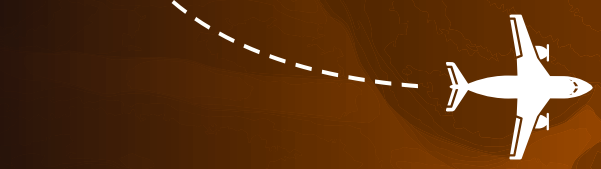
SANS



CLOUD SECURITY FOUNDATIONS, FRAMEWORKS, AND BEYOND

In partnership with:

aws Google Cloud Microsoft

Forward

Multicloud is a reality for organizations of all sizes. Aa a result, security leaders need to build capabilities and expertise for any cloud provider that the business chooses. How can security professionals get a handle on this complex world of cloud security?

For the first time, in this book, we have security leaders from the three major cloud providers – Amazon Web Services (AWS), Microsoft Azure, Google Cloud – along with independent technical experts from SANS Institute sharing where cloud security has been, mistakes that have been made along the way, and what the future may hold.

The book covers foundational principles and strategies for cyber defense to mitigate risk. A key part of this is understanding the myths, missteps and best practices that arise in cloud migration. As organizations mature, a more comprehensive plan is also required. This is where Zero Trust provides important architectural principles for modern security capabilities. Looking forward even further artificial intelligence (AI) promises to help improve our cloud security posture and ability respond to threats even more effectively.

As you go through this book I suggest using it as a guide. Know that there will be many twists and turns on your journey but, by leveraging the practices introduced here, you can get a handle on cloud security.

