1 Asset Management

Asset management is one of the most critical components of a vulnerability management program (VMP). Of all the fundamental building blocks of a successful VMP, it's crucial to get asset management right and complete before focusing on other aspects of vulnerability management.

Asset management is the listing or inventory of all hardware and software of an environment. Each environment has a different makeup of assets, including everything from mobile devices (e.g., laptops and cell phones) to application libraries and third-party software-as-a-service (SaaS) software. Without a comprehensive asset management program, organizations are limited in building mature VMPs with secure configuration, patch management, and continuous monitoring.

Asset management has evolved quite a bit over the last 10 years, with the advent of cloud infrastructure, increased use of SaaS, exponential growth of open source software use, and incredibly large and complex development environments. Years ago, asset management could be as simple as a spreadsheet with a list of asset names, tag numbers, and potentially an asset owner or IP address. Hardware and software inventories were kept separately and possibly managed by that same IT administrator. Yet with the increased use of cloud infrastructure, whether infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS), or SaaS, traditional asset management methods are simply no longer viable. Using a spreadsheet to manage complex and dynamic assets is not maintainable or feasible to keep updated information available for management.

Traditional vulnerability management components are no longer able to mature with manual or incomplete asset inventories. It's increasingly difficult to manage dynamic assets such as containers, which are meant

to come online and be torn down at will. These asset types require a dynamic asset management program—one that can be updated quickly and at scale with large-scale development projects. An asset library can no longer be solely used for managing mobile devices or hardware assets but must be capable of keeping updated information on ephemeral applications and tools.

Without a modern approach to asset management, organizations have limited visibility of the hardware and software used by employees, which can have several cascading effects. Without knowledge of a laptop, for example, there is no way to determine if it has proper monitoring software installed, if it's still in the employee's possession, if it's checking for updated patches, or if it's compliant with organizational policies. And if an organization does not have the ability to see what software is installed on what systems, they have no way of knowing the number of vulnerabilities it has, what its potential attack surface is, or what dependencies that software might have on other systems.

Other limitations of an immature asset management program are the "unknown unknowns." If there are hardware or software assets that aren't effectively managed or visible to IT operations staff, organizations do not know the scope of vulnerabilities, inherent risks, or the interconnectivity of devices and applications. These limitations make it impossible to prioritize and remediate vulnerabilities effectively. It also makes it difficult to determine if applications are at the right patch level, if the application's version is at end of life/support, and if there are outstanding vulnerabilities or missing configurations that could lead to cyberattacks like distributed denial-of-service (DDoS) attacks, malware, or ransomware.

Asset management can be performed in a variety of ways. Organizations are using IT operations software, vulnerability scanning tools, cloud inventories, and even other configuration management software like ServiceNow (`www.servicenow.com`). This type of software can not only keep track of assets, but can also tie tickets and ongoing management of those devices with a system owner. Smaller organizations might still be managing assets manually, which limits the maturity and capability of a VMP. In this chapter, we discuss the common limitations of asset management tools and processes, possible impacts of an immature asset management program, and what organizations can do to create a modern approach to asset management.

Physical and Mobile Asset Management

In traditional data centers, asset management consists of the physical components in server racks—for example, networking devices, servers, power management, and any other physical devices in the organization. However, organizations have moved to a much more digital workforce, utilizing multiple mobile devices per employee. One employee might have a tablet, laptop, and smartphone, and use primarily online applications for collaboration versus solely working on a physical desktop located in an office setting.

Many organizations are moving to hybrid work environments where employees are working between an organization's office and their home or an off-site location. This type of work environment complicates the management of these devices, given that they may or may not be connected to the organization's virtual private network (VPN) or potentially cloud assets and servers. This setup has increased the challenge of managing and tracking mobile devices.

In modern organizations, managing all these mobile devices requires an asset management solution to handle all the operating systems (OSs) and types of applications required for online collaboration. A mobile toolkit includes asset management and inventory software, as well as configuration management, usually performed by a mobile device management (MDM) solution. This tool provides a management console to catalog each mobile device and assigns policies and security configurations as determined by the organization.

Several SaaS solutions are also available as well as tools provided by the mobile carrier. For example, mobile solutions provided by Apple (e.g., iPhones and iPads) have their own asset management solution like Jamf software. Other devices or applications, however, can be managed by MDM solutions like Miradore and Citrix Endpoint Management.

Because most organizations are moving away from on-premises data centers, there are fewer servers and network devices requiring asset management. With the advent of the cloud, more organizations are migrating their physical assets to a cloud infrastructure and using more ephemeral servers like containers. Yet on-premises data centers still require an asset management solution to provide full visibility to all systems. And it's not just for security reasons—they also must manage systems and ensure they are properly online and functioning without hardware failures. All the physical assets could be providing warning

indicators of cyberattacks, and if not monitored properly, an organization could be missing critical data to determine risk.

While physical risk management is typically focused on mobile devices, there has been an increased "return to work" effort across large organizations. It means that physical assets and MDM could grow in complexity and include a mix of bring-your-own-device (BYOD) and corporate-owned assets. Such complexity might require integration with either multiple products or the use of two separate applications to manage the physical assets, versus more configuration settings on laptops and tablets. Because most organizations use a tool for inventory and a separate tool for configuration management, this complexity adds another layer for system owners to review and manage assets for consistency.

Consumer IoT Assets

Another category of assets that has become a major risk for organizations is Internet of Things (IoT) devices. With the interconnectivity of devices, IoT could be anything from a thermostat to a treadmill, home automation devices, or wearable devices like smartwatches. Because many organizations, particularly healthcare and medical organizations, use Wi-Fi or wireless connections, employees may have the option to connect their wearable devices to the local network.

