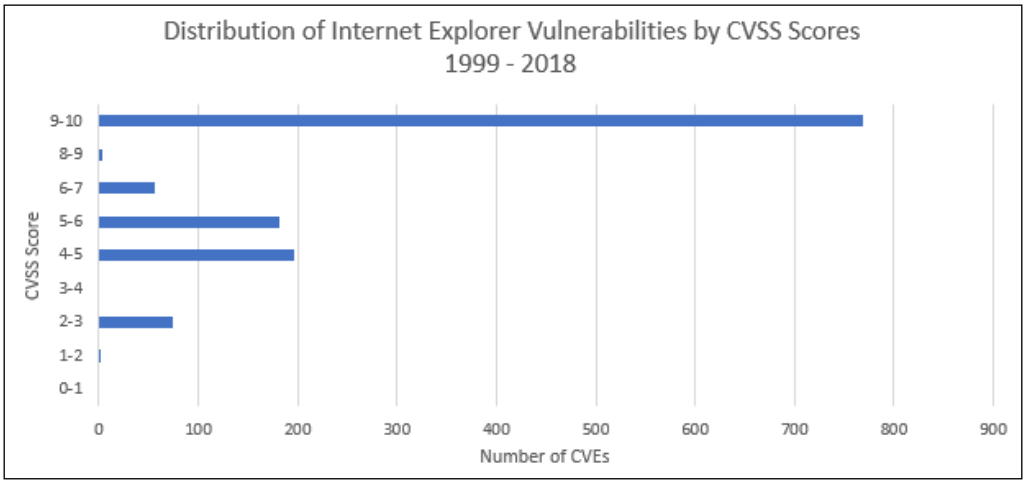


Figure 2.35: Distribution of CVSS scores for CVEs in IE (1999–2018)

Despite the high volume of CVEs and the large number of critical and high rated CVEs, IE fairs well when we put this data into our vulnerability improvement framework focusing on the 3 years between 2016 and the end of 2018. The effort to drive down CVEs from their highs in 2014 and 2015 shows up as a 44% decline in CVEs and a 41% decline in critical and high rated CVEs between 2016 and 2018. Additionally, there were zero low complexity CVEs in 2018. Microsoft has met the criteria in our vulnerability improvement framework and, more importantly, the goals of the SDL. Nice work, Microsoft!

Next, let's examine the Edge browser.

Microsoft Edge Vulnerability Trends

Edge is the web browser that Microsoft released with Windows 10 in 2015. Microsoft made numerous security enhancements to this browser based on the lessons they learned from IE (Microsoft Corporation, n.d.).

According to CVE Details, there were 525 CVEs for Edge between 2015 and the end of 2018 (CVE Details, n.d.). On average, this is 131 vulnerabilities per year and 95 critical and high severity CVEs per year. *Figure 2.36* illustrates the volume of these CVEs per year along with the number of critical and high severity vulnerabilities, and the number of low complexity CVEs. The number of CVEs climbed quickly in the first few years as vulnerabilities that weren't fixed before Edge was released were found and disclosed. This means that Edge won't meet the criteria for our vulnerability improvement framework. However, the decline in CVEs in 2018 continued into 2019 with a further 57% reduction. If I included 2019 in my analysis, Edge could potentially meet the criteria.

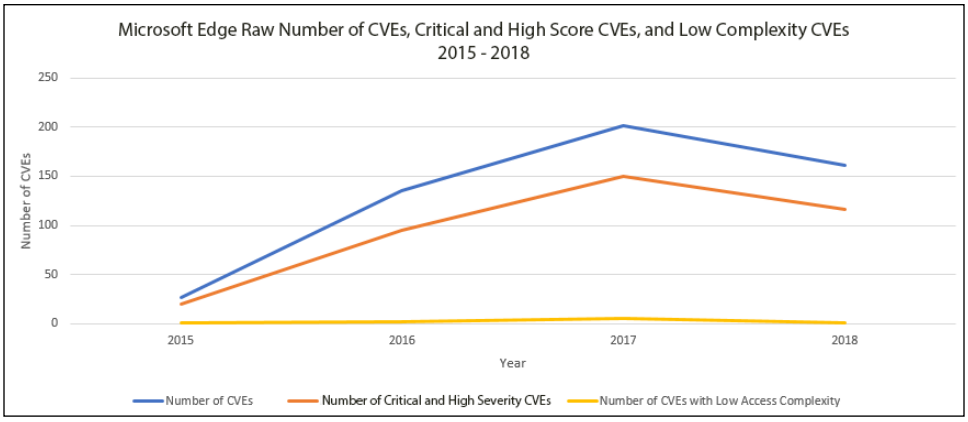


Figure 2.36: The number of CVEs, critical and high severity CVEs and low complexity CVEs in Microsoft Edge (2015–2018)

This analysis is likely moot, because in December 2018 Microsoft announced that they would be adopting the Chromium open source project for Edge development (Microsoft Corporation, n.d.). We'll have to wait for a few years to see how this change is reflected in the CVE data.

Let's examine Google Chrome next.

Google Chrome Vulnerability Trends

The Google Chrome browser was released in 2008, first on Windows and then later on other operating systems. There were 1,680 CVEs for Chrome between 2008 and the end of 2018, an average of 153 vulnerabilities per year. 68 vulnerabilities per year, on average, were rated critical or high severity (CVE Details, n.d.). As illustrated in *Figure 2.37*, there was a dramatic increase in CVEs for Chrome between 2010 and 2012. In the three 3 years between 2016 and the end of 2018, there was a 44% reduction in CVEs, and 100% reductions in low complexity CVEs, as well as critical and high severity CVEs.

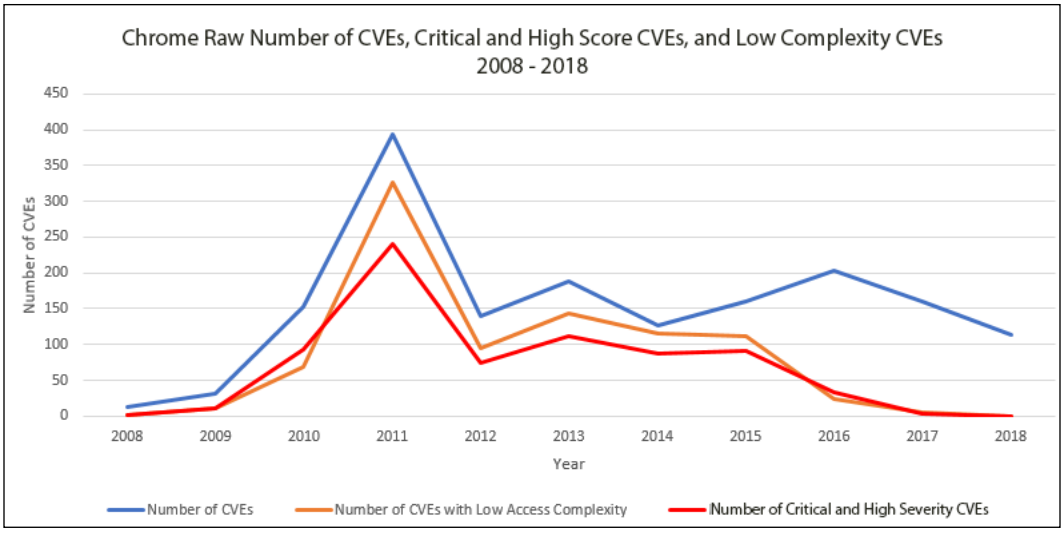


Figure 2.37: The number of CVEs, critical and high severity CVEs and low complexity CVEs in Google Chrome (2008–2018)

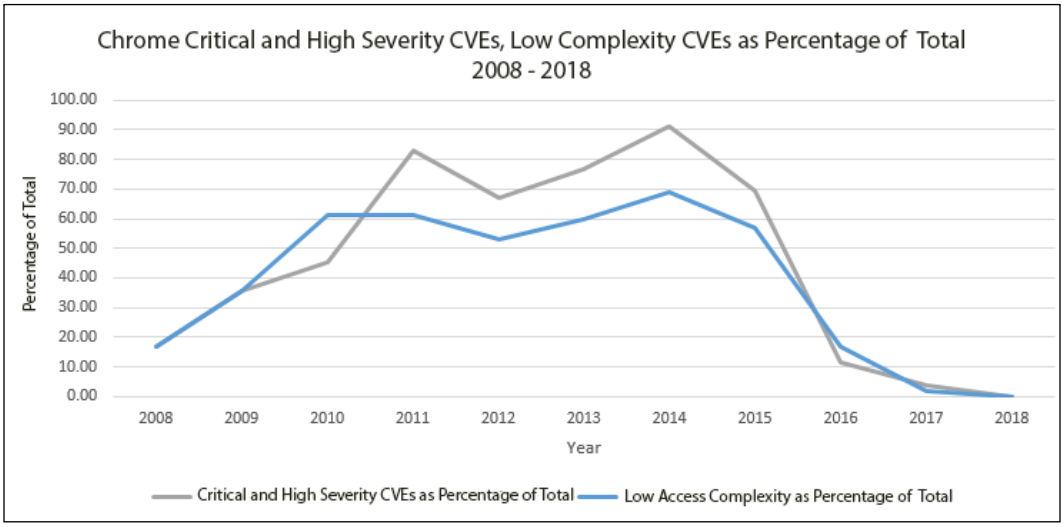


Figure 2.38: Critical and high severity rated CVEs and low complexity CVEs as a percentage total of all Google Chrome CVEs (2008–2018)

Chrome satisfies the criteria we have in our vulnerability improvement framework. Excellent work Google!

Mozilla Firefox Vulnerability Trends

Mozilla Firefox is a popular web browser that was initially released in 2002. CVEs started showing up in the NVD for it in 2003. Between 2003 and the end of 2018, there were 1,767 CVEs for Firefox, edging out Google Chrome for the browser with the most CVEs. Firefox had, on average, 110 CVEs per year during this period, 51 of which were rated critical or high severity (CVE Details, n.d.).

As illustrated by *Figure 2.39*, Firefox almost accomplished the aspirational goal of zero CVEs in 2017 when only a single CVE was filed in the NVD for it. Unfortunately, this didn't become a trend as 333 CVEs were filed in the NVD in 2018, an all-time high for Firefox in a single year. In the 3 years between 2016 and the end of 2018, CVEs increased by 150%, critical and high severity vulnerabilities increased by 326%, while low complexity CVEs increased by 841%. The number of CVEs decreased from 333 to a more typical 105 in 2019 (CVE Details, n.d.).

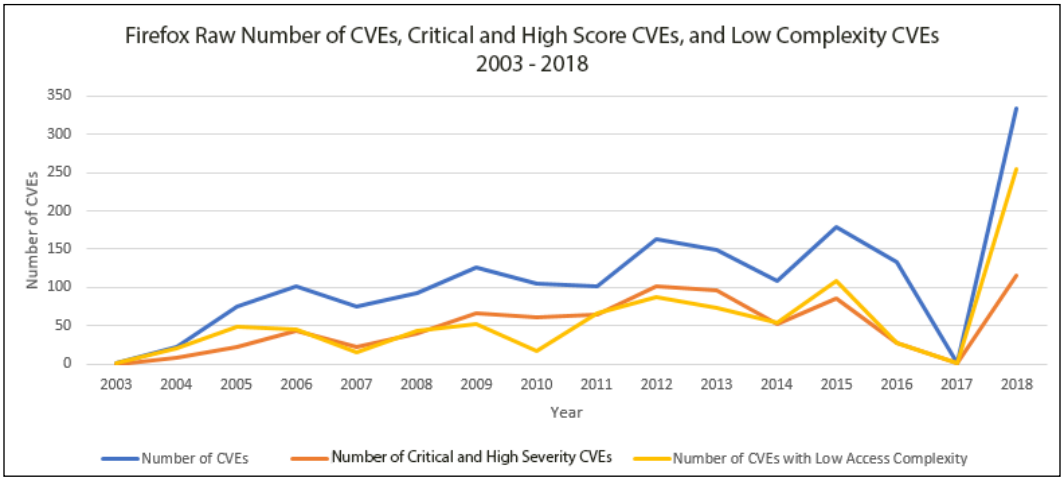


Figure 2.39: The number of CVEs, critical and high severity CVEs and low complexity CVEs in Firefox (2003–2018)

Had Mozilla been able to continue the trend in vulnerability disclosures that started in 2015, Firefox would have met the criteria for our vulnerability improvement framework. The spike in *Figure 2.40* in 2017 is a result of having a single CVE that year that was rated high severity with low access complexity (CVE Details, n.d.).

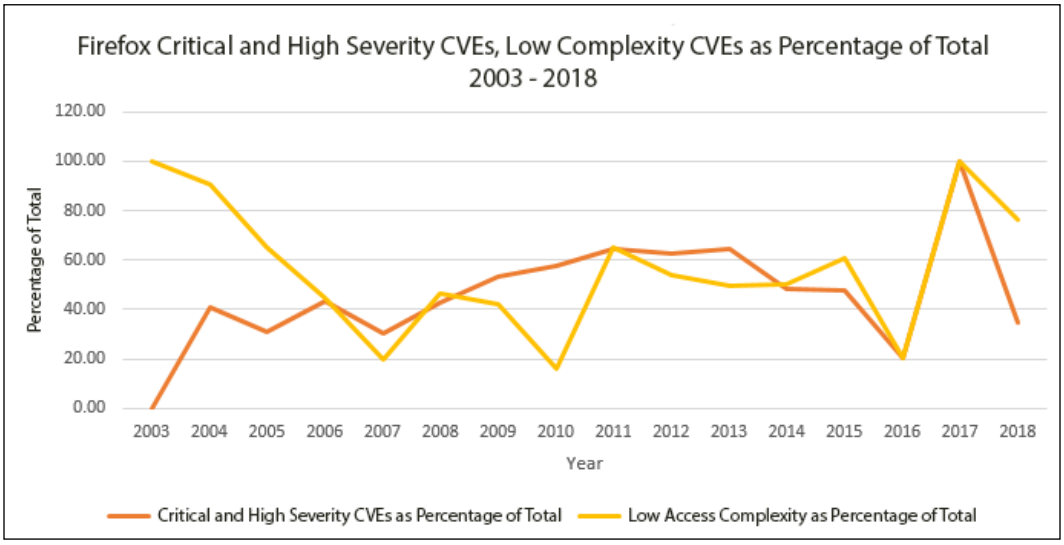


Figure 2.40: Critical and high severity rated CVEs and low complexity CVEs as a percentage total of all Firefox CVEs (2003–2018)

Apple Safari Vulnerability Trends

The last web browser I'll examine is Apple Safari. Apple initially released Safari in January 2003. On average, Safari had 60 vulnerabilities per year, with 17 CVEs rated critical or high per year on average. Between 2003 and the end of 2018 a total of 961 CVEs were disclosed in Safari.

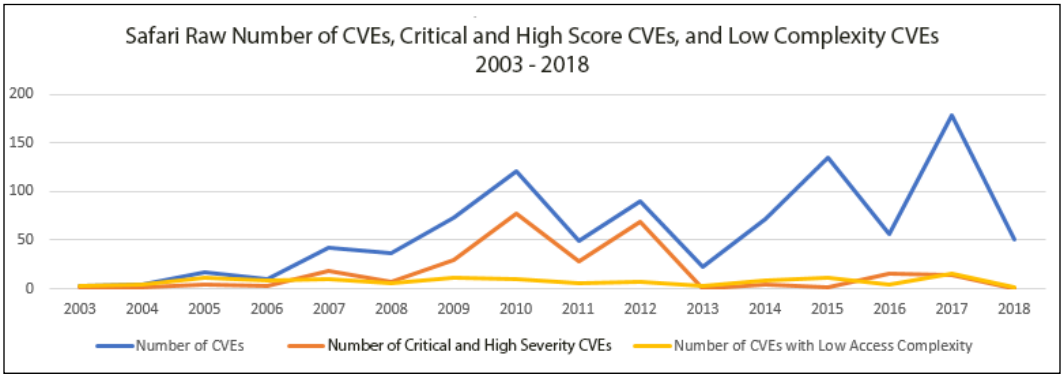


Figure 2.41: The number of CVEs, critical and high severity CVEs and low complexity CVEs in Apple Safari (2003–2018)

As illustrated by *Figure 2.41*, there were relatively large increases in CVEs in Safari in 2015 and 2017. Between 2016 and the end of 2018, there was an 11% decline in CVEs, a 100% decline in critical and high rated CVEs, and an 80% decline in low complexity vulnerabilities (CVE Details, n.d.). Apple once again meets the criteria of our vulnerability improvement framework.

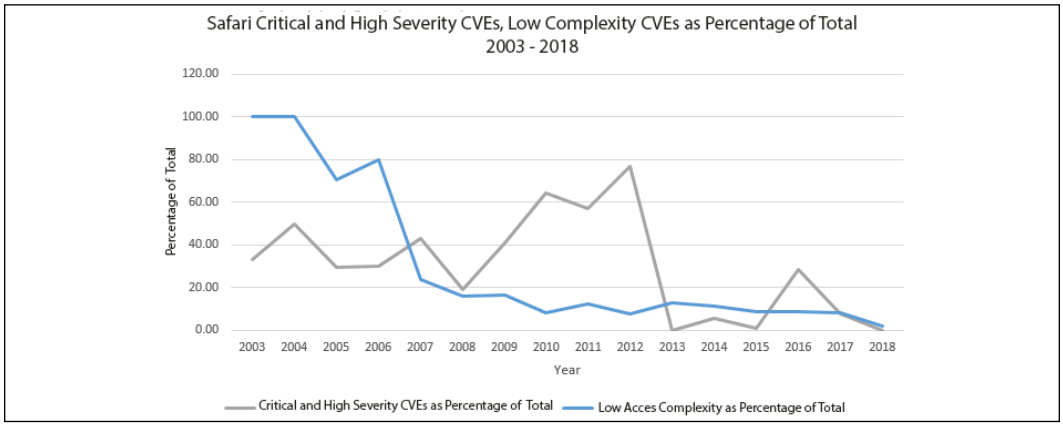


Figure 2.42: Critical and high severity rated CVEs, and low complexity CVEs as a percentage total of all Apple Safari CVEs (2003–2018)

Web Browser Vulnerability Trend Summary

Three of the web browsers that I examined met the goals of our vulnerability improvement framework. Microsoft Internet Explorer, Google Chrome, and Apple Safari all made the grade.

Browser	Fewer CVEs?	Fewer CVEs with CVSS Score 7-10?	Fewer Low Access Complexity CVEs?
Microsoft Internet Explorer	Yes	Yes	Yes
Microsoft Edge	No	No	Yes
Google Chrome	Yes	Yes	Yes
Mozilla Firefox	No	No	No
Apple Safari	Yes	Yes	Yes

Table 2.6: Results of applying the vulnerability improvement framework for the period (2014–2018)

Web Browser	Total Number of CVEs at the end of 2018	Average Number of CVEs/Year	Average Number of Critical and High CVEs/Year
Microsoft Edge	525	131	95
Apple Safari	961	60	17
Microsoft Internet Explorer	1597	80	57
Google Chrome	1680	153	68
Mozilla Firefox	1767	110	51

Table 2.7: Web browser vital statistics (1999–2018)

Table 2.7 provides a summary of some interesting CVE data for the web browsers we examined (CVE Details, n.d.). Apple Safari stands out based on the low number of average CVEs per year and an average number of critical and high severity CVEs that is well below the others.

After presenting this type of data and analysis on web browsers to people that are really passionate about their favorite browser, they are typically in disbelief, sometimes even angry, that their favorite browser could have so many vulnerabilities. Questions about the validity of the data and analysis usually quickly follow. Some people I've shared this type of data with also feel that the number of vulnerabilities in their least favorite browser has somehow been under-reported. It's like arguing about our favorite make of car! But remember that this data is imperfect in several respects. And there certainly is an opportunity to dive deeper into the data and analyze CVE trends for specific versions and service packs and releases to get a more granular view of differences between browsers. You can do this using the vulnerability improvement framework that I've provided in this chapter. But perhaps more importantly, remember that this data illustrates how the development and test processes of these vendors have performed over many years and whether they have been continuously improving.

After all, every version of IE was developed by Microsoft, and every version of Safari was developed by Apple, and so on. Their customers don't just use a version of their browsers; they use the outputs of their vendors' development, test, and incident response processes. The key question to answer is which of these vendors has managed their vulnerabilities in a way that lowers the costs to your organization while reducing risk. Let me finish this chapter by providing some general guidance on vulnerability management.

Vulnerability Management Guidance

A well-run vulnerability management program is critical for all organizations. As you've seen from the data and analysis in this chapter, there have been lots of vulnerabilities disclosed across the industry and the volumes have been increasing, not decreasing. At the end of 2019, there were over 122,000 CVEs in the NVD. Attackers know this and understand how challenging it is for organizations to keep up with the volume and complexity of patching the various hardware and software products they have in their environments. Defenders have to be perfect while attackers just have to be good or lucky once. Let me provide you with some recommendations regarding vulnerability management programs.

First, one objective of a vulnerability management program is to understand the risk that vulnerabilities present in your IT environment. This is not static or slow moving. Vulnerabilities are constantly being disclosed in all hardware and software. Because of this, data on the vulnerabilities in your environment gets stale quickly. The organizations that I have met that decided they would deploy security updates once per quarter, or every six months, have an unusually high appetite for risk; although, paradoxically, some of these same organizations tell me they have no appetite for risk. It is always interesting to meet people that believe their highest priority risks are their vendors, instead of the cadre of attackers who are actively looking for ways to take advantage of them. Attackers who, given the chance, will gladly encrypt all their data and demand a ransom for the decryption keys.

When I meet an organization with this type of policy, I wonder whether they really do have a data-driven view of the risk and whether the most senior layer of management really understands the risk that they are accepting on behalf of the entire organization.

Do they know that on average in 2019, 33.4 new vulnerabilities were disclosed per day, and in 2018, there were 45.4 disclosures per day? If they are patching quarterly, that is equivalent to 4,082 vulnerabilities potentially unpatched for up to 90 days in 2018 and 3,006 in 2019. Double those figures for organizations that patch semi-annually. On average, more than a third of those vulnerabilities are rated critical or high. Attackers only require one exploitable vulnerability in the right system to successfully initially compromise an environment. Instead of avoiding patching and rebooting systems to minimize disruption to their business, most of these organizations need to focus on building very efficient vulnerability management programs with the goal of reducing risk in a more reasonable amount of time. Attackers have a huge advantage in environments that are both brittle and unpatched for long periods.

For most organizations, my recommendation is that vulnerability management teams scan everything, every day. Read that line again if you have to. Remember the submarine analogy I used in the preface section of this book. Your vulnerability management program is one of the ways in which you look for defects in the hull of your submarine. Scanning every asset you have in your environment for vulnerabilities every day will help identify cracks and imperfections in the hull that, if exploited, would sink the boat. Scanning everything every day for vulnerabilities and misconfigurations provides the organization with important data that will inform their risk decisions. Without up-to-date data, they are managing risk in an uninformed way.

However, it's important to note that mobile devices, especially of the BYOD variety, pose a significant challenge to vulnerability management teams. Most organizations simply can't scan these devices the same way they scan other assets. This is one reason why many cyber security professionals refer to BYOD as "bring your own disaster". Instead, limiting mobile devices' access to sensitive information and HVAs is more common. Requiring newer operating system versions and minimum patch levels in order to connect to corporate networks is also common. To this end, most of the enterprises I've met with over the years leverage **Mobile Device Management** (**MDM**) or **Mobile Application Management** (**MAM**) solutions.

For some organizations, scanning everything every day will require more resources than they currently have. For example, they might require more vulnerability scanning engines than they currently have in order to scan 100% of their IT assets every day. They might also want to do this scanning in off hours to reduce network traffic generated by all this scanning during regular work hours. This might mean that they have to scan everything, every night during a defined number of hours. To accomplish this, they'll need a sufficient number of vulnerability scanning engines and staff to manage them. Once they have up-to-date data on the state of the environment, then that data can be used to make risk-based decisions; for example, when newly discovered vulnerabilities and misconfigurations should be addressed. Without up-to-date data on the state of the environment, hope will play a continual and central role in their vulnerability management strategy.

The data generated by all this vulnerability scanning is gold dust for CISOs, especially for security programs that are relatively immature. Providing the C-suite and Board of Directors with data from this program can help CISOs get the resources they need and communicate the progress they are making with their security program. Providing a breakdown of the number of assets in inventory, how many of them they can actually manage vulnerabilities on, the number of critical and high severity vulnerabilities present, and an estimate of how long it will take to address all these vulnerabilities, can help build an effective business case for more investment in the vulnerability management program. Providing senior management with quantitative data like this helps them understand reality versus opinion. Without this data, it can be much more difficult to make a compelling business case and communicate progress against goals for the security program.

The cloud can change the costs and effort related to vulnerability management in a dramatically positive way. I'll discuss this in *Chapter 8*, *The Cloud – A Modern Approach to Security and Compliance*.

Chapter summary

Hopefully, I didn't blind you with too much science in this chapter—there were a lot of numbers to digest! Allow me to recap some of the key take-aways for this chapter.

Risk is a combination of probability and impact. The Common Vulnerability Scoring System (CVSS) is used to estimate the risk for each vulnerability (CVE) in the National Vulnerability Database (NVD). This data should be used to inform your vulnerability management program. Using vendors who have been successful at reducing the number of vulnerabilities in their products can potentially reduce the time, effort, and costs related to your vulnerability management program. If you choose vendors who have also invested in reducing attackers' return on investment by making the exploitation of vulnerabilities in their products hard or impossible, you'll also be reducing your risk and costs.

Of the vendors examined in this chapter, only Apple met the criteria of our vulnerability improvement framework by reducing the number of vulnerabilities in their products, reducing the severity of vulnerabilities in their products, and reducing the number of low access complexity vulnerabilities (those with the highest risk) over the 5 years studied. The operating systems that I examined that achieved the objectives of our vulnerability improvement framework over a 3-year period were Linux Kernel and Apple macOS. The web browsers I examined with the best vulnerability management track record between 2016 and 2018 included Apple Safari, Google Chrome, and Microsoft Internet Explorer. The way vulnerabilities were managed in these browsers during these 3 years reduced the risk to their users.

Please keep in mind that the data used for these comparisons has many biases and is not complete or completely accurate. But you can do your own CVE research and use the informal "vulnerability improvement framework" I've provided.

Vulnerability management teams that scan everything, every day, provide the best visibility for their organizations to manage risk. Data from vulnerability management programs provide CISOs with some of the data they need to manage the performance of their security programs and steer future investments into the programs.

In the next chapter, we are going to dive into malware infection data from hundreds of millions of systems around the world to examine how the threat landscape has evolved over the years. Did you know that socio-economic factors, such as GDP, are related to regional malware infection rates? We are going to look at this as well. Additionally, I'm going to provide you with some tips and best practices for consuming threat intelligence.

References

1. Brian Martin, S. C. (December 3, 2013). *Black Hat USA 2013 - Buying into the Bias: Why Vulnerability Statistics Suck*. Retrieved from YouTube: https://www.youtube.com/watch?time_continue=20&v=3Sx0uJGRQ4s
2. Common Vulnerabilities and Exposures. (n.d.). *CVE Numbering Authorities*. Retrieved from Common Vulnerabilities and Exposures: https://cve.mitre.org/cve/cna.html
3. CVE Details. (January 1, 2020). *Top 50 Vendors By Total Number Of "Distinct" Vulnerabilities*. Retrieved from CVE Details: https://www.cvedetails.com/top-50-vendors.php
4. CVE Details. (n.d.). *Apple list of products*. Retrieved from CVE Details: https://www.cvedetails.com/product-list/vendor_id-49/Apple.html
5. CVE Details. (n.d.). *Apple Mac OS X vulnerability details*. Retrieved from CVE Details: https://www.cvedetails.com/product/156/Apple-Mac-Os-X.html?vendor_id=49
6. CVE Details. (n.d.). *Apple Safari vulnerability statistics*. Retrieved from CVE Details: https://www.cvedetails.com/product/2935/Apple-Safari.html?vendor_id=49
7. CVE Details. (n.d.). *Apple Vulnerability Statistics*. Retrieved from CVE Details: https://www.cvedetails.com/vendor/49/Apple.html
8. CVE Details. (n.d.). *Google Android vulnerability statistics*. Retrieved from CVE Details: https://www.cvedetails.com/product/19997/Google-Android.html?vendor_id=1224
9. CVE Details. (n.d.). *Google Chrome vulnerability details*. Retrieved from CVE Details: https://www.cvedetails.com/product/15031/Google-Chrome.html?vendor_id=1224
10. CVE Details. (n.d.). *Google List of Products*. Retrieved from CVE Details: https://www.cvedetails.com/product-list/vendor_id-1224/Google.html
11. CVE Details. (n.d.). *Google Vulnerability Statistics*. Retrieved from CVE Details: https://www.cvedetails.com/vendor/1224/Google.html
12. CVE Details. (n.d.). *How does it work?* Retrieved from CVE Details: https://www.cvedetails.com/how-does-it-work.php

13. CVE Details. (n.d.). *IBM List of Products*. Retrieved from CVE Details: https://www.cvedetails.com/product-list/product_type-/firstchar-/vendor_id-14/page-1/products-by-name.html?sha=6d92323b7a6590a46e9131e6e1f4a17a96434ea7&order=3&trc=1056
14. CVE Details. (n.d.). *IBM Vulnerability Statistics*. Retrieved from CVE Details: https://www.cvedetails.com/vendor/14/IBM.html
15. CVE Details. (n.d.). *Linux Kernel vulnerability statistics*. Retrieved from CVE Details: https://www.cvedetails.com/product/47/Linux-Linux-Kernel.html?vendor_id=33
16. CVE Details. (n.d.). *Microsoft Edge vulnerability statistics*. Retrieved from CVE Details: https://www.cvedetails.com/product/32367/Microsoft-Edge.html?vendor_id=26
17. CVE Details. (n.d.). *Microsoft Internet Explorer vulnerability details*. Retrieved from CVE Details: https://www.cvedetails.com/product/9900/Microsoft-Internet-Explorer.html?vendor_id=26
18. CVE Details. (n.d.). *Microsoft List of Products*. Retrieved from CVE Details: https://www.cvedetails.com/product-list/product_type-/firstchar-/vendor_id-26/page-1/products-by-name.html?sha=4b975bdf63b781745f458928790e4c8fd6a77f94&order=3&trc=525
19. CVE Details. (n.d.). *Microsoft Vulnerability Statistics*. Retrieved from CVE Details: https://www.cvedetails.com/vendor/26/Microsoft.html
20. CVE Details. (n.d.). *Mozilla Firefox vulnerability details*. Retrieved from CVE Details: https://www.cvedetails.com/product/3264/Mozilla-Firefox.html?vendor_id=452
21. CVE Details. (n.d.). *Mozilla Firefox vulnerability statistics*. Retrieved from CVE Details: https://www.cvedetails.com/product/3264/Mozilla-Firefox.html?vendor_id=452
22. CVE Details. (n.d.). *Oracle List of Products*. Retrieved from CVE Details: https://www.cvedetails.com/product-list/product_type-/firstchar-/vendor_id-93/page-1/products-by-name.html?sha=b4dc68699904240f1eab0f9453fb5a2f9213a78f&order=3&trc=644
23. CVE Details. (n.d.). *Oracle Vulnerability Statistics*. Retrieved from CVE Details: https://www.cvedetails.com/vendor/93/Oracle.html

24. CVE Details. (n.d.). *Top 50 Products By Total Number Of "Distinct" Vulnerabilities*. Retrieved from CVE Details: https://www.cvedetails.com/top-50-products.php
25. CVE Details. (n.d.). *Top 50 Products By Total Number Of "Distinct" Vulnerabilities*. Retrieved from CVE Details: https://www.cvedetails.com/top-50-products.php
26. CVE Details. (n.d.). *Windows 10 Vulnerability Details*. Retrieved from CVE Details: https://www.cvedetails.com/product/32238/Microsoft-Windows-10.html?vendor_id=26
27. CVE Details. (n.d.). *Windows 7 Vulnerability Statistics*. Retrieved from CVE Details: https://www.cvedetails.com/product/17153/Microsoft-Windows-7.html?vendor_id=26
28. CVE Details. (n.d.). *Windows Server 2012 Vulnerability Details*. Retrieved from CVE Details: https://www.cvedetails.com/product/23546/Microsoft-Windows-Server-2012.html?vendor_id=26
29. CVE Details. (n.d.). *Windows Server 2016 Vulnerability Details*. Retrieved from CVE Details: https://www.cvedetails.com/product/34965/Microsoft-Windows-Server-2016.html?vendor_id=26
30. CVE Details. (n.d.). *Windows XP Vulnerability Statistics*. Retrieved from CVE Details: https://www.cvedetails.com/product/739/Microsoft-Windows-Xp.html?vendor_id=26
31. Matt Miller, M. (February 14, 2019). *BlueHat IL 2019 - Matt Miller*. Retrieved from YouTube: https://www.youtube.com/watch?v=PjbGojjnBZQ
32. Microsoft Corporation. (January 2020). *Support for Windows 7 has ended*. Retrieved from Microsoft Corporation: https://www.microsoft.com/en-us/windows/windows-7-end-of-life-support-information
33. Microsoft Corporation. (n.d.). *Microsoft Edge: Building a safer browser*. Retrieved from Microsoft: https://blogs.windows.com/msedgedev/2015/05/11/microsoft-edge-building-a-safer-browser/#tFLjZDzG1LORHcy3.97

34. Microsoft Corporation. (n.d.). *Microsoft Edge: Making the web better through more open source collaboration*. Retrieved from Microsoft: https://blogs.windows.com/windowsexperience/2018/12/06/microsoft-edge-making-the-web-better-through-more-open-source-collaboration/#53oueSHZ9BtuhB1G.97
35. Microsoft. (n.d.). *Security Engineering*. Retrieved from Microsoft: https://www.microsoft.com/en-us/securityengineering/sdl
36. NIST. (n.d.). Retrieved from National Vulnerability Database: https://nvd.nist.gov/vuln
37. NIST. (n.d.). *Common Vulnerability Scoring System Calculator*. Retrieved from National Vulnerability Database: https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator
38. NIST. (n.d.). *CVE-2018-8653 Detail*. Retrieved from National Vulnerability Database: https://nvd.nist.gov/vuln/detail/CVE-2018-8653
39. NIST. (n.d.). *Vulnerability Metrics*. Retrieved from National Vulnerability Database: https://nvd.nist.gov/vuln-metrics/cvss
40. Ribeiro, R. (n.d.). *Understanding the Security Benefits of Windows 10*. Retrieved from BizTech: https://biztechmagazine.com/article/2016/04/understanding-security-benefits-windows-10
41. Wikipedia. (n.d.). *Common Vulnerability Scoring System*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Common_Vulnerability_Scoring_System
42. Wikipedia. (n.d.). *Project Zero*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Project_Zero

3

The Evolution of the Threat Landscape – Malware

I have always thought of malware as a synonym for "attackers' automation." Purveyors of malware seek to compromise systems for a range of motivations, as I described in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*. Any system that sends and receives email, surfs the web, or takes other forms of input can be attacked, regardless of whether it was manufactured in Redmond, Raleigh, Cupertino, Helsinki, or anywhere else. The AV-TEST Institute, one of the world's premier independent anti-virus testing labs, based in Germany, has one of the world's largest malware collections. (AV-Test Institute, 2020) They have accumulated this collection over 15 years. "Every day, the AV-TEST Institute registers over 350,000 new malicious programs (malware) and potentially unwanted applications (PUA)" (AV-Test Institute, 2020). The statistics that they have published indicate that the volume of total malware has increased every year between 2011 and 2019, starting that period with 65.26 million malware samples detected and ending it with 1.04032 billion (a 16x increase) (AV-Test Institute, 2020). According to the data that AV-Test has published in their annual security reports, the share of malware developed for Windows operating systems was 69.96% in 2016 (AV-Test Institute, 2017), 67.07% in 2017 (AV-Test Institute, 2018), and 51.08% in 2018 (AV-Test Institute, 2019).

The operating system with the next highest share of malware samples in these years was Google Android, with less than 7% of the share in every year reported (AV-Test Institute, 2020). The number of new malware samples detected for Linux operating systems was 41,161 in March of 2019 (the latest data available), while malware samples for Windows during the same time was 6,767,397 (a 198% difference) (AV-Test Institute, 2019). Malware samples for macOS during this month surged to 11,461 from 8,057 the month before (AV-Test Institute, 2019).

This data clearly suggests that the platform of choice for malware authors is the Windows operating system. That is, more unique malware is developed to attack Windows-based systems than any other platform. Once Windows systems are compromised, attackers will typically harvest software and game keys, financial information such as credit card numbers, and other confidential information they can use to steal identities, sometimes taking control of the system and its data for ransom. Many attackers will use compromised systems as platforms to perpetrate attacks from using the anonymity that the compromised systems provide to them.

Given that attackers have been targeting and leveraging Windows-based systems more than any other platform, and given the ubiquity of Windows, security experts need to understand how and where attackers have been using these systems. CISOs, aspiring CISOs, security teams, and cybersecurity experts can benefit from understanding how Windows-based systems are attacked, in at least a few ways:

- CISOs and security teams that are responsible for Windows systems in their environment should understand how attackers have been attacking Windows-based systems with malware, as well as how this has evolved over time:
    - Being knowledgeable about malware will help security teams do their jobs better.
    - This knowledge can be useful to help recognize the fear, uncertainty, and doubt that some security vendors use to sell their products and services; understanding how attackers have been using malware will help CISOs make better security-related investments and decisions.

- CISOs and security teams that are responsible for Linux-based systems, and other non-Microsoft operating systems, should have some insight into how their adversaries are compromising and using Windows systems to attack them. Attackers don't care if the tech they compromise was developed in Redmond, Raleigh, Cupertino, or China; we can take lessons from the Windows ecosystem, which also applies to Linux-based systems and other platforms and learn from them. Very often, the methods that malware authors use on the Windows platform will be adapted to attack other platforms, albeit usually on a smaller scale. Understanding malware authors' methods is important for security teams, regardless of the types of systems they protect. Unfortunately, CISOs don't get to tune out of Windows-based threats, even if they don't use Windows in their environments.
- Finally, in my opinion, it's hard for cybersecurity subject matter experts to use that moniker if they are blissfully unaware of malware trends in an online ecosystem consisting of over a billion systems that supports more than half of all the malware in the world. It doesn't matter if there are more mobile devices, more IoT devices, or more secure operating systems. It is undeniable that Windows is everywhere. Subsequently, all cybersecurity experts should know a little about the largest participant in the global threat landscape.

This chapter will provide a unique, detailed, data-driven perspective of how malware has evolved around the world over the past decade, and in some cases, I will provide data for longer periods. There are some very interesting differences in regional malware encounter rates and infection rates that I'll also dive into in this chapter. This view of the threat landscape will help CISOs and security teams understand how the malware threats they face have changed over time. Not only is this data super interesting, but it can help take some of the fear, uncertainty, and doubt out of conversations about malware and how to manage the risks it poses.

I'll also give you some pointers on how to spot good threat intelligence versus the nonsense I see so often in the industry today; after publishing thousands of pages of threat intelligence during my time at Microsoft, I have a few tips and tricks to share with you that I think you'll appreciate.

Throughout this chapter, we'll cover the following topics:

- Some of the sources of data that threat intelligence for Windows comes from
- Defining malware categories and how their prevalence is measured
- Global malware evolution and trends
- Regional malware trends for the Middle East, the European Union, Eastern Europe and Russia, Asia, as well as North and South America
- How to identify good threat intelligence

Before I introduce you to the data sources I used for this chapter, let's begin with an interesting and hopefully somewhat entertaining story.

Introduction

In 2003, when I worked on Microsoft's customer-facing incident response team, we began finding user mode rootkits on compromised systems with some regularity, so much so that one of our best engineers built a tool that could find user mode rootkits that were hiding from Windows. A user mode rootkit runs like any other application that a normal user would run, but it hides itself. Then, one day, we received a call from a Microsoft support engineer who was helping troubleshoot an issue that a customer had on an Exchange email server. The symptom of the problem was that once every few days, the server would blue screen. The support engineer couldn't figure out why and was doing a remote debug session, trying to find the code that caused the server to blue screen. It took weeks, but once he found the code responsible for the blue screen, he couldn't explain what the code was, nor how it was installed on the server. This is when he called us for help.

When the sever blue screened and rebooted, this enabled us to look at a partial memory dump from the system. After a few days of analysis, we determined that the server was compromised in a way we had never seen before. A device driver on the system was hiding itself and other components. We had found the first kernel mode rootkit that we had ever seen in the wild.

This was a big deal. Unlike a user mode rootkit, developing and installing a kernel mode rootkit required incredible expertise. This is because this type of rootkit runs in the most privileged part of the operating system, which few people really understand. At the time, although the concept of kernel mode rootkits was discussed among security experts, finding one installed on a server running in an enterprise's production environment signaled that attackers were becoming far more sophisticated than they had been in the past. Graduating from user mode rootkits to kernel mode rootkits was a major leap forward in the evolution of malware.

To our incident response team, this was a call to action. We had to let the Windows kernel developers at Microsoft know that the thing that makes Windows a trusted computing base, its kernel, was being directly attacked by sophisticated authors of malware. Until then, a kernel mode rootkit running in the wild was mythical. But now, we had evidence that these rootkits were real and were being used to attack enterprise customers. We scheduled a meeting with the lead developers, testers, and program managers on the Windows Kernel development team. We gathered in a room used for training, with an overhead projector, so that we could walk the developers through the memory dump we had from the compromised server to show them how the rootkit worked. We provided them with some context about the server, such as where it was running, the operating system version, the service pack level, a list of all the applications running on the sever, and so on. We answered numerous questions about how we debugged the source of the blue screen, found the hidden driver, and discovered how it worked.

At first, the Windows Kernel team was completely skeptical that we had found a kernel mode rootkit running on a Windows server. But after we presented all the evidence and showed them the debug details, they gradually came to accept the fact that it was a kernel mode rootkit. Our team expected adulation and respect for all the very technical work we had done, as well as our expertise on Windows kernel internals that allowed us to make this discovery. Instead, the kernel developers told us that our tools and our methods were as bad as the malware authors. They warned us to stop using our tools to find rootkits as the tools could make the Windows systems they ran on unstable unless rebooted. Finally, they offered to do nothing to harden the kernel to prevent such attacks in the future. It was a disappointing meeting for us, but you can't win them all!

After the successful large-scale worm attacks of 2003 and 2004, this tune changed. The entire Windows team stopped the development work they were doing on what would later become Windows Vista. Instead, they worked on improving the security of Windows XP and Server 2003, releasing Windows XP Service Pack 2 and Windows Server 2003 Service Pack 1. There was even talk of a new version of Windows, code-named Palladium, that had a security kernel to help mitigate rootkits like the one we discovered, but it never came to pass (Wikipedia, n.d.). Ultimately, our work on detecting kernel mode rootkits did help drive positive change as future 64-bit versions of Windows would not allow kernel mode drivers, like the one we discovered, to be installed unless they had a valid digital signature.

Later in my career at Microsoft, I had the chance to work with world-class malware researchers and analysts in Microsoft's anti-malware research and response lab, who were protecting a billion systems from millions of new malware threats. Malware like the kernel mode rootkit we had discovered 4 or 5 years earlier was now a commodity. Attackers were using large-scale automation and server-side polymorphism to create millions of unique pieces of malware every week. To win this war, the anti-virus industry was going to have to have bigger and better automation than large scale purveyors of commodity malware, which has proven to be surprisingly difficult to accomplish.

Why is there so much malware on Windows compared to other platforms?

There are certainly more mobile internet-connected devices today than there are Windows-based systems. Mobile device adoption exploded as Apple, Google, Samsung, and others brought very popular products to the global marketplace. But if there are far more mobile devices, shouldn't there be far more families of malware developed for those platforms?

The answer to this question lies in how applications get distributed in these ecosystems. Apple's App Store was a game-changer for the industry. Not only did it make it easy for iPhone users to find and install applications, but it almost completely eliminated malware for iOS-based devices.

Apple was able to accomplish this by making the App Store the one and only place consumers could install applications from (jailbreaking aside). **Independent Software Vendors** (**ISVs**) who want to get their apps onto consumers' iOS-based devices, such as iPhones and iPads, need to get their apps into Apple's App Store. To do this, those apps need to meet Apple's security requirements, which they verify behind the scenes. This makes the App Store a perfect choke point that prevents malware from getting onto Apple devices.

By contrast, Microsoft Windows was developed in more naive times, when no one could predict that, one day, there would be more malicious files in the Windows ecosystem than legitimate files. One of the big advantages of Windows, for developers, was that they could develop their software for Windows and sell it directly to consumers and businesses. This model was the predominant software distribution model for PCs for decades. Since software can be installed without regard for its provenance, and with limited ability to determine its trustworthiness, malware flourished in this ecosystem and continues to do so. Microsoft has taken numerous steps over the decades to combat this "side effect" of this software distribution model, with limited success.

Some would argue that the Android ecosystem has ended up somewhere in between these two extremes. Google also has an app store, called Google Play. Google has also taken steps to minimize malware in this app store. However, third-party app stores for Android-based devices didn't all maintain Google's high security standards, subsequently allowing malware for these devices to get into the ecosystem. But, as I mentioned earlier, the number of malware samples detected for Android-based devices is many times smaller than that of Windows-based devices.

These differences in software distribution models, at least partially, help to explain why there is so much more malware developed for Windows than other platforms. Cybersecurity professionals can take some lessons from this into their own IT environments. Controlling how software is introduced to an enterprise IT environment can also help minimize the amount of malware in it. This is one advantage of leveraging **Continuous Integration** (**CI**)/**Continuous Deployment** (**CD**) pipelines. CI/CD pipelines can help enterprises build their own app store and restrict how software is introduced into their environments.

Now that we've briefly discussed how software distribution models can impact the distribution of malware, let's dive deep into malware. Security teams can learn a lot from studying malware developed for Windows operating systems, even if they don't use Windows themselves. The methods that malware authors employ on Windows can and are used for malware developed for many different platforms, including Linux. Studying how malware works in the largest malware ecosystem can help us defend against it almost everywhere else. But before I dive right into the malware trend data, it's important for you to understand the sources of the data that I'm going to show you. Threat intelligence is only as good as its source, so let's start there.

Data sources

The primary source for the data in this chapter is the Microsoft Security Intelligence Report (Microsoft Corporation, n.d.). During my time working with the researchers and analysts in the **Microsoft Malware Protection Center** (**MMPC**), I was the executive editor and a contributor to the Microsoft Security Intelligence Report, which we called "*the SIR*." During the 8 or 9 years I helped produce the SIR, we published more than 20 volumes and special editions of this report, spanning thousands of pages. I gave literally thousands of threat intelligence briefings for customers around the world, as well as press and analyst interviews. I have read, re-read, and re-re-read every page of these reports—I know the ins and outs of this data very well.

The data in these reports comes from Microsoft's anti-malware products, including the Malicious Software Removal Tool, Microsoft Safety Scanner, Microsoft Security Essentials, Microsoft System Center Endpoint Protection, Windows Defender, Windows Defender Advanced Threat Protection, Windows Defender Offline, Azure Security Center, and the SmartScreen filter built into Microsoft web browsers. Other non-security products and services that provide valuable data for volumes of this report include Exchange Online, Office 365, and Bing. Let me explain in more detail how this eclectic group of data sources helps paint a well-rounded picture of the threat landscape.

The Malicious Software Removal Tool

The **Malicious Software Removal Tool** (**MSRT**) is an interesting tool that provides valuable data (Microsoft Corporation, n.d.). In the wake of the Blaster worm attacks (there were variants) (Microsoft Corporation, n.d.) in the summer of 2003, Microsoft developed a free "Blaster Removal Tool" designed to help customers detect and remove the Blaster worm and its variants (Leyden). Remember that, at this time, very few systems ran up-to-date, real-time anti-virus software. The Blaster Removal Tool was free. This tool made a huge difference as tens of millions of systems ran it. Because of the tool's success and the constant barrage of malware attacks that followed it in history, such as Sasser, MyDoom, and many others, and the fact that so few systems had anti-virus software running, Microsoft decided to release a "malicious software removal tool" every month. The MSRT was born.

It was meant to be a way to detect infected systems and clean the most prevalent or serious malware threats from the entire Windows ecosystem. Microsoft's anti-malware lab decides what new detections to add to the MSRT every month. A list of all the malware it detects is published on Microsoft's website (Microsoft Corporation). Between January 2005 and October 2019, there were 337 malware families added in the detections for the MSRT. Keep in mind that there are at least hundreds of thousands, if not millions, of known malware families, so this is a very small subset of the total that real-time anti-malware software packages detect. The MSRT has been released monthly (more or less) with security updates on "Patch Tuesday," the second Tuesday of every month. It gets automatically downloaded from Windows Update or Microsoft Update to every Windows system in the world that has opted to run it. During the time I was publishing data from the MSRT in the SIR, the MSRT was running on hundreds of millions of systems per month on average.

Once the EULA is agreed to, the MSRT runs silently without a user interface as it's a command-line tool. If it doesn't find any malware infections, it stops execution and is unloaded from memory. No data is sent back to Microsoft in this case. But if malware is detected by the MSRT, then it will try to remove the malware from the system and report the infection to the user and to Microsoft. In this case, data is sent back to Microsoft.

Microsoft publishes the specific list of data fields that the MSRT sends back for analysis, including the version of Windows that the malware was detected on, the operating system locale, and an MD5 hash of the malicious files removed from the system, among others (Microsoft Corporation, n.d.). Administrators can download the MSRT and run it manually; the MSRT can also be configured not to send data back to Microsoft. Most enterprises that I talked to that ran the MSRT typically blocked data sent to Microsoft at their firewall. Subsequently, my educated guess is that 95% or more of the hundreds of millions of systems returning MSRT data to Microsoft are likely consumers' systems.

The MSRT provides a great post malware exposure snapshot of a small list of known, prevalent malware that is infecting consumers' systems around the world. When Microsoft's anti-malware lab adds a detection to the MSRT for a threat that's very prevalent, we should expect to see a spike in detections for that malware family in the data. This happens from time to time, as you'll see in the data. Keep in mind that the infected systems might have been infected for weeks, months, or years prior to the detection being added to the MSRT. Since the MSRT runs on systems all over the world and it returns the Windows locale and country location of infected systems, it provides us with a way to see regional differences in malware infections. I will discuss this in detail later in this chapter.

Real-time anti-malware tools

Unlike the MSRT, which cleans Windows-based systems that have already been successfully infected with prevalent malware, the primary purpose of real-time, anti-malware software is to block the installation of malware. It does this by scanning incoming files, monitoring systems for tell-tale signs of infection, scanning files when they are accessed, and periodically scanning storage. Real-time anti-malware software can also find pre-existing infections on systems when the real-time anti-malware package is initially installed. Real-time anti-malware software typically get signature and engine updates periodically (daily, weekly, monthly, and so on). This helps it block emerging threats but also threats it didn't previously know existed.

For example, if detection is added for a malware threat, but that malware threat has already successfully infected systems that are running the real-time anti-malware software, the update enables the anti-malware software to detect, and hopefully remove, the existing infection.

My point is that data from real-time anti-malware software provides us with a different view of the threat landscape compared to MSRT. Microsoft Security Essentials, Microsoft System Center Endpoint Protection, Windows Defender, and Windows Defender Advanced Threat Protection are all examples of real-time anti-malware software that are data sources. Windows Defender is the default anti-malware package for Windows 10-based systems, which now runs on over half of all personal computers in the world (Keizer, Windows by the numbers: Windows 10 resumes march toward endless dominance). This means that Windows Defender could be potentially running on hundreds of millions of systems around the world, making it a great source of threat intelligence data.

During some of the threat intelligence briefings I've done, some attendees asserted that this approach only provides a view of malware that Microsoft knows about. But this isn't quite true. The major anti-malware vendors share information with each other, including malware samples. So, while the first anti-malware lab that discovers a threat will have detections for that threat before anyone else, over time, all anti-malware vendors will have detections for it. Microsoft manages several security information sharing programs, with the goal of helping all vendors better protect their shared customers (Microsoft Corporation, 2019).

Although Internet Explorer and Microsoft's Edge web browsers don't have as large a market share as some of the other web browsers available, the SmartScreen filter built into these browsers gives us a view of malware hosted on the web (Microsoft Corporation). SmartScreen is like anti-malware software for the browser. As users browse the web, SmartScreen will warn them about known malicious websites they try to visit and scans files that are downloaded in the browser looking for malware. The data on sites hosting malicious software, and the malicious files themselves, can give us a view of the most common threats hosted on the web, as well as where in the world threats are hosted most and the regions that the victim populations are in.

Non-security data sources

Sources of data, such as email services and internet search services, can provide an additional dimension to threat intelligence. For example, data from Office 365 and Outlook.com provides visibility of the threats that flow through email, including the sources and destinations of these threats and their volumes. The volume of data that Microsoft has from Office 365 is mind-boggling, with hundreds of billions of email messages from customers all over the world flowing through it every month (Microsoft Corporation, 2018).

Bing, Microsoft's internet search engine service, is also a rich source of threat intelligence data. As Bing indexes billions of web pages so that its users can get quick, relevant search results, it's also looking for drive-by download sites, malware hosting sites, and phishing sites. This data can help us better understand where in the world malware is being hosted, where it moves to over time, and where the victims are.

When data from some select non-security data sources is combined with data from some of the security sources of data I discussed previously, we can get a more rounded view of the threat landscape. Office 365 and Outlook.com receive emails sent from all sorts of non-Microsoft clients and email servers, and Bing indexes content hosted on all types of platforms. Certainly, the combination of this data does not provide us with perfect visibility, but the scale of these data sources gives us the potential for good insights.

Now that you know where I'm getting malware-related data from, let's take a quick look at the different categories of malware that are included in the data and analysis.

About malware

Before we dive into the threat data, I need to provide you with some definitions for terms I'll use throughout the rest of this chapter.

Malicious software, also known as malware, is software whose author's intent is malicious. The developers of malware are trying to impede the confidentiality, integrity, and/or accessibility of data and/or the systems that process, transmit, and store it.

As I discussed in *Chapter 1, Ingredients for a Successful Cybersecurity Strategy*, malware authors can be motivated by many different things, including hubris, notoriety, military espionage, economic espionage, and hacktivism.

Most malware families today are blended threats. What I mean by this is that many years ago, threats were discrete—they were either a worm or a backdoor, but not both. Today, most malware has characteristics of multiple categories of malware. Analysts in anti-malware labs that reverse-engineer malware samples typically classify malware by the primary or most prominent way each sample behaves.

For example, a piece of malware might exhibit characteristics of a worm, a Trojan, and ransomware. An analyst might classify it as ransomware because that's its dominant behavior or characteristic. The volume of threats has grown dramatically over the years. Malware researchers in major anti-malware labs generally don't have time to spend weeks or months researching one malware threat, as they might have done 20 years ago. However, I have seen analysts in CERTs or boutique research labs do this for specific sophisticated threats found in their customer's environments. Protecting vast numbers of systems from an ever-growing volume of serious threats means that some major anti-virus labs are spending less time researching, as well as publishing, detailed findings on every threat they discover. Also, most enterprise customers are more interested in blocking infections or recovering from infections as quickly as possible and moving on with business, versus diving into the inner workings of malware *du jour*.

Generally speaking, malware research and response is more about automation and science now than the art it once was. Don't get me wrong; if you can understand how a piece of malware spreads and what its payload is, then you can more effectively mitigate it. But the volume and complexity of threats seen today will challenge any organization to do this at any scale. Instead, security teams typically must spend time and resources mitigating as many malware threats as possible, not just one popular category or family. As you'll see from the data I will provide in this chapter, some attackers even use old-school file infectors (viruses).

How malware infections spread

Malware isn't magic. It must get into an IT environment somehow. Hopefully, you'll remember the cybersecurity usual suspects, that is, the five ways that organizations are initially compromised, which I wrote about in detail in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*. To refresh your memory, the cybersecurity usual suspects are:

- Unpatched vulnerabilities
- Security misconfigurations
- Weak, leaked, and stolen credentials
- Social engineering
- Insider threats

Malware threats can use all the cybersecurity usual suspects to compromise systems. Some malware is used to initially compromise systems so that threat actors achieve their objectives. Some malware is used in IT environments, after the environment has already been compromised. For example, after attackers use one or more of the cybersecurity usual suspects to initially compromise a network, then they can use malware that will encrypt sensitive data and/or find cached administrator credentials and upload them to a remote server. Some malware is sophisticated enough to be used for both initial compromise and post-compromise objectives. As I mentioned earlier, I have always thought of malware as a synonym for "attackers' automation." Instead of the attacker manually typing commands or running scripts, malware is a program that performs the illicit activities for the attacker, autonomously or in a semiautonomous fashion. Malware helps attackers achieve their objectives, whether their objective is destruction and anarchy, or economic espionage.

The categories of malware I'll discuss in this chapter include Trojans, backdoor Trojans, Trojan downloaders and droppers, browser modifiers, exploits, exploit kits, potentially unwanted software, ransomware, viruses, and worms. Microsoft provides definitions for these categories of malware and others (Microsoft Corporation, n.d.). Your favorite anti-malware provider or threat intelligence provider might have different definitions than these. That's perfectly OK, but just keep in mind that there might be some minor nuanced differences between definitions. I'll provide you with my own, less formal, definitions to make this chapter easier to read and understand.

Trojans

I'll start with Trojans since, worldwide, they have been the most prevalent category of malware for the last decade. A Trojan relies on social engineering to be successful. It's a program or file that represents itself as one thing when really it is another, just like the Trojan horse metaphor that it's based on. The user is tricked into downloading it and opening or running it. Trojans don't spread themselves using unpatched vulnerabilities or weak passwords like worms do; they have to rely on social engineering.

A backdoor Trojan is a variation of this. Once the user is tricked into running the malicious program (scripts and macros can be malicious too), a backdoor Trojan gives attackers remote access to the infected system. Once they have remote access, they can potentially steal identities and data, steal software and game keys, install software and more malware of their choice, enlist the infected system into botnets so that they can do "project work" for attackers, and so on. Project work can include extortion, **Distributed Denial of Service** (**DDoS**) attacks, storing and distributing illicit and questionable content, or anything else the attackers are willing to trade or sell access to their network of compromised systems for.

Trojan downloaders and droppers are yet another variation on this theme. Once the user is tricked into running the malicious program, the Trojan then unpacks more malware from itself or downloads more malicious software from remote servers. The result is typically the same—malicious servitude and harvesting the system for all that it is worth. Trojan downloaders and droppers were all the rage among attackers in 2006 and 2007, but have made dramatic appearances in limited time periods since then. A great example of a Trojan downloader and dropper is the notorious threat called **Zlob**. Users were tricked into installing it on their systems when visiting malicious websites that had video content they wanted to view. When they clicked on the video file to watch it, the website told them they didn't have the correct video codec installed to watch the video. Helpfully, the website offered the video codec for download so that the user could watch the video. The user was really downloading and installing Zlob (Microsoft Corporation, 2009). Once installed, it would then expose the user to pop-up advertisements for free "security software" that would help them secure their system. Users that clicked on the ads to download and install the security software were giving the attackers more and more control over their systems.

Potentially unwanted software

While I am discussing threats that use social engineering, another near-ubiquitous threat category is called *potentially unwanted software*, also known by the names *potentially unwanted applications*, *potentially unwanted programs*, and a few others. Why does this category have so many seemingly unassuming names? This is a category of threats that lawyers invented. That's not necessarily a bad thing—it really is an interesting threat category. There are some shades of gray in malware research, and this category exposes this.

Let me give you a hypothetical example of potentially unwanted software that isn't based on any real-world company or organization. What would happen if a legitimate company offered consumers a free game in exchange for monitoring their internet browsing habits, all so that they could be targeted more accurately with online advertising? I think most people I know would think that's creepy and not give up their privacy in exchange for access to a free game. But if this privacy trade-off was only listed in the free game's **End User License Agreement** (**EULA**), where very few people would read it, how many people would simply download the free game and play it? In this case, let's say the free game ended up as a malware sample in an anti-malware company's threat collection. The analysts in the anti-malware lab could decide that the game company wasn't being transparent enough with the game's users, and categorize the game as a Trojan. The anti-malware company would then update the signatures for their anti-malware products to detect this new threat. The anti-malware company's anti-malware solution would then detect and remove the game from every system where it was running. Did the anti-malware company help its customers by removing the game and its ability to track their internet browsing habits? Or did it damage a legitimate company's business by deeming their product as malware and removing it from their customers' systems without permission?

The answer that the anti-malware industry came up with was to call it "Potentially Unwanted Software" (or a similar such name), flag it for users when it's detected, and ask the users to explicitly approve or disapprove its removal. This way, the game company's customer decides whether they want to remove the game company's product, not the anti-malware company. This helps mitigate the predictable damage claims and litigation that the anti-malware industry faces with potentially unwanted software.

Many, many variations of the example I described here are being offered on the internet today and are installed on systems all over the world. Some of them are legitimate companies with legitimate businesses, while others are threat actors pretending to be legitimate companies with legitimate products. Some families of this threat category start off as legitimate programs, but later turn malicious when their supply chain is compromised, or their operators turn malevolent. Other examples of this category include fake anti-virus software, fake browser protector software, software bundles that contain a bunch of different software offerings and components, and so on. My advice and mantra for many years has been, don't trust the software if you don't trust the people who wrote it. You'll see potentially unwanted software appear prominently in the threat data of this chapter.

Exploits and exploit kits

Next, let's look at exploits and exploit kits. *Chapter 2*, *Using Vulnerability Trends to Reduce Risk and Cost*, was dedicated to the topic of vulnerabilities. Remember that a vulnerability can allow an attacker to compromise the confidentiality, integrity, or availability of hardware or software. Exploits are malware that take advantage of vulnerabilities. You might also remember from my discussion of vulnerabilities in *Chapter 2* that not all vulnerabilities are the same. Some vulnerabilities, if exploited, have a higher potential impact on the system than others. Exploits for critical rated vulnerabilities are highly sought after by attackers. This is because they give attackers the best chance to take full control of the vulnerable system and run arbitrary code of their choice. That arbitrary code can do anything that the user context it is running in can do. For example, it can download more malware from servers on the internet that will enable attackers to remotely control the system, steal identities and data, enlist the system into a botnet, and so on.

Working exploits for vulnerabilities in web browsers, operating systems, and file parsers (for file formats like .pdf, .doc, .xlsx, and so on) can be worth a lot of money because of the ubiquity of these products. Subsequently, a sophisticated marketplace has developed over the last two decades around the supply and demand for exploits. Some examples of vulnerabilities that were used in attacks, according to Microsoft's research, include CVE-2017-0149 and CVE-2017-0005 (Microsoft Corporation, 2017).

Exploits must be delivered to their target. They can be delivered in several different ways, some of which rely on social engineering to succeed. For example, an attacker might deliver an exploit by developing a malformed .pdf file designed to exploit a specific unpatched vulnerability in a parser like Adobe Reader or Microsoft Word.

When a victim opens the .pdf file with a parser that isn't patched for the vulnerability that the attacker is using, and if no other mitigations are in place, then the vulnerability is exploited on the system, potentially running arbitrary code of the attacker's choice. But how does the attacker get the victim to run the exploit? One way is social engineering. The malformed .pdf file can be sent to the victim via an email, with the sender masquerading as a co-worker or friend of the victim. Since the victim trusts their co-worker or friend, they open the email attachment and the exploit is executed. Exploits can be hosted on web pages as downloads for victims, sent via social networks, and distributed on USB drives and other removal media.

An exploit kit is a library of exploits with some management software that makes it easier for attackers to manage attacks that use exploits. A kit's exploit library can contain any number of exploits for any number of products. An exploit kit might also provide attackers with web pages that make it easy to deliver the exploits in its exploit library to victims. Some level of management software built into the kit helps attackers understand which exploits are successfully exploiting vulnerabilities on victims' systems and which are not. This helps attackers make better decisions about which exploits to use and where to maximize their return on investment. This management software might also help attackers identify and replace exploits on their web pages that are no longer effective with new exploits. Examples of exploit kits include Angler (also known as Axpergle), Neutrino, and the notorious Blackhole exploit kit. This approach underpins a new business model and has led to the coining of a new phrase, **Malware as a Service** (**MaaS**).

Worms

Another threat category that is known to exploit unpatched vulnerabilities is worms. A worm provides its own delivery mechanism so that it can automatically spread from system to system. Worms can use unpatched vulnerabilities, security misconfigurations, weak passwords, and social engineering to propagate themselves from system to system. A great example of a worm is Conficker. There were at least a few variants of this worm. It used unpatched vulnerabilities, like MS08-067, a hardcoded list of weak passwords, and Autorun feature abuse (a feature in Windows) to spread from Windows system to Windows system (Rains, Defending Against Autorun Attacks, 2011). It could spread via removable drives, like USB drives, as well as across networks. Successful worms can be very difficult to get out of an IT environment once they get into the environment. This is because they can "hide" in online and offline storage media and operating system images.

Other examples of successful worms include SQL Slammer and Microsoft Blaster, which both spread like wildfire around the world using unpatched vulnerabilities. There are also worms like MyDoom that spread via email. It's interesting that millions of people were willing to double-click on an email attachment called *MyDoom* when it arrived in their inbox. Opening this attachment ran the worm that then sent a copy of itself to all the email addresses in the user's contact list. Worms are not a threat from the distant past. Since the days of Conficker (2007 timeframe), there have been a few wormable vulnerabilities in Windows that were accessible through default exceptions in the Windows Firewall. In all of these cases, Microsoft was able to patch hundreds of millions of systems on the internet quickly enough so that large-scale worm attacks were avoided. But this is as dangerous a scenario as it can get for a world that relies so heavily on technology.

Let me paint you a picture of the worst-case worm scenario, based on past successful global worm attacks. An attacker discovers a new zero-day vulnerability in a Windows service. The service runs by default on the vast majority of Windows systems in the world.

The vulnerable service uses a well-known TCP port to listen on the network for connection attempts to it. There is a default rule in the Windows Firewall on every system that allows network connections directly to the vulnerable service. The attacker designs a worm capable of exploiting this zero-day vulnerability and releases it on the internet.

The worm uses the vulnerability to spread before Microsoft is aware of the vulnerability and before a security update is available to patch the vulnerability. With a default rule in the Windows Firewall that allows the worm to talk directly to the TCP port that the vulnerable service is listening on, there is nothing preventing the worm from exploiting the vulnerability on virtually every consumer system running Windows that is directly connected to the internet and does not have an additional firewall protecting it. Vulnerable Windows systems behind professionally managed enterprise firewalls wouldn't be safe as infected laptops would introduce the worm into corporate IT environments when they connect via DirectAccess, VPN, or on their wireless networks (Microsoft Corporation, n.d.). The worm propagates from system to system around the world in a matter of minutes.

The public internet and most private networks would be disrupted and rendered unusable. First, the network traffic generated by the worm as it attempts to propagate and re-propagate over and over again, from system to system, would significantly disrupt legitimate network traffic on the internet, as well as the private networks it found its way into. After a system gets infected, the worm tries to infect all the systems it has network connectivity with. It simply tries to connect to the vulnerable service via the TCP port it is listening on, on every system the infected system can reach. Hundreds of millions of systems doing this at the same time would disrupt the global internet and private networks. When the worm exploits the unpatched vulnerability, it causes the target system to destabilize, causing a "Blue Screen of Death," a memory dump, and a system reboot. This exacerbates the problem because it's harder to disinfect and patch systems that are constantly rebooting.

All the systems rebooting generate even more network traffic. When each system comes back up, they generate **Address Resolution Protocol** (**ARP**) traffic and ask their DHCP servers for IP addresses. When the network segments with DHCP servers get saturated with requests for IP addresses, the DHCP servers are prevented from giving rebooting systems IP addresses. Subsequently, rebooting systems start using automatic private IP addresses that are typically non-routable (169.254.x.x). Subsequently, in some cases, these systems can no longer be reached by management software used to patch them, update anti-malware signatures, or deploy possible mitigations or workarounds to them.

The damage such an attack could do shouldn't be underestimated. The United States government has identified 16 critical infrastructure sectors. These sectors are deemed critical because if their network or systems are disrupted, it would have dire consequences on the security, economy, and public health and safety of the country. These sectors include the chemical sector, the commercial facilities sector, the communications sector, the critical manufacturing sector, the dams sector, the defense industrial base sector, the emergency services sector, the energy sector, the financial services sector, the food and agriculture sector, the government facilities sector, the healthcare and public health sector, the information technology sector, the nuclear reactors, materials, and waste sector, the transportation systems sector, and the water and wastewater systems sector (US Department of Homeland Security, n.d.).

When the worm exploits the zero-day vulnerability on vulnerable systems in these sectors, the economy, energy, water, communications, transportation, hospitals, and many other critical functions for society are disrupted and potentially taken offline. If the attacker included a malicious payload with the worm, such as encrypting data or destroying storage media, recovery would be slow and aspirational in most cases. Recovering from such an attack would require lots of manual intervention as management software tools and automation systems would be disrupted, as would the networks they are connected to. If underlying storage media on infected systems also had to be replaced, the damage from such an attack would linger for years.

Of course, I've painted a picture of a worst-case scenario. What are the chances that such a worm attack could actually be perpetrated? There were three wormable vulnerabilities in Windows operating systems in 2019 alone. On May 14, 2019, Microsoft announced the existence of a critical rated vulnerability (CVE-2019-0708) in Windows Remote Desktop Services that was wormable (NIST, n.d.). In their announcement, the Microsoft Security Response Center (MSRC) wrote the following:

> *"This vulnerability is pre-authentication and requires no user interaction. In other words, the vulnerability is 'wormable', meaning that any future malware that exploits this vulnerability could propagate from vulnerable computer to vulnerable computer in a similar way as the WannaCry malware spread across the globe in 2017. While we have observed no exploitation of this vulnerability, it is highly likely that malicious actors will write an exploit for this vulnerability and incorporate it into their malware."*
>
> *– (Microsoft Corporation, n.d.)*

CVE-2019-0708, the so-called **BlueKeep** vulnerability, applied to Windows 7, Windows Server 2008, and Windows Server 2008 R2; a third of all Windows systems were still running Windows 7 in 2019 (Keizer, Windows by the numbers: Windows 10 resumes march toward endless dominance, 2020). This vulnerability was so serious that Microsoft released security updates for old, unsupported operating systems like Windows XP and Windows Server 2003. They did this to protect the large number of systems that have never been upgraded from old operating systems that are now out of support. Protecting these old systems, which no longer get regular security updates, from a highly probable worm attack leaves less "fuel" on the internet for a worm to use to attack supported systems. Large numbers of systems that lack security updates for critical rated vulnerabilities are a recipe for disaster as they can be used for all sorts of attacks after they are compromised, including DDoS attacks.