Frank Kim

Fellow and Curriculum Lead

SANS Institute

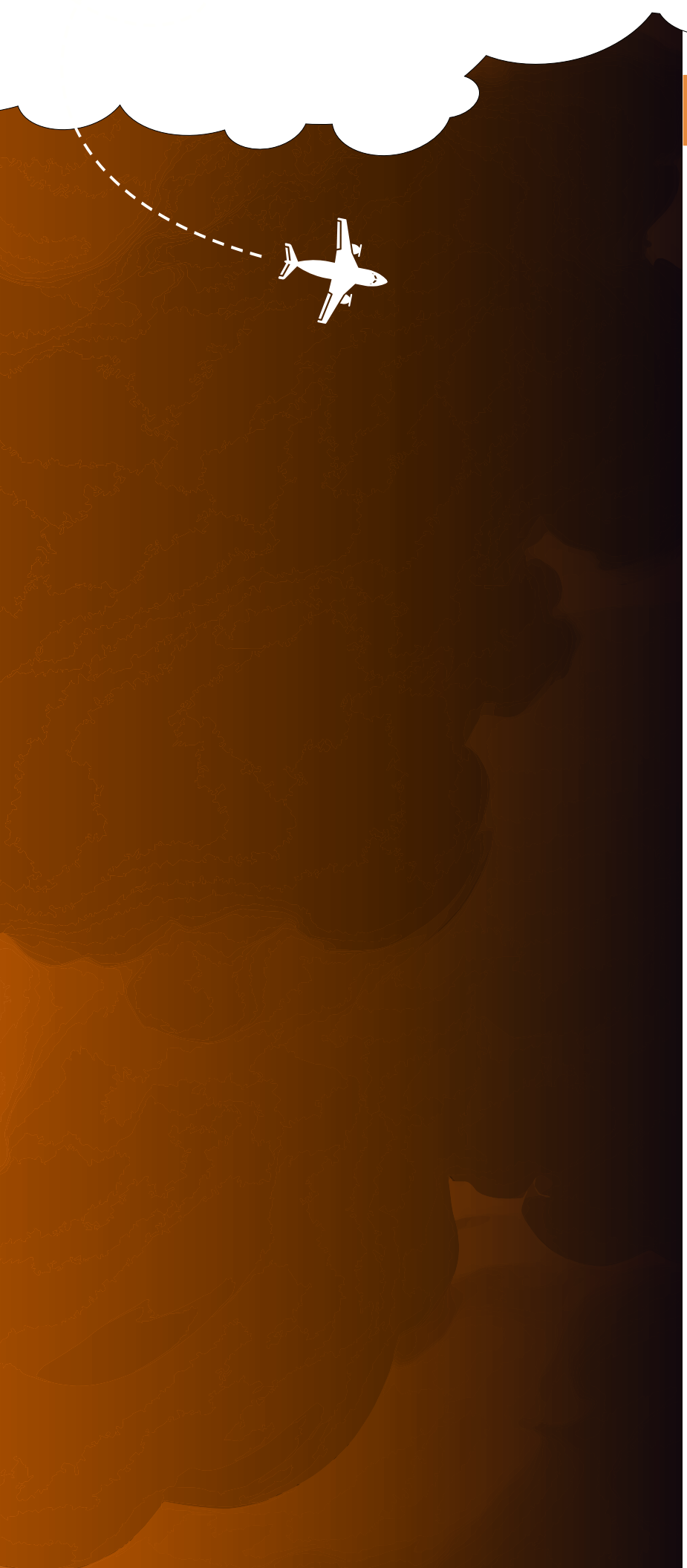


Table of Contents

Chapter 1

Zero Trust: Charting a Path to Stronger Security

Written by Mark Ryland and Ashish Rajan
Contributor: Quint Van Deman

Introduction

Security has become a top priority for organizations looking to build customer trust, enhance workforce mobility, and unlock digital business opportunities. However, the traditional approach of defined security perimeters that separate "trusted" from "untrusted" network zones has proven to be inadequate. Today's distributed enterprise requires a new approach to ensuring the right levels of security and accessibility for systems and data. Increasingly, zero trust is being described as the solution.

Gartner, a company that delivers actionable, objective insight to executives and their teams, predicts that by 2025, over 60% of organizations will embrace zero trust as a starting place for security.[1]

Zero trust is a journey that's different for every organization. For some, the journey is a natural evolution of cybersecurity in general, and defense in depth in particular. For others, it's driven by policy considerations, and by the growing patchwork of data protection and privacy regulations across the globe.

Regardless of the rationale—and despite the hype that surrounds the term—zero trust can meaningfully improve both technical and business outcomes. However, implementing a zero trust architecture is a process that requires careful consideration. Organizations often find themselves asking, "What exactly is zero trust?," "How do I get started?," "How do I make continued progress?," and "How do I demonstrate return on investment (ROI)?"

This chapter explores these important questions and cuts through the zero trust hype with best practices for designing a successful strategy that supports secure access to resources with a broad range of evaluation factors.

Defining Zero Trust

While zero trust has quickly grown from concept to strategic priority, there may still be some confusion around exactly what it is. Definitions vary, but zero trust is essentially a security model and associated set of mechanisms that focus on providing security controls around digital assets that don't solely or fundamentally depend on traditional network controls or network perimeters. Zero trust encourages you to incorporate a wide range of context about any particular access request, including identity, device, data, behavior, and more, so your systems can make increasingly granular, continuous, and adaptive policy-based access control decisions (see Figure 1).

[1] *Gartner, "Gartner Predicts 2023: Zero Trust Moves Past Marketing Hype Into Reality,"* John Watts, Jeremy D'Hoinne, Dale Koeppen, Charlie Winckless, 6 December 2022. *GARTNER is a registered trademark and service mark of Gartner, Inc. and/or its affiliates in the US and internationally and is used herein with permission. All rights reserved.*

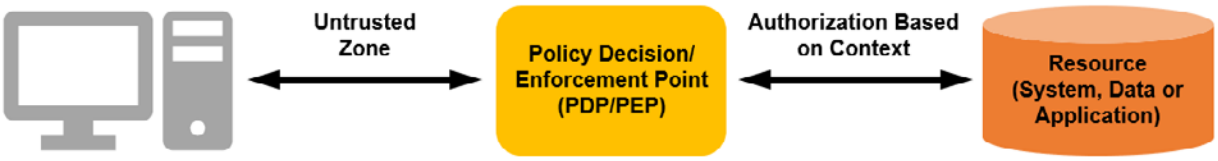


Figure 1. Zero Trust Access

The focus on access control is important because, although authentication and associated concepts like identity federation have been reasonably modernized and centralized, authorization typically remains spread across countless downstream systems. Authorization rules exist in access control lists, table grants, in-app permissions, and other similar constructs in ways that are difficult to configure and manage, much less consistently track and audit. When you distill zero trust down to its essence, ubiquitous and increasingly centralized authorization is one of the fundamental problems it aims to solve.

In practice, zero trust can also be thought of as the convergence of networking, identity, and security. Ideally, in a zero trust architecture, networking and identity-based controls aren't just simultaneously present and configured, they're actually aware of one another. An illustrative example of this is an Amazon Virtual Private Cloud (VPC) endpoint. VPC endpoints provide private network connectivity to AWS services from your own virtual private cloud, and allow you to specify access control policies. These policies and their associated enforcement engine understand not only the network, but also the identities and resources that are flowing across this border network control. They can make authorization decisions that consider this converged context.

Additionally, zero trust allows previously siloed security capabilities—such as the management of unified endpoints, vulnerabilities, service ownership, identity, and everything in between—to share data, signals, and telemetry to make more informed decisions. Improvements can come from both declarative policies that consider cross-silo factors, and from machine learning-powered processes that identify anomalous patterns or behaviors and either suggest policy enhancements to administrators or dynamically adjust authorization decisions based on risk. Convergence in these areas will take time, but it will serve as your North Star on the journey to zero trust.