Allowing these potentially vulnerable IoT devices to gain access to the network causes many concerns. The National Institute of Standards and Technology (NIST) has published a consumer's guide on the risks and potential security concerns around IoT devices. The NIST guide, "IoT Cybersecurity Criteria for Consumer Labeling Program," came out in early 2022 and details a growing need for more consumer cybersecurity information around risks of IoT devices. The Biden–Harris administration recently released additional guidance around consumer labeling to ensure consumers understand risks associated with products (see `www.whitehouse.gov/briefing-room/statements-releases/2023/07/18/biden-harris-administration-announces-cybersecurity-labeling-program-for-smart-devices-to-protect-american-consumers/#:~:text=This%20new%20labeling%20program%20would,trustworthy%20products%20in%20the%20marketplace`).

Based on an article by Mary K. Pratt in TechTarget titled "Top 10 security threats and risks to Prioritize" on page (`www.techtarget.com/iotagenda/tip/5-IoT-security-threats-to-prioritize`), there are numerous ways that IoT devices can pose risk to organizations. One of the biggest threats to all organizations that is highlighted in the article is the increased attack surface. Similar to mobile devices and increased teleworking or mobile workforces, the more devices that connect to the network, the more risks and possible attack vectors there are. Organizations must have a good grasp of what IoT devices may exist on their network, by using either network scanning or sniffing to detect rogue or unexpected IoT devices. *Sniffing* is a technique used by hackers to detect if there are unsecured devices or systems that may be exploitable. There are many ways to detect attacks in an environment and these are covered at length in later chapters.

Software Assets

Software inventories have become an increasingly important topic. While this area will be covered in depth in a later chapter, it's important to cover the basics here. Recent attacks and zero-days against SolarWinds, Log4J, and MOVEit have been big motivators for understanding the software landscape and attack surface. To understand large attack surfaces, organizations need to catalog and inventory their use of software tools, libraries, and dependencies. A *zero-day* is a vulnerability that was previously unknown in software or hardware that can be majorly exploitable.

Without a proper software inventory, organizations may scramble to find zero-days in their applications, which leaves little time for remediation and more time for attackers to exploit vulnerabilities. With many organizations leveraging larger and more complex development environments, software asset discovery and continuous monitoring become a crucial aspect of risk management.

For example, if an organization has limited visibility into which libraries developers are adding, removing, patching or not patching, their security team will be unable to determine risk and prioritize patching and remediation. If any libraries and dependencies go undetected, or are not reported automatically to an inventory tool, the organization would be unaware of the number and severity of vulnerabilities that do exist.

Another concern is the possibility of using open source software that may not be patched or maintained regularly. And the larger the development environment, the more possibility there is for unknown and undetected vulnerabilities and missing secure configurations.

Cloud Asset Management

With digital transformation, agile software development, and an increasing focus on artificial intelligence (AI), the move to the cloud for systems is an integral step of managing infrastructure and complex development environments. More organizations are considering multicloud or hybrid cloud environments using either two cloud providers or potentially a private and public cloud deployment with the same provider. Multicloud environments allow for more resiliency and scalability, whereas private and public cloud options (i.e., a hybrid cloud) allow organizations to keep specific assets apart from the public cloud infrastructure.

Figure 1.1 provides a simple explanation of the differences between hybrid and multicloud environments. A hybrid cloud setup uses a combination of a private and public cloud option, but typically within the same cloud service provider (CSP). A multicloud solution uses two or more different CSPs to host the infrastructure.

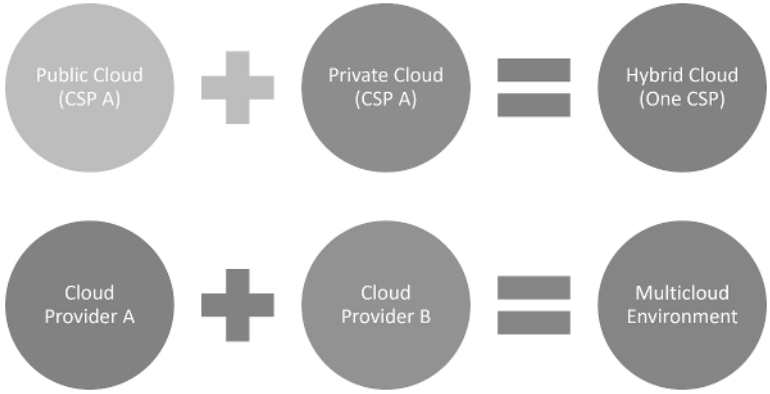

Figure 1.1: Hybrid vs. multicloud environments

Figure 1.1 shows the unique characteristics of multicloud environments compared to hybrid cloud environments. Hybrid cloud is

made up of one public cloud and one (or more) private cloud environments while using the same CSP, whereas a multicloud solution uses a combination of private and public cloud environments across multiple CSPs.

Multicloud Environments

In some multicloud environments, an organization may need multiple cloud providers. One example is the need to run production and nonproduction workloads in one cloud environment and use a second cloud for resiliency and quick transfer in the event of network or regional failure in one of their providers. Another example is to run production and nonproduction workloads in one cloud environment and have backups and long-term storage for recovery in the event of data loss in another cloud environment.

Unfortunately, using multiple cloud providers complicates an asset management strategy. One of the biggest concerns of using multiple cloud providers in a multicloud strategy is that collecting, automating, and keeping track of assets between both environments may require multiple tools. There are more modern organizations using a multicloud strategy and third-party tools can sync data between those disparate workloads. Tools like CloudSphere are working to solve secure configuration and inventory concerns by collecting and maintaining asset data. But this means that each cloud environment may need to open various ports and create service accounts to manage the information. It would be incredibly easy to lose sight of the ephemeral systems of each environment unless they were mirrored.

Hybrid Cloud Environments

A hybrid cloud solution could potentially be used for similar reasons, but the architecture is quite different. A hybrid cloud utilizes both public and private cloud environments. Organizations, for example, might use this strategy to store certain high-impact data and assets in the private cloud, while keeping lower-impact items in a lower-cost public cloud environment. This may complicate asset management in a few ways, but it can also be beneficial for organizations looking to strategize spending and budget over time. Hybrid cloud environments can also be a great solution for organizations who want to keep intellectual property

(IP), personally identifiable information (PII), or other sensitive data in a private cloud, while keeping other data and workloads that are less critical to the business in the public cloud environment.