Then on August 13, 2019, Microsoft announced the existence of two more wormable vulnerabilities (CVE-2019-1181 and CVE-2019-1182). More Windows versions contained these vulnerabilities, including Windows 7, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, Windows 8.1, and all versions of Windows 10 (including Server versions). In the announcement, the MSRC wrote:

> *"It is important that affected systems are patched as quickly as possible because of the elevated risks associated with wormable vulnerabilities like these…"*
>
> *– (Microsoft Corporation, 2019)*

In each of these three cases in 2019, Microsoft was able to find and fix these critical, wormable vulnerabilities before would-be attackers discovered them and perpetrated worm attacks that would have had crippling affects like the ones I painted here.

Ransomware

Another category of malware that can have potentially devastating consequences is ransomware. Once ransomware gets onto a system using one or more of the cybersecurity usual suspects, it will then encrypt data and/or lock the user out of the desktop of the system. The locked desktop can show a message that demands a ransom to be paid and instructions on how to pay it. Successful ransomware attacks have made headlines around the world. Examples of ransomware families include Reveton (Microsoft Corporation, n.d.) and Petya (Microsoft Corporation, n.d.). Attackers that use ransomware are brazen in their attempts to extort all sorts of organizations, including hospitals and all levels of government.

Although ransomware gets headlines, as you'll see from the data in this chapter, it is actually one of the least prevalent threat categories, from a global perspective. Even old-fashioned viruses are typically more prevalent than ransomware. But remember that risk is composed of probability and impact. The thing that makes ransomware a high-risk threat isn't the probability of encountering it; it's the impact when it's encountered. Data that has been encrypted by ransomware that utilizes properly implemented strong encryption is gone forever without the decryption keys. Subsequently, many organizations decide to pay the ransom without any guarantees that they will be able to recover all of their data. Spending time and resources to implement a ransomware mitigation strategy is a good investment. Making offline backups of all datasets that are high-value assets is a good starting point. Backups are targets for attackers that use ransomware. Therefore, keeping backups offline is an effective and necessary practice.

Also, keep in mind that nothing stays the same for long, and ransomware is constantly evolving. There is nothing preventing authors of more prevalent and successful threats from incorporating ransomware tactics as the payloads in their malware. Ransomware has been used in targeted attacks for years. One thing that likely governs the use of ransomware tactics is just how criminal the attackers are; it's one thing to develop and anonymously release malware on the internet that disrupts people and organizations, but holding assets for ransom and collecting that ransom is a different proposition usually perpetrated by a different kind of criminal altogether. Regardless, organizations need to have a mitigation strategy in place for this threat.

Viruses

Earlier, I mentioned viruses. Viruses have been around for decades. They are typically self-replicating file infectors. Viruses can spread when they are inadvertently copied between systems. Because they infect files and/or the master boot record (MBR) on systems, sometimes indiscriminately, they can be very "noisy" threats that are easy to detect, but hard to disinfect. In the last decade, viruses seem to have come back into fashion with some attackers. Modern attackers that develop viruses typically don't just infect files like their predecessors did decades ago; they can be more imaginative and malicious. Remember, most threats are blended. Modern viruses have been known to download other malware once they infect a system, disable anti-malware software, steal cached credentials, turn on the microphone and/or video camera on a computer, collect audio and video data, open backdoors for attackers, and send stolen data to remote servers for attackers to pick up. Viruses are nowhere near as prevalent as Trojans or Potentially Unwanted Software, but there always seems to be some volume of detections. A great example of a virus family that has been around for years is Sality (Microsoft Corporation, n.d.).

Browser modifiers

The final threat category I'll discuss here is browser modifiers. These threats are designed to modify browser settings without users' permission. Some browser modifiers also install browser add-ons without permission, change the default search provider, modify search results, inject ads, and change the home page and pop-up blocker settings.

Browser modifiers typically rely on social engineering for installation. The motivation for browser modifiers is typically profit; attackers use them to perpetrate click fraud. But like all threats, they can be blended with other categories and provide backdoor access and download command and control capabilities for attackers.

Measuring malware prevalence

In the next section, I will discuss how malware infections have evolved over the last decade. Before getting into that, I'll explain two ways that the prevalence of malware is measured. The first one is called **Computers cleaned per mille** (**CCM**) (Microsoft Corporation, n.d.). The term "per mille" is Latin for "in each thousand." We used this measure at Microsoft to measure how many Windows systems were infected with malware for every 1,000 systems that the MSRT scanned. You'll remember that the MSRT runs on hundreds of millions of systems when it's released the second Tuesday of every month with the security updates for Microsoft products.

CCM is calculated by taking the number of systems found to be infected by the MSRT in a country and dividing it by the total number of MSRT executions in that country. Then, multiply it by 1,000. For example, let's say the MSRT found 600 systems infected with malware after scanning 100,000 systems; the CCM would be (600/100,000)*1,000 = 6 (Microsoft Corporation, 2016).

The CCM is helpful because it allows us to compare malware infection rates of different countries by removing the Windows install base bias. For example, it's fair to say there are more Windows systems running in the United States than in Spain. Spain is a smaller country with a smaller population than the US. If we compared the raw number of systems found infected in the US with the raw number of infected systems in Spain, the US would likely look many, many more times infected than Spain. In actual fact, the CCM exposes that for many time periods, the number of systems infected for every 1,000 scanned in Spain was much higher than the number in the US.

Before a system can get infected with malware, it must encounter it first. Once a system encounters malware, the malware will use one or more of the cybersecurity usual suspects to try to infect the system. If the malware successfully infects the system, then the MSRT runs on the system, detects the infection, and cleans the system. This will be reflected in the CCM.

The malware **Encounter Rate** (**ER**) is the second definition you need to know about in order to understand the data I'm going to share with you. Microsoft defines the ER as:

> "*The percentage of computers running Microsoft real-time security software that report detecting malware or potentially unwanted software, or report detecting a specific threat or family, during a period.*"
>
> *– (Microsoft Corporation, 2016)*

Put another way, of the systems running real-time anti-malware software from Microsoft that I described earlier in this chapter, the ER is the percentage of those systems where malware was blocked from installing or where a malware infection was cleaned.

I'll use these two measures to show you how the threat landscape has changed over time. The only drawback to using this data is that Microsoft did not publish both of these measures for every time period. For example, they published CCM data from 2008 to 2016 and then stopped publishing CCM data. They started published ER data in 2013 and continued to publish some ER data into 2019. But as you'll see, they did not publish ER data for the second half of 2016, leaving a hole in the available data. Additionally, sometimes, data was published in half-year periods and other times in quarterly periods. I've done my best to compensate for these inconsistencies in the analysis I'll share with you next.

Global Windows malware infection analysis

I have aggregated data from over 20 volumes and special editions of the SIR to provide a view of how the threat landscape has evolved over time. The first measure we'll look at is the worldwide average CCM. This is the number of systems that the MSRT found to be infected with malware for every 1,000 systems it scanned around the world. *Figure 3.1* includes all the time periods that Microsoft published CCM data for in the SIR, each quarter between the third quarter of 2008 and the second quarter of 2016:

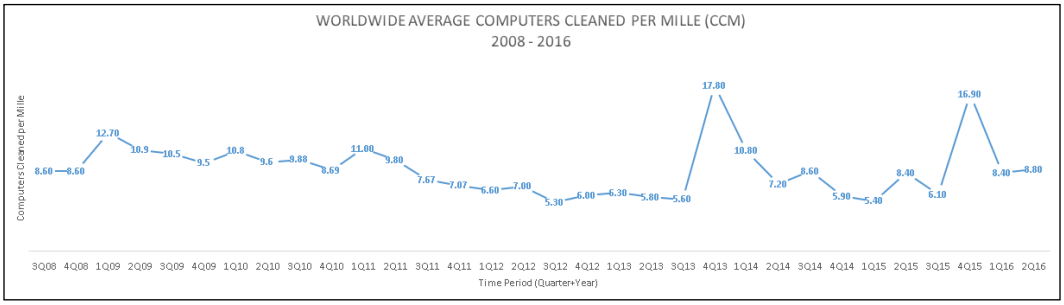


Figure 3.1: Worldwide average malware infection rate (CCM) 2008–2016 (Microsoft Corporation, n.d.)

The horizontal axis illustrates the time periods represented by the quarter and year. For example, 3Q08 is shorthand for the third quarter of 2008, while 4Q13 is the fourth quarter of 2013. The vertical axis represents the worldwide CCM for each time period. For example, in the 1st quarter of 2009 (1Q09), the worldwide average CCM was 12.70.

The worldwide average CCM for all 32 quarters illustrated in *Figure 3.1* is 8.82. To make this number clearer, let's convert it into a percentage: 8.82/1000*100 = 0.882%. For the 8-year period between the third quarter of 2008 and the end of the second quarter of 2016, the worldwide average infection rate, as measured by the MSRT, is a fraction of 1 percent. This will likely surprise some of you who have long thought that the Windows install base has always had really high malware infection rates. This is why comparing the infection rates of different countries and regions is interesting. Some countries have much higher infection rates than the worldwide average, and some countries have much lower CCMs. I'll discuss this in detail later in this chapter. The other factor contributing to a lower malware infection rate than you might have been expecting is that the source of this data is the MSRT. Remember that the MSRT is a free ecosystem cleaner designed to clean largely unprotected systems from the most prevalent and serious threats. If you look at the dates when detections were added to the MSRT, you will see that it is really cleaning a tiny fraction of the known malware families. For example, according to the list, at the end of 2005, the MSRT had detected 62 malware families (Microsoft Corporation). But it's a certainty that there were orders of magnitude more malware in the wild in 2005.

While the MSRT is only capable of detecting a fraction of all malware families, it does run on hundreds of millions of systems around the world every month. This provides us with a limited, but valuable, snapshot of the relative state of computer populations around the world. When we cross-reference MSRT data with data from real-time anti-malware solutions and some of the other data sources I outlined, we get a more complete picture of the threat landscape.

Another aspect of the MSRT that's important to understand is that it is measuring which malware families have successfully infected systems at scale. Microsoft researchers add detections to the MSRT for malware families they think are highly prevalent. Then, when the MSRT is released with the new detections, the malware researchers can see whether they guessed correctly. If they did add detections for a family of malware that was really widespread, it will appear as a spike in the malware infection rate. Adding a single new detection to the MSRT can result in a large increase in the worldwide infection rate. For example, between the third and fourth quarters of 2015 (3Q15 and 4Q15 in *Figure 3.1*), the CCM increased from 6.1 to 16.9. This is a 177% change in the malware infection rate in a single quarter. Then, in the next quarter, the CCM went down to 8.4. What drove this dramatic increase and then decrease? Microsoft malware researchers added detections to the MSRT for a threat called Win32/Diplugem in October 2015 (Microsoft Corporation). This threat is a browser modifier that turned out to be installed on a lot of systems. When Microsoft added detection for it to the MSRT in October, it cleaned Diplugem from lots of systems in October, November, and December. Typically, when a new detection is added to the MSRT, it will clean lots of infected systems the first month, fewer the second month, and fewer yet in the third month. There were a lot of systems cleaned of Diplugem in the three months of the fourth quarter of 2015. Once the swamp was mostly drained of Diplugem in 4Q15, the infection rate went down 50% in the first quarter of 2016.

This type of detection spike can also be seen between the third and fourth quarters of 2013 (3Q13 and 4Q13, in *Figure 3.1*) when the CCM increased from 5.6 to 17.8. This is a 218% change in the malware infection rate in a single quarter. Five new detections were added to the MSRT in the fourth quarter of 2013.

The detection rate spike in 4Q13 was a result of adding detection to the MSRT for a threat called Win32/Rotbrow (Microsoft Corporation, n.d.), which is a family of Trojans that can install other malware like Win32/Sefnit (Microsoft Corporation, n.d.). After the big CCM increase that this detection produced, the CCM receded back to lower levels over the next two quarters.

In order to see what's happening in a more recent time period, we'll have to use the malware ER instead of the CCM because Microsoft stopped publishing CCM data in 2016. *Figure 3.2* illustrates the ER for the period beginning in the first quarter of 2013 (1Q13) to the fourth quarter of 2018 (4Q18). Microsoft didn't publish a worldwide average ER for the second half of 2016, so we are left without data for that period:

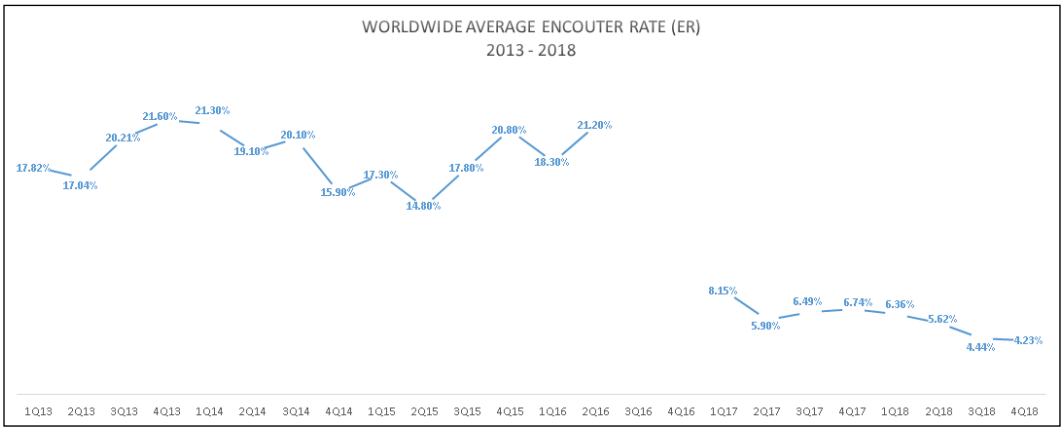


Figure 3.2: Worldwide average encounter rate (ER) 2008–2016

The average ER for the period between 2013 and the end of the first half of 2016 was 18.81%. This means that about 19% of Windows systems that were running Microsoft real-time, anti-malware software encountered malware. Almost all of these encounters likely resulted in anti-malware software blocking the installation of the malware. Some smaller proportion of encounters likely resulted in a disinfection.

The ER dropped 62% between the second quarter of 2016 (2Q16) and the first quarter of 2017 (1Q17) and didn't go back up to normal levels. In 2017 and 2018, the worldwide average ER was only 6%. I haven't seen a satisfactory explanation for this reduction and so its cause remains a mystery to me.

That has given you a long-term view of malware trends, on Windows operating systems, from a global perspective. Many of the CISOs and security teams that I've briefed using similar data expressed surprise at how low the global ER and CCM numbers are, given all the negative press malware on Windows has generated over the years. In fact, during some of my speaking engagements at conferences, I would ask the attendees what percentage of Windows systems in the world they thought were infected with malware at any given time. Attendees' guesses would typically start at 80% and work their way up from there. CISOs, security teams, and security experts need to be firmly grounded in reality if they want to lead their organizations and the industry in directions that truly make sense. That's what makes this data helpful and interesting.

That said, I find regional perspectives much more interesting and insightful than the global perspective. Next, let's look at how malware encounters and infections differ between geographic locations around the world.

Regional Windows malware infection analysis

I started studying regional malware infection rates back in 2007. At first, I studied a relatively small group of countries, probably six or seven. But over time, our work in the SIR was expanded to provide malware CCM and ER data for all countries (over 100) where there was enough data to report statistically significant findings. Over the years, three loosely coupled groups of locations emerged from the data:

1. Locations that consistently had malware infection rates (CCMs) lower than the worldwide average.
2. Locations that typically had malware infection rates consistent with the worldwide average.
3. Locations that consistently had malware infection rates much higher than the worldwide average.

Figure 3.3 illustrates some of the locations with the highest and lowest ERs in the world between 2015 and 2018. The dotted line represents the worldwide average ER so that you can see how much the other locations listed deviate from the average. Countries like Japan and Finland have had the lowest malware encounter rates and the lowest malware infection rates in the world since I started studying this data more than 10 years ago. Norway is also among the locations with low CCM and ER. Ireland is a newer addition to the list of least impacted locations. The CCM and ER for Ireland were typically lower than the worldwide average, just not one of the five or six lowest. For example, in 2008, the worldwide average CCM was 8.6 while Japan had a CCM of 1.7 and Ireland's CCM was 4.2 (Microsoft Corporation, 2009). It might be tempting to think, duh, a lower encounter rate means a lower infection rate, right? Some locations have both low CCM and low ER. But that's not always the case.

Over time, I have seen plenty of examples of locations that have high ERs but still maintain low CCMs, and vice versa. One reason for this is that not all locations have the same adoption rate of anti-malware software. This is one reason Microsoft started giving real-time anti-malware software away as a free download and now offers it as part of the operating system. There were parts of the world with alarmingly low anti-malware adoption rates. If these locations became heavily infected, they could be used as platforms to attack the rest of the world. Countries with high anti-malware protection adoption can have high ERs, but generally have much lower CCMs. This is because the real-time anti-malware software blocks malware from installing, thus increasing the ER and leaving less prevalent threats for the MSRT to clean, thereby lowering the CCM.

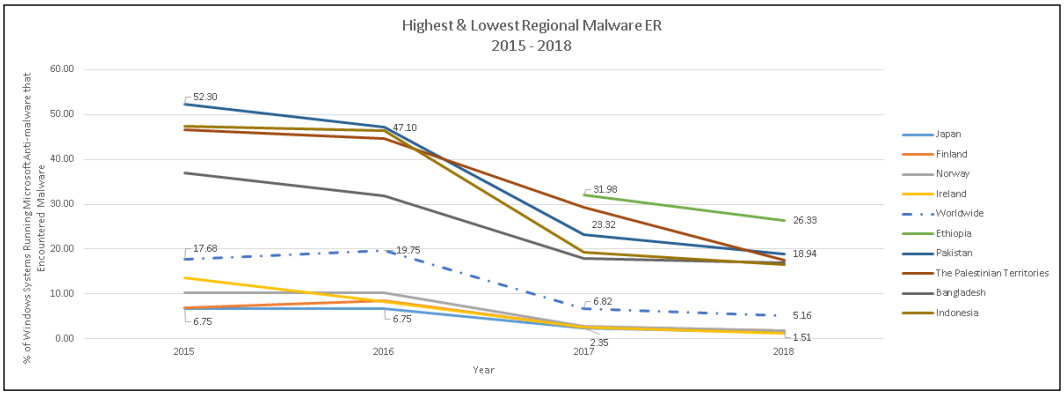


Figure 3.3: Highest and lowest regional malware encounter rates (ER) (Microsoft Corporation, n.d.)

10 years ago, locations like Pakistan, the Palestinian Territories, Bangladesh, and Indonesia all had much lower CCMs than the worldwide average (Microsoft Corporation, 2009). But over time, this changed, and these locations have had some of the highest ERs in the world in recent years. Unfortunately, we can't see whether the CCM for these countries has also increased because Microsoft stopped publishing CCM data in 2016. The last CCMs published for these locations in 2006 are shown in *Table 3.1*. (Microsoft, 2016). The CCMs for these locations are many times higher than the worldwide average, while Japan, Finland, and Norway are much lower:

Location	1Q16	2Q16
Bangladesh	29.2	24.9
Indonesia	45.1	34.4
Pakistan	42.6	37.3
The Palestinian Territories	48.0	47.9
Worldwide average	8.4	8.8
Norway	5.4	3.1
Finland	4.7	2.1
Japan	2.5	2.2

Table 3.1: Highest and lowest regional malware infection rates (CCM) in the first and second quarters of 2016 (Microsoft Corporation, n.d.)

At this point, you might be wondering why there are regional differences in malware encounter rates and infection rates. Why do places like Japan and Finland always have ultra-low infection rates, while places like Pakistan and the Palestinian Territories have very high infection rates? Is there something that the locations with low infection rates are doing that other locations can benefit from? When I first started studying these differences, I hypothesized that language could be the key difference between low and highly infected locations. For example, Japan has a hard language to learn as it's sufficiently different from English, Russian, and other languages, so it could be a barrier for would-be attackers. After all, it's hard to successfully attack victims using social engineering if they don't understand the language you are using in your attacks. But this is also true of South Korea, yet it had one of the highest CCMs in the world back in 2012, with a CCM that ranged between 70 and 93 (one of the highest CCMs ever published in the SIR) (Rains, Examining Korea's Rollercoaster Threat Landscape, 2013).

Ultimately, we tried to develop a model we could use to predict regional malware infection rates. If we could predict which locations would have high infection rates, then we were optimistic that we could help those locations develop public policy and public-private sector partnerships that could make a positive difference. Some colleagues of mine in Trustworthy Computing at Microsoft published a Microsoft Security Intelligence Report Special Edition focused on this work: *The Cybersecurity Risk Paradox, Impact of Social, Economic, and Technological Factors on Rates of Malware* (David Burt, 2014). They developed a model that used 11 socio-economic factors in 3 categories to predict regional malware infection rates. The categories and factors included (David Burt, 2014):

1. Digital access:
    1. Internet users per capita
    2. Secure Net servers per million people
    3. Facebook penetration
2. Institutional stability:
    1. Government corruption
    2. Rule of law
    3. Literacy rate
    4. Regime stability
3. Economic development:
    1. Regulatory quality
    2. Productivity
    3. Gross income per capita
    4. GDP per capita

The study found that, as developing nations increased their citizens' access to technology, their CCM increased. But more mature nations that increased their citizens' access to technology saw decreases in their CCMs. This suggests that there is a tipping point for developing nations as they transition from developing to more mature across the aforementioned categories, where increasing access to technology no longer increases CCM; instead, it decreases it.

An example of a country that appeared to make this transition in 2011–2012 was Brazil. With some positive changes in some of the socio-economic factors in the digital access and institutional stability categories, Brazil's CCM decreased from 17.3 to 9.9 (a 42% reduction) between 2011 and 2012 (David Burt, 2014).

Another nuance from the study is that the locations that had some of the highest CCMs and worst performing socio-economic factors tended to be war-torn countries, like Iraq. Another interesting insight is that in locations that don't have very good internet connectivity, whether it's because they are landlocked in the center of Africa or perpetual military conflict has impacted the availability and quality of the internet, malware infects systems via USB drives and other types of removal storage media; that is, when the internet is not able to help attackers propagate their malware, malware that doesn't rely on network connectivity becomes prevalent. When internet connectivity and access improve, then CCMs tend to increase in these locations until socio-economic conditions improve to the point that the governments and public-private sector partnerships start to make a positive difference to cybersecurity in the region. Strife and the poverty that can follow it can slow down technology refresh rates, making it easier for attackers to take advantage of people. This is a super interesting area of research. If you are interested in learning more about it, I spoke about it at Microsoft's Virtual CIO Summit in 2015 in a video recorded session called "Cyberspace 2025: What will Cybersecurity Look Like in 10 Years?" (Microsoft Corporation, 2015). We are now halfway through the period between when I recorded this video and 2025, and I think our predictions about the future using this research remain relevant and interesting.

Looking at individual countries is interesting and helpful because it illuminates what's happening in the most and least impacted locations. We can learn from the failures and successes of these locations. But, very often, CISOs ask about the threat landscape in the groups of countries where their organizations do business or where they see attacks coming from. Examining malware trends for groups of locations makes it easy to identify anomalies in those groups. It also helps to identify which countries are maintaining low malware ER and CCM, despite their neighbors who are struggling with malware. What can we learn from these countries that we can apply in other locations to improve their ecosystems? In the next section, I'll show you the trends for the following groups of countries:

- **The Middle East and Northern Africa**: There's always high interest in what's happening in this region, especially in Iran, Iraq, and Syria. This data is super interesting.
- **The European Union** (**EU**): The EU prides itself on maintaining low malware infection rates. However, this hasn't always been the case and has not been consistent across all EU member states.
- **Eastern Europe, including Russia**: Many of the CISOs I've talked to believe this area of the world is the source of much of the world's malware. But what do these countries' own malware infection rates look like?
- **Asia**: There is always high interest in malware trends in locations like China, Pakistan, and India. It's even more interesting looking at trends in East Asia, South Asia, Southeast Asia, and Oceania.
- **North and South America**: The US and Brazil are big markets that always garner high interest, but what about their neighbor's situations?

Some of these regions might not interest you. Please feel free to skip to the section on the region that interests you the most. Let's start by looking at perhaps the most interesting region in the world from a threat perspective, the Middle East and Northern Africa.

The long-term view of the threat landscape in the Middle East and Northern Africa

As a region, the Middle East and Northern Africa has had elevated malware encounter rates and malware infection rates for many years. I've had the opportunity to visit CISOs and security teams in a few of these locations over the years. The 14 locations I've included in my analysis had an average quarterly malware infection rate (CCM) of 23.9 across the 26 quarters between 2010 and 2016, while the worldwide average over the same period was 8.7 (Microsoft Corporation, n.d.). These locations as a group had nearly three times the average CCM as the rest of the world. The average quarterly malware encounter rate of these locations for the 23 quarters between the last half of 2013 and 2019 was 21.9, while the worldwide average was 12.5. *Figure 3.4* illustrates the CCM for several locations in this region for the period, starting in the first quarter of 2010 and ending in the second quarter of 2016 when Microsoft stopped publishing CCM data (Microsoft Corporation, n.d.).

10-year regional report card for the Middle East and Northern Africa

- **Region**: Middle East and Northern Africa
- **Locations included in analysis**: Algeria, Bahrain, Iran, Iraq, Israel, Jordan, Kuwait, Lebanon, Oman, Palestinian Authority, Qatar, Saudi Arabia, Syria, and United Arab Emirates
- **Average CCM (2010–2016)**: 23.9 (93% higher than worldwide average)
- **Average ER (2013–2019)**: 21.9% (55% higher than worldwide average)

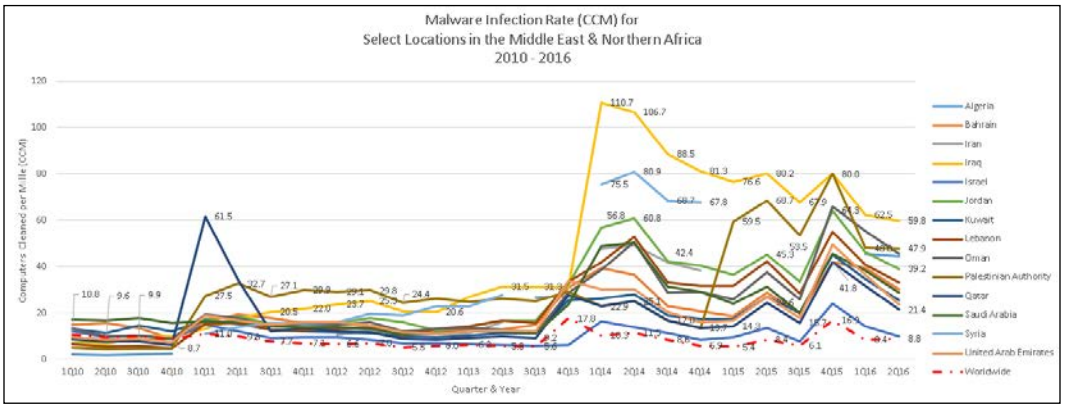


Figure 3.4: Malware infection rates for select locations in the Middle East and Africa 2010–2016 (Microsoft Corporation, n.d.)

Perhaps the most extreme example of malware infection rates climbing out of control as socio-economic factors turned very negative is Iraq. In the fourth quarter of 2013, the CCM in Iraq was 31.3, while the worldwide average was 17.8 (which, by the way, is the highest worldwide average recorded during this 5-year period). In the first quarter of 2014, the CCM in Iraq increased 254% to 110.7 (one of the highest CCMs ever recorded). During this time in Iraq, the Iraqi government lost control of Fallujah to Islamist militants (Aljazeera, 2014). The first quarter of 2014 saw waves of violence in Iraq with multiple suicide and car bombings; police were being attacked and violence was ramping up in anticipation of parliamentary elections (Wikipedia). As the country's economy suffered and its government and social underpinnings faded into the darkness of these extreme conditions, malware thrived.

Malware infection rates remained many times the worldwide average for at least the next 2 years, after which we no longer have CCM data. The malware encounter rate data does suggest that the ER in Iraq declined to points below the worldwide average in 2017, before normalizing at roughly three times the worldwide average in the last quarter of 2018 and in 2019. The ER data also shows us that Iraq didn't have the highest ER in the region, with Algeria, the Palestinian Authority, and Egypt all having higher ERs at points between 2013 and 2019:

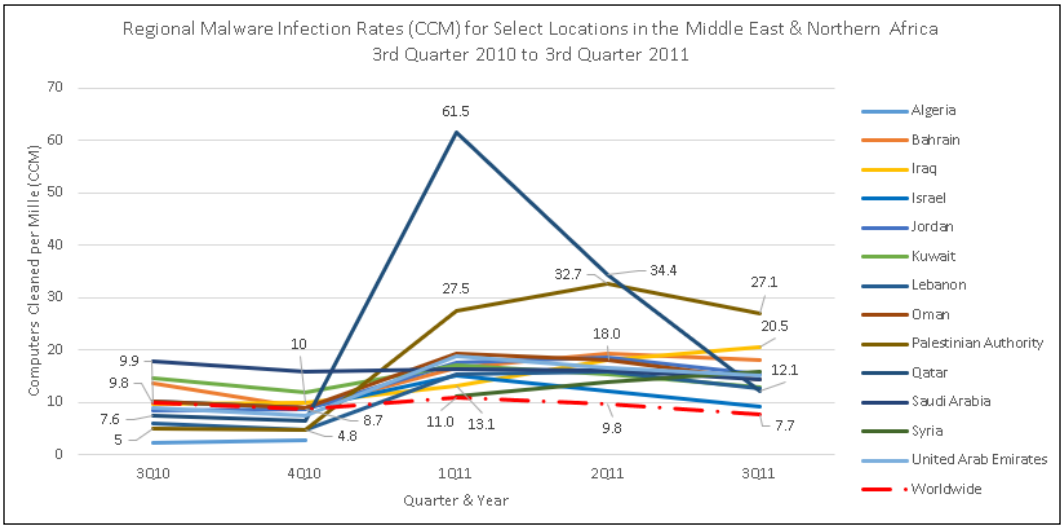


Figure 3.5: Close up of the spike in regional malware infection rates in MENA in 2011 (Microsoft Corporation, n.d.)

Another more subtle example of regional changes in CCMs that could be linked to socio-economic changes can be seen between the fourth quarter of 2010 (4Q10) and the first quarter of 2011 (1Q11). The Arab Spring started in this region in December 2010, which led to a turbulent period in several locations (Wikipedia). One week earlier, I had just returned to the US from a business trip to Egypt, and it was unnerving to see a government building I had just visited burning on CNN. Civil unrest and mass protests led to changes in government leadership in several key locations in the region. During this same time, malware infection rates increased in all the locations I have data from in the region. Locations that typically had CCMs lower than the worldwide average, such as Lebanon, Palestinian Authority, and Qatar, suddenly had higher CCMs than the worldwide average. The CCMs for these locations would never again be below the worldwide average.

As mass protests impacted the economies of some key locations in the region, and reports of crime increased dramatically, government services were interrupted and malware flourished. You might be also wondering about the big increase in the malware infection rate in Qatar in 1Q11. During this time, the prevalence of worms in Qatar was well above the worldwide average. Worms like Rimecud, Autorun, and Conficker were infecting systems with great success. All three of these worms use Autorun feature abuse to spread themselves. Once the infected systems in Qatar were disinfected, the infection rate returned to a more normal range:

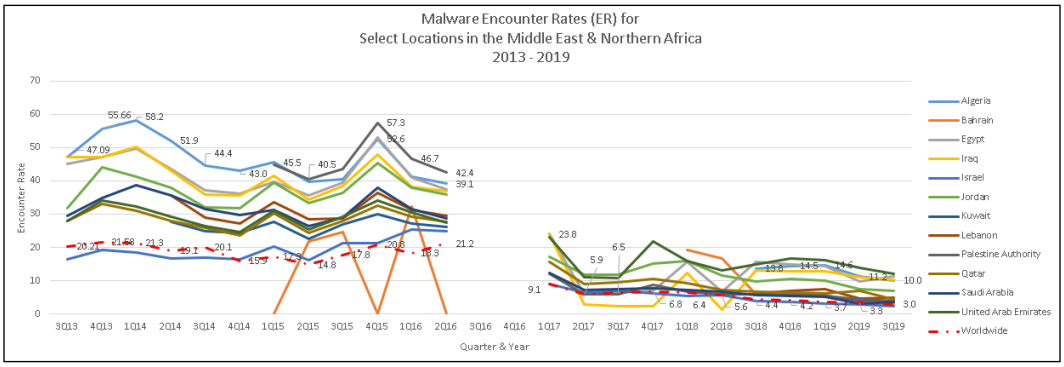


Figure 3.6: Malware encounter rates (ER) for select locations in MENA 2013–2019 (Microsoft Corporation, n.d.)

The Middle East and Northern Africa is a very interesting region. I could probably dedicate an entire chapter in this book to the things I've observed in the data from this region over the years. From a cybersecurity threat perspective, it continues to be one of the most active regions of the world, if not the most interesting.

We turn our gaze now to the threat landscape in Europe.

The long-term view of the threat landscape in the European Union and Eastern Europe

Prior to Brexit, there were 28 sovereign states in the **European Union** (**EU**). I lived in the United Kingdom during the period when Brexit was happening and traveled to continental Europe to visit CISOs there almost every week. It was a very interesting experience being at the intersection of Brexit, the advent of GDPR, the introduction of the CLOUD Act, the growing popularity of cloud computing, and heightened concern over cybersecurity. I learned a lot about European perspectives on so many topics, including data privacy and data sovereignty. I can highly recommend international experience for both personal and career growth.

From a malware perspective, in contrast to the Middle East and Northern Africa, the EU has typically had much lower infection rates. The 28 locations in the EU had an average quarterly CCM of 7.9 for the 26 quarters between 2010 and 2016. The worldwide average CCM over the same period was 8.7. The average quarterly malware encounter rate for the EU for the 23 quarters between the last half of 2013 and 2019 was 11.7, while the worldwide average was 12.5. As a group, the EU has had lower CCM and ER than the worldwide average. *Figure 3.7* illustrates the CCM for the 28 locations in the EU for the period starting in the first quarter of 2010, and ending in the second quarter of 2016, when Microsoft stopped publishing CCM data.

10-year regional report card for the European Union

- **Region**: European Union
- **Locations included in analysis**: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, and United Kingdom
- **Average CCM (2010–2016)**: 7.9 (10% lower than worldwide average)
- **Average ER (2013–2019)**: 11.7% (7% lower than worldwide average):

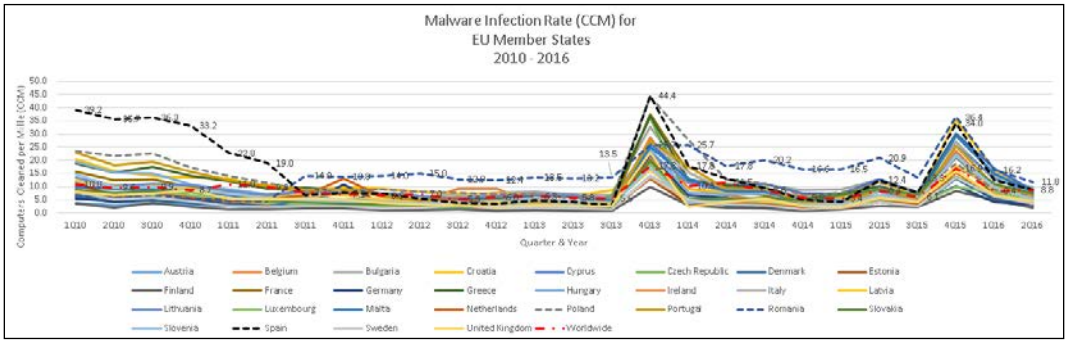


Figure 3.7: Malware infection rates (CCM) for European Union member states 2010–2016 (Microsoft Corporation, n.d.)

The first thing you might notice about this data is that Spain had the highest, or one of the highest, infection rates in the EU for several quarters in 2010, 2011, 2013, and 2015. Spain's ER was above the worldwide average for 16 of the 23 quarters between 2013 and 2019. Spain has had a very active threat landscape; over the years, I've seen malware show up first at the local level in Spain before becoming growing global threats.

In 2010, worms like Conficker, Autorun, and Taterf (Microsoft Corporation, n.d.) drove infection rates up. Romania is also among the most active locations in the EU, at times having the highest CCM and ER in the region.

The spike in malware infection rates in the fourth quarter of 2013 (4Q13) was due to three threats that relied on social engineering, Trojan downloaders Rotbrow and Brantall, and a Trojan called Sefnit (Microsoft Corporation, n.d.). The CCM spike in the fourth quarter of 2015 (4Q15) was due to the global rise in the prevalence of one browser modifier called Diplugem (Microsoft Corporation, n.d.):

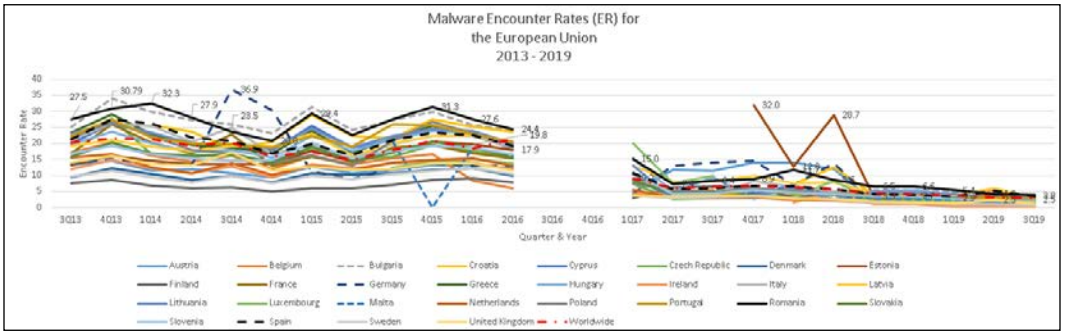


Figure 3.8: Malware encounter rates (ER) for select locations in the European Union 2013–2019 (Microsoft Corporation, n.d.)

The spike seen in Germany's ER in the third and fourth quarters of 2014 was due to some families of threats that were on the rise in Europe during that time, including EyeStye (also known as SpyEye), Zbot (also known as the Zeus botnet), Keygen, and the notorious BlackHole exploit kit (Rains, New Microsoft Malware Protection Center Threat Report Published: EyeStye).

The locations with the consistently lowest CCMs and ERs in the EU are Finland and Sweden. Neither Finland's CCM nor Sweden's CCM has gone above the worldwide average. Sweden's ER did not get above the worldwide average, while Finland's all-time high ER was a fraction of a point above the worldwide average. The positive socio-economic factors at work in the Nordics, including Norway, Denmark, and Iceland, seem to have inoculated them from malware compared to most of the rest of the world:

Location	1Q10 - 2Q16 Average CCM
Spain	15.5
Romania	14.5
Poland	13.2
Greece	11.4
Croatia	11.3
Bulgaria	10.8
Portugal	10.5
Cyprus	9.9
Lithuania	9.3
Hungary	8.7

Location	1Q10 - 2Q16 Average CCM
Finland	2.4
Denmark	3.7
Sweden	4.1
Austria	4.4
Germany	5.0
Luxembourg	5.2
Estonia	5.3
Ireland	5.4
United Kingdom	5.7
Czech Republic	5.9

Table 3.2: Left: EU locations with the highest average CCM, 1Q10–2Q16; right: EU locations with the lowest average CCM, 1Q10–2Q16 (Microsoft Corporation, n.d.)

Location	3Q13 - 3Q19 Average ER
Bulgaria	20.5
Romania	17.9
Croatia	15.9
Greece	15.4
Spain	13.8
Estonia	13.8
Lithuania	13.7
Latvia	13.7
Portugal	13.7
Italy	13.3

Location	3Q13 - 3Q19 Average ER
Finland	5.2
Sweden	6.2
Malta	6.5
Denmark	7.2
Ireland	8.0
Luxembourg	8.1
United Kingdom	8.2
Netherlands	8.2
Austria	9.7
Belgium	10.5

Table 3.3: Left: EU locations with the highest average ER, 3Q13–3Q19; right: EU locations with the lowest average ER, 3Q19–3Q19 (Microsoft Corporation, n.d.)

Of course, when discussing malware, there's always high interest in Russia and their Eastern European neighbors. In my career, I've had the chance to visit CISOs and cybersecurity experts in Russia, Poland, and Turkey. I always learn something from cybersecurity experts in this region as there is always so much activity. My experience also suggests that there isn't a bad restaurant in Istanbul!

Russia's CCM has hovered around or below the worldwide average consistently over time. This is despite the ER in Russia being typically above the worldwide average. Russia did suffer the same malware infection spikes in 2013 and 2015 as the rest of Europe did.

The most active location in this region has been Turkey. The CCM and ER in Turkey have been consistently significantly higher than the worldwide average. It has had the highest CCM of these locations in all but one quarter, between 2010 and 2016. Turkey had the highest ER of these locations until the second half of 2016, when the ER of Ukraine started to surpass it. Turkey's threat landscape is as unique as its location as the point where Europe and Asia meet, driven by an eclectic mix of Trojans, worms, and viruses. There was a big increase in both the CCM and ER in Turkey in 2014. Interestingly, 2014 was a presidential election year in Turkey (Turkey's Premier Is Proclaimed Winner of Presidential Election, 2014), and saw large anti-government protests related to proposed new regulations of the internet there (Ece Toksabay, 2014). There were also significant spikes in CCM and ER in Turkey at the end of 2015 and into 2016. Again, it's interesting that a general election was held in June of 2015 and there were a series of ISIS-related bombings and attacks in Turkey during this time.

Estonia has had the lowest CCM and ER for much of the period I studied, both typically below the worldwide average. But there are spikes in the ER data in the fourth quarter of 2017 and the second quarter of 2018. At the time of writing, Microsoft had not yet published an explanation for this, but we can get some idea from the 2018 report (Republic of Estonia Information System Authority, 2018) and 2019 report (Authority, 2019) published by the Estonian Information System Authority, which seems to point the finger at the WannaCry and NotPetya ransomware campaigns and the exploitation of unpatched vulnerabilities.

10-year regional report card for select Eastern European locations

- **Region**: Select Eastern European locations
- **Locations included in analysis**: Bulgaria, Estonia, Latvia, Slovakia, Russia, Turkey, and Ukraine
- **Average CCM (2010–2016)**: 10.5 (19% higher than worldwide average)
- **Average ER (2013–2019)**: 17.2% (32% higher than worldwide average):

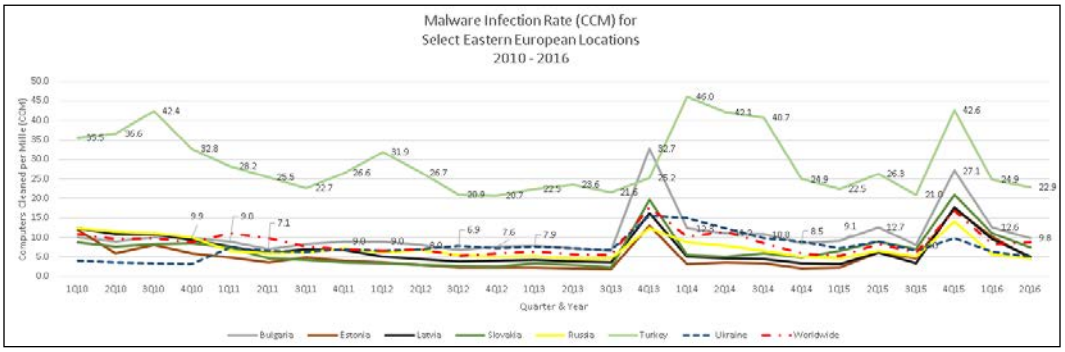


Figure 3.9: Malware infection rates for select locations in Eastern Europe 2010–2016 (Microsoft Corporation, n.d.)

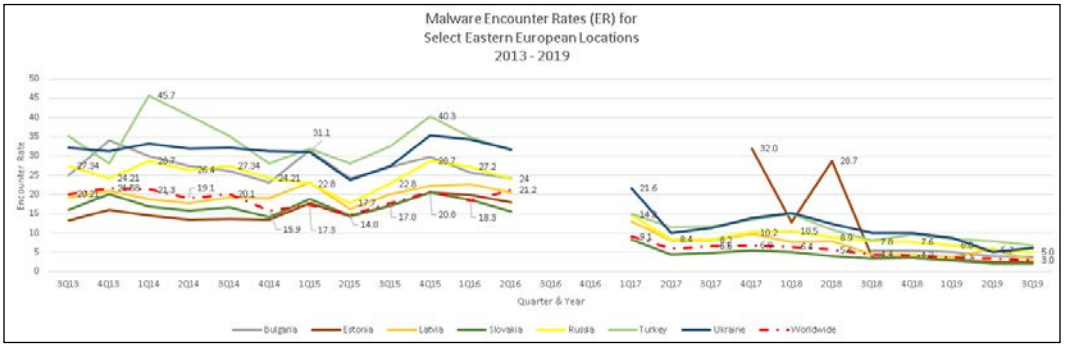


Figure 3.10: Malware encounter rates (ER) for select locations in Eastern Europe 2013–2019 (Microsoft Corporation, n.d.)

Location	1Q10 - 2Q16 Average CCM
Turkey	29.1
Bulgaria	10.8
Worldwide	8.7
Ukraine	7.6
Russia	7.3
Latvia	6.9
Slovakia	6.8
Estonia	5.3

Location	3Q13 - 3Q19 Average ER
Turkey	23.0
Ukraine	21.7
Bulgaria	20.5
Russia	17.2
Estonia	13.8
Latvia	13.7
Worldwide	12.5
Slovakia	10.9

Table 3.4: Left: Select Eastern European locations, average CCM, 1Q10–2Q16; right: Select Eastern European locations, average ER, 3Q19–3Q19 (Microsoft Corporation, n.d.)

Having looked at the landscape in Europe and Eastern Europe, let's shift gears and examine trends for some locations across Asia.

The long-term view of the threat landscape in select locations in Asia

Did you know that about 60% of the world's population lives in Asia? I've been lucky enough to visit Asia several times in my career, visiting CISOs and security teams in Japan, Korea, Singapore, Hong Kong, Malaysia, India, China, the Philippines, Australia, New Zealand, and so many other cool places there. Asia also has an interesting threat landscape where, as a whole, it has a significantly higher ER and CCM than the worldwide averages. Several locations in Asia have CCMs and ERs far above the worldwide average. Pakistan, Korea, Indonesia, the Philippines, Vietnam, India, Malaysia, and Cambodia all have much higher CCMs than the worldwide average. Locations like Japan, China, Australia, and New Zealand have much lower infection rates than the rest of Asia, well below the worldwide average.

Location	1Q10 - 2Q16 Average CCM	Location	1Q10 - 2Q16 Average CCM
Pakistan	36.7	Japan	2.7
Korea	24.5	China	3.1
Indonesia	24.3	Australia	5.4
Philippines	22.2	New Zealand	5.5
Vietnam	21.9	Hong Kong SAR	6.3
India	20.4	Singapore	7.7
Malaysia	15.9	Worldwide	8.7
Cambodia	15.6	Taiwan	12.3
Taiwan	12.3	Cambodia	15.6
Worldwide	8.7	Malaysia	15.9

Table 3.5: Left: Locations in Asia with the highest average CCM, 3Q13–3Q19; right: Locations in Asia with the lowest average CCM, 3Q19–3Q19 (Microsoft Corporation, n.d.)

Location	3Q13 - 3Q19 Average ER	Location	3Q13 - 3Q19 Average ER
Pakistan	35.9	Japan	5.3
Indonesia	32.7	New Zealand	7.4
Vietnam	29.2	Australia	8.6
India	25.4	Hong Kong SAR	8.7
Philippines	25.1	Singapore	9.5

Table 3.6: Left: Locations in Asia with the highest average ER, 3Q13–3Q19; right: Locations in Asia with the lowest average ER, 3Q19–3Q19 (Microsoft Corporation, n.d.)

10-year regional report card for Asia

- **Region**: Asia
- **Locations included in analysis**: Australia, Cambodia, China, Hong Kong SAR, India, Indonesia, Japan, Korea, Malaysia, New Zealand, Pakistan, Philippines, Singapore, Taiwan, and Vietnam

- **Average CCM (2010–2016)**: 10.5 (19% higher than worldwide average)
- **Average ER (2013–2019)**: 17.2% (32% higher than worldwide average):

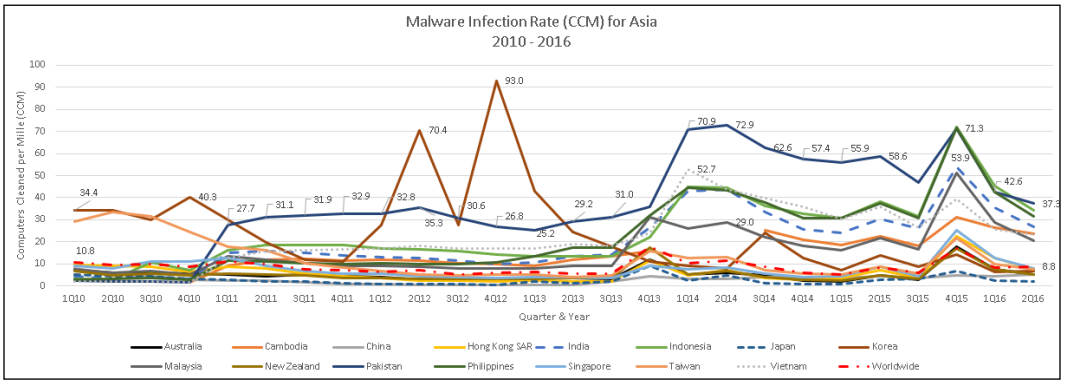


Figure 3.11: Malware infection rates (CCM) for select locations in Asia, 2010–2016 (Microsoft Corporation, n.d.)

There were big increases in the malware infection rate in South Korea in the second and fourth quarters of 2012. Korea had the highest malware infection rate in Asia during this time, even higher than Pakistan, which has one of the most active threat landscapes in the world. These infection rate spikes were driven by just two families of threats that relied on social engineering to spread. One of these threats was fake anti-virus software that was found on a significant number of systems in Korea. Notice that this spike only happened in Korea. Social engineering typically relies on language to trick users to make poor trust decisions. Apparently, a Korean language version of this fake antivirus software was very successful at the time. But that threat wouldn't trick very many non-Korean language speakers. I remember visiting South Korea at the time to drive awareness among public sector and commercial sector organizations of the country's high malware infection rate. Many of the people I talked to in Seoul expressed surprise and even disbelief that the country had the highest infection rate in the world.

You might also notice the sharp increase in the malware infection rate in Pakistan in 2014. Pakistan also had one of the highest ERs in Asia during this time period, along with Indonesia. It's noteworthy that there were numerous violent events in Pakistan during 2014, including multiple bombings, shootings, and military actions (Wikipedia, n.d.).

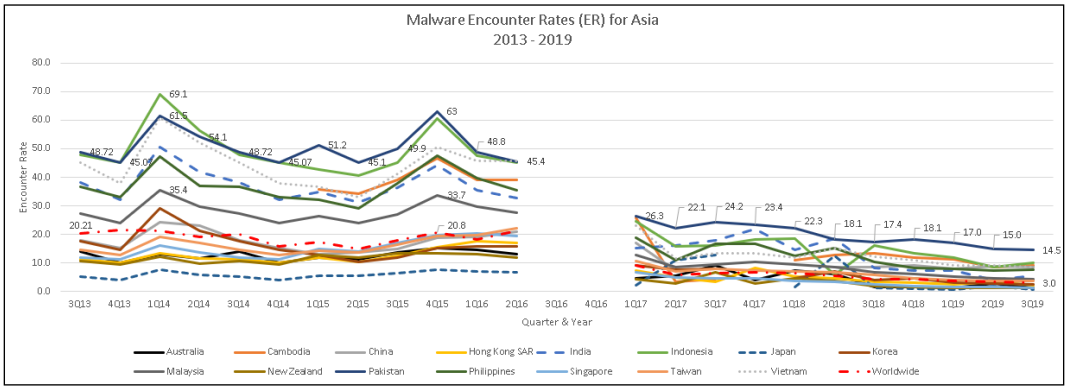


Figure 3.12: Malware encounter rates (ER) for select locations in Asia, 2013–2019 (Microsoft Corporation, n.d.)

Asia is so large and diverse that we can get better visibility into the relative CCMs and ERs of these locations by breaking the data into sub-regions. My analysis doesn't include every country in every region, but the results are interesting nonetheless. Oceania has the lowest infection rate and encounter rate of any region in Asia; the CCM and ER of Oceania are below the worldwide average, while those of every other region in Asia are above the worldwide average. Without the aforementioned CCM spike in South Korea, East Asia's CCM likely would have also been below the worldwide average. This data clearly illustrates that South Asia has significantly higher levels of malware encounters and infections than anywhere else in Asia. These are even higher than the average CCM and ER in the Middle East and Northern Africa, at 23.9 and 21.9%, respectively.

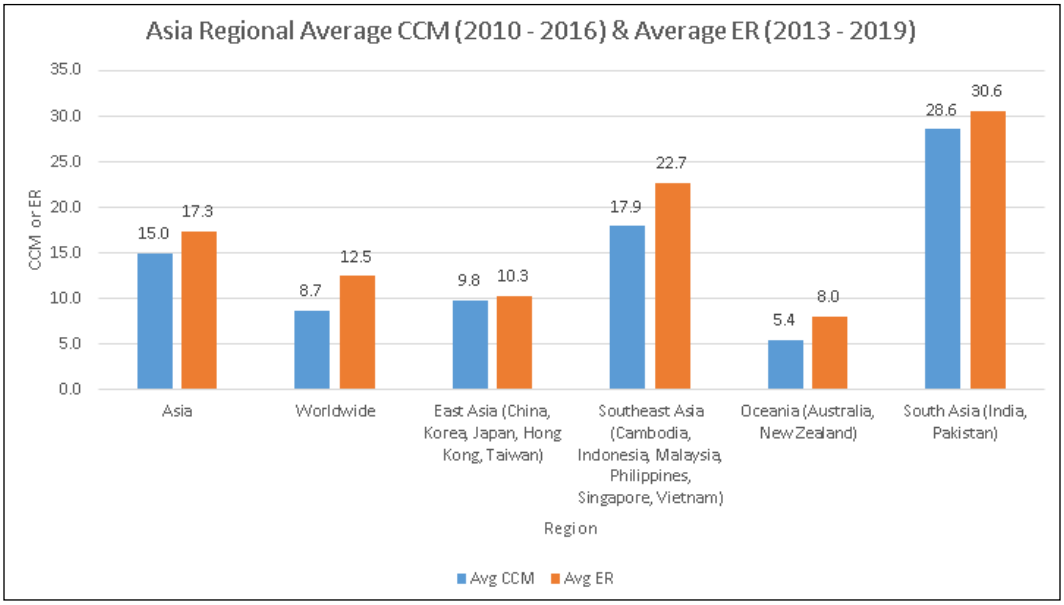


Figure 3.13: Asia regional malware infection rates (2010–2016) and encounter rates (2013–2019) (Microsoft Corporation, n.d.)

Next, let's examine the situation in the Americas. I've had the opportunity to live in both the United States and Canada, where I have met with countless CISOs and security teams over the years. I have also had the opportunity to visit CISOs in different locations in South America.

The long-term view of the threat landscape in select locations in the Americas

When I examine CCM data from 2007 and 2008, I can find periods where the United States had a malware infection rate above the worldwide average. But for most of the period between 2010 and 2016, the CCM in the US hovered near or below the worldwide average. The ER in the US is also typically below the worldwide average.

It used to be that the US was a primary target for attackers because consumers' systems in the US had relatively good internet connectivity, relatively fast processors, and lots of available storage—all things that attackers could use for their illicit purposes. But over time, consumers in the US became more aware of attackers' tactics, and vendors started turning on security features in newer systems by default. Over time, the quality of the internet improved in other countries, as did consumers' computer systems. Attackers followed new populations as they came online and focus on attacking consumer systems in the US receded. In more recent periods, locations like Brazil, Argentina, Mexico, Venezuela, and Honduras have had the highest malware infection rates in the Americas.

10-year regional report card for the Americas

- **Region**: The Americas
- **Locations included in analysis**: Argentina, Bolivia, Brazil, Canada, Chile, Colombia, Costa Rica, Ecuador, Guatemala, Honduras, Mexico, Nicaragua, Panama, Paraguay, Peru, United States, Uruguay, and Venezuela
- **Average CCM (2010–2016)**: 13.4 (43% higher than worldwide average)
- **Average ER (2013–2019)**: 16.5% (26% higher than worldwide average)

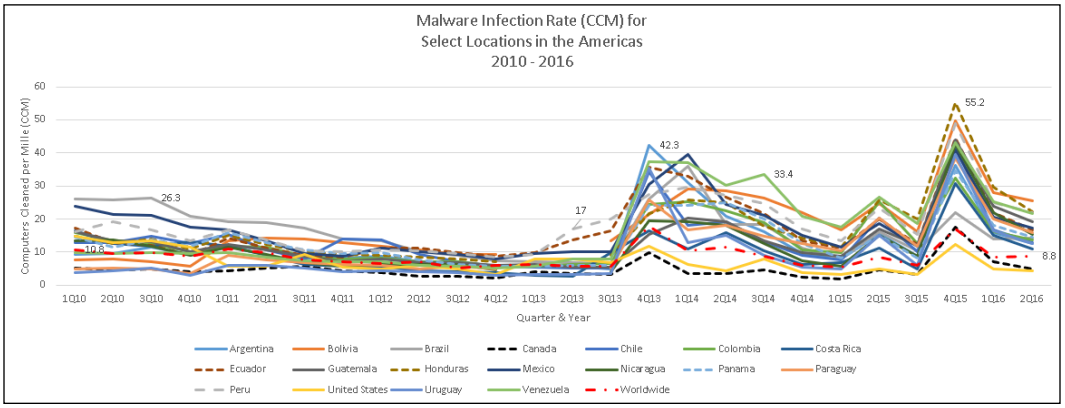


Figure 3.14: Malware infection rates for select locations in the Americas, 2010–2016 (Microsoft Corporation, n.d.)

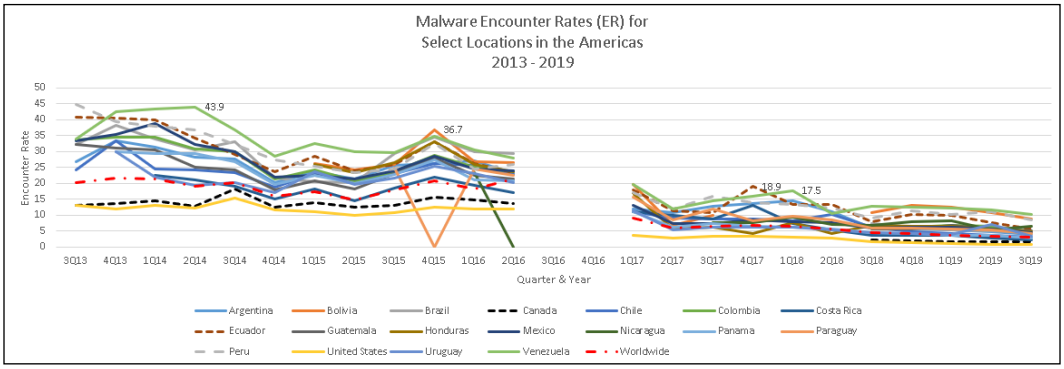


Figure 3.15: Malware encounter rates (ER) for select locations in the Americas 2013–2019 (Microsoft Corporation, n.d.)

Location	1Q10 - 2Q16 Average CCM
Peru	18.3
Bolivia	17.7
Mexico	17.4
Honduras	17.1
Ecuador	16.6
Venezuela	16.5
Brazil	16.3
Chile	13.7
Panama	13.5
Colombia	13.5

Location	1Q10 - 2Q16 Average CCM
Canada	4.8
United States	7.3
Worldwide	8.7
Uruguay	9.1
Costa Rica	10.0
Paraguay	11.7
Nicaragua	12.3
Guatemala	13.1
Argentina	13.1
Colombia	13.5
Panama	13.5

Table 3.7: Left: Locations in the Americas with the highest average CCM, 3Q13–3Q19; right: Locations in the Americas with the lowest average CCM, 3Q19–3Q19 (Microsoft Corporation, n.d.)

Location	3Q13 - 3Q19 Average ER
Venezuela	24.5
Brazil	22.2
Peru	22.1
Ecuador	21.6
Colombia	20.9
Guatemala	18.2
Mexico	18.1
Argentina	18.0
Bolivia	18.0
Chile	16.0

Location	3Q13 - 3Q19 Average ER
United States	7.4
Nicaragua	8.2
Canada	10.1
Paraguay	10.8
Costa Rica	12.3
Worldwide	12.5
Honduras	13.7
Uruguay	13.9
Panama	15.1
Chile	16.0

Table 3.8: Left: Locations in the Americas with the highest average ER, 3Q13–3Q19; right: Locations in the Americas with the lowest average ER, 3Q19–3Q19 (Microsoft Corporation, n.d.)

As a whole, the Americas has a higher CCM and ER than the worldwide average. However, North America, Central America, and South America all have slightly different levels of malware encounters and infections. Although my analysis doesn't include all the locations in the Americas, breaking the data out by region makes it a little easier to compare them.

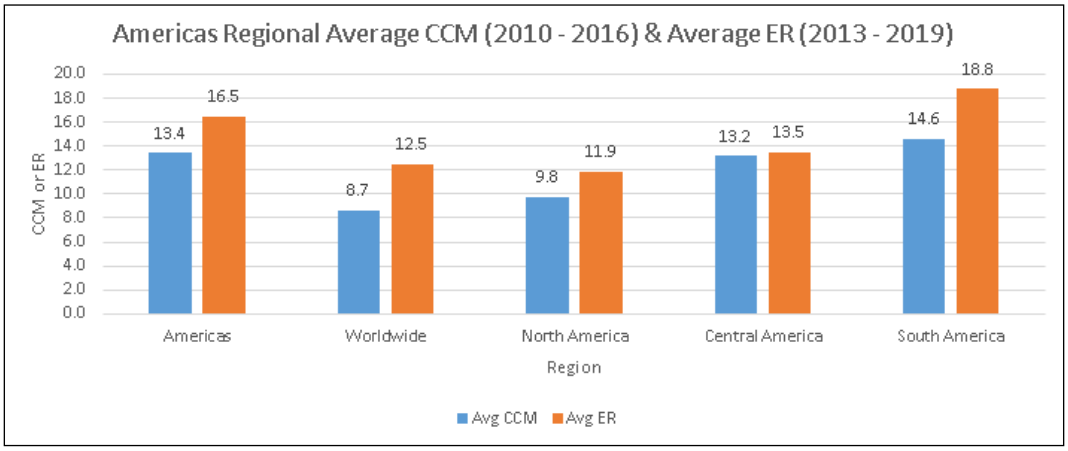


Figure 3.16: Americas average regional malware infection rates (2010–2016) and encounter rates (2013–2019) (Microsoft Corporation, n.d.)

I hope you enjoyed this tour around the world. It took me months to do this research and analysis, so obviously, I find regional malware trends really interesting. And for the security teams that live in these regions, especially outside of the United States, credible regional threat intelligence can be hard to find, while fear, uncertainty, and doubt always seems to be close by. Let me share some conclusions from this analysis with you.

Regional Windows malware infection analysis conclusions

Figure 3.17 illustrates the regional breakdown data on a single graph, which makes it easier to see the relative CCM and ER levels around the world. Over the past decade, systems in South Asia, Southeast Asia, and the Middle East and Northern Africa have encountered more malware than anywhere else in the world. This is likely a primary contributing factor to these regions also having the highest malware infection rates in the world.

This is contrasted by the much lower ERs and CCMs of Oceania, East Asia, and the EU.

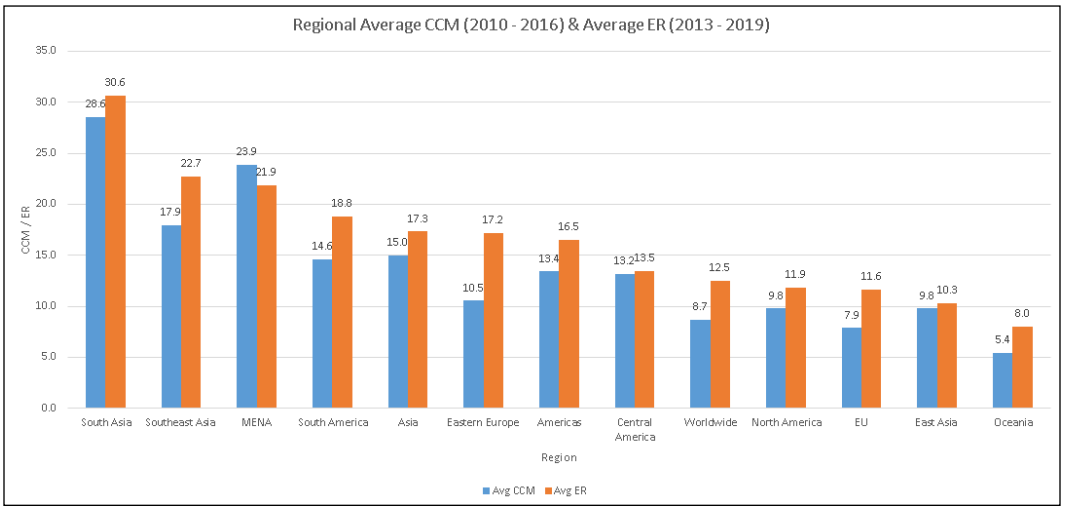


Figure 3.17: Average CCM and ER for regions worldwide, 2013–2019 (Microsoft Corporation, n.d.)

The top 10 locations with the highest average CCMs and ERs in the world are listed in *Table 3.9* here. The worldwide average CCM for the same period is 8.7, and the average ER is 12.5. All of these locations have at least twice the ER and CCM than the worldwide average.

Location	1Q10 - 2Q16 Average CCM	Location	3Q13 - 3Q19 Average ER
Iraq	43.6	Algeria	35.9
Pakistan	36.7	Pakistan	35.9
Syria	34.2	Indonesia	32.7
Palestinian Authority	30.3	Vietnam	29.2
Jordan	27.3	Egypt	27.7
Iran	25.0	Iraq	26.2
Korea	24.5	India	25.4
Oman	24.5	Philippines	25.1
Indonesia	24.3	Jordan	25.0
Lebanon	23.1	Venezuela	24.5

Table 3.9: Locations with the highest CCMs and ERs in the world 1Q10–2Q16 (Microsoft Corporation, n.d.)

What does this all mean for CISOs and enterprise security teams?

I've met many teams over the years that block all internet traffic originating from China, Iran, and Russia because of the attacks they see that originate from those country-level IP address ranges. From what CISOs have told me, including attribution reports published by the US and UK governments and reports in the press, there certainly doesn't seem to be any doubt that many attacks originate from these three locations. But of course, attackers are not limited to using IP address ranges from their home country or any particular country, so this isn't a silver bullet mitigation. And remember that the systems of the victims of such attacks are used to perpetrate attacks against other potential victims, so their IP addresses will be the sources of many attacks.

When systems are compromised by malware, some of them are used in attacks, including DDoS attacks, drive-by download attacks, watering hole attacks, malware hosting, and other "project work" for attackers. Therefore, some CISOs take the precautionary step to block internet traffic to/from the locations with the highest malware infection rates in the world. If your organization doesn't do business in these locations or have potential partners or customers in them, minimizing exposure to systems in these locations might work as an additional mitigation for malware infections. Many organizations use managed firewall and WAF rules for this very reason. But given my analysis is for a full decade, in order to make the list of most infected locations, these locations essentially must have consistently high infection rates. Limiting the places that Information Workers can visit on the internet will reduce the number of potential threats they get exposed to.

For security teams that live in these locations or support operations in these locations, I hope you can use this data to get appropriate support for your cybersecurity strategy, from your C-suite, local industry, and all levels of government. Using that submarine analogy I wrote about in the preface of this book, there's no place on Earth with more pressure on the hull of the submarine than in these locations.

This is a double-edged sword as it puts more pressure on security teams in these locations, but also provides them with the context and clarity that organizations in other parts of the world do not have. Use this data to drive awareness among your cybersecurity stakeholder communities and to get the support you need to be successful.

Some of the CISOs I know have used CCM and ER data as a baseline for their organizations. They use their anti-malware software to develop detection, blocked, and disinfection data for their IT environments. They compare the CCM and ER from their environments to the global figures published by Microsoft or other anti-malware vendors. They will also compare their CCM and ER datapoints to regional figures in the countries where they have IT operations. This allows them to compare whether their organization is more, or less, impacted than the average consumer systems in their country or globally. Their goal is to always have lower CCM and ER figures than their country has and lower than the global averages. They find global and regional malware data to be a useful baseline to determine whether they are doing a good job managing malware in their environment.

From a public policy perspective, it appears as though some of the governments in Oceania, East Asia, and the EU have something to teach the rest of the world about keeping the threat landscape under control. Specifically, governments in Australia, New Zealand, the Nordics, and Japan should help highly infected regions get on the right track. But this will be no easy task, as high levels of strife seems to be the underlying factor impacting the socio-economic factors that are linked to high regional malware infection rates. Addressing government corruption, embracing the rule of law, improving literacy rates, regime stability, regulatory quality, productivity, gross income per capita, and GDP per capita are the first orders of business in order to reduce malware infection rates in many locations. Corporate CISOs and cybersecurity leaders in the public sector can contribute to a better future by educating their nations' public policy influencers.

Now that I've provided you with a deep dive into regional malware encounters and infections, let's look at how the use of different categories of malware has evolved over time globally. At the risk of sounding like a cybersecurity data geek, this data is my favorite malware-related data! Social engineering is a mainstay technique for attackers, and this 10-year view of how attackers have used malware illustrates this clearly.

Global malware evolution

Understanding the evolution of malware will help CISOs and security teams put the hysteria they read in the news into context. Keep the cybersecurity usual suspects in the back of your mind as you read this section.