> **"Zero trust itself isn't the goal; it's the how, not the what." —Mark Ryland, Director, Office of the CISO, AWS**

Tightening your focus from "security for security's sake" to objectives such as end user mobility, digital transformation, and customer trust—and the technical use cases that empower them—can help you move beyond going through the "we need to do something about zero trust" motions and articulate the need to invest time and resources in ways that relate to the business. This is important, because it makes it possible for you to stay focused on the fact that zero trust is all about facilitating desired business and technical outcomes.

Foundations and Fallacies

Zero trust requires foundational security capabilities to be in place. However, existing guidance often suggests a level of comprehensiveness, even perfection, across these foundations that can make even getting started feel like a Herculean task. It's important to understand which foundational capabilities are truly critical on your journey to zero trust and to avoid common fallacies along the way.

Foundation #1: A Solid Approach to Identity and Access Management

Identity is arguably the most important contextual factor in a zero trust authorization decision. Whether the primary actor is a user, an application, or a device—and whether the resource being accessed is on premises or in the cloud—prioritizing the deployment of several specific identity and access management (IAM) capabilities is key. These include:

- **Multi-factor authentication—**Modern multi-factor authentication (MFA) solutions, such as FIDO2 hardware-based security keys and associated processes for distribution, enrollment, and ongoing management, are vital to your zero trust efforts. The use of FIDO2 security keys, in particular, not only provides a high level of authentication assurance for zero trust authorization decisions, but also offers benefits such as phishing resistance. It also strikes an excellent balance between security (e.g., private keys that can never leave the device), usability (e.g., the user simply taps the device to authenticate), and interoperability (e.g., support that's automatically baked into modern operating systems and browsers via the WebAuthN web standard).
- **Single sign-on (SSO)—**Your MFA implementation should be paired with the services of an SSO/federated identity provider. Support for modern identity protocols that includes OpenID Connect (OIDC) for authentication, and System for Cross-domain Identity Management (SCIM) for replication of identity-related information is essential. This support is typically provided by most top-tier IAM solutions, and you can prioritize support for Security Assertion Markup Language (SAML), Kerberos, and other older protocols according to legacy and migration needs.
- **Identity governance processes—**Verify that your IAM capabilities include well-functioning embedded or surrounding processes for identity governance (e.g., covering joiners, movers, and leavers in enterprise group management). These processes, and the identity groups and attributes they control, are not only vital to authorization decisions but also serve as the basis of resource ownership information (as you'll see in Foundation #3 below).

Foundation #2: Unified Endpoint Management (UEM)

Understanding the health and security posture of a user's device is typically the second most important contextual component in a zero trust authorization decision.

You need to be confident that an endpoint is in a proper state before allowing it access to corporate data and resources. UEM solutions support this confidence by providing capabilities that include device provisioning, ongoing configuration and patch management, security baselining and telemetry reporting, and device cleansing and retirement.

Focus on the form factors most relevant to your workforce. This typically means starting with corporate-issued laptops and desktops, followed by mobile devices and cloud desktops. Depending on your organization's business needs and constraints, you may wish to consider allowing access to less confidential systems and data from uncontrolled endpoints as a risk-based decision (e.g., if your organization has a bring-your-own-device policy). Access to sensitive data from uncontrolled systems should be avoided wherever practical, in the absence of compensating controls such as the use of a virtual desktop solution or a secure enterprise browser.

Foundation #3: Resource Ownership Tools and Processes

Successful zero trust implementation requires a reliable system for cataloging the enterprise resources being accessed, and understanding who owns them. In this context, the "who" may not be a single individual but may instead be represented by a flexible grouping mechanism such as a ticket queue. Properly managed ticket queues have owners (who can change seamlessly over time), natural workflows, escalations, priority definitions, and other mechanics that help keep resource information accurate and can flexibly adapt to reorganization or reassignment as ownership of a given resource evolves. If your organization doesn't have extensive rigor around ticketing, you can use alternate mechanisms, such as email distribution lists. However, it is important to keep the maxim "When everybody owns it, nobody owns it" in mind when employing one-to-many mechanisms.

Your source of truth around ownership needs to provide, or be closely integrated with, workflows that facilitate access requests, associated approval decisions, and regular human reviews by responsible parties (i.e., "baselining"). Although some types of access can be inferred from attributes, job roles, and group memberships, ad hoc requests often outnumber rule-based access grants by a wide margin. These workflows should support an individual (or a proxy) requesting access to a given resource, which is then routed for approval, memorialized with descriptive data about ***why*** the access was needed and approved, and regularly revisited to verify that the need still exists. In time, this source of truth will contain the bulk of the information needed within the organization to answer the question, "Who can access what?" which will be used for both authorization policies and audit/compliance.

In addition to a technical repository, your organization should agree on an appropriate governance model for this kind of critical data that provides answers to questions about who can access what: Resource owners? A central team? A combination of the two? The answers don't need to be uniform across the entire organization, but your governance model should be clear and uncomplicated.