Figure 1.2 highlights the various layers within an organization for which you should build an IT infrastructure. The top layer includes everything from the platform to cloud management and infrastructure, as well as the overall networking architecture. The mid-layer includes the infrastructure operations applications, development environment, and the major security components that continuously monitor the environment. A sovereign cloud environment is one where the provider stores each organization's data within their own country.

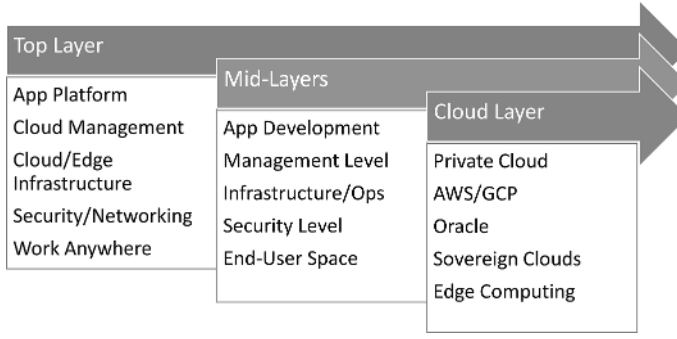

Figure 1.2: IT infrastructure layers

Figure 1.2 illustrates the differing layers in platform services in cloud environments. In the top layer, there are services like cloud and edge computing, management interfaces, as well as the application platform. The middle layers are composed of development environments, infrastructure operations, as well as the largest security components. The cloud layer is really the platform itself, whether it's Amazon Web Services (AWS) or Oracle.

One of the main concerns in using hybrid cloud solutions is the potential limitations between the private and public cloud environments. These limitations include the sheer complexity of managing two separate cloud environments as well as the security concerns of using two separate cloud environments and manually implementing the same controls. Another possible solution would be to run the same tool in both environments to segment the networks and aggregate the data elsewhere. But allowing access to the private cloud from the public cloud could increase the risk of compromise between both environments.

Third-Party Software and Open Source Software (OSS)

Many traditional asset management tools did not account for third-party software or open source software (OSS) being used in modern organizations. But the rampant use of OSS has complicated the asset management and software library processes and the ability to calculate risk.

As displayed in Figure 1.3, software assets are used across the various enterprise layers. Starting with the business layer, applications like Java and Log4j (i.e., OSS components) build the foundation for development environments. Additional software in the presentation and service layers may be required to integrate and communicate to build complex applications.

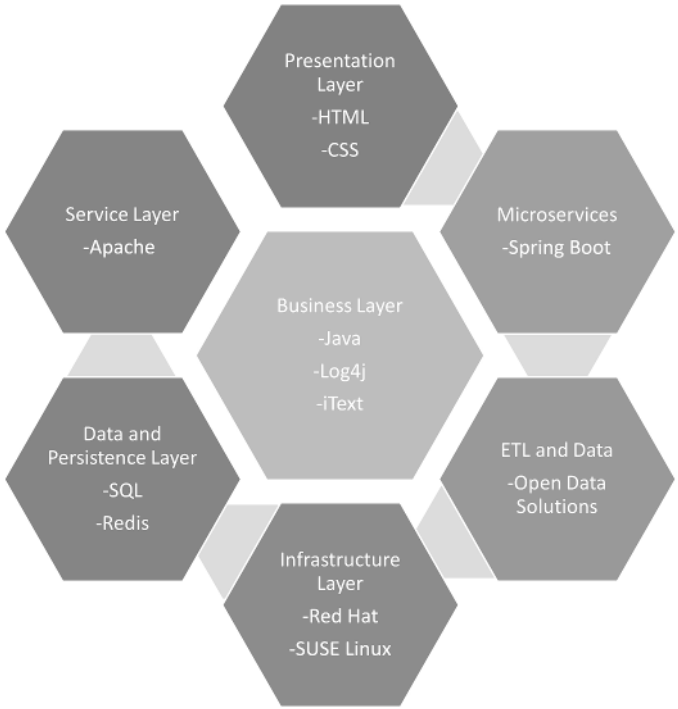

Figure 1.3: Various enterprise layers

Figure 1.3 outlines the various layers that work together across an enterprise. The business layer is the backbone of the rest of the

layers, and it has major connectivity between all the other enterprise environments—everything from the data and persistence layer that contains databases, to the infrastructure layer using Red Hat and OS components. Each piece of this matrix works together to create a comprehensive platform to support business functions.

Due to the increased OSS use, organizations are witnessing the dependencies and intricacies of how OSS works in complex and large application environments. Many developers leverage OSS because of the mean time to delivery, meaning the developer can spend less time rebuilding code that already exists by using tools that other developers have built. Lowering their time spent coding and providing some consistency in their applications allows developers to spend their time on more complex and nuanced development cycles. Yet with the increased use of OSS comes the need to catalog and understand what types of libraries and tools are being used within the applications.

Third-Party Software (and Risk)

One of the difficult items to collect and maintain within an inventory is the number of third-party companies and applications, contractors, SaaS products, and any other external software or hardware involved. For example, an organization might choose to use a firewall service provider rather than running their own firewall appliances and network configuration, due to a lack of skilled personnel or other resources to manage those assets.

Another third-party assets example is when an organization outsources their accounting or IT helpdesk firm. These third parties must have access to corporate resources, potentially requiring domain credentials or open ports/access to an organization's SaaS or infrastructure. A third-party contractor might have mobile devices that require access to an environment, thus spreading the potential attack surface.

Since the early 2020s, malicious actors have been leveraging open account access or infrastructure from third-party applications to gain access to corporate secrets. Cataloging these third-party applications can be performed using a variety of tools and methods but may be discovered by vulnerability scanning tools like Tenable or Qualys. Therefore, it's critical for organizations to determine what method is best for discovering and monitoring these third-party applications in the environment to protect themselves from risk.