In the wake of the successful large-scale worm attacks of 2003 and early 2004, Microsoft introduced Windows XP Service Pack 2 in August of 2004. Among other things, Windows XP Service Pack 2 turned on the Windows Firewall by default for the first time in a Windows operating system. Prior to this, it was an optional setting that was left to customers to turn on, configure, and test with their applications. This service pack also offered **Address Space Layout Randomization** (**ASLR**) and **Data Execution Prevention** (**DEP**) for the first time in a Windows operating system (David Ladd, 2011). These three features blunted the success of future mass worm attacks that sought to use the same tactics as SQL Slammer and MSBlaster. A vulnerability in a service listening on a network port cannot be exploited if there's a host-based firewall blocking packets from getting to the port. The memory location of a vulnerability might not be the same on every system, making it harder to find and exploit.

18 months after Windows XP Service Pack 2 was released and its adoption was widespread, the data shows us that worms and backdoors fell out of favor with attackers. As shown in *Figure 3.18*, the number of detections of these categories of malware saw dramatic reductions in 2006, 2007, and 2008.

A different type of worm, one that didn't just use unpatched vulnerabilities, became popular with attackers in 2009, 5 years after Windows Firewall, ASLR, and DEP were turned on in Windows operating systems.

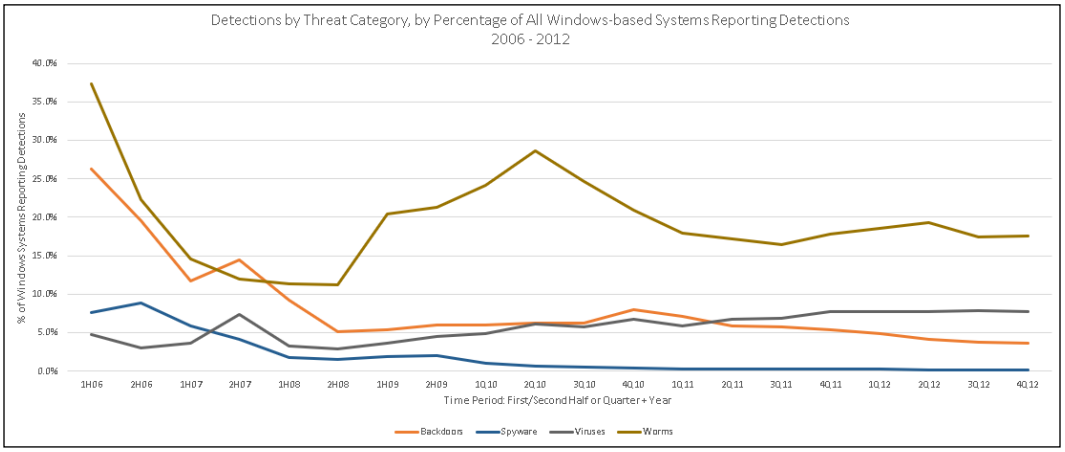


Figure 3.18: Detections by threat category, including Backdoors, Spyware, Viruses, and Worms by percentage of all Windows-based systems reporting detections, 2006–2012 (Microsoft Corporation, n.d.)

Once worms were no longer effective for mass attacks, the data shows us that Miscellaneous Potentially Unwanted Software became popular in 2006, 2007, and 2008. You can see this marked increase in *Figure 3.19*. As I described earlier in this chapter, this category of threat typically relies on social engineering to get onto systems. Fake anti-virus software, fake spyware detection suites, and fake browser protectors were all the rage during this period:

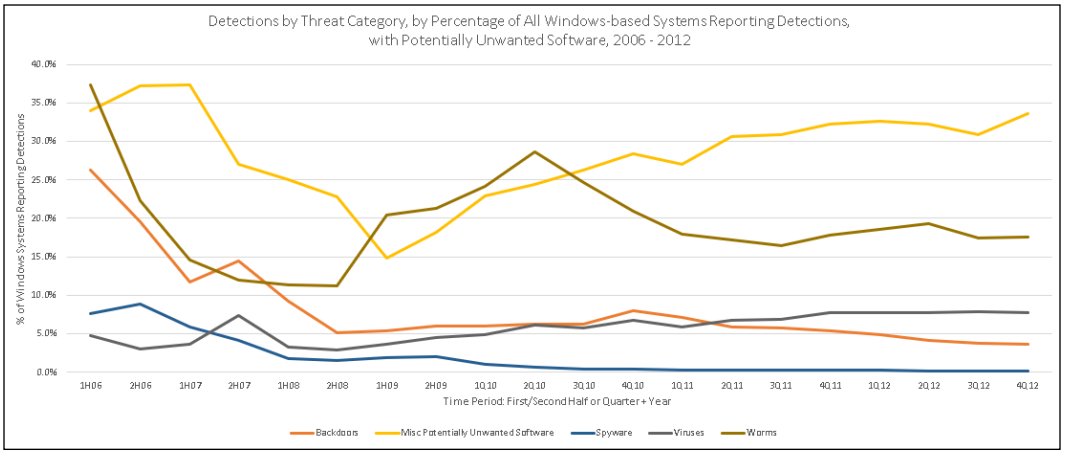


Figure 3.19: Detections by threat category, including Backdoors, Spyware, Viruses, Worms, and Miscellaneous Potentially Unwanted Software by percentage of all Windows-based systems reporting detections, 2006–2012 (Microsoft Corporation, n.d.)

As the use of potentially unwanted software peaked in 2006 and more people were getting wise to them, detections trended down in 2007 and 2008. During this time, the data shows us that Trojan Downloaders and Droppers came into fashion. This is clearly reflected in *Figure 3.20*. This category of threat also primarily relies on social engineering to initially compromise systems. They trick the user into installing them and then unpack or download more malware to the system to give attackers further control. During this time, it was not uncommon for Trojan Downloaders and Droppers to enlist their victims' systems into botnets for use in other types of attacks.

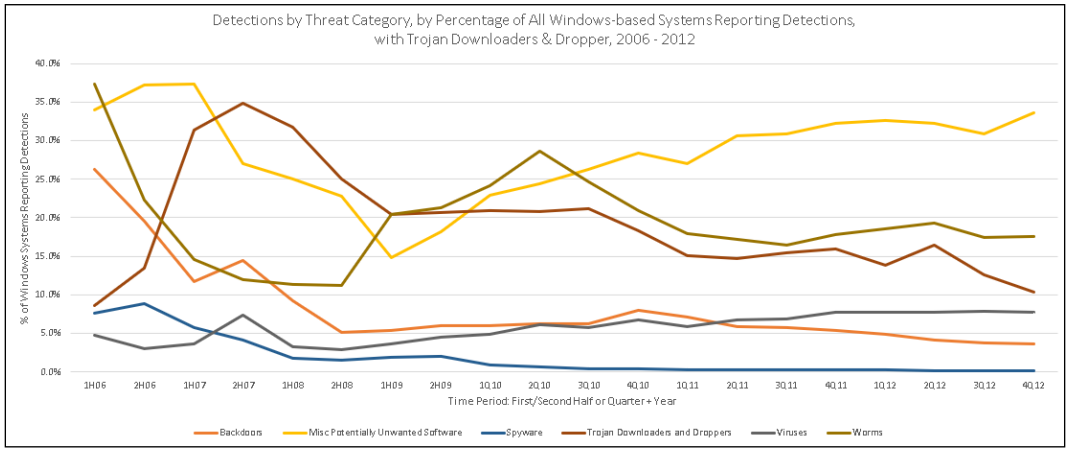


Figure 3.20: Detections by threat category, including Backdoors, Spyware, Viruses, Worms, Miscellaneous Potentially Unwanted Software, and Trojan Downloaders and Droppers by percentage of all Windows-based systems reporting detections, 2006–2012 (Microsoft Corporation, n.d.)

As people caught on to the dirty tricks that attackers were using with Trojan Downloaders and Droppers, and anti-virus companies focused on eradicating this popular category of malware, the data shows the popularity of Droppers and Downloaders receding, while detections of miscellaneous Trojans peaked in 2008 and again in 2009. This category of threat also relies primarily on social engineering to be successful. The data also shows us that there was a significant increase in detections of password stealers and monitoring tools between 2007 and 2011.

There was a resurgence in the popularity of worms in 2008, when Conficker showed attackers what was possible by combining three of the usual suspects into a single worm.

Since then, worms that rely on AutoRun feature abuse, weak, leaked, and stolen passwords have remained popular. In *Figure 3.21*, notice the slow but steady rise of Exploits starting in 2009. This trend peaked in 2012, when Exploit Kits were all the rage on the internet. Also, notice that there is no significant volume of ransomware throughout this entire period. As we leave this period at the end of 2012, the categories at the top-right corner of the graph, Trojans and Potentially Unwanted Software, rely on social engineering to be successful.

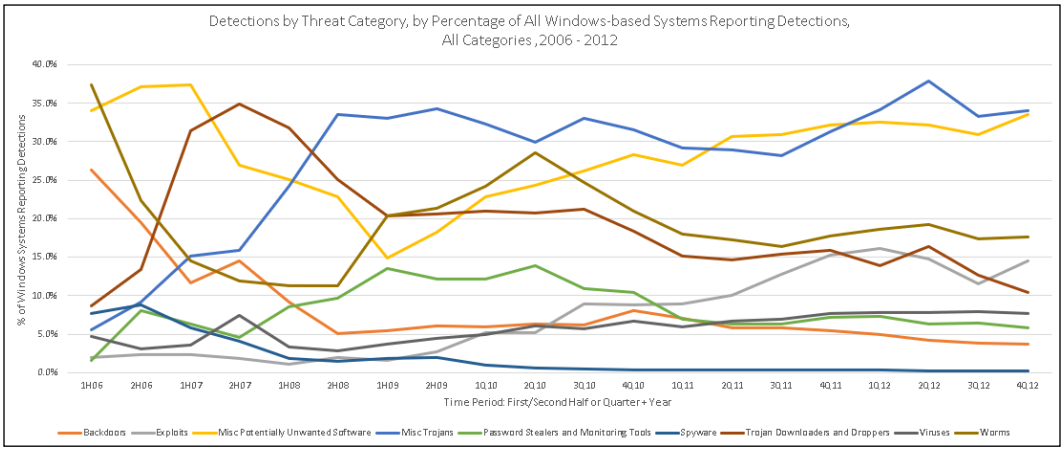


Figure 3.21: Detections by threat category, all categories, by percentage of all Windows-based systems reporting detections, 2006–2012 (Microsoft Corporation, n.d.)

Entering 2013, Microsoft started using the ER to measure threat detections. Note that the measure used between 2013 and 2017 is ER versus the detections measure used in the prior period. These are slightly different data points. Microsoft did not publish ER data in the third and fourth quarters of 2016, so there is a hole in the data for this period. The ER data confirms that Miscellaneous Trojans were the most frequent threat category encountered in 2013. Unfortunately, I could not find a published data source for the ER of Potentially Unwanted Software, so it's missing from *Figure 3.22*. The ER spike for Trojan Downloaders and Droppers in the second half of 2013 was due to three threats: Rotbrow, Brantall, and Sefnit (Microsoft, 2014).

At the end of this period, in the fourth quarter of 2017, ransomware had an ER of 0.13%, while Miscellaneous Trojans had an ER of 10.10%; that's a 195% difference. Although ransomware has a low ER, the impact of a ransomware infection can be devastating.

Thus, don't forget to look at both parts of a risk calculation, that is, the probability and the impact of threats. This is a trend that continues into the last quarter of 2019. It appears that the investments Microsoft made in memory safety features and other mitigations in Windows operating systems have helped drive down the global ER, despite increasing numbers of vulnerability disclosures in Windows. If ER is an indicator, the one tactic that the purveyors of malware seem to get a solid **Return on Investment** (**ROI**) from is social engineering.

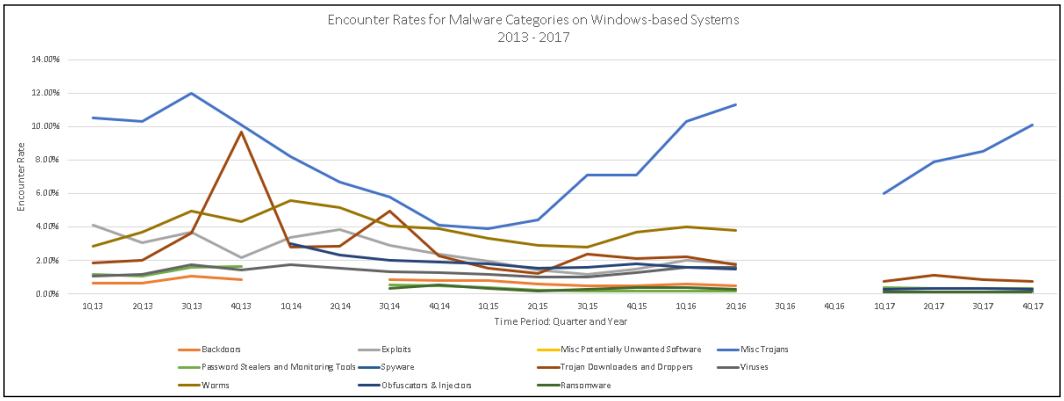


Figure 3.22: Encounter rates by threat category on Windows-based systems reporting detections, 2013–2017 (Microsoft Corporation, n.d.)

The vast majority of the data I just walked you through is from consumers' systems around the world that have reported data to Microsoft. There are some differences between the prevalence of threats on consumers' systems and in enterprises that security teams and cybersecurity experts should be aware of. After studying these differences for many years, I can summarize them for you. Three helpful insights from the data reported to Microsoft from enterprise environments are:

1. **Worms**: This was typically the number one category of threat in enterprise environments that were reported to Microsoft over the years. This category of malware self-propagates, which means worms can spread quickly and be very difficult to get rid of once they are inside of an enterprise environment. Worms can hide in enterprise IT environments and resurface quickly. For example, they can hide in storage area networks where no anti-virus software has been deployed.

They can hide in old desktop and server images that, when used to build new systems, reintroduce worms back into the environment. They can also be resurrected from backups when they are restored. Many CISOs I know battled worms like Conficker for years after their initial introduction into their environments.

These worms typically spread three ways: unpatched vulnerabilities, weak passwords, and social engineering. Sound familiar? They should, because these are three of the five cybersecurity usual suspects. Focusing on the cybersecurity fundamentals will help you keep worms out and contain those already inside your environment. Deploying up to date anti-malware everywhere is important to stop these threats.

2. **USB drives and other removable storage media**: Many threats, such as worms and viruses, are introduced into enterprise environments on USB drives. Putting policies in place that block USB port access on desktops and servers will prevent Information Workers from introducing such threats into your IT environment. Configuring anti-malware software to scan files on access, especially for removable media, will also help block these threats, many of which are well-known by anti-malware labs and are many years old.
3. **Malicious or compromised websites**: Drive-by download attacks and watering hole attacks expose Information Workers' systems to exploits and, if successful, malware. Carefully think about whether your organization really needs a policy that allows Information Workers to surf the internet unfettered. Does everyone in the organization need to get to every domain on the internet, even IP addresses in the countries with, consistently, the highest malware infection rates in the world? Only permitting workers to get to trusted sites that have a business purpose might not be a popular policy with them, but it will dramatically reduce the number of potential threats they are exposed to.

> This mitigation won't work for every organization because of the nature of their business, but I dare say that it will work for a lot more organizations than those that currently use it today. Think through whether unfettered access to the internet and visiting sites with content in foreign languages is really necessary for your staff, as well as whether the security team can make some changes that have high mitigation value and low or zero impact on productivity. Managed outbound proxy rules, IDS/IPS, and browser whitelists are all controls that can help.

And of course, patch, patch, patch! Drive-by download attacks don't work when the underlying vulnerabilities they rely on are patched. This is where those organizations that patch once a quarter or once per half really suffer; they allow their employees to go everywhere on the internet with systems they know have hundreds or thousands of publicly known vulnerabilities on them. What could possibly go wrong?

Global malware evolution conclusions

This malware category data shows us that purveyors of malware really are limited to only a few options when trying to initially compromise systems. Exploiting unpatched vulnerabilities is a reliable method for only limited periods of time, but this doesn't stop attackers from attempting to exploit old vulnerabilities for years after a security update has become available. Worms come in and out of fashion with attackers and require technical skills to develop. But the one tactic that is a mainstay tactic is social engineering. When the other four cybersecurity usual suspects are not viable options, many attackers will attempt to use good old-fashioned social engineering.

Despite all the malware data that I just shared with you, some cybersecurity experts still assert that anti-malware software isn't worthwhile for enterprises. Let's dive into this argument to see whether it holds water.

The great debate – are anti-malware solutions really worthwhile?

Allow me to offer my opinion on the efficacy of anti-malware software. Over the years, I've heard some cybersecurity experts at industry conferences ridicule the efficacy of anti-malware solutions and recommend that organizations not bother using such solutions. They tend to justify this point of view by pointing out that anti-malware software cannot detect and clean all threats. This is true. They also point out that the anti-malware solutions can have vulnerabilities themselves that can increase the attack surface area instead of reducing it. This is also true. Since anti-malware software typically has access to sensitive parts of operating systems and the data they scan, they can be an effective target for attackers. Some anti-malware vendors have even been accused of using the privileged access to systems that their products have, to provide illicit access to systems (Solon, 2017). Other vendors have been accused of improperly sharing information collected by their products (Krebs on Security, 2017).

But remember that malware purveyors are churning out *millions* of unique malware threats per week. As anti-malware labs around the world get samples of these threats, they inoculate their customers from them. So, while anti-malware solutions cannot protect organizations from all threats, especially new and emerging threats, it can protect them from hundreds of millions of known threats. On the other hand, if they don't run an anti-malware solution, they won't be protected from any of these threats. Do the risk calculation using recent data and I think you'll see that running anti-malware software is a no-brainer. For enterprises, failing to run up-to-date anti-malware software from a trustworthy vendor is gross negligence.

Not all anti-malware products are equal. In my experience, anti-malware vendors are only as good as the researchers, analysts, and support staff in their research and response labs. Vendors that minimize false positives while providing the best response times and detections for real-world threats can be very helpful to security teams. To compare products on these measures, check out the third-party testing results from AV-Test and AV Comparatives. There's been discussion in the anti-malware lab community for decades about the best way to test their products.

In the past, the debate has focused on how test results can be skewed based on the collection of malware samples that products are tested against. For example, if a particular lab is really good at detecting root kits, and the tests include more samples of root kits, then that anti-malware product might score better than average, even if it's sub-par at detecting other categories of threats. The opposite is also true—if the test doesn't include rootkits or includes very few rootkits, the product could score lower than average. Since anti-malware tests can't include every known malware sample, because of real-world resource constraints, whatever samples they do test will influence the score of the products tested. Some anti-malware labs have argued that this forces them to keep detections for older threats that are no longer prevalent, in their products, rather than allowing them to focus on current and emerging threats that their customers are more likely to encounter. The counter-argument is that anti-malware solutions should be able to detect all threats, regardless of their current prevalence. The tests and the industry continue to evolve with better tests, more competitors, and novel approaches to detecting, blocking, and disinfecting threats. Many vendors have evolved their products far beyond simple signature-based detection systems by leveraging heuristics, behavioral analysis, AI, ML, and cloud computing, among other methods.

This concludes my marathon discussion on malware, anti-malware solutions, and the global Windows threat landscape. I feel like I have only scratched the surface here, but we have so many other interesting topics to discuss! Before we come to the end of this chapter, let me share some best practices and tips related to consuming threat intelligence.

Threat intelligence best practices and tips

I want to give you some guidance on how to identify good threat intelligence versus questionable threat intelligence. After publishing one of the industry's best threat intelligence reports for the better part of a decade (OK, I admit I'm biased), I learned a few things along the way that I'll share with you here. The theme of this guidance is to understand the methodology that your threat intelligence vendors use.

If they don't tell you what their methodology is, then you can't trust their data, period. Additionally, the only way you'll be able to truly understand if or how specific threat intelligence can help your organization is to understand its data sources, as well as the methodology used to collect and report the data; without this context, threat intelligence can be distracting and the opposite of helpful.

Tip #1 – data sources

Always understand the sources of threat intelligence data that you are using and how the vendors involved are interpreting the data. If the source of data is unknown or the vendors won't share the source of the data, then you simply cannot trust it and the interpretations based on it. For example, a vendor claims that 85% of all systems have been successfully infected by a particular family of malware. But when you dig into the source of the data used to make this claim, it turns out that 85% of systems that used the vendor's online malware cleaner website were infected with the malware referenced. Notice that "85% of all systems" is a dramatic extrapolation from "85% of all systems that used their online tool."

Additionally, the online tool is only offered in US English, meaning it's less likely that consumers who don't speak English will use it, even if they know it exists. Finally, you discover that the vendor's desktop anti-virus detection tool refers users to the online tool to get disinfected when it finds systems to be infected with the threat. The vendor does this to drive awareness that their super great online tool is available to their customers. This skews the data as 100% of users referred to the online tool from the desktop anti-virus tool were already known to be infected with that threat. I can't count how many times I've seen stunts like this over the years. Always dive deep into the data sources to understand what the data actually means to you.

Tip #2 – time periods

When consuming threat intelligence, understanding the time scale and time periods of the data is super important. Are the data and insights provided from a period of days, weeks, months, quarters, or years? The answer to this question will help provide the context required to understand the intelligence. The events of a few days will potentially have a much different meaning to your organization than a long-term trend over a period of years.

Anomalies will typically warrant a different risk treatment than established patterns. Additionally, the conclusions that can be made from threat intelligence data can be dramatically altered based on the time periods the vendor uses in their report.

Let me provide you with an example scenario. Let's say a vendor is reporting on how many vulnerabilities were disclosed in their products for a given period. If the data is reported in regular sequential periods of time, such as quarterly, the trend looks really bad as large increases are evident. But instead of reporting the trend using sequential quarterly periods, the trend looks much better when comparing the current quarter to the same quarter last year; there could actually be a decrease in vulnerability disclosures in the current quarter versus the same quarter last year. This puts a positive light on the vendor, despite an increase in vulnerability disclosures quarter over quarter.

Another potential red flag is when you see vendor report data that isn't for a normal period of time, such as monthly, quarterly, or annually. Instead, they use a period of months that seems a little random. If the time period is irregular or the reason it's used isn't obvious, the rational should be documented with the threat intelligence. If it's not, ask the vendor why they picked the time periods they picked. Sometimes, you'll find vendors use a specific time period because it makes their story more dramatic, garnering more attention, if that's their agenda. Or the period selected might help downplay bad news by minimizing changes in the data. Understanding why the data is being reported in specific time scales and periods will give you some idea about the credibility of the data, as well as the agenda of the vendor providing it to you.

Tip #3 – recognizing hype

One of the biggest mistakes I've seen organizations make when consuming threat intelligence is accepting their vendor's claims about the scope, applicability, and relevance of their data. For example, a threat intelligence vendor publishes data that claims 100% of attacks in a specific time period involved social engineering or exploited a specific vulnerability. The problem with such claims is that no one in the world can see 100% of all attacks, *period*.

They'd have to be omniscient to see all attacks occurring everywhere in the world simultaneously, on all operating systems and cloud platforms, in all browsers and applications. Similarly, claims such as 60% of all attacks were perpetrated by a specific APT group are not helpful. Unless they have knowledge of 100% of attacks, they can't credibly make claims about the characteristics of 60% of them. A claim about the characteristics of all attacks or a subset that requires knowledge of all attacks, even when referencing specific time periods, specific locations, and specific attack vectors, simply isn't possible or credible. A good litmus test for threat intelligence is to ask yourself, does the vendor have to be omniscient to make this claim? This is where understanding the data sources and the time periods will help you cut through the hype and derive any value the intelligence might have.

Many times, the vendor publishing the data doesn't make such claims directly in their threat intelligence reports, but the way new intelligence is reported in the headlines is generalized or made more dramatic in order to draw attention to it. Don't blame threat intelligence vendors for the way the news is reported, as this is typically beyond their control. But if they make such claims directly, recognize it and adjust the context in your mind appropriately. For many years, I made headlines around the world regularly speaking and writing about threats, but we were always very careful not to overstep the mark from conclusions supported by the data. To make bolder claims would have required omniscience and omnipotence.

Tip #4 – predictions about the future

I'm sure you've seen some vendors make predictions about what's going to happen in the threat landscape in the future. One trick that some threat intelligence vendors like to use is related to time periods again. Let's say I'm publishing a threat intelligence report about the last 6-month period. By the time the data for this period is collected and the report is written and published, a month or two might have gone by. Now, if I make a prediction about the future in this report, I have a month or two of data that tells me what's been happening since the end of the reporting period.

If my prediction is based on what the data tells us already happened, readers of the report will be led to believe that I actually predicted the future accurately, thus reinforcing the idea that we know more about the threat landscape than anyone else. Understanding when the prediction was made relative to the time period it was focused on will help you decide how credible the prediction and results are, and how trustworthy the vendor making the prediction is. Remember, predictions about the future are guesses.

Tip #5 – vendors' motives

Trust is a combination of credibility and character. You can use both to decide how trustworthy your vendors are. Transparency around data sources, time scales, time periods, and predictions about the future can help vendors prove they are credible. Their motives communicate something about their character. Do they want to build a relationship with your organization as a trusted advisor or is their interest limited to a financial transaction? There's a place for both types of vendors when building a cybersecurity program, but knowing which vendors fall into each category can be helpful, especially during incident response-related activities, when the pressure is on. Knowing who you can rely on for real help when you need it is important.

Those are the tips and tricks I can offer you from 10 years of publishing threat intelligence reports. Again, the big take-away here is understanding the methodology and data sources of the threat intelligence you consume—this context is not optional. One final word of advice: do not consume threat intelligence that doesn't meet this criterion. There is too much fear, uncertainty, and doubt, and too much complexity in the IT industry. You need to be selective about who you take advice from.

I hope you enjoyed this chapter. Believe it or not, this type of data is getting harder and harder to find. The good news is that threat intelligence is being integrated into cybersecurity products and services more and more, which means protecting, detecting, and responding to threats is easier and faster than ever.

Chapter summary

This chapter required a lot of research. I tried to provide you with a unique long-term view of the threat landscape and some useful context. I'll try to summarize the key take-aways from this chapter.

Malware uses the cybersecurity usual suspects to initially compromise systems; these include unpatched vulnerabilities, security misconfigurations, weak, leaked, and stolen passwords, insider threat, and social engineering. Of these, social engineering is attackers' favorite tactic, as evidenced by the consistently high prevalence of malware categories that leverage it. Malware can also be employed after the initial compromise to further attackers' objectives.

Some successful malware families impact systems around the world quickly after release, while others start as regional threats before growing into global threats. Some threats stay localized to a region because they rely on a specific non-English language to trick users into installing them. Regions have different malware encounter and infection rates. Research conducted by Microsoft indicates that some socio-economic factors, such as GDP, could be influencing these differences. Regions with unusually high levels of strife and the socio-economic conditions that accompany it, typically have higher malware encounter and infection rates.

Focusing on the cybersecurity fundamentals, which address the cybersecurity usual suspects, will help mitigate malware threats. In addition, running up-to-date anti-malware solutions from a trusted vendor will help block installation of most malware and disinfect systems that get infected. Blocking Information Workers' access to regions of the internet that do not have legitimate business purposes, can help prevent exposure to malware and compromised systems in these regions.

So far, we've examined the long-term trends for two important types of threats, vulnerabilities, and malware. In the next chapter, we'll explore the ways attackers have been using the internet and how these methods have evolved over time.

References

1. Aljazeera (January 4, 2014). *Iraq government loses control of Fallujah*. Retrieved from Aljazeera.com: https://www.aljazeera.com/news/middleeast/2014/01/iraq-government-loses-control-fallujah-20141414625597514.html
2. Authority, R. O. (2019). *Estonian Information System Authority Annual Cyber Security Assessment 2019*. Republic of Estonia Information System Authority.
3. AV-Test Institute (2017). *The AV-TEST Security Report 2016/2017*. Magdeburg, Germany: AV-Test Institute
4. AV-Test Institute (2018). *The AV-TEST Security Report 2017/2018*. Magdeburg, Germany: AV-Test Institute
5. AV-Test Institute (April 2019). *The AV-TEST Security Report 2018/2019*. Magdeburg, Germany: AV-Test Institute. Retrieved from AV-Test: https://www.av-test.org/fileadmin/pdf/security_report/AV-TEST_Security_Report_2018-2019.pdf
6. AV-Test Institute (April, 2020). *About the AV-TEST Institute. Retrieved from AV-Test*: https://www.av-test.org/en/about-the-institute/
7. AV-Test Institute (April, 2020). *AV-Test Malware Statistics*. Retrieved from AV-Test: https://www.av-test.org/en/statistics/malware/
8. AV-Test Institute (April, 2020). *International Presence and Publications*. Retrieved from AV-Test Institute: https://www.av-test.org/en/about-the-institute/publications/
9. David Burt, P. N. (2014). *The Cybersecurity Risk Paradox, Microsoft Security Intelligence Report Special Edition*. Microsoft. Retrieved from: https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/REVroz
10. David Ladd, F. S. (2011). *The SDL Progress Report*. Microsoft. Retrieved from: http://download.microsoft.com/download/c/e/f/cefb7bf3-de0c-4dcb-995a-c1c69659bf49/sdlprogressreport.pdf

11. Ece Toksabay (February 22, 2014). *Police fire tear gas at Istanbul anti-government protest*. Retrieved from Reuters: https://www.reuters.com/article/us-turkey-protest/police-fire-tear-gas-at-istanbul-anti-government-protest-idUSBREA1L0UV20140222
12. Keizer, G. (January 4, 2020). *Windows by the numbers: Windows 10 resumes march towards endless dominance*. Retrieved from Computerworld UK: https://www.computerworld.com/article/3199373/windows-by-the-numbers-windows-10-continues-to-cannibalize-windows-7.html
13. Keizer, G. (n.d.). *Windows by the numbers: Windows 10 resumes march towards endless dominance*. Retrieved from Computer World UK: https://www.computerworld.com/article/3199373/windows-by-the-numbers-windows-10-continues-to-cannibalize-windows-7.html
14. Krebs on Security (August 17, 2017). *Carbon Emissions: Oversharing Bug Puts Security Vendor Back in Spotlight*. Retrieved from Krebs on Security: https://krebsonsecurity.com/2017/08/carbon-emissions-oversharing-bug-puts-security-vendor-back-in-spotlight/
15. Leyden, J. (n.d.). *Microsoft releases Blaster clean-up tool*. Retrieved from The Register: https://www.theregister.co.uk/2004/01/07/microsoft_releases_blaster_cleanup_tool/
16. Microsoft (2014). *Microsoft Security Intelligence Report Volume 16*. Retrieved from Microsoft Security Intelligence Report Volume 16: https://go.microsoft.com/fwlink/p/?linkid=2036139&clcid=0x409&culture=en-us&country=us
17. Microsoft (December 14, 2016). *Microsoft Security Intelligence Report. Microsoft Security Intelligence Report Volume 21*. Retrieved from Microsoft Security Intelligence Report Volume 21: https://go.microsoft.com/fwlink/p/?linkid=2036108&clcid=0x409&culture=en-us&country=us
18. Microsoft Corporation (April 8, 2019). *Microsoft Security Intelligence Report Volume.* Retrieved from Microsoft Security Intelligence Report Volume 6: https://go.microsoft.com/fwlink/p/?linkid=2036319&clcid=0x409&culture=en-us&country=us

19. Microsoft Corporation (2015). *VIRTUAL EXCLUSIVE: Cyberspace 2025: What will Cybersecurity Look Like in 10 Years? Microsoft*. Retrieved from: https://channel9.msdn.com/Events/Virtual-CIO-Summit/Virtual-CIO-Summit-2015/VIRTUAL-EXCLUSIVE-Cyberspace-2025-What-will-Cybersecurity-Look-Like-in-10-Years
20. Microsoft Corporation (July 7, 2016). *Microsoft Security Intelligence Report Volume 20*. Retrieved from Microsoft Security Intelligence Report Volume 20: https://go.microsoft.com/fwlink/p/?linkid=2036113&clcid=0x409&culture=en-us&country=us
21. Microsoft Corporation (August 17, 2017). *Microsoft Security Intelligence Report Volume 22*. Retrieved from Microsoft Security Intelligence Report Volume 22: https://go.microsoft.com/fwlink/p/?linkid=2045580&clcid=0x409&culture=en-us&country=us
22. Microsoft Corporation (2018). *Microsoft Security Intelligence Report Volume 23*. Retrieved from Microsoft Security Intelligence Report Volume 23: https://go.microsoft.com/fwlink/p/?linkid=2073690&clcid=0x409&culture=en-us&country=us
23. Microsoft Corporation (August 10, 2019). *Industry collaboration programs*. Retrieved from Microsoft: https://docs.microsoft.com/en-us/windows/security/threat-protection/intelligence/cybersecurity-industry-partners
24. Microsoft Corporation (August 13, 2019). *Patch new wormable vulnerabilities in Remote Desktop Services (CVE-2019-1181/1182)*. Retrieved from Microsoft Security Response Center Blog: https://msrc-blog.microsoft.com/2019/08/13/patch-new-wormable-vulnerabilities-in-remote-desktop-services-cve-2019-1181-1182/
25. Microsoft Corporation (n.d.). *Diplugem description*. Retrieved from Microsoft Security Intelligence: https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Win32/Diplugem&threatId=
26. Microsoft Corporation (n.d.). *DirectAccess*. Retrieved from Microsoft Corporation: https://docs.microsoft.com/en-us/windows-server/remote/remote-access/directaccess/directaccess

27. Microsoft Corporation (n.d.). *How Microsoft identifies malware and potentially unwanted applications*. Retrieved from Microsoft Corporation: `https://docs.microsoft.com/en-us/windows/security/threat-protection/intelligence/criteria`
28. Microsoft Corporation (n.d.). *Malware encounter rates*. Retrieved from Microsoft Security Intelligence Report: `https://www.microsoft.com/securityinsights/Malware`
29. Microsoft Corporation (n.d.). *Microsoft Security Intelligence Report*. Retrieved from Microsoft Security
30. Microsoft Corporation (n.d.). *Over a decade of reporting on the threat landscape*. Retrieved from Microsoft Corporation: `https://www.microsoft.com/en-us/security/operations/security-intelligence-report`
31. Microsoft Corporation (n.d.). *Petya description*. Retrieved from Microsoft Security Intelligence: `https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Ransom:DOS/Petya.A&threatId=-2147257025`
32. Microsoft Corporation (n.d.). *Prevent a worm by updating Remote Desktop Services (CVE-2019-0708)*. Retrieved from Microsoft Security Response Center blog: `https://msrc-blog.microsoft.com/2019/05/14/prevent-a-worm-by-updating-remote-desktop-services-cve-2019-0708/`
33. Microsoft Corporation (n.d.). *Remove specific prevalent malware with Windows Malicious Software Removal Tool*. Retrieved from Microsoft Corporation: `https://support.microsoft.com/en-us/help/890830/remove-specific-prevalent-malware-with-windows-malicious-software-remo`
34. Microsoft Corporation (n.d.). *Remove specific prevalent malware with Windows Malicious Software Removal Tool*. Retrieved from Microsoft Corporation: `https://support.microsoft.com/en-us/help/890830/remove-specific-prevalent-malware-with-windows-malicious-software-remo#covered`
35. Microsoft Corporation (n.d.). *Reveton description*. Retrieved from Microsoft Security Intelligence: `https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Ransom:Win32/Reveton.T!lnk&threatId=-2147285370`

36. Microsoft Corporation (n.d.). *Rotbrow description*. Retrieved from Microsoft Security Intelligence: https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?name=win32%2frotbrow

37. Microsoft Corporation (n.d.). *Sality description*. Retrieved from Microsoft Security Intelligence: https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Virus%3aWin32%2fSality

38. Microsoft Corporation (n.d.). *Sefnit description*. Retrieved from Microsoft Security Intelligence: https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Win32/Sefnit

39. Microsoft Corporation (n.d.). *SmartScreen: FAQ*. Retrieved from Microsoft Corporation: https://support.microsoft.com/en-gb/help/17443/windows-internet-explorer-smartscreen-faq

40. Microsoft Corporation (n.d.). *Taterf description*. Retrieved from Microsoft Security Intelligence: https://www.microsoft.com/en-us/wdsi/threats/malware-encyclopedia-description?Name=Win32/Taterf&threatId=

41. Microsoft Corporation (n.d.). *Virus alert about the Blaster worm and its variants*. Retrieved from Microsoft Corporation: https://support.microsoft.com/en-us/help/826955/virus-alert-about-the-blaster-worm-and-its-variants

42. NIST (n.d.). *CVE-2019-0708 Detail*. Retrieved from National Vulnerability Database: https://nvd.nist.gov/vuln/detail/CVE-2019-0708

43. Rains, T. (June 27, 2011). *Defending Against Autorun Attacks*. Retrieved from Microsoft Official Security Blog: https://www.microsoft.com/security/blog/2011/06/27/defending-against-autorun-attacks/

44. Rains, T. (September 24, 2013). *Examining Korea's Rollercoaster Threat Landscape*. Retrieved from Microsoft Official Security Blog: https://www.microsoft.com/security/blog/2013/09/24/examining-koreas-rollercoaster-threat-landscape/

45. Rains, T. (n.d.). *New Microsoft Malware Protection Center Threat Report Published: EyeStye*. Retrieved from Microsoft Official Security Blog: https://www.microsoft.com/security/blog/2012/07/20/new-microsoft-malware-protection-center-threat-report-published-eyestye/

46. Republic of Estonia Information System Authority (2018). *Estonian Information System Authority: Annual Cyber Security Assessment 2018. Republic of Estonia Information System Authority*. Retrieved from: https://www.ria.ee/sites/default/files/content-editors/kuberturve/ria-csa-2018.pdf
47. Solon, O. (September 13, 2017). *US government bans agencies from using Kaspersky software over spying fears*. Retrieved from The Guardian: https://www.theguardian.com/technology/2017/sep/13/us-government-bans-kaspersky-lab-russian-spying
48. *Turkey's Premier Is Proclaimed Winner of Presidential Election (August 10, 2014)*. Retrieved from The New York Times: https://www.nytimes.com/2014/08/11/world/europe/erdogan-turkeys-premier-wins-presidential-election.html?_r=0/
49. US Department of Homeland Security (n.d.). *CRITICAL INFRASTRUCTURE SECTORS*. Retrieved from CISA Cyber Infrastructure: https://www.dhs.gov/cisa/critical-infrastructure-sectors
50. Wikipedia (n.d.). *2014 in Iraq*. Retrieved from Wikipedia.com: https://en.wikipedia.org/wiki/2014_in_Iraq
51. Wikipedia (n.d.). *2014 in Pakistan*. Retrieved from Wikipedia.com: https://en.wikipedia.org/wiki/2014_in_Pakistan
52. Wikipedia (n.d.). *Next-Generation Secure Computing Base*. Retrieved from Wikipedia.com: https://en.wikipedia.org/wiki/Next-Generation_Secure_Computing_Base
53. Wikipedia (n.d.). *Timeline of the Arab Spring*. Retrieved from Wikipedia.com: https://en.wikipedia.org/wiki/Timeline_of_the_Arab_Spring

4

Internet-Based Threats

Over the past 25 years, attackers have learned to leverage the internet to compromise the IT environments of their victims, achieve their illicit objectives and satisfy their motivations. CISOs and Security teams can inform their cybersecurity strategies by studying how attackers use the internet. In this chapter, we'll look at some of the ways attackers have been using the internet and how these methods have evolved over time.

In this chapter, we'll look at the following topics:

- Phishing attacks
- Drive-by download attacks
- Malware hosting sites

Let's get started by looking at the anatomy of a typical attack pattern.

Introduction

In the last two chapters, I provided a deep examination of data and trends for vulnerability disclosures and malware. Both types of threats are constantly leveraged by attackers seeking to compromise organizations and consumers around the world. Subsequently, the risk that these threats represent are actively managed by enterprises. But the ways that attackers deliver their weapons, whether they are exploits for vulnerabilities or malware that provides illicit backdoors for attackers, are varied.

In this chapter, we'll look at some of the methods attackers use to attack their victims; understanding these are just as important as understanding how vulnerabilities and malware have evolved.

The threats we've examined so far have the potential to enable attackers to compromise applications, clients, servers, consumer and IoT devices, routing and switching equipment, and other systems that enterprises rely on. Whether these attacks are designed to victimize massive numbers of organizations and consumers, or are targeted at specific organizations, attackers will use the cybersecurity usual suspects to initially compromise IT systems. As a reminder, these include:

- Unpatched vulnerabilities
- Security misconfigurations
- Weak, leaked, and stolen credentials
- Social engineering
- Insider threat

It's rare that an attacker is physically sitting at the keyboard of the system they are attempting to compromise. The vast majority of attackers perpetrate their attacks remotely over networks, none more than the internet. The same way that the internet has allowed small businesses to compete with large multinationals, it enables individuals and small groups to attack a massive number of consumers and the world's largest organizations.

Now let's look at a typical attack pattern as an example of how attackers have learned to leverage the internet.

A typical attack

In this fictional example, the attacker is physically located in Australia and the intended victim of the attack is headquartered in the United States. The attacker's motivation is profit and they seek to steal valuable information from the organization they are targeting and sell it.

The intended victim has a CISO and a Security team. The attacker's constant vulnerability scans of the victim's perimeter reveal that they are proficient at vulnerability management, as vulnerabilities on internet facing systems are quickly and efficiently patched. After doing some research on the victim organization, the attacker decides to use a multi-pronged approach to initially compromise the organization.

The attacker has always been successful, one way or another, using social engineering to trick non-technical business people into making poor trust decisions that could be capitalized on. A poor trust decision in this context is where the victim decides to open an attachment or click on a URL in an email, lower their system's security settings, open firewall ports, or take other such actions that enables the attacker to more easily victimize them. In this case, the attacker is going to use two different tactics to try to compromise a few Information Workers' laptops, with the goal of getting access to their email inboxes. Both tactics will leverage email as a delivery mechanism and rely on social engineering and sloppy security mitigations to succeed.

The first tactic is to send phishing emails to specific individuals the attacker has identified as working in the company's Finance department using the company's website. It didn't take long to get a list of email addresses for the people the attacker wanted to target. The goal of the phishing emails is to trick one or more of the targeted Information Workers into sharing their Office 365 credentials, which the attacker can then use to access their email inbox.

The second tactic is to send emails to the same Information Workers that contain a malicious link to a drive-by download site. If the Information Workers take the bait and click on the link, their web browser will take them to a malicious web page that will expose them to several exploits for browser and operating system vulnerabilities. If their client isn't fully patched, there's a good chance that the attacker will be able to install a backdoor onto their system that might allow them to get privileged access to the victim's laptop and, ultimately, to their email.

Of course, if the attacker does get privileged access to the victim's laptop, they might be able to harvest all sorts of other valuable information in addition to email. Examples include documents stored locally on the laptop, contact lists, access to social networking accounts, software license keys, expense and credit card information, banking information and credentials, personal information that can be used for identity theft, and so on. If the laptop is passively managed by IT, it could be used to store illicit material, enrolled in a botnet and used in attacks against other targets. For example, it could be used for spam and phishing campaigns, to host drive-by download attacks, host malware, advertising click-fraud, DDoS attacks, or whatever "project work" the attacker decides to undertake.

Additionally, the attacker could sell or trade away any of the information they pilfered, including account credentials. The criminals they give this information to could turn out to be located much closer to the victim and much more aggressive at leveraging the information to maximize their profit and/or damage to the victim.

This type of attack is all too typical. It involved three of the five cybersecurity usual suspects, including social engineering, unpatched vulnerabilities, and stolen credentials. Let's now take a closer look at some of these methods, how they work, and how popular they really are. To do this, I'll draw on threat intelligence and data that has been published by industry leaders over the years. Let's start by looking at phishing.

Phishing attacks

Social engineering is a mainstay tactic for attackers around the world. Phishing is at the intersection of two of the cybersecurity usual suspects; social engineering and weak, leaked, and stolen passwords. Many of the largest data breaches in history started with a phishing attack. In simple terms, phishing is a social engineering tactic where the attacker tries to trick their victim into sharing confidential information with them. Attackers use emails, websites, and advertising to entice people into disclosing account credentials, personal details, credit card and financial account information, among other things. The information that victims disclose might be used to illegally access online accounts, conduct illegal financial transactions, and steal the victims' identities, among other purposes.

Some attackers cast an indiscriminate wide net for their phishing attacks to snare as many people as possible in order to increase the odds of success. Some attackers focus their phishing activities on an industry or group of targets. Spearing phishing is used to focus attacks on individuals, presumably because they have access to information or wealth that the attacker desires.

Very often, after attackers successfully compromise an Information Worker's system, the victims' own contact lists are used to attack their friends, family, co-workers, and business contacts. For example, once a victim's social networking account has been compromised, attackers can use the victim's account to communicate with the victim's social network. Since the communications are seemingly coming from a trusted source, others in the victim's social network are easily tricked by phishing emails and websites shared via the victim's account. Attackers do not limit themselves to attacking their target's corporate accounts and will seek to compromise the personal systems of Information Workers knowing that these systems often have remote access to corporate assets. Installing keyloggers or other types of malware to automate the collection of data from victims' systems is common.

Phishing attacks can involve several technology components, including the victims' clients and the infrastructure used to attack the victims. For example, the email servers that phishing emails originate or the web servers that phishing pages are hosted on. Very often, these email servers and web servers are hosted on legitimate systems that have been compromised and are subsequently used for phishing campaigns. Botnets, which are potentially large networks of compromised systems that are being illicitly remote controlled, are commonly used for phishing campaigns. Using compromised systems for phishing campaign infrastructure reduces the costs for attackers, protects their identities, and helps them achieve a scale they likely could not by any other means. The availability of phishing kits makes it easy for almost anyone to wage a phishing attack.

Let's take a closer look at where phishing sites are hosted and where their victims are. First, it's important to realize the scale of this problem. By volume, phishing, along with Trojans (as I discussed in *Chapter 3*, *The Evolution of the Threat Landscape – Malware*), are the tactics attackers use most. Just how many phishing websites are there?

Good sources of data for phishing sites are internet search engines and web browsers. After all, Google and Bing are constantly indexing billions of web pages on the internet so that searches can result in fast, accurate results. Additionally, many millions of people use Google Chrome and Microsoft web browsers to surf the internet. Browsers allow users to report sites that are suspicious or outright unsafe. Google and Microsoft employ capabilities in their browsers and search engines to look for phishing sites, malware hosting sites, and other types of malicious websites. Then they help users of their products and services avoid the malicious sites they find by integrating continuously updated lists of malicious URLs and IP addresses into their products and services. Both browsers and search engines, among other services, can warn users when they attempt to visit a known malicious website, such as a phishing site. This generates data on malicious websites that both Google and Microsoft periodically publish.

For example, Google's technology that looks for malicious websites is called Safe Browsing. This is how Google describes it:

> *"Google's Safe Browsing technology examines billions of URLs per day looking for unsafe websites. Every day, we discover thousands of new unsafe sites, many of which are legitimate websites that have been compromised. When we detect unsafe sites, we show warnings on Google Search and in web browsers. You can search to see whether a website is currently dangerous to visit."*
>
> *– (Google, 2020)*

In 2019, Google's Safe Browsing detected 32,677 new phishing sites per week, on average. This volume is reflected in *Figure 4.1*. Factors that likely influence the volume of new phishing sites include the number of people employing social engineering tactics, the availability of phishing kits and other automation (like Botnets) that help facilitate attacks, continued low operating costs, and acceptable success rates.

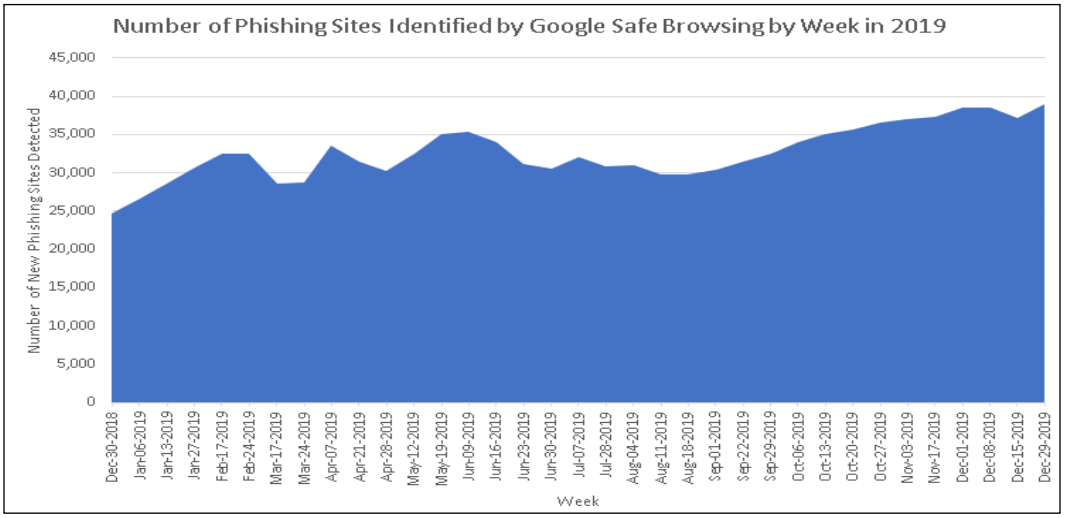


Figure 4.1: The number of phishing websites detected by Google Safe Browsing by week in 2019 (Google, 2020). Google and the Google logo are registered trademarks of Google LLC, used with permission.

Google, Microsoft, and many other organizations have tried to make it easy for consumers and enterprises to report phishing sites. When phishing sites are reported or detected, legal and technical processes are employed to take down these malicious sites.

This process seemed to contain the number of phishing sites on the internet to below 200,000 in the six and a half years leading up to the fourth quarter of 2015, according to the data Google has published (Google, 2020). Then in the fourth quarter of 2015, the number of phishing sites started an expansion that hasn't receded. In April 2020, Google reported that Safe Browsing had detected more than 1.8 million phishing sites (Google, 2020). That's nearly a 1,000% increase in the number of phishing sites on the internet between August 2015 and April 2020. Much of this dramatic increase is illustrated in *Figure 4.2*, which provides the average number of phishing sites in each quarter between the first quarter of 2017 (1Q17) and the first quarter of 2020 (1Q20).

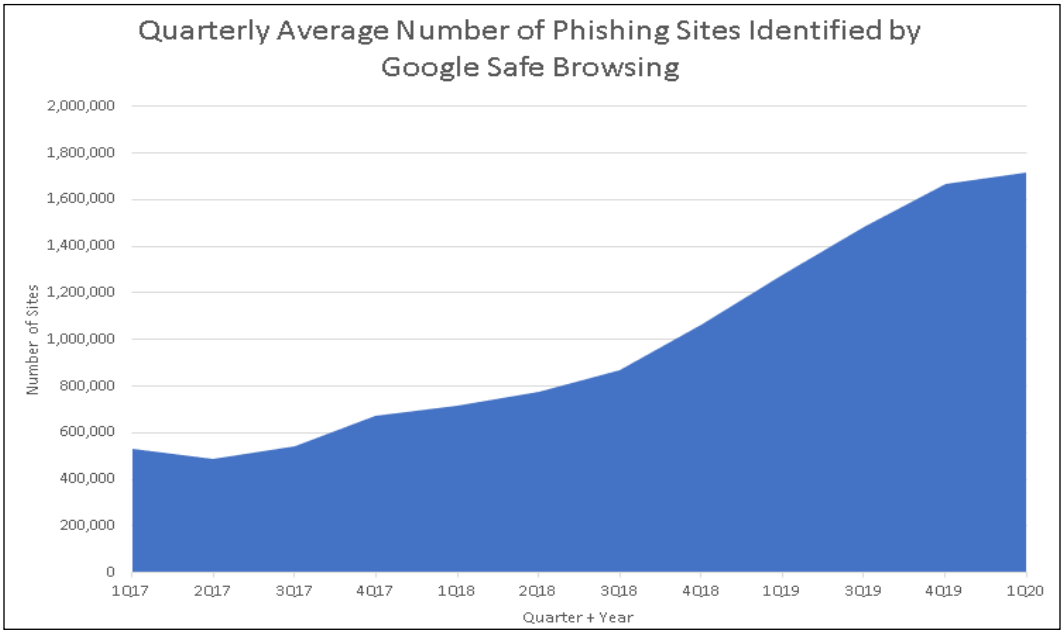


Figure 4.2: The average number of phishing websites Google Safe Browsing identified by quarter in 2017-2020 (Google, 2020). Google and the Google logo are registered trademarks of Google LLC, used with permission.

The volume of phishing emails has also increased over time. A great source of data on phishing emails are massive email services, like Microsoft Office 365 and Google Gmail, among others, that receive and filter phishing requests for enterprise customers around the world. Microsoft reported a huge increase in phishing emails going to recipients using Office 365 in 2018:

> "*Microsoft analyzes and scans in Office 365 more than 470 billion email messages every month for phishing and malware, which provides analysts with considerable insight into attacker trends and techniques The share of inbound emails that were phishing messages increased 250 percent between January and December 2018.*"
>
> *– (Microsoft Corporation, 2019)*

Microsoft indicated that the peak month for phishing emails in 2018 was November, where 0.55% of total inbound emails were phishing emails; that is the equivalent of 2,585,000,000 phishing emails in one month (Microsoft Corporation, 2019).

July 2019 appears to be the month with the highest levels in the 2018/2019 time period, with 0.85% of phishing emails detected out of the total volume of emails analyzed by Microsoft worldwide. Assuming the same 470 billion email message volume per month, this is equivalent to 3,995,000,000 phishing email messages in one month. Of course, there are many other on-premises and online email services that receive significant volumes of phishing emails that are not captured in these figures.

For example, in August 2019, Google revealed that it was blocking 100 million phishing emails every day:

> *"The roughly 100 million phishing emails Google blocks every day fall into three main categories: highly targeted but low-volume spear phishing aimed at distinct individuals, "boutique phishing" that targets only a few dozen people, and automated bulk phishing directed at thousands or hundreds of thousands of people."*
>
> *– (Pegoraro, 2019)*

That's approximately 3 billion phishing emails per month on average, in the same ballpark as Microsoft. The volumes of phishing emails and the number of active phishing sites make phishing attackers' most widely used tactic. Most phishing emails include a hyperlink to a phishing website. "More than 75% of phishing mails include malicious URLs to phishing sites." (Microsoft Corporation, 2018). Phishing emails typically attempt to take advantage of popular sports and social events, crisis situations, strife, the offer of sales and opportunities, as well as claims of overdue bills, bank account issues, and package shipping glitches, to play on the emotions of their victims and create a sense of urgency. Phishers will use any topic to grab potential victims' attention and compel them to take action that ultimately leads to poor trust decisions and information disclosure.

Frequent targets for phishing attacks include online services, financial sites, social networking sites, e-commerce sites, and so on. *The Anti-Phishing Working Group's Phishing Activity Trends Report for the 4th Quarter of 2019* indicates that SaaS/Webmail (30.80%), Payment (19.80%), and Financial Institutions (19.40%) were the most frequently targeted sectors for phishing attacks during the quarter (Phishing Activity Trends Report 4th Quarter 2019, 2020).

Also noteworthy, the report indicates that:

> "*Phishing against Social Media targets grew every quarter of the year, doubling over the course of 2019.*"
>
> *– (Phishing Activity Trends Report 4th Quarter 2019, 2020)*

Phishing websites can be hosted anywhere in the world. Some locations have higher concentrations of phishing sites than others. *Table 4.1* illustrates the locations with higher than average concentrations of phishing sites as published in numerous volumes (v19, v21, v22, v23) of the Microsoft Security Intelligence Report (SIR) that are available for download at `www.microsoft.com/sir`. The time periods reflected include the first quarter of 2015 (1Q15), the first half of 2015 (1H15), the first half of 2016 (1H16), March 2017, and the second half of 2017 (2H17).

Period	Location	Phishing sites per 1,000 Internet Hosts	Worldwide Average	Data Source
1Q15	Bulgaria	98.5	5	Microsoft SIRv19
1H15	Libya	15.6	5	Microsoft SIRv19
1H15	Belize	14.5	5	Microsoft SIRv19
1H16	Ukraine	18.8	9.1	Microsoft SIRv21
1H16	South Africa	15.4	9.1	Microsoft SIRv21
1H16	Australia	14.5	9.1	Microsoft SIRv21
March 2017	Ukraine	13.2	6.3	Microsoft SIRv22
March 2017	South Africa	10.3	6.3	Microsoft SIRv22
March 2017	Indonesia	9.6	6.3	Microsoft SIRv22
March 2017	Denmark	9.7	6.3	Microsoft SIRv22
2H17	Ukraine	19.1	5.8	Microsoft SIRv23
2H17	Belarus	12.3	5.8	Microsoft SIRv23
2H17	Bulgaria	12.2	5.8	Microsoft SIRv23
2H17	Indonesia	10.8	5.8	Microsoft SIRv23

Table 4.1: Locations with higher than average concentrations of phishing sites 2015–2017 (Microsoft Corporation, 2015 -2017)

You'll notice that some locations are on this list more than once, like Bulgaria, Indonesia, South Africa, and Ukraine. The biggest deviation from the worldwide average was Bulgaria in the first quarter of 2015, that had nearly twenty times the number of phishing sites than the average.

Recall from *Chapter 3*, *The Evolution of the Threat Landscape – Malware*, leading up to the second half of 2017, Indonesia's malware infection rate was nearly three times the worldwide average. The high number of compromised systems can, in some cases, help partially explain why so many phishing sites are hosted in Indonesia. It's a similar situation in Bulgaria and Ukraine, although they didn't have quite as elevated malware infection and encounter rates as Indonesia had during that time period.

But, it's not always the case that locations with higher numbers of compromised systems also have higher levels of malicious websites. In fact, there are too many exceptions to this to call it a rule. For example, take the locations in *Table 4.1* that had the highest number of phishing sites in the first half of 2016 (1H16). These locations include Ukraine, South Africa, and Australia. South Africa's malware infection rate (CCM) is nearly twice the worldwide average during this period; the number of phishing sites per 1,000 internet hosts is also nearly double the worldwide average. However, the figures for Ukraine and Australia are not consistent with South Africa. They both have above average levels of phishing sites but have below average malware infection rates:

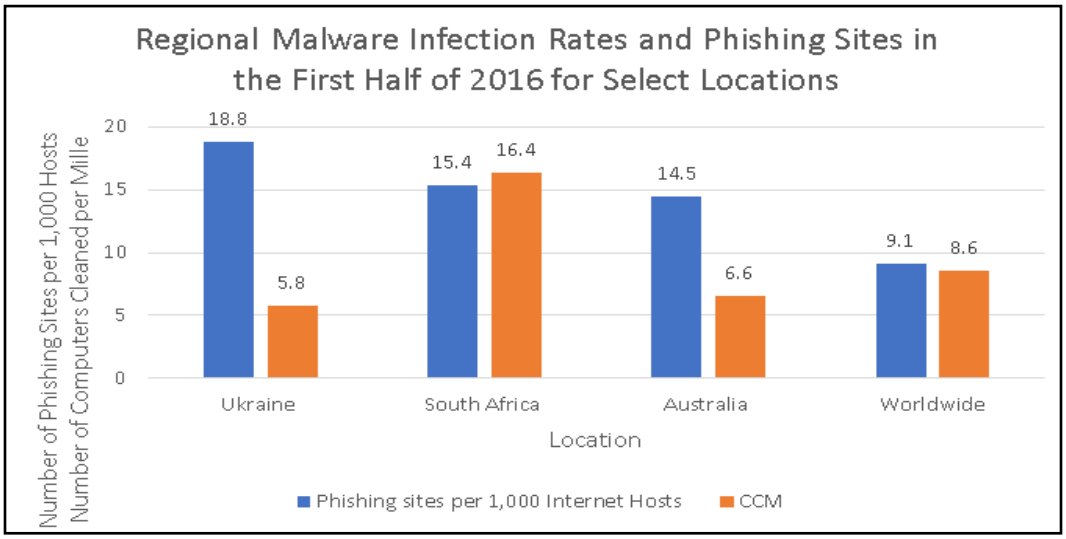


Figure 4.3: Comparing the malware infection rates of the locations with the highest number of phishing sites per 1,000 internet hosts (Microsoft Corporation, 2016)

More recent data for the fourth quarter of 2019, published by the Anti-Phishing Working Group, found that the Country Code Top-Level Domains (ccTLD) with the most phishing sites included Brazil, the UK, Russia, and India (Phishing Activity Trends Report 4th Quarter 2019, 2020). Interestingly, according to data published by Microsoft, the malware Encounter Rate in Brazil and Russia were only one or two percent above the worldwide average in the fourth quarter of 2019 and the UK was well below the average (Microsoft Corporation, 2020). However, we don't have malware infection rate data for this period, so it's harder to draw conclusions about the availability of the number of compromised systems in these locations to host phishing sites. Historically, Brazil and India have had relatively high malware infection rates, while Russia and the UK have not.

Clearly, more data is required to draw any real conclusions. But it doesn't appear that phishers rely on the availability of large numbers of compromised systems in order to set up relatively large numbers of phishing sites.

Regardless of where attackers host their phishing operations, organizations want to mitigate these attacks. Next, let's discuss some of the mitigations that enterprises can employ to manage phishing attacks.

Mitigating phishing

Phishing websites used to be easier for users to identify than they are today. If a webpage was asking you for credentials or confidential information, but was not protecting that data in transit using HTTPS (the lack of the legitimate lock icon in the web browser indicates this), then why would you type anything into that page? But this is no longer an effective way to identify phishing sites, as the Anti-Phishing Working Group found in their research:

> "*But by the end of 2019, 74% of all phishing sites were using TLS/SSL*"
>
> *– (Phishing Activity Trends Report 4th Quarter 2019, 2020)*

Mitigating phishing attacks is both easy and hard. For example, phishing attacks that seek to steal credentials can largely be mitigated by enforcing the requirement to use **multi-factor authentication** (**MFA**). According to studies conducted by Microsoft:

> *"Your password doesn't matter, but MFA does! Based on our studies, your account is more than 99.9% less likely to be compromised if you use MFA."*
>
> *– (Weinert, 2019)*

Requiring a second factor for authentication largely mitigates the risks associated with weak, leaked, and stolen passwords. If an attacker successfully tricks a user into disclosing their credentials in a phishing attack, but access to the account requires another factor, such as physical access to a token, landline, or mobile phone, then the credentials by themselves won't give attackers access to the account. Of course, that doesn't stop attackers from trying to use those stolen credentials on hundreds of online financial and ecommerce sites, betting on the chance that the user used the same credentials multiple times; their scripts do this within seconds of obtaining leaked and stolen credentials. Reusing the same password across accounts is still too common but can be largely mitigated by leveraging MFA everywhere.

But as I mentioned in an earlier chapter, MFA isn't available everywhere, especially in enterprise environments with decades of legacy applications. Even when MFA is available, a surprising low percentage of consumers and enterprises seem to embrace it. CISOs and security teams should be huge advocates of MFA everywhere because it can be so effective.

Also remember that at a minimum, senior executives should all use MFA everywhere and are the last people that should be exempt from MFA policies; after all, they are the primary targets of business email compromise and other social engineering attacks. Making executives lives easier by giving them exceptions for the very security policies and controls that mitigate attacks against them specifically isn't prudent, and is very literally a gift to attackers.

One effective tool I've seen used in cases where executives demand exceptions for security policies is risk acceptance letters. A risk acceptance letter or risk acknowledgement letter documents that the risks associated with the security policy exception have been explained to the executive, they understand these risks, and accept them on behalf of their entire organization.

Periodically, these risk acceptance letters should be reviewed by the CISO, senior executives, and potentially the Board of Directors, to ensure that systemic, long-term risk has not been inappropriately accepted. When confronted with one of these letters, executives who want security policy exceptions typically pause at the last minute once they have time to reflect on the potential consequences to their organizations and to their careers. In the end, many such executives prudently decide not to demand security policy exceptions.

Of course, phishing isn't limited to credential theft. Attackers use phishing in their attempts to trick people into disclosing information that they otherwise would not share. MFA doesn't mitigate these types of attacks. In these cases, the best mitigation is education. Training Information Workers to recognize potential phishing attacks and other social engineering tactics isn't foolproof but can be very effective. Some organizations simply refuse to approve phishing exercises designed to train their Information Workers to recognize phishing attacks. The management of these organizations do their employees a disservice with such decisions; after all, the only beneficiaries of such decisions are attackers, who prey on Information Workers that never get social engineering training.

One of the tools that CISOs have, who are faced with management teams that do not support this type of training, is risk management. In my experience, CISOs that quantify risk for their management teams have a better chance of success; it helps put their efforts into context, even when nothing bad happens. Remember that risk is the combination of probability and impact. The fact that most of the largest and highest profile data breaches in history started with a phishing email can help communicate the risk. So can the volume of phishing emails and the number of phishing sites that I provided in this chapter. The data tells us that a minimum of 100 million phishing emails are sent every day, and the total number is likely a multiple of this. Additionally, tens of thousands of new active phishing websites come online every week (at a minimum). Combine this with phishing data from your own organization to quantify the probability that Information Workers receive phishing emails and visit compromised websites, how many, and how often.

Then develop some quantitative impact estimates, ranging from no impact because phishing emails were filtered before they made it to Information Workers, to a successful compromise that involved data exfiltration and subsequent reputational damage and legal liability for the organization. Such figures can make the decision to train people to recognize social engineering attacks less abstract and easier to compare to the other risks that management teams already manage.

Also consider whether your organization's Information Workers really require unfettered access to the internet. Do they really need to visit websites located in the places that host the most phishing sites? Is there really a legitimate business need to allow everyone in an organization to go everywhere on the internet? The .com domain typically has more phishing sites than any other generic top-level domain – isn't this enough risk without enabling everyone in an organization to visit any site in the country code top-level domains that typically have two or three times the number of phishing sites than the worldwide average? Whitelisting sites with legitimate business purposes in these domains and blocking connections to other sites from corporate managed assets seems like it could reduce the chances of visiting a phishing site hosted in a country code top-level domain. Employing actively managed web filtering solutions can make this mitigation relatively easy.

Now let's look at the second tactic the attackers used in our example of a typical attack, a drive-by download attack.

Drive-by download attacks

While phishing attacks are at the intersection of social engineering and weak, leaked, and stolen passwords, drive-by download attacks are at the intersection of social engineering and unpatched vulnerabilities. Drive-by attacks are typically performed by attackers using social engineering to trick users into visiting a malicious website. They can do this several ways, including via email, online ads, putting links to malicious sites in the comments sections of webpages and social network posts, and many other tactics. Sometimes, attackers compromise a legitimate website and use it to host drive-by download attacks; the more popular the website, the better for the attackers as it increases their chances of successfully compromising as many systems as possible.

Getting potential victims to malicious websites under the control of attackers is the first step in the attack. The next step is to exploit unpatched vulnerabilities on the victims' systems. To do this, attackers will attempt to run scripts that have embedded URLs or they will use an **inline frame** (**IFrame**) to load another HTML document page unbeknownst to the user. Iframes have legitimate uses making it complicated to distinguish between good ones and malicious ones. Attackers will place IFrames the size of a pixel on their malicious webpages so that users cannot see them. When these HTML documents load, they can, among other things, run scripts that detect the victim's operating system and browser versions, select and download corresponding exploits for common vulnerabilities for these versions, and ultimately download and install other malware that gives attackers illicit control of the compromised system. Such malicious IFrames can be placed on webpages of legitimate websites that have been compromised. This means that visiting a trusted website with a system that is not fully patched, can result in a compromised system that attackers can control remotely, cripple with ransomware, and so on.

As illustrated in *Figure 4.4*, between July 2012 and January 2020, the highest number of drive-by download pages discovered on the internet was in 2013, where more than one drive-by download page was found per one thousand URLs indexed by Microsoft's Bing search engine. However, more recently, the worldwide average was 0.09 and 0.08 of these malicious sites per 1,000 URLs indexed in 2018 and 2019, respectively. That's a 173% difference in the number of drive-by download sites between 2013 and 2019. The data in *Figure 4.4* has been collated from Microsoft's Security Intelligence Report and the Interactive Microsoft Security Intelligence Reports (`https://www.microsoft.com/securityinsights/Driveby`).

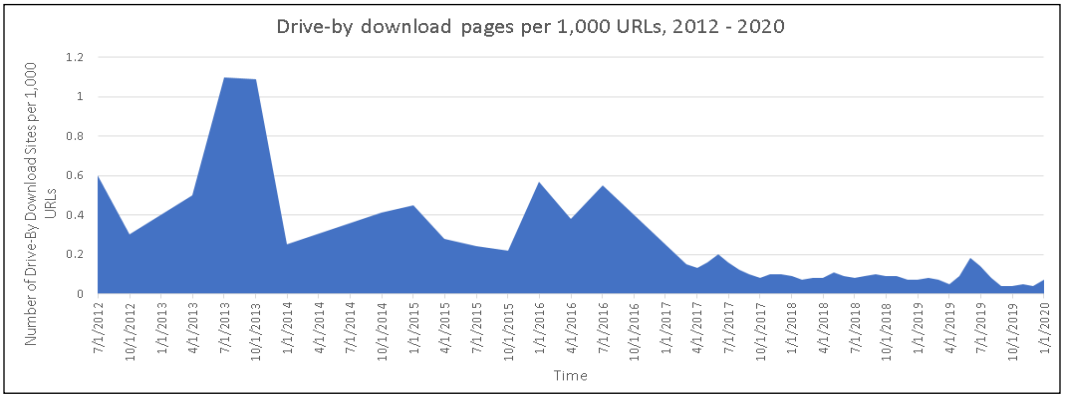


Figure 4.4: Drive-by download pages per 1,000 URLs indexed by Microsoft's Bing search engine between 2012–2020 as published in the Microsoft Security Intelligence Report Volumes 14–21 (Microsoft Corporation, 2012–2017) (Microsoft Corporation, 2020)

The components used in drive-by download attacks can be distributed, with several different remote systems hosting them. The scripts that run can be hosted on different "redirector" servers, the exploits used to exploit unpatched vulnerabilities can be hosted on separate exploit servers, and the malware that ultimately gets downloaded to the victims' systems can be hosted on separate malware hosting servers. Distributing components of drive-by download attacks this way provides several advantages to attackers. It allows attackers to be more agile, enabling them to adjust their attacks quickly. This helps them optimize their attacks and makes it harder to find and dismantle all the components attackers use.

Subsequently, the infrastructure used to host the components of drive-by download attacks are distributed all over the world. *Table 4.2* and *Table 4.3* provide the locations with the highest number of drive-by download URLs per 1,000 URLs indexed by Microsoft's Bing search engine in 2018 and 2019 respectively (Microsoft Corporation, 2020).