Foundation #4: Data Classification

Identifying, protecting, and managing access to your organization's core asset—your data—is an important step on the path to zero trust. However, not all data is created equal. You need visibility into the data you're collecting and storing in order to determine the right levels of data importance and sensitivity. Investing in data classification can help you divide information into predefined groups that share a common risk, and identify the corresponding security controls required to secure each group.

Access to data based on classification will help you prioritize incremental efforts to implement zero trust capabilities. Once zero trust improvements have begun, data classification can also help limit the potential exposure of data to a limited set of users and make security events that require further investigation more straightforward to manage. Encryption of data at rest and in transit adds another layer of security to classified when it's being stored or is required by a user.

Although data classification is relatively simple to apply technically, it's important to set the right expectations and approach. Focus on iterative efforts geared toward constant progress, rather than waiting for anything like perfection. Full data classification can be an expensive and cumbersome activity for organizations that have been storing data for a long time (e.g., since before digitization). As you begin to apply the zero trust model and data classification to your organization's environment, you may decide to simplify the task by setting a time limit (such as two or three years), before which all otherwise-unclassified data is categorized at the least sensitive but nonpublic level. That makes the job simpler and more realistic (without a major impact on risk) because important data types, such as personally identifiable information (PII) or sensitive intellectual property, may already be classified.

Foundation #5: An Established Security Data Lake or Unified Logging

Zero trust architectures and technologies provide additional trust signals that result in more valuable data in security logs. However, this additional data needs to be centralized and standardized to realize its full benefits. Normalizing security telemetry across various security products and services is a key step toward the converged operation of previously siloed security capabilities. Instead of dealing with a variety of proprietary formats, the unified storage and formatting of data simplifies findings enrichment and incident response activities almost immediately and can quickly evolve into a powerful source of insight and continued progress in reducing access privileges.

The Open Cybersecurity Schema Framework (OCSF)[2] is an open standard designed specifically for this purpose. It provides a common language for the kind of security telemetry typically used in threat detection and investigation and has the broad support of well-established security technology providers. Licensed under the Apache License

[2] "Understanding the Open Cybersecurity Schema Framework," Github, May 2023. https://github.com/ocsf/ocsf-docs/blob/main/Understanding%20OCSF.pdf

2.0, OCSF is agnostic in storage format and data collection and can help you minimize the amount of extract, transform, and load (ETL) processing required during ingestion.

Before you can start sending all your organization's security and adjacent telemetry data to a common repository, that repository needs to be properly established. Start by picking a standard storage pattern for the data (preferably based on OCSF or a similar framework) and a raw storage repository—such as Amazon Security Lake, which natively supports OCSF—that can scale to meet current and future capacity and analytical performance needs, based on projected growth.

Be deliberate about your storage hierarchy pattern, and store data consistently. If one tool stores data in a region/host/date hierarchy, but another chooses date/region/host, the queries necessary to join these data sets may be unnecessarily difficult. Finally—although it's important for this core capability to exist—you don't need to wait for all the log sources across your organization to be fully integrated. Instead, these sources can and should be enumerated, prioritized, and integrated opportunistically, with care taken to demonstrate overall system intelligence improvement with each integration.

Foundation #6: Incident Response (IR) Testing

Once you've achieved a reasonable level of zero trust maturity, you can expect to prevent more security events and increase your threat detection capabilities due to an increase in the quantity and quality of security-related signals coming into your security tooling. However, an effective and enhanced IR process that takes advantage of these new data sources is important so you can identify and remediate even minor security events quickly (see Figure 2). This will allow you to disrupt the sequence of events that can escalate an initial incursion into a more high-impact incident.