Accounting for Open Source Software

Static lists will not capture changing versions, patches, or removal of any OSS within an environment. Organizations must move to dynamic asset discovery and categorization because of the possibility for human error and missed assets with a manual process. Every missed asset is a possible entry point for an attacker with exploitable vulnerabilities or misconfigurations. The process should be as automated as possible—allowing developers to consistently change their applications without running into major hurdles with configuration management activities.

Using something like GitHub or another open source tool (made for developers) is a possible solution for dynamic OSS application inventories. The recommendation is to use the open source repository that the developers are already using, whether that's GitHub, GitLab, or another platform. The most important component of each of these options is to have a consistent process known among all developers.

Documentation and the standard operating procedure (SOP) for OSS inventory management is just as important as the tools that perform inventory management. These options allow developers to manage OSS, are usually cross-platform, and provide additional functionality over the standard cloud inventory management systems. There are also several "for purchase" options, and organizations should carefully weigh their own unique needs *before* selecting a product.

On-Premises and Cloud Asset Inventories

While many small to medium businesses (SMBs) are choosing to create cloud environments from the start, there are still many organizations who have on-premises environments or who are choosing smaller on-premises data centers to manage specific data. Because there's still a mix of solutions for organizations, this complicates the tooling landscape for managing hardware and software appropriately. Hardware in data centers includes everything from servers to network devices, as well as all the IoT devices that may tie into the corporate network. Software incorporates everything from SaaS products like email services, to the actual tools and libraries used by developers like Python and Tomcat.

In reviewing Figure 1.4, it's easy to see how complex physical data centers can be compared to their cloud environment competitors. Physical data centers require power management, servers, racks, cables, and physical storage devices like storage area networks (SANs).

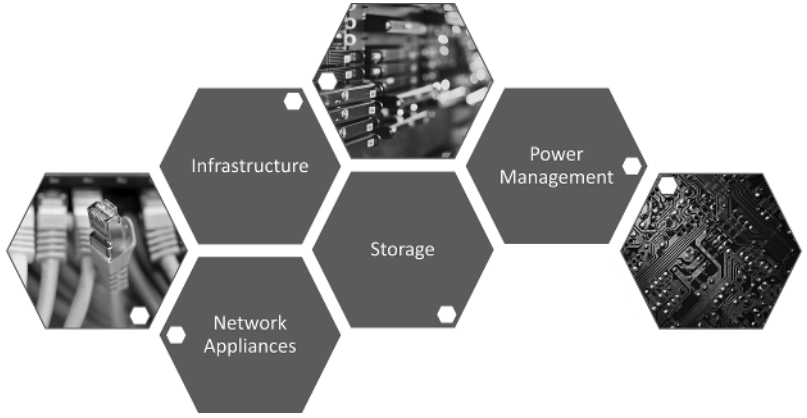

Figure 1.4: Physical data centers

Source: pixelnest/Adobe Stock Photos, khamkula/Adobe Stock and shymar27/Adobe Stock.

On-Premises Data Centers

In on-premises environments, assets are a mix of hardware and software, in addition to any other SaaS products that the organization is using. Part of the trouble is that many organizations who have on-premises environments are also supporting workloads in the cloud.

It's rare to find a tool to manage all of an organization's systems and applications and parse the information into one spot. But organizations should work toward using as few tools as possible, while also balancing the needs of an ever-changing hardware and software landscape.

Because hardware fails and must be replaced over time, having a tool like Microsoft Configuration Manager may be good for both inventory and patch management. Organizations can benefit from this automation and reduce the overhead of manual patching and remediation activities.

Figure 1.5 shows the vast difference between on-premises and cloud environments. On-premises environments require appliances and physical devices like firewalls and physical servers that will sit in a server rack, whereas cloud environments will require additional tooling to look at static and dynamic application scanning, web application firewalls, and more software devices.

In Figure 1.5, it is easy to see how different on-premises and cloud environments are, based on the types of hardware and software supported. In a data center, there are hardware firewalls and network

switches, as well as all of the physical servers that would sit in a server rack. However, in a cloud environment, there would be web application firewalls (WAFs), cloud-native tooling, as well as containers and virtual servers.

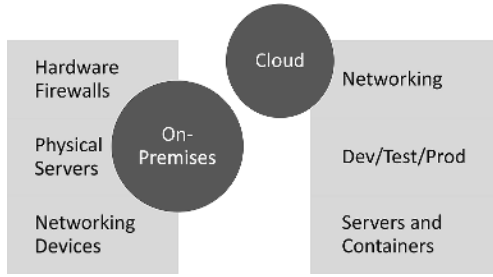

Figure 1.5: On-premises vs. cloud environments

Tooling

There are multiple tool options to determine whether assets—physical or virtual, hardware or software—are available for on-premises and cloud environments. For most organizations, a combination of inventories from cloud systems and application libraries may need to be consolidated into one platform. A few tools are available today that will catalog and categorize hardware, software, continuous integration/continuous delivery (CI/CD) pipelines, SaaS, and cloud platform inventories into one dashboard.

But there is hope; asset management is a moving target that must be evaluated any time new products or devices are brought on board. Just like patch management, monitoring and logging, and all other cybersecurity activities, asset management must be an iterative and continuous process.

Asset Management Tools

To begin, tools like Salesforce, ServiceNow, Microsoft Configuration Manager, and others, have been standard IT asset management tools for many years. Many large organizations leverage ServiceNow or a similar ticketing system because of its ability to catalog assets and assign tickets for maintenance and operations to those assets and

their respective owners. However, this may not be an option for SMBs. Smaller organizations may need to leverage open source tools or the inventory management systems that come with their CSP. If you're using a small cloud environment, whether private or public, it makes more sense to leverage the CSP's in-house capabilities and compare those results to a vulnerability scanner for validation. One possible open source tool is Asset Panda, which can be used to manage inventory for cloud environments.

The most important point of choosing an asset management tool, whether off-the-shelf or open source, is to select a tool that's scalable for the environment. For smaller businesses, consider an option that automatically updates inventory based on dynamic scanning. Otherwise it could become an incredibly manual task that the organization may not have enough personnel to do. Organizations should consider the best option for their environment as well—if there's a large development environment with test, development, and production in place, use a tool that covers ephemeral devices and dynamically updates. Doing so provides a real-time view of the environment instead of using a tool that requires manual input.