Concentrations of drive-by download pages are significantly higher than the worldwide average in these locations:

Location	Number of drive-by download URLs per 1,000 URLs
Myanmar	6.78
Samoa	2.64
Taiwan	2.54
El Salvador	1.97
Palestinian Territory	1.26
Bolivia	1.06
Afghanistan	1.05
Papua New Guinea	1.00
French Polynesia	0.86
Zimbabwe	0.71
Worldwide Average	0.09

Table 4.2: Locations with the highest number of drive-by download sites in 2018 (Microsoft Corporation, 2020)

Location	Number of drive-by download URLs per 1,000 URLs
Oman	687.3
French Polynesia	2.22
Gambia	1.89
Afghanistan	1.85
Saint Lucia	1.57
Myanmar	1.16
Libya	1.09
Colombia	0.48
Russia	0.45
El Salvador	0.43
Worldwide Average	0.08

Table 4.3: Locations with the highest number of drive-by download sites in 2019 (Microsoft Corporation, 2020)

The number of drive-by download pages per 1,000 URLs in Oman in 2019 isn't a typo. According to data published by Microsoft, there were 687.3 drive-by download URLs for every 1,000 URLs in Oman, averaged across the twelve months of 2019 (Microsoft Corporation, 2020). That's 8,591.25 times higher than the worldwide average.

In November of 2019, Microsoft reports that there were 1,251.94 drive-by download URLs for every 1,000 URLs found by Bing in Oman (Microsoft Corporation, 2020). That suggests a very high concentration of drive-by download URLs in this **country code Top Level Domain** (**ccTLD**) at the time.

Although this could be a simple error in the data, there could be another, less banal explanation. The ccTLD for Oman is .om. Attackers could be registering and using domain names in this ccTLD to catch web browser users that type `.om` instead of `.com`. This hypothesis seems plausible given how often people could make the trivial mistake of typing `google.om` instead of `google.com`, `apple.om` instead of `apple.com`, and so on. How many people would make mistakes like this every day? It seems like it could be enough to get the attention of attackers leveraging drive-by download sites. This is what some cybersecurity researchers were reporting back in 2016. Could this tactic still be in widespread use almost three years later in the last quarter of 2019?

> "*According to Endgame security researchers, the top level domain for Middle Eastern country Oman (.om) is being exploited by typosquatters who have registered more than 300 domain names with the .om suffix for U.S. companies and services such as Citibank, Dell, Macys and Gmail. Endgame made the discovery last week and reports that several groups are behind the typosquatter campaigns.*"
>
> *– (Spring, 2016)*

Mitigating drive-by download attacks

These attacks tend to rely on unpatched vulnerabilities to be successful. Attackers have exploit libraries that they leverage for their drive-by download attacks. Studies have shown that attackers have used between one and over twenty exploits on a single drive-by URL. If the underlying vulnerabilities that these exploits try to take advantage of are patched, these attacks won't be successful. Therefore, a well-run vulnerability management program will mitigate drive-by download attacks.

Additionally, preventing exposure to malicious websites like drive-by download sites can be helpful. Consider whether allowing Information Workers and systems administrators unfettered access to the internet is required and worth the risk. Why do they need access to the .om ccTLD for example, or any of the other ccTLD domains where there likely aren't legitimate business reasons to visit? Leveraging actively managed web filtering services can be helpful; blocking access to parts of the internet from corporate assets that don't have a clear business purpose can also be helpful.

Don't allow system administrators to visit the internet using web browsers from servers that process anything, or from systems that are important. Secure Access Workstations or Privileged Access Workstations should be used for server administration to limit risk to important systems. Browsing to sites on the public internet should be strictly forbidden on such systems and prevented with technical controls.

Running up-to-date anti-malware software from a trusted anti-malware vendor can also be an effective mitigation. Drive-by download attacks typically result in malware being downloaded to the victim's system. If anti-malware software detects the exploit attempt and blocks the download and installation of such malware, potential disaster is averted.

I mentioned that attackers typically distribute components of drive-by download attacks across separate infrastructure located in different places around the world. Let's now take a closer look at malware distribution sites, which can be used as part of drive-by download attacks or used to deliver malware employing other tactics to victims.

Malware hosting sites

We've seen that a great source of data for malicious websites, like phishing sites and drive-by download sites, are internet search engines and popular web browsers. These data sources can also give us a glimpse into malware hosting sites on the internet. I say a glimpse, because things can change very quickly as many attackers have become adept at covering their tracks and making it hard to find the infrastructures they use for their attacks. Remember, no one is omniscient.

We have a bunch of data snap shots that we can stitch together over time to provide us with a glimpse of the threat landscape. Frequently, the landscape changes before researchers can collect, analyze, understand, and act on such data.

This is where the promise of **Machine Learning** (**ML**) and **Artificial Intelligence** (**AI**) can and is helping – churning though massive amounts of complicated data sets much faster than humans can do this job manually. And, of course, attackers have been busy the past few years trying to find ways to defeat systems that leverage ML and AI (Microsoft Defender ATP Research Team, 2018).

But let's start by looking at some data that Google has published on malware hosting sites. They have a unique view of malware hosting sites as they operate the world's most popular internet search engine. Google publishes data on the malware hosting sites they find via their Safe Browsing service. They describe this as the following:

> "*Malware can hide in many places, and it can be hard even for experts to figure out if their website is infected. To find compromised sites, we scan the web and use virtual machines to analyze sites where we've found signals that indicate a site has been compromised.*"
>
> *– (Google, 2020)*

Google provides data on "attack sites" from January 2007 up until the present. From this data, it appears the most attack sites hosting malware that they found was in November 2012. The week of November 11, 2012, Google's Safe Browsing service identified 6,192 attack sites on the internet (Google, 2020). Another notable peak was the week starting September 15, 2013, when 5,282 attack sites were identified (Google, 2020). These relatively huge numbers have dwindled in more recent times. Between 2018 and 2019, the highest number of attack sites identified by Safe Browsing was 379, and between January and April 2020, 30 attack sites appears to be the maximum identified in any single week (Google, 2020). Like drive-by download sites, the number of malware hosting sites appears to have dwindled over time.

Google also provides insights into where these malware hosting sites are located. In the quarter ending April 8, 2020, the locations identified with the most malware hosting sites included China, Poland and Hungary.

All with 2% of the sites scanned hosting malware (Google, 2020). Locations with 1% of sites scanned found to be hosting malware included Australia, Indonesia, Thailand, Vietnam, India, South Africa, Turkey, Bulgaria, Romania, Ukraine, Czech Republic, Slovakia, Austria, Italy, France, Portugal, Sweden, Brazil, and Argentina (Google, 2020). Other locations such as the United States, United Kingdom, Canada, Russia, and others all had less than 1% during this time (Google, 2020).

The data on malware hosting sites that Microsoft has published in numerous volumes of the Microsoft Security Intelligence Report provides a different snapshot from a different perspective. Microsoft's data includes data from the SmartScreen Filter in various versions of the web browsers they offer. According to this data, there was a significant increase in the number of malware hosting sites found in the first half of 2016 as illustrated in *Figure 4.5*. These are the most up-to-date figures available:

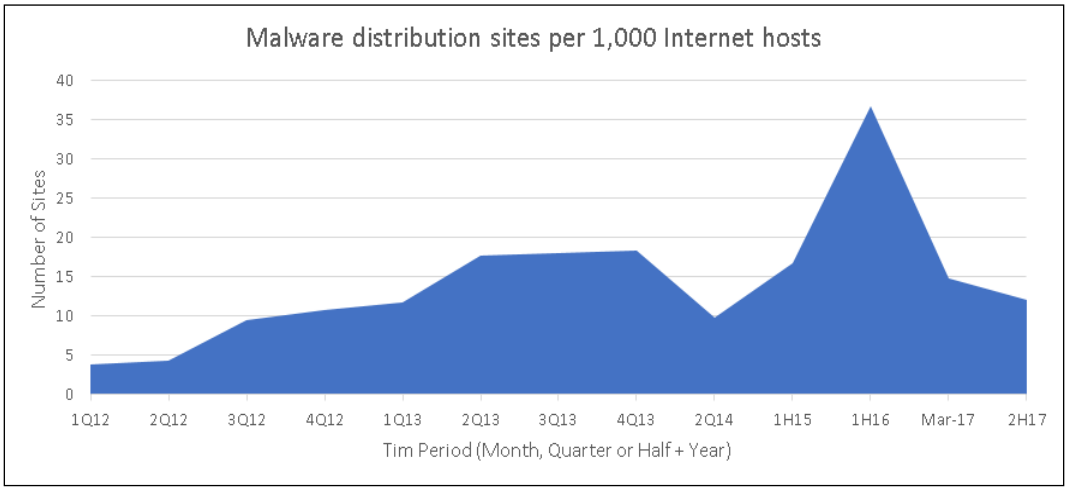


Figure 4.5: Malware distribution sites per 1,000 internet hosts as reported in the Microsoft Security Intelligence Report Volumes 13–23 (Microsoft Corporation, 2012–2017)

Another source of data on malware distribution sites is URLhaus (`https://urlhaus.abuse.ch/statistics/`). URLhaus collects URLs for malware hosting sites and shares them with Google Safe Browsing, among others. Their purpose, according to their website, is as follows:

> *"URLhaus is a project operated by abuse.ch. The purpose of the project is to collect, track and share malware URLs, helping network administrators and security analysts to protect their network and customers from cyber threats."*
>
> *– (URLhaus, 2020)*

According to data published by URLhaus, between April 10, 2020 and May 7, 2020, there were hundreds, sometimes thousands, of unique malware hosting URLs submitted every day (URLhaus, 2020). Hosting networks in the United States and China appear most often in their lists of top malware hosting networks (URLhaus, 2020).

One conclusion we can draw from the data is that malware hosting sites are more common than phishing sites. For example, according to the data published by Microsoft, on average, there were between 5.0 and 9.1 phishing sites for every 1,000 internet hosts between 2015 and 2017 as illustrated in *Table 4.1*; the average number of malware hosting sites per 1,000 internet hosts between 2015 and 2017 is 20.1, as illustrated by the data in *Figure 4.5*.

Subsequently, a cybersecurity strategy that focuses on mitigating phishing attacks, but does not include drive-by download attacks and malware distribution sites, could be missing mitigations for a higher probability threat. Let's now consider some of the ways that malware distribution can be mitigated.

Mitigating malware distribution

Legitimate websites that are compromised and then used to distribute malware can lead to many poor outcomes for consumers and organizations alike. For this reason, it is important that organizations that operate websites understand and focus on the cybersecurity fundamentals. Recall from *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*, that cybersecurity fundamentals are the part of a cybersecurity strategy that focuses on mitigating the cybersecurity usual suspects. The cybersecurity usual suspects include unpatched vulnerabilities, security misconfigurations, weak, leaked and stolen credentials, insider threats, and social engineering. Managing the cybersecurity fundamentals are critical to prevent websites from becoming malware distribution sites. Everyone setting up a website on the internet must accept this responsibility.

The vendors and organizations that scour the internet looking for malware distribution sites will typically contact the webmasters of sites that they find distributing malware. According to data that Google published on their notification activities, the average response time of the webmasters they notified the week of January 12, 2020, was 20 days; this is the lowest average response time since the week of July 20, 2014 (Google, 2020). The data suggests typical average response times of 60 or 90 days (Google, 2020).

Given this data, the call to action is clear. If your organization operates websites on the internet, it's your organization's responsibility to pay attention to abuse reports. Reviewing abuse reports for corporate assets isn't something that IT staff should do in their spare time; it should be part of every enterprise's governance processes.

The table stakes for operating a website on the internet are actively managing the cybersecurity fundamentals and monitoring and acting on abuse reports in a responsible period of time. If an organization isn't willing to do these things, it should do everyone a favor and shut its website down.

Running current anti-malware solutions, from a trusted vendor, on internet connected systems can also be an effective mitigation. But remember that attackers will often seek to subvert anti-malware solutions once they successfully initially compromise a system. The anti-malware vendors know this and make it harder for attackers to do this. But once an attacker has System or Administrator access on a system, they own that system, making it much harder to prevent the compromise of system security defenses. For this reason, I like performing periodic offline anti-virus scans. For example, Microsoft offers Windows Defender Offline, which will scan the system without using the active operating system's kernel. Windows Defender Offline is baked into Windows 10 and is available for older versions of Windows via a download that can be run from a DVD or USB drive (Microsoft Corporation, 2020).

Of course, for organizations using the cloud, they can simply shut down systems every couple of hours and automatically rebuild them. Short-lived systems like this provide very little time for attackers to make use of compromised systems.

However, even in short-lived environments, a well-run vulnerability management program and anti-malware solutions can be useful. I'll discuss this further in *Chapter 8*, *The Cloud – A Modern Approach to Security and Compliance*.

But now, let's look at the final stage of the typical attack pattern we started this chapter with, some of the typical post compromise activities.

Post compromise – botnets and DDoS attacks

Once systems have been initially compromised via one of the cybersecurity usual suspects, like unpatched vulnerabilities and/or social engineering as we discussed in this chapter, any information of value is siphoned from victims' systems to be sold or traded. At this point, attackers have full control of the systems they have compromised. Many times, victims' systems are enlisted into botnets and used to perform whatever illicit projects their operators desire, including DDoS attacks.

There's a lot that can be written about botnets, how they operate, and the projects they are typically employed on. In fact, entire books have been dedicated to botnets. I won't try to duplicate those here. But I do want to briefly mention a few things on this topic.

It goes without saying that botnets have garnered a lot of attention over the years. When I worked at Microsoft, the **Microsoft Digital Crimes Unit** (**DCU**) worked with law enforcement and industry experts to disrupt some of the largest botnets in operation. This work helped to dramatically reduce spam on the internet and degrade the attack power these botnets provided to their operators. Some of these botnets were composed of hundreds of thousands or millions of compromised systems and were capable of sending tens of billions of spam and phishing email messages per day. Rustock and Waledac are two examples of such botnets. To do this, the DCU had to approach the problem as a novel legal challenge in which they sought and were given legal control over the domains and physical infrastructure these botnets used (Jones, 2011).

Attackers will drain anything of value from the systems they have complete control over, including cached credentials. Massive lists of leaked and stolen credentials have been found on the internet over the years (Solomon, 2017). If the compromised systems or accounts have authenticated and authorized access to other systems in the environment, attackers will potentially have access and control over them as well, exacerbating the damage to the organization.

Accelerating detection and recovery activities can reduce the amount of time attackers control these assets, thus potentially reducing the damage they do to other victims and potentially reducing the costs associated with recovery and the restoration of normal operations. Threat intelligence can help organizations identify systems communicating with known botnet command and control infrastructures. Attackers know this and have been hosting some of their infrastructures in public web hosting and cloud environments in an effort to hide their operations among legitimate network traffic.

One of the illicit purposes that botnets have been used for over the years has been **Distributed Denial of Service** (**DDoS**) attacks. Modern DDoS attacks use sophisticated techniques to overwhelm their targets with network traffic, thus depriving legitimate use of the services hosted by the victim.

How large can DDoS attacks get? The largest documented attack so far occurred in February 2018, when attackers launched an attack on GitHub. This DDoS attack is said to have peaked at 1.35 terabytes per second (TBps), which is the equivalent of more than 126 million packets per second (Kottler, 2018). This attack used a novel approach, by abusing memcached instances that were not secured. This approach enabled attackers to amplify their attack by a factor of 51,000; put another way, for every 1 byte of network traffic that attackers sent, up to 51,000 bytes (51 KB) were sent to their target. This allowed attackers to overwhelm GitHub's network capacity with a massive amount of UDP traffic that interrupted network connectivity to the site for almost 10 minutes.

Perhaps a less sophisticated, but a more interesting DDoS attack from the history books was the attack on critical infrastructure in Estonia in 2007. Some attributed this attack to Russia (Anderson, 2007). The reason this is interesting is that perhaps it gave us a preview of what to expect in future cyberwar conflicts. Simultaneous kinetic and online attacks that overwhelm the ability to wage warfare physically and logically. But that's the topic of an entire other book!

Of course, not all DDoS attacks need to be that large or innovative to be effective. But organizations have options to help them mitigate such attacks. There are many vendors that offer DDoS protection services, some of which include AWS Shield, Amazon CloudFront, Google Cloud Armor, Microsoft Azure DDoS Protection, Cloudflare, Akamai, and many others. In addition to protection services, the cloud offers techniques that can be used to scale infrastructure automatically as needed during DDoS attacks (Amazon Web Services, December 2019).

To summarize, the key is to focus on the cybersecurity fundamentals so your systems do not end up being part of a botnet and used to attack countless other organizations and consumers. As we will discuss in *Chapter 5*, *Cybersecurity Strategies*, investing in detection and response capabilities will help organizations minimize the damage and costs associated with botnets and the grief they bring with them to the internet.

Chapter summary

This chapter focused on internet-based threats. We examined phishing attacks, drive-by download attacks, and malware distribution sites. So many attacks leverage social engineering that CISOs and security teams must spend time and resources to mitigate it. For example, every week, tens of thousands of new phishing sites are connected to the internet, and every month, billions of phishing emails are sent to prospective victims.

Locations that have historically hosted above average concentrations of phishing sites include Bulgaria, Ukraine, and Indonesia. Most phishing emails include a link to a phishing site (Microsoft Corporation, 2018) and most phishing sites leverage HTTPS (SSL/TLS) (Phishing Activity Trends Report 4th Quarter 2019, 2020). Accounts are nearly 100% less likely to be compromised when MFA is enabled (Weinert, 2019). Anti-social engineering training for Information Workers can also be an effective mitigation.

Drive-by download attacks leverage unpatched vulnerabilities to install malware unbeknownst to the user. The number of drive-by URLs has been dramatically reduced from the peak in 2013 to the current low levels in 2019-2020. According to data released by Microsoft, Oman's ccTLD hosted 8,591 times more drive-by download sites than the worldwide average in 2019 (Microsoft Corporation, 2020). This could indicate that attackers are using the `.om` domain to attack users that mistype URLs in the `.com` domain. A well-run vulnerability management program and running up-to-date anti-malware from a trusted vendor can be effective mitigations for drive - by downloads.

Malware hosting sites are more common on the internet than phishing sites. Subsequently, a cybersecurity strategy that focuses on mitigating phishing attacks, but does not include drive-by download attacks and malware distribution sites, could be missing mitigations for a higher probability threat.

Systems compromised by phishing attacks, drive-by downloads, and other malicious websites can end up being enlisted into botnets and used to attack other organizations and consumers, including participating in DDoS attacks.

That wraps up our look at internet-based threats. Next, in *Chapter 5*, *Cybersecurity Strategies*, we'll examine cybersecurity strategies that organizations can employ to mitigate these threats.

References

1. Amazon Web Services (December, 2019). *AWS Best Practices for DDoS Resiliency*. Retrieved from Amazon Web Services: https://d1.awsstatic.com/whitepapers/Security/DDoS_White_Paper.pdf
2. Anderson, N. (May 14, 2007). *Massive DDoS attacks target Estonia; Russia accused*. Retrieved from Ars Technica: https://arstechnica.com/information-technology/2007/05/massive-ddos-attacks-target-estonia-russia-accused/
3. Google (May, 2020). *Google Safe Browsing*. Retrieved from Google Transparency Report: https://transparencyreport.google.com/safe-browsing/overview?unsafe=dataset:0;series:malware,phishing;start:1148194800000;end:1587279600000&lu=unsafe
4. Google (May, 2020). *Safe Browsing site status*. Retrieved from Google Transparency Report: https://transparencyreport.google.com/safe-browsing/search?hl=en
5. Google (May 9, 2020). *Where malware originates*. Retrieved from Google Transparency Report: https://transparencyreport.google.com/archive/safe-browsing/malware?autonomous_scan_history=systemId:18779;dataset:0&lu=global_malware&global_malware=time:q
6. Jones, J. (March 17, 2011). *Microsoft Takedown of Rustock Botnet*. Retrieved from Microsoft: https://www.microsoft.com/security/blog/2011/03/17/microsoft-takedown-of-rustock-botnet/
7. Kottler, S. (March 1, 2018). *February 28th DDoS Incident Report*. Retrieved from The GitHub Blog: https://github.blog/2018-03-01-ddos-incident-report/
8. Microsoft Corporation (2012-2017). *Microsoft Security Intelligence Report*. Redmond: Microsoft Corporation.
9. Microsoft Corporation (2012-2017). *Microsoft Security Intelligence Report Volumes 14–21*. Redmond: Microsoft Corporation.

10. Microsoft Corporation (2015-2017). *Microsoft Security Intelligence Report Volumes 19–23*. Redmond: Microsoft Corporation. Retrieved from: www.microsoft.com/sir
11. Microsoft Corporation (2016). *Microsoft Security Intelligence Report Volume 21*. Redmond: Microsoft Corporation. Retrieved from: https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RE2GQwi
12. Microsoft Corporation (2018). *Microsoft Security Intelligence Report Volume 23*. Redmond: Microsoft Corporation. Retrieved from: https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RWt530
13. Microsoft Corporation (2019). *Microsoft Security Intelligence Report Volume 24*. Redmond: Microsoft Corporation. Retrieved from: https://info.microsoft.com/ww-landing-M365-SIR-v24-Report-eBook.html?lcid=en-us
14. Microsoft Corporation (May 8, 2020). *Drive-by download pages*. Retrieved from Microsoft: https://www.microsoft.com/securityinsights/Driveby
15. Microsoft Corporation (May 15, 2020). *Help protect my PC with Microsoft Defender Offline*. Retrieved from Microsoft: https://support.microsoft.com/en-us/help/17466/windows-defender-offline-help-protect-my-pc
16. Microsoft Corporation (May 8, 2020). *Malware encounter rates*. Retrieved from Microsoft: https://www.microsoft.com/securityinsights/Malware
17. Microsoft Defender ATP Research Team (August 9, 2018). *Protecting the protector: Hardening machine learning defenses against adversarial attacks*. Retrieved from Microsoft: https://www.microsoft.com/security/blog/2018/08/09/protecting-the-protector-hardening-machine-learning-defenses-against-adversarial-attacks/
18. Pegoraro, R. (August 9, 2019). *We keep falling for phishing emails, and Google just revealed why*. Retrieved from Fast Company: https://www.fastcompany.com/90387855/we-keep-falling-for-phishing-emails-and-google-just-revealed-why

19. (2020) *Phishing Activity Trends Report 4th Quarter 2019*. Anti-Phishing Working Group. Retrieved from `https://docs.apwg.org/reports/apwg_trends_report_q4_2019.pdf`
20. Solomon, H. (December 11, 2017). *Searchable database of 1.4 billion stolen credentials found on dark web*. Retrieved from IT World Canada: `https://www.itworldcanada.com/article/searchable-database-of-1-4-billion-stolen-credentials-found-on-dark-web/399810`
21. Spring, T. (March 14, 2016). *Typosquatters Target Mac Users With New '.om' Domain Scam*. Retrieved from Threatpost: `https://threatpost.com/typosquatters-target-apple-mac-users-with-new-om-domain-scam/116768/`
22. URLhaus (May 9, 2020). *About*. Retrieved from URLhaus: `https://urlhaus.abuse.ch/about/`
23. URLhaus (May 9, 2020). *Statistics*. Retrieved from URLhaus: `https://urlhaus.abuse.ch/statistics/`
24. Weinert, A. (July 9, 2019). *Your Pa$$word doesn't matter*. Retrieved from Microsoft Corporation: `https://techcommunity.microsoft.com/t5/azure-active-directory-identity/your-pa-word-doesn-t-matter/ba-p/731984#`

5

Cybersecurity Strategies

Every enterprise should have a cybersecurity strategy and the CISO of each of these organizations should be able to articulate it. Whether your organization has a strategy or not, I hope this chapter provokes some thought and provides some tools that are helpful.

In this chapter, we'll cover the following topics:

- A sampling of cybersecurity strategies that have been employed over the past two decades, including:
    - The Protect and Recover Strategy
    - The Endpoint Protection Strategy
    - The Physical Control and Security Clearances Strategy
    - Compliance as a Security Strategy
    - The Application-Centric Strategy
    - The Identity-Centric Strategy
    - The Data-Centric Strategy
    - Attack-Centric Strategies
- A look at DevOps
- A brief introduction to Zero Trust

Let's begin by discussing which strategy is the right strategy for your organization.

Introduction

In *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*, I discussed the ingredients for a successful cybersecurity strategy. These include what I consider to be a critical ingredient for understanding the cybersecurity usual suspects, that is, the five ways that organizations get initially compromised. I have spent the last three chapters discussing the most common threats that CISOs and security teams are typically concerned about, including vulnerabilities, exploits, malware, and government access to data. In this chapter, I will combine all these concepts into an examination of some of the cybersecurity strategies that I have seen employed in the industry over the past couple of decades. You have probably seen some of these before and perhaps have used some of them. My objective for this chapter isn't to show you a bunch of strategies so that you can select one to use. My objective is to provide you with a framework for determining the efficacy of cybersecurity strategies, including strategies that I won't discuss in this chapter, but you might encounter in your career. In other words, I hope to teach you how to fish instead of giving you a one-size-fits-all strategy that I know will only help a fraction of organizations that use it.

The right strategy for your organization is the one that helps mitigate the most important risks to your organization. Risk is relative; therefore no one strategy can be a silver bullet for all organizations. I'll resist the temptation to simply tell you to use the NIST Cybersecurity Framework (NIST, n.d.), ISO/IEC 27001 (ISO, n.d.), or any of the other great frameworks that are available. Your organization has likely already embraced one or more of these frameworks, which is unavoidable for enterprise-scale organizations from a **Governance, Risk, and Compliance** (**GRC**) perspective; that is, your organization has to prove it's doing what the rest of the industry is doing, or it will be seen as an outlier. GRC frameworks are typically designed to help insulate organizations from liability after an incident, and subsequently many organizations prioritize them. However, the pace of data breaches hasn't slowed down, despite the number of great frameworks available. For example, the **European Data Protection Board** (**EDPB**) published a report on the results of the **General Data Protection Regulation** (**GDPR**) after the first nine months that GDPR was enforceable. Almost 65,000 data breach notifications were filed with the EDPB in those first nine months.

The vast majority of these organizations were likely fully compliant with their own security policies, thus illustrating the difference between cybersecurity and compliance. This is likely the tip of the iceberg, but it gives us some indication that organizations, both large and small, need help with cybersecurity strategy.

> *"The total number of cases reported by SAs from 31 EEA countries is 206.326. Three different types of the cases can be distinguished, namely cases based on complaints, cases based on data breach notifications and other types of cases. The majority of the cases are related to complaints, notably 94.622 while 64.684 were initiated on the basis of data breach notification by the controller." (European Data Protection Board, 2019)*

In this chapter, I'll give you a slightly contrarian view that is meant to be food for thought. If your organization already has a cybersecurity strategy and it uses industry frameworks, this chapter will give you some questions to ask yourself about the effectiveness of your current strategy. If your organization doesn't have a cybersecurity strategy that you can articulate, this chapter will give you some ideas about some of the strategies that other organizations have used, their advantages, disadvantages, and a way to measure their potential effectiveness.

As you saw in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*, where I described what a cybersecurity strategy is, I'm purposely simplifying the descriptions of these strategies. I have talked to some CISOs that had incredibly dense cybersecurity strategies that few people in their organization could fully comprehend or repeat. Unfortunately, most of these organizations have had data breaches. Ultimately, in every one of these cases, the initial compromise was due to a lack of focus on, or a mistake managing, the cybersecurity fundamentals. Keeping the strategy simple makes it easier for the stakeholder community and the people doing the work to understand the strategy and explain it to their teams (repeat it). It's likely that there are only a few teams within IT and the cybersecurity group that are responsible for understanding and executing the full strategy. You can reserve the super complicated version of the strategy, with overlays for governance, risk, compliance, product development, recruiting, supporting local cybersecurity educational programs, succession planning, and other components, for stakeholders that need and appreciate all that detail and ambition.

Regardless of how sophisticated a cybersecurity strategy is, its success relies on the ingredients I described in *Chapter 1* and crucially, how well it addresses the cybersecurity fundamentals. Measuring how that strategy performs over time is important, so that adjustments can be made to improve it. Let's look at measuring efficacy next.

Measuring the efficacy of cybersecurity strategies

Let me reacquaint you with two concepts that I introduced in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*. We are going to use these two concepts to measure the potential efficacy of the strategies that we examine.

Remember that the five ways that organizations get initially compromised, the cybersecurity usual suspects, include:

1. Unpatched vulnerabilities
2. Security misconfigurations
3. Weak, leaked, or stolen credentials
4. Social engineering
5. Insider threat

These are the five ways that organizations get initially compromised. Once an IT environment has been initially compromised there are many, many **tactics, techniques, and procedures** (**TTPs**) that attackers can use to move laterally, steal credentials, compromise infrastructure, remain persistent, steal information, destroy data and infrastructure, and so on. Most of these TTPs have been around for years. Occasionally, the industry will see attackers employing novel approaches. Mitigating the cybersecurity usual suspects is what I call the cybersecurity fundamentals. Organizations that focus on getting really good at the cybersecurity fundamentals make it much harder for attackers to be successful. Focusing on the things that all attackers do to initially compromise networks, the cybersecurity usual suspects, is a hard requirement for any strategy or combination of strategies that organizations employ.

Put another way, if an organization's cybersecurity strategy doesn't include being excellent at the cybersecurity fundamentals, it is setting itself up for failure. Why? We know that 99.9% of successful compromises start with the cybersecurity usual suspects. If that's true, why would your organization use a strategy that doesn't at least mitigate these attack vectors? Why would you use a strategy that you know has gaps in it, that attackers have used for decades, to attack other organizations? Remember the submarine analogy that I used in the preface section of this book. Why would you set sail in a submarine that you know has flaws in its hull? Would you be confident enough to dive two miles under the surface of the ocean in that submarine, and allow immense pressure to build on every square millimeter of that hull? That sounds foolhardy, right? There will be some of you that will be willing to take some risks so that you can compete in fast-moving, competitive industries.

This is where some executives I've met feel like they must make a choice between cybersecurity and moving fast. But moving fast and dynamic changes are not mutually exclusive; this isn't a choice that they have to make as they can get both cybersecurity efficacy AND business speed, agility, and scalability if they have a strategy that enables them to do so. Investing in approaches that willfully fail to address the most common ways organizations get compromised is a fool's errand. Additionally, if a network is already compromised, the organization still needs to focus on the cybersecurity fundamentals in order to prevent even more attackers from getting a foothold in their network, thereby preventing the attackers already inhabiting the network from getting back into it, if they can ever be driven from it. Whatever strategy an organization employs, it needs to incorporate the cybersecurity fundamentals.

Once an organization hones its ability to do the cybersecurity fundamentals and establishes a foundation that it can build on, then it makes sense to invest in advanced cybersecurity capabilities – capabilities focused on things other than the cybersecurity fundamentals. Your strategy needs a solid foundation, even if it has advanced cybersecurity capabilities, because the platforms these capabilities rely on for information and their own security can be undermined by unpatched vulnerabilities, security misconfigurations, social engineering, insider threats, and weak, leaked, and stolen passwords.

Being excellent at addressing all the cybersecurity fundamentals in both Production and Development/Test environments is a requirement for successfully deploying and operating advanced cybersecurity capabilities in your IT environment. For example, if an organization doesn't have a plan to scan and patch security vulnerabilities and misconfigurations in the hardware and software they deploy as part of their advanced cybersecurity capabilities, they shouldn't bother deploying them because, over time, they will just increase the organization's attack surface.

You might be wondering why you must invest in advanced cybersecurity capabilities at all if your organization is really good at the cybersecurity fundamentals. Because you have to plan for failure. You have to assume that the organization will be breached – it's not a matter of if, only a matter of when and how often it will happen. This "assume breach" philosophy is important for at least two reasons. First, history has taught us that planning to achieve 100% perfect protection for large on-premises IT environments for a sustained period of time is a wildly optimistic ambition. People in your organization and in your supply chain will make mistakes, and some of these will be security-related.

For example, your applications, whether development is done in-house or through vendors, will have bugs in them. Some of these bugs will be security vulnerabilities. Some of these vulnerabilities will be exploitable. You need to plan for this eventuality. You need to plan for the mistakes that administrators make that lead to security misconfigurations. You need to plan for the scenario where the trusted vendors in your supply get compromised or turn malevolent. This is an area where Red Teams can ground strategy in reality as they specialize in taking advantage of unrealistic assumptions.

The second reason organizations need to adopt an assume breach philosophy is that it gives their security teams permission to think about some key questions that security teams who believe they can achieve 100% effective protection, forever, never ask themselves.

For example, how would they know when they have been compromised? What will they do when they get compromised? These are questions that many security teams never ask themselves because they will not, or cannot, adopt an assume breach philosophy.

Some corporate cultures will not tolerate failure, so the idea that they plan for failure makes no sense to them; it's like admitting that they aren't good enough to do their jobs. In other organizations, senior executives will not support a plan for failure. I've met many executives that do not understand that they are in a submarine under immense pressure, surrounded by badness. Some of the executives I've talked to believe they are in a winnable battle. They believe that if they are smart enough, hire the right people, and buy the right protection capabilities, they will win the battle. But cybersecurity is a journey, not a destination. It doesn't have a beginning and an ending the way a battle does. It's constant, like pressure on the hull of a submarine. Planning for failure is the antithesis of their world view, so they refuse to support CISOs that know they need to embrace a more modern approach to cybersecurity. This is one reason why I've always tried to spend time with cybersecurity strategy stakeholders, other than the CISO and the security team. Very often, the security team understands everything I've written here, but one or two executives or board members have uninformed views.

Advanced cybersecurity capabilities are the part of your strategy that will help you identify, protect, detect, respond and recover (NIST, n.d.). This is the part of your strategy that helps augment and identify shortcomings in the cybersecurity fundamentals. You need them both for the strategy to be successful. The **High Value Assets** (**HVAs**) component of the strategy acknowledges the importance of HVAs. As I mentioned in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*, if the confidentiality, integrity, or availability of an HVA is compromised, this typically means the organization itself will fail. The sustained compromise of an HVA could be an extinction event for a company (Ashford, 2016) and drive public sector organizations back to using pencils, paper, and the processes they used before they invested in IT.

Planning and investing in security specifically focused on HVAs, in addition to the cybersecurity fundamentals and advanced cybersecurity capabilities, will help organizations manage the risk to their most important assets.

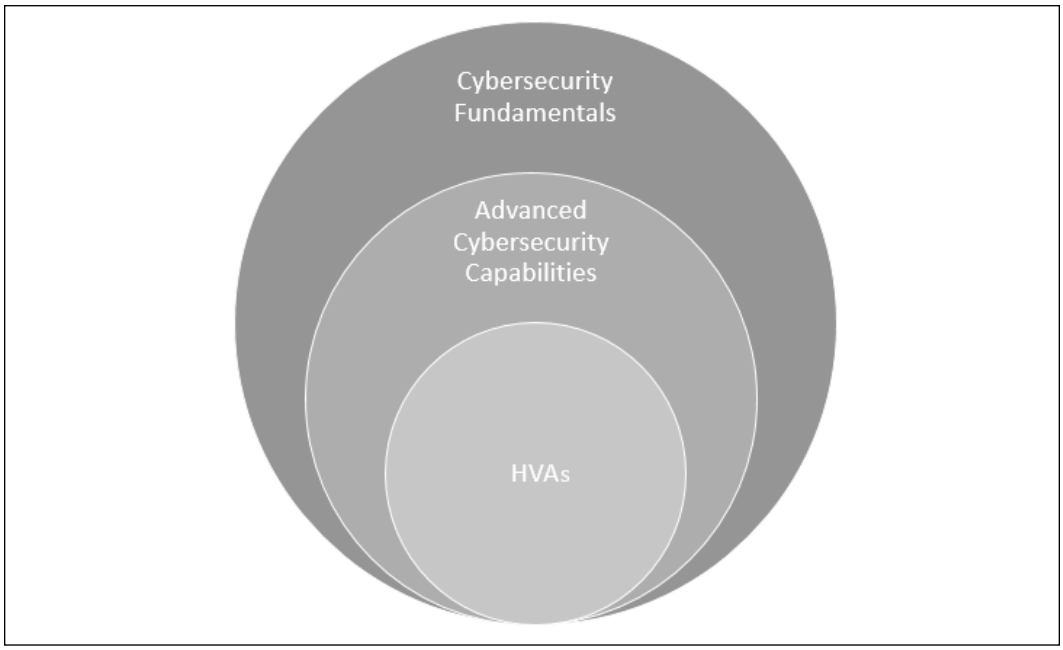


Figure 5.1: An illustration of a cybersecurity strategy

Regardless of organizations' HVAs and which advanced cybersecurity capacities they decide to invest in (which is highly variable between organizations), the entire strategy model I've outlined here relies on the foundation that the cybersecurity fundamentals provide. Without a solid foundation provided by focusing on the cybersecurity fundamentals, a strategy will fail over time. Any cybersecurity strategy that an enterprise pursues needs to focus on the cybersecurity fundamentals at a minimum. Given this, I'm going to introduce a simple method to help determine a strategy's potential efficacy, by estimating how well it incorporates the cybersecurity fundamentals and mitigates the cybersecurity usual suspects.

I will estimate the potential efficacy of the cybersecurity strategies we examine by using a simple scoring system. I call this system the **Cybersecurity Fundamentals Scoring System** (**CFSS**). This system assigns a score between 0 and 10 for each of the cybersecurity usual suspects, based on how well the strategy mitigates the risk. Higher scores mean that the strategy is more effective at mitigating each particular cybersecurity usual suspect. For example, a score of 20 means the strategy fully mitigates the risk associated with a specific cybersecurity usual suspect. A low score, such as a score of 1 for example, means that the strategy's ability to mitigate the risk is relatively low. The CFSS includes a separate score for each of the five cybersecurity usual suspects as shown in *Table 5.1*:

Cybersecurity Usual Suspect	Score
Unpatched vulnerabilities	0 – 20
Security misconfigurations	0 – 20
Weak, leaked, stolen credentials	0 – 20
Social engineering	0 – 20
Insider threat	0 – 20

Table 5.1: CFSS summary

The total of all five of the scores is the CFSS total score for the strategy. The lowest possible CFSS total score for a strategy is zero, while the highest is 100. For example, as shown in *Table 5.2*, let's say we have a strategy called "XYZ" and we estimate scores for the five measures in the CFSS. When we add up the individual scores, we get a CFSS total score of 23 out of a possible 100 points:

XYZ Strategy Example	Score
Unpatched vulnerabilities	10
Security misconfigurations	10
Weak, leaked, stolen credentials	2
Social engineering	0
Insider threat	1

Table 5.2: An example of the CFSS

The goal is to find a strategy that gives us a perfect 100 score, although this is likely more aspirational than probable. But this type of scoring system gives us a way to estimate a strategy's ability to mitigate all five ways organizations get initially compromised, as well as a way to compare strategies across the cybersecurity fundamentals. Potentially, this approach can help us identify a combination of strategies that gives us a perfect score, or a high score, across the cybersecurity usual suspects if no single strategy does this. Finally, it can also help us determine where the gaps are in a strategy that's currently in use by an organization. If you know where the weaknesses or gaps are, then you can develop a plan to address these inadequacies.

Before we start measuring strategies using this framework, I want to point out a hidden risk using this type of rating. Like most risk-based approaches, it is based on the assumption that CISOs and security teams will be able to accurately estimate the level of risk and identify effective mitigations. In my experience, I have seen some CISOs overestimate their capabilities and their ability to effectively mitigate risks, all while simultaneously underestimating the risks themselves and the effectiveness of the cybersecurity fundamentals.

Now that we have a cybersecurity strategy concept in mind and a scoring system to help us determine the relative efficacy of different approaches, let's examine numerous cybersecurity strategies in more detail.

Cybersecurity strategies

As I mentioned in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*, some of the cybersecurity professionals I have met with have a negative reaction when the term "strategy" is used in a cybersecurity context. This is a word that can be used in at least a few different ways. Security and compliance professionals sometimes use the term "strategy" when they are referring to frameworks, models, or standards. I explained what I mean when I use this term, in detail, in *Chapter 1*. If you haven't read that chapter already, I recommend that you read it because it provides a bunch of context that I won't repeat here. You'll see me use the terms framework, approach, model, and so on, interchangeably throughout all the chapters. Please feel free to associate whatever term makes the most sense to you when I use any of these terms.

The following list contains many of the strategies that I have seen in use over the last two decades in the industry. I'm going to examine each of these strategies in detail and provide an estimated CFSS score for each one. The CFSS scores that I provide are my own subjective opinion and are subject to my own assumptions and biases. I provide you with some context on why each cybersecurity usual suspect was scored the way it was, so that you can understand my approach and agree or disagree with it.

I invite you to think through your own CFSS score estimate for each of these strategies:

- Protect and Recover Strategy
- Endpoint Protection Strategy
- Physical Control and Security Clearances as a Strategy
- Compliance as a Security Strategy
- Application-Centric Strategy
- Identity-Centric Strategy
- Data-Centric Strategy
- Attack-Centric Strategy

As we review these strategies, even if your organization doesn't use any of them, please ask yourself if you know if the vendors that are part of your supply chain use any of them. If you don't know about the strategies they are using to manage the risk to their organizations and to their customers, then you might want to ask them how they are mitigating the cybersecurity usual suspects. This is the minimum they should be doing for themselves and for their customers.

There are two approaches that I think are more about the intersection of culture, philosophy, process, implementation, and supporting technologies than strategy, per se. These are DevSecOps and Zero Trust. These aren't cybersecurity strategies in the same classical sense that the others on my list are. An organization might still use one or more of the strategies on my list, in addition to DevSecOps and/or Zero Trust. For this reason, I'll cover these separately from the other approaches listed previously.

Protect and Recover Strategy

Let's start with a relatively old strategy that I call the Protect and Recover Strategy. It's also known as the Perimeter Security Strategy. As the cliché goes, it's typically described as having a hard outer shell and a soft, sometimes gooey, center. This analogy is often used because once an organization's perimeter defenses get penetrated, little or nothing impedes attackers from moving laterally in the environment and staying persistent indefinitely. The organization is left trying to recover the original data and IT environment, usually with mixed success. It is considered an old-fashioned strategy by today's standards, but I find a surprising number of organizations still cling to it.

As the name suggests, the focus of this strategy is to prevent attackers from being successful by investing in protection technologies such as firewalls, **Demilitarized Zones** (**DMZs**), proxy servers, and micro-segmentation. Let's go back to 2003 for a great example of why this strategy became so popular. By 2003, there had already been successful mass worm attacks on the internet such as Code Red and Nimda. The risk of such attacks was no longer theoretical, as many people had argued it was at the time. The industry was just starting to understand that software had vulnerabilities and that some of these were exploitable. At that time, I was working on Microsoft's customer-facing Security Incident Response team. Many of the organizations I helped blamed Microsoft for not doing more to protect Windows from such attacks.

There was a widespread belief among enterprise customers that if they replaced their Microsoft Windows with another operating system, then they'd be secure. They were the manufacturer of the world's most used operating system, and subsequently garnered a lot of attention from legitimate security researchers and attackers alike. Of course, now, all these years later, I think everyone understands that all vendors have vulnerabilities in their software and hardware. If you still have any doubts about this, please go back and re-read *Chapter 2*, *Using Vulnerability Trends to Reduce Risk and Costs*. In 2003, the mitigation for the risk that unpatched vulnerabilities posed was the firewall. When Microsoft turned on Windows Firewall by default in Windows XP Service Pack 2, it was hoped that this would prevent the exploitation of vulnerabilities in Windows services and applications listening on the network.

Windows Firewall, together with several other security mitigations, including automatic updates, successfully blunted the mass worm attacks of the era. Many enterprise-scale organizations already had corporate firewalls in place at the perimeter of their networks in 2003. But most of them had exceptions for all traffic going to and from ports 80 and 443 so that HTTP and HTTPS traffic could flow freely; these are the so-called "universal firewall by-pass ports." For the next few years, enterprises that didn't already have DMZs put them in place to enforce better control on network traffic coming from and going to the internet.

This evolution in security strategy was an important and effective step for the industry. But somewhere along the way, the original benefits of perimeter security were distorted. Ultimately, perimeter security was supposed to provide organizations with two things. First, it protected resources that were supposed to be private from public access. Second, blocking anonymous in-bound network traffic to vulnerable services listening on the network gave organizations more time to test and deploy security updates. But the idea that firewalls, DMZs, and network segmentation could somehow provide a long-term solution to vulnerability management or the other four cybersecurity usual suspects, 5 to 10 years before application layer capabilities were built into some of these products, was misguided.

The underlying assumption of the Protect and Recover Strategy is that the organization will be able to deploy and operate adequate protection technologies and processes. If these fail, then recovery is their plan. Because the organization will be so good at protection, it doesn't really need to invest in detection and response capabilities. Most of the organizations that embraced this approach also invested in backup and recovery capabilities. They didn't necessarily invest in backup and recovery capabilities for security purposes; rather, these mitigated the risk of data loss. When their protection strategy ultimately failed, their backup and recovery capabilities were their backstop. So, although these two components weren't necessarily meant to be parts of a coherent cybersecurity strategy, they have been so commonly deployed in enterprise environments that they complement each other very well. If the assumption that the organization can effectively protect themselves 100% of the time, forever, turns out to be untrue, then they can restore from backup.

This approach is characterized by investments primarily in perimeter and network protection, as well as backup and recovery. Professionals with networking expertise could extend their expertise into the security domain. This made a lot of sense since nearly 100% of attacks happened using networks. For many enterprises, their networking groups extended the scope of their charters to include network security, DMZs, and managing firewalls.

The Protect and Recover Strategy has some advantages. Technologies and disciplines like TCP/IP, routing and switching, firewall configuration, and operations are areas that have a trained workforce compared to other security disciplines such as application security, malware reverse engineering, or red and blue teaming. Because it's a relatively mature strategy, there is a very well-developed vendor and consulting ecosystem that has decades of experience supporting it. A trained workforce, and this ecosystem, make this strategy a natural choice for organizations that constrain themselves to primarily using IT staff and vendors they already have contracts with for cybersecurity.

Of course, this strategy also has some disadvantages. History has shown this to be a poor cybersecurity strategy. Some of you might disagree with my description of this strategy, but you can't disagree that in literally every major breach that made headlines in the last 15 or 20 years, the victim organization had been using this approach in some way. The reason this approach has failed time and again is because its underlying assumption is deeply flawed. The assumption that the organization will never be compromised because it will be 100% successful at protecting itself is wildly optimistic. Today, enterprises that don't invest in detection and response capabilities, in addition to protection and recovery capabilities, could be considered negligent.

Reducing the time between compromise and detection is seen as a modern cybersecurity mantra that the Protect and Recover Strategy was not designed to embrace. Subsequently, organizations that use this strategy can have very long periods between compromise and detection, sometimes hundreds of days or spanning years. This strategy doesn't recognize that attackers have a disproportionate advantage over defenders; defenders need to be perfect 100% of the time, which is an unrealistic aspiration, while attackers only need to be good or lucky once.

This strategy relies on developers, administrators, vendors, partners, and users not to make any mistakes or poor trust decisions that could lead to compromise. But as we've seen for decades, users will unwittingly bring threats through layers of perimeter defenses themselves. Without detection and response capabilities, once an organization is penetrated, attackers can typically persist indefinitely, making recovery aspirational and expensive.

The good news is that many of the organizations that used the Protect and Recover Strategy in the past have matured their approach over time. They still employ this strategy but use it in combination with other strategies. They've also upgraded the technologies and products they rely on. Today, next-generation firewalls go far beyond filtering TCP and UDP ports and can perform deep packet inspection. But a question these organizations still need to consider is whether their business partners and supply chain partners still employ this old strategy. For many years, attackers have been targeting small, less mature partners and suppliers in order to get access to their large customers' infrastructures and data. Small legal firms, marketing and advertising firms, and even heating and air conditioning vendors have been targeted for this purpose. Many small firms like these, in countries around the world, still use the Protect and Recover Strategy.

Cybersecurity fundamentals scoring system score

How well does the Protect and Recover Strategy mitigate the cybersecurity usual suspects? *Table 5.3* contains my CFSS score estimates:

Cybersecurity Fundamentals	Score (0 - 20)
Unpatched vulnerabilities	10
Security misconfigurations	10
Weak, leaked, stolen credentials	0
Social engineering	5
Insider threat	0
CFSS Total Score (max = 100)	25

Table 5.3: The CFSS score estimate for the Protect and Recover Strategy

As you might have gleaned from my description of this strategy, although it has some benefits, it doesn't address the cybersecurity fundamentals very well. For unpatched vulnerabilities, I gave this strategy 10/20.

This score reflects that firewalls and segmentation can make it harder for attackers and malware to access exploitable vulnerabilities listening on network ports. If network traffic can't make it to the vulnerable service's port, then the vulnerability can't be exploited. But this mitigation isn't a permanent condition for an exploitable vulnerability. As soon as an administrator changes the rule for the firewall filter blocking the port, then the vulnerability could potentially become instantly exploitable, unbeknownst to the administrator. Typically, filters will block unsolicited in-bound traffic to a port, but they allow in-bound traffic, which is a result of legitimate outbound traffic on the same port. Under the right conditions, the service or application could be enticed to make an outbound connection to a destination under the control of attackers. Firewalls only provide a temporary mitigation to unpatched vulnerabilities, thus giving vulnerability management teams more time to find and patch vulnerabilities. The vulnerable software needs to be uninstalled from the system (which can't be easily done for most operating system components) or needs to be patched. The Protect and Recover Strategy doesn't focus on vulnerability management. The same is true for security misconfigurations. This strategy doesn't help us fully mitigate these two cybersecurity usual suspects – the best it can do is delay exploitation. For this reason, I gave it partial marks in these two areas.

This strategy does nothing to address weak, leaked, or stolen credentials or insider threat. Therefore, both received a score of zero. Finally, I gave this strategy's ability to mitigate social engineering partial marks. Firewalls and DMZs can filter connections based on URLs and IP addresses. They can prevent users who are tricked into clicking on malicious links from connecting to known malicious servers and unauthorized sites. Outbound traffic can be blocked and flagged, as well as inbound replies to such outbound traffic. The challenge has been keeping up with attackers who use compromised systems all over the world to host complex multi-component attacks, and constantly changing sources and destinations for attacks. History has taught us that this approach does not mitigate social engineering attacks very effectively. This is because it still relies on users and administrators to make sound trust decisions, which has always been challenging. Nonetheless, I gave it partial marks for social engineering for what it can do.

With a CFSS total score of 25 out of a possible 100, clearly, this strategy must be used in combination with other strategies in order to really focus on the cybersecurity fundamentals, as well as provide a foundation that an enterprise can build on. Many organizations have already come to this conclusion and have evolved their approaches. But some of the smaller organizations in their supply chain likely still use this strategy because they lack the expertise and resources to evolve. How many small businesses and independent consultants still rely on the firewalls built into their wireless access points for protection?

Protect and Recover Strategy summary

The CFSS total score for this strategy is 25/100. It must be used in combination with other strategies.

Advantages:

- Large vendor ecosystem to help organizations implement and operate
- Relatively large trained workforce with years of experience

Disadvantages:

- History has shown this to be a poor strategy
- Attackers have a disproportionate advantage over defenders because defenders must be perfect
- It relies on developers, administrators, and users not making mistakes or poor trust decisions that lead to compromise
- Individuals bring threats through the perimeter and host-based defenses themselves
- Question to ask yourself: Do your partners or supply chain still use this strategy?
- Once penetrated, attackers can persist indefinitely making recovery aspirational because of a lack of investment in detection and response capabilities

Now, let's examine a strategy that doesn't focus on the network perimeter.

Endpoint Protection Strategy

Next, I'll discuss another relatively old strategy, the Endpoint Protection Strategy. This is what I call a "proxy" strategy. The idea here is that endpoints, such as personal computers, mobile devices, some types of IoT devices, and so on, are used to process, store, and transmit data. Therefore, if we protect these endpoints, we are, by proxy, protecting the data, which is the whole point of data protection. Stated another way, the data will be compromised if the endpoints/devices are compromised, so the endpoints must be protected. Once upon a time, many organizations used this strategy by itself to protect their assets. The underlying assumption is that protecting endpoints and devices is an effective proxy for protecting the organization's data.

The Endpoint Protection Strategy is characterized by investments in host protection technologies like inventorying and vulnerability management solutions, anti-malware solutions, file integrity monitoring, host-based firewalls, application whitelisting, web browser protections, mobile device management, enterprise configuration management, and endpoint hardening, among others. Many of these capabilities are already built into Windows and Linux operating systems, but that doesn't stop endpoint protection vendors from offering better implementations of these features that typically have integrated management and reporting capabilities.

What's an endpoint? It turns out there are a lot of possible definitions. First, it's important to understand that different operating system manufacturers allow different levels of system access to third party ISVs, which can have a big impact on what their solutions are capable of. Vendors that sell endpoint protection solutions have their own definitions that support their specific value propositions. This used to be a short list of major antivirus vendors, but in recent years, the list has grown, and the vendors have become far more diverse. Currently, I count at least 20 different vendors that are actively marketing endpoint protection platform solutions. These include (in alphabetical order): BitDefender, BlackBerry Cylance, Carbon Black, Check Point Software Technologies, Cisco, Crowdstrike, ESET, FireEye, Fortinet, F-Secure, Kaspersky, Malwarebytes, McAfee, Microsoft, Palo Alto Networks, Panda Security, SentinelOne, Sophos, Symantec, and Trend Micro. There are a bunch of other vendors in this space, including regional vendors in China, among others.

Some of these vendors have anti-virus labs with decades of experience, while others are leveraging security company acquisitions and innovations from other areas to try to disrupt the endpoint protection market. Many vendors now include analytics and cloud capabilities as part of their solutions.

Having worked around an anti-malware lab for many years and on a security incident response team, I have an appreciation for this approach. Endpoints are where most of the action happens during a data breach. No matter how good firewall and IDS vendors' products get, they simply do not have the same vantage point as the endpoint device typically has itself. You can see the fish a lot better when you are in the fishbowl versus watching from outside of it. Solutions installed directly on the endpoints enable continuous monitoring and a range of automated actions when triggers are hit. Endpoint protection scanning engines are some of the most impressive feats of programming in the world. These engines have to unpack numerous file compression and obfuscation formats that can be nested by attackers, in virtual computing environments that simulate real operating systems, in order to determine if files are malicious in near real time.

Threats can be file-based, macros, scripts, polymorphic viruses, boot viruses, root kits, and so on, across different operating systems and filesystems. Of course, they have a lot more functionality like heuristics, behavioral analysis, browser protections, malicious IP address filtering, and much, much more. When you dig into the functionality of some of these endpoint protection solutions and consider how hard it is to develop them and keep them current, they are super impressive. However, engineering alone is not enough. These solutions are only as good as the research and response labs that care for and feed them. Maintaining critical masses of great researchers, analysts, and supporting staff is an important function that these vendors provide. The combination of impressive engineering and a world-class research and response lab is the key to selecting an effective endpoint protection vendor. The large vendor ecosystem that I described earlier is very positive. This because it creates healthy competition and these vendors keep each other honest by supporting third-party testing (`av-test.org` and `av-comparatives.org`, among others) and industry conferences (annual Virus Bulletin International Conference (Virus Bulletin, n.d.)) where they discuss how to govern their industry, among other things.

But of course, this approach also has challenges. History has taught us that the Endpoint Protection Strategy, by itself, is insufficient. Have any of the victims in massive data breaches that have hit the headlines in the last 10 years not been running endpoint protection solutions? First, relying on a patient to diagnose and cure itself is an optimistic approach. Once the trusted computing base of a system has been compromised, how can endpoint protection solutions reliably use it to detect threats on the system and clean them? Endpoint protection solutions have been targets for attackers and their malware for decades. One of the first things many families of malware do, after they initially compromise a system, is disable or subvert the endpoint protection solution. This is where remote attestation services can help, but in my experience, few organizations use such services because of their complexity. Some vendors use virtualization techniques to protect their solutions from attackers. But rest assured that attackers will continue to research ways to subvert endpoint protection solutions.

The playing field is never level in this game. Attackers can buy all the endpoint solutions available on the market and test their malware and tools, prior to attacking with them, to ensure no solution can detect or clean them. The endpoint protection vendors don't have that same advantage. But more fundamentally, can the patient really be trusted to cure itself? Some organizations will clean compromised systems with endpoint solutions and allow them to continue running in production, while others have policies to flatten and rebuild any system that has been compromised. Virtualization has made this easier and the cloud, as I'll discuss in detail later, makes this even easier and more effective. But the key to this approach is still accurate threat detection. Keep in mind that although the aspirational goal for all these solutions is to detect, block, and, if necessary, clean 100% of threats, this isn't realistic. The internal goals of research and response labs are typically more realistic and attainable. For example, detection for 100% of threats in the "zoo" (their private malware library) or 100% detection for "in the wild" malware (commonly found threats) are likely common goals among these vendors. But detection goals for emerging threats might be 80%. After all, it takes time for research and response labs to get samples of threats, process them, and deploy appropriate protections to their customers, especially when attackers are using mass automation to generate millions of them constantly.

Would you set sail in a submarine that had the goal of keeping 80% of the water outside the hull? Probably not. But as I wrote in *Chapter 3, The Evolution of the Threat Landscape – Malware*, if you don't use endpoint protection because it doesn't protect the endpoint for 100% of threats, then you aren't protecting the endpoints from the millions of threats that endpoint protection solutions do protect against.

Cybersecurity fundamentals scoring system score

Let's look at how the Endpoint Protection Strategy helps organizations address the cybersecurity fundamentals. *Table 5.4* contains my CFSS score estimates. Remember that these are just estimates based on my experience and they don't reflect the state of the art in endpoint protection. Please feel free to develop your own estimates if you think I'm way off base with mine:

Cybersecurity Fundamentals	Score (0 - 20)
Unpatched vulnerabilities	20
Security misconfigurations	20
Weak, leaked, stolen credentials	15
Social engineering	10
Insider threat	10
CFSS Total Score (max = 100)	75

Table 5.4: The CFFS score estimate for the Endpoint Protection Strategy

I gave this strategy full marks for mitigating unpatched vulnerabilities and security misconfigurations. The combination of inventorying, scanning, updating, hardening, and monitoring can be very effective. For weak, leaked, and stolen credentials, I estimated endpoint protection mitigating 15/20. Organizations that use Secure Access Workstations or Privileged Access Workstations (endpoints hardened for attacks specifically looking for cached administrator credentials) as part of their endpoint strategy can mitigate this type of threat to a large extent, but not completely. Endpoint protection solutions can also help partially mitigate social engineering, as well as insider threat, by making it harder for users and administrators to make some of the common mistakes and poor trust choices that lead to compromise, but it won't fully mitigate malicious insiders.

Although the Endpoint Protection Strategy is insufficient by itself, it would be hard to imagine a successful enterprise cybersecurity strategy that didn't use it in combination with other strategies. It seems like the industry agrees with this assessment as more and more organizations I have talked to are planning to evaluate and adopt **Security Orchestration, Automation and Response** (**SOAR**) solutions in the near future. Some vendors describe SOAR as an evolutionary step in endpoint protection in that it combines functionality from a stack of different capabilities, including endpoint protection and response.

Endpoint Protection Strategy summary

The CFSS total score for this strategy is 75/100. It must be used in combination with other strategies.

Advantages:

- Superior visibility and control running on the endpoint
- Large vendor ecosystem to help with decades of experience
- Constant threat research, response, and evolving technologies to stay ahead of attackers

Disadvantages:

- History has shown this to be a poor strategy by itself as it didn't prevent many of the major data breaches that have been in the headlines.
- Users resist systems that are too restrictive or impact productivity; individuals bring threats through defenses themselves in many cases. This approach can only partially mitigate the mistakes or poor trust decisions that developers, administrators, and users make that lead to compromise.
- Speed is a factor. Relatively slow and complicated vulnerability management processes give attackers an advantage. Organizations that have a good endpoint strategy but deploy security updates and other protections relatively slowly accept more risk.

- Endpoint protection suites have inconsistent performance histories and aspirational performance goals. Organizations that don't understand the internal goals of the endpoint protection vendors might not fully understand the associated risks.
- Managing endpoint security relies on accurate and timely asset inventorying and management capabilities. This has been notoriously hard in on-premises environments. I will discuss how the cloud makes this easier later.
- Many organizations allow employees to use personal unmanaged or partially managed mobile devices, known as the **Bring Your Own Device** (**BYOD**) strategy. Subsequently, the risk associated with the transmission, storage, and processing of corporate data on these devices might not be fully understood.
- Routing, switching, storage, IoT, and other hardware devices might not be integrated into an organization's endpoint protection strategy, but should be.

That's the Endpoint Protection Strategy. Now, let's move on to security strategies involving physical control and security clearances.

Physical Control and Security Clearances as a Security Strategy

I see this next strategy in widespread use, especially by public sector organizations. I call this strategy the Physical Control and Security Clearances Strategy. As you can probably tell from the name, it relies on having physical control of the infrastructure used to transmit, store, and process data, as well as data classification and associated security clearances. The idea behind this strategy is that not all data has the same relative value to the organization that controls it. By classifying the data into different categories that reflect the relative value of the data, we can ensure the most valuable data is protected in ways that are commensurate with that value.

There are many different data classification schemes in use in the public and private sectors; many organizations have developed their own data classification schemes. We don't have to look any further than the US federal government to see a great example of a data classification scheme that has been deployed on a massive scale. Executive Order 13926 (United States Government Publishing Office, 2009) defines a three-tier system for classifying national security information. It defines those three tiers as Top secret, Secret, and Confidential. Another similar example is the UK government's security classification for third-party suppliers (U.K. Cabinet Office, 2013). It also defines three classifications that indicate the sensitivity of the information. These categories include Top secret, Secret, and Official. There are many other examples of data classification schemes.

Data classification policies such as these can dictate the people, processes, and technologies that must be employed to handle data in each category. As such, the number and nature of the categories in the data classification schemes that organizations adopt can have a huge effect on organizations' cultures, recruiting practices, IT investments, and budgets, among other things.

This is where security clearances can become a factor. For some organizations, in order for personnel to be granted access to information that has been classified into a specific category, those personnel must have a current security clearance that permits access to information in that category. For example, if someone doesn't have a clearance that permits access to data that has been classified as secret, then they should not be granted access to information that has been classified as secret. In order to get a security clearance, there can be background checks involved, some of which are much deeper and more involved than others.

For example, some security clearances require a criminal history check. Other, deeper, background checks require a criminal history check, an employment background check, and a financial credit score check, in addition to the applicant providing personal references, who will be interviewed as part of the background check process. Some security clearances have specific citizenship requirements. Some clearances have a one-time process that applicants go through, while other clearances need to be periodically renewed. Some technology vendors give their customers insight into the background checks they subject their employees to. Microsoft is an example; they've published Office 365 Personnel Controls (Microsoft Corporation, 2019).

You might be wondering why employers simply don't perform all these checks periodically as a matter of course. Different countries and jurisdictions have local labor laws and statutory regulations that protect the privacy and the rights of employees. For example, in the US, too many credit checks can lower an individual's credit score. Allowing employers to institute administrative procedures that potentially negatively impact current or potential employees is not cool. Note that some data classification schemes don't require security clearances, because they are designed to simply provide a way for the staff handling the data to understand how it should be handled.

From a security perspective, organizations that are serious about this approach are essentially trying to create a closed system for their data that has a high level of security assurance. People that handle data, especially sensitive data, will be vetted to minimize the likelihood that they have malicious intent or could be easily bribed or blackmailed to break their organization's policies. This concept of assurance also extends to their processes and technology. For example, some organizations have policies that dictate that data will only be transmitted, stored, and processed by hardware and software that has gone through their certification processes. All other electronics are never allowed into their on-premises environments. This includes anything that has a power cord or a battery. The business processes that these vetted employees use to operate their certified systems are carefully engineered to ensure auditability and ensure that multiple people participate, to keep each other honest. The underlying assumptions that make this closed system work is that the organization has end-to-end control of their entire infrastructure, as well as that their supply chain is subject to their security clearances and certification processes. Numerous trusted IT suppliers participate in these types of supply chains.

The essence of this strategy can be traced back decades, if not centuries, where it's been heavily employed by militaries and national security organizations throughout the world. Of course, there have been national security failures throughout history, which tells us this approach isn't foolproof. In modern times, this model has been evolving. It works well on a small scale, but it gets incrementally harder to manage as it scales up. To scale their operations, it became harder for these organizations to manage all their IT in-house. The types of organizations that use this model face the same IT resource and recruiting challenges as other industries.

Subsequently, many of them have outsourced much of their IT to cope with these challenges. In many cases, this means that the contractors they use to manage their IT have physical access to the datacenters and servers processing their data.

More specifically, in the course of their work, these contractors have access to the operating systems and hypervisors running on those servers, the virtualized workloads, and the data in those workloads. But the organization that owns the data must maintain their closed system to protect the data – that's their strategy. Because the contractors potentially have access to classified data, they require the same security clearances as the organization's regular personnel. The contractor's datacenters and the IT infrastructure in them also must go through the organization's certification processes. Despite all of the effort dedicated to clearances, history has taught us that they don't mitigate insider threat completely. Since this is all complicated and very expensive to accomplish, to make it economically viable, the contracts between these organizations and qualified contractors tends to be very long term, sometimes 10, 20, or even 30 years in duration. This managed service provider model is the way that IT has been outsourced to these organizations for the last 20+ years. Of course, there's a bunch of advantages and disadvantages to using managed service providers; I'll touch on a few of these later.

To recap, the focus of the Physical Control and Security Clearances Strategy is the security assurance of hardware and software, and periodic background checks of datacenter staff and administrators. It is characterized by investments in people, processes, and technologies that help maintain physical security, assurance, and confidence in the character of datacenter staff and administrators. Data classification typically plays a critical role in helping protect the most important data. This approach has numerous benefits. Some governments literally have hundreds of years of practice using this type of strategy. It can help partially mitigate insider threat by identifying potentially risky job candidates and personnel that could have access to sensitive data. Third-party verification or attestation of the trustworthiness of hardware and software contributes to security assurance and helps demonstrate due diligence. There is a large vendor ecosystem to help organizations that want to pursue this type of strategy.

Of course, this strategy also has some important disadvantages and limitations. First, data classification is challenging for most organizations. Using data classification can be very helpful for organizations that want to ensure that their most sensitive data is protected appropriately. Treating all data as if it has the same relative value to the organization is the most expensive way to manage data. But data classification schemes are notoriously difficult to successfully institute in large organizations. In my experience, the organizations that have the most success with data classifications are those organizations where security is deeply embedded in the culture. Military and paramilitary organizations, law enforcement agencies, national defense departments, and intelligence agencies are some examples of organizations where data classification is deeply engrained into the culture, people, process, and supporting technologies.

Many commercial organizations have tried and failed, some multiple times, to institute data classification schemes. The typical challenge for these organizations is finding a way to classify data that doesn't make it hard or impossible for information workers to get work done. Organizations that allow the same people who create the data to classify the data usually end up with large amounts of data that have been over-classified or under-classified, depending on the consequences to employees. For example, in military organizations, under-classifying data could lead to severe consequences such as loss of life or criminal charges. Data in these organizations tends to get over-classified because workers are better safe than sorry; they'll rarely get into trouble for over-classifying data, despite the immense extra costs when everyone in a large organization does this habitually.

In organizations where there aren't life or death consequences or national security concerns, data can be under-classified more easily, making it easier for information workers to get their work done. Executives in some of these organizations believe the rules don't apply to them and demand ad hoc access to whatever data they need, regardless of how it is classified or why. This is one reason they are often the targets for Business Email Compromise schemes and other social engineering attacks. They can get access to any data and often, they are exempted from the inconvenient security controls that mitigate such attacks. A recipe for disaster that is often realized.

Of course, in neither of these scenarios, where data is under- or over-classified, does data classification fulfill its promise. Some commercial and public sector organizations decide not to institute data classification schemes because their past attempts to do so have all failed or have not achieved their desired objectives. Instead, these organizations have concluded that data classification is too complex and expensive to be worthwhile. For them, it's easier and more effective to treat all their data as if it's the same value. Some of them will employ less formal, very simple data classification schemes by marking some documents and data as confidential or internal only. But the data protection requirements are the same for all their data.

Keep in mind that in many organizations, the one system that typically stores, processes, and transmits the data of all classifications is email. It's relatively rare for organizations to have two separate email systems – one email system for unclassified data and one for classified data. Subsequently, data of all classifications can end up in emails, which can become a source of data leakage.

Data residency is often a requirement for organizations that embrace this security strategy. That is, they require that all datacenters that process and store their data must be located in a specific country or jurisdiction. For example, all the data for a federal government department must stay within the national borders of their country. There are a few different reasons for data residency requirements, but the most common one is that data residency provides better security for the data, and that the organization requires data sovereignty, which they likely will not have within the borders of another country. In order to maintain their closed system, they cannot risk putting a datacenter in a location that another government has sovereign control over.

Data residency doesn't mitigate any of the cybersecurity usual suspects. This is because 99% of the attacks happen remotely over the network, regardless of the physical location of the datacenter. Attackers don't care where the datacenter is physically located because that is not an effective mitigation for the vast majority of attacks.

This is why many organizations that embrace the Physical Control and Security Clearances Strategy put "air gaps" into their networks. Put another way, their networks are not directly connected to the internet. Organizations try to accomplish air gaps a few ways. Some simply don't procure internet connectivity from an ISP. Some use data diodes that are certified to only allow network traffic flow in one direction. Some organizations call a network "air-gapped" when it's behind a DMZ with very specific firewall rules. To truly air gap a network can be incredibly difficult to accomplish and maintain over time. The ubiquity of mobile devices, IoT devices, and common office equipment, like photocopiers, that want to phone home with inventory and service information makes it challenging to keep a disconnected network, disconnected. Some organizations maintain two networks, one for classified information and the other for non-classified information. Information workers in these environments typically have two computers on their desks, one connected to each of these networks. Some of the organizations that use air-gapped networks require all mobile devices, laptops, and electronics to be kept in lockers at the front door of their facilities.

Organizations that achieve and maintain air-gapped networks can make it much harder for attackers to leverage the cybersecurity usual suspects to initially compromise their networks. However, as the Stuxnet attack and so many other attacks on air-gapped networks over the years have proven, it's not an insurmountable challenge. Moreover, data residency is far less effective than other available controls that help mitigate the risks that these organizations have in mind with their data residency requirements, such as encryption and effective key management. As I'll discuss later, with modern encryption and key management technologies, organizations can achieve very strong data protection while operationalizing data so that it can help them make better decisions faster.