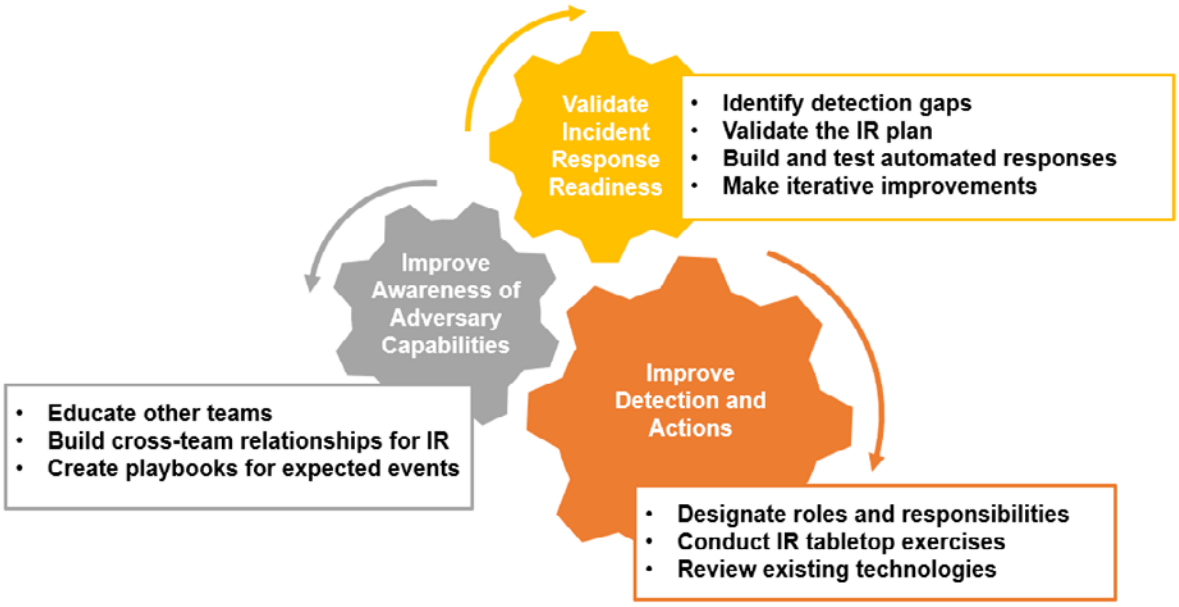


Figure 2. Strengthening Incident Response Readiness

Regardless of the IR framework or methodology your organization aligns with, you should test your IR plan regularly. Tabletop exercises, simulations, and red teaming provide opportunities to practice IR in realistic settings, uncover tooling and capability gaps, and build the experience and confidence of incident responders.

Fallacy #1: You Can't Start Without a Perfect Inventory of Systems, Identities, and Data

When it comes to making progress on zero trust, a ***perfect*** source of truth about your environment may be ideal but is not realistic. Accurate inventories have eluded traditional on-premises environments for decades. Configuration management databases (CMDBs) typically have poor data hygiene. Additionally, discovery tools are often cumbersome to deploy, and organizations struggle to use them to comprehensively capture assets due to existing network controls and segmentation. So, although you need to establish a source of truth, you can divide the effort into a scope that makes a "good enough" inventory quickly achievable, so you don't make the all-too-common mistake of allowing "perfect" to be the enemy of "good." Organizations that are all-in on the cloud, or have heavily migrated to it, may not find achieving inventory accuracy as daunting. Cloud environments significantly ease the process via descriptive application programming interfaces (APIs) and inventory services that allow you to instantly query the running state of your environment via the control plane in a way that's more accurate and up to date.

Fallacy #2: You Can Buy a Product and Quickly "Check the Box" on Zero Trust

Zero trust is a security model, not a product. Although you almost certainly will consume products and services from one or more vendors, you shouldn't lull yourself into thinking the journey to zero trust is as simple as buying and deploying a product that claims to solve your problems. Losing sight of this will, at best, lead to additional expense that doesn't fundamentally change your security model and isn't tied to business outcomes. At worst, approaching zero trust in this way can distract you from your true objectives and provide a false sense of being "done" when, in reality, little to no security improvement has been made.

Fallacy #3: You Need a Clear End-State Vision from the Start

Developing a general North Star vision is important, but don't expect a perfectly clear view of your journey's end before it begins. Careful evaluation of what works for your organization—and what doesn't— along the way precedes the ability to definitively outline your end state. Adjustments will undoubtedly be needed as you make progress and gain insight. Take a flexible approach to initial architectural diagrams and technical standards that depict what "good" looks like and be ready to adapt them as your efforts solidify and you become better-informed. Setting your focus on immediate needs and considering how you can make incremental security improvements that allow for value recognition, real-world experience building, and continuous progress toward an authentic zero trust future will keep your efforts practical, and help you avoid getting hung up on hypotheticals.

Fallacy #4: You Will No Longer Need a Traditional Network Perimeter

You should think of zero trust as largely additive to existing security controls. Network controls are well-understood, broadly deployed, and generally help demarcate an organization's enterprise resources. Network location is also among the important pieces of context that can be evaluated during authorization decisions in a zero trust architecture. And although these decisions must be evaluated and enforced from the edge to deep within the core, traditional network perimeters are some of the first and most logical enforcement points your organization can choose to enhance to take advantage of zero trust access control, because they already exist at various points throughout the network. Recognizing the idea that traditional network controls aren't relevant as a fallacy can help you avoid unnecessary power struggles that may arise when one function or department feels they may be obviated by, rather than integral to, zero trust efforts.

Common Use Cases

There are a number of common use cases that can benefit from the enhanced security provided by zero trust. It's important to work backward from the specific use cases that apply to your organization to determine the optimal zero trust patterns, tools, and strategies that can help you achieve meaningful security advancements. Approaching each use case with an eye to the big picture facilitates progress.