Vulnerability Scanning Tools

Organizations typically have a vulnerability management tool in place that serves as a vulnerability scanner and secures configuration validation, in addition to many other functions like reporting and asset discovery. These tools can also be used to validate inventory alongside configuration management tools like ServiceNow. Vulnerability scanning tools should not be the only source of truth, and assets should be checked regularly by the owners and operators of ephemeral systems.

Given the dynamic nature of development environments, automation should be used wherever possible to capture and remove systems when they are no longer required. Outdated systems with years of vulnerabilities take an incredible amount of time for administrators to sort through and determine what is truly vulnerable. To save time and resources, asset discovery should be automated and validated using at least two tools.

Off-the-shelf tools like Tenable or Qualys can double as an inventory checking tool to ensure that all assets are being scanned for vulnerabilities and secure configurations. Most of these tools serve double duty

to validate that the servers, applications, and other systems in place are being scanned properly. If an asset is missing from the scanner, the tool can be updated with proper IP ranges, or even be set up to discover unknown assets by scanning the entire network. Daily or weekly reports can then be configured to notify system administrators and account owners of these new or unexpected servers and systems.

Cloud Inventory Management Tools

Cloud inventory management is easier with AWS or Google Cloud Platform (GCP) because they have inventory built into the management console. This is in stark contrast to managing a data center inventory where assets need to be managed with some additional tool or managed via a spreadsheet. Major cloud providers have made it much easier to identify, manage, and organize assets, even over multiple cloud accounts. There are an incredible number of benefits to using cloud systems, including the built-in categorization for containers, servers, workers, nodes, and more.

AWS has the AWS Systems Manager Inventory, GCP has the Cloud Asset Inventory, and Microsoft Azure uses an inventory system called Change Tracking and Inventory in their Azure Automation suite. These cloud inventories provide the ability to categorize systems and include tagging or metadata for system type, system-specific drivers or components, as well as instance details and network configuration. Microsoft's tracking and inventory tools leverage Log Analytics to monitor and manage assets.

Cloud providers have integrated several essential components into their inventory systems, so a traditional inventory tool may not even be capable of tracking their inventory. For example, the ability to leverage log analysis within an inventory system provides cloud engineers and security analysts with the ability to review logs without switching from an inventory management system to a security information and event management (SIEM) tool.

Organizations benefit from these cloud inventories based on their ease of use, "single pane of glass" to review assets, and lower administrative overhead to manage multiple tools. A "single pane of glass" means that there is one dashboard or panel that a group of security tools can be part of. This allows an organization to review one website instead of administrators or analysts having to log in to multiple tools and use

numerous dashboards. Using a cloud environment may also reduce the number of separate tools required to manage the infrastructure, also reducing risk and cost to the organization.

Figure 1.6 illustrates the increasing complexity when starting with an asset inventory system, as well as the additional tools and considerations for the whole environment. Having an inventory management system is the backbone to building a comprehensive security tooling strategy.

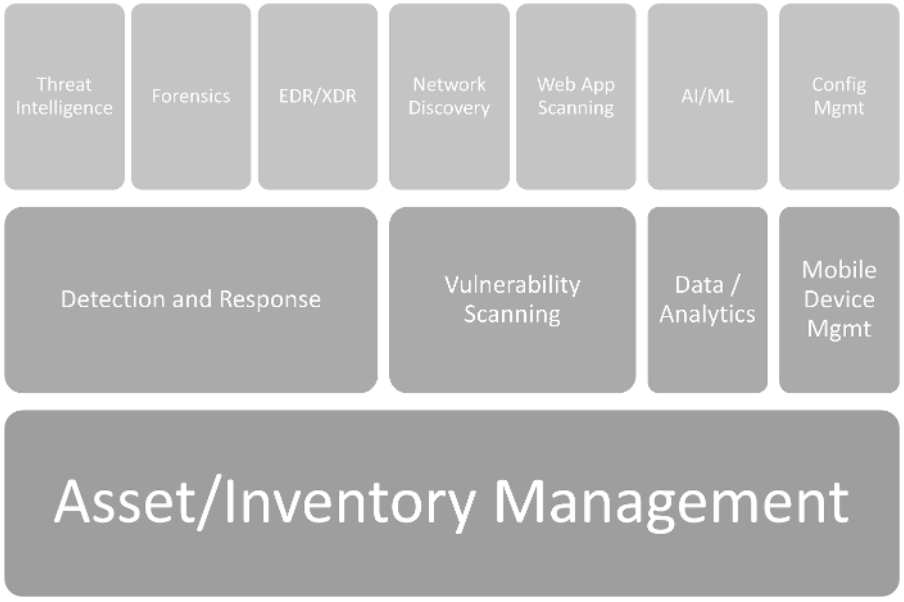

Figure 1.6: Complexity of an asset inventory system

Ephemeral Assets

One of the major challenges in modern IT infrastructure and cloud environments is *ephemeral assets*, which include containers, nodes, workloads, and several types of modern container technologies. The typical life span of these system types can be anywhere from minutes to hours to days, and they only come online to serve a scalability purpose. For example, a new worker node may come online during business hours to support the increased workload from major website traffic. That node may go offline at 5 p.m. and a new instance is then brought online the

next morning at 8 a.m. So, how can an organization's inventory systems and containers stay online for minutes or hours? Easy; they can utilize any number of the cloud-native tools and automation when bringing servers and containers online to keep their assets up-to-date.

Accounting for ephemeral assets is a growing requirement for organizations, but it is incredibly difficult because accounting for these systems by hostname or IP is not sufficient. Because they're only online for minutes or hours, asset inventories must be dynamic and account for IP ranges and expected hostname ranges, versus a static hostname or IP. Many cloud environments can monitor these assets, but vulnerability scanners and endpoint detection and response (EDR) solutions must be configured properly to monitor entire IP spaces and use dynamic scanning to capture all systems.