Perhaps the biggest challenge for the Physical Control and Security Clearances Strategy is that the world has changed in some keys ways, all of which will make this strategy harder to pursue and less effective as time goes on. The organizations that currently use strategies like this one are being challenged in several ways.

For example, most organizations today want to take advantage of Machine Learning and Artificial Intelligence. They'll be challenged to do this in a scalable way in their accredited on-premises, air-gapped IT environments or via their traditional managed service providers' datacenters. In order to keep up with their adversaries, who aren't encumbered by the same certification and accreditation processes, organizations are going to have to change the way they procure and operate IT services. To do this, they will have to give up some of the end-to-end control they have had for decades. Their closed systems will have to evolve. For some of these organizations, this kind of change is super hard because it's initially uncomfortably different from how they've done governance, risk, and compliance for the last few decades. This doesn't mean they have to settle for a less secure IT environment, but they do have to re-evaluate how to mitigate the risks they care about in a world where they don't own the infrastructure end-to-end. Rising on-premises IT costs to maintain the status quo, in the face of tsunami after tsunami of innovation in the cloud, means that organizations that employ this type of strategy will either successfully evolve or become increasingly irrelevant.

Cybersecurity fundamentals scoring system score

Let's look at *Table 5.5* – how well does the Physical Control and Security Clearances Strategy help address the cybersecurity fundamentals? I'll estimate scores for two flavors of this strategy, one with an air-gapped network and one without an air-gapped network. As you'll see, this makes a big difference in terms of the scores.

Cybersecurity Fundamentals	Score (0 - 20)
Unpatched vulnerabilities	10
Security misconfigurations	10
Weak, leaked, stolen credentials	15
Social engineering	10
Insider threat	10
CFSS Total Score (max = 100)	55

Table 5.5: The CFSS score estimate for the
Physical Control and Security Clearances Strategy with an air-gapped network

None of this strategy's attributes, such as data classification, security clearances, or end-to-end control of certified hardware help to fully mitigate unpatched vulnerabilities, security misconfigurations and weak, leaked, or stolen passwords. Like the Protect and Recover Strategy, an air-gapped network can give security teams more time to address these cybersecurity usual suspects, but they still must be addressed. Weak, leaked, and stolen credentials are harder to use if there is no remote network access to the target network. If the principle of least privilege is applied accurately and consistently, this can make it harder to achieve unauthorized access to sensitive data.

As I discussed earlier in this chapter, data classification and security clearances can help mitigate insider threat, particularly the malicious insider. But it doesn't fully mitigate users and administrators that make mistakes or poor trust decisions that lead to compromise. Because of this, I gave it partial marks for insider threat and social engineering. This approach seems to be optimized to mitigate unlawful government access to data, such as military espionage. For the types of organizations that I have talked to that use this strategy, this is definitely a real risk for them – perhaps their highest priority risk. But clearly, this isn't the only high priority risk they need to mitigate.

I've seen organizations use this strategy without implementing an air-gapped network. Without the air-gapped network, relying on data classification, security clearances, and end-to-end certified hardware is far less effective at addressing the cybersecurity fundamentals:

Cybersecurity Fundamentals	Score (0 - 20)
Unpatched vulnerabilities	0
Security misconfigurations	0
Weak, leaked, stolen credentials	0
Social engineering	10
Insider threat	10
CFSS Total Score (max = 100)	20

Table 5.6: The CFSS score estimate for the
Physical Control and Security Clearances Strategy without an air-gapped network

To really mitigate the cybersecurity usual suspects, whether an air-gapped network is used or not, this approach needs to be used in combination with other cybersecurity strategies. I've met with many organizations that already know this and have been pursuing complementary strategies for years. But the cultures of many of these organizations make it difficult for them to adopt new approaches and technologies; to coin a phrase, they have a glacial approach in an era of unmitigated global warming. The internet and the cloud have democratized IT, giving everyone capabilities that they never had before. The challenge for organizations that have used this strategy for years or decades is adapting their current approach quickly enough, all to enable them to mitigate a larger number of well-resourced adversaries than they've ever had in the past.

Physical Control and Security Clearances Strategy summary

The CFSS total estimated score for this strategy, using air-gapped networks, is 55/100. For organizations that use this strategy, but without an effective air-gapped network, my estimate of the CFSS total score is 20/100. My conclusion is that this strategy must be used in conjunction with other cybersecurity strategies in order to fully address the cybersecurity fundamentals.

Advantages:

- Militaries and governments have hundreds of years of practice using similar approaches
- Air-gapped networks can help partially mitigate some of the cybersecurity usual suspects
- Helps partially mitigate insider threat, including unlawful government access to data, by making it harder for malicious insiders to succeed
- Third-party verification/attestation of hardware contributes to security assurance and helps demonstrate due diligence
- Has a large vendor ecosystem to help organizations that pursue this approach

Disadvantages:

- Enormous costs are usually associated with the type of certified infrastructure typically leveraged with this approach
- The underlying assumption that data residency provides better security is not valid
- Since most attacks are perpetrated remotely without physical access to hardware and regardless of the physical location of data, the success of this approach depends heavily on network air gaps to partially mitigate the cybersecurity usual suspects
- Data in highly restrictive, air-gapped environments can be harder to operationalize
- Doesn't fully mitigate insider threat because it focuses on malicious insiders, not automation, which could help also mitigate non-malicious insider threats
- Gives attackers an advantage because they can use newer technologies faster than defenders

Now, let's move on and consider how some organizations use compliance as a security strategy.

Compliance as a Security Strategy

Compliance and cybersecurity are two different, slightly overlapping disciplines. Compliance typically focuses on proving that an organization meets requirements defined in regulated, industry, and/or internal standards. Compliance can be helpful in numerous ways, chief among them would be for cybersecurity insurance purposes and demonstrating due diligence to limit liability. This is different from cybersecurity, which focuses on identifying, protecting, detecting, responding, and recovering (NIST, n.d.). But I have seen many organizations conflate these different disciplines because they can overlap each other, as I've illustrated in *Figure 5.2*. I've seen similar illustrations where compliance is a subset of cybersecurity or vice versa. I think arguments can be made for all of these approaches. The approach that some organizations I have discussed this with have taken is to rotate these two circles on top of each other and pretend that they are the same thing.

That's not to say that organizations can't align their efforts in order to pursue both compliance and cybersecurity. This is what most organizations need to do, but many fail to do so:

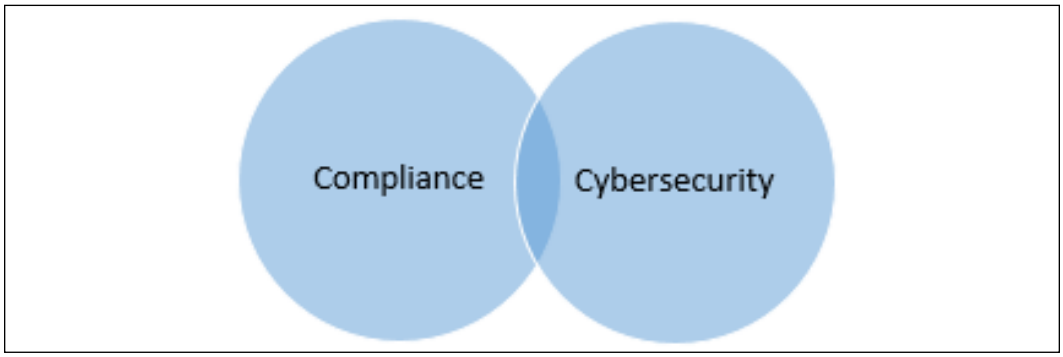


Figure 5.2: Compliance and security disciplines overlap but are different

I've discovered that there are a variety of reasons that organizations conflate compliance and cybersecurity. First, some regulated standards have non-compliance penalties or fines associated with them. This provides an incentive for organizations to prove that they are meeting these standards and invest in compliance programs. But since most organizations have resource constraints, many of them believe they are forced to decide whether to use their resources on compliance or cybersecurity. In some cases, organizations end up using this strategy because their well-resourced, well-intentioned compliance organization over-functions. That is, they extend their efforts beyond proving they meet applicable standards, to performing functions that you'd typically see a security team perform. There's nothing wrong with this, but we need to recognize that their area of expertise and the center of gravity for their program is compliance. Some of the organizations that I have seen using this strategy do so, simply because their compliance program is older and more mature than their cybersecurity program; they've had compliance obligations for years or decades in their industry, and cybersecurity is a relatively new investment for them.

The underlying assumption of this strategy is that meeting compliance obligations is sufficient for protecting the organization's data. Subsequently, the focus is meeting the organization's regulatory, industry, and internal compliance obligations, and demonstrating this in audits.

These could include standards like PCI, HIPAA, GDPR, NIST standards, ISO standards, or an organization's own internal IT security standards, among others. This strategy is characterized by investments in people, processes, and technologies that help organizations meet their compliance obligations. This typically manifests itself as well-defined control sets and repeatable processes that are periodically audited.

This strategy can be very advantageous, healthy, and positive for organizations that do not have a cybersecurity strategy or have immature governance practices. Most of the regulated security-related standards that have been instituted in industries provide a minimum set of requirements that organizations should work to achieve. The steps that organizations typically need to take to get their IT governance, infrastructure, and operations in shape to be audited for the first time against an industry standard can dramatically improve their security posture and their overall cybersecurity program. Organizations should not underestimate the effort and potential change related to complying with regulated standards and industry standards. This effort is typically rewarded with much better security than where they started, as well as a foundation they can potentially extend and continue to build on.

The challenge for many organizations is to recognize that most regulated security-related standards are minimum requirements, not some sort of certification that means they can't be compromised. Although regulatory compliance is required for many organizations, it's insufficient to protect their systems and data from modern-day threats. This is where Compliance as a Security Strategy tends to fall short. History has taught us that this is a poor strategy. There is no shortage of examples of large, well-funded organizations that met regulated standards but were breached all the same. Think about all the financial institutions, retailers, and restaurants that met their industries' regulated standards, but were breached anyway. Think about all the organizations in the healthcare industry around the world that worked hard to comply with stringent regulated industry data protection standards, who lost control of patient data to attackers. My own personal data has been compromised multiple times in data breaches in all of these industries over the past 15 years. This doesn't mean that regulated security-related standards are worthless. As I mentioned, they are very positive for many, many organizations. I'd rather use my credit card in a restaurant that tries to comply with PCI DSS than one that doesn't.

Regulated security-related standards are insufficient by themselves. There's at least a couple of reasons for this. First, standards like these typically have a defined scope, such as credit card holder information or patient information. The control sets to support these standards are designed for the infrastructure and data that are in the defined scope. But what about the other HVAs that organizations have? If the organization uses its limited resources to only address the scope that's audited and subject to penalties, they are likely not paying enough attention to other HVAs and their broader infrastructure. The second reason regulated standards are insufficient is that they rarely keep pace with the threat landscape or advances in technology. This has more to do with how slowly standards can be adopted in industries and their economic impact than with the standards bodies themselves. Deploying updated security-related standards requirements to millions of retailers and restaurants around the world takes years. This is why organizations need a broader cybersecurity strategy that embraces compliance but supplements its shortcomings in material ways. Simply put, enterprises need to do both.

Cybersecurity fundamentals scoring system score

My CFSS score estimates for Compliance as a Security Strategy reveals that this strategy can partially mitigate all the cybersecurity usual suspects. Remember, the goal is to find a strategy or combination of strategies that give us a perfect 100/100 CFSS total score. Subsequently, this strategy will need to be used in combination with other strategies to fully address the cybersecurity usual suspects:

Cybersecurity Fundamentals	Score (0 - 20)
Unpatched vulnerabilities	10
Security misconfigurations	10
Weak, leaked, stolen credentials	10
Social engineering	10
Insider threat	10
CFSS Total Score (max = 100)	50

Table 5.7: The CFSS score estimate for Compliance as a Security Strategy

I gave this strategy partial marks across the board because it can help organizations mitigate all these threats, but it's typically used with limited scope and is slow to adapt to changes in the threat landscape. This strategy can and does create a foundation, albeit incomplete, that many organizations can build on with complementary approaches.

Compliance as a Security Strategy summary

The CFSS total estimated score for this strategy is 50/100. This strategy can be very beneficial for organizations as a starting point for a broader cybersecurity strategy. Organizations that integrate their compliance requirements into a more holistic cybersecurity strategy can potentially reduce complexity, costs, and achieve better security.

Advantages:

- Can be very positive for organizations that have no security strategy or have immature governance practices
- Third-party verification/attestation by auditors is valuable to demonstrate due diligence
- Large vendor and audit firm ecosystem to help
- Organizations that integrate compliance requirements into a holistic cybersecurity strategy potentially reduce complexity, costs, and achieve better security
- Complying with some regulated standards, like GDPR, for example, will raise the bar for many organizations

Disadvantages:

- History has shown this to be a poor strategy as many organizations that complied with standards were breached anyway.
- Typically relies on compliance and audit teams, as well as third-party auditors, to arbitrate the organization's security posture.
- Focuses on implementing control sets specified in regulated standards with a specific scope that typically does not include all HVAs.

- Only attains minimum requirements specified by regulations when they were last published; rarely reflects modern-day risks and mitigations.
- Attackers have a disproportionate advantage over defenders. This is because they can have complete visibility into the control sets required to comply, and those control sets rarely keep pace with changes in the threat landscape.
- In some cases, regulatory compliance uses resources that could otherwise be used for more effective cybersecurity.

Now, let's look at the Application-Centric Strategy.

Application-Centric Strategy

This is another proxy strategy. Applications process, store, and transmit data. If we protect the application, then by proxy, we are protecting the data. This approach focuses on protecting applications by reducing the number of vulnerabilities in them and the severity of those vulnerabilities. It also endeavors to make the vulnerabilities that are inevitably left in applications really difficult, if not impossible, to exploit. These are the same principles that underpin the vulnerability improvement framework that I introduced in *Chapter 2*, *Using Vulnerability Trends to Reduce Risk and Costs*. An underlying assumption of this approach is that it is much less expensive to fix bugs and mitigate vulnerabilities before an application is released. This involves investments in people, processes, and technologies, which can include threat modeling, security development life cycles, static and dynamic code analysis tools, penetration testing, mobile device management, mobile application management, bug bounties, and others.

I'm a big believer in this strategy; after all, would you set sail in a submarine where someone is drilling holes in the hull from the inside? This continues to be an underestimated risk, as enterprises still don't seem to select vendors or solutions based on their security development practices.

I led Marketing Communications for Microsoft's **Security Development Life Cycle** (**SDL**) (Microsoft Corporation, n.d.) for several years and saw how it could help development teams first-hand.

You don't have to have a massive development organization like Microsoft to benefit from this strategy. As the saying goes, a rising tide lifts all boats. CISOs, security teams, compliance professionals, and development organizations can all help raise the security tide mark for their organization over time by implementing security development education, policies, and processes that are supported by tools, to help improve the quality of both the software developed in-house and procured from third parties. For example, requiring that every in-house developed application requires a threat model, prior to developers writing any code, can help improve design and mitigate potential vulnerabilities. Similarly, requiring static code analysis at specific milestones in development can help reduce the number and severity of vulnerabilities that make it into production. Organizations that don't enforce security requirements in every phase of the development process typically pay a higher price for this decision, after their applications have been deployed.

But like all the other strategies, this one has drawbacks and limitations as well. The same operating system features, tools, IDEs, development libraries, and frameworks (C++, the JRE, .NET, and so on) that are used to protect applications can also be a persistent source of vulnerabilities. The **Java Runtime Environment** (**JRE**) is the perennial example. It saves development teams lots of time and expense, but the opportunity cost is that their application could inherit vulnerabilities that need to be patched in the JRE itself. The time between vulnerabilities being discovered in these frameworks and being fixed represents a risk to the users of their applications.

Another drawback of this strategy I've seen organizations grapple with numerous times is that although fewer vulnerabilities and lower severity vulnerabilities are measurable metrics, they are hard to translate into business value. Arguing that attacks didn't happen because of the investment in application security can be tough arguments for CISOs and development organization leaders to make and other executives to understand. What seems like common sense to CISOs and vulnerability management teams can remain nebulous to other stakeholders.

As I wrote in *Chapter 2*, *Using Vulnerability Trends to Reduce Risk and Costs*, using data from your vulnerability management program on the state of the environment can help you make the case for application security. Trying to drive the number of unpatched vulnerabilities to zero and using data to help other executives understand the progress against this goal and the associated costs can help them understand why it is important to prevent new vulnerabilities from being introduced into the environment via third-party and in-house applications.

Cybersecurity fundamentals scoring system score

All that said, let's see how the Application-Centric Strategy scores in the CFSS:

Cybersecurity Fundamentals	Score (0 - 20)
Unpatched vulnerabilities	20
Security misconfigurations	20
Weak, leaked, stolen credentials	10
Social engineering	10
Insider threat	10
CFSS Total Score (max = 100)	70

Table 5.8: The CFSS score estimate for the Application-Centric Strategy

I gave this strategy full marks for its ability to mitigate unpatched vulnerabilities and security misconfigurations. I realize this is a little optimistic for most organizations, but there are some scenarios where this could be possible. I gave this strategy partial marks for its ability to mitigate insider threat, social engineering, and weak, leaked, or stolen credentials. For example, designing applications that require MFA and provide rich logging and audit capabilities can help partially mitigate these threats.

Application-Centric Strategy summary

All organizations can benefit from this approach. However, by itself, its CFSS total estimated score is 70/100. I recommend that organizations embrace this strategy and subsidize it with other approaches that will help fully address all the cybersecurity fundamentals.

Advantages:

- Can reduce the number and severity of vulnerabilities in software that the organization procures and develops in-house.
- Can lower maintenance costs, minimize business disruptions, and measurably improve application security.
- Leverages mitigations built into operating systems, IDEs, development libraries and frameworks (C++, the JRE, .NET, and so on) and containers. This reduces complexity, costs, and effort for development teams while potentially improving security.
- Large existing vendor ecosystem to help.

Disadvantages:

- Relies on developers to produce vulnerability-free source code or makes it impossible for vulnerabilities to be exploited; history teaches us this is optimistic
- Subject to vulnerabilities in the operating systems, IDEs, development libraries, frameworks, and containers, among others
- Business ROI can be challenging to communicate

Onward now, to the Identity-Centric Strategy.

Identity-Centric Strategy

You'll remember that one of the cybersecurity usual suspects is weak, leaked, and stolen passwords. Credentials and the assets that they protect have been currency for attackers for decades. Many people reuse passwords across applications, systems, and services. When one of those is compromised and the credentials are stolen, attackers immediately try those credentials on other systems and services across the internet, such as major online banking portals, e-commerce sites, social networks, and so on. The industry has long wanted to deprecate passwords in favor of better authentication methods and use data from authentication and authorizations systems to make better resource access decisions. These concepts are central to the Identity-Centric Strategy.

Although the concept of identity and proving your identity is ancient, the Identity-Centric Strategy is a relatively new strategy that gained popularity rapidly. The idea behind this strategy is that during most successful data breaches, at some point, attackers will use legitimate credentials. How can we use this to our advantage to protect, detect, and respond to attacks? Well, authentication and authorization processes can potentially generate some useful metadata. For example, if we can ascertain the approximate location that an authentication or authorization request is coming from, we might be able to calculate a level of confidence in its legitimacy. Similarly, if we can compare some key attributes of the request to characteristics of past requests from the same account, this too might help provide us with some level of confidence that the request was legitimate. There's a bunch of metadata like this that can help organizations protect, detect, and respond to attacks. Here's a partial list of such data:

- Strength of the credential used for the request (older protocols versus newer protocols)
- Location and temporal data:
    - Origin location of request
    - Time of day of request
    - Time between requests from different locations – is it impossible to travel between those locations in the time between the requests?
- Trustworthiness of the device the request is coming from:
    - Does it have a valid digital certificate installed by the organization?
    - Is it a corporate-managed device or an unmanaged personal device?
    - Does the hardware or operating system version have known unpatched vulnerabilities?
- User behavior:
    - How many times did the user enter incorrect credentials?
    - When was the last time the user was promoted for MFA and what was the result?

The underlying assumption of this strategy is that organizations can better protect data, detect compromises, and respond faster by better protecting the identities used to access data, and by using identity metadata to look for indicators of compromise. The focus of this approach is protecting the credentials used to access the organization's data, and especially the credentials of privileged accounts, such as administrators. Incident response teams, forensics experts, as well as Red and Blue teams, all know that privileged account credentials are like gold to attackers. When I worked on Microsoft's customer-facing incident response team, attackers' modus operandi was very consistent; once the attackers initially compromise an IT environment using one of the cybersecurity usual suspects, within seconds, their scripts were running, trying to harvest cached credentials on the compromised system. They would use those credentials to move laterally through the environment if they could, looking for more cached credentials along the way. Finding cached credentials for privileged accounts made it much easier for attackers to penetrate the environment even deeper, and then get access to more resources and data. If attackers were able to exfiltrate a copy of the victim's Microsoft Active Directory, they would perform an offline attack, using rainbow tables and/or other tools to get access to more credentials relatively quickly (Wikipedia, n.d.). Once attackers got to this stage, recovery was aspirational. I met numerous organizations over the years that found themselves in this scenario. Some of them decided to "share" their IT environment with attackers because recovery was too expensive and resource-intensive. Others decided to rebuild their infrastructure from scratch or used the compromise as the impetus to start fresh in the cloud. Since attackers try to harvest credentials as a matter of course, many organizations focus on protecting credentials and use identity metadata to accelerate detection.

The Identity-Centric Strategy is characterized by investments in MFA, enforcing the principle of least privilege, identity management technologies, credential vaulting and hygiene practices, and detecting credentials that are being misused (Pass-the-Hash and Golden Ticket attacks are examples). For example, to counter attacks on Microsoft Active Directory, Microsoft has taken numerous steps to make it harder for attackers to succeed. In addition to engineering improvements in their products, they have published guidance on how to harden Active Directory (Microsoft Corporation, 2017).

They also published a lot of content on what is referred to as a "Red Forrest" or **Enhanced Security Administrative Environment** (**ESAE**) (Microsoft Corporation, n.d.). This type of architecture helps protect privileged credentials and makes it much harder for attackers to get access to them. But these advanced architectures and configurations are not for the faint of heart. Using **Privileged Access Workstations** (**PAWs**) in isolated domains sounds good in theory, but very few organizations have the administrative self-discipline required to govern and operate their IT in such a strictly controlled environment. However, protecting credentials in an on-premises distributed environment has never been easy.

The identity space has exploded over the past 20 years. There are vendors that specialize in access management, privileged access, identity governance, and several other areas. Some vendors that sell technologies that support an Identity-Centric Strategy call identity the "new perimeter" to highlight the importance of protecting credentials and credential hygiene. There are several vendors in the identity space that can help make protecting credentials easier and provide access to valuable metadata to accelerate anomaly detection. Some of the vendors that I have seen organizations leverage include CyberArk, Okta, Ping Identity, BeyondTrust, Microsoft, and others.

Cybersecurity fundamentals scoring system score

How does the Identity-Centric Strategy score in the CFSS? It doesn't fully address any of the cybersecurity fundamentals:

Cybersecurity Fundamentals	Score (0 - 20)
Unpatched vulnerabilities	5
Security misconfigurations	5
Weak, leaked, stolen credentials	15
Social engineering	10
Insider threat	10
CFSS Total Score (max = 100)	45

Table 5.9: The CFSS score estimate for the Identity-Centric Strategy

This strategy doesn't mitigate unpatched vulnerabilities or security misconfigurations. But some vulnerabilities and security misconfigurations require authenticated access in order to be exploited. Organizations that focus on enforcing the principle of least privilege and practice good credential hygiene can make reliable exploitation of vulnerabilities and misconfigurations much more difficult and "limit the blast radius." Subsequently, I gave this strategy partial marks for these two cybersecurity fundamentals. I couldn't give it full marks for mitigating weak, leaked, and stolen credentials because legacy applications tend to fall through the cracks with this strategy; MFA typically can't be deployed everywhere, and metadata isn't always going to be available. Similarly, this approach can help partially mitigate insider threat by implementing **Just-in-Time** (**JIT**) and **Just-Enough-Administration** (**JEA**) models, credential vaulting, and other mitigations. Social engineering can be partially mitigated with MFA and least privilege, among other controls, but can't be completely mitigated.

Identity-Centric Strategy summary

This strategy needs to be used in combination with other strategies to fully mitigate the cybersecurity usual suspects. Although it didn't score particularly high, it's certainly a valuable, modern, complementary approach to improving protection, detection, and containment capabilities. However, that might be understating the importance of identity in a modern cybersecurity strategy. Identity will remain central to an effective cybersecurity strategy. Investments in this area can pay big dividends for CISOs.

Advantages:

- Focuses on improving governance and technologies with a historically poor track record
- A large vendor ecosystem to help
- Can help manage risk related to weak, leaked, and stolen passwords
- Multifactor authentication is becoming ubiquitous
- The strength of a credential, the location of a login attempt, the trustworthiness of the device and multifactor authentication controls can all help build confidence in the legitimacy of authentication requests

- Can quickly identify authentication/authorization anomalies
- Can add friction to the authentication/authorization processes, which makes it harder for attackers to infiltrate
- Can bolster containment efforts and make it harder for attackers to move laterally

Disadvantages:

- Traditionally, federated identity systems have been complex, expensive, and hard to govern and manage; simply put, identity has always been hard
- Legacy applications can be challenging to govern and secure using a modern Identity-Centric Strategy
- MFA is typically not implemented everywhere, leaving gaps and opportunities for attackers
- Can be complicated, time-consuming, and expensive to fully implement in enterprise on-premises environments.

Next, let's look at a strategy that has had a resurgence of popularity – the Data-Centric Strategy.

Data-Centric Strategy

The Data-Centric Strategy has been growing in popularity for several reasons, including many high-profile data breaches, revelations about government data collection programs, and the increasing threat of intellectual property theft. There are also increasing regulatory demands that aim to help protect consumer privacy and have significant noncompliance fines associated with them, such as GDPR, for example. In addition, because of the challenges we discussed with the Protect and Recover Strategy, the Endpoint Protection Strategy, the Application-Centric Strategy, the popularity of **Bring Your Own Device** (**BYOD**) IT environments, and the emergence of IoT, some organizations have decided to stop using strategies that solely rely on proxies to protect their data. Instead of relying on the security provided by firewalls, endpoints, and applications, their strategy is to protect the data, no matter where it is.

Whether their data is inside their perimeter, accessed from a managed device, or processed by an application that meets their security development requirements, the data still needs to be protected. Some CISOs make the assumptions that endpoints cannot be fully trusted, and that data can move in unexpected ways without their knowledge. They want to ensure that even in scenarios where they are not in control of their data, it's still protected.

This is where the Data-Centric Strategy can help. There are several underlying assumptions for this approach. First, data, not the systems that process it, transmit it, or store it, is the HVA. Instead of focusing on the security of hardware and software that handles data, the focus should be on the data itself. Another assumption is that data will move without the organization's approval or knowledge, and therefore it must be protected, regardless of where it is. Some CISOs go so far to assume that some of the systems that process their data are compromised, and that the data must be protected in a compromised environment. Finally, organizations still require that their data can be shared appropriately within their organization and with authorized partners, such as outside manufacturing, marketing, PR, and law firms. That is, although the data must be secure, it still must be accessible and useable internally and externally. The focus of this strategy is to protect data wherever it is transmitted, processed, and stored, preferably forever, but for a reasonably long period of time. This approach is characterized by investments in **Data Loss Prevention** (**DLP**), encryption and key management technologies, and potentially data classification.

A simplified example of this is encrypted PDF files, which can be read by authorized users, but the content cannot be copied and pasted. A more complicated example is, of course, the extreme data-centric solutions offered by blockchain platforms that implement data protection mechanisms as part of the data itself.

The heart of this strategy is encryption and key management; if data is encrypted everywhere, all the time, the attack surface area can be dramatically reduced. For example, instead of trying to protect all files, everywhere they currently are and will be in the future, forever, encryption can help make this more manageable.

Encrypting all the files reduces the attack surface by shifting the focus from protecting all the files to protecting a much smaller number of encryption keys. If strong, properly implemented encryption is employed, the primary focus can shift from the security of the encrypted files to managing the keys that are used to encrypt and decrypt them. Of course, if you don't have access to the encrypted files, you can't decrypt them, and the data is lost. So, you shouldn't be cavalier with your data just because it's encrypted. However, the mathematical properties of properly implemented strong encryption can help reduce risk.

Besides reducing the attack surface, encryption buys organizations time. That is, properly encrypted data looks the same as random noise, and without the keys to decrypt the data, it will likely take many years of effort for attackers to decrypt a portion of the data. The confidentiality and integrity of the data is preserved during that time. But it is still prudent to assume that encrypted data has a finite lifespan. Periodically rotating keys and re-encrypting data can help extend this lifespan, but at some point, the algorithms or key lengths used will no longer provide adequate protection in the face of new technologies and advances in cryptoanalysis. A thoughtful approach to managing encryption, decryption, and keys is required; this is not a "set it and forget it" solution to data protection.

You might be wondering, given that various types of encryption have been around for millennia, if encryption and key management are so powerful, then why haven't organizations always been encrypting everything, everywhere? Why have there been so many data breaches involving unencrypted data? Traditionally, there's been a tension between securing information and operationalizing information. Let me give you an example of this tension. I'll use a completely fictional scenario, where there are life and death consequences for unauthorized access to information – a witness protection program.

In this fictional scenario, the list of witnesses that the program is protecting is handwritten on paper. The list hasn't been digitized in any way; it only exists on paper. No one person has ever seen the entire list, as parts of the list are managed by separate program managers and are physically compartmentalized. The list is put into a fireproof filing cabinet that has a combination lock and has steel bars locked across its drawers. The keys to these locks are given to separate program officers, requiring all of them to be present to open the filing cabinet.

The filing cabinet is in a vault, in a secured area in the middle of police headquarters, surrounded by on-duty police officers, with armed guards at the one fortified entrance to the building 24 hours a day. Of course, the building has an extensive security system, including video surveillance, mantraps, and card key access points. The vault can only be opened by following a specific protocol that requires the participation of two additional senior law enforcement officials, under specific conditions.

I hope you agree that the list in this scenario has been secured in a way that mitigates many potential risks and that unauthorized access to the list would require extraordinary measures. Ethan Hunt from Mission Impossible might be able to breach all these controls, but I'm sure you'll agree it would be difficult for most other people. However, an additional consequence of these controls is that legitimate, authorized access to the list has been encumbered, making it a complicated and slow process. In this scenario, since there can be life and death consequences to unauthorized access, access is purposely designed to be slow, cumbersome, and meticulous. However, if there was an emergency or some other need for quick access or repetitive access to the list, this process would frustrate those needs.

In another fictional scenario, a company that specializes in providing real-time advice on trading stocks has a different challenge. This company will go out of business if they can't access information, process it, and provide valuable advice to their large customer base in near real time. The data they have typically loses its value within minutes. Security controls are important to the company as they have very aggressive competitors and regulators that would like to understand what their secret to success is. However, if security controls encumber the near real-time distribution of information inside the company or to their customers, the company will fail to keep its promises to its customers and go out of business in a hyper-competitive market. This company purposely prioritizes speed and agility over security. If they don't, they won't be in business very long.

These two scenarios demonstrate the tension between the need for data security and the need to operationalize information, which has traditionally challenged organizations. Combine this tension with the fact that encryption and key management have traditionally required specific, relatively hard to find and expensive expertise, and this begins to explain why organizations haven't simply encrypted all their data, all the time.

Because of this tension and the traditional challenges associated with encryption, many organizations decide to encrypt only their most sensitive data. This reduces complexity and costs, while still ensuring their most valuable data is protected. To do this, many organizations have adopted data classification in order to identify and more effectively protect high-value data. But as I discussed earlier in this chapter, data classification policies are notoriously difficult for organizations to implement and adhere to. Many of the organizations I have talked to, particularly those that tried to implement data classification policies and failed, have concluded that it is more efficient to treat all data as if it's the same value. For them, this approach is less complicated and less expensive than trying to consistently identify the relative value of individual datasets and apply different security control sets based on that value. But these organizations are still faced with the challenge of managing encryption and key management.

Wouldn't it be cool if CISOs didn't have to make these trade-offs? That is, they could have it all – uncompromising data security, the operational capabilities that enable organizations to move fast, the ability to share data when needed, and better visibility and control. Who wouldn't want that? This is what the Data-Centric Strategy seeks to enable. Instead of just managing the security of the hardware and software that handles data, secure the data itself using encryption, key management, authentication, and authorization. In a world where data breaches have become common, this strategy can provide an effective line of defense when all other protection mechanisms fail. In addition, if encryption and decryption functions require authentication and authorization, the metadata generated from these activities can provide useful information on where the data is and who is trying to access it.

From a high level, the technologies used to support these capabilities include client-side or server-side encryption libraries or applications, **Public Key Infrastructures** (**PKIs**), federated identity systems with authorization capabilities, as well as logging and reporting capabilities. A good example of a service that combines all these components is **Azure Rights Management** (**Azure RMS**) (Microsoft Corporation, 2019). Let me give you an example of how this service works, from a high level.

A company needs to protect confidential information from falling into the wrong hands, but needs to share it with their outside law firm in a way that still protects the confidentiality and integrity of the data. They encrypt the file using Azure RMS and assign a policy to it that defines who is authorized to open and decrypt the file. They send the file to the law firm via Office 365 email. When staff at the law firm try to open the file, they get prompted to enter their Azure Active Directory credentials. Because they are also an Office 365 corporate user and have an identity federation configured with the company's account, when they enter their credentials, Azure Active Directory authenticates them and reads the policy to determine what type of actions they are permitted to do with the file. The policy allows the law firm to open the file, decrypt it, and read it. If the file is forwarded to someone that doesn't have those permissions, they won't be able to open it or decrypt it. Meanwhile, the company can track where in the world authentication requests to open the file have come from, which credentials were used in authentication requests, failed and successful attempts to open the file, and so on. Pretty cool. I'll discuss other cool capabilities that the cloud provides in *Chapter 8*, *The Cloud – A Modern Approach to Security and Compliance*.

You might have noticed that the one critical component that enables the example scenario I described is identity. An identity strategy, like the Identity-Centric Strategy I described earlier in this chapter, is required for this Data-Centric Strategy to be successful. Without authentication and authorization capabilities, the Data-Centric Strategy isn't scalable.

Data Loss Prevention (**DLP**) can also be employed in a Data-Centric Strategy. DLP can be a powerful tool to help prevent data from leaving an organization in an unauthorized way, including malicious and non-malicious data theft and leakage. DLP can monitor data that moves via the network, email, USB drives, and other removable media. But increasingly ubiquitous encryption can make it more difficult for DLP to achieve complete visibility. Additionally, DLP policy violations rarely result in consequences for the employees and executives that break them; this provides little incentive to pay attention to DLP-related policies. Finally, DLP can only slow down malicious insiders as they steal information, not stop them completely.

They will almost always find a way to smuggle information out of an IT environment, like using the camera on their mobile phone to take a picture of it right off the screen of a secure workstation with DLP running on it, for example. However, DLP combined with the Physical Control and Security Clearances Strategy, an air-gapped network in a facility that enforces a policy prohibiting all outside electronics including mobile phones, has physically removed USB and peripheral ports on computers in the facility, and searches employees as they enter and leave the facility has a much better chance of preventing data theft. But few organizations outside those responsible for national security impose these types of controls.

Cybersecurity fundamentals scoring system score

Perhaps unexpectedly, the Data-Centric Strategy does not earn a great CFSS score by itself. After all, if the underlying infrastructure used for encryption, key management, authentication, authorization, logging, DLP, and other functions is compromised using one or more of the cybersecurity usual suspects, then attackers can potentially get access to the data before it gets encrypted, or they could get access to credentials or decryption keys. Protecting the data is a powerful mitigation, but it requires that the components that make it possible are also protected:

Cybersecurity Fundamentals	Score (0 - 20)
Unpatched vulnerabilities	5
Security misconfigurations	5
Weak, leaked, stolen credentials	0
Social engineering	15
Insider threat	15
CFSS Total Score (max = 100)	40

Table 5.10: The CFSS score estimate for the Data-Centric Strategy

I gave this approach partial marks for unpatched vulnerabilities and security misconfigurations because it can protect the confidentiality and integrity of the data, while vulnerability management teams scan and update systems; like the Protect and Recover Strategy, this approach can give vulnerability management teams more time to get this done.

It can also protect data for a period of time after the exploitation of vulnerabilities and misconfigurations. But it doesn't prevent the attackers from destroying the data or encrypting it themselves using ransomware. Crucially, it doesn't prevent attackers from exploiting vulnerabilities in the infrastructure, moving laterally, collecting credentials, persisting, and collecting data before it gets encrypted in web browsers and email clients, and so on. Of course, most credentials in Microsoft Active Directory and other modern directory services are encrypted, but that's not the focus of the Data-Centric Strategy. It offers nothing new to protect passwords, as it relies on identity systems and federated identities. Subsequently, I gave it zero out of 20 for weak, leaked, and stolen passwords.

This strategy can mitigate some forms of social engineering when used with the principle of least privilege and a meaningful separation of duties. This is also true of insider threats. Encrypted data can remain confidential, even when administrators make mistakes that lead to poor security outcomes, but there are limits. Malicious insiders will potentially have a harder time with a meaningful separation of duties that limits their access to key material. Thus, I gave both social engineering and insider threat partial marks.

Data-Centric Strategy summary

Despite its relatively low CFSS score, I am a fan of the Data-Centric Strategy. Authenticated, authorized encryption and decryption operations can be very effective for protecting data. I think using the metadata that I described can also be very helpful to security teams. For CISOs who try to protect everything as if it's the same value to the organization (which can be a recipe for disaster), dramatically reducing the attack surface area that they must focus on can be very helpful.

For many organizations, data classification can help determine which datasets they need to focus on protecting. But data classification is notoriously hard to implement and adhere to. Modern approaches to encryption and key management make it much easier and less expensive to encrypt everything all the time, especially in the cloud.

Advantages:

- Potentially reduces the surface area to protect by focusing on data on the endpoint, email, network, proxy servers, and in the cloud.
- Can help protect data, detect data breaches, and respond to incidents quicker than traditionally possible.
- Modern, properly implemented encryption can effectively protect data from unauthorized access for relatively long periods. This time can be helpful as security teams can then focus on the cybersecurity fundamentals and other advanced capabilities with more confidence.
- Encryption can help make data destruction easier; destroying the keys effectively destroys the data.
- DLP can be a powerful tool to help prevent data from leaving an organization and to help detect data leakage.

Disadvantages:

- Many organizations find data classification policies and technologies hard to implement and use consistently over time. Subsequently, many organizations have tried and failed to do data classification in a meaningful way.
- Key management can be challenging for some organizations. An on-premises PKI is not for the faint of heart and requires technical expertise. A failed PKI can have disastrous implications; the cloud makes this much easier.
- Many organizations terminate encrypted communications to inspect data and apply DLP policies as it moves. Increasing the use of encryption for data in transit and at rest has made it more challenging for DLP to be effective.
- Enforcing DLP policy violations can be challenging for some CISOs; how often is a senior executive reprimanded for breaking DLP policies? Many organizations do not adequately enforce policy violations when they're flagged by DLP.
- Relies on a sound identity strategy and federated identity implementation, which can be challenging to architect, implement, operate, and govern.

Moving on, the final cybersecurity strategy that I will discuss is the Attack-Centric Strategy.

Attack-Centric Strategy

The idea behind the Attack-Centric Strategy is that the ways CISOs protect systems, detect compromises, and respond to attackers, should they be informed by the **TTPs** that attackers actually use. Put another way, understanding how attackers operate and planning defenses around that makes those defenses more effective. The underlying assumption of this approach is that forcing attackers to be successful multiple times during intrusion attempts makes it much harder for them and decreases detection and recovery times. The focus of this approach is understanding how attackers operate and making each step and each tactic they use ineffective. Lowering attackers' ROI by increasing the time, effort, and costs associated with their attack will force attackers to rethink or abandon their attack. This approach is characterized by investments in numerous areas to block or impede attackers at each stage of their attack.

Two consummate examples of this approach are Lockheed Martin's Intrusion Kill Chain (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.) and MITRE ATT&CK® (MITRE). Both of these complementary approaches are informed by the steps attackers take to attack their victims, and the specific tactics, techniques, and procedures they use. For example, the Intrusion Kill Chain Approach defines seven phases or stages during an attack: Reconnaissance, Weaponization, Delivery, Exploitation, Installation, Command and Control, and Actions on Objectives (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.) Attackers could use some or all of these phases in their attacks. Knowing this, organizations can layer their defenses to detect, deny, disrupt, degrade, deceive, and destroy at every stage of the attack (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.). This will make it much harder for attackers to succeed because they must potentially defeat multiple layers of defenses, specifically designed around their modus operandi.

Similarly, MITRE ATT&CK® is designed to be a knowledge base of attackers' TTPs. Currently, there are three flavors of ATT&CK, that is, PRE-ATT&CK, ATT&CK for Enterprise, and ATT&CK for Mobile (MITRE ATT&CK®). PRE-ATT&CK focuses on the earliest stages of an attack, prior to victims' compromise.

In Intrusion Kill Chain parlance, PRE-ATT&CK covers all the phases in an attack prior to exploitation. ATT&CK then covers the rest of the phases of the attack, but at a lower level, more granular way than described by the Intrusion Kill Chain approach. For example, ATT&CK helps defenders design layers of capabilities across Initial Access, Execution, Persistence, Privilege Escalation, Defense Evasion, Credential Access, Discovery, Lateral Movement, Collection, Command and Control, Exfiltration, and Impact (MITRE). This approach makes a lot of sense to me as it is very well aligned with the strategy that I introduced in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*, which includes the cybersecurity fundamentals and advanced cybersecurity capabilities. Based on this, let's see how the Attack-Centric Strategy scores using the CFSS.

Cybersecurity fundamentals scoring system score

The Attack-Centric Strategy has the highest CFSS score of any of the individual strategies I've examined in this chapter. In fact, my estimates of how capable it is of addressing all the cybersecurity fundamentals gives it a near-perfect score as shown in *Table 5.11*:

Cybersecurity Fundamentals	Score (0 - 20)
Unpatched vulnerabilities	20
Security misconfigurations	20
Weak, leaked, stolen credentials	20
Social engineering	15
Insider threat	20
CFSS Total Score (max = 100)	95

Table 5.11: The CFSS score estimate for the Attack-Centric Strategy

The reason this approach scores so well is that it focuses on the ways that attackers initially compromise IT environments and the methods and tools they use post initial compromise. That is, it covers all the bases. The reason I didn't give it a perfect 100/100 is that social engineering is nearly impossible to completely mitigate in enterprises. Someone once coined the phrase "the problem exists between the chair and the keyboard" (PEBCK).

Despite the industry's best efforts to educate Information Workers, executives, and IT administrators, and design software and hardware to make it harder for social engineering attacks to be successful, attackers are relying on it more and more. In an environment where mitigations for the cybersecurity usual suspects are well managed, attackers are forced to turn to the one tactic they know has the best chance of succeeding: social engineering. They will continue to rely on humans to make mistakes and poor trust decisions, as the research I provided in *Chapter 3*, *The Evolution of the Threat Landscape – Malware* suggests.

Attack-Centric Strategy summary

The Attack-Centric Strategy garnered a very high CFSS score. It can help CISOs and their teams focus on the cybersecurity fundamentals, which, in turn, creates a solid foundation for other, more advanced cybersecurity capabilities. This strategy is also capable of helping security teams go beyond the fundamentals and thoughtfully implement advanced cybersecurity capabilities and help protect their HVAs. That said, for most organizations that have limited resources, it isn't easy or inexpensive to design, procure, implement, operate, and support layers and layers of cybersecurity capabilities. Many organizations that aspire to use this approach realize they don't have the technical expertise or budget to truly embrace it in the long term.

Depending on the previous strategy or strategies that an organization has leveraged, they might have only invested in protection, but not necessarily detection and response. Subsequently, if they start using the Attack-Centric Strategy, they will likely increase investment in detection and response.

Advantages:

- Potentially levels the playing field between attackers and defenders as they both understand attacker TTPs
- Forces attackers to be successful multiple times instead of just once or twice like many of the other cybersecurity strategies are designed for
- Designed to help detect intrusions as early in the attack as possible, in order to reduce remediation and recovery time and costs
- Vast ecosystem of vendors to help

Disadvantages:

- This approach requires most organizations to increase investments in detection and response capabilities, thus typically increasing complexity and costs.
- Typically relies on technology from multiple vendors to work in concert to protect, detect, and respond to threats. This could require technical expertise across multiple vendors' technologies; this might not be a realistic requirement for many organizations with limited resources and technical talent.
- Because of all the layers this approach requires, it can be challenging to architect, deploy, and operate.
- This can be a relatively expensive strategy to pursue.

We have covered quite a bit of ground! Let's conclude our review of these strategies by summarizing what we've been discussing.

Cybersecurity strategies summary

We have reviewed several popular cybersecurity strategies. These strategies include:

- Protect and Recover Strategy
- Endpoint Protection Strategy
- Physical Control and Security Clearances Strategy
- Compliance as a Cybersecurity Strategy
- Application-Centric Strategy
- Identity-Centric Strategy
- Data-Centric Strategy
- Attack-Centric Strategy

A summary of my CFSS score estimates for these strategies is provided in *Table 5.12*. As you can see, I gave the Attack-Centric Strategy the highest estimated CFSS score. In my view, it is the only strategy that has the greatest potential to help organizations address the cybersecurity fundamentals and mitigate the cybersecurity usual suspects:

Cybersecurity Strategy	Unpatched vulnerabilities	Security misconfigurations	Weak, leaked, stolen credentials	Social engineering	Insider threat	Total Score
Protect and Recover Strategy	10	10	0	5	0	25
Endpoint Protection Strategy	20	20	15	10	10	75
Physical Control and Security Clearances Strategy	10	10	15	10	10	55
Compliance as a Cybersecurity Strategy	10	10	10	10	10	50
Application-Centric Strategy	20	20	10	10	10	70
Identity-Centric Strategy	5	5	15	10	10	45
Data-Centric Strategy	5	5	0	15	15	40
Attack-Centric Strategy	20	20	20	15	20	95

Table 5.12: CFSS score estimate summary

The reality is most organizations that I have met with use a combination of some of these strategies. For example, it would be bold for an enterprise not to have both a perimeter security strategy and an endpoint security strategy, even as the industry offers newer, shinier technologies. Many organizations have some regulatory compliance requirements that they must pay attention to. It can be helpful for those organizations that already use some of these approaches to deliberately and thoughtfully reconcile where there has been over-investment and under-investment, and where gaps currently exist. This is another advantage that the Attack-Centric Strategy has over these other strategies and combinations of them – investment and gap analysis is built right into it. I will discuss this in more detail in *Chapter 7, Measuring Performance and Effectiveness*.

You might disagree with my CFSS score estimates for some or all of these strategies. That's good. I encourage you to use the CFSS to perform your own scoring estimates for all the approaches I examined in this chapter and others I didn't cover. Security professionals all have different experiences, which could lead them to score one or more of these strategies higher or lower than I have. Frankly, this is to be expected as I've never met a security professional that didn't have an opinion. Despite this, most organizations do not have a cybersecurity strategy that their CISOs or other executives can articulate. My objective for this chapter is to provoke critical thought about the ways that organizations have been approaching cybersecurity and perhaps hold a mirror for CISOs and security teams to look into.

Now, let's look at a couple of other potentially helpful approaches that are different, in some important ways, from the more classical approaches discussed in this chapter. Let's start with DevOps.

DevOps and DevSecOps

DevOps represents a change in the way that organizations have traditionally approached application development and deployment. Traditionally, developers and operations staff were managed as separate disciplines that rarely worked together. Developers would write code to specifications and when they wanted to deploy it, they "threw it over the fence" to the operations team. Sometimes, the operations team encountered issues deploying the application, so they would send it back to the development team with the issues that were preventing successful deployment. Developers and operations would iterate on this process, typically at a slow and frustrating pace. Because these groups only communicated with each other periodically, the developers often lacked the operational and environmental context that would help them develop applications that could be deployed and operated in a real IT environment. Similarly, the operations teams often didn't have the technical details on the application to help them perform successful deployments. The feedback loop between teams was slow, leading to milestone delays, slow development cycles, and quality issues.

DevOps tries to address these challenges by tightly integrating developers and operations staff. They can give each other feedback more efficiently and faster when they work with each other day in, day out. Operations staff can inform the design and functionality choices that the developers make while they are developing the application. The developers can get constant feedback on the viability and supportability of their ideas from the operations staff. This can lead to faster development and deployment cycles, better quality applications, less rework, and happier teams.

DevOps typically includes concepts like continuous testing, **Continuous Integration** (**CI**), **Continuous Delivery** (**CD**), continuous deployment, and continuous performance monitoring. This goes beyond the technologies, services, and products that support these concepts, because most organizations have to make significant changes to their development philosophies, cultures, and processes to embrace DevOps.

DevSecOps is DevOps with the explicit acknowledgment that security must be embedded in the philosophies, cultures, processes, and supporting technologies for this approach to be successful. Some argue that the "Sec" in DevSecOps is gratuitous because DevOps cannot be done properly without embedding security in it. I agree whole-heartedly. If your organization is currently doing DevOps and has decided that they'll evolve into a DevSecOps approach later, then you are likely already doing DevOps wrong. Remember, someone recently said that "culture eats strategy for breakfast." This is why DevOps is potentially so powerful and transformational for IT organizations.

The value of DevOps is extended when it is used together with containers and/or cloud computing. For example, since infrastructure is code in the cloud, infrastructure is deployed, configured, and supported using code. This means that provisioning and managing infrastructure in the cloud can benefit from the virtues of DevOps. Developers can specify the hardware, software, and configuration for infrastructure in the code they write, informed by the requirements and continuous feedback provided by operations teams. This approach enables organizations to provision infrastructure faster than traditional approaches and at virtually any scale desired.

From a security perspective, DevOps offers a powerful model for building and deploying applications and infrastructure. This is where the concept of a CI/CD pipeline is useful. The pipeline typically handles functions like checking code into a repository, automated building, automated testing, and deploying the tested code into production. The pipeline itself can be composed of a combination of tools, products, and services from one or multiple vendors. Some organizations that have embraced DevOps deploy all applications and all infrastructure via a CI/CD pipeline. Put another way, nothing gets into their production environments unless it goes through a pipeline. Enforcing pipeline policies like this can offer organizations at least a few advantages versus legacy approaches. For example, when applications and infrastructure are required to go through a pipeline and the pipeline has automated checks to ensure regulatory, industry, and internal security standards are met, then everything that makes it into production is in this known good state.

This assurance makes short-lived environments possible by enabling infrastructure to be discarded and redeployed in a known good state, every few hours. If that infrastructure gets compromised, attackers will only have control of that asset for a relatively short time before it gets blown away and replaced. This can make it harder for attackers to get a foothold in an environment and remain persistent. It can also help dramatically reduce the amount of work for vulnerability management teams. Instead of constantly performing inventories of systems, they can scan them for security vulnerabilities, patching, and rebooting them. They can scan and patch the relatively small number of "gold images" used for infrastructure deployments. When a short-lived infrastructure is discarded and replaced, the new infrastructure is based on the up-to-date gold image. Verifying the patch status of a short-lived infrastructure is less work for vulnerability management teams, and less disruption to the business. There are similar advantages for compliance teams, as well as internal and external auditors.

Of course, DevOps isn't a panacea. DevOps and CI/CD pipelines done poorly can be a bad thing for organizations. To date, most of the organizations I've discussed DevOps with only do use it in parts of their IT environment, and the rest of the organization is still chained to legacy models. Developers can become enamored with CI/CD pipelines. For example, developers that embrace CI/CD pipelines can end up spending more of their time developing tools and automation for their pipelines than working on applications and infrastructure. Organizations can also end up with too many CI/CD pipelines. Predictably, some attackers see potential victims shifting to DevOps and using CI/CD pipelines, so they target the pipeline infrastructure itself; CI/CD pipelines could end up becoming HVAs for some organizations, and require more security rigor than they were initially prepared for.

I think the security and non-security advantages of DevOps and CI/CD pipelines outweigh any challenges they present. This is the reason the entire industry has been moving to this model and will continue to do so for many years to come.

Zero Trust

One of the underlying assumptions of all the strategies I've discussed in this chapter is that once a user or system has authenticated access to the IT environment, then it is trusted. The popularity of this is evidenced by the ubiquity of **Single Sign-On** (**SSO**) requirements among enterprises. It's interesting that this assumption is as old as the oldest strategies I examined. That assumption hasn't changed much since enterprises started procuring their first PCs. Some will argue that this assumption is one reason the industry has seen so many data breaches over the decades. I think it's fair to say that champions of the Zero Trust model would agree with this. Although this approach is nascent, it was first conceived about 15 years ago by a group of CISOs, according to industry lore.

The concept behind this model is that all resources, including those inside the perimeter, should be untrusted. This makes a lot of sense in a world where less and less IT infrastructure and fewer and fewer information workers are behind corporate firewalls. For example, the ongoing explosion of IoT devices should easily outnumber the number of desk-bound PCs and servers in datacenters, the same way that mobile devices have dramatically eclipsed them over the past 15 years. Additionally, as I discussed in my examination of the Protect and Recover Strategy, history has taught us that the old-school perimeter security approach, by itself, is a failure because its underlying assumptions have been proven to be wildly optimistic. You'll remember that one of those assumptions was that security teams could achieve perfect protection, forever, and they didn't require investments in detection and response capabilities.

If we assume that all network traffic, systems, devices, and users cannot be trusted, regardless of whether they are behind an enterprise perimeter, this could potentially change a security team's approach in a substantial way. Authenticating and authorizing applications, network connections, devices, and users for each operation they attempt, instead of just at the time of first access, can make it harder for attackers to initially compromise an environment, move laterally, and stay persistent. Don't trust and always verify.

Marry this rigor with the capabilities of the Identity-Centric Strategy that I discussed, and it can help make better authentication and authorization decisions in real time. This approach might also benefit from many of the capabilities of the Endpoint Protection Strategy to provide the visibility and control needed on endpoints. Some vendors are resurrecting **Network Access Control** (**NAC**) and **Network Access Protection** (**NAP**) to ensure endpoints meet corporate policies for security update status and anti-virus protection, among other requirements. In fact, this approach could borrow something from all the strategies I discussed in order to address the cybersecurity fundamentals.

Assuming everything is untrusted can definitely lead to positive improvements in many organizations' security postures. I don't think there's any doubt about that. For example, it might challenge some developers to try to design e-commerce applications capable of doing transactions on systems that are assumed to be compromised. The result should be better than assuming the system will *never* be compromised, right?

However, the success of this model will depend on its implementation. For example, I mentioned that some vendors are using NAC/NAP in their Zero Trust solutions. The reason NAC/NAP failed the first time they became popular in the industry is because of the horrible user experience they imposed on users. All VPN users that connected to their office, where NAC/NAP were implemented, had the same dreaded experience at one time or another; they just wanted to check their emails, download a presentation, or quickly get access to some information, only to be quarantined and forced to slowly download and install security updates, anti-virus signatures, endure reboots, and so on. Despite the positive advantages of ensuring systems were patched before connecting to the corporate network, it degraded the user experience so much that users would avoid connecting to the network for as long as they could. When they finally had to connect to the network, the user experience was even worse because of the backlog of updates the system required. This had the opposite effect on security to what was intended. Those vendors that offer Zero Trust solutions that leverage this same approach are doomed to the same fate. Users will only deal with so much overhead in their daily work before they actively try to avoid it or work around it.

The user experience shouldn't be worse in environments with Zero Trust implementations – it needs to be better. This one factor will likely decide the effectiveness and fate of the Zero Trust model.

Chapter summary

CISOs and security teams should select their organization's cybersecurity strategy based on how well it addresses the cybersecurity fundamentals, as the minimum bar. Without examining how their strategy mitigates all the cybersecurity usual suspects, they could be lulling themselves into a false sense of security. The Cybersecurity Fundamentals Scoring System (CFSS) can help security teams determine how well their current or future strategies address the cybersecurity fundamentals.

Of the strategies examined in this chapter, the Attack-Centric Strategy was deemed as the strategy most capable of mitigating the cybersecurity usual suspects and enabling advanced cybersecurity capabilities. The Endpoint Protection Strategy and the Application-Centric Strategy rounded out the top three strategies in this evaluation, but will need to be used in combination with other strategies to fully address the cybersecurity fundamentals.

DevOps is a holistic approach that leads to changes in development philosophies, cultures, and processes for the organizations that embrace it. This is the destination that many organizations aspire to get to. This approach might not be as beneficial for legacy IT environments, where the more traditional cybersecurity strategies that I examined might be used during the transition to modern architectures, like the cloud.

The Zero Trust model holds the potential to raise the security waterline for the entire industry. But how this approach is implemented and the user experience it imposes will determine its effectiveness and its fate.

That completes my examination of cybersecurity strategies. In the next chapter, we will dive deep into an implementation example of the strategy that had the highest CFSS estimated total score, the Attack-Centric Strategy.

References

1. Ashford, W. (August 3, 2016). *One in five businesses hit by ransomware are forced to close, study shows*. Retrieved from ComputerWeekly: https://www.computerweekly.com/news/450301845/One-in-five-businesses-hit-by-ransomware-are-forced-to-close-study-shows
2. Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D. (n.d.). *Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains*. Retrieved from Lockheed Martin: https://lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/LM-White-Paper-Intel-Driven-Defense.pdf
3. European Data Protection Board. (2019). *First overview on the implementation of the GDPR and the roles and means of the national supervisory authorities*. Brussels: European Data Protection Board.
4. ISO. (n.d.). *ISO/IEC 27001 INFORMATION SECURITY MANAGEMENT*. Retrieved from ISO: https://www.iso.org/isoiec-27001-information-security.html
5. Microsoft Corporation. (May 31, 2017). *Best Practices for Securing Active Directory*. Retrieved from Microsoft: https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/security-best-practices/best-practices-for-securing-active-directory
6. Microsoft Corporation. (October 19, 2019). *Office 365 Personnel Controls*. Retrieved from Microsoft Corporation: https://docs.microsoft.com/en-us/office365/enterprise/office-365-personnel-controls
7. Microsoft Corporation. (November 30, 2019). *What is Azure Rights Management?* Retrieved from Microsoft Corporation: https://docs.microsoft.com/en-us/azure/information-protection/what-is-azure-rms
8. Microsoft Corporation. (n.d.). *Active Directory administrative tier model*. Retrieved from Microsoft: https://docs.microsoft.com/en-us/windows-server/identity/securing-privileged-access/securing-privileged-access-reference-material
9. Microsoft Corporation. (n.d.). *Microsoft Security Engineering*. Retrieved from Microsoft Corporation: https://www.microsoft.com/en-us/securityengineering/sdl/

10. MITRE ATT&CK®. (n.d.). *MITRE ATT&CK Frequently Asked Questions*. Retrieved from MITRE ATT&CK®: https://attack.mitre.org/resources/faq/
11. MITRE. (n.d.). *MITRE ATT&CK*. Retrieved from MITRE ATT&CK®: https://attack.mitre.org/
12. NIST. (n.d.). *Cybersecurity Framework*. Retrieved from NIST: https://www.nist.gov/cyberframework/online-learning/five-functions
13. NIST. (n.d.). *Cybersecurity Framework*. Retrieved from NIST: https://www.nist.gov/cyberframework
14. PA Consulting. (n.d.). *Oak Door Data Diode*. Retrieved from PA Consulting: https://www.paconsulting.com/services/product-design-and-engineering/data-diode/
15. U.K. Cabinet Office. (October 2013). *Introducing the Government Security Classifications*. Retrieved from GOV.UK: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/251481/Government-Security-Classifications-Supplier-Briefing-Oct-2013.pdf
16. United States Government Publishing Office. (December 29, 2009). *Executive Order 13526 of December 29, 2009*. Retrieved from GovInfo: https://www.govinfo.gov/content/pkg/CFR-2010-title3-vol1/pdf/CFR-2010-title3-vol1-eo13526.pdf
17. Virus Bulletin. (n.d.). *Virus Bulletin*. Retrieved from Virus Bulletin: https://www.virusbulletin.com/
18. Wikipedia. (n.d.). *Rainbow table*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Rainbow_table

6

Strategy Implementation

In the previous chapter, I discussed numerous cybersecurity strategies. In this chapter, I'll take one of those strategies and illustrate how it can be implemented in a real IT environment. The objective is to take the theoretical and make it a little more real for you. I'll provide some tips and tricks I've learned in my career along the way.

In this chapter we will cover the following:

- What is the Intrusion Kill Chain?
- Some ways that the traditional Kill Chain model can be modernized
- Factors to consider when planning and implementing this model
- Designing security control sets to support this model

Let's begin by deciding which of the strategies we discussed previously will be implemented in this chapter.

Introduction

The Attack-Centric Strategy had the highest **Cybersecurity Fundamentals Scoring System** (**CFSS**) estimated total score. It earned nearly a perfect score with 95 points out of a possible 100. It earned such a high score because it almost fully addresses all of the cybersecurity fundamentals, with the exception of social engineering, which can't really be fully mitigated.

Two popular examples of Attack-Centric frameworks used by security professionals in the industry include the **Intrusion Kill Chain** (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D) and the **MITRE ATT&CK®** model (MITRE).

In this chapter, I'll provide an example of how an Attack-Centric Strategy can be implemented. The model I will focus on is the Intrusion Kill Chain framework first pioneered by Lockheed Martin. I have found that security professionals either love or hate this model. There seems to be plenty of misconceptions about this model; I hope this chapter will contribute to clearing some of these up. I've actually had the opportunity to do a big budget implementation of it, so I have some first-hand experience with it. As I contemplated this implementation, which I led strategy and architecture for, I realized an Intrusion Kill Chain could probably be implemented in several different ways. I'll describe one way this framework can be implemented, fully recognizing that there are other ways it can be implemented, and that mine might not be the best way.

The Intrusion Kill Chain framework is based on Lockheed Martin's paper *Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains*. In my opinion, this paper is required reading for all cybersecurity professionals. Some of the concepts in this paper might seem mainstream now, but when it was first published, it introduced concepts and ideas that changed the cybersecurity industry. Some might argue that this model has seen its best days and that there are now better approaches available, like the MITRE ATT&CK model. This isn't quite true as ATT&CK is meant to be complementary to the Intrusion Kill Chain approach. According to MITRE:

> *"ATT&CK and the Cyber Kill Chain are complementary. ATT&CK sits at a lower level of definition to describe adversary behavior than the Cyber Kill Chain. ATT&CK Tactics are unordered and may not all occur in a single intrusion because adversary tactical goals change throughout an operation, whereas the Cyber Kill Chain uses ordered phases to describe high level adversary objectives."*
>
> *— (MITRE, n.d.)*

Also, keep in mind that the CFSS score suggests that the Intrusion Kill Chain approach can nearly fully mitigate the cybersecurity usual suspects. Regardless of what this approaches' champions or its detractors say about it, *Chapter 5*, *Cybersecurity Strategies* gave you the CFSS method to decide its potential efficacy for yourself. I recommend making use of this tool when faced with disparate opinions about cybersecurity strategies. Additionally, keep in mind that this approach can be used in on-premises IT environments, in cloud environments, and in hybrid environments. Another strength of this approach is that it is technology neutral, meaning it isn't limited to a specific technology or vendor. This means it can be used by most organizations now and into the future as they evolve their IT strategies.

What is an Intrusion Kill Chain?

An Intrusion Kill Chain is the stages or phases that can be used in attacks by attackers. The phases provided in Lockheed Martin's paper include:

- Reconnaissance
- Weaponization
- Delivery
- Exploitation
- Installation
- Command and Control (C2)
- Actions on Objectives

Although you can probably tell from the name of each of these phases what they encompass, let me quickly summarize them for you. Note that this is based on my own interpretation of Lockheed Martin's paper, and other interpretations are possible.

Attackers select their target in the **Reconnaissance** phase (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.) Certainly, many attackers select targets opportunistically, many times by coincidence, as evidenced by all the commodity malware present on the internet.

Other attackers might spend time and effort researching who they should target based on their motivations for the attack. They will likely spend time in this phase discovering the IP address space their target uses, the hardware and software they use, the types of systems they have, how the business or organization works, which vendors are in their supply chain, and who works there. They can use a range of tools to conduct this research, including technical tools to do DNS name lookups, IP address range scans, the websites where the organization advertises job openings that typically include technical qualifications based on the hardware and software they use, among many others. In the case of a mass malware attack, the attacker has decided to attack everyone. However, they still need to make that decision, and it occurs in this stage of an attack.

Once attackers have selected their target and have some understanding of where they are on the internet and the technologies they use, then they figure out how they are going to attack the victim. This phase is called **Weaponization** (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.). For example, based on their research on the target, they see that they use Adobe products. So, they plan to try to initially compromise the environment by exploiting potentially unpatched vulnerabilities for Acrobat Reader, for example. To do this, they construct a malformed .pdf file that is capable of exploiting a particular vulnerability (CVE ID) when a victim opens it. Of course, this attack will only work if the vulnerability they are using has not been patched in the target's environment.

Now that the attackers have decided how they are going to try to initially compromise the target's environment, and they've built a weapon to do this. Next, they have to decide how they will deliver their weapon to the target. In the **Delivery** phase, they decide if they are going to send the malformed .pdf file as an email attachment, use it as part of a watering hole attack, put it on USB drives and throw it into the organization's parking lot, and so on.

Once the weapon has been delivered to a potential victim, attackers need a way to activate the weapon. In our malicious .pdf example, the attacker hopes that the victim tries to open the malformed file so that their exploit runs on the victim's system. This phase is aptly called the **Exploitation** phase (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.). If the victim's system isn't patched for the specific vulnerability that the exploit is designed to take advantage of, then the attacker's exploit will successfully execute.

When the attackers exploit executes, it could download more malware to the victim's system or unpack it from within itself. Typically, this will give the attacker remote access to the victim's system. This phase is called **Installation** (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.).

Once the attackers have successfully installed their tools on the victim's system, they can send commands to their tools or to the system itself. The attackers now control the system fully or partially, and they can potentially run arbitrary code of their choice on the victim's system. This phase of the attack is the **Command and Control** (**C2**) phase (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.). They might try to further penetrate the environment by attempting to compromise more systems.

Actions on Objectives is the final phase of the Intrusion Kill Chain. Now that attackers control one or more compromised systems, they pursue their objectives. As I discussed in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*, their motivations could include profit, economic espionage, military espionage, notoriety, revenge, and many others. Now, they are in a position to achieve the specific objectives to satisfy their motivation. They might steal intellectual property, stay persistent to collect information, attempt a kinetic attack to physically damage their victim's operations, destroy systems, and so on.

Note that I have written that these are phases in an attack that attackers *can* use in attacks. I didn't write that each of the phases is *always* used in attacks. This is a nuance that some of the detractors of this framework typically miss. They often argue that attackers don't have to use all seven phases that are listed in Lockheed Martin's paper. They only use the phases they have to use. Therefore, the model is flawed. I will admit that I have never understood this argument, but I hear it often when discussing this framework. This argument has some flaws. It's helpful to keep the intended purpose of this framework in mind—to make it harder for attackers to succeed. Also, remember the tip I gave you in *Chapter 3*, *The Evolution of the Threat Landscape – Malware* about claims of omniscience? This argument relies on omniscience. We will never know what *all* attackers do. This leads to the second flaw in this argument. Because we don't know what attackers will do in the present or the future, we must be prepared to protect, detect, and respond to whatever they chose to do. That is, we need to be grounded in the reality that attackers *can* use *any* of these phases.

For example, some environments are already compromised, making it easier for attackers in the present and potentially in the future to penetrate the victim's environment without going through the first three or four phases. That doesn't mean that an attacker didn't already successfully go through these phases in a previous attack, and it doesn't mean attackers won't use them in the future. We don't know what the future holds and we don't control attackers. We aren't omniscient or omnipotent. We do know attackers will always use at least one of these phases—they have to. Subsequently, defenders must be prepared, regardless of which phases attackers use.

Knowing what the attacker's Intrusion Kill Chain looks like can help defenders make it much harder for attackers to be successful. By significantly increasing the effort required for attackers to be successful, we reduce their return on investment, and potentially their determination. To do this, the authors of the Intrusion Kill Chain paper suggest that defenders use a **Courses of Action Matrix** (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.). This matrix allows defenders to map out how they plan to detect, deny, disrupt, degrade, deceive, and destroy the attacker's efforts in each of the seven phases of their Intrusion Kill Chain. An example of this is illustrated in *Table 6.1*:

Kill Chain Phase	Detect	Deny	Disrupt	Degrade	Deceive	Destroy
Reconnaissance						
Weaponization						
Delivery						
Exploitation						
Installation						
Command and Control (C2)						
Actions on Objectives						

Table 6.1: A Courses of Action Matrix (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.)

By layering controls into this matrix, the objective is to make it much harder, or impossible, for attackers to progress through their Intrusion Kill Chain. Multiple complementary capabilities can be included in each of the cells in the matrix. Stopping attackers as early in their Intrusion Kill Chain as possible reduces the potential damage and associated recovery time and costs. Instead of attackers being successful after they defeat a firewall or a single set of controls, they must overcome the layered defenses in the Courses of Action Matrix for each step in their attack.

Modernizing the kill chain

One consideration before implementing this framework is whether defenders should use the original Intrusion Kill Chain framework or update it. There are several ways this framework can be modernized. I'll give you some ideas on how this can be done in this section. However, don't be afraid to embrace the notion of iterative improvement based on your organizations' experiences with this framework or others.

Mapping the cybersecurity usual suspects

In *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*, I introduced the cybersecurity usual suspects and have referred to them throughout this book. I hope I have imparted the importance of mitigating the five ways that organizations are initially compromised. The Intrusion Kill Chain framework can be modified or reorganized around the cybersecurity usual suspects to ensure that they are mitigated and make it easier to identify gaps in an organization's security posture. This can be done in a couple of different ways. First, they can be integrated into the traditional Kill Chain framework. That is, the controls used to mitigate the cybersecurity usual suspects are spread across the Courses of Action Matrix like all the other controls are. The challenge with this approach is that it can make it difficult to identify over-investment, under-investment, and gaps in those specific areas, especially if your matrix is large. To compensate for this, a column could be added to the matrix where the cybersecurity usual suspect that each control mitigates is tracked. Some rows won't have an entry in this column because many controls will be advanced cybersecurity capabilities, not necessarily focused on the cybersecurity usual suspects.