Use Case #1: Human-to-Application

Many organizations start with the human-to-application use case. It's commonly referred to as Zero Trust Network Access (ZTNA) and is often confused with zero trust in its entirety. Although preceding sections of this chapter have largely related to this primary use case, many aspects of the foundations and fallacies apply equally to the additional use cases described below.

In the human-to-application use case, zero trust principles are used to allow employees to access the internal applications they need to do their jobs from anywhere, without relying on a virtual private network (VPN). Although this use case is most often focused on workforce mobility and productivity, it can help your organization realize additional benefits, such as a relatively effortless transition from dated application-level identity protocols, such as Kerberos, to modern identity standards, such as OIDC.

Use Case #2: Service-to-Service

The service-to-service—or machine-to-machine—use case helps you consider pathways within and between workloads, and minimize those that are unnecessary (particularly those that lead to data). Although the human-to-application use case controls how a given actor reaches an application, this use case often controls the resulting flows within an application or between microservices that are composed into an application.

It can be useful to separate efforts related to custom-built services from those focused on services consumed through your cloud provider—at least until you can determine whether they will be implemented with the same or disparate technology and associated controls.

Use Case #3: Internet of Things and Operational Technology

This increasingly common use case supports organizations that are pursuing the interconnection of devices, machines, facilities, infrastructure, and processes outside the traditional network perimeter as part of digital transformation. Internet of things (IoT) and operational technology (OT), also known as Industrial internet of things (IIOT), devices often transmit telemetry and predictive maintenance information directly to the cloud, requiring the application of security controls that extend beyond the traditional perimeter approach to protect workloads.

Use Case #4: Operator-to-Infrastructure

Many organizations are interested in moving beyond development and operations (DevOps) to a fully automated IT environment that requires no hands-on operations work distinct from software development and automated pipelines for testing and promoting code to production (NoOps). However, although NoOps can help you achieve a faster deployment process, it is a journey in and of itself. Regular or break-glass style operator access—which often involves privileged levels of access to operating systems (OSs), database engines, or container infrastructure—needs to be supported along the way, and likely forever to some limited extent. This makes the enhanced access controls afforded by zero trust an imperative. This use case is best approached separately from end-user access, due to divergent tools and access patterns. For example, a user accessing a system through a web application has different security implications compared to direct access to an interactive shell through a protocol such as Secure Shell (SSH).

Use Case #5: Human-to-Data

Organizations of all sizes are using data to enhance customer experience and build new revenue streams with artificial intelligence (AI), machine learning (ML), and advanced analytics. Many of these advancements are driven by data scientists whose work requires access to large amounts of raw data, much of it highly sensitive. Today's binary approach to access runs counter to the zero trust model. Thinking of the difficulty involved in "keeping humans away from the data" helps highlight the need for more granular and flexible preventive and detective controls in this area.

Use Case #6: Authorization Inside Custom Applications

Zero trust involves making access control decisions on individual data elements, artifacts, and other small resources that number in the millions or billions. Although patterns vary, these small resources—think single rows or even cells in a database—are often conceptually modeled at a lower level within custom application business logic

that is more granular than the cloud services or data repositories that store them. For example, a single Parquet file containing records in JavaScript Object Notation (JSON) format in an object storage service might contain thousands or even millions of records, each requiring unique permissions. Most organizations will begin approaching zero trust at higher levels that involve coarser authorization decisions. However, it's important to keep the most granular use cases in mind and verify that your organization's zero trust tooling is capable of further development to cover more granular access controls in the future.

A Key Consideration

Early in your zero trust journey, you'll likely come to a fork in the road as you consider a question that's basic to your overall strategy: Do you want to achieve consistency of ***outcomes*** or consistency of ***implementation***?

A consistency of outcomes strategy views zero trust as a model and a set of ideals that should be implemented with all of the features available in each major compute environment used. Organizations taking this approach are willing to accept some level of heterogeneity in tooling, templating, and reporting to achieve desired security outcomes. These outcomes include things like development and operational efficiencies, integrations, and inherent capabilities or other benefits that would have to be sacrificed or duplicated when a consistency of implementation approach is used.

A consistency of implementation strategy prioritizes standardization and the efficiency it provides the entire organization, over an optimal quality of implementation for each narrower domain. This typically requires ignoring native or default capabilities in favor of solutions that attempt to address the overall requirements of the organization. This approach has some advantages. However, it can lead not only to less tailored and optimized results in a given domain, but also to the duplication of features that can leave some teams confused about the tooling choices, as they are unaware of the broader context and the expected value of organization-wide standardization.