If there are static IPs or hostnames in those scans, these scans will be insufficient for finding vulnerabilities and providing an accurate view of the vulnerability landscape. Organizations will need to not only review their inventory tool, but also their vulnerability scanners and any other security tools to ensure the entire organization is accounting for ephemeral assets.

Sources of Truth

Many security teams, developers, and platform engineers use multiple sources of truth to validate vulnerabilities, secure configurations, and asset/inventory management activities. A *source of truth* is the ability for an organization to aggregate data into a single dashboard or tool to verify that (in this case) the assets are configured properly. This would be a combination of multiple tools and utilizing that "single pane of glass" mentioned previously. If a team is using a single tool for all these activities—for example, using a vulnerability scanner to double as an inventory record system—the team might be missing out on assets that either cannot or do not get scanned by those tools. One concern would be whether the team is using a tool that does not have the functionality or ability to identify and manage certain types of assets.

For example, it's possible that some vulnerability scanners aren't able to inventory or find vulnerabilities on containers or infrastructure as code (IaC). Because of this limitation, organizations must understand the full functionality and capability of the tools that they use for specific purposes. A vulnerability scanner without the ability to find

vulnerabilities in a container would not be sufficient for a cloud workload that consists of only containers.

Organizations should have a primary source of truth—a configuration management tool that they can rely on for enterprise asset discovery and management. However, a VMP should have a second source of truth to validate that the proper assets are in place, being patched consistently, and scanned regularly by both the vulnerability scanner and the EDR solution. The ability to validate between two tools provides clarity for teams and prioritization to investigate any inconsistencies between the tools.

Asset Management Risk

Asset management is the basis of any vulnerability management program. Without a comprehensive understanding of all possible assets across an organization, it's impossible to understand the organization's risk landscape or to even prioritize vulnerability management activities. The "unknown unknowns" are typically the highest risk to any organization, because without an understanding of assets there's no way to know which vulnerabilities are still exploitable on the network. Without any visibility, the risk of an organization moves from a "known known" to an "unknown unknown."

Log4j

An excellent example of the need for proper software inventories is the incident that occurred in December 2021 called Log4j, or Log4Shell. Based on the guidance from the Cybersecurity and Infrastructure Security Agency (CISA), CVE-2021-44228 was a remote code execution (RCE) that malicious actors leveraged to gain access to systems, conduct a ransomware attack, and exfiltrate data (`www.cisa.gov/news-events/news/apache-log4j-vulnerability-guidance`).

The UK's National Cyber Security Centre (NCSC) noted just how prevalent Log4j was based on the open source nature of the tool and how developers needed such a tool for logging functionality (`www.ncsc.gov.uk/information/log4j-vulnerability-what-everyone-needs-to-know`). The NCSC also noted, however, how difficult it was for organizations to determine or identify that Log4j was in their environment.

This article also mentions the increased need for the communication required between vendors and developers. The Center for Internet Security (CIS) released an article discussing the severity of the vulnerability, rated a 10.0 on the Common Vulnerability Scoring System (CVSS; `www.cisecurity.org/log4j-zero-day-vulnerability-response`).

One of the major challenges with this incident was the inability for organizations to find and detect Log4j instances within their pipelines or development environments. Any delay in patching this vulnerability left precious time for malicious actors to exploit this vulnerability. Because the vulnerability was made public, it was possible for actors and hackers to exploit the vulnerability in the time it took organizations to identify whether they had Log4j in place. This example highlights the need to understand the software, libraries, and dependency components of any development environment. Organizations must have a dynamic library of all software components, both off-the-shelf and OSS. OSS will be covered more in-depth in later chapters, but the Log4j incident highlights the harmony required between asset management and OSS.

Missing and Unaccounted-for Assets

This section might seem redundant—but enterprise risk around assets is tied very closely to unknown or unaccounted-for hardware and software assets. Without the proper inventory or management of devices, servers, containers, and applications, organizations cannot account for those risks. Any server that's unaccounted for is most likely not being patched, missing secure configurations, and ultimately increasing the risk for the whole environment. That server could become the entry point for an attacker or actor to gain access to privileged credentials or compromise the entire network. Each hardware or software asset that's unaccounted for is a potential entry method, providing the ability to gain a foothold in the environment.

It's also impossible to monitor servers and systems for compromise if they are not in the EDR or asset inventory lists. Administrators and security analysts would be unaware of such compromises and not be able to monitor and review potential alerts on unmanaged devices and applications. If there is malware, a potential compromise, or even a system that's being leveraged to gain access to other systems, it may go undetected and unnoticed until the asset comes into the inventory management system.

Detecting hardware and software assets necessitates both tools and processes to detect, monitor, and bring those assets into alignment. An organization's asset management team should be heavily involved in the process and policy, continuously monitoring for unaccounted-for and missing assets.

Unknown Unknowns

In the world of vulnerabilities, there are *known knowns*, *known unknowns*, and *unknown unknowns*. What is of major concern to any organization are the unknown unknowns. Based on the 2022 article by Nathan Wenzler from Tenable, unknown unknowns are a major concern when understanding an enterprise risk landscape (`www.tenable.com/blog/finally-finding-the-unknown-unknowns-across-your-entire-attack-surface`).

These unknown unknowns (henceforth known as UUs) are typically items classified as zero-day or being actively exploited in the wild by the time that they're disclosed. These types of vulnerabilities are incredibly difficult to plan for within a VMP. But VMPs should have their own processes and people in place to manage events and be prepared when these vulnerabilities are found.

One example of a UU is the SolarWinds attack that took place in September 2019. The U.S. Government Accountability Office (GAO) in 2021 noted that the SolarWinds attack was the largest hacking campaign against both government and private organizations (`www.gao.gov/blog/solarwinds-cyberattack-demands-significant-federal-and-private-sector-response-infographic`). What is interesting is the SolarWinds attack was just the beginning for software supply chain attacks and the understanding that even patches can contain malicious code, leaving organizations vulnerable to UUs.