Another way to make it easier to ensure the cybersecurity usual suspects are fully mitigated is to use two separate lists. Inventory the controls that mitigate the cybersecurity usual suspects in one list and everything else in a separate Courses of Action Matrix. This way, you'll have complete, unobscured visibility into the controls implemented to mitigate the cybersecurity usual suspects, and all other controls as well. This might mean that there is some duplication of controls in these lists that make it more complicated to track changes over time.

I prefer the second approach, that is, using two separate lists. I like clear visibility into the controls that mitigate the cybersecurity usual suspects. This approach makes it easier to keep track of the controls that represent the foundation of the strategy. However, feel free to use either approach or a different one that works best for your organization. This is the approach I will use in the example provided in this chapter. I've already discussed the cybersecurity usual suspects and the cybersecurity fundamentals extensively in other chapters. The example I'll provide here will focus on the *advanced cybersecurity capabilities* component of the strategy.

Updating the matrix

Another modification to this approach worth considering is whether to update the phases and the actions in the Courses of Action Matrix. For example, the Reconnaissance phase of the Intrusion Kill Chain can be split into two separate phases. This separation recognizes that there are potentially two different times in an intrusion attempt that attackers typically perform reconnaissance. Prior to the attack, attackers might spend time selecting their target and researching ways that they could be attacked. After one of the cybersecurity usual suspects is used to initially compromise the victim, then the attackers might perform some reconnaissance again to map out the victim's network and where the **High-Value Assets** (**HVAs**) are. The reason why separating these two phases can be helpful is that the tools, techniques, tactics, and procedures used by attackers can be different before and after initial compromise. Updating the matrix by replacing the **Reconnaissance** phase with the **Reconnaissance I** and **Reconnaissance II** phases will enable security teams to map different controls to stop attackers in each of these phases. Keep in mind that, in both of these cases, attackers might us non-intrusive reconnaissance tactics or choose to use intrusive reconnaissance tactics.

Another potential update to the phases is dropping the **Weaponization** phase. That might seem like a significant change to the original framework, but in my experience, it doesn't change the controls defenders typically use. This phase of an attack is where the attackers, who have now decided how they are going to attack the victim, plan to reuse old weapons or build and/or buy new weapons to use in their attack.

Most of this activity happens out of the view of defenders. Subsequently, very few of the attacker's activities in this phase can be influenced by controls available to defenders. If attackers are cavalier about the sources they procure weapons from, threat intelligence vendors or law enforcement could get tipped off about their activities and perhaps their intentions. This could be helpful if the weapon is a zero-day vulnerability that the intended victim could deploy workarounds to mitigate, but frankly, focusing on the other attack phases will likely have a much higher return on investment for defenders as they potentially have more visibility and control. The Weaponization phase is too opaque for most organizations to realistically influence. Put another way, CISOs typically do not have very effective controls for protection and detection prior to the Delivery phase; prioritizing investments in mitigations that have a clear, measurable value is important.

The Courses of Action Matrix can be updated to include some different actions. For example, **Destroy** could be dropped in favor of some more realistic actions, such as **Limit** and **Recover**. Using Limit as an action recognizes that defenders want to make it hard or impossible for attackers to move freely during their attack. For example, limiting the delivery options available to attackers, and limiting the scope of the infrastructure that attackers can control, both make it harder for attackers to be successful. Using a Restore action helps organizations plan their recovery if all the other mitigations layered in the model fail to perform as expected. For both Limit and Restore, not every cell in the matrix will necessarily have controls in them. For example, there likely is no control that will help Recover during the Reconnaissance I phase because the environment hasn't been attacked yet. There will potentially be several cells in the matrix without entries—this is to be expected. An example of the updated Courses of Action Matrix is illustrated in *Table 6.2*:

Kill Chain Phase	Detect	Deny	Disrupt	Degrade	Deceive	Limit	Restore
Reconnaissance I							
Delivery							
Exploitation							
Installation							
Command and Control (C2)							
Reconnaissance II							
Actions on Objectives							

Table 6.2: An example of an updated Courses of Action Matrix (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.)

Of course, these updates are completely optional. Implementing the original Intrusion Kill Chain model can be an effective way for many organizations to improve their security posture. I suggest that before CISOs get serious about implementing this model, they spend some time thinking through whether any modifications to the original model will be advantageous. Then, they should update the Courses of Action Matrix before moving forward with this model as this will save time, expense, and potentially frustrating rework.

Getting started

Existing IT environments, especially those already under the management of a CISO, will likely have some cybersecurity controls already deployed in them. If an Attack-Centric Strategy and the Intrusion Kill Chain approach is new to an organization, chances are, that the existing controls were deployed in a way that isn't necessarily consistent with the Courses of Action Matrix. Mapping currently deploying cybersecurity controls to the Courses of Action Matrix will help determine where potential gaps exist between currently deployed cybersecurity capabilities and a fully implemented Courses of Action Matrix. It can also help identify areas of over-investment and under-investment. For example, after mapping their current cybersecurity capabilities to this matrix, the security team realizes that they have invested heavily in capabilities that deny the delivery of the attacker's weapons, but have not invested anything that helps detect delivery attempts; in fact, they now realize they have under-invested in detection capabilities across the entire Kill Chain. This mapping exercise can help expose optimistic assumptions about organizations' security capabilities. Some security professionals call this type of exercise **cartography**.

This exercise can be illuminating, but also challenging to perform, especially in large complex environments. Most organizations I've advised didn't have a complete, up-to-date list of products, services, and configurations that are useful in an exercise like this one. Even organizations that have change control management systems often find their data to be incomplete, out of date, and inaccurate. I've seen industry estimates suggesting that the average on-premises IT environment has 20% undocumented assets and services, and even higher estimates in some industries, like healthcare.

Some organizations try to use procurement artifacts to determine what their IT department bought, but this is usually different than what was actually deployed. Faced with the challenge of getting an accurate, up-to-date inventory of the cybersecurity capabilities they have running in their environment, most organizations start with the data they have, and manually verify what has been implemented. This isn't necessarily a bad thing because it can provide a view that is accurate and current, but also includes qualitative insights that can't be rendered from an inventory database.

Maturity of current cybersecurity capabilities

I have had the opportunity to do this mapping exercise in some large, complicated IT environments. Let me share some of the things I've learned, to save you some time if you are faced with the same challenge. As you map current cybersecurity capabilities to the Courses of Action Matrix, one factor to be aware of is the maturity of the implementation of each control or capability. That is, an item on a software inventory list might not offer any clue as to whether the control is fully implemented or partially implemented. Understanding the maturity of each control's implementation is key to really understanding where gaps exist, and where under and over-investment has occurred.

For example, an organization procures a suite of cybersecurity capabilities from a top tier industry vendor. This suite is capable of providing several important functions including file integrity monitoring, anti-malware scanning, and data loss prevention for desktops and servers. When mapping capabilities to the Courses of Action Matrix, it is easy to look at the capabilities the suite *can* provide and include all of them in the inventory of the organization's current capabilities. However, the question is, how many of the suite's capabilities have *actually* been deployed? A related question is, who is responsible for operating and maintaining these controls? These can be difficult questions to answer in large, complicated IT environments. However, without uncovering the truth about the maturity of the current implementation, the confidence of the mapping and the potential efficacy of the strategy can be undermined. Remember the submarine analogy I've used throughout this book; would you really be keen to set sail in a submarine if you didn't really know if all the critical systems were fully operational? Probably not.

Many organizations aspire to have a world-class cybersecurity team. To support this aspiration, a principle some of them use when evaluating and procuring cybersecurity capabilities is that they only want best of breed technologies. That is, they only want the best products and won't settle for less than that. For most organizations, this is highly ambitious because attracting and retaining cybersecurity talent is a challenge for the entire industry. Adopting a "best of breed" procurement philosophy makes this acute talent challenge even harder. This is because it potentially narrows the number of people that have experience with these expensive and relatively rare "best of breed only" implementations. This approach can also be dangerous for organizations that are cash rich and believe they can simply buy effective cybersecurity instead of developing a culture where everyone participates. Most of the organizations that I've seen with this philosophy end up buying a Ferrari and using its ashtray. They simply do not have the wherewithal to architect, deploy, operate, and maintain only the best of breed, so they only use a fraction of the available capabilities. In some cases, organizations that find themselves in this scenario over-invest in an area by procuring the same or similar capabilities, but they do this by procuring products they can successfully deploy and operate. Performing this mapping exercise in organizations that have found themselves in this scenario can be especially hard. This is because it uncovers hard truths about overly optimistic ambitions and assumptions, as well as cybersecurity investments with marginal returns. However, this process can be a necessary evil for organizations with the courage to look in the mirror and the willingness to make positive, incremental changes to their current security posture. There's nothing wrong with being ambitious and aiming high if those ambitions are realistically attainable by the organization.

It can be challenging to quantify how much of a cybersecurity suite or set of capabilities has been successfully deployed. One approach I've tried, with mixed results, is to break out the functionality of the set of capabilities into its constituent categories and use a maturity index to quantify how mature the deployment is using a scale between 1 and 5, where 5 is most mature. This can help determine whether more investment is required in a particular area. In large, complex environments, this is easier said than done, and some might wonder if it's worth the time and effort as they struggle through it. However, the more detail security teams have about the current state of their affairs, the more confidence they'll have moving forward with this strategy.

Who consumes the data?

One principle I have found helpful in mapping the current security capabilities of an IT environment to the Courses of Action Matrix is that the data generated by every control set needs someone or something to consume it. For example, a security team performing this mapping discovers that the network management team implemented potentially powerful IDS/IPS capabilities that were included with a network appliance they procured last fiscal year. Although these capabilities are enabled, they discover that no one in the network management team is actively monitoring or reviewing alerts from this system and that the organization's **Security Operations Center** (**SOC**) wasn't even aware that they existed. The net result of these capabilities is equivalent to not having them at all, since no one is consuming the data they generate. A human doesn't necessarily have to consume this data; orchestration and automation systems can also take actions based on such data. However, if neither a human nor a system is consuming this data, then security teams can't really include these capabilities in their mappings of currently implemented controls, unless those deficiencies are addressed.

As security teams perform this mapping, for each control they identify, they should also record who or what consumes the data it generates. Recording the name of the person that is the point of contact for the consumption of this data will pay dividends to security teams. A point of contact might be a manager in the SOC or in the **Network Operations Center** (**NOC**), an Incident Response team member or a vendor. This information is valuable in building confidence in the organization's true cybersecurity capabilities. However, it is also very valuable in measuring the efficacy of your strategy, which I will discuss in detail in *Chapter 7*, *Measuring Performance and Effectiveness*:

Kill Chain Phase	Detect	Detect Maturity Index (1-5)	Detect Data Consumer	Detect Consumer PoC	Deny	Deny Maturity Index (1-5)	Deny Data Consumer	Deny Consumer PoC
Reconnaissance I								
Delivery								
Exploitation								
Installation								
Command and Control (C2)								
Reconnaissance II								
Actions on Objectives								

Table 6.3: An example of a partial Courses of Action Matrix (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.), which includes a maturity index, who or what is going to consume data from each control, and a point of contact (PoC)

As shown in *Table 6.3*, as the Courses of Action Matrix is updated, it expands quickly. I have used a spreadsheet to do this mapping in the past. I'll admit that this isn't the most elegant way to perform such mappings. One mapping I did was over 120 pages of controls in a spreadsheet; navigating that spreadsheet wasn't much fun. Additionally, using a spreadsheet is not the most scalable tool and reporting capabilities are limited. If you have a better tool, use it! If you don't have a better tool, rest assured that the mapping exercise can be done using a spreadsheet or a document. However, the bigger and more complex the environment is, the more challenging using these tools becomes.

Cybersecurity license renewals

Most software and hardware that is procured from vendors have licensing terms that include a date when the licenses expire. When a license expires, it must be renewed or the product must be decommissioned. Another update to the Courses of Action Matrix to consider, which can be very helpful, is to add a column to track the contract renewal date for each capability listed. If you are taking the time to inventory the software and hardware used for cybersecurity, also record the expiry/renewal date for each item. This will give you an idea of the time each item on the list has before its license expires and renewal is required. Embedding this information into the control mapping itself will give you visibility of the potential remaining lifetime for each capability and can help remind the security team when to start reevaluating each product's effectiveness and whether to renew or replace existing capabilities.

Another similar date that can be helpful to track is the end of life/support dates for products; typically, after this date, manufactures deprecate products and no longer offer security updates for them. Over time, these products increase the attack surface in IT environments as vulnerabilities in them continue to be disclosed publicly, even after their end of support dates. Tracking these dates can help us avoid surprises. Tracking these dates as part of a modified Courses of Action Matrix is optional.

CISOs and security teams shouldn't rely on their Procurement departments to flag renewal dates for them; it should work the other way around. Many of the CISOs I've talked to want to have visibility into this "horizon list," how it impacts their budgets, and key milestone dates when decisions need to be made. What CISO wouldn't want some advanced notice that their network IDS/IPS was going to be turned off because their license was about to lapse? The more lead time these decisions have, the fewer last-minute surprises security teams will have. Additionally, when I discuss measuring the efficacy of this strategy in the next chapter, you'll see that having this information at your fingertips can be helpful.

Of course, this update to the matrix is optional. Renewal dates can be tracked in a separate document or database. However, being able to cross reference the renewal dates and the cybersecurity capabilities in your mapping should be something CISOs can do easily. They need to have sufficient lead time to determine whether they want to keep the products and services they already have in production or replace them:

Kill Chain Phase	Detect	Detect Maturity Index (1-5)	Detect Data Consumer	Detect Consumer PoC	Detect Renewal Date	Deny	Deny Maturity Index (1-5)	Deny Data Consumer	Deny Consumer PoC	Deny Renewal Date
Reconnaissance I										
Delivery										
Exploitation										
Installation										
Command and Control (C2)										
Reconnaissance II										
Actions on Objectives										

Table 6.4: An example of a partial Courses of Action Matrix (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.), which includes a maturity index, who or what is going to consume data, a point of contact (PoC), and the renewal date for each control

Implementing this strategy

By the end of the mapping process, CISOs and security teams should have a much better inventory of the cybersecurity capabilities and controls that have been deployed, as well as how the data from these are being consumed by the organization. This is a great starting point for implementing the Intrusion Kill Chain framework. However, do not underestimate how challenging it can be for organizations with large, complex IT environments to accomplish this.

For some organizations, it will be easier to divide mapping work into smaller, more achievable projects focused on parts of their environment, than trying to map their entire environment. Moving forward with this strategy without an accurate, current mapping can easily lead to over-investments, under-investments, and gaps in security capabilities. Although these can be corrected over time, it will likely make it more expensive and time-consuming that it needs to be.

Kill Chain Phase	Detect	Detect Description	Detect Maturity Index (1-5)	Detect Data Consumer	Detect Consumer PoC	Detect Renewal Date	Deny	Deny Description	Deny Maturity Index (1-5)	Deny Data Consumer	Deny Consumer PoC	Deny Renewal Date
Reconnaissance I												
Delivery	control_name	Network IDS/IPS	3	SOC	Leonard Nimoy	09/15/24	control_name	email spam filter	2	eMail Administrator	Will Smith	01/01/24
Delivery	control_name	System image agent	1	IR Team	Bruce Dern	06/03/22	control_name	USB disable policy	2	Server Support	Bruno Mars	n/a
Exploitation	control_name	Windows Error Reporting	1	No one	No one	n/a	control_name	Anti-virus suite	4	SOC	Leonard Nimoy	04/15/21
Exploitation	control_name	System Integrity Monitor	2	SOC	Leonard Nimoy	09/09/22	control_name	File Integrity Monitoring	2	SOC	Leonard Nimoy	06/15/22
Exploitation	control_name	Anti-virus suite	4	Desktop Support	Kathy Bates	04/15/21	control_name	ASLR	5	Desktop Support	Kathy Bates	n/a
Installation	control_name	File Integrity Monitoring	2	Desktop Support	Kathy Bates	06/15/22	control_name	Anti-virus suite	4	Desktop Support	Kathy Bates	04/15/21
Installation	control_name	File Integrity Monitoring	2	Server Support	Bruno Mars	06/15/22	control_name	Anti-virus suite	4	Server Support	Bruno Mars	04/15/21
Installation	control_name	Anti-virus suite	4	SOC	Leonard Nimoy	04/15/21						
Installation	control_name	Anti-virus suite	4	IR Team	Bruce Dern	04/15/21						
Command and Control (C2)	control_name	Network IDS/IPS	3	NOC	Keanu Reeves	09/15/24	control_name	Network IDS/IPS	3	NOC	Keanu Reeves	09/15/24
Command and Control (C2)	control_name	Network IDS/IPS	3	SOC	Leonard Nimoy	09/15/24						
Reconnaissance II	control_name	Network IDS/IPS	3	NOC	Keanu Reeves	09/15/24	control_name	Network IDS/IPS	3	NOC	Keanu Reeves	09/15/24
Reconnaissance II	control_name	Network IDS/IPS	3	SOC	Leonard Nimoy	09/15/24						
Actions on Objectives	control_name	File Integrity Monitoring	2	Server Support	Bruno Mars	06/15/22	control_name	File Integrity Monitoring	2	Server Support	Bruno Mars	06/15/22
Actions on Objectives	control_name	File Integrity Monitoring	2	Desktop Support	Kathy Bates	06/15/22	control_name	File Integrity Monitoring	2	Desktop Support	Kathy Bates	06/15/22
Actions on Objectives	control_name	Deception technology	2	SOC	Leonard Nimoy	05/31/25						

Table 6.5: An example of a part of an updated Courses of Action Matrix (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.) that contains a mapping of an organization's current cybersecurity capabilities

I've provided an example of what the first two actions across an updated set of phases looks like in Table 6.5. An actual mapping for a large organization could potentially be much larger, but I want to give you an idea of what a mapping looks like. In an actual mapping, **control_name** will be the names of the specific products, services, features, or functionality that detect, deny, disrupt, and so on for each phase of an attack. The **Description** field is meant to be a short description of what each control does. I suggest providing more detail in this field than I have here so that it's clear what each control's function and scope is.

There is a **Maturity Index** for each control, ranging from 1 to 5, with 5 indicating that the implementation is as full and functional as possible. A maturity index of one or two indicates that while the product or feature has a lot of functionality, relatively little of it has been deployed or is being operated. This index will help inform our assumptions about how effective each control currently is versus its potential. This helps avoid the trap of assuming a control is operating at peak efficiency when in reality it isn't fully deployed, or it's not being actively operated or monitored. Color-coding this field or entire line items based on this field, can make it even easier to understand the maturity of each control.

The **Data Consumer** for each action is the specific group or department in the organization that is using data from the control to detect, deny, disrupt, degrade, and so on. The **Consumer PoC** column contains the names of each point of contact in the group or department that is consuming the data from each control. This can make it easier to periodically verify that the data from each control is still being consumed as planned. After all, there's no point deploying mitigations if no one is actually paying any attention to them. The time, effort, and budget spent on such controls can likely be used more effectively somewhere else in the organization.

Finally, the **Renewal Date** column for each action provides visibility into the potential expiry date of each control. It does this to help minimize potential unexpected lapses in the operational status of each control. This helps avoid the revelation that a mitigation you thought was fully operational has actually been partially, or entirely disabled because of a lapse in licensing or a product going out of support; these surprises can burn CISOs and security teams.

Rationalizing the matrix – gaps, under-investments, and over-investments

Without a mapping of current cybersecurity capabilities to the Courses of Action Matrix, it can be very easy to over-invest or under-invest in cybersecurity products and have gaps in protection, detection, and response capabilities. What exactly do I mean by over-investments, under-investments, and gaps? Performing a mapping of existing cybersecurity capabilities and controls to an Intrusion Kill Chain framework can be a lot of work. However, for some CISOs, it can result in an epiphany. Performed correctly, this mapping can reveal key areas where organizations haven't invested at all—a gap. For example, in *Table 6.5*, the **Reconnaissance I** row doesn't have any entries in it; this can be a clear indication that the organization has a gap in their control set, which could make this phase of the attacker's Intrusion Kill Chain easier for them. It isn't uncommon for organizations to fail to invest in this area. A gap like this is a clear opportunity for improvement.

Under-investments in an area can be more subtle in the Courses of Action Matrix. An under-investment can appear as a relatively small number of entries for an activity or phase in the matrix. This where the maturity index and description can help.

A single entry in the matrix with a maturity index of 5 might be all the investment that is needed for that action. The combination of the maturity index and the description should help make this determination. However, the entry's description should be verbose enough for us to understand if the functionality and scope of the capability will really break the attacker's Kill Chain or if more investment is warranted in that area of the matrix. The right control might be deployed; but if it's only partially implemented or partially operational, it might not be sufficient to break a Kill Chain or be effective in all scenarios. Further investment into maturing that control might be the solution to this problem. Another possible solution is investing in a different control to supplement the current mitigation. From this perspective, the Courses of Action Matrix becomes an important document to help during an incident and is the center of negotiations over budgets and resources with non-technical executives.

Over-investing in areas is a common problem that I've seen both public and private sector organizations suffer from. It can occur slowly over time or quickly in the wake of a data breach. In the Courses of Action Matrix, it can appear as a lot of entries in one or two areas that perform the same or similar functions. For example, I've seen organizations procure multiple identity and access management products and fully deploy none of them. This can happen for a range of reasons. For example, they may have been unrealistic about their ability to attract and retain the talent required to deploy these products. Another example is that in the wake of a successful intrusion, it's not uncommon for a victimized organization to decide that it's time to make a big investment in cybersecurity. With a newfound sense of urgency and exuberance, they don't take the time to get an inventory of current capabilities and their maturity before they go on a shopping spree. Mergers and acquisitions can also leave organizations with over-investments in some areas of the matrix. Finally, simply put, some salespeople are really good at their jobs. I've seen entire industries and geographical areas where everyone has literally procured the same SIEM or endpoint solution during the same 1 or 2 fiscal years. There's nothing wrong with this, but it's unlikely they all started with the same environments, comparable cybersecurity talent, and with the same licensing renewal dates for their current products. When a good salesperson is exceedingly successful, this can sometimes lead to over-investments in areas.

Planning your implementation

It's important to identify gaps, under-invested areas, and areas of over-investment as these will inform the implementation plan. Hopefully, many of the areas that the organization has already invested in won't require changes. This will allow them to focus on addressing gaps and shortcomings in their current security posture. At the point where they have a current mapping and have identified gaps, areas of under-investment, and areas of over-investment, they can start planning the rest of their implementation.

What part of the Courses of Action Matrix should security teams work on first? For some organizations, focusing on addressing existing gaps will offer the highest potential ROI. However, there are some factors to consider, including the availability of budgets and cybersecurity talent. The over-arching goal is to break the attacker's Kill Chains. However, remember that there are some efficiencies to doing this as early in the Kill Chain as possible. Stopping an attack before exploitation and installation can help minimize costs and damage. However, as I discussed in regard to the Protect and Recover Strategy, the assumption that security teams will be able to do this 100% of the time is overly optimistic and will likely set the organization up for failure. Subsequently, some of the CISOs I've discussed this with decided to invest a little bit in every part of the matrix. However, sufficient budget and resource availability can be limiting factors for this approach.

Most CISOs I've talked to have limited budgets. For those that don't, they are typically still limited by their ability to architect, deploy, and operate new capabilities quickly; the cybersecurity talent shortage is industry-wide. The renewal date for each item in the matrix can help inform a timeline used to address gaps and investment issues. Choosing not to renew licenses for less effective products in areas of over-investment might help free up some of the budget that can be used to address gaps and areas of under-investment. Not every organization has over-investments, and many are chronically under-invested across the matrix. For organizations in this category, taking advantage of as many of the "free" controls in operating systems and integrated development environments as possible can be helpful.

For example, **Address Space Layout Randomization** (**ASLR**) and **Data Execution Prevention** (**DEP**) can help make the Exploitation phase of an attack harder to accomplish and inconsistent. These features are built into most modern operating systems from major vendors today. However, not all applications take advantage of them. Thoughtfully using such free or low-cost controls can help organizations with limited budgets pursue this strategy.

Another way I've seen CISOs plan their implementation is to use results from Red Team and Blue Team exercises and penetration tests. Penetration tests typically focus on confirming the effectiveness of security controls that have been implemented, where the Red team exercises focus on outrunning and outsmarting defenders. This is a direct way of testing the effectiveness of the people, processes, and technologies that are part of your current implementation. Just as important as identifying gaps, these exercises can identify controls and mitigations that are not performing as expected. These exercises can also help inform the maturity indexes in your mapping and help prioritize items in your implementation plan in a practical, less theoretical way.

Finally, one other way I've seen CISOs decide to implement frameworks like this one is to invest in high ROI areas first. They do this by identifying where they get the biggest bang for their investment. This is done by identifying controls that provide mitigations in multiple parts of the matrix. For example, if the same control potentially helps break the attacker's Kill Chains in the Delivery, Exploitation, and Command and Control phases, they'll prioritize that one over controls that only potentially break one phase of an attack (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.). Put another way, they look for areas where they can use two or three mitigations for the price of one. The more detailed their matrix is, the more of these opportunities they can identify.

I will revisit many of the factors that I discussed in this section in *Chapter 7*, *Measuring Performance and Effectiveness*.

Designing control sets

With a current control set mapping, identified gaps, areas of under-investment, areas of over-investment, and a plan for which of these areas will be addressed, security teams can start designing control sets. This part of the process can be challenging, but a lot of fun as well.

After all, designing controls to make it as hard as possible for attackers to succeed is fun! For some people, spending money is fun too, and there is an opportunity to lay the groundwork to do that in this exercise.

There are more combinations and permutations of possible control sets than I can cover in this book. This section is meant to provide you with more detail on each part of the updated Courses of Action Matrix that I outlined, and provoke some thought about ways that security teams could design control sets for their organization. This isn't a blueprint that should be followed; it's really just a high-level example. I didn't receive any promotional payments for any products or companies I mentioned in this section and I don't endorse them or make any claims or warranties about them or their products. Please use whatever companies, products, services, and features meet your requirements. If you'd like professional recommendations, I recommend consuming the reports and services of industry analyst firms such as Forrester and Gartner, among others. This is where CISO councils, professional societies, and gated social networks can be very helpful. Getting first-hand accounts of the efficacy of strategies, products, and services directly from other CISOs can be very helpful. Analyst firms can't be too publicly critical of a company or its products, but I haven't met very many CISOs that weren't willing to be candid in private conversations behind closed doors.

Attack phase – Reconnaissance I

In this phase of an attack, attackers are selecting their targets, performing research, mapping and probing their target's online presence, and doing the same for the organizations in their intended victim's supply chain. Attackers are seeking answers to the basic questions of what, why, when, and how. Their research isn't limited to IP addresses and open TCP/UDP ports; people, processes, and technologies are all potential pawns in their attacks.

The challenge for defenders in this stage of the attack is that these types of reconnaissance activities blend in with legitimate network traffic, emails, telephone calls, employees, and so on. It can be very difficult to identify the attacker's reconnaissance activities when they aren't anomalous. Still, it can be worthwhile to invest in cybersecurity capabilities in this stage because, as I mentioned earlier, breaking the attacker's Kill Chains as early as possible typically has the highest return on investment.

Categorizing reconnaissance activities into passive and active groups (H. P. Sanghvi, 2013) can help security teams decide where investments are practical. For example, it might be prohibitively expensive to try to identify attackers performing passive research by reading an organization's job postings website, just to identify the types of hardware and software it uses. However, it might be practical to detect and block IP addresses of systems that are actively scanning for vulnerabilities on corporate firewalls. Many passive reconnaissance activities can be conducted out of the sight of defenders and subsequently won't generate log entries or alerts that defenders can use. However, many threat intelligence vendors offer services to their customers that scrape social media sites and illicit marketplaces, all to look for chatter in the dark web about their IP address ranges, domains, known vulnerabilities, credentials for sale, and imminent attacks. Active reconnaissance activities tend to interact directly with the victims and their supply chains, potentially providing defenders with a more direct glimpse of them.

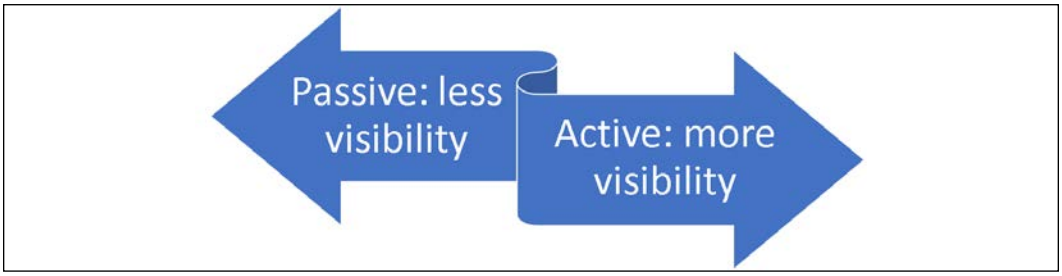


Figure 6.1: Reconnaissance activity categories

Some cybersecurity capabilities that can help in this phase of an attack include:

- Threat intelligence services can help detect passive reconnaissance activities, potentially giving defenders notice of known vulnerabilities in their defensive posture and imminent attacks. Ideally, this can give them some time to address these known vulnerabilities and better prepare themselves for an attack. Some examples of threat intelligence vendors that currently offer such services include:

    - Digital Shadows
    - FireEye
    - Kroll
    - MarkMonitor
    - Proofpoint
    - Many, many others, including smaller, boutique firms

- **Web Application Firewalls** (**WAF**) can detect application layer attacks like SQL injection, cross-site scripting, and so on. A WAF can help detect, deny, disrupt, and degrade application layer attacks. Some examples of WAFs include:
    - Amazon Web Services
    - Barracuda
    - Cloudflare
    - F5
    - Imperva
    - Microsoft
    - Oracle
    - Many, many others

- There are at least a few different flavors of firewalls. Firewalls can detect, deny, disrupt, and degrade some active network reconnaissance activities. There are too many examples of vendors that offer firewall products to list, but some examples include:
    - Barracuda
    - Cisco
    - Check Point Software Technologies
    - Juniper Networks
    - Palo Alto Networks
    - SonicWall
    - Many, many others

- Deception technologies can be employed to deceive attackers performing active reconnaissance. Deception technology systems present systems as the legitimate infrastructure of the intended target or vendors in their supply chain. Attackers spend time and resources performing reconnaissance on these systems instead of production infrastructure and systems. Examples of deception technology vendors include:
    - Attivo Networks
    - Illusive Networks
    - PacketViper
    - TrapX Security
    - Many, many others
- Automation can be combined with threat intelligence and detection capabilities to enable dynamic responses to reconnaissance activities. For example, if a WAF or firewall detects probes from known malicious IP addresses, automation could be triggered to dynamically adjust the lists of blocked IP addresses for some period of time, or automation could try to degrade reconnaissance and waste the attacker's time by allowing ICMP network traffic from malicious IP addresses, blocking TCP traffic to ports 80, 443, and other open ports. This would allow attackers to see systems were online, but not connect to services running on them. This type of automation might be harder to accomplish in legacy on-premises environments, but it's baked into the cloud by default and relatively easy to configure. I'll discuss cloud capabilities in more detail in *Chapter 8*, *The Cloud – A Modern Approach to Security and Compliance*.

This is what the Courses of Action Matrix for the Reconnaissance I phase looks like based on the capabilities I discussed in this section. Of course, this is just scratching the surface of what's possible in this phase, but it provides you with some ideas of what some actions might look like for this first stage in an attack. You'll notice that I didn't include any entries for the restore action.

Since reconnaissance typically doesn't result in damage or compromise, there's nothing in this stage of an attack to recover.

As I mentioned, creating a Courses of Action Matrix using Excel isn't ideal, but it works. However, the tables this exercise creates are too large to print here, in a book, and still be readable. Subsequently, I'm going to provide lists of controls for each section of an example matrix. I don't include controls for phases, like Restore for example, unless there are items in it. To simplify things further, I don't include any of the modifications I discussed earlier because they are unique to each organization. This list isn't meant to be exhaustive; it provides examples of basic controls that you can use as a starting point to develop your own Courses of Action Matrix. Some of the items are repeated multiple times in the Courses of Action Matrix because those controls can perform multiple roles in the matrix.

The following controls are example controls that can be used to **Detect** attacker activities in the **Reconnaissance I** phase of an attack:

- **Deception Technology**: Can help detect the attacker's reconnaissance activities.
- **Web Application Firewall (WAF)**: Can detect application layer attacks like SQL injection, cross-site scripting, and so on.
- **Firewalls**: Can detect network probes and some recon activity.
- **Threat intelligence reconnaissance services**: Can help detect passive reconnaissance activities, giving defenders notice of known vulnerabilities in their defensive posture and imminent attacks.

The following controls are example controls that can be used to **Deny** attacker activities in the **Reconnaissance I** phase of an attack:

- **Automation**: Use automation when reconnaissance activities are detected to adjust firewall rules and other controls in ways that deny, disrupt, degrade, or limit their activities.
- **Web Application Firewall (WAF)**: Can block application layer attacks like SQL Injection, cross-site scripting, and so on.
- **Firewalls**: Can block network probes and some recon activity.

The following controls are example controls that can be used to **Disrupt** attacker activities in the **Reconnaissance I** phase of an attack:

- **Automation**: Use automation when reconnaissance activities are detected to adjust firewall rules and other controls in ways that deny, disrupt, degrade, or limit their activities.
- **Web Application Firewall (WAF)**: Can disrupt application layer attacks like SQL injection, cross-site scripting, and so on.
- **Firewalls**: Can disrupt network probes and some recon activity.

The following controls are example controls that can be used to **Degrade** attacker activities in the **Reconnaissance I** phase of an attack:

- **Automation**: Use automation when reconnaissance activities are detected to adjust firewall rules and other controls in ways that deny, disrupt, degrade, or limit their activities.
- **Web Application Firewall (WAF)**: Can degrade application layer attacks like SQL injection, cross-site scripting, and so on.
- **Firewalls**: Can degrade network probes and some recon activity.

The following controls are example controls that can be used to **Deceive** attackers in the **Reconnaissance I** phase of an attack:

- **Deception technology**: Deception technologies can trick attackers into spending time performing recon on fake assets instead of real ones.

The following controls are example controls that can be used to **Limit** attacker activities in the **Reconnaissance I** phase of an attack:

- **Automation**: Use automation when recon activities are detected to adjust firewall rules and other controls in ways that deny, disrupt, degrade, or limit their activities.

One of the most effective methods used to inform investment decisions for the Reconnaissance I phase is for security teams to perform reconnaissance on their own network.

Attack phase – Delivery

At this point in an attack, the attackers have already decided which organization to target, done some research to help them execute their attack, and potentially done some active reconnaissance scanning and probed the intended victim's internet presence. Based on this information, they've also gone through some Weaponization stage or process where they procured and/or built weapons that will help them initially compromise their targets and enable their activities afterwards. This Weaponization process typically happens out of the sight of defenders. However, as I mentioned in the Reconnaissance I phase, some threat intelligence vendors' services can sometimes get an insight into these activities.

The attacker's weapons can include people, processes, and technologies. With all this in hand, attackers must deliver these weapons to their targets; this is the objective of the Delivery phase. Attackers have a range of options to deliver their weapons to their targets and the vendors in their supply chains. Some examples of delivery mechanisms include malicious email attachments, malicious URLs in emails, malicious websites that attract the victims' attention, malicious insiders, self-propagating malware such as worms, leaving malicious USB drives in victims' premises, and many others.

Some investments that can help in this phase of an attack include:

- **Education/training**: Recall the research I provided in *Chapter 3*, *The Evolution of the Threat Landscape – Malware*. It's clear that different types of malware go in and out of vogue with attackers, but their mainstay approach has always been social engineering. Therefore, educating information workers and training them to spot common social engineering attacks can be very helpful in detecting the delivery of the attacker's weapons. The challenge is that social engineering training isn't a one-time activity, it's an ongoing investment. When training stops, current employees start to forget these lessons and new employees don't get trained. Note that the training itself needs to be kept up to date in order to continue being effective.

Some organizations simply don't have a culture that supports social engineering training that includes actual phishing campaigns and other social engineering attacks against employees. However, organizations that don't do this type of training miss the opportunity to let their employees learn from experience and from failure. A culture where everyone tries to help the CISO is much more powerful than those where the security team is always reacting to uninformed, poor trust decisions that untrained information workers will make every day.

- **Microsoft Office 365 Advanced Threat Protection (APT)**: Email is a major vector for social engineering attacks. The volume of email-based attacks is relatively huge in any period of time. Offering information workers email inboxes without effective protection is setting the organization up for failure. Cloud-based services like Microsoft Office 365 APT help inoculate all their users by blocking threats that any of their users get exposed to. Services this large can easily identify the IP addresses that botnets and attackers use for spam, phishing, and other email-based attacks, and block them for all their users.
- **Deception technology**: I'm a big fan of deception technology. This technology goes beyond honeypots and honey-nets, offering full-blown environments that attract attackers, signal their presence, and waste their time, driving down their return on investment. Using deception technology to present vulnerable systems to attackers, systems that are critical infrastructure, or systems that store or have access to potentially valuable data can divert their efforts from legitimate systems.
- **Anti-malware suites**: Anti-malware software can detect and block the attempted delivery of different types of weapons. As I discussed in *Chapter 3*, *The Evolution of the Threat Landscape – Malware*, anti-malware software isn't optional in a world where the number of malicious files easily outnumbers legitimate files. Some of the anti-malware vendors that offer products include:
    - Blackberry Cylance
    - CrowdStrike
    - Carbon Black

    - FireEye
    - F-Secure
    - Kaspersky
    - McAfee
    - Microsoft
    - Trend Micro
    - Many others

- **Web browser protection technologies**: Blocking access to known bad websites and insecure content, as well as scanning content before the browser downloads it, can help prevent exposure to drive-by download attacks, phishing attacks, malware hosting sites and other malicious web-based attacks.
- **File Integrity Monitoring (FIM)**: FIM can help detect, block, disrupt, and degrade the delivery phase by maintaining the integrity of operating system and application files.
- **IDS/IPS**: Several vendors offer IDS/IPS systems including Cisco, FireEye, and others.
- **Operating System Mandatory Access Control**: Can help disrupt and degrade delivery.
- **Short-lived environments**: Systems that only live for a few hours can disrupt and degrade the attacker's ability to deliver their weapons, especially more complicated multi-stage delivery scenarios. The cloud can make leveraging short-lived environments relatively easy; I'll discuss this concept more in *Chapter 8*, *The Cloud – A Modern Approach to Security and Compliance*.
- **Restore**: I've met with many organizations over the years that rely on blocking mechanisms like anti-malware software to detect and block delivery, but will rebuild systems if there is any chance they were compromised. If delivery is successful, even if exploitation and installation is blocked, some organizations want to flatten and rebuild systems or restore data from backups to ensure that everything is in a known good state.

Next, we'll look at what the Courses of Action Matrix for the Delivery phase looks like based on the capabilities I discussed in this section.

The following controls are example controls that can be used to **Detect** attacker activities in the **Delivery** phase of an attack:

- **Education/training**: Information worker education and training to spot social engineering attacks.
- **Microsoft Office 365 Advanced Threat Protection**: Detects and blocks delivery of malicious email and files.
- **Deception Technology**: Deception technologies can attract attackers and detect weapon delivery to deception assets.
- **Anti-malware suites**: Anti-malware can detect and block delivery of malicious content from storage media, the network, and via web browsers.
- **File Integrity Monitoring (FIM)**: FIM can detect and block system file replacement attempts.
- **IDS/IPS**: Can detect and potentially disrupt or stop delivery.

The following controls are example controls that can be used to **Deny** attacker activities in the **Delivery** phase of an attack:

- **USB drive prohibit policy**: Blocking USB and removable media from mounting can prevent delivery.
- **Microsoft Office 365 Advanced Threat Protection**: Detects and blocks delivery of malicious email and files.
- **Web browser protection technologies**: Some browsers block their users from getting to known malicious web sites.
- **Anti-malware suites**: Anti-malware can detect and block delivery of malicious content from storage media, the network, and via web browsers.
- **File Integrity Monitoring (FIM)**: FIM can detect and block system file replacement attempts.
- **IDS/IPS**: Can detect and potentially disrupt or stop delivery.

The following controls are example controls that can be used to **Disrupt** attacker activities in the **Delivery** phase of an attack:

- **Operating System Mandatory Access Control**: Controls access to files and devices in ways that can disrupt or degrade delivery.
- **Short-lived environments**: Systems that are replaced every few hours can make delivery harder.
- **Anti-malware suites**: Anti-malware can detect and block delivery of malicious content from storage media, the network, and via web browsers.
- **File Integrity Monitoring (FIM)**: FIM can detect and block system file replacement attempts.
- **IDS/IPS**: Can detect and potentially disrupt or stop delivery.

The following controls are example controls that can be used to **Degrade** attacker activities in the **Delivery** phase of an attack:

- **Operating System Mandatory Access Control**: Controls access to files and devices in ways that can disrupt or degrade delivery.
- **Short-lived environments**: Systems that are replaced every few hours can make delivery harder.
- **Anti-malware suites**: Anti-malware can detect and block delivery of malicious content from storage media, the network, and via web browsers.
- **File Integrity Monitoring (FIM)**: FIM can detect and block system file replacement attempts.
- **IDS/IPS**: Can detect and potentially disrupt or stop delivery.

The following controls are example controls that can be used to **Deceive** attackers in the **Delivery** phase of an attack:

- **Deception technology**: Deception technologies can attract attackers and detect weapon delivery to deception assets.

The following controls are example controls that can be used to **Limit** attacker activities in the **Delivery** phase of an attack:

- **Identity and Access Management technologies**: Enforcing the principle of least privilege and meaningful separation of duties can help limit delivery in an IT environment.

The following controls are example controls that can be used to **Restore** in the **Delivery** phase of an attack:

- **Backups**: Restoring from backups as necessary.
- **Images and containers**: Rebuilding infrastructure as necessary.

The examples I have provided here are simple, but I hope they give security teams some ideas. Layering capabilities into the mix that break the Delivery phase, regardless of the delivery vector, is key.

Attack phase – Exploitation

After attackers have successfully delivered their weapons to their targets, the weapons must be activated. Sometimes, the Delivery and Exploitation phases occur in immediate succession, such as a drive-by download attack. In this scenario, a user is typically tricked into going to a malicious website via a URL in an email or online content. When they click the link and their web browser performs name resolution and loads the page, scripts on the malicious page will detect the operating system and browser and then try to deliver exploits for that software. If the software isn't patched for the vulnerabilities those exploits are designed for, then attackers will typically download more malware to the system, install tools, and continue with their Kill Chain. The Delivery and Exploitation phases happen at almost the same time in this type of attack. In other attacks, like email-based attacks, delivery can happen minutes, hours, days, weeks, or even months before the user opens the email and clicks on a malicious attachment or URL to a malicious website. In this scenario, the Delivery and Exploitation phases are distinct (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.). Some attackers seek instant gratification, while others prefer the "low and slow" method.

Defenders must be prepared for attacks across this spectrum. They cannot assume that the Delivery and Exploitation phases will always occur at nearly the same time, but they definitely must be prepared for such scenarios. Breaking the Exploitation phase of the attackers Kill Chain is critical, because if they successfully complete this phase of their attack, they could potentially have a foothold into the environment from which they can further penetrate it. After this phase in an attack, managing defenses can become harder for defenders. Because many attacks are automated, post-Exploitation phase activities can happen very quickly. Breaking the attacker's Kill Chains "left of boom", as the saying goes, is a prudent goal for security teams.

The best way to prevent exploitation of unpatched vulnerabilities and security misconfigurations (two of the cybersecurity usual suspects) is to scan and patch everything every day. Scanning all IT assets every day minimizes the times where unpatched vulnerabilities and security misconfigurations exist in the environment, thus surfacing the residual risk so that it can be mitigated, transferred, or accepted consciously.

In addition to patching everything every day, the following list provides you with some example controls that can be used to **Detect** attacker activities in the **Exploitation** phase of an attack. Hopefully, this will give you some ideas on how to make the Exploitation phase much more challenging for attackers:

- **Anti-malware suites**: Anti-malware can detect and block exploitation of vulnerabilities.
- **Containerization and supporting security tools**: Containers can reduce attack surface area and tools can help detect and prevent exploitation.
- **File Integrity Monitoring (FIM)**: FIM can detect some exploitation attempts.
- **Log reviews**: Reviewing various system logs can reveal indicators of exploitation.

The following controls are example controls that can be used to **Deny** attacker activities in the **Exploitation** phase of an attack:

- **Anti-malware suites**: Anti-malware can detect and block exploitation of vulnerabilities.
- **Containerization and supporting security tools**: Containers can reduce attack surface area and tools can help detect and prevent exploitation.
- **Address Space Layout Randomization (ASLR)**: Operating systems' ASLR can make exploitation inconsistent or impossible.
- **Data Execution Prevention (DEP)**: Operating systems' DEP can make exploitation inconsistent or impossible.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can deny exploitation in some scenarios.

The following controls are example controls that can be used to **Disrupt** attacker activities in the **Exploitation** phase of an attack:

- **Anti-malware suites**: Anti-malware can disrupt exploitation of vulnerabilities.
- **Containerization and supporting security tools**: Containers can reduce attack surface area and tools can help detect and prevent exploitation.
- **Address Space Layout Randomization (ASLR)**: Operating systems' ASLR can make exploitation inconsistent or impossible.
- **Data Execution Prevention (DEP)**: Operating systems' DEP can make exploitation inconsistent or impossible.

The following controls are example controls that can be used to **Degrade** attacker activities in the **Exploitation** phase of an attack:

- **Anti-malware suites**: Anti-malware can degrade exploitation of vulnerabilities
- **Containerization and supporting security tools**: Containers can reduce attack surface area and tools can help detect and prevent exploitation.
- **Address Space Layout Randomization (ASLR)**: Operating systems' ASLR can make exploitation inconsistent or impossible.

- **Data Execution Prevention (DEP)**: Operating systems' DEP can make exploitation inconsistent or impossible.
- **Short-lived environments**: Systems that are replaced every few hours can make exploitation harder.

The following controls are example controls that can be used to **Deceive** attackers in the **Exploitation** phase of an attack:

- **Deception technology**: Deception technologies can attract attackers and deceive them into attacking fake environments.
- **HoneyPots**: Attracts attackers and can expose the exploits they use.

The following controls are example controls that can be used to **Limit** attacker activities in the **Exploitation** phase of an attack:

- **Address Space Layout Randomization (ASLR)**: Operating systems' ASLR can make exploitation inconsistent or impossible.
- **Data Execution Prevention (DEP)**: Operating systems' DEP can make exploitation inconsistent or impossible.

The following controls are example controls that can be used to **Restore** in the **Exploitation** phase of an attack:

- **Backups**: Restoring from backups as necessary.
- **Images and containers**: Rebuilding infrastructure as necessary.

Some of the capabilities discussed in the list above can help in this phase of an attack include:

- **Address Space Layout Randomization (ASLR)**: This memory safety feature can make exploiting vulnerabilities harder for attackers by randomizing address space locations. This makes it harder for attackers to consistently predict the memory locations of vulnerabilities they wish to exploit. ASLR should be used in combination with Data Execution Prevention (Matt Miller, 2010).
- **Data Execution Prevention (DEP)**: Another memory safety feature that stops attackers from using memory pages meant for data to execute their code. DEP should be used in combination with ASLR (Matt Miller, 2010).

- **Containerization and supporting security tools**: Using container technologies such as Docker and Kubernetes has many advantages, not least in helping to reduce the attack surface area for systems and applications. Of course, containers are software too and subsequently have vulnerabilities of their own. There are vendors that offer tools to help detect and prevent exploitation in environments that leverage containers. Some examples include:
    - Aqua Security
    - CloudPassage
    - Illumio
    - Tenable
    - Twistlock
    - Others
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make it harder for exploitation of vulnerabilities. Sometimes, the attacker's code runs under the account context of the user that executed it, instead of under elevated privileges. Limiting user privileges can make it harder for exploitation to succeed or have the intended effect.
- **Short-lived environments**: Systems that only live for a few hours and are replaced with fully patched systems can make it much harder for exploitation to succeed.

Spending time carefully layering controls to break the Exploitation phase of an attacker's Kill Chain is time well spent (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.). An entire chapter in this book could be devoted to exploitation; I have only scratched the surface here, but I encourage CISOs and security teams to spend more time researching and considering how to implement this particular phase of this framework in their environments.

Attack phase – Installation

Simply successfully exploiting a vulnerability isn't the goal for most modern-day attackers, as it was back in 2003. Notoriety has been replaced by much more serious and sinister motivations. Once attackers successfully deliver their weapons and exploitation is successful, they typically seek to expand their scope of control in their victims' environments.

To do this, they have a range of options available to them, such as unpacking malware or remote-control tools from within the exploit itself, or downloading them from another system under their control.

More recently, "living off the land" has regained popularity with attackers that seek to use tools, scripts, libraries, and binaries that are native and pre-installed with operating systems and applications. This tactic allows them to further penetrate compromised environments, all while evading defenders that focus on detecting the presence of specific files associated with malware and exploitation. Be aware that "living off the land" tactics can be used in several phases of attackers Kill Chain, not just in the **Installation** phase. Also, note that although it has been modernized somewhat, this tactic is as old as I am and relies on the tribal knowledge of past defenders being lost in time.

When I worked on Microsoft's Incident Response team in 2003, every attacker "lived off the land." We saw a lot of creative tactics being used by attackers in those days. One lesson I learned was that removing all the built-in support tools native to the operating system, such as `ping.exe`, `tracert.exe`, and many others that attackers relied on, forced attackers to bring more of their own tools. Finding any of those tools on systems in the supported IT environment was an indicator of compromise. In the meantime, Desktop and Server Support personnel could download their own tools from a network share for troubleshooting purposes and remove them when they were done. Today, attackers are more sophisticated, using system binaries and libraries that can't really be removed without potentially damaging the operating system. However, leaving them with as little land to live off as possible can help defenders in multiple phases of an attack.

Attackers also rely on a lot of tricks to stay hidden on a system. For example, they would run components of their remote-control or surveillance software on a victim's system by naming it the same as a system file that administrators would expect to be running on the system, but running it from a slightly different directory. The file and the process look normal, and most administrators wouldn't notice it was running from the `system` directory instead of the `system32` directory. This tactic was so common that I developed some popular support tools for Windows that could help detect such shenanigans, including Port Reporter, Port Reporter Parser, and Portqry (Microsoft Corporation, n.d.).

These tools are still available on the Microsoft Download Center for free download, although I doubt that they will run properly on Windows 10-based systems today as many Windows APIs have changed since I developed these tools. Of course, I had to have some fun when I developed these tools; my name appears in the Port Reporter log files and when the hidden `/dev` switch is run with Portqry:

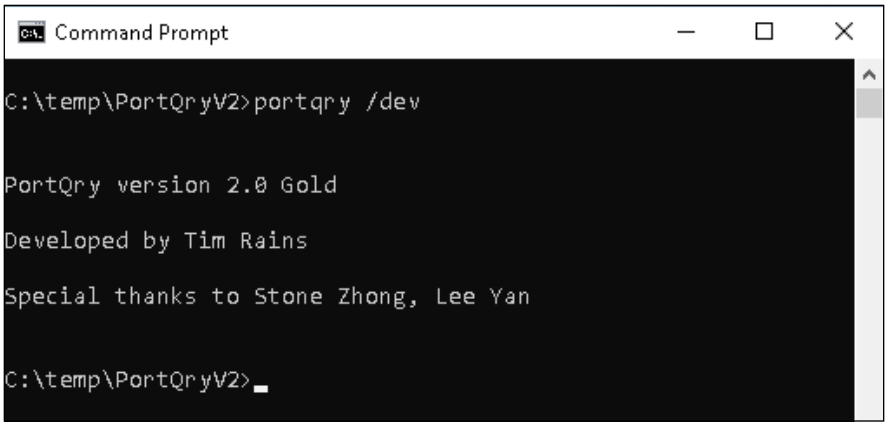


Figure 6.2: Easter egg fun with Portqry version 2.0

Some of the capabilities that will help break the Installation phase of attacks include:

- **Anti-malware suites**: Anti-malware software can detect and block the attempted installation of different types of weapons. Keep anti-malware suites up to date; otherwise, they can increase the attack surface themselves.
- **File Integrity Monitoring (FIM)**: I'm a fan of FIM. When it works properly, it can help detect installation attempts and, ideally, stop them. It can also help meet compliance obligations that many organizations have. FIM capabilities are built into many endpoint protection suites and can be integrated with SIEMs. Some of the FIM vendors/products I've seen in use include:
    - McAfee
    - Qualys
    - Tripwire
    - Many others

- **Identity and Access Management controls**: Adhering to the principle of least privilege can make it harder for installation to succeed.
- **Windows Device Guard**: This can lock down Windows 10 systems to prevent unauthorized programs from running (Microsoft Corporation, 2017). This can help prevent exploitation and installation during an attack.
- **Mandatory Access Control, Role-Based Access Control on Linux systems**: These controls help enforce the principle of least privilege and control access to files and processes, which can make installation much harder or impossible.

The following controls are example controls that can be used to **Detect** attacker activities in the **Installation** phase of an attack:

- **Anti-malware suites**: Anti-malware can detect and block installation.
- **File Integrity Monitoring (FIM)**: FIM can detect and prevent changes to systems and application files.
- **Log reviews**: Reviewing various system logs can reveal indicators of installation.

The following controls are example controls that can be used to **Deny** attacker activities in the **Installation** phase of an attack:

- **Anti-malware suites**: Anti-malware can detect and block installation.
- **File Integrity Monitoring (FIM)**: FIM can detect and prevent changes to systems and application files.
- **Short-lived environments**: Systems that are replaced every few hours can make installation harder.
- **Windows Device Guard**: Can make it harder for unauthorized programs to run.
- **Mandatory Access Control, Role-Based Access Control on Linux systems**: Can make it harder for unauthorized programs to run.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make installation much harder or impossible.

The following controls are example controls that can be used to **Disrupt** attacker activities in the **Installation** phase of an attack:

- **Anti-malware suites**: Anti-malware can detect and block installation.
- **File Integrity Monitoring (FIM)**: FIM can detect and prevent changes to systems and application files.
- **Short-lived environments**: Systems that are replaced every few hours can make installation harder.
- **Windows Device Guard**: Can make it harder for unauthorized programs to run.
- **Mandatory Access Control, Role-Based Access Control on Linux systems**: Can make it harder for unauthorized programs to run.

The following controls are example controls that can be used to **Degrade** attacker activities in the **Installation** phase of an attack:

- **Anti-malware suites**: Anti-malware can detect and block installation.
- **FIM**: FIM can detect and prevent changes to systems and application files.
- **Short-lived environments**: Systems that are replaced every few hours can make installation harder.
- **Windows Device Guard**: Can make it harder for unauthorized programs to run.
- **Mandatory Access Control, Role-Based Access Control on Linux systems**: Can make it harder for unauthorized programs to run.

The following controls are example controls that can be used to **Deceive** attackers in the **Installation** phase of an attack:

- **Deception technology**: Deception technologies can attract attackers and deceive them into attacking fake environments.

The following controls are example controls that can be used to **Limit** attacker activities in the **Installation** phase of an attack:

- **File Integrity Monitoring (FIM)**: FIM can detect and prevent changes to systems and application files.

- **Windows Device Guard**: Can make it harder for unauthorized programs to run.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can limit installation.

The following controls are example controls that can be used to **Restore** in the **Installation** phase of an attack:

- **Backups**: Restoring from backups as necessary.
- **Images and containers**: Rebuilding infrastructure as necessary.

There are lots of other controls that can help detect, deny, disrupt, degrade, deceive, and limit attackers during the Installation phase of their attack. If attackers are successful in this phase, most organizations will not rely on anti-malware or host-based restore points to recover; they will format the system and rebuild it from scratch, using images or backups. The cloud makes this much easier, as I discussed earlier, with short-lived environments, autoscaling, and other capabilities.

Attack phase – Command and Control (C2)

If attackers are successful in the Installation phase of their attack, typically they seek to establish communications channels with the compromised systems. These communications channels enable attackers to send commands to the systems that they compromised, enabling them to take a range of actions in the next phases of their attacks. A botnet is a great illustrative example. Once attackers have compromised systems and installed their C2 software on them, they can now use those "zombie" systems for a plethora of illicit purposes including identity theft, intellectual property theft, DDoS attacks, and so on.

There are numerous techniques that attackers can employ for C2 communications. Some are more innovative and interesting than others. Communicating across the network is the most straightforward approach and attackers have developed many different methods and protocols to facilitate C2 communications; these range from simply listening on a predefined TCP or UDP port number for commands to using more elaborate protocols like RPC and DNS, custom-built protocols, and employing proxies to further obfuscate their communications.

All these techniques can potentially help attackers remotely control compromised environments while evading detection. They want their network traffic to blend in with other legitimate network traffic. Some attackers have developed impressive domain generation algorithms that allow attackers to dynamically change IP addresses used for C2 communications. Conficker was the first big worm attack to use this method, more than a decade ago. Some attackers have developed obfuscated and encrypted protocols that make it harder for defenders to detect and stop the attacker's commands. The MITRE ATT&CK® framework provides a great list of techniques attackers use for C2 communications (MITRE, 2019). This is a good example of how the ATT&CK framework (MITRE) and the Intrusion Kill Chain framework complement each other.

By detecting, denying, disrupting, degrading, deceiving, and limiting C2 communications, defenders can minimize damage and expense to their organizations and accelerate recovery, all while increasing the expense to attackers. This is an area where vendors that have extensive networking expertise and capabilities, married with threat intelligence, can really add value. Some of the ways that defenders can do this include:

- **IDS/IPS**: These systems can detect and block C2 communications at several places on networks. Many organizations run IDS/IPS in their DMZs and inside their corporate networks. Many vendors offer IDS/IPS systems, including:
    - Cisco
    - FireEye
    - HP
    - IBM
    - Juniper
    - McAfee
    - Others
- **Network micro-segmentation**: This can provide granular control by enabling organizations to apply policies to individual workloads. This can make it harder for attackers to use compromised systems for C2 communications.

- **Log reviews**: Analyzing logs, net flow data, and DNS queries in an environment can help detect C2 communications. Since there can be too much data for humans to do this manually, many organizations now employ Artificial Intelligence and/or Machine Learning to do this for them. Of course, the cloud makes this much easier than trying to do this on-premises.

The following controls are example controls that can be used to **Detect** attacker activities in the **C2** phase of an attack:

- **IDS/IPS**: Can detect and stop communications.
- **Firewalls and proxy servers**: Communication with remote networks can be detected and blocked by firewalls and proxy servers.
- **Log reviews**: Reviewing various system logs, including DNS queries, can reveal indicators of C2 communications.

The following controls are example controls that can be used to **Deny** attacker activities in the **C2** phase of an attack:

- **IDS/IPS**: Can detect and stop communications.
- **Firewalls and proxy servers**: Communication with remote networks can be detected and blocked by firewalls and proxy servers.
- **Short-lived environments**: Systems that are replaced every few hours can make C2 communications harder to achieve and inconsistent.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make some C2 communications techniques much more difficult.
- **Network micro-segmentation**: Enforcing rules that restrict communications can make C2 communications more difficult.

The following controls are example controls that can be used to **Disrupt** attacker activities in the **C2** phase of an attack:

- **IDS/IPS**: Can detect and stop communications.
- **Firewalls and proxy servers**: Communication with remote networks can be detected and blocked by firewalls and proxy servers.

- **Short-lived environments**: Systems that are replaced every few hours can make C2 communications harder to achieve and inconsistent.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make some C2 communications techniques much more difficult.
- **Network micro-segmentation**: Enforcing rules that restrict communications can make C2 communications more difficult.

The following controls are example controls that can be used to **Degrade** attacker activities in the **C2** phase of an attack:

- **IDS/IPS**: Can detect and stop communications.
- **Firewalls and proxy servers**: Communication with remote networks can be detected and blocked by firewalls and proxy servers.
- **Short-lived environments**: Systems that are replaced every few hours can make C2 communications harder to achieve and inconsistent.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make some C2 communications techniques much more.
- **Network micro-segmentation**: Enforcing rules that restrict communications can make C2 communications more difficult.

The following controls are example controls that can be used to **Deceive** attackers in the **C2** phase of an attack:

- **Deception technology**: Attackers communicating with fake environments waste their time and energy.

The following controls are example controls that can be used to **Limit** attacker activities in the **C2** phase of an attack:

- **IDS/IPS**: Can detect and stop communications.
- **Firewalls and proxy servers**: Communication with remote networks can be detected and blocked by firewalls and proxy servers.

- **Short-lived environments**: Systems that are replaced every few hours can make C2 communications harder to achieve and inconsistent.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make some C2 communications techniques much more difficult.
- **Network micro-segmentation**: Enforcing rules that restrict communications can make C2 communications more difficult.

A critical aspect of detecting and preventing C2 communications is threat intelligence. Keep the tips I provided in *Chapter 3*, *The Evolution of the Threat Landscape – Malware*, on threat intelligence vendors, in mind while evaluating vendors to help in this phase of the framework. Providing old intelligence, commodity intelligence, and false positives are rarely helpful, but seem to be common challenges many vendors have. I've also found that unless C2 communications or other malicious network traffic can be traced back to a specific identity context in the compromised environment, it can be less actionable. Subsequently, C2 detection and prevention systems that are integrated with identity systems seem to have an advantage over those that do not have such integrations. The value of these systems seems to be a function of the time and effort spent fine-tuning them, especially to minimize false positives.

Attack phase – Reconnaissance II

One of the things that attackers often command the compromised systems that they control to do is help them map out their victim's network. Attackers often want to explore their victims' networks, looking for valuable data, valuable intellectual property, and high-value assets that they can steal, damage, or demand a ransom for their return. They also look for information, accounts, infrastructure, and anything else that might help them gain access to the aforementioned list of valuables. Again, they are trying to blend their reconnaissance activities into the common, legitimate network traffic, authentication, and authorization processes that occur on their victims' networks. This helps them evade detection and stay persistent on the network for longer periods.

Detecting reconnaissance activities can help defenders discover compromised systems in their environment. Additionally, making this type of reconnaissance difficult or impossible for attackers to perform might help limit the damage and expense associated with a compromise. This can be easier said than done, especially in legacy environments with lots of homegrown applications and older applications whose behavior can be surprising and unpredictable in many cases. Many a SOC analyst have spotted a sequential port scan on their network, only to find some homegrown application using the noisiest possible way to communicate on the network. This behavior can usually be traced back to a developer trying to solve a problem while making their life easier. The world is full of applications like this, which make detecting true anomalies more work.

This is another phase where attackers routinely "live off the land." Whether they are running scripts to perform reconnaissance or doing it manually, when defenders leave most of the tools attackers need, installed by default on systems, it makes the attacker's jobs easier, not harder. Removing or restricting the use of these common tools everywhere possible inconveniences attackers and will make it easier to detect when these tools, or others like them, are used in the environment. However, it's unlikely that security teams will be able to remove all the binaries and libraries that attackers can use from their environments.

This is another good example of an integration point between the MITRE ATT&CK® framework (MITRE) and the Intrusion Kill Chain framework. The ATT&CK framework provides a list of assets that attackers commonly try to discover and a list of techniques attackers use to evade detection (MITRE, 2019). These can be used to design control sets that detect, deny, disrupt, degrade, deceive, and limit reconnaissance.

The following controls are example controls that can be used to **Detect** attacker activities in the **Reconnaissance II** phase of an attack:

- **Deception technology**: Deception technologies can help detect the attacker's reconnaissance activities.
- **Log reviews**: Reviewing various system logs, including DNS queries, can reveal indicators of compromise.

- **User Behavior Analytics**: Can detect anomalous behavior.
- **SAW/PAW**: Monitored and audited SAWs/PAWs help detect unusual use of privileged credentials.

The following controls are example controls that can be used to **Deny** attacker activities in the **Reconnaissance II** phase of an attack:

- **Network micro-segmentation**: Enforcing rules that restrict network traffic can make reconnaissance more difficult.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make it harder to perform reconnaissance.
- **Encryption everywhere**: Encrypting data in-transit and at rest can protect data from attackers.
- **SAW/PAW**: Can make it much harder for attackers to steal and use credentials for administrator accounts and other accounts with elevated privileges.
- **Active Directory hardening**: Makes it harder for attackers to access and steal credentials.

The following controls are example controls that can be used to **Disrupt** attacker activities in the **Reconnaissance II** phase of an attack:

- **Network micro-segmentation**: Enforcing rules that restrict network traffic can make reconnaissance more difficult.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make it harder to perform reconnaissance.
- **Encryption everywhere**: Encrypting data in-transit and at rest can protect data from attackers.
- **SAW/PAW**: Can make it much harder for attackers to steal and use credentials for administrator accounts and other accounts with elevated privileges.
- **Active Directory hardening**: Makes it harder for attackers to access and steal credentials.

The following controls are example controls that can be used to **Degrade** attacker activities in the **Reconnaissance II** phase of an attack:

- **Network micro-segmentation**: Enforcing rules that restrict network traffic can make reconnaissance more difficult.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make it harder to perform reconnaissance.
- **Encryption everywhere**: Encrypting data in-transit and at rest can protect data from attackers.
- **SAW/PAW**: Can make it much harder for attackers to steal and use credentials for administrator accounts and other accounts with elevated privileges.
- **Active Directory hardening**: Makes it harder for attackers to access and steal credentials.

The following controls are example controls that can be used to **Deceive** attackers in the **Reconnaissance II** phase of an attack:

- **Deception technology**: Deception technologies can trick attackers into spending time performing reconnaissance on fake environments instead of real ones.

The following controls are example controls that can be used to **Limit** attacker activities in the **Reconnaissance II** phase of an attack:

- **Network micro-segmentation**: Enforcing rules that restrict network traffic can make reconnaissance more difficult.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make it harder to perform reconnaissance.
- **Encryption everywhere**: Encrypting data in-transit and at rest can protect data from attackers.
- **SAW/PAW**: Can make it much harder for attackers to steal and use credentials for administrator accounts and other accounts with elevated privileges.

- **Active Directory hardening**: Makes it harder for attackers to access and steal credentials.

Some of the capabilities included in the preceding lists include:

- **Deception technology**: Whether the party performing reconnaissance inside the network is an attacker or an insider, deception technology can be helpful in detecting their presence. When someone starts poking at assets that no one in the organization has any legitimate business touching, this can be a red flag. Additionally, if attackers take the bait offered by deception technologies, like stealing credentials, for example, and they use those credentials somewhere else in the environment, that's a very good indication of reconnaissance activities.
- **User Behavior Analytics (UBA)**: UBA, or Entity Behavioral Analytics, can help identify when users and other entities access resources out of the norm. This can indicate an insider threat or stolen credentials being used by attackers and uncover reconnaissance activities. There are many vendors that provide products that do this type of detection, including:
    - Aruba
    - Exabeam
    - ForcePoint
    - LogRhythm
    - Microsoft
    - RSA
    - Splunk
    - Many others
- **SAW/PAW**: Secure administrator workstations (SAW) or Privileged Access Workstations (PAW) will make it much harder for attackers to steal and use credentials for administrator accounts and other accounts with elevated privileges. Monitored and audited SAWs/PAWs help detect unusual use of privileged credentials.

- **Active Directory hardening**: Makes it harder for attackers to access and steal credentials.
- **Encryption everywhere**: Protecting data while it travels across the network and everywhere it rests can be a powerful control for preventing effective reconnaissance. Of course, this relies on effective key management.

There are many more ways to detect and make reconnaissance harder for attackers. Though it seems like only after a successful compromise, during the response, are the tell-tale signs of reconnaissance spotted, investments in this phase of the framework can have big returns for security teams. The cloud can also make it easier to detect and prevent reconnaissance.

Attack phase – Actions on Objectives

Remember that there are many possible motivations for attacks, including notoriety, profit, military espionage, economic espionage, revenge, anarchy, and many others. Once attackers make it to this phase in their attack, their objectives are potentially within their reach. In this phase, they might lock administrators out of systems, exfiltrate data, compromise the integrity of data, encrypt data and infrastructure, make systems unbootable, or simply just stay persistent to watch their victims and collect data. Actions on Objectives (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.) depend on their motivations.

In some cases, this is the defender's last chance to detect and stop attackers before recovery becomes more expensive and potentially aspirational. However, the fact that attackers made it to this phase in their Kill Chain does not automatically mean they have access to all resources and are in full control of the IT environment; their objective might be much more tightly scoped, or the security controls that have been deployed to impede their progress might have had the intended effect. This could mean that many of the controls used to break other phases of the Kill Chain can still be helpful in this phase. If attackers were able to defeat or bypass controls in earlier phases of their attack, this doesn't mean they can do so everywhere in the IT environment, anytime. Detecting and denying attackers is ideal, but disrupting, degrading, deceiving, and limiting their attacks is highly preferable to recovering from them.

Actions on Objectives is another phase where there's great potential integration between the Intrusion Kill Chain model and the MITRE ATT&ACK® framework. MITRE has published a list of impact techniques commonly employed by attackers (MITRE, 2019). This list can inform the controls used to break the attacker's Kill Chains in this phase.

Some of the controls to consider when mitigating this phase of an attack include:

- **Data backups**: If attackers choose to destroy data by damaging storage media or firmware, wiping storage media, encrypting data, or otherwise tampering with the integrity of data, backups can be very helpful. Offline backups are highly recommended as attackers will happily encrypt online backups if they can with their ransomware or cryptoware.
- **SAW/PAW**: SAW or PAW can make it much harder for attackers to use privileged accounts to lock administrators out of the systems they manage.
- **Encryption everywhere**: Remember that encryption not only provides confidentiality, but it can also safeguard the integrity of data; encryption can help detect data that has been altered.
- **Identity and Access Management controls**: Identity is central to security. If attackers already own the Active Directory in the environment, then it's going to be very hard or impossible to expel them. However, if they only have access to some accounts, Identity and Access Management controls can still help limit the scope of their attack.

The following controls are examples of controls that can be used to **Detect** attacker activities in the **Actions on Objectives** phase of an attack:

- **Anti-malware suites**: Anti-malware can detect and block malware.
- **File Integrity Monitoring (FIM)**: FIM can detect and prevent changes to systems and application files.
- **Log reviews**: Reviewing various system logs can reveal indicators of compromise.

- **User Behavior Analytics**: Can detect anomalous behavior.
- **Deception technology**: Deception technologies can detect the attacker's actions on assets and use of deception assets.
- **SAW/PAW**: Monitored and audited SAWs/PAWs help detect unusual use of privileged credentials.