Trade-offs are familiar to most organizations and technology leaders. One example: Complex, heterogeneous environments (such as those running on both Windows and Linux) can either be managed by distinct teams with distinct skills, tools, and modes of work, or those environments can be managed by a uniform abstraction that operates under the premise that "patching is patching," regardless of the OS. Neither point of view is incorrect, but this decision should not be made lightly, as it may not be easy to reverse down the line. When choosing your approach, be careful to avoid common estimation errors. Examples include undervaluing the inherent capabilities provided by cloud environments, overvaluing the flexibility and abstraction provided by a consistency of implementation approach, and underestimating the time and skills necessary to define, build, and maintain zero trust for more than one environment.

Getting Started

Organizations can quickly become overwhelmed by the scope of their zero trust journey. Working to establish the foundations described above, while avoiding mistakes that can result from common fallacies will support your efforts as you make small, well-defined steps toward zero trust. Several best practices can help you chart a path to success:

- **Articulate goals—**Clearly define why you're moving toward zero trust and communicate the goals your organization aims to achieve. This will be more

valuable than describing a technical architecture meant to represent a future state. List key stakeholders (e.g., business users, developers, C-level leadership, board of directors, and security administrators) in your organization, and write a concise summary for each one that articulates why they should care about your zero trust efforts and how those efforts will directly benefit them. Be prepared to consistently deliver, reinforce, and refine these messages as your journey to zero trust progresses.

- **Work on use cases—**Although there are numerous use cases—as detailed above—most organizations should start on "the big two" use cases: human-to-application and service-to-service. These use cases are typically the easiest to separate into a manageable amount of work, they naturally fit back-to-back, and they're straightforward in terms of visibly measuring value and progress. They also tend to involve different groups within the organization, allowing progress to be made in parallel.

  Human-to-application (or ZTNA) is typically expressed as something like, "allowing workforce users to access internal applications from any coffee shop in the world, no VPN required." This use case forces the organization toward the recognition that strongly authenticating a human, evaluating the health and posture of their device, and continuously assessing security state as part of each access request are now the most critical parts of an authorization decision. It is important to focus on this use case early because it directly touches and improves the experience of everyone in the organization who will use it to get their work done every day. One major benefit of starting with the human-to-application use case is that the business leaders who are prioritizing and funding the effort will have a very real and tangible appreciation for the transformation, since they too are users of the new capabilities.

  The service-to-service use case (or machine-to-machine) involves tackling the relative lack of east-west network controls and visibility that often plagues traditional networks and their associated perimeter-based security models. By being deliberate about which components you expect to talk to which other components and how, your organization can disrupt the lateral movement that's often a key part of a security event, while also making the detection and remediation of any network intrusion, however minor, much simpler. By doing so, you can realize a very real and measurable risk reduction.

  The service-to-service use case will also clarify the decision between consistency of outcomes and consistency of implementation, given the stark difference between traditional on-premises networks and API-driven cloud connectivity patterns and the fact that service identities are generally a "solved problem" in the cloud, while root of trust and secrets management and distribution challenges are still meaningful obstacles on premises. Organizations that are willing to move toward the consistency of outcomes approach will likely find that the service-to-service patterns available in the cloud make it possible to completely rethink

traditional implementation patterns and reduce the surface area of compute services, while dramatically simplifying the experience for developers, network engineers, auditors, and security professionals alike.

- **Develop living reference architectures—**Develop an initial, dynamic architecture depicting what "good" looks like for each use case. This will allow you to begin building, yet be ready to adapt as your efforts progress. These reference architectures should be thought of as living artifacts that will continue to evolve. Beyond acknowledging that things will change, this will encourage teams to think about templatizing the architectures for consumption over time.
- **Scope and build authenticity—**Focus your attention within use cases on making progress and gaining momentum. Start with a reasonably sized group of applications, where the business value of the data or the greatly increased convenience for users—or both—is worth the effort required to implement zero trust. By initially focusing on a small and meaningful set, you can refine the necessary technical and operational processes in a flexible and iterative way, while building the authenticity and experience necessary to expand efforts to an increasing percentage of your organization's IT environment. The department leading your zero trust initiative may wish to move one of their own applications or application groupings first to give others confidence that the team has already walked in the footsteps they're asking the rest of the organization to follow.
- **Consider retrofitting versus modernization—**Consider the relative effort and value of retrofitting zero trust into a particular application for a particular use case as-is-where-is versus building zero trust into the application as part of a broader modernization or cloud migration initiative. Although you should be careful about intertwining efforts such as zero trust, application modernization, and cloud migrations if they're already underway or planned, there may be an opportunity to implement zero trust with little to no additional effort.
- **Fuel the adoption with champions—**Think explicitly about rollout, adoption, and value creation as you start your journey. This is not a "build it and they will come" endeavor. Fortunately, there are natural incentives that will drive the rollout. Zero trust makes life easier for end users, so they will become your biggest advocates for getting applications onboarded. It makes life easier for developers by offloading security concerns that previously had to be dealt with in their application logic (or perhaps weren't being dealt with at all), and often providing a "free upgrade" to modern application identity. It produces real outcomes for security teams by increasing levels of assurance for application access and ultimately providing a pathway to shrink an abundance of network connectivity and surface area out of dynamic environments such as office buildings. When it's a win-win for everyone involved, the rollout will typically progress quickly, without the need for large-scale campaigns or program management of a forced "security mandate."