To plan for such events, organizations should have a strategy and procedure to handle UU events like zero-days and highly exploitable vulnerabilities. Using resources like the CISA Known Exploitable Vulnerabilities (KEV) catalog and the National Vulnerability Database (NVD), organizations can set up alerts for when possible exploitable vulnerabilities have been released. These processes tie well together with an incident response (IR) plan and any cyber-resiliency tooling and procedures.

Patch Management

Patch management is one of the major areas of concern when it comes to what servers, systems, and applications exist in any on-premises or cloud environment. And of course, without a thorough understanding of what hardware and software assets exist, an organization will have an ineffective patch management program. If an asset program is missing any systems or devices, it will be impossible to determine what patches are missing, leading to unknown risks.

To create an effective patch management strategy, an organization must first understand the levels of OSs, applications, libraries, container versions, and so on. Each of these pieces helps create a complete picture of their vulnerabilities and allows organizations to prioritize remediation activities.

An example of the complexity of software asset patch management is the migration from one OS to another. Chances are that an organization who's migrating from one server version to another (e.g., Windows Server 2019 to 2022 or Red Hat Enterprise Linux [RHEL] 8 to 9) will be managing two levels of patch sets. The administrators will need to download and install patches for both version levels, meaning that they will need to ensure that the proper patches are being both downloaded and installed properly. This necessity doubles administrative overhead unless the team leverages some automation or automatic patching for servers and applications.

Patch management will be covered more in-depth in the next chapter, but the connection between inventory management and patch strategy is undeniable.

Increased complexity with patch management starts when an organization cannot reboot or patch systems outside of maintenance windows. Many environments will require some customization and considerations for service level agreements (SLAs), customer requirements, maintenance windows, and stability of the environment. There will always be unique requirements, but organizations must have a patch management strategy to combat these complexities. Recommendations include building resiliency into systems that are considered unstable, patching test and development environments first for monitoring, and instituting a rollback plan for any patches or secure configuration changes.

Recommendations for Asset Management

There are several recommendations for organizations, large or small, to get a handle on their assets. Whether on-premises, cloud-based, multicloud, or any combination of software and hardware assets, organizations must start with the people who will manage those assets. The people-process-technology aspect of asset management is an important combination of having the personnel who own and manage the assets, installing the proper tooling in place for inventory systems, and understanding the processes to discover, manage, and organize their assets. This section covers multiple areas for organizations to consider when creating an asset management program.

Asset Manager Responsibilities

The people part of the people-process-technology trilogy is just as important as the tools and processes used for inventory management. An account owner might be designated to manage the cloud infrastructure, an operations team may be in place to manage the OS and application layers, and a security team may be in place for vulnerability management. But without properly identifying who will own the asset inventory tooling and processes, organizations may spend hours, days, or weeks trying to find the proper technology owner. This wastes company resources and time, and potentially compiles risk for vulnerable systems without an owner.

Designating a primary and secondary asset manager helps alleviate some of these potential concerns. A primary asset manager should be making the decisions about the frequency of vulnerability scanning, continuous monitoring, provisioning and decommissioning of systems, and the categorization of the data. For a thorough understanding of how to catalog and understand asset management, any organization can start with the Risk Management Framework (RMF) from NIST.

The NIST RMF is a framework used to help organizations understand and determine their security and risk management activities. There are seven steps to the RMF: preparation, categorization of systems, selection of security controls, implementation of controls, assessment, authorization of a system for use, and finally monitoring of the selected

controls. This framework is a starting point for any organization to determine risk in their environments and continuously monitor to ensure the proper security controls are in place.

A secondary asset manager provides backup, can perform an additional check for unexpected devices, or aids in understanding workloads to determine expected versus unexpected systems. Having a secondary manager also provides an extra layer of security when the primary leaves the company, goes on vacation, or simply needs another pair of eyes on an incident. Having an additional pair of eyes on missing devices can provide context as well.

Asset Discovery

Asset discovery is the continuous monitoring component of any asset and configuration management program. With the increased complexity and incredible speed at which systems are built, brought online, and put into production, organizations need to continuously detect and monitor new systems. This ensures that any team is aware of expected versus unexpected (or rogue) devices on their network. As mentioned previously, having assets that are unknown increases risk and there is no accounting for vulnerabilities, missing configurations, or even understanding if rogue devices have been placed on the network.

To start an asset discovery process, you will need the right tools in place. Use discovery tools to regularly scan for expected assets, or use the dynamic scanning process that exists in your current tools. For example, vulnerability scanners may have special scans to monitor the entire network and report on any new and unaccounted-for systems and servers. These reports should be run daily and alerts should be set up for any unexpected devices.

After these reports are put in place, a process should be enacted to determine why a system wasn't accounted for and who the system's owner is, and to catalog that asset properly. For cloud environments, the in-house inventory system is a good starting place and should be compared to any of the other scanning or inventory tools that the team employs. This asset discovery program is a balance of tools and processes to ensure that no rogue or unexpected systems are online.

Part of IT project management is asset discovery and the ongoing management of the asset and configuration management processes. To align with the asset discovery tooling, the processes must pick up

where the tooling leaves off. To do so, proper alerting and notifications should be installed to notify the appropriate system or account owner. This alert or notification should trigger a process to assign the right technical owner, bring the system into alignment with the organization's inventory, and then have continuous monitoring and vulnerability scanning occurring. Similarly, if an unexpected server is brought online and discovered, an incident response process should be triggered to notify the security team and contain or investigate it, as appropriate.

Getting the Right Tooling

Asset management is part of everyone's responsibility. Leveraging a tool like ServiceNow for asset and configuration management is part of the puzzle but not the entire picture. Each organization has their unique requirements, but it is important for the IT, development, and management teams to come together to select a tool that will work for today and for five years from now. When organizations are first formed, it's a perfect time for the IT, security, and architecture staff to come together to decide on a solution. But asset inventory tooling will not necessarily grow and scale with the organization; the tooling should be reviewed and measured over time for success.

Organizations many years ago maintained data centers or server rooms. Then "the cloud" became a viable solution for many businesses to reduce cost and speed up access to development resources. It created a more agile and scalable environment that allowed organizations to move away from physical data centers.