The following controls are examples of controls that can be used to **Deny** attacker activities in the **Actions on Objectives** phase of an attack:

- **Anti-malware suites**: Anti-malware can detect and block malware.
- **File Integrity Monitoring (FIM)**: FIM can detect and prevent changes to systems and application files.
- **Short-lived environments**: Systems that are replaced every few hours can make installation harder.
- **Windows Device Guard**: Can make it harder for unauthorized programs to run.
- **Mandatory Access Control, Role-Based Access Control on Linux systems**: Can make it harder for unauthorized programs to run.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make it harder for the attacker's actions on objectives.
- **SAW/PAW**: Can make it much harder for attackers to steal and use credentials for administrator accounts and other accounts with elevated privileges.
- **Encryption everywhere**: Encrypting data in-transit and at rest can protect data from attackers.

The following controls are examples of controls that can be used to **Disrupt** attacker activities in the **Actions on Objectives** phase of an attack:

- **Anti-malware suites**: Anti-malware can detect and block malware.
- **File Integrity Monitoring (FIM)**: FIM can detect and prevent changes to systems and application files.
- **Short-lived environments**: Systems that are replaced every few hours can make installation harder.

- **Windows Device Guard**: Can make it harder for unauthorized programs to run.
- **Mandatory Access Control, Role-Based Access Control on Linux systems**: Can make it harder for unauthorized programs to run.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make it harder for the attacker's actions on objectives.
- **SAW/PAW**: Can make it much harder for attackers to steal and use credentials for administrator accounts and other accounts with elevated privileges.
- **Encryption everywhere**: Encrypting data in-transit and at rest can protect data from attackers.

The following controls are examples of controls that can be used to **Degrade** attacker activities in the **Actions on Objectives** phase of an attack:

- **Anti-malware suites**: Anti-malware can detect and block malware.
- **File Integrity Monitoring (FIM)**: FIM can detect and prevent changes to systems and application files.
- **Short-lived environments**: Systems that are replaced every few hours can make installation harder.
- **Windows Device Guard**: Can make it harder for unauthorized programs to run.
- **Mandatory Access Control, Role-Based Access Control on Linux systems**: Can make it harder for unauthorized programs to run.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make it harder for the attacker's actions on objectives.
- **SAW/PAW**: Can make it much harder for attackers to steal and use credentials for administrator accounts and other accounts with elevated privileges.
- **Encryption everywhere**: Encrypting data in-transit and at rest can protect data from attackers.

The following controls are examples of controls that can be used to **Deceive** attackers in the **Actions on Objectives** phase of an attack:

- **Deception technology**: Deception technologies can attract attackers and deceive them into attacking fake environments.

The following controls are examples of controls that can be used to **Limit** attacker activities in the **Actions on Objectives** phase of an attack:

- **Anti-malware suites**: Anti-malware can detect and block malware.
- **File Integrity Monitoring (FIM)**: FIM can detect and prevent changes to systems and application files.
- **Windows Device Guard**: Can make it harder for unauthorized programs to run.
- **Identity and Access Management controls**: Strictly following the principle of least privilege can make it harder for the attacker's actions on objectives.
- **SAW/PAW**: Can make it much harder for attackers to steal and use credentials for administrator accounts and other accounts with elevated privileges.
- **Encryption everywhere**: Encrypting data in-transit and at rest can protect data from attackers.

The following controls are examples of controls that can be used to **Restore** in the **Actions on Objectives** phase of an attack:

- **Backups**: Restoring from backups as necessary.
- **Images and containers**: Rebuilding infrastructure as necessary.
- **Disaster Recovery processes and technologies**

Conclusion

That's one way to implement the Intrusion Kill Chain framework. Obviously, there are other possible interpretations and approaches to implementing this model. I've seen some very well thought out and sophisticated approaches to this framework at conferences and documented on the internet, but the *best* way is the one that addresses the specific HVAs and risks that your organization is concerned about.

Remember that "best practices" are based on the threats and assets that someone else has in mind, not necessarily yours.

This might be obvious, but the Intrusion Kill Chain framework can help CISOs and security teams take a structured approach to managing intrusions. Arguably, intrusions are the most serious threats for most organizations because of their potential impact, but there are other threats that CISOs need to address. DDoS attacks, for example, typically don't involve intrusion attempts or require a Kill Chain framework to address.

Additionally, this approach has become a little dated in a world where the cloud has disrupted and improved upon traditional approaches to IT and cybersecurity. Although this approach still has the potential to be highly effective in on-premises and hybrid environments, a framework designed to break Intrusion Kill Chains and stop so-called **advanced persistent threat** (**APT**) actors isn't as relevant in the cloud. Used effectively, CI/CD pipelines, short-lived environments, autoscaling, and other capabilities the cloud offers simply leave no place for APT actors or other attackers to get a foothold in order to move laterally and remain persistent. Simply put, the cloud gives CISOs the opportunity to change the playing field dramatically. I'll discuss the cybersecurity benefits the cloud offers in more detail in *Chapter 8*, *The Cloud – A Modern Approach to Security and Compliance*.

Given that the industry will continue to transition from the old-fashioned on-premises IT world to the cloud over the next decade, the Intrusion Kill Chain framework still seems well poised to help organizations as a transitional cybersecurity strategy. It can help organizations on-premises and in the cloud as they modernize their workforces to take advantage of DevOps, as well as Zero Trust methods, as they come to fruition. Crucially, employing this strategy is still potentially far superior to not having a cybersecurity strategy or using many of the other strategies I examined in *Chapter 5*, *Cybersecurity Strategies*. If your organization doesn't have a cybersecurity strategy or it does but no one can articulate it, you could likely do far worse than to embrace the Intrusion Kill Chain strategy.

To do so, in many cases, you'll have to get far more detailed and specific than the high-level example that I have provided here. However, I think I have provided you with a head-start on the best scoring cybersecurity strategy that we have examined. This is not a bad thing to have.

Chapter summary

CISOs and security teams have numerous cybersecurity strategies, models, frameworks, and standards to choose from when developing their approach to protecting, detecting, and responding to modern-day threats. One Attack-Centric Strategy that we examined in *Chapter 5*, *Cybersecurity Strategies*, the Intrusion Kill Chain, deserves serious consideration as it garnered the highest CFSS estimated total score. It earned nearly a perfect score with 95 points out of a possible 100. This chapter sought to provide you with an example of one way this model can be implemented.

The Intrusion Kill Chain model was pioneered by Lockheed Martin; the Kill Chain phases provided in Lockheed Martin's paper on this topic include Reconnaissance, Weaponization, Delivery, Exploitation, Installation, Command and Control (C2), and Actions on Objectives (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.). One consideration before implementing this framework is whether defenders should use the original Intrusion Kill Chain framework or to update it.

There are several ways this framework can be modernized. It can be modified or reorganized around the cybersecurity usual suspects to ensure that they are mitigated and make it easier to identify gaps in an organization's security posture. Split the Reconnaissance phase into two phases instead of one; the one attackers use before initial compromise and the one after compromise. The Weaponization phase can be dropped as CISOs typically do not have very effective controls for protection and detection prior to the Delivery phase. The Destroy phase can be replaced with more pragmatic phases such as Limit and Restore. Adding a maturity index, to capture and communicate how much or how well each cybersecurity capability mitigates threats, can help identify areas of under-investment and potential gaps in defenses. Adding a point of contact for each mitigation, to make it clear who is consuming the data generated by cybersecurity capabilities, will help ensure there are no unmanaged mitigations in the environment. Tracking cybersecurity license renewals and support deadlines will help prevent lapses in capabilities.

Rationalizing mitigations can help identify gaps and areas of under-investment and over-investment. Where to start with an implementation can be informed by many factors, including budget, resources, gaps, and areas of under-investment and over-investment. Implementing controls that help break Kill Chains in multiple places might offer security teams higher ROIs.

That concludes my example of how a cybersecurity strategy can be implemented. I hope the tips and tricks I have provided are helpful to you. In the next chapter, I'll examine how CISOs and security teams can measure whether the implementation of their strategy is effective. This can be an important, yet elusive goal for security teams..

References

1. Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D. (n.d.). *Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains*. Retrieved from Lockheed Martin: `https://lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/LM-White-Paper-Intel-Driven-Defense.pdf`
2. H. P. Sanghvi, M. S. (2013). *Cyber Reconnaissance: An Alarm before Cyber Attack*. International Journal of Computer Applications (0975 – 8887), Volume 63– No.6, pages 2-3. Retrieved from `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.278.5965&rep=rep1&type=pdf`
3. Matt Miller, M. (December 8, 2010). *On the effectiveness of DEP and ASLR*. Retrieved from Microsoft Security Response Center: `https://msrc-blog.microsoft.com/2010/12/08/on-the-effectiveness-of-dep-and-aslr/`
4. Microsoft Corporation. (October 13, 2017). *Control the health of Windows 10-based devices*. Retrieved from Microsoft Docs: `https://docs.microsoft.com/en-us/windows/security/threat-protection/protect-high-value-assets-by-controlling-the-health-of-windows-10-based-devices`

5. Microsoft Corporation. (n.d.). *PortQry Command Line Port Scanner Version 2.0*. Retrieved from Microsoft Download Center: `https://www.microsoft.com/en-us/download/details.aspx?id=17148`
6. MITRE. (July 2019). *Command and Control*. Retrieved from MITRE ATT&CK®: `https://attack.mitre.org/tactics/TA0011/`
7. MITRE. (July 2019). *Defense Evasion*. Retrieved from MITRE ATT&CK®: `https://attack.mitre.org/tactics/TA0005/`
8. MITRE. (July 2019). *Discovery*. Retrieved from MITRE ATT&CK®: `https://attack.mitre.org/tactics/TA0007/`
9. MITRE. (July 25, 2019). *Impact*. Retrieved from MITRE ATT&CK®: `https://attack.mitre.org/tactics/TA0040/`
10. MITRE. (n.d.). *MITRE ATT&CK®*. Retrieved from MITRE ATT&CK®: `https://attack.mitre.org/`
11. MITRE. (n.d.). *MITRE ATT&CK® FAQ*. Retrieved from MITRE ATT&CK®: `https://attack.mitre.org/resources/faq/`

7

Measuring Performance and Effectiveness

How do we know if the cybersecurity strategy we've employed is working as planned? How do we know if the CISO and the security team are being effective? This chapter will focus on measuring the effectiveness of cybersecurity strategies.

Throughout this chapter, we'll cover the following topics:

- Using vulnerability management data
- Measuring performance and efficacy of cybersecurity strategies
- Examining an Attack-Centric Cybersecurity Strategy as an example
- Using intrusion reconstruction results

Let's begin this chapter with a question. Why do CISOs need to measure anything?

Introduction

There are many reasons why cybersecurity teams need to measure things. Compliance with regulatory standards, industry standards, and their own internal security standards are usually chief among them.

There are hundreds of metrics related to governance, risk, and compliance that organizations can choose to measure themselves against. Anyone who has studied for the **Certified Information Systems Security Professional** (**CISSP**) certification knows that there are numerous security domains, including Security and Risk Management, Asset Security, Security Architecture and Engineering, Communication and Network Security, **Identity and Access Management** (**IAM**), and a few others. (ISC2, 2020) The performance and efficacy of the people, processes, and technologies in each of these domains can be measured in many ways. In fact, the number of metrics and the ways they can be measured is dizzying. If you are interested in learning about the range of metrics available, I recommend reading Debra S. Herrmann's 848 page leviathan book on the topic, *Complete Guide to Security and Privacy Metrics: Measuring Regulatory Compliance, Operational Resilience, and ROI* (Herrmann, 2007).

Besides measuring things for compliance reasons, cybersecurity teams also try to find meaningful metrics to help prove they are adding value to the businesses they support. This can be challenging and a little unfair for CISOs. **Key Performance Indicators** (**KPIs**) typically measure performance against a target or objective. For security teams, it's failing to achieve an objective that tends to do the damage. It can be tough to find meaningful data that helps prove that the investments and efforts of the CISO and cybersecurity team are the reasons why the organization hasn't been compromised or had a data breach. Was it their work that prevented attackers from being successful? Or did the organization simply "fly under the radar" of attackers, as I've heard so many non-security executives suggest? This is where that submarine analogy that I introduced in the preface can be helpful. There is no flying under the radar on the internet where cybersecurity is concerned; there's only constant pressure from all directions. Besides, hope is not a strategy, it's the abdication of responsibility.

Nevertheless, CISOs need to be able to prove to their peers, the businesses or citizens they support, and to shareholders that the results they've produced aren't a by-product of luck or the fulfillment of hope. They need to show that their results are the product of successfully executing their cybersecurity strategy. I've seen many CISOs try to do this through opinion and anecdotal evidence.

But without data to support opinions and anecdotes, these CISOs tend to have a more difficult time defending the success of their strategy and cybersecurity program. It's only a matter of time before an auditor or consultant offers a different opinion that challenges the CISO's description of the current state of affairs.

Data is key to measuring performance and efficacy of a cybersecurity strategy. Data helps CISOs manage their cybersecurity programs and investments and helps them prove that their cybersecurity program has been effective and constantly improving. In this chapter, I'll provide suggestions to CISOs and security teams on how they can measure the effectiveness of their cybersecurity strategy. To do this, I'll use the best scoring strategy I examined in *Chapter 5*, *Cybersecurity Strategies* and *Chapter 6*, *Strategy Implementation*, the Attack-Centric Strategy, as an example. I'll also draw on concepts and insights that I provided in the preceding chapters of this book. I will not cover measuring things for compliance or other purposes here as there are many books, papers and standards that already do this. Let's start by looking at the potential value of vulnerability management data.

Using vulnerability management data

For organizations that are just bootstrapping a cybersecurity program or for CISOs that have assumed leadership of a program that has been struggling to get traction in their organization, vulnerability management data can be a powerful tool. Even for well-established cybersecurity programs, vulnerability management data can help illustrate how the security team has been effectively managing risk for their organization and improving over time. Despite this, surprisingly, I've met some CISOs of large, well-established enterprises who do not aggregate and analyze, or otherwise use data from their vulnerability management programs. This surprises me when I come across it. This is because this data represents one of the most straightforward and easy ways available for CISOs to communicate the effectiveness of their cybersecurity programs.

A challenge for CISOs and IT executives is to develop a performance overview based on data that aligns with the way business executives measure and communicate performance. The impact of such data can also be entirely different for CISOs.

For example, when a production site is behind target, additional resources and action plans will kick in to help compensate. But for CISOs, additional resources are rarely the result of being behind target; for the most part, security programs are supposed to be "dial tone."

As I discussed at length in earlier chapters, unpatched vulnerabilities and security misconfigurations are two of the five cybersecurity usual suspects that are managed via a vulnerability management program. Subsequently, a well-run vulnerability management program is not optional. As I discussed in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*, asset inventories that are complete and up to date are critical to the success of vulnerability management programs and cybersecurity programs overall. After all, it's difficult for security teams to manage assets that they do not know exist.

Vulnerability management teams should scan everything in their inventories every single day for vulnerabilities and misconfigurations. This will help minimize the amount of time that unmitigated vulnerabilities and misconfigurations are present and exploitable in their environments. Remember that vulnerabilities and misconfigurations can be introduced into IT environments multiple ways; newly disclosed vulnerabilities at the average rate of between 33 and 45 per day (over the past 3 years), software and systems built from old images or restored from backup, legacy software and systems that go out of support, orphaned assets that become unmanaged over time, among other ways.

Every day that a vulnerability management team scans all their assets, they will have a new snapshot of the current state of the environment that they can stitch together with all the previous days' snapshots. Over time, this data can be used multiple ways by the cybersecurity team. Let me give you some examples of how this data can be used.

Assets under management versus total assets

The number of assets under the management of the vulnerability management team versus the total number of assets that the organization owns and operates, can be an interesting data point for some organizations. The difference between these two numbers potentially represents risk, especially if there are assets that are not actively managed for vulnerabilities and misconfigurations by anyone. I've seen big differences between these two numbers in organizations where IT has been chronically understaffed for long periods and there isn't enough documentation or tribal knowledge to inform accurate asset inventories. Subsequently, there can be subnets of IT assets that are not inventoried and are not actively managed as part of a vulnerability management program.

I've also seen big differences in these numbers when CISOs do not have good relationships with IT leadership; in cases like this, inaccurate IT inventories seem common and represent real risk to the organization. In some of the cases I've seen, IT knows where all or most of the assets are but won't proactively work with the CISO to ensure they are all inventoried and patched. As I wrote in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*, CISOs must work to have good relationships with their stakeholder communities, especially with their IT organizations. CIOs and CTOs also need to realize that, more and more, their roles have a shared destiny with the CISO; when the vulnerability management program fails, they all fail and should share the "glory." The days where the CISO is the sole scapegoat for IT security failures are largely in the past. CISOs that find themselves in this scenario should work to improve their relationship with their IT partners. In some cases, this is easier said than done.

In the example scenario illustrated in *Figure 7.1*, the vulnerability management program continues to manage vulnerabilities and misconfigurations for the same number of IT assets throughout the year. They are blissfully unaware that there are subnets with IT assets they are not managing. They are also not actively managing the new IT assets that have been introduced into the environment during the year. The space between the two lines in the graph represents risk to the organization:

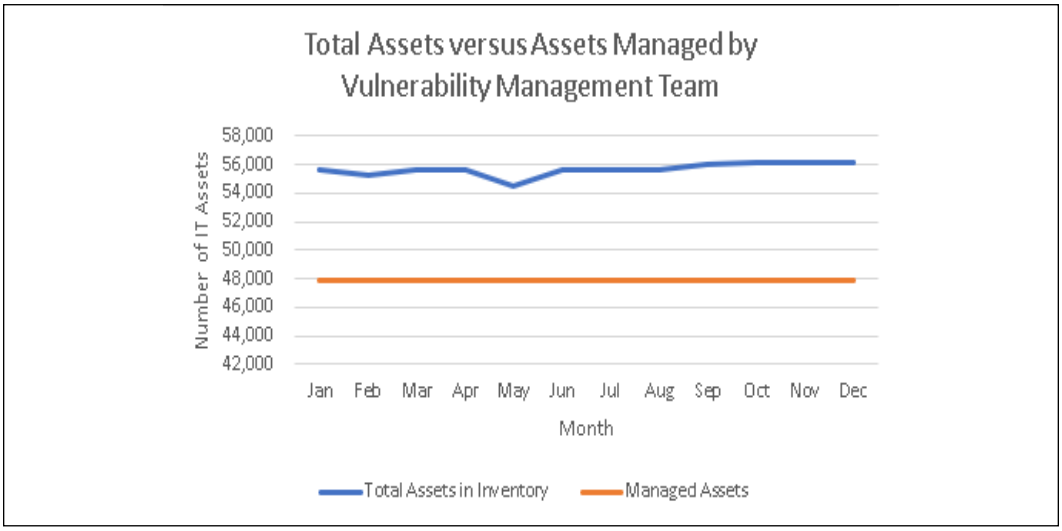


Figure 7.1: An example of trend data illustrating the difference between the total number of IT assets in inventory and the number of assets enrolled in the vulnerability management program

The total number of IT assets and the total number of assets that are actively managed for vulnerabilities and misconfigurations every day should be identical, in order to minimize risk. However, there might be good reasons, in large complex environments, for there to be exceptions to this rule. But exceptions still need to be known, understood, and tracked by the teams responsible for managing vulnerabilities; otherwise, the risk to the organization does not get surfaced to the right management level in the organization. Put another way, if the organization is going to have unpatched systems, the decision to do this and for how long needs to be accepted by the highest appropriate management layer and revisited periodically.

The appropriate management layer for decisions like this might not be in IT at all—it depends on the organization and the governance model they have adopted. Remember, a decision to allow an unpatched system to run in the environment is a decision to accept risk on behalf of the entire organization, not just the owner or manager of that asset. I've seen project managers all too enthusiastic to accept all manner of risks on behalf of their entire organization in order to meet the schedule, budget, and quality goals of their projects. This is despite the fact that the scope of their role is limited to the projects they work on. If a risk is never escalated to the proper management level, it could remain unknown and potentially unmanaged forever. Risk registries should be employed to track risk and periodically revisit risk acceptance and transference decisions.

In environments where the total number of IT assets and the total number of assets that are actively managed for vulnerabilities are meaningfully different, this is an opportunity for CISOs and vulnerability program managers to show how they are working to close that gap and thus reduce risk for the organization. They can use this data to educate IT leadership and their Board of Directors on the risks posed to their organizations. To do this, they can use partial and inaccurate asset inventories and talk about the presence of unmanaged assets. CISOs can provide stakeholders with regular updates on how the difference between the number of assets under the management of the vulnerability management team and the total number of assets that the organization owns and operates trends over time as IT and their cybersecurity team work together to reduce and minimize it. This data point represents real risk to an organization and the trend data illustrates how the CISO and their vulnerability management team has managed it over time. If this number trends in the wrong direction, it is the responsibility of senior leadership and the management board to recognize this and to help address it.

Figure 7.2 illustrates that the CISO and vulnerability management team have been working with their IT partners to reduce the risk posed by systems that have not been enrolled in their vulnerability management program.

This is a positive trend that this CISO can use to communicate the value of the cybersecurity program:

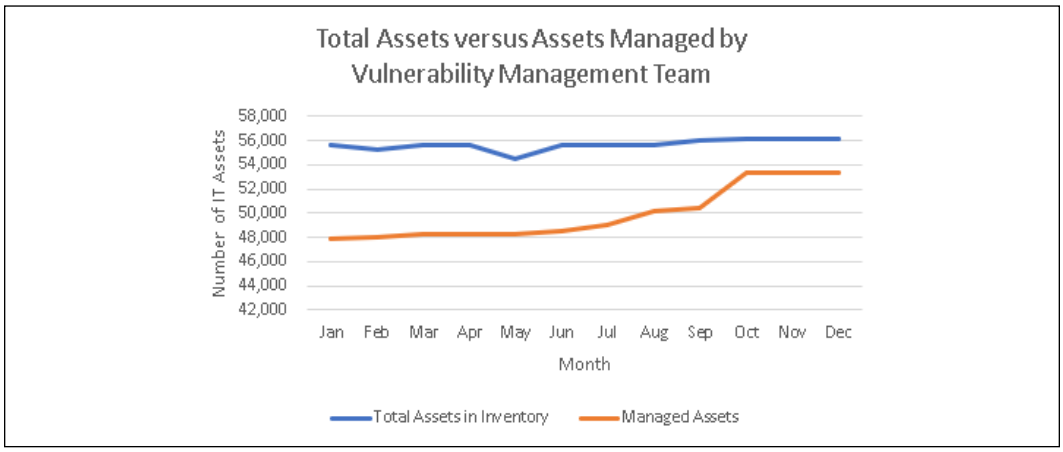


Figure 7.2: An example of trend data illustrating an improving difference between the total number of IT assets in inventory and the number of assets enrolled in the vulnerability management program

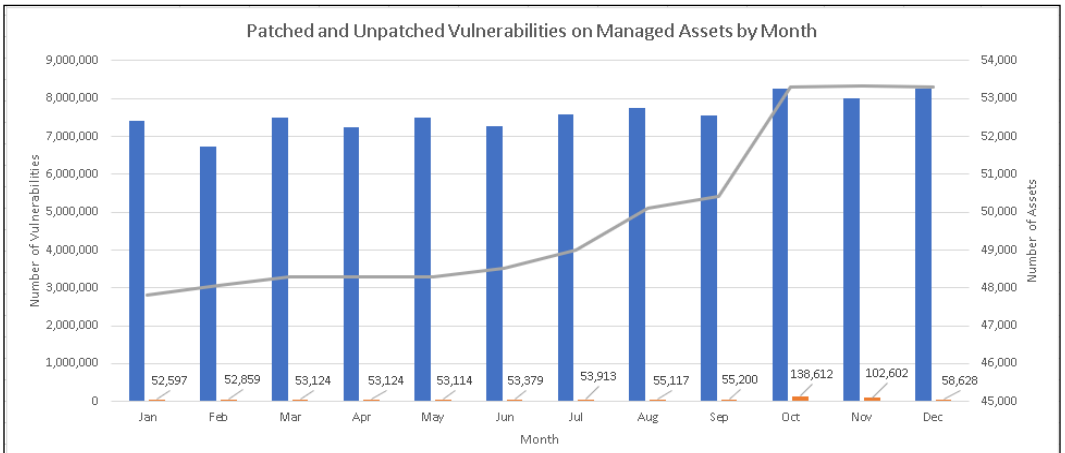
Known unpatched vulnerabilities

Another key data point from vulnerability management programs is the number of known unpatched vulnerabilities that are present in an environment. Remember that there are many reasons why some organizations have unpatched systems in their IT asset inventories. To be perfectly frank, the most frequently cited reason I have heard for this is a lack of investment in vulnerability management programs; understaffed and under-resourced programs simply cannot manage the volume of new vulnerabilities in their environments. Testing security updates and deploying them requires trained people, effective processes, and supporting technologies, in addition to time.

Regardless of the reasons, it is still important to understand which systems are unpatched, the severity of the unpatched vulnerabilities, and the mitigation plan for them. Regularly sharing how the number of unpatched vulnerabilities is reduced overt time can help communicate how the CISO and cybersecurity team are contributing to the success of the business. One nuance for rapidly changing environments to consider is how the number of vulnerabilities was reduced despite material changes to infrastructure or increases in the number of IT assets. To communicate this effectively,

CISOs might have to educate some of their stakeholder community on the basics and nuances of vulnerability management metrics, as well as their significance to the overall risk of the organization. There's typically only one or two members on a Board of Directors that have cybersecurity experience in their backgrounds, and even fewer executives with that experience in the typical C-suite. In my experience, educating these stakeholders is time well spent and will help everyone understand the value that the cybersecurity team is providing. In cases where the vulnerability management team is under-resourced, this data can help build the business case for increased investment, in an easy to understand way.

Figure 7.3 illustrates a scenario where a vulnerability management team was successfully minimizing increases in unpatched vulnerabilities in their environment, despite modest increases in the number of IT assets enrolled in their program. However, an acquisition of a smaller firm that closed in October introduced a large number of new IT assets that the vulnerability management team was expected to manage. This led to a dramatic increase in the number of unpatched vulnerabilities that the team was able to reduce to more typical levels by the end of the quarter:



	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Patched Vulns	7,411,325	6,727,571	7,485,624	7,244,152	7,484,175	7,278,964	7,596,812	7,766,523	7,560,675	8,261,461	8,000,948	8,261,167
Unpatched Vulns	52,597	52,859	53,124	53,124	53,114	53,379	53,913	55,117	55,200	138,612	102,602	58,628
Unpatched Change	n/a	0.50%	0.50%	0.00%	-0.02%	0.50%	1.00%	2.23%	0.15%	151.11%	-25.98%	-42.86%
Managed Assets	47,815	48,054	48,294	48,294	48,285	48,526	49,012	50,107	50,405	53,300	53,340	53,298
Managed Assets Change	n/a	0.50%	0.50%	0.00%	-0.02%	0.50%	1.00%	2.23%	0.59%	5.74%	0.07%	-0.08%

Figure 7.3: An example of trend data illustrating the number of patched vulnerabilities, the number of unpatched vulnerabilities and the number of systems enrolled in an organization's vulnerability management program

With data like this, the CISO and cybersecurity team look like heroes. Without data like this, it would be much harder to describe the scope of the challenge that the acquisition brought with it for the vulnerability management team, the subsequent increased workload, and the positive results. It's not all positive news, though, as this organization has a significant and growing number of unpatched vulnerabilities in their environment. The CISO should be able to articulate the plan to reduce the number of unpatched vulnerabilities to as close to zero as possible, using this same data to ask for more resources to accelerate that effort. Note that the figures I used in this example are completely fictional; actual data can vary wildly, depending on the number of assets, hardware, software, applications, patching policies, governance practices, and so on.

But reducing the number of unpatched vulnerabilities can be easier said than done for some organizations. Some known vulnerabilities simply can't be patched. There are numerous reasons for this. For example, many vendors will not offer security updates for software that goes out of support. Some vendors go out of business and subsequently, security updates for the products their customers have deployed will never be offered. Another common example is legacy applications that have compatibility issues with specific security updates for operating systems or web browsers. In cases like this, often, there are workarounds that can be implemented to make exploitation of specific vulnerabilities unlikely or impossible, even without installing the security updates that fix them. Typically, workarounds are meant to be short-term solutions until the security update that fixes the vulnerabilities can be deployed. However, in many environments, workarounds become permanent tenants. Reporting how known unpatched vulnerabilities are being mitigated using workarounds, instead of security updates, can help communicate risk and how it's being managed. Providing categories such as *workarounds in progress*, *workarounds deployed*, and *no workaround available* can help business sponsors see where decisions need to be made. The number of systems with workarounds deployed on them, as well as the severity of the underlying vulnerabilities that they mitigate, provides a nuanced view of risk in the environment. Marry this data with the long-term mitigation plan for the underlying vulnerabilities and CISOs have a risk management story they can share with stakeholders.

Unpatched vulnerabilities by severity

Another potentially powerful data point is the number of vulnerabilities unpatched in the environment, categorized by severity. As I discussed at length in *Chapter 2*, *Using Vulnerability Trends to Reduce Risk and Costs*, critical and high severity vulnerabilities represent the highest risk because of the probability and impact of their exploitation. Understanding how many of these vulnerabilities are present in the environment at any time, how long they have been present, and time to remediation are all important data points to help articulate the risk they pose. Longer term, this data can help CISOs understand how quickly these risks are being mitigated and uncover the factors that lead to relatively long lifetimes in their environments. This data can help vulnerability management program managers and CISOs build the business case for more resources and better processes and technologies. This data can also be one of the most powerful indicators of the value of the cybersecurity team and how effectively they have been managing risk for the organization, because the risk these vulnerabilities pose is among the most serious and it is easy to articulate to executives and boards.

Don't discount the value of medium severity vulnerabilities in IT environments for attackers. Because of the monetary value of critical and high rated vulnerabilities, attackers have been finding ways to use a combination of medium severity vulnerabilities to compromise systems. CISOs and vulnerability management teams need to manage these vulnerabilities aggressively to minimize risk to their environments. This is another opportunity to show value to the businesses they support and communicate progress against patching these vulnerabilities constantly.

Vulnerabilities by product type

Another potentially useful dataset is vulnerabilities categorized by product type. Let's face it; most of the action occurs on user desktops because they bring threats through perimeter defenses into IT environments. Just as eyes are the windows to the soul, so too are browsers to operating systems. Attackers are constantly trying to find and exploit vulnerabilities in web browsers and operating systems.

The data explored in **Figure 7.4** is also touched upon in *Chapter 2*, *Using Vulnerability Trends to Reduce Risk and Costs*:

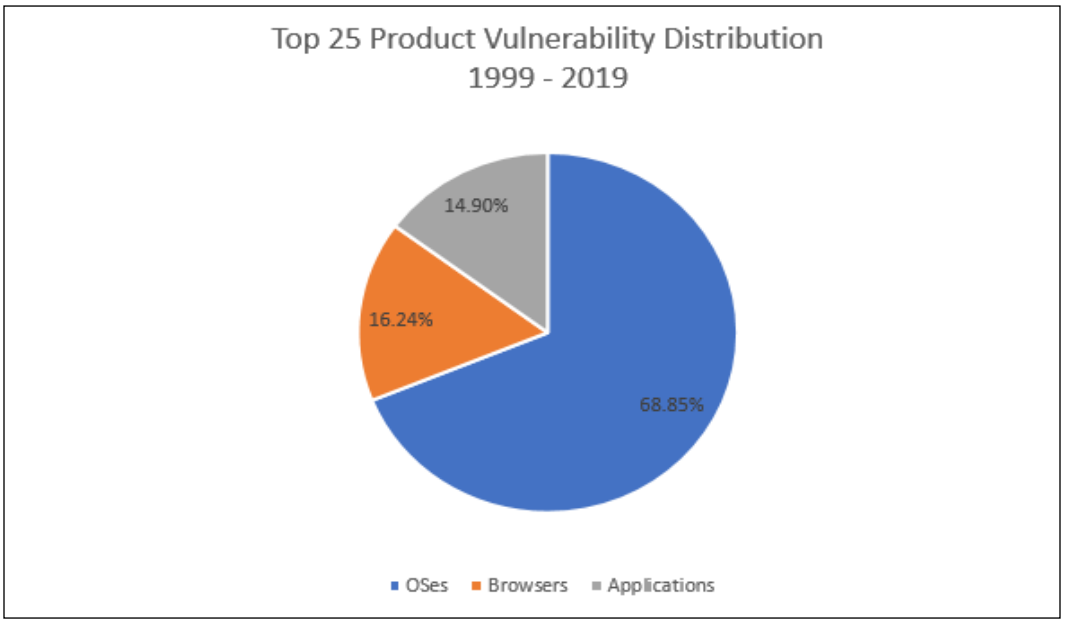


Figure 7.4: Vulnerabilities in the 25 products with the most CVEs categorized by product type (1999–2019) (CVE Details, 2019)

Vulnerability management teams can develop similar views for their environments to illustrate the challenge they have and their competence and progress managing it. Data like this, combined with the previous data points I discussed, can help illustrate where the risk is for an organization and help optimize its treatment. The number of unpatched, critical, high, and medium severity vulnerabilities in operating systems, web browsers, and applications in an environment, along with the number of systems not managed by the vulnerability management program, can help CISOs and their stakeholders understand the risk in their IT environment. Of course, depending on the environment, including data pertaining to cloud-based assets, mobile devices, hardware, firmware, appliances, routing and switch equipment, and other technologies that are in use in each IT environment will provide a more complete view. The mix of these technologies, and their underlying vulnerabilities, is unique to each organization.

Providing executive management teams and board members with quantitative data like this helps them understand reality versus opinion. Without this type of data, it can be much more difficult to make compelling business cases and communicate progress against goals for cybersecurity programs. This data will also make it easier when random executives and other interested parties, such as overly aggressive vendors, ask cybersecurity program stakeholders about the "vulnerability du jour" that makes it into the news headlines. If senior stakeholders know that their CISO and vulnerability management team are managing vulnerabilities and misconfigurations in their environment competently and diligently, a lot of noise that could otherwise be distracting to CISOs can be filtered out.

This reporting might sound complicated and intimidating to some. The good news is that there are vulnerability management products available that provide rich analytics and reporting capabilities. CISOs aren't limited to the ideas I've provided in this chapter, as vulnerability management vendors have lots of great ways to help measure and communicate progress. The key is to use analysis and reporting mechanisms to effectively show stakeholders how your vulnerability management program is reducing risk for the organization and to ask for resources when they are needed.

Although data from vulnerability management programs can be very helpful for CISOs, it only helps them manage two of the five cybersecurity usual suspects. There is potentially much more data that can help CISOs understand and manage the performance and efficacy of their cybersecurity strategies. Let's explore this next using the example I discussed at length in *Chapter 6*, *Strategy Implementation*, an Attack-Centric Strategy, the Intrusion Kill Chain framework (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.).

Measuring performance and efficacy of an Attack-Centric Strategy

As I mentioned in *Chapter 5*, *Cybersecurity Strategies* and *Chapter 6*, *Strategy Implementation*, the Intrusion Kill Chain framework has many attributes that make it an attractive cybersecurity strategy. First, it earned the highest **Cybersecurity Fundamentals Scoring System** (**CFSS**) estimated total score in *Chapter 5*.

This means it had the greatest potential to fully mitigate the cybersecurity usual suspects. Additionally, this approach can be used in on-premises environments and hybrid and cloud environments. Perhaps the thing I like most about this framework is that its performance and efficacy can be measured in a relatively straightforward way. Let's examine this in detail.

Performing intrusion reconstructions

This will likely seem odd when you read it, but when it comes to measuring the performance and efficacy of a cybersecurity strategy, intrusion attempts are gifts from attackers to defenders. They are gifts because they test the implementation and operation of defenders' cybersecurity strategies. But in order to derive value from intrusion attempts, every successful, partially successful, and failed intrusion attempt must be decomposed and studied. In doing this, there are two key questions to be answered. First, how far did attackers get with their Intrusion Kill Chain (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.) before they were detected and ultimately stopped? Second, how did attackers defeat or bypass all the layers of mitigating controls that the cybersecurity team deployed to break their Intrusion Kill Chain? Put another way, if attackers made it to phase four of their Intrusion Kill Chain how did they get past all the mitigations layered in phases one, two, and three?

These are the central questions that intrusion reconstructions (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.) should help answer. In seeking the answers to these two questions, intrusion reconstructions should also answer many other questions that will help measure the performance and efficacy of each implementation of this approach. As you'll see as I describe this process, the underlying theme of these questions is whether the people, processes, and technologies that are working to break attacker's Intrusion Kill Chains are effective. We want to uncover if any changes are required in each phase of our Attack-Centric Strategy. Let's get started.

The concept of intrusion reconstructions is discussed in Lockheed Martin's paper on Intrusion Kill Chains. Again, I recommend reading this paper. The approach I'll describe in this chapter is slightly different from the approach described in Lockheed Martin's paper. There are at least a few ways intrusion reconstructions can be done; I'll describe one way that I've used with some success in the past.

This approach assumes that defenders will not be able to perform attribution with any confidence, so it doesn't rely on attribution the way that other approaches might. I consider this an advantage as the likelihood of false attribution increases as attackers become more sophisticated. The goal of my approach to intrusion reconstructions is to identify areas where the implementation of the Intrusion Kill Chain framework can be improved, not identify attackers and take military or legal action against them.

Let me offer some advice on *when* to do intrusion reconstructions. Do not perform reconstructions while incident response activities are underway. Using valuable resources and expertise that have a role in your organization's incident response process, during an active incident, is an unnecessary distraction. The reconstruction can wait until periods of crisis have passed. Ideally, reconstructions can be done while the details are still fresh in participants' minds in the days or weeks after the incident has passed. However, if your organization is always in crisis mode, then ignore this advice and get access to people and information when you can. Maybe you can help break the crisis cycle by identifying what deficiencies are contributing to it.

To perform an intrusion reconstruction, I strongly suggest that you have at least one representative from all of the teams that are responsible for cybersecurity strategy, architecture, protection, detection, response, and recovery. In really large environments, this can be scoped to the relevant teams that were responsible for the areas involved in the intrusion attempt. Once the organization gets good at doing reconstructions, the number of participants can likely be reduced even more. But you need the expertise and visibility that each team has to reconstruct what happened during each failed, partially successful, and fully successful intrusion attempt. Remember that one of the modifications I made to the Courses of Action Matrix (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.) in *Chapter 6*, *Strategy Implementation* was adding a "data consumer point of contact" for each mitigation. This information can be helpful in identifying the right people from different teams to participate in reconstructions.

A decision should be made regarding whether to invite vendors to participate in these meetings. I typically found it helpful to have trusted representatives from some of the cybersecurity vendors we used participating in intrusion reconstructions.

There are at least a couple of benefits to this approach. First, vendors should be able to bring expertise around their products and services and provide insights that might otherwise be missed. Second, it's important to share the "gifts" that attackers give you with the vendors that you've selected to help you defend against them. These exercises can inform your vendors' efforts to make better products, which your organization and others can benefit from. But it also gives you the opportunity to see how helpful your vendors really are willing to be, and whether they are willing to be held accountable for their shortcomings. I found that some of the vendors I used, who I thought would have my back during security incidents, folded up like a circus tent and left town when I really needed them. During intrusion reconstructions, these same vendors had the courage to participate, but typically blamed their customers for their products' failure to perform as expected. If you do enough reconstruction exercises with vendors, you'll likely be able to determine whether they really have the desire and capability to help your organization in the way you thought they would. This knowledge comes in handy later when their product license renewal dates approach. I'll discuss this more later in this chapter.

All that said, inviting vendors to participate in reconstructions also has risk associated with it. Simply put, some vendors are really poor at keeping confidential information confidential. My advice is to discuss including vendors in these meetings, on a case by case basis, with the stakeholders that participate in the reconstruction exercises. If a vendor adds enough value and is trustworthy, then there is a case for including them in these exercises. Discussing this idea with senior leadership for their counsel is also a prudent step, prior to finalizing a decision to include vendors in these exercises.

If your organization has a forensics team or uses a vendor for forensics, these experts can be incredibly helpful for intrusion reconstruction exercises. The tools and skills they have can help determine if systems in the reconstruction have been compromised, when, and likely how. In my experience, I've come across two flavors of forensics teams. The first is the traditional forensics team, which has certified forensics examiners who follow strict procedures to maintain the integrity of the evidence they collect.

In my experience with organizations that have this type of forensics team, they have the need for a full-time team of experts that can preserve evidence, maintain the chain of custody, and potentially testify in court in the criminal matters they help investigate. More often, organizations outsource this type of work.

The other flavor of forensics team, that I see much more often, perform a different function and are sometimes simply referred to as Incident Responders. They too seek to determine if systems have been compromised. But these teams typically do not have certified forensics professionals, do not maintain integrity of evidence, and do not plan to testify in a court of law. In fact, many times, their efforts to determine if a system has been compromised results in destroying what would be considered evidence in a criminal proceeding. This is where I've encountered interesting and sometimes provincial attitudes among certified forensics experts, as many of them wouldn't call these efforts *forensics* at all because they destroy evidence rather than properly preserve it. But these folks need to keep in mind that many engineers that wear pinky rings (Order of the Engineer, n.d.) resent IT engineers using "engineer" in their titles; architects that design buildings don't like IT architects using their title either, and the title "security researcher" makes many academic researchers cringe. But I digress. The reality is, not every organization wants to spend time and effort tracking down attackers and trying to prosecute them in a court of law. Organizations need to decide which flavor of forensics professionals they need and can afford. Both types of forensics experts can be worth their weight in gold, when they help determine if systems have been compromised and participate in intrusion reconstruction exercises.

Who should lead reconstruction exercises? I recommend that the individual or group responsible for cybersecurity strategy leads these exercises. This individual or group is ultimately responsible for the performance and efficacy of the overall strategy. They are also likely responsible to make adjustments as needed to ensure the success of the strategy. An alternative to the strategy group is the **Incident Response** (**IR**) team. The IR team should have most, if not all, of the details required to lead an intrusion reconstruction (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.). If they don't, you've just identified the first area for improvement.

The IR team manages incidents, so they really should have most of the information related to partially and fully successful intrusion attempts at their fingertips. But they might not be involved in failed attempts that don't qualify as incidents. In these cases, SOC personnel, operations personnel, and architects likely have key information for the reconstruction.

Keep in mind that the goal isn't to triage every port scan that happens on the organization's internet-facing firewalls. I suggest getting agreement among the groups that will participate in reconstruction exercises most often on a principle that is used to determine the types of intrusions that reconstructions should be performed on. That is, define the characteristics of intrusion attempts that determine whether a formal reconstruction is performed. As shown in *Table 7.1*, using our updated Courses of Action Matrix from *Chapter 6*, *Strategy Implementation*, an effective principle could be that any intrusion that makes it further than the *Deny* action in the *Delivery* phase should be reconstructed. A much less aggressive principle could be that any intrusion attempt that results in a *Restore* action should be reconstructed. There are numerous other options between these two examples.

The goal of such a principle is to impose consistency that helps appropriately balance risk and the valuable time of reconstruction participants. This principle doesn't need to be chiseled into stone—it can change over time. When an organization first starts performing reconstructions, they can have a relatively aggressive principle that enables them to learn quickly. Then, once lessons from reconstructions have "normalized" somewhat, a less aggressive principle can be adopted. But getting agreement among the stakeholders in these reconstruction exercises on the principle used to initiate them is important for their long-term success, and therefore the success of the cybersecurity strategy. Too few reconstructions relative to intrusion attempts could mean the organization isn't paying enough attention to the gifts it's being given by attackers, and is potentially adjusting too slowly to attacks. Too many reconstructions can be disruptive and counterproductive. The agreed upon principle should strike the right balance for the stakeholder community over time.

Kill Chain Phase	Detect	Deny	Disrupt	Degrade	Deceive	Limit	Restore
Reconnaissance I							
Delivery							
Exploitation							
Installation							
Command and Control (C2)							
Reconnaissance II							
Actions on Objectives							

Table 7.1: An example of an updated Course of Action Matrix from Chapter 6, Strategy Implementation (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.)

Once the appropriate participants, or their delegates, have been identified, and an intrusion reconstruction leader is ready to facilitate, a reconstruction meeting can be scheduled. Providing participants enough lead time and guidance to gather the appropriate data for a reconstruction will help save time and frustration. In my experience, some reconstruction exercises are straightforward because the intrusion attempt was detected and stopped in an early phase. In these cases, the number of participants and the amount of data they need to reconstruct the intrusion attempt can be relatively minor. Subsequently, the amount of time typically needed for this exercise is relatively short, such as 45 minutes or an hour, for example. If you are just starting to do reconstructions in your organization, you'll naturally need a little more time than you'll need after becoming accustomed to them. For more complicated intrusion attempts, especially when attackers make it to later stages of their Kill Chain, more participants with more data might be required, increasing the amount of time needed for reconstruction exercises.

Many of the organizations I've worked with label security incidents with code names. All subsequent communications about an incident uses its codename. This way, if an email or other communications are seen by someone who has not been read into the incident, its context and significance is not obvious. Communications about, and invitations to, intrusion reconstructions should use incident code names when organizations label incidents with them. If you decide to use incident code names, be thoughtful about the names you use, avoiding labels that are potentially offensive. This includes names in languages other than English.

Consider the potential impact to the reputation of the organization if the code name ever became public knowledge. Stay away from themes that are inconsistent with the organization's brand or the brand it aspires to build in the mind of their customers. There really is no compelling business reason to use anything but benign codenames. These are boring, but effective on multiple levels.

Now we have a codename for our reconstruction exercise, participants that are going to bring relevant data, potentially some trustworthy vendors that will participate, and a leader to facilitate the exercise. The point of the exercise is to reconstruct the steps that attackers took in each phase of their Kill Chain. It might not be possible to do this with complete certainty, and some assumptions about their tactics and techniques might be necessary. But the more detail the reconstruction can include, the easier it will be to identify areas where people, processes, and technologies performed as expected or underperformed. Be prepared to take detailed notes during these exercises. A product of intrusion reconstruction exercises should be a report that contains the details of the intrusion attempt, as well as the performance of the defenses that the cybersecurity team had in place. These artifacts will potentially have value for many years as they will provide helpful continuity of knowledge about past attacks, even when key staff leave the organization. Put another way, when the lessons learned from these intrusion attempts are documented, they are available for current and future personnel to learn from. This is another reason I call intrusion attempts "gifts".

Our updated Kill Chain framework has seven phases. Where should a reconstruction exercise start? In the first phase, or perhaps the last phase? The answer to this question is, it depends. Sometimes, an intrusion is straightforward and can be charted from beginning to end in sequential order. However, with complicated intrusions or intrusions that started months or years earlier, it might not be prudent or possible to approach a reconstruction that way. Start with the phase that you have the best information on and most certainty about. This could be late in the Kill Chain. From your starting point, build a timeline in both directions, using the data and insights that the reconstruction participants can offer. It might not be possible to build the entire timeline because of a lack of data, or because of uncertainty.

The more details the reconstruction uncovers, the better, as this will help identify opportunities for improvement, gaps, and failures in defenses. In my example, I will simply start at the first phase and work forward through the Kill Chain. But just be aware that this might not be possible to do for every intrusion. Let's start with the **Reconnaissance I** phase.

It might not be possible to attribute any particular threat actor's activities in the Reconnaissance I phase, prior to their attack. With so much network traffic constantly bombarding all internet-connected devices, it is typically challenging to pick out specific probes and reconnaissance activities conducted by specific attackers. But it's not impossible. This is an area where the combination of **Artificial Intelligence** (**AI**), **Machine Learning** (**ML**), good threat intelligence, and granular logs is very promising. Using AI/ML systems to churn through massive amounts of log data, such as network flow data, DNS logs, authentication and authorization logs, API activity logs, and others, in near real-time to find specific attackers' activities is no longer science fiction. Cloud services can do this today at scale. The icing on the cake is that you can get security findings read to your SOC analysts by Amazon Alexa (Worrell, 2018)! These are the types of capabilities that, until recently, were only possible in science fiction. But now, anyone with a credit card and a little time can achieve this with capabilities that cloud computing provides. Truly amazing! I'll discuss cloud computing more in the next chapter.

Collecting data and insights from the **Delivery** phase of an attack is obviously super important. The key question is, how did the attackers defeat or bypass the layers of mitigations that the cybersecurity team deployed to break this phase of their Kill Chain? How did they successfully deliver their weapon and what people, processes, and technologies were involved?

To answer these questions, I have found it useful to draw system flow charts on a whiteboard during the reconstruction exercise with the participants' help. Start by drawing the infrastructure that was involved with as much detail as possible, including perimeter defenses, servers, clients, applications, system names, IP addresses, and so on. Draw a map of the infrastructure involved and chart how data is supposed to flow in this infrastructure, protocols used, authentication and authorization boundaries, identities involved, storage, and so on. Then, draw how the attackers delivered the weapon during their intrusion attempt and what happened during delivery.

What enabled the attacker's success in this phase? The answer to this question involves asking and answering numerous other questions. Let me give you some examples. A useful data point in an intrusion reconstruction is how long it took for the attack to be detected. Building an attack timeline can be a useful tool to help determine how an attack was executed. In the context of the **Delivery** phase (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.), was the delivery of the weapon detected, and what control detected it? If delivery wasn't detected, document which controls were supposed to detect it. If there is a clear gap here in your implementation of the Kill Chain framework document that. This information will be very useful later when you remediate deficiencies in the implementation of the strategy.

Were there any controls that should have detected delivery, but failed to do so? Why did these controls fail to operate as expected? Did they fail because they simply did not do what the vendor said they would do? Did they fail because of integrations or automation between controls, or systems did not work as intended? This is where log data and other sources of data from systems in the reconstruction flow chart can be very helpful. Try to piece together how the weapon was delivered, step by step, through data in logs of various systems in the flow chart. Does it look like all these systems performed as expected? If not, identify anomalies and the weak links. In some cases, log data might not be available because logging wasn't turned on or aggressive data retention controls deleted the log data. Is there a good justification for not enabling logging on these systems and storing logs to help in the future?

Was there enough data to determine how the weapon was delivered? Sometimes, it's simply not possible to determine how the weapon was delivered with the data that is available. Some IR teams refer to the first system that was compromised in an intrusion as "patient zero". In some intrusions, the attacker's entry point is very obvious and can be tracked back to an email, a visit to a malicious website, a USB drive, malware, and so on. In other cases, if the initial compromise was achieved weeks, months, or years earlier, and attackers were adept at covering their tracks, finding patient zero is aspirational, and simply might not be possible. Think about what would have helped you in this scenario. Would increasing the verbosity of logging have helped? Would archiving logs for longer periods or shipping logs offsite have helped? Is there some capability that you don't currently have that would have helped fill this gap?

Did the data consumers for the Delivery phase mitigations get the data they needed to detect and break this phase? For example, did the SOC get the data they needed to detect intrusion? Did the data consumers identified in the updated Courses of Action Matrix receive or have access to the data as intended? If not, what went wrong? Did the data delivery mechanism fail, or was the required data filtered out at the destination for some reason? There could have been multiple failures in the collection, delivery, and analysis of the data. Dig into this to identify the things that did not work as planned and document them.

Did the controls, automation and integrations work as expected, but people or processes were the source of the failure? This scenario happens more than you might think. The architecture was sound, the systems worked as expected, the technologies performed as expected, the weapon was detected, but no one was paying attention, or the alert was noticed but was dismissed. Unfortunately, people and process failures are as common, if not more common, than technical control failures. Failures in SOC processes, poor decision-making, vendors that make mistakes, and sometimes just laziness among key personnel can lead to failures to detect and break attacks.

Did attackers and/or defenders get lucky anywhere in this phase of the attack? Some security professionals I've met have told me they don't believe in luck. But I attribute this belief to naivety. I've seen attacks succeed because of a comedy of errors that likely could not be repeated or duplicated. Combinations of people, processes, technologies, and circumstances can lead to attack scenarios as likely as winning a lottery. Don't discount the role that luck can play. Remember that not all risks can truly be identified; "black swan" events can happen (Taleb, 2007).

Once the reconstruction team understands how the Delivery phase of the attack was accomplished and this has been documented, we can move on to the next phase of the attack, the **Exploitation** phase (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.). Here, the reconstruction team will repeat the process, using data to try to determine if exploitation was attempted, detected and stopped. The same questions we asked for the Delivery phase apply in this phase as well. What controls failed to prevent and detect exploitation? Where did gaps in protection and detection controls contribute to attacker success in this phase of their attack?

Did vendors' cybersecurity mitigations work as advertised? Did data consumers get the data they required to detect and break this phase? Did the IR process start and work as planned? What can we learn from attackers' success in this phase to make such success harder or impossible in the future? Document your findings.

Continue to perform this investigation for all the phases of the Kill Chain. There might be phases where nothing occurred because attackers were stopped prior to those phases. Note where and when the attack was successfully detected and successfully broken. If the attack had not been broken in the phase it was, would the mitigations layered in later phases have successfully detected and stopped the attack? Be as candid with yourselves as possible in this assessment; platitudes, optimism, and plans in the undefined future may not be enough to break the next attacker's Intrusion Kill Chain. However, sober determination to make it as difficult as possible for attackers can be helpful. Remember to document these thoughts.

Now the reconstruction is complete, and you have asked and answered as many questions as needed to uncover what happened, ideally in every step of the attack. Next, let me provide some examples of the specific actionable things the reconstruction should have identified in the wake of failed, partially successful, and fully successful attacks.

Using intrusion reconstruction results

First, recall the discussion on identifying gaps and areas of over and under investment in *Chapter 6*, *Strategy Implementation*. An intrusion reconstruction can confirm some of the analysis on gaps and under investments that were done while planning the implementation of this strategy. For example, if a gap in detection in the Delivery phase was identified during planning and later intrusion reconstruction data also illustrates this same gap, this is strangely reassuring news. Now, the CISO has more data to help build the business case for investment to mitigate this gap. It's one thing for a CISO to say they need to invest in detection capabilities or bad things can happen. But such requests are much more powerful when CISOs can show senior executives and the Board of Directors that attackers have been actively using known gaps.

It counters any notion that the risk is theoretical when CISOs can provide evidence that the risk is real. It also helps build a sense of urgency where there was none before. If the intrusion attempt led to unplanned expenses related to response and recovery activities, this will help illustrate the current and potential future costs related to the gap. This data can inform both the probability and the impact sides of the risk equation, making it easier to compare to other risks. Using data like this, CISOs can give their management boards updates on gaps and under investment areas at every cybersecurity program review meeting until they are mitigated.

When reconstruction exercises uncover previously unknown gaps or areas of under investment, this truly is a gift from attackers. In doing so, attackers provide CISOs valuable insights into deficiencies in the implementations of their strategies, as well as a clear call to action to implement new mitigations or improve existing ones. Intrusion reconstruction data can also help to inform cybersecurity investment roadmaps. Remember that stopping attackers as early in the Intrusion Kill Chain as possible is highly preferable to stopping them in later phases. Reconstruction data can help cybersecurity teams identify and prioritize mitigations that will help make it harder or impossible for attackers to make it to later phases of their attack. Helping cybersecurity teams understand deficiencies and areas for improvement in the *Delivery* and *Exploitation* phases is a key outcome of intrusion reconstruction exercises. This data can then be used to plan the investment roadmap that charts the people, processes, and technologies the organization plans to deploy and when. Since most organizations have resource constraints, reconstruction data and the investment roadmaps they inform can become central to a cybersecurity team's planning processes.

Remember those cybersecurity imperatives and their supporting projects I discussed in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*? An imperative is a big audacious multi-year goal, ideally aligned with the organization's business objectives. Upgrading to a much-needed modern identity system or finally getting rid of VPN in favor of modern remote access solutions for thousands of Information Workers are two examples. Reconstruction data can help provide supporting data for cybersecurity imperatives and provide a shared sense of purpose for the staff that work on them. Conversely, reconstruction data might not always support the notion that planned imperatives are the right direction for the organization.

There's no expectation that these will necessarily align, especially in large organizations with complex environments and multiple imperatives. But when lightning strikes and reconstruction data suggests that an imperative is critical to the organization, it can supercharge the project teams that are working on it. This type of positive momentum can be beneficial by helping to maintain project timelines and getting projects across their finish lines.

Identifying lame controls

Another potential action area stemming from an intrusion reconstruction is correcting mitigations that failed to perform as expected. These are controls that have been deployed and are actively managed, but did not protect, detect, or help respond to the intrusion attempt as designed. To state the obvious, CISOs and security teams can't rely on controls that don't work the way they should. There are a range of possible root causes for controls that fail.

A common root cause for failure is that the control doesn't actually perform the function that the security team thought it did. Mismatches between security controls' functions and security teams' expectations are, unfortunately, very common. Some controls are designed to mitigate very specific threats under specific circumstances. But such nuances can get lost in vendors' marketing materials and sales motions. This is a critical function that architects play on many cybersecurity teams: to really understand the threats that each control mitigates and how controls need to be orchestrated to protect, detect, and respond to threats to their organizations. They should be thoughtfully performing the cybersecurity capabilities inventories I discussed in *Chapter 6*, *Strategy Implementation* and making changes to those inventories to minimize gaps and areas of under investment. But, as I also mentioned in *Chapter 6*, the maturity of the controls' implementation is an important factor, as is the consumption of the data generated by controls. This is something architects can have a hand in, that is, inventorying and planning, but data consumers, operations personnel, and SOC engineers, among others, need to help complete this picture. Otherwise, mismatches between control functions and expectations can burn the cybersecurity team.

Another common cause for mitigations failing to perform as expected is they simply don't work the way vendors say they work. I know this is a shocking revelation for few people, and it's an all too common challenge for security teams. If vendors kept all their promises, then there wouldn't be a global cybersecurity challenge, nor would there be a multi-billion-dollar cybersecurity industry. This is one reason it is prudent to have layers of defenses, so that when one control fails, other controls can help mitigate the threat. This is an area where CISOs can share and learn a lot from other CISOs. Professional experiences with specific vendors and specific products are often the best references to have.

Another common reason for mitigations failing to protect, detect, or respond, is that the trusted computing base that they rely on has been compromised. That is, attackers have undermined the mitigations by compromising the hardware and/or software they depend on to run. For example, one of the first things many attackers do once they use one or more of the cybersecurity usual suspects to compromise a system is disable the anti-malware software running on it. A less obviously visible tactic is to add directories to the anti-malware engine's exceptions list so that attacker's tools do not get scanned or detected. Once attackers or malware initially compromise systems, it is common for them to undermine the controls that have been deployed to protect systems and detect attackers. Therefore, becoming excellent at the cybersecurity fundamentals is a prerequisite to deploying advanced cybersecurity capabilities. Don't bother deploying that expensive attacker detection system that uses AI to perform behavioral analysis unless you are also dedicated to managing the cybersecurity fundamentals for that system too. Attackers will undermine those advanced cybersecurity capabilities if unpatched vulnerabilities, security misconfigurations, and weak, leaked, or stolen passwords enable them to access the systems they run on. I discussed this at length in earlier chapters, but I'll reiterate here again. No cybersecurity strategy, not even a high scoring strategy like the Intrusion Kill Chain framework, will be effective if the cybersecurity fundamentals are not managed effectively.

Additionally, it's important that the cybersecurity products themselves are effectively managed with the cybersecurity fundamentals in mind. Anti-malware engines and other common mitigations have been sources of exploitable vulnerabilities and security misconfigurations in the past. They too must be effectively managed so that they don't increase the attack surface area instead of decreasing it.

Another action item, related to failed controls, that can emerge from reconstruction exercises is addressing control integrations that failed. For example, an intrusion attempt wasn't detected until relatively late in an attacker's Kill Chain because, although a control successfully detected it in an earlier phase, that data never made it to the SIEM. Broken and degraded integrations like this example are common in large complex IT environments and can be difficult to detect. It would be ideal if cybersecurity teams could simply rely on data consumers to identify anomalies in data reporting from cybersecurity controls, but in many cases, the absence of data isn't an anomaly. Technical debt in many organizations can make it challenging to identify and remediate poor integrations. Many times, such integrations are performed by vendors or professional services organizations who have limited knowledge of their customers' IT environments. This is where SOC engineers can be valuable; they can help ensure integrations are working as expected and improve them over time.

Learning from failure

In addition to identifying gaps and suboptimal controls and integrations, intrusion reconstructions can help CISOs and cybersecurity teams confirm that they have the right investment priorities. Data from reconstructions can help re-prioritize investments so that the most critical areas are addressed first. Not only can this data help rationalize investment decisions, it can also help CISOs justify their investment decisions, especially in the face of criticism from CIOs and CTOs who have different opinions and possibly differing agendas. Investing in areas that break attackers' efforts instead of new capabilities that IT has dependencies on, might not be a popular choice among IT leadership. But using reconstruction data to defend such decisions will make it harder for others to disagree.

Besides identifying technologies that didn't work as expected, reconstructions can provide an opportunity to improve people and processes that performed below expectations. For example, in cases where lapses in governance led to poor security outcomes, this can be good data to help drive positive changes in governance processes and associated training. If complying with an internal standard or an industry standard wasn't helpful in protecting, detecting, or responding to an attack, reconstructions might be an impetus for change.

Allowing people in the organization to learn from failure is important. After spending time and effort to understand and recover from failures, organizations can increase their return on these investments by disseminating lessons from failures to the people in the organization who will benefit the most from them. Reconstruction data can help build a case for social engineering training for executives or the entire organization, for example.

Identifying helpful vendors

Vendors are important partners for organizations as they typically provide technologies, services, people, and processes that their customers rely on. Intrusion reconstruction data can help identify vendors that are performing at or above expectations. It can also help identify vendors that are failing to perform as expected. This includes how vendors participate in intrusion reconstruction exercises themselves. Reconstruction exercises can help reveal those vendors who tend to blame their customers for failures in their products' and services' performance, which is rarely helpful. This, along with data on how vendors' products and services performed, can help inform vendor product license renewal negotiations. Once security teams get a taste of how the vendors' products really perform and how helpful they are willing to be during intrusions, they might be willing to pay much less for them in the future, or not willing to use them at all. If your organization doesn't already do this, I suggest maintaining a license renewal and end-of-life "horizon list" that shows you when key dates related to renewals and products' end of life are approaching.

Ensure the organization gives itself enough prior notice so they can spend a reasonable amount of time to re-evaluate whether better mitigations now exist. After deploying and operating vendors' products, the organization likely has much more data and better data on their current vendors' performance to inform product evaluations than they did when they originally procured them.

Reward the vendors who are helpful and consider replacing vendors that don't understand their core value is supposed to be customer service. Looking at all the vendors I mentioned in *Chapter 6*, *Strategy Implementation*, in addition to all the vendors I didn't mention, there is no shortage of companies competing for your organization's business. Don't settle for vendors that blame your organization for their failures. Even if it is true, they should be helping you overcome these challenges instead of playing the blame game. Intrusion reconstruction exercises are their opportunity to prove they are invested in your success, instead of being an uninterested third party on the sidelines, waiting for the next license renewal date. If they have been trying to help your organization get more value out of their products, but your organization hasn't been receptive, then this should be reconciled prior to making rash decisions. Replacing good vendors that have been constantly swimming upstream to help your organization doesn't help you and could set your cybersecurity program back months, or even years. But their products either work as advertised and they are willing to help you get them into that state in a reasonable period of time, or they should be replaced. Otherwise, they just increase the attack surface area while using resources that could be used elsewhere to better protect, detect, and respond to threats.

Reconstruction data is likely the best data you'll have to truly gauge your cybersecurity vendors' performance. Use it in license renewal negotiations to counter marketing fluff and sales executives' promises that the latest version or the next version solves all your challenges, including their inability to provide essential levels of customer service. Sometimes, desperate vendors, sensing they are going to lose business, decide to "end run" the CISO and cybersecurity team by appealing directly to other executives or the Board of Directors. This can turn out to be suboptimal for CISOs that get saddled with products that don't help them.

But it's harder for executives and the Board to award more business to such vendors when the CISO has been briefing them on intrusion reconstruction results, as well as showing them how helpful or unhelpful some of their vendors have been. If executives still decide to award more business to vendors who, the data indicates, have not been performing to expectations, they have decided to accept risk on behalf of the entire organization. CISOs get stuck managing this type of risk all the time. But as the data continues to mount, it will become harder for everyone to simply accept the status quo. Data instead of opinion alone should help organizations make better decisions about the cybersecurity capabilities they invest in.

Informing internal assessments

The last potential action item area stemming from the results of intrusion reconstructions that I'll discuss is penetration testing and Red/Blue/Purple Team exercises. Many organizations invest in penetration testing and Red/Blue/Purple Teams so that they can simulate attacks in a more structured and controlled way. Lessons from intrusion reconstruction exercises can inform penetration testing and Red Team/Purple Team exercises. If reconstruction exercises have uncovered weaknesses or seams that attackers can use in an implementation of a cybersecurity strategy, these should be further tested until they are adequately addressed. When professional penetration testers and Red Teams are provided with intrusion reconstruction results, it can help them devise tests that will ensure these weaknesses have been properly mitigated. Ideally, penetration testers and Red/Blue/Purple Teams find implementation deficiencies before attackers get the chance to.

Chapter summary

Cybersecurity teams need to measure many different things for a range purposes including complying with regulatory, industry, and internal standards. However, this chapter focused on how CISOs and cybersecurity teams can measure the performance and efficacy of the implementation of their cybersecurity strategy, using an Attack-Centric Strategy as an example.

Data helps CISOs manage their cybersecurity programs and investments and helps them prove that their cybersecurity program has been effective and constantly improving; it can also help illustrate the effectiveness of corrective actions after issues are detected. A well-run vulnerability management program is not optional; leveraging data from it represents one of the easiest ways for CISOs to communicate effectiveness and progress. Vulnerability management teams should scan everything in their inventories every single day for vulnerabilities and misconfigurations. This helps minimize the amount of time that unmitigated vulnerabilities and misconfigurations are present and exploitable. Valuable trend data can emerge from vulnerability management scanning data over time. Some examples of valuable data include:

- The number of assets under the management of the vulnerability management team versus the total number of assets that the organization owns and operates.
- The number of vulnerabilities unpatched in the environment by vulnerability severity.
- Vulnerabilities by product type can help illustrate where the most risk exists in an environment; the number of unpatched, critical, high, and medium severity vulnerabilities in operating systems, web browsers, and applications in an environment, along with the number of unmanaged systems, can help CISOs and their stakeholders understand the risk in their IT environment.

Attack-Centric strategies, like the Intrusion Kill Chain, make it relatively easy to measure performance and efficacy; to do this, intrusion reconstructions are used. Intrusion reconstruction results can help CISOs in many different ways, not least by identifying mitigations that failed to perform as expected. To derive value from intrusion attempts, every successful, partially successful, and failed intrusion attempt must be decomposed and studied to answer two key questions:

1. How far did attackers get with their Intrusion Kill Chain before they were detected and ultimately stopped?
2. How did attackers defeat or bypass all the layers of mitigating controls that the cybersecurity team deployed to break their Intrusion Kill Chain, before they were stopped?

In the next chapter of this book, we will look at how the cloud can offer a modern approach to security and compliance and how it can further help organizations with their cybersecurity strategy.

References

1. Order of the Engineer (n.d.). Retrieved from Order of the Engineer: `https://order-of-the-engineer.org/`
2. CVE Details (2019). *Top 50 Products By Total Number Of "Distinct" Vulnerabilities*. Retrieved from CVE Details: `https://www.cvedetails.com/top-50-products.php`
3. Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D. (n.d.). *Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains*. Retrieved from Lockheed Martin: `https://lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/LM-White-Paper-Intel-Driven-Defense.pdf`
4. Herrmann, D. S. (2007). *Complete Guide to Security and Privacy Metrics: Measuring Regulatory Compliance, Operational Resilience, and ROI*. Auerbach Publications.
5. ISC2 (2020). *CISSP Domain Refresh FAQ*. Retrieved from ISC2 Certifications: `https://www.isc2.org/Certifications/CISSP/Domain-Refresh-FAQ#`
6. Taleb, N. N. (2007). *The Black Swan: The Impact of the Highly Improbable*. Penguin Books.
7. Worrell, C. (April 3, 2018). *How to Use Amazon Alexa to Get Amazon GuardDuty Statistics and Findings*. Retrieved from AWS Security Blog: `https://aws.amazon.com/blogs/security/how-to-use-amazon-alexa-to-get-amazon-guardduty-statistics-and-findings/`

8

The Cloud – A Modern Approach to Security and Compliance

The cloud offers a modern approach to security and compliance. This chapter will introduce some concepts that will help put the cloud into context for CISOs and Security and Compliance professionals who haven't fully embraced it yet.

Throughout this chapter, we'll cover the following topics:

- The power of Application Program Interfaces
- The advantages of automation to help mitigate the cybersecurity usual suspects
- Cybersecurity strategies in the cloud
- Encryption and key management

Let's begin by looking at how the cloud is different from what we've been doing on-premises.

Introduction

The emergence of commercial cloud computing in 2006 led to a lot of debate among some organizations as to whether the cloud could be trusted, as well as whether it is as secure as on-premises IT environments. However, for many organizations, the cloud represents much more than new technology. Simply put, the cloud represents change. Let's face it, change is easy for some organizations, like startups, while it can be more difficult for large, well-established and highly regulated organizations, such as financial services institutions or some verticals in the public sector.

Very often, it's the CISO in these organizations who is change averse, operating as if the ideal outcome is a stalemate with attackers, in IT environments where CISOs have some control over change. As long as nothing changes, they can maintain this state of relative success and continue to improve. However, of course, things are constantly changing; it just takes time for us busy humans to notice it. Businesses that don't keep pace with technological advancements fall behind their competitors and fall prey to the startups seeking to disrupt their industry – the wolf is always at the door. However, CISOs can't be faulted for hoping to maintain the status quo when they have been successful. However, CISOs that don't spend some of their time pretending to be a CTO can do their organizations a disservice by slowing them down too much and hampering innovation.

This doesn't mean that CISOs can, or should, advocate for the adoption of every new technology that appears on the horizon. However, after more than a decade of being debated, the verdict is clear – the cloud is a game changer for security and compliance professionals. This chapter will provide an overview of how the cloud is the great cybersecurity talent amplifier that can help organizations execute on their current cybersecurity strategy or even embrace a more modern approach to security and compliance. Let's start with a quick introduction to cloud computing.

How is cloud computing different?

Among **Cloud Service Providers** (**CSPs**) such as IBM, Oracle, Alibaba and others, the three most popular CSPs in the world are **Amazon Web Services** (**AWS**), Microsoft and Google. These CSPs are often referred to as hyperscale CSPs because their cloud offerings are available all over the globe.