Now organizations are leveraging hybrid or multicloud solutions to increase scalability even further. To reduce risk, and administrative overhead of OS-level infrastructure, organizations are now moving to infrastructure as code or low-code solutions to manage their platforms and infrastructure. Each step of the way, organizations are looking to reduce cost and overhead, and allow for lower risk and a smaller attack surface.

Figure 1.7 shows the progression over the years from on-premises and physical infrastructure, through the great cloud migration, and now to infrastructure as code.

As an organization grows, the asset inventory and configuration management tools need to grow too. For example, an organization that manages one server rack in a local data center has very different

business requirements than a multicloud environment. The right tool needs to be scalable and grow with the organization, but that's not always possible. So as other security tools and vulnerability scanners are being reviewed for efficiency, the asset inventory tool should be reviewed at the same time. If deficiencies or missing functionalities exist, consider the cost of ripping and replacing before choosing a new tool. An example of when an organization does a "rip and replace" is when a company removes one vulnerability scanner and replaces it with another vulnerability scanner from a different vendor. One example is removing Tenable Security Center and replacing it with a Qualys suite of tools. This would require the removal of servers and infrastructure, and applying all new permissions and access for the new tooling. The cost and management of these tools is just as important as the automation and reduction in overhead.

Any IT or security team should also be aware of asset inventory tooling that could potentially lead to confusion, mismanagement, or inconsistent information.

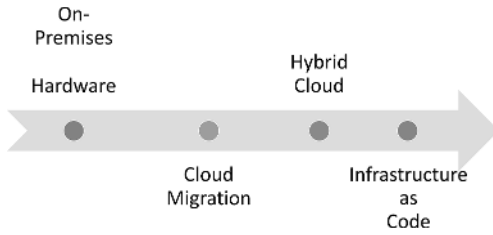

Figure 1.7: Progression of organizational management over the years

Digital Transformation

Digital transformation should be aligned with an asset management provisioning and decommissioning process. In a 2021 article by Michael Pease from NIST, he noted that digital transformation (DX) is divided into three phases: digitization, digitalization, and digital transformation (`www.nist.gov/blogs/manufacturing-innovation-blog/supporting-digital-transformation-legacy-components`). Basically, this means bringing data to the business using more digital assets and transforming the business to move away from legacy systems. Digital transformation requires enhanced technical skills to manage dynamic and emerging technologies (see Figure 1.8).

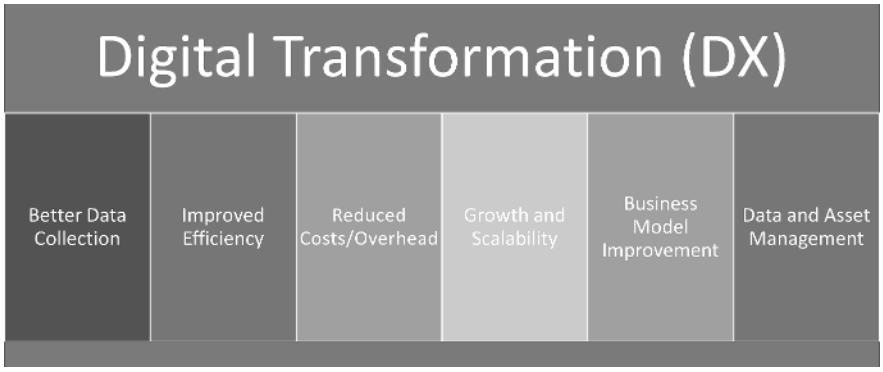

Figure 1.8: Digital transformation (DX)

Speaking of legacy systems, organizations will see the decommissioning and removal of many legacy applications, systems, devices, and appliances as they move toward a more DX strategy. Improving a DX strategy means having the ability to remove and decommission end-of-life (EoL) and legacy applications quickly and efficiently.

There are so many benefits to using a DX strategy, including lowering risk, removing old vulnerabilities, and implementing newer applications and methods of development that reduce complexity. However, with DX comes the need to manage newer systems that the current workforce may not be prepared for based on their current skillset. Along with using updated methods for inventory management, IT and development teams will need training to manage new and emerging technology.

Establishing and Decommissioning Standard Operating Procedures

A final recommendation for organizations is the process behind the inventory management tooling. For asset discovery and management, the processes and procedures are just as important for both auditing and an overall understanding of the environment. The establishment of service documents should outline the inventory component of bringing systems online, whether they are servers, containers, or applications. This document, known as the standard operating procedures (SOPs), provides the steps to validate that the system is checking the right inventory tool and is being properly scanned by the vulnerability scanner of choice.

The SOPs ensure that no system is brought online without some visibility by the administrative and operations team. Similarly, a decommissioning process will ensure that any system taken offline or removed is documented properly and removed from the inventory as appropriate. Having systems showing as still online and vulnerable can be confusing and waste time for operations teams, versus spending time on real live vulnerabilities.

Decommissioning and removing EoL products is an essential part of the asset management program. Organizations can determine their own schedule for validating if their systems are online or offline, but within minutes of a system coming online, it should be scanned before being put into production or set as externally facing.

Protection of new systems is paramount to ensuring that vulnerable or misconfigured systems do not enter a production environment without being cataloged and scanned. Without these processes in place, systems that are vulnerable may be set to be decommissioned but stay online and accessible to malicious actors, opening an organization to risk. Having an asset inventory and management program is not just good IT practice, but a risk management activity to protect the business.

Summary

This chapter encompassed a number of topics within asset and inventory management. It began with the physical components of organizations like on-premises data centers and mobile device management, then covered the challenges of managing that physical infrastructure. Moving from physical environments, the chapter covered all aspects of cloud assets and software management managed in hybrid or multicloud environments. Risk was discussed across all types of environments, focusing on how important it is for organizations to understand their assets to determine the appropriate risk level for their infrastructure. Starting with a proper asset and inventory management program is crucial, especially as organizations then consider a patch management process, which will be discussed in the next chapter.