When organizations first contemplate leveraging services offered by CSPs, the first topics some of them want to explore are security and compliance. They need to understand how CSPs can provide the IT capabilities they need, while meeting or exceeding industry security standards, regulated standards and their own internal security standards. I've heard a lot of myths about cloud computing and I've seen the cloud help organizations achieve things they couldn't possibly achieve in their own on-premises IT environments. I'll share some of the things I've learned about the cloud in this chapter, but please note that all the views and opinions written in this chapter, as well as the rest of this book, are my own personal opinions and not those of any of my past or present employers. Let's get started.

Although cloud computing is being adopted by industries all over the world, this has happened unevenly and more slowly in some regions of the world. As cloud computing started to get traction with enterprises, service model descriptions made it easy to educate people on what the cloud is and what it isn't. Three cloud computing service models became popular, including **Infrastructure as a Service** (**IaaS**), **Platform as a Service** (**PaaS**) and **Software as a Service** (**SaaS**). These service model descriptions made it easier for everyone to understand the types of cloud services available and where they might fit into each organization's IT portfolio. For example, organizations could run their own virtual servers in a CSP's IaaS offering, such **Amazon Elastic Compute Cloud** (**Amazon EC2**), Microsoft Azure Virtual Machines, or Google Compute Engine.

CSPs offer services based on massive physical IT infrastructures that they've built around the world. Over time, the physical infrastructure model that CSPs have roughly coalesced around is based on the model that AWS pioneered; the concept of regions and availability zones. In AWS parlance, an availability zone is a cluster of datacenters and a region is a cluster of availability zones. There are meaningful differences in the size and scope of CSPs' infrastructures and how they leverage components of this model. You can learn about each CSP's infrastructure on their websites:

- **AWS**: `https://aws.amazon.com/about-aws/global-infrastructure/`
- **Google**: `https://cloud.google.com/compute/docs/regions-zones/`
- **Microsoft**: `https://azure.microsoft.com/en-us/global-infrastructure/regions/`

Although the terms IaaS, PaaS and SaaS are still in widespread use today, they are slowly becoming obsolete. Back when they were first coined, CSPs only had a handful of services and these service models helped describe how each service was deployed. However, this has been changing rapidly. At the time of writing, the three aforementioned hyperscale CSPs offered hundreds of cloud services. Subsequently, newer acronyms like **Containers as a Service** (**CaaS**), **Identity as a Service** (**IDaaS**) and **Function as a Service** (**FaaS**) have cropped up. This proliferation of services has been accelerating because the developers of new services can use existing services as building blocks. For example, if a CSP is developing a new cloud service and they need storage for it, instead of building a new storage infrastructure from scratch, they can simply use one of the existing cloud storage services that meets their requirements. Not only does this approach help accelerate the development of new cloud services, but it means services could have IaaS, PaaS and/or SaaS components, blurring the lines between these old service model descriptions. In other words, solutions to specific problems are becoming more important than maintaining service model definitions. As this cloud services proliferation continues, enterprises will be able to procure solutions for the specific problems that they want to solve and the old service models will become less and less important.

One important distinction when it comes to service models is the difference between the services that hyperscale CSPs provide and that of traditional **Managed Service Providers** (**MSPs**). Many organizations around the world have leveraged MSPs for decades. Governments, for example, tend to sign very long-term agreements with MSPs to manage their datacenters and provide IT services to them. MSPs have played an important role for such organizations, for at least a couple of reasons. First, MSPs have successfully maintained a critical mass of IT talent that would otherwise be challenging for enterprises to attract and retain themselves. Second, MSPs became intimately familiar with the IT environments of their customers because they managed them; this tribal knowledge provided the continuity that enterprises needed in order to minimize potential disruptions when key staff turned over. Of course, MSPs offer other benefits to their customers as well.

More and more organizations want to move from a CAPEX model to an OPEX model, where they don't have to make large investments up front and hope their capacity and utilization estimates are correct; paying for just the specific resources that they use is more efficient. MSPs and CSPs can help their customers with this shift. However, MSPs tend to have an outsourcing-based business model, where CSPs offer a self-service model for transformation instead of replication of existing processes.

One mistake that is easy to make for enterprises that contemplate using the cloud for the first time is the assumption that CSPs are just another flavor of MSP. They aren't. Hyperscale CSPs offer an extremely scalable and agile self-service IT model where their customers only pay for what they use, measured in compute seconds and the amount of data they store or transfer across networks. Anyone with a credit card can open an account and get access to hundreds of services that would be prohibitively expensive to build in on-premises or MSP IT environments. When customers are finished using their CSP's services, they can typically walk away from them with no obligations whatsoever.

Conversely, MSPs manage datacenters and systems on behalf of their customers. Because of the up-front investments required to physically build datacenters and the systems that run in them, the MSP model typically requires long-term contracts that ensure MSPs can derive appropriate returns on their investments. This model puts MSPs and their customers at a disadvantage. CSPs spread their expenses across millions of customers around the world, where MSPs tend to have a much smaller set of customers to service, who must pay for everything themselves. Some MSPs have built their customers their own private clouds, which seek to mimic the elasticity and the other characteristics of cloud computing. However, in my experience, the term *private cloud* is a euphemism for limited scale, limited services and slow to change. In some cases, a private cloud is simply just an outsourced datacenter. Comparing these to the range of services that hyperscale CSPs offer isn't really an apples to apples comparison. Subsequently, many MSPs have evolved their products and services to run on top of CSP's services. This makes a lot of sense, as they too can benefit from the scale of economies that the hyperscale CSPs provide.

They do this by dramatically reducing capital expenses, getting virtually unlimited scale for their products and enabling them to embrace an incredible pace of innovation that they could likely not achieve themselves. There is a huge opportunity for MSPs to design, build and manage systems for their customers. However, instead of focusing on IT infrastructure administration, they can focus more on innovation. They can also achieve better security for their customers. I'll discuss some of the ways the cloud can provide better security and compliance in this chapter.

The failure to understand the difference between CSPs and MSPs can slow organizations down when they evaluate the security and compliance of the cloud. Many organizations spend an inordinate amount of time trying to understand how they maintain the status quo if they choose to leverage the cloud. However, as I mentioned earlier, the cloud represents change; reconciling these two things is one of the first things organizations are confronted with when they first contemplate using the cloud. This reconciliation can manifest itself several different ways. Let me give you a couple of examples.

As I mentioned earlier, as a group, hyperscale CSPs offer hundreds of services to their customers. Despite this, many enterprises still choose to *lift and shift* applications into the cloud. This typically means that they take an application they have been running on servers in their on-premises IT environment and run it on servers hosted in the cloud. This type of transition to the cloud allows them to maintain the people, processes and technologies that they have been using, while moving from CAPEX to OPEX, in many cases, for years. For many organizations, this is completely natural as they have deep expertise building and managing these systems in their on-premises IT environment and they can continue to leverage this expertise when they move those same systems into the cloud. In the cloud, they can leverage the same or similar hardware and software that they have been using on-premises. Subsequently, this type of transition can be relatively easy and quick.

The challenge with lifting and shifting applications is that complexity, inefficiencies and technical debt also get shifted into the cloud with the application. Still, for some organizations, this type of transition can be a starting point for bigger and better things in the future.

Typically, once organizations start using the cloud, develop some expertise with it and explore its broader set of capabilities, they make broader use of it in the future. Instead of lifting and shifting more applications, they re-platform applications, repurchase applications, or refactor applications using cloud-native capabilities. Over time, they stop managing the cloud like they managed on-premises IT and real innovation begins to flourish. However for some organizations, this transition and evolution can take time.

To speed things up, some organizations decide to make big, bold moves. Instead of lifting and shifting legacy applications to the cloud, they decide to migrate a mission-critical application to the cloud. Their logic is that since the application is critical to the business, it will get done right the first time and the things they learn in the process can be applied to all the other less critical applications that follow it to the cloud; this approach will accelerate their digital transformations and help them to potentially leap frog their waffling competitors.

Some CISOs grapple with the change that the cloud represents and seek to maintain the status quo. This is because they have successfully managed their cybersecurity program in their organizations' current IT environment. Change can represent risk for some organizations. The one place I've seen this illustrated most often is with the security assessments that enterprise security teams use to determine if new solutions meet their security standards and requirements. Such assessments seek to determine if a minimum set of controls are in place to protect the organization's data while it's being processed, stored and transmitted by new solutions. For example, one assessment question could determine whether the new solution protects data in-transit with the newest version of the **Transport Layer Security** (**TLS**) protocol. Another assessment question could determine if data at rest in the solution is encrypted using a specific algorithm. Another assessment question could be whether the vendor has a specific third-party security attestation or certification, like ISO 27001, for example.

In some organizations, when new cloud-based solutions come to the security team for a security assessment, they apply the same assessment process they have been using to assess new solutions in their on-premises IT environment. This seems reasonable; after all, the assessment checks whether solutions meet the organization's security standards.

Some of the security assessment questionnaires that I've seen over the years have been elaborate and include hundreds of questions. Many of these questionnaires were developed over a period of many years and have been customized to reflect the specific IT environments and compliance requirements of the organizations that employ them.

However, many of the questions in such assessment questionnaires are based on some key underlying assumptions; for example, an assumption that the assessors will have physical access to the hardware in order to answer their questions. Another similar example is that the assessors will be assessing systems that the organization manages themselves. Another popular assumption I've seen is that the technology used by a solution will never deviate from current commercially available technologies. For example, the hypervisor that a solution's virtualized workloads run on, runs exactly the same way as the hypervisors they have been running in their on-premises IT environments. One last example is the assumption that the vendor providing the solution only has one solution and not a huge suite or stack of technologies that can be combined in different ways to solve problems. When any of these assumptions and others are not true, the assessments that are based on them cannot be fully completed. When this happens, some security teams simply reject a solution because they couldn't determine if it met their standards using their tried and true security assessment questionnaire. However, the glaring flaw in their assessment process is that it didn't check if the solution met the organization's security standards, it checked whether the questions in their questionnaire could be answered as written; this is a subtle but important difference.

Let me use an exaggerated analogy to illustrate what I mean. For the past few decades, car owners have been able to take their cars to professionally managed garages to have multi-point inspections completed. In some cases, these inspections are mandated by law, like emissions inspections, for example. However, what happened to the owner of the first fully electric car when they took their car for the legally mandated emissions inspection? Was the garage able to process the assessment that is required by law? Did the car have an exhaust pipe or catalytic converter for the garage to test? After all, every car *must* have these technologies, right?

Given that the garage couldn't test this car the same way they had been testing cars for decades, should they fail to certify the car, even though it exceeds car emissions standards in a way that legacy internal combustion engines could *never* achieve? Some security teams reject cloud-based solutions because they cannot assess them the same way they've always assessed solutions.

Few security teams spontaneously question the assumptions that their years' old assessment processes are based on. Their security requirements don't necessarily have to change. However, they need to evolve and modernize their assessment processes to determine if new technologies can meet or exceed those requirements. The goal of security assessments is to ensure new solutions meet organizations' security requirements, not to ensure their security assessment questions never have to change. Enterprises need to question their assumptions occasionally to check if they are still accurate and relevant.

Let's jump right into it! Next, I'll share why I think the cloud is a game changer for security and compliance professionals.

Security and compliance game changers

There are numerous ways that the cloud can tilt the playing field in favor of defenders. In this section, I'll cover two security and compliance game changers.

The power of APIs

Application Program Interfaces (**APIs**) provide a powerful mechanism for systems to interact with humans and other systems. There are different kinds of APIs, but generally, APIs define the specific inputs a system is willing to accept and the outputs it will provide. The details of how the system processes inputs and provides outputs can be abstracted from view, thus simplifying the system for humans and other systems that want to use it. In other words, I don't need to know how the system works internally in order to use it. I just need to know about its APIs. I can call an API and pass it the information it requires and then wait for the output, while the magic of software happens.

Magic here is a euphemism for all the smart engineers' and developers' work on the hardware, firmware, operating systems and software that make up the stack of technologies that the API and its system rely on.

APIs can be programming language-specific and thus included as part of **Software Development Kits** (**SDKs**). This makes it easy for developers that know C++, Java or other popular programming languages to leverage APIs. Although APIs were once primarily used by developers to help them develop applications, operations roles now also make use of APIs to deploy and operate IT infrastructure, thus helping to herald the DevOps era.

In the context of cloud computing, APIs can be called from within an application, from a command line, or from the web console provided by the CSP. Let me give you some examples.

Let's say we wanted to provision and launch five virtual machines in Amazon EC2, in one of the three currently available Availability Zones in the London region. We could use the RunInstances API (AWS, 2020):

```
https://ec2.amazonaws.com/?Action=RunInstances
&ImageId=i-030322d35173f3725
&InstanceType=t2.micro
&MaxCount=5
&MinCount=1
&KeyName=my-key-pair
&Placement.AvailabilityZone=eu-west-2a
&AUTHPARAMS
```

If we used the AWS Console to do the same thing, the Launch Instance wizard would collect all the configuration information for the virtual machines and make the same type of API call on our behalf. We could also use the AWS **Command-Line Interface** (**CLI**) to launch these virtual machines, specifying the same parameters and the CLI would make the same type of API call for us:

```
aws ec2 run-instances --image-id i-030322d35173f3725 --count
5 --instance-type t2.micro --key-name my-key-pair --placement
"AvailabilityZone= eu-west-2a"
```

Under the covers of the system that this AWS CLI command is run from, it will send this type of request to Amazon EC2 using the HTTPS protocol on TCP port 443 (AWS, 2020).

One important thing to keep in mind is that API calls require authentication, authorization, integrity and confidentiality mechanisms. I won't get into these details here.

Of course, like AWS, Google Cloud and Microsoft Azure have similar APIs and support a range of programming and scripting languages, as well as command-line interfaces. This is an example from the Command-Line Interface SDK from Google, first creating a virtual machine and then starting it (Google, n.d.):

```
gcloud compute instances create example-instance --image-family=rhel-8
--image-project=rhel-cloud --zone=us-central1-a

gcloud compute instances start example-instance --zone=us-central1-a
```

A similar example can be seen here, regarding to the creation of a virtual machine in Microsoft Azure using **Representational State Transfer** (**REST**) APIs (Microsoft Corporation, 2020). Once the virtual machine has been created, another API call will start it. This can also be done using the Azure CLI, Azure PowerShell and Azure Portal:

```
{
  "location": "westus",
  "properties": {
    "hardwareProfile": {
      "vmSize": "Standard_D1_v2"
    },
    "storageProfile": {
      "osDisk": {
        "name": "myVMosdisk",
        "image": {
          "uri": "http://{existing-storage-account-name}.blob.core.
windows.net/{existing-container-name}/{existing-generalized-os-
image-blob-name}.vhd"
        },
        "osType": "Windows",
        "createOption": "FromImage",
        "caching": "ReadWrite",
        "vhd": {
          "uri": "http://{existing-storage-account-name}.blob.core.
windows.net/{existing-container-name}/myDisk.vhd"
        }
```

```
        }
      },
      "osProfile": {
        "adminUsername": "{your-username}",
        "computerName": "myVM",
        "adminPassword": "{your-password}"
      },
      "networkProfile": {
        "networkInterfaces": [
          {
            "id": "/subscriptions/{subscription-id}/resourceGroups/
  myResourceGroup/providers/Microsoft.Network/networkInterfaces/
  {existing-nic-name}",
            "properties": {
              "primary": true
            }
          }
        ]
      }
    }
  }
```

As you've seen, using APIs enables the users of these services to deploy infrastructure, such as servers, firewalls, network load balancers and third-party appliances. However, it also allows us to configure that infrastructure exactly the way we want it to be configured. For example, when deploying servers, we can specify the operating systems, IP addresses, network security configurations, routing tables, and so on This is extremely powerful. With a single command, we can start one virtual machine or a hundred thousand virtual machines, all configured exactly the way we want them configured. Because we know exactly how our systems should be configured, we can compare the current configurations of the systems that are running in production to our standard configuration and determine if there are any differences. We can do this constantly in order to detect changes that could be indicators of compromise.

In on-premises IT environments, this would typically involve deploying agents or management software on the servers that will monitor configuration changes.

One challenge that many organizations have is deploying and managing multiple agents and management suites from different vendors. Each agent requires some level of management and security updates to ensure it doesn't increase the attack surface area. Typically, CISOs and CIOs look for ways to reduce the number of agents running on systems and resist the idea of deploying more of them in their environments. Meanwhile, the sources of system configuration changes can include all sorts of things – administrators, management software, users, malware, restoring from backups and so on. This can make it challenging to detect changes and determine if changes to systems are indicators of compromise.

In the cloud, since everything happens via APIs, the APIs provide the perfect choke point for visibility and control. If organizations can monitor their API calls and take action based on what's happening, they will have great visibility and control. In this environment, deploying agents and management software to hundreds or thousands of systems is optional because the APIs are baked into the cloud. If an organization has regulatory compliance requirements that dictate specific control configurations, they can monitor those controls to ensure that they are always in compliance.

In practice, API calls are logged to API logging services for this purpose. For example, AWS CloudTrail is an API logging service that logs the API calls in AWS accounts (AWS, 2020). Earlier, when we ran the command that started five virtual machines in AWS EC2, if AWS CloudTrail was enabled, it would have logged an event that captured the details of that API call. This event contains an incredible amount of detail, including which account was used, the principal that made the call, some authentication and authorization details, the time, the region, the source IP address the call came from, details on the virtual machine and some details regarding its configuration. These logs can be combined with other logging data, aggregated and analyzed by humans and data analytics systems, imported into SIEMS in the cloud and/or downloaded to systems in on-premises IT environments. These logs are also essential for incident response investigations. Google offers Cloud Audit Logs (Google, 2020), while Microsoft provides Azure Monitor (Microsoft Corporation, October 7, 2019), in addition to other logging mechanisms, for similar purposes.

Here is a truncated example of an event logged by AWS CloudTrail:


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "Example:user123",
        "arn": "arn:aws:sts::Example:assumed-role/Admin/user123",
        "accountId": "Example-ID",
        "accessKeyId": "Example-access-key",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "Example-principle",
                "arn": "arn:aws:iam::Example:role/Admin",
                "accountId": "Example-ID",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2020-04-01T05:09:15Z"
            }
        }
    },
    "eventTime": "2020-04-01T05:09:26Z",
    "eventSource": "ec2.amazonaws.com",
    "eventName": "RunInstances",
    "awsRegion": "eu-west-2",
    "sourceIPAddress": "169.254.35.31",
    "userAgent": "aws-cli/1.3.23 Python/2.7.6 Linux/2.6.18-164.el5
",
    "requestParameters": {
        "instancesSet": {
            "items": [
                {
                    "imageId": " i-030322d35173f3725",
                    "minCount": 1,
                    "maxCount": 5,
                    "keyName": "my-key-pair"
```

```
                }
            ]
        },
        "instanceType": "t2.micro"
```

To recap, every interaction with the cloud happens via an API call. This model has numerous benefits, including security and compliance benefits. Because of this, the visibility and control that the cloud offers are superior to most on-premises IT environments, not to mention the simplicity and cost benefits of this approach. Plus, it enables new approaches to IT and security operations. For example, we know that every system that we deploy is configured to meet our security and compliance standards, because that's how it has been defined in the code that we use to deploy them. Since storage and networking are decoupled from compute services, nothing prevents us from simply shutting down systems and deploying new systems to replace them every few hours. It just takes a few lines of code in a script or application to do this, as we saw earlier. If systems are short-lived, it makes it harder for administrators and management software to introduce security misconfigurations over time that attackers can use to get a foothold in the environment.

APIs are powerful, but they too must be properly implemented so that they do not create a porous attack surface. Of course, the CSPs know this and employ expertise, processes and technology in the development of their APIs to minimize risk. Layer in authentication and authorization mechanisms, protection, monitoring, detection, response and audit capabilities; APIs rock!

I've discussed one scenario here, which is using APIs to configure and start virtual machines. Now, imagine if you could use APIs to control hundreds of cloud services that perform all sorts of functions, such as compute, storage, networking, databases, containers, serverless computing, artificial intelligence, machine learning, IoT and security, to name just a few. Imagine having programmatic control over all of that, at virtually any scale, anywhere in the world – truly amazing. This is the power of APIs! They really are a game changer for security and compliance professionals. The power of APIs is not only available for large organizations with large IT budgets; anyone with a credit card can open an account with a CSP and get the power of these APIs. Next, let's look at another game changer, automation.

The advantages of automation

As we've seen, the power of APIs enables us to configure and control most things in the cloud using code, even infrastructure. To take full advantage of the power of APIs, the cloud offers high levels of automation. In addition to running CLI commands, you can automate complex workflows using scripts, templates, applications and cloud services.

CSPs offer rich automation capabilities. These capabilities are spread across different cloud services, just like the APIs they leverage. Some examples of services that help automate some functions include Microsoft Azure Automation (Microsoft Corporation, October 18, 2018), Google Cloud Composer (Google, 2020) and AWS CloudFormation (AWS, 2020). There are also automation solutions available from third parties, such as Chef (Chef, 2020), Puppet (Puppet, 2020), Ansible (Ansible, 2020), Terraform (Hashicorp, 2020) and many others.

For security and compliance professionals, all these automation capabilities and tools can help provision, configure, manage, monitor, re-configure and deprovision infrastructure and other cloud services. In addition, these rich automation capabilities can help to protect, detect, respond and recover, while maintaining compliance to regulated standards, industry standards and internal security standards. In many cases, all of this can happen in near real time because automation, not humans, are performing these operations.

In fact, reducing human participation in these operations has many advantages. Recall the cybersecurity usual suspects that I discussed at length in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*; let's look at some examples of how automation can help us mitigate some of these. Let's start by looking at insider threat and social engineering.

Mitigating insider threat and social engineering

Remember the two types of insider threats that I defined earlier: malicious insiders who abuse their privileged access to resources and non-malicious insiders who make mistakes that lead to poor security outcomes. Automation can help mitigate both types of threats. For example, the more automation we develop, test and implement, the fewer chances administrators will have to make mistakes that have security consequences.

Using automation to complete repeatable processes can lead to more consistent and quicker outcomes that are less prone to human error.

Automating administrative processes will also result in fewer opportunities for malicious insiders to act. This is where the concepts of *just-in-time administration* and *just enough administration* can be helpful. With high levels of automation in place, administrators will require less access to systems, thus reducing the opportunities they have to steal data or damage infrastructure. Highly automated environments also make it easier to detect when administrators access systems because such occasions will be exceptions to the rules. When malicious insiders know there is increased visibility and scrutiny on them when they directly access data and systems, the frequency that they will attempt to access resources without legitimate reasons is reduced.

Automation can help minimize the amount of access administrators have. For example, instead of allowing administrators full access to systems they connect to, only allowing them to run pre-tested and approved scripts and automation on those systems will reduce the opportunities they have to run arbitrary commands. With enough automation, the only time administrators have legitimate cause to run arbitrary commands is in "break-glass" scenarios where existing automation cannot fix a problem. These cases can be monitored and audited to reduce the chances that a malicious insider will act. During such scenarios, employing quorum-based administration procedures with two or more participants can also help mitigate insider threat. Adding more automation over time to cover more support scenarios can dramatically reduce the opportunities that administrators have to run arbitrary commands.

There are also privacy benefits to using automation. If humans don't have access to sensitive data, then they can't be exposed to **Personally Identifiable Information** (**PII**) or **Protected Health Information** (**PHI**), or sensitive financial information. Using automation to interact with data, instead of humans, helps organizations fulfill the privacy promises they make to their customers or citizens.

Sounds great right? Maybe too good to be true? Can't we already do this in on-premises IT environments by using bastion hosts and Secure Shell (SSH) sessions? Great questions. Let's look at a real-world example.

The requirements the security team in this example have are that the administrators cannot directly access the systems they are managing. This means using SSH to access systems directly isn't going to meet requirements. If they did use SSH to access these systems, then they might be able to run arbitrary commands on these systems, which is something they want to avoid.

The security team in this scenario also wants to limit the use of bastion hosts in their environment. They have been burned using bastion hosts in the past. Bastion hosts typically span a higher security zone and a lower security zone, allowing administrators to get access to systems in the higher security zone from the lower security zone; subsequently, bastion hosts need to be managed as if they are part of the higher security zone. It turns out that this can be harder than it sounds and lapses in this fictional organization's processes led to a system compromise in their environment. Having been burned once, they want to minimize the number of bastion hosts in their environment.

One way to meet these requirements using AWS, for example, is to use the AWS Systems Manager service to run commands on virtual machines running in the Amazon EC2 service. To do this, the Systems Manager Agent will be installed on those virtual machines. Once that agent is properly configured, administrators can run tested and approved scripts from the AWS Systems Manager console that will execute on those virtual machines via the Systems Manager Agent (AWS, 2020).

There are a few cool advantages to this approach. First, administrators do not need to have administrator credentials for the virtual machines they are managing. Since they are running scripts from the AWS Systems Manager service in the cloud, they don't need local credentials to access individual systems. If administrators don't know the usernames and passwords for those systems, they can't log directly into them. They are limited to running the tested and approved scripts from the cloud. This helps to mitigate the risk of insider threat for those systems.

This approach also mitigates the some of the risk associated with social engineering on these systems. Administrators can't be tricked into giving up credentials for those systems because they don't know them. Since the only way the administrators interact with these systems is by remotely running pre-approved scripts on them, they can't be tricked into running arbitrary commands or installing new software, which can undermine the security of these systems and lead to bad security outcomes. Of course, given how insidious social engineering is, this approach must be married with some other mitigations to fully mitigate it; for example, **Multi-Factor Authentication** (**MFA**) for the AWS accounts themselves. However, I hope you can see the potential advantages of this approach when it comes to mitigating typical social engineering attacks against administrators. When administrators only have access when they need it and that access is tightly scoped and controlled, there's less opportunity for typical social engineering tactics to be successful.

Remember that one of the big advantages of using the cloud is scalability. If we install the Systems Manager Agent on every virtual machine that we deploy, using automation, of course, we will have the ability to use this administration method on as many systems as required – the scale is virtually unlimited. Using automation, we can manage three systems or three thousand systems using same the technique and amount of effort. As the number of systems that we manage increases or decreases, there is no additional work required by administrators as they run the same scripts regardless of the number of systems they manage; managing more systems doesn't mean administrators have more access.

If we are logging the API calls that are generated by the administrators' interactions with the AWS Systems Manager service in AWS CloudTrail, then their activities can be monitored and audited in near real time (AWS, 2020). We can also monitor and audit any interaction administrators have with the virtual machines themselves to ensure administrators only access these systems in break-glass events.

Of course, other CSPs have rich automation capabilities as well. For example, Microsoft offers a range of services and capabilities to help, including Azure Automation, Azure PowerShell, Azure Monitor and others. Google offers several services as well, including Cloud Monitoring, Cloud Functions and Cloud Asset Inventory, among others.

Automation allows us to design systems that don't require direct human interaction very often. This makes it easier to detect when those incidents happen and better mitigate insider threat and social engineering. Next, let's look at how another one of the cybersecurity usual suspects, unpatched vulnerabilities, can be mitigated in this scenario.

Mitigating unpatched vulnerabilities

Let's look at how we can use automation to help manage vulnerabilities on the virtual machines we use. As we saw in *Chapter 2*, *Using Vulnerability Trends to Reduce Risk and Costs*, Vulnerability Management teams have been faced with as many as 45 new vulnerability disclosures per day across the industry that potentially impact their systems. Automation in the cloud can help reduce the amount of work related to inventorying systems, scanning systems, and patching systems.

For example, recall that I wrote in *Chapter 2*, that accurate inventories are critical to vulnerability management teams. In the cloud, because nothing gets provisioned or deprovisioned without using APIs, APIs and automation help provide accurate inventories quickly. Inventorying environments like this doesn't take hours or days – it can be nearly instantaneous.

There are many methods available to scan and patch virtual machines in the cloud. In our AWS example, AWS Systems Manager can be used to patch systems. Also, chances are, the vendors your organization uses for vulnerability management software in your on-premises IT environment also have similar capabilities built for the cloud. This allows your organization to take the expertise it has developed from managing vulnerabilities in its on-premises IT environment and continue to leverage it in the cloud.

You might be wondering how vulnerability management processes are potentially impacted for virtual machines running in the cloud when the number of systems can be scaled up and down completely dynamically to meet load and application availability targets. In this scenario, Amazon EC2 Auto Scaling, for example, can be used to accomplish this (AWS, 2016). It can also help keep systems up to date. Instead of scanning and patching every system in a big fleet of systems, Auto Scaling can be used to dramatically reduce this effort. To do this, scan the Amazon Machine Image used to build your virtual machines for vulnerabilities and install security updates as needed to ensure the image is up to date, testing to ensure it works as expected. Then, shut down a virtual machine running in production that is based on the older version of that image. Based on the load and availability rules you set for Auto Scaling, when Auto Scaling decides it's time to launch a new virtual machine, it does so using the image that you just patched and tested. When the new virtual machine starts, it is fully patched. You can use automation to thoughtfully shut down the virtual machines running, that are based on the old image and Auto Scaling will restart new, fully patched virtual machines to replace them. No scanning, no patching and pain from reboots is mitigated. This is a much easier way to do something that has long been a pain point for large enterprises.

Google and Microsoft also provide tools to make finding and mitigating vulnerabilities efficient. For example, Google offers OS inventory management, OS patch management (currently in Beta), and Cloud Security Scanner, while Microsoft offers Azure Automation and Azure Security Center, among other tools. There are numerous third-party vendors that provide vulnerability management solutions for cloud environments, including Qualys, Tenable, IBM QRadar and many others.

Of course, this is just one method to perform patching – there are others. There is also the potential to eliminate patching altogether by using services that the CSPs manage for you. As I mentioned earlier in this chapter, IaaS is but one type of service in the cloud; there are hundreds of services from CSPs that do not require you to provision, manage and patch servers at all. If you don't need to manage servers yourself, why bother?

Let the CSPs manage infrastructure for you and you can spend the time normally relegated to such tasks to reducing technical debt in other areas, project work that never seems to get done, or innovating – imagine that. Imagine spending time figuring out how to use serverless computing, AI, ML and IoT to better protect, detect and respond to threats, instead of testing patches and rebooting servers.

The cloud can definitely help mitigate unpatched vulnerabilities and make this much easier than it is in most on-premises environments; something that has plagued enterprises for decades. Now, let's see how automation in the cloud can help mitigate another of the cybersecurity usual suspects, security misconfigurations.

Mitigating security misconfigurations

As I wrote in *Chapter 1, Ingredients for a Successful Cybersecurity Strategy*, security misconfigurations can be poor default settings in hardware, operating systems and applications, or can occur over time as systems "skew" out of their organization's standards based on the tweaks administrators or software updates introduce. Additionally, in big IT environments, abandoned technology can quickly become a forgotten risk that isn't actively managed. Because of the constant struggle large enterprises have had with keeping things configured the way they need them, Change Management emerged as a full-blown IT discipline, supported by an entire industry of vendors. This is important, not just for security purposes, but also for compliance purposes. Ensuring systems comply with regulated standards, industry standards and internal IT standards is important and in many cases, required.

In our example scenario, organizations can choose to install management software on the servers that they deploy in the cloud. They can continue to measure and remediate configuration changes in much the same way they have been in their on-premises IT environment.

They can also harness the power of APIs and the automation built into the cloud. For example, AWS Config is a cloud service that monitors resources for configuration changes and enables you to take a range of actions based on those changes.

In our example scenario, the security team might decide that one type of change should be automatically remediated; when the change is detected, automation will change the configuration back to its standard setting. Alternatively, just to be safe, automation can be used to shut down the misconfigured system and if enabled Auto Scaling will start a new system that meets all of the organization's standards to replace it.

The Security team might deem another type of change to be an indicator of compromise that needs to be investigated by their Incident Response team. In this case, automation can take a snapshot of the virtual machine, create a new **Virtual Private Cloud** (**VPC**) – let's call it *IR Clean Room* – copy the snapshot into the isolated IR Clean Room, connect the IR team's forensics software to the image, send a message to the IR team to investigate it and shut down the original virtual machine. If configured, Auto Scaling will start a new, pristine virtual machine that meets all approved standards to take its place. It does this all in near real time. Notice that in these examples, there was no management software or agent on the virtual machine and no SOC analysts performing manual queries looking for indicators of compromise. Since infrastructure is code, we can automate any number of actions to suit the organization's needs.

In a compliance context, this functionality is powerful as it can help keep things configured in a way that complies with standards. When we use automation to detect changes and take appropriate actions, we can also use that automation to generate compliance artifacts that will help the organization prove continuous compliance with the specific standards that apply to them. This helps reduce manual audits and manual remediation of misconfigured systems.

Microsoft Azure Automation and Google Cloud Asset Inventory provide similar capabilities for their respective services. There are also third parties that provide automation solutions such as Ansible, Chef, Terraform and several others.

Next, let's look at how automation in the cloud helps mitigate the last of the cybersecurity usual suspects: weak, leaked and stolen passwords.

Mitigating weak, leaked and stolen passwords

CSPs and numerous third-party vendors offer identity and access management solutions for the cloud and hybrid environments. For example, Microsoft offers Azure Active Directory and supporting services such as just-in-time privileged access capabilities via Azure Active Directory **Privileged Identity Management** (**PIM**) (Microsoft Corporation, 2020). Third parties such as Aporeto, Centrify, CyberArk and many others also provide services that can help several in different scenarios. Google Cloud offers Cloud Identity and Access Management, while AWS offers AWS Identity and Access Management.

CSPs offer MFA, which, as I discussed in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*, is a highly effective control that mitigates weak, leaked and stolen passwords to a great extent. Leveraging MFA and limiting the amount of time users have access to resources between authentication requests can make it much harder for attackers to use stolen and leaked credentials successfully. Using a secrets manager to manage access keys, certificates and credentials in order to automatically change and rotate them periodically can also be effective. To do this, Google offers Google Cloud Secret Manager (Google, 2020), Microsoft offers Azure Key Vault (Microsoft Corporation, 2020) and AWS provides AWS Secrets Manager (AWS, 2020). Again, there are many third-party vendors that also offer solutions, including Docker Secrets, SecretHub, Confidant and others.

In fact, there are so many capabilities and so much functionality in identity and access services and solutions, entire books have been dedicated to this topic area. Identity is the key to security. I highly recommend spending some time learning about the powerful identity and access capabilities that CSPs and other vendors have to offer.

Security and compliance game changers – summary

The power of APIs and automation in the cloud are two game changers for security and compliance professionals. That's not to say that APIs and automation are not available in on-premises IT environments. However, the investment and effort to bring these capabilities on par with those baked into the cloud would be prohibitively expensive and difficult to implement; considering anyone with a credit card and a few minutes to open an account with a CSP gets these capabilities by default, it would be difficult to justify implementing on-premises versions.

We've now seen that the cloud can offer some effective and innovative ways to address all the cybersecurity usual suspects. Put another way, the cloud makes addressing the cybersecurity fundamentals easier than mitigating them in on-premises IT environments. We've only really scratched the surface here because the example scenario I used throughout this section was an IaaS example. As I mentioned, CSPs offer hundreds of services that span and blend IaaS, PaaS, SaaS, FaaS, IDaaS and others. Not to mention, I didn't dive into any of the security services these CSPs offer. Entire books have been dedicated to the topic of cloud security.

Let's now look at how the cloud can support the cybersecurity strategies that we examined in *Chapter 5*, *Cybersecurity Strategies*.

Using cybersecurity strategies in the cloud

In *Chapter 5*, *Cybersecurity Strategies*, we examined several cybersecurity strategies that I have seen employed in the industry over the past two decades. We evaluated these strategies using the **Cybersecurity Fundamentals Scoring System** (**CFSS**). The CFSS score estimate for each strategy helps us understand how well they address the cybersecurity fundamentals. To refresh your memory, a summary of the CFSS scores for each strategy is provided in *Table 8.1*:

Cybersecurity Strategy	Unpatched vulnerabilities	Security misconfigurations	Weak, leaked, stolen credentials	Social engineering	Insider threat	Total Score
Protect and Recover Strategy	10	10	0	5	0	25
Endpoint Protection Strategy	20	20	15	10	10	75
Physical Control and Security Clearances Strategy	10	10	15	10	10	55
Compliance as a Cybersecurity Strategy	10	10	10	10	10	50
Application-Centric Strategy	20	20	10	10	10	70
Identity-Centric Strategy	5	5	15	10	10	45
Data-Centric Strategy	5	5	0	15	15	40
Attack-Centric Strategy	20	20	20	15	20	95

Table 8.1: CFSS score estimate summary

Almost any of these strategies can be used in the cloud. Let's now look at a few of these strategies in the context of the cloud.

Using the protect and recover strategy in the cloud

CSPs offer granular firewall and network controls that can help organizations adopt and operate the Protect and Recover Strategy. The power of APIs and automation in the cloud enable Network teams and Security teams to provision and operate Web Application Firewalls, as well as network firewalls at the edge of their cloud estates and build and operate DMZs. They also provide Virtual Private Clouds or Virtual Networks that add another layer of control over network traffic, in addition to network ACLs, routing tables, subnet rules, host-based firewalls, and so on. CSPs typically offer a dizzying array of network controls.

Since all these controls can be provisioned and monitored via code and automation, it's much easier to execute this strategy in the cloud versus on-premises. In the cloud, there is no hardware to order and receive, no racking and stacking in the datacenter and nothing requiring more rack space, power, or cooling. You just run code and the CSPs do everything else. If you need to scale your infrastructure up or down, it's just more code and automation. You only pay for what you use and can shut it down any time your organization decides to. The Protect and Recover Strategy is a poor scoring strategy, as we discussed in *Chapter 5*, *Cybersecurity Strategies*. It can be used in combination with other strategies to more fully address the cybersecurity fundamentals. It's easier to extend this strategy in the cloud too, because everything is code. Let's look at a better scoring strategy now.

Compliance as a cybersecurity strategy in the cloud

Let's look at another strategy from *Chapter 5*, *Cybersecurity Strategies*, Compliance as a Cybersecurity Strategy. Earlier in this chapter, we looked at how APIs and automation in the cloud help mitigate security misconfigurations. Those same capabilities can help organizations continuously comply with security standards, whether they are regulated, industry, or internal standards. I've already discussed how APIs and automation can ensure that systems are properly configured and continuously monitored for configuration changes. However, there's one important nuance to executing this strategy to be aware of.

Many security teams and compliance teams that contemplate using the cloud for the first time wonder how they can prove that they are complying to standards, that is, when they don't own the datacenters their infrastructures are running in and subsequently can't get their auditors access to these facilities. Regardless of who owns the datacenters, many organizations still must prove to their auditors and regulators that they are complying with required standards.

In most cases, this is another advantage of leveraging hyperscale CSPs. AWS, Google and Microsoft all have numerous certifications and attestations across their cloud services. For example, ISO27001 is table stakes for any CSP today – they all must have this certification to satisfy requirements for their enterprise customers. There are two certifications that are most valuable to many CISOs.

The first is the American Institute of CPAs' **System and Organization Controls** (**SOC**), in particular the SOC2 Type II certification (AICPA, 2020). There are at least a couple of things that make this certification valuable to CISOs, Security teams and Compliance teams. First, the scope of controls that are audited in a SOC2 Type II typically answer most of the questions that enterprises have about security. Second, this isn't a "point in time" snapshot of control settings or architectural design; it takes organizations that pursue the SOC2 Type II 6 months of continuous audit to achieve it. The steps that organizations take to get ready for this type of audit can dramatically improve their security posture. Then, to achieve this certification and maintain it over time and continuously prove that services are being operated the way they are described, can be a big challenge. Many enterprises would never even attempt to get this certification because it's hard to do and can be expensive. However, the hyperscale CSPs achieve and maintain this certification across many of their services in order to keep their security standards among the highest in the industry.

CSPs will typically share their SOC2 Type II audit reports with their customers. For Security teams and Compliance teams, it is worth downloading these reports and reviewing them to ensure the solution(s) they are evaluating meet or exceed their standards. Questions not answered by the SOC2 Type II audit report can be directed to the CSPs themselves, who are typically happy to answer them.

Another attestation that many CISOs and security teams find valuable is the **Cloud Computing Compliance Controls Catalog** (**C5**), designed by the **Federal Office for Information Security** (**BSI**), a federal government office in Germany (The BSI, 2020). The C5 is an in-depth security assurance attestation. It has criteria for many domains including policies, personnel, physical security, identity and access management, encryption and others. Again, the scope and complexity of this attestation can make it a challenge to achieve and maintain. Like the SOC2 Type II, for CISOs, this attestation contains answers to many of the questions they have about CSPs' security control sets.

The SOC2 Type II and the C5 are like treasure troves of security information for CISOs, Security teams, Compliance teams and auditors. CSPs typically combine these with numerous other certifications and attestations to help their customers prove they are meeting their compliance requirements. However, customers of CSPs have a role to play in this as well. Remember that CSPs are different from **Managed Service Providers** (**MSPs**). CSPs offer self-service clouds. Their customers and ISVs can build on top of those clouds to create solutions. However, the CSPs' certifications' and attestations' scopes do not cover the portion of solutions that are architected and operated by their customers; unlike MSPs, CSPs typically don't have the visibility, or the access, required to do this.

This arrangement means that CSPs and their customers both bear responsibility for their respective portions of the solutions they architect and operate. Google, Microsoft and AWS all refer to this arrangement as a *shared responsibility*. Both CSPs and their customers provide the appropriate certifications and attestations to prove that their respective portions of their solutions meet the requirements of the standards they are bound to. This arrangement typically saves CSPs' customers time and money. This is because the portion of their solutions that they must attest to can be dramatically reduced in almost all cases. For example, since CSPs' customers don't own the datacenters that their infrastructures are running in, they have essentially delegated the responsibility to audit and certify those datacenters to their CSPs. Put another way, they no longer have to deal with the complexity and cost of physical datacenters, as the CSPs do this for them. It's a win for CSPs' customers because they can meet or exceed the security standards they are responsible for while reducing the amount of effort and cost to them.

Information on the compliance programs that CSPs operate on can be found on their respective websites, but the auditors' reports themselves are typically reserved for CSPs' customers; here are the locations that contain compliance program information for AWS, Google, and Microsoft:

- **AWS**: `https://aws.amazon.com/compliance/programs/`
- **Goggle**: `https://cloud.google.com/security/compliance/`
- **Microsoft**: `https://docs.microsoft.com/en-us/microsoft-365/compliance/offering-home?view=o365-worldwide`

The combination of APIs, automation and the certifications and attestations provided by CSPs can help organizations that want to pursue Compliance as a Cybersecurity Strategy. For organizations that want to extend this strategy to fully address the cybersecurity fundamentals, the cloud typically makes this easier than in on-premises IT environments. This is because of the APIs and automation capabilities we have discussed. Everything is code. Let's look at one more strategy that we examined in *Chapter 5*, *Cybersecurity Strategies* and how it can be implemented in the cloud.

Using the Attack-Centric Strategy in the cloud

The best scoring of all the strategies that we examined was the Attack-Centric Strategy. In *Chapter 6*, *Strategy Implementation*, we did a deep dive into this strategy and illustrated one way it could be implemented. In *Chapter 7*, *Measuring Performance and Effectiveness*, we examined one way the efficacy of this strategy can be measured. However, can this strategy be implemented in the cloud?

The short answer to this question is, yes, it can be implemented in the cloud. In some cases, some of the work has already been done for organizations that want to pursue this strategy in cloud environments. For example, MITRE provides "tactics and technique [sic] representing the MITRE ATT&CK® Matrix for Enterprise covering cloud-based techniques. The Matrix contains information for the following platforms: AWS, GCP, Azure, Azure AD, Office 365, SaaS." (MITRE, October 9, 2019).

I mentioned in *Chapter 6*, *Strategy Implementation*, the MITRE ATT&CK® framework can complement the Intrusion Kill Chain model (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.) which we examined in depth. The Kill Chain approach can also be implemented in the cloud. To do this, you'll likely want to scope your efforts to developing a Courses of Action Matrix (Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D.) like we did in *Chapter 6*, for the solution you are implementing in the cloud. Put another way, since this can be a time-intensive exercise, as you saw, you don't need to build a Courses of Action Matrix for every cloud service that a CSP offers, only the ones you plan to use.

Doing this for solutions developed for IaaS environments is, in some respects, similar to performing this mapping for on-premises IT environments. This is because much of the hardware and software can be the same or similar. For example, the operating system mitigations identified for a solution running on Linux or Windows will be very similar, regardless of whether that operating system is running in the cloud or on-premises. However, as we discussed earlier, cloud-native controls and third-party solutions can be also be layered into the environment, in addition to these operating system mitigations, to implement a set of controls that will make it much harder for attackers to be successful. For example, the same services that help us detect configuration changes will help us detect indicators of compromise in the cloud, in near real time. The same identity and access capabilities we discussed will make it much harder for attackers to use stolen credentials to move laterally. The techniques we talked about to help keep systems up to date will make it harder for attackers to find and exploit unpatched vulnerabilities.

Note that although the Kill Chain approach lends itself well to solutions that are built in IaaS environments, this approach is less helpful for solutions that are built using cloud services higher up the stack, like managed services. In these cases, CSPs are responsible for securing the underlying IT environment, typically leaving less direct access and less direct control of the underlying IT infrastructure to organizations' security teams. This doesn't mean security teams don't have the visibility and control they require – it's just the opposite, as we've discussed. However, the types of mitigating controls will likely be different than traditional solutions developed for on-premises or IaaS environments.

The controls should be different because some of the threats and risks are certainly different. Subsequently, the Kill Chain might not be the best scoring approach for organizations in the cloud, depending on the types of services they use. As enterprises consume more and more services that blur the boundaries between IaaS, PaaS, SaaS, FaaS, IDaaS and other models, the less relevant the Kill Chain approach becomes.

This isn't a bad thing – it's just more change to embrace. Remember, the role of CISOs and security teams isn't to ensure the status quo, it's to protect their organizations' data, even when these organizations decide it's time to evolve the technologies and processes they use, in order to stay competitive and/or relevant. The cloud offers the opportunity to modernize not only technologies and processes, but also the cybersecurity strategy that can be employed. Let's explore this concept a little further and look at a more modern approach cybersecurity that I mentioned in *Chapter 5*, *Cybersecurity Strategies*, called *DevOps*.

DevOps – A modern approach to security in the cloud

For the lack of a better name, let's simply call this approach *DevOps*. This strategy represents a more modern approach compared to the other cybersecurity strategies that we've examined. It recognizes that development and IT operations disciplines have been joining forces, partly because, together, these roles are aptly positioned to take advantage of the power of APIs and automation. Because everything is code in the cloud, including infrastructure, teams that understand both development and IT infrastructure operations can take full advantage of all the cloud has to offer. Let's look at some of the ways that a DevOps-driven security strategy can help security teams protect, detect and respond to modern threats in cloud-based environments.

Remember back to *Chapter 3*, *The Evolution of the Threat Landscape – Malware*, where I described why the Windows ecosystem has so much more malware than the Apple iOS ecosystem. The key, it would seem, is how software has traditionally been distributed in these ecosystems. Microsoft allowed software developed by anyone to be feely installed by its customers on their Windows-based systems.

Apple, on the other hand, provides a single source for all applications destined for iOS-based devices, their App Store. While Windows customers were left to make their own decisions about the trustworthiness of the software they wanted to run, Apple imposed a security standard for all ISVs to meet before their apps could be distributed to iOS-based devices. This difference in software distribution methods, at least partially, explains why the Apple iOS ecosystem has maintained such a low prevalence of malware.

Let's take this lesson and apply it to our approach to security in the cloud. Leveraging continuous testing, **Continuous Integration** (**CI**), and **Continuous Delivery** or **Continuous Deployment** (**CD**) can help minimize how much questionable software makes it into the cloud-based environments that CSPs' customers build and operate. In their CI/CD pipelines, they can impose automated (and manual) security and compliance checks. These ensure that any software or infrastructure that gets deployed into production environments through these pipelines meets their organizations' security and compliance requirements.

To do this, each step of the CI/CD pipeline will have the appropriate security and compliance checks automated in them. For example, a DevOps team could develop or procure automation that looks for issues contained in the OWASP Top 10 (OWASP, 2020). Another common example is the requirement to perform static code analysis and/or a specific set of functional security tests. Infrastructure will have to meet the control setting requirements defined by each organizations' compliance team and this will be verified as items go through the pipeline.

Implementing such tests is typically done in code and automation, so the number and types of checks that can be conducted are almost unlimited. Of course, because this can be effective and fun, once some DevOps teams start developing these checks, they'll spend more time on the development of their CI/CD pipelines than they will on applications and infrastructure.

If an application or infrastructure item does not pass one of these checks, the pipeline will stop, the appropriate staff can be alerted, the item will not progress, and it will not be introduced into the production environment as planned. The deficiency in the application or infrastructure item will have to be addressed in order to pass the check that failed and then go through the entire pipeline again.

This way, only items that pass every security and compliance check in the pipeline will make it into production. This means Security and Compliance teams can have high confidence that everything being introduced into their production environment meets all their security and compliance requirements and that they will not introduce more risks into that environment. To accomplish this, everything must go through a CI/CD pipeline. Put another way, the only way to get an application or infrastructure item into production is through a CI/CD pipeline. For the best chance of success, organizations need to have the discipline, as well as the governance mechanisms, to enforce this requirement. Managing multiple CI/CD pipelines is a predictable and common outcome, especially for large, distributed organizations. The risk for some organizations is that the number of CI/CD pipelines proliferates to levels that begin to compromise the high security and compliance standards that the initial pipelines imposed; too many pipelines can turn into a governance issue.

Also, note that some attackers have clued into the fact that more and more organizations are using DevOps and CI/CD pipelines. This makes the CI/CD pipelines themselves a potential target for attackers. Understanding the stack of technologies and automations that your organization uses for their pipelines and taking steps to protect them is important. For some organizations, CI/CD pipelines can become high value assets and warrant special attention, as I discussed in *Chapter 1*, *Ingredients for a Successful Cybersecurity Strategy*.

Now that security and compliance teams have confidence in their deployments, how do they keep those environments in that pristine condition over time? They can use the services and automation we discussed earlier in this chapter to monitor for configuration changes. When configurations change, they can use automation to bring them back into compliance or impose deeper investigations into how and why they changed.

As we discussed earlier, there is a range of options for vulnerability management in the cloud. Continuing to use the technologies and processes that your organization has used for years in their on-premises environment is likely one possible option.

However, using automation, like the Auto Scaling example I provided earlier, has the potential to simplify and accelerate vulnerability management. Another option is for organizations to evolve from managing servers and applications themselves, to using cloud services higher up the stack and leave infrastructure patching to the CSPs.

One of the reasons that Attack-Centric strategies gained such popularity in the industry is that they can make it hard for "advanced" threat actors to be successful – the so-called **Advanced Persistent Threat** (**APT**). However, this is where the power of APIs and high levels of automation can also be helpful. For example, when organizations shut down subsets of servers running in the cloud every few hours and replace them with new ones that meet all requirements, it can make it harder for attackers to get and maintain a foothold in that environment. Short-lived, relatively immutable systems can leave very little oxygen for attackers to use, unlike systems that remain running for months or years.

The detection capabilities in the cloud are superior to those found in most on-premises environments. Remember the power of APIs and automation in the cloud provides visibility and control that few on-premises environments can achieve. The cloud can make it easy to log API calls, network traffic, authentication and authorization operations, encryption /decryption key operations, and so on. However, one challenge most security teams share, whether they use the cloud or not, is that the vast amount of data in all these logs make it nearly impossible for humans to review it and use it in a timely way. This is where the cloud can also help. AI and ML services can be used to review all of these logs and API activity, instead of security team members, and identify things that really warrant their attention. This is possible because AI/ML services can scale as large as needed to churn through enormous log datasets far, far faster than humans can. As they do this, these services, with the help of automation, can detect and respond to all sorts of attacks, including DDoS, malware, exploitation of vulnerabilities, insider threat and many more.

Finally, if all of these capabilities failed to protect, detect and respond to attackers, DevOps and the cloud can make recovering production environments much easier than in typical on-premises environments. Since everything is code, rebuilding environments in the cloud can be relatively easy, when some planning and thoughtful preparation is given. However, Business Continuity Planning is the topic for another book.

Again, I feel like we haven't even scratched the surface here. However, I hope you have enough information to contemplate if a DevOps strategy can help your organization. It can take time to transition from traditional strategies to DevOps and some organizations pursue a thoughtful combination of DevOps and traditional strategies during this period.

This concludes this section on cybersecurity strategies in the cloud. However, before we come to the end of this chapter and this book, I do want to highlight another important set of capabilities that the cloud provides: encryption and key management.

Encryption and key management

You might be wondering why I left this topic until the very last section of this book. In my experience, most conversations about security in the cloud end with encryption and key management. No matter what topics the conversation starts with, such as vulnerabilities, exploits, malware, or internet-based threats, they end by discussing encryption and key management. This is because encryption is a powerful data protection control that helps provide confidentiality and integrity for data.

No matter which cybersecurity strategy or combination of strategies organizations pursue, when the rubber hits the road, protecting the data is the objective. That's what can be so distracting about the cybersecurity strategies we examined that are *proxies* for data protection. Security teams get so focused on protecting endpoints or applications that they lose sight that the underlying objective is to protect data. The proxies I mentioned are important and must be effectively managed, but don't forget about the data!

The CSPs all know this and offer their customers rich sets of encryption and key management capabilities. Their goal is to protect data when it is in transit and at rest. TLS (version 1.2) is the de facto internet standard for protecting data in transit. Subsequently, CSPs support TLS, in addition to providing other mechanisms for protecting data in-transit, like VPN connections or directly connecting to their cloud infrastructures, as examples.

CSPs typically offer a range of encryption options to protect data at rest, enabling their customers to encrypt data before they put it in the cloud (in some scenarios) and/or after they put it in the cloud. The current encryption standard that CSPs offer for encrypting data at rest is the **Advanced Encryption Standard** (**AES**), typically using 128-bit or 256-bit key lengths.

If an attacker had access to data (access is typically authenticated and authorized) protected by AES256, breaking this type of encryption using brute-force techniques and lots of conventional compute power would likely take far, far more time than the value lifetime of the data.

An important nuance for Security teams to understand is exactly what is being encrypted and which risks encryption mitigates. For example, if the underlying storage media is encrypted, but the data being written to the media is not encrypted prior to being written, then the risks being mitigated are the loss or theft of the storage media. Encrypted storage media helps mitigate attacks where attackers have physical access to the storage media. If someone gets physical access to the encrypted storage media but doesn't possess the keys to mount and decrypt it, the data written on it is protected from unauthorized access. However, if attackers seek to access the data logically instead of physically, over a network, for example, then storage-level encryption will likely not mitigate this risk because the data is decrypted as it is accessed from the network.

It's important to understand the specific risk that needs to be mitigated and the specific mitigations for that particular risk, in order to have confidence that the risk has truly been mitigated. If the desire is to prevent unauthorized access to data at rest, over a network, then encrypting the data itself, instead of just the storage media, will be a more effective mitigation. This might sound obvious, but this is a common mistake Security teams make during application security assessments.

In addition to offering data encryption options, CSPs are really providing authenticated and authorized data encryption. That is, each encryption operation API call is authenticated and must be authorized; encryption and decryption operations will not occur without being authenticated and authorized first. Using Identity and Access services this way provides Security teams with a lot of flexibility. For example, one person or group of people can be authorized to encrypt data, but not authorized to decrypt it. Another group can be given permissions to decrypt data, but not to do both encryption and decryption operations. Authenticated and authorized encryption enables a separation of duties that can be helpful in many scenarios.

For many organizations, one of the most challenging parts of encryption can be key management. The stakes are high because if an organization's keys are damaged, lost, or stolen, it could have a catastrophic impact on them. Generally speaking, CSPs want to make key management easy and safe for their customers. Google offers Cloud Key Management Service (Google, 2020), Microsoft offers Azure Key Vault (Microsoft Corporation, 2020), and AWS provides the AWS Key Management Service (AWS, 2020). Of course, there are third-party vendors that also offer encryption and key management services, such as Thales, Gemalto, Equinix and others.

The CSPs' key management services can offer an interesting advantage in that they can be integrated into their other cloud services. This means that some cloud services could perform encryption and decryption on behalf of users. The data protection advantage here is that the data can be protect by AES encryption until it's in the physical memory of the servers running the service that is going to process it. Once processing completes, the service could re-encrypt the data again, before moving it into storage or other services for more processing. The keys used for encryption and decryption can be protected in-transit between the key management services and the services that use them. This means that unencrypted data only sees the light of day in very controlled environments that are authorized by the data owner. This can help maximize the number of places and the time that the data is protected with encryption. CSPs' key management services tend to be designed for low latency and high availability in order to potentially process billions of requests.

Some organizations want a separation of duties between the vendor they use for compute and storage and the vendors that provide key management services. Third-party vendors that offer key management services can play this role or CSPs' customers themselves can operate and maintain their own key management infrastructures. The organizations that choose this option should be comfortable managing their own key management infrastructure or allowing a third party do it for them. However, managing **Hardware Security Modules** (**HSMs**) and **Public Key Infrastructures** (**PKIs**) is notoriously difficult. This makes using CSPs' key management services a popular option.

For organizations that need to keep their keys on-premises but still want to get the benefits of the cloud, client-side encryption is a potential solution. Using client-side encryption means that the data owner encrypts the data before they put it into a cloud service. For example, the data owner has their own on-premises key management infrastructure. Prior to putting data into a cloud storage service, they generate a key on-premises and then use an application also running on-premises to encrypt the data using this key. Then, they authenticate and securely transfer the encrypted data to the cloud storage service. In this scenario, their CSP never had access to their unencrypted data or the encryption key as neither left their on-premises IT environment. To decrypt this data, the data owner authenticates to the cloud storage service, securely downloads the encrypted data and uses their on-premises application and on-premises key to decrypt the data. Again, neither the unencrypted data nor the encryption key was ever shared with the CSP.

Client-side encryption isn't limited to storage scenarios; it can be used with other services, like databases, for example. In this scenario, client-side encryption is used to encrypt records or individual fields as they are written to a database service running in the cloud. To do this, an encryption key is retrieved from the on-premises key management system and temporarily used for encryption operations by the application performing the encryption. Once the record is encrypted as it's written to the database, the encryption key can be removed from the memory of the application that performed the encryption operation, thus reducing the time the key is resident on a system outside of the on-premises key management system. The application performing encryption and decryption operations on the database records can run on-premises or in the cloud. Since the CSP's customer has full control of the keys, the CSP cannot get access to the keys unless the customer grants them access. Indexes and database keys are left unencrypted so that indexing and searching the database can still be performed. For this reason, it's important not to put sensitive data into these fields. To decrypt the data, the appropriate records are retrieved and decrypted after the key is provided from the on-premises key management system. After the decryption operation, the key can once again be removed from the memory of the application performing the decryption operation.

There are many different ways to perform client-side encryption and key management. However, this method can be more complicated and expensive to implement than simply using the integrated encryption and key management services that CSPs offer. Some organizations that start off using client-side encryption with keys kept on-premises, over time, conclude that using CSPs' key management services mitigates the risks they are most concerned about and simplifies their applications. After all, encryption and decryption operations in the cloud are performed using API calls that are authenticated, authorized, monitored and potentially controlled using automation, as we discussed earlier.

Combining properly implemented encryption and effective key management, along with the power of APIs and automation in the cloud, helps protect data in ways that would be more complex to duplicate in on-premises IT environments. Encryption and key management helps to protect data from many of the threats we discussed throughout this book; they are powerful data protection controls that should be part of whichever cybersecurity strategies your organization pursues.

Conclusion

For organizations that haven't adopted the cloud yet, or won't in favor of their on-premises IT environments, a quote comes to my mind:

> "*The future is already here – it's just not evenly distributed.*"
>
> *– (Gibson, 2003)*

The opportunity to leverage the power of APIs and cloud automation on a scale not imagined before is waiting for every organization. Not only do these game changers make provisioning, configuring, operating and deprovisioning applications and IT infrastructure much easier, but they provide security and compliance professionals the visibility and control they likely haven't had in the past. I encourage CISOs and Security teams to embrace the cloud as a way to do more with less and offset the industry's perpetual cybersecurity talent shortage.

Chapter summary

This chapter introduced some of the security and compliance benefits of cloud computing. I focused my discussion on the world's most popular CSPs' basic capabilities, that is, of Amazon Web Services, Google, and Microsoft.

The physical infrastructure model that hyperscale CSPs have roughly coalesced around is based on the concept of regions and availability zones. This concept is that an availability zone is a cluster of datacenters and a region is a cluster of availability zones. There are meaningful differences in the size and scope of CSPs' infrastructures and how they leverage components of this model. Although the terms IaaS, PaaS, and SaaS are still in widespread use today, they are slowly becoming obsolete. Newer services that solve specific problems can blur the lines between IaaS, PaaS, and SaaS service models, making them less important.

CSPs are different from traditional **Managed Service Providers** (**MSPs**) in some key ways. It is important that executives recognize this when they first contemplate using the cloud, in order to avoid confusion that will slow them down. MSPs that build on top of CSPs' offerings continue to play important roles for their customers and the industry.

In this chapter, I discussed two security and compliance game changers that the cloud provides:

- The power of **Application Program Interfaces** (**APIs**)
- The advantages of automation

Every interaction with the cloud via administration consoles, command-line interfaces, and applications happens using APIs. APIs provide the perfect choke point for visibility and control. If organizations can monitor their API calls and take action based on what's happening, they will have great visibility and control. To take full advantage of the power of APIs, the cloud offers high levels of automation. In addition to running CLI commands, you can automate complex workflows using scripts, templates, applications, and cloud services. Automation in the cloud can help address the cybersecurity fundamentals in ways that are potentially more efficient than in traditional IT environments.

The cloud is flexible enough to support almost any of the cybersecurity strategies that we discussed in *Chapter 5*, *Cybersecurity Strategies*. DevOps offers a more modern approach compared to the other cybersecurity strategies that we examined. Because everything is code in the cloud, including infrastructure, teams that understand both development and IT infrastructure operations can take full advantage of all the cloud has to offer. Continuous Integration (CI), Continuous Delivery and Continuous Deployment (CD) pipelines can have the appropriate security and compliance checks automated in them; for example, the OWASP Top 10 (OWASP, 2020).

I hope you found this book both educational and entertaining at times. I firmly believe that if we can be specific enough about the risks we care about, as well as honest enough with ourselves about the effectiveness of the mitigations and strategies that we employ, cybersecurity will come into sharper focus.

Bon voyage!

References

1. AICPA. (April 2020). *SOC 2® - SOC for Service Organizations: Trust Services Criteria*. Retrieved from AICPA: `https://www.aicpa.org/interestareas/frc/assuranceadvisoryservices/aicpasoc2report.html`
2. Amazon Web Services. (March 2020). *Cloud Computing with AWS*. Retrieved from AWS: `https://aws.amazon.com/what-is-aws`
3. Ansible. (April 2020). *Red Hat Ansible*. Retrieved from Red Hat Ansible: `https://www.ansible.com/`
4. AWS. (October 20, 2016). *Fleet Management Made Easy with Auto Scaling*. Retrieved from AWS Compute Blog: `https://aws.amazon.com/blogs/compute/fleet-management-made-easy-with-auto-scaling/`
5. AWS. (April 2020). *Amazon Elastic Compute Cloud API Reference*. Retrieved from AWS: `https://docs.aws.amazon.com/AWSEC2/latest/APIReference/API_RunInstances.html`

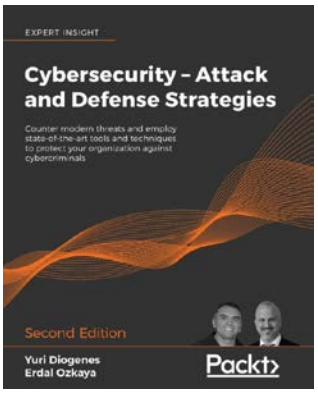
6. AWS. (April 2020). *AWS CloudFormation*. Retrieved from AWS: `https://aws.amazon.com/cloudformation/`
7. AWS. (April 2020). *AWS CloudTrail*. Retrieved from AWS: `https://aws.amazon.com/cloudtrail/`
8. AWS. (April 2020). *AWS Key Management Service (KMS)*. Retrieved from AWS: `https://aws.amazon.com/kms/`
9. AWS. (April 2020). *AWS Secrets Manager*. Retrieved from AWS: `https://aws.amazon.com/secrets-manager/`
10. AWS. (April 2020). *Logging AWS Systems Manager API calls with AWS CloudTrail*. Retrieved from AWS: `https://docs.aws.amazon.com/systems-manager/latest/userguide/monitoring-cloudtrail-logs.html`
11. AWS. (April 2020). *Remotely Run Commands on an EC2 Instance*. Retrieved from AWS: `https://aws.amazon.com/getting-started/hands-on/remotely-run-commands-ec2-instance-systems-manager/`
12. AWS. (April 2020). *Using the AWS CLI*. Retrieved from AWS: `https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-using.html`
13. Chef. (April 2020). *Chef*. Retrieved from Chef: `https://www.chef.io/products/chef-infra/`
14. Eric M. Hutchins, Michael J. Cloppert, Rohan M. Amin, Ph.D. (n.d.). *Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains*. Retrieved from Lockheed Martin: `https://lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/LM-White-Paper-Intel-Driven-Defense.pdf`
15. Gibson, W. (December 4, 2003). Books of the Year. *The Economist*.
16. Google. (April 2020). *Cloud Audit Logs*. Retrieved from Google Cloud: `https://cloud.google.com/logging/docs/audit`
17. Google. (April 2020). *Cloud Key Management Service*. Retrieved from Google Cloud: `https://cloud.google.com/kms/`

18. Google. (April 2020). *Cloud Composer*. Retrieved from Google Cloud: https://cloud.google.com/composer
19. Google. (April 2020). *Introducing Google Cloud's Secret Manager*. Retrieved from Google Cloud Blog: https://cloud.google.com/blog/products/identity-security/introducing-google-clouds-secret-manager
20. Google. (n.d.). *gcloud compute instance create*. Retrieved from Google: https://cloud.google.com/sdk/gcloud/reference/compute/instances/create
21. Hashicorp. (April 2020). *Terraform*. Retrieved from Terraform by Hashicorp: https://www.terraform.io/
22. Microsoft Corporation. (October 18, 2018). *An introduction to Azure Automation*. Retrieved from Microsoft Azure: https://docs.microsoft.com/en-us/azure/automation/automation-intro
23. Microsoft Corporation. (October 7, 2019). *Azure Monitor overview*. Retrieved from Microsoft Azure: https://docs.microsoft.com/en-us/azure/azure-monitor/overview
24. Microsoft Corporation. (March 2020). *Azure Products*. Retrieved from Microsoft Azure: https://azure.microsoft.com/en-us/services/
25. Microsoft Corporation. (April 2020). *Key Vault*. Retrieved from Microsoft Azure: https://azure.microsoft.com/en-us/services/key-vault/
26. Microsoft Corporation. (April 2020). *Manage secrets in your server apps with Azure Key Vault*. Retrieved from Microsoft Learn: https://docs.microsoft.com/en-us/learn/modules/manage-secrets-with-azure-key-vault/
27. Microsoft Corporation. (April 2020). *Virtual Machines - Start*. Retrieved from Microsoft Corporation: https://docs.microsoft.com/en-us/rest/api/compute/virtualmachines/start
28. Microsoft Corporation. (April 2020). *What is Azure AD Privileged Identity Management?* Retrieved from Microsoft Azure: https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/pim-configure

29. MITRE. (October 9, 2019). *Cloud Matrix*. Retrieved from MITRE ATT&CK®: `https://attack.mitre.org/matrices/enterprise/cloud/`
30. OWASP. (April 2020). *Top 10 Web Application Security Risks*. Retrieved from OWASP: `https://owasp.org/www-project-top-ten/`
31. Puppet. (April 2020). *Puppet Enterprise*. Retrieved from Puppet: `https://puppet.com/products/puppet-enterprise/`
32. The BSI. (April 2020). *Criteria Catalogue C5*. Retrieved from Federal Office for Information Security: `https://www.bsi.bund.de/EN/Topics/CloudComputing/Compliance_Criteria_Catalogue/Compliance_Criteria_Catalogue_node.html`

Other Books You May Enjoy

If you enjoyed this book, you may be interested in these other books by Packt:



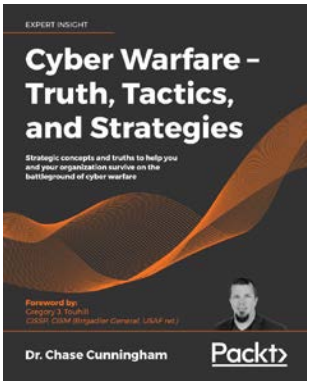
Cybersecurity – Attack and Defense Strategies – Second Edition

Yuri Diogenes, Erdal Ozkaya

ISBN: 978-1-83882-779-3

- The importance of having a solid foundation for your security posture
- Use cyber security kill chain to understand the attack strategy
- Boost your organization's cyber resilience by improving your security policies, hardening your network, implementing active sensors, and leveraging threat intelligence

- Utilize the latest defense tools, including Azure Sentinel and Zero Trust Network strategy
- Identify different types of cyberattacks, such as SQL injection, malware and social engineering threats such as phishing emails
- Perform an incident investigation using Azure Security Centre and Azure Sentinel
- Get an in-depth understanding of the disaster recovery process
- Understand how to consistently monitor security and implement a vulnerability management strategy for on-premises and hybrid cloud
- Learn how to perform log analysis using the cloud to identify suspicious activities, including logs from Amazon Web Services and Azure



Cyber Warfare – Truth, Tactics and Strategies – Second Edition

Dr. Chase Cunningham

ISBN: 978-1-83921-699-2

- Hacking at scale – how machine learning (ML) and artificial intelligence (AI) skew the battlefield
- Defending a boundaryless enterprise
- Using video and audio as weapons of influence
- Uncovering DeepFakes and their associated attack vectors
- Using voice augmentation for exploitation
- Defending when there is no perimeter
- Responding tactically to counter-campaign-based attacks

Leave a review - let other readers know what you think

Please share your thoughts on this book with others by leaving a review on the site that you bought it from. If you purchased the book from Amazon, please leave us an honest review on this book's Amazon page. This is vital so that other potential readers can see and use your unbiased opinion to make purchasing decisions, we can understand what our customers think about our products, and our authors can see your feedback on the title that they have worked with Packt to create. It will only take a few minutes of your time, but is valuable to other potential customers, our authors, and Packt. Thank you!

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

Z