Driving zero trust adoption will take time and effort, as you begin to experience the implementation and its benefits. The team championing zero trust within your organization should be deliberate about partnering with the other stakeholders necessary to complete the initial waves previously described. However, once started, the steady growth of adoption should build momentum on its own, as users demand an improved experience across more enterprise assets, and engineering teams recognize the operational benefits of the implementation.

Measuring Progress

As with any strategic initiative, measuring progress, return on investment, and solution efficacy are key to quantifying the positive impact, maintaining executive buy-in, and justifying budget allocation and investment. However, the impact of zero trust often amounts to measuring what *didn't happen*—or what would *otherwise have happened*—if protections were not in place. Although it's impossible to measure these outcomes with perfect accuracy, you can present metrics that reasonably approximate these impacts. When combined with anecdotes and day-to-day hands-on experiences, these metrics can present a sufficient view of impact and progress.

A basic accounting of rollout progress provides a good starting point. Example metrics might include:

- The number of workforce users properly equipped to access zero trust-ready workloads and those that have the necessary MFA and/or managed devices
- The number of zero trust-enabled workloads, with breakouts for critical or highly sensitive workloads
- The number of security systems sending telemetry to the security data lake or other unified logging sink

For each metric, when the total number is known or reasonably approximated, each scalar value should also be expressed as a percentage, even if the denominator changes over time.

Next, you can strive to account for bad outcomes that were either prevented or minimized by additional zero trust controls. Metrics of this nature will typically require some level of additional labeling, computation, or analysis. Examples include:

- The number of security events prevented by zero trust controls that would not otherwise have been prevented (e.g., denies based on zero trust-specific context)
- The mean time to detect (MTTD) security events—for events that aren't prevented (zero trust should lower MTTD)
- The number of detected security events that were remediated before reaching sensitive data or systems (by lowering MTTD, we also should reduce—with the goal of zero—the number of significant security events)

- Rate of false positives within detected security events (by using a cross-cutting set of telemetry to make security detections, the false-positive rate should decline over time from the pre-zero trust baseline as the system learns)

If your organization has a calculated or industry-approximated per-occurrence dollar figure you are comfortable with, these metrics also can be expressed in terms of "estimated savings," with appropriate caveats. Any such calculation should attempt to account for both direct costs (e.g., external incident response avoided) and indirect costs (e.g., brand reputation or privacy-related fines).

Conclusion

The changing workforce landscape, shifting regulatory requirements, and a need for more precise and least-privileged access controls have led to zero trust becoming a pragmatic choice for IT security strategies. But the journey to zero trust is an iterative process, and it's different for every organization. By considering your own environment, establishing the right foundations, and avoiding common fallacies along the way, you can move beyond traditional security approaches and make continuous progress toward achieving strong levels of security for systems and data.

Chapter 2

Cloud Security: Shared Fate, Identity, Secure Data, and the Coming AI

Written by Dave Shackleford and Anton Chuvakin

Google

Introduction

It's common today for most organizations to have both platform-as-a-service (PaaS) and infrastructure-as-a-service (IaaS) capabilities in one or more cloud environments. Many organizations are increasing the scope of public cloud deployments steadily and have been for some time. At the same time, large cloud service providers have expanded their catalog of tools and advanced cloud infrastructure and services, making it easier for a wide variety of IT and business teams to take advantage of cloud scale and capabilities. As the types of available cloud services grow and organizations begin to deploy large PaaS and IaaS environments that employ numerous interconnected services, the range of cloud security controls needed and potential threat surface also expand. To keep up with the array of different cloud services in use, security teams will need to learn and use more advanced controls and develop more dynamic and continuous processes for evaluating security conditions in their cloud environments.

Fortunately, the landscape of cloud security capabilities and controls is getting better all the time. Cloud service providers are constantly improving their capabilities—both internally and what they offer to customers. At the same time, organizations are becoming savvier about cloud security design and operations. As we progressively deploy more into the cloud, everyone learns more about security best practices, common threats to the cloud, and defenses we can implement at all layers. It's truly a case where "a rising tide lifts all boats."

Cloud as a Secure Business Enabler

As the use of cloud computing has grown, so has the concept of the "shared responsibility model" for data protection and cybersecurity in general. While not a new concept (we've shared security responsibilities with most outsourcing arrangements for many years), the nature of shared security responsibilities has changed with the advent of cloud. Most major cloud providers make it clear as to how they define shared responsibility in the cloud, but organizations need to adapt their risk management strategies to accommodate this concept and relate it to their business and IT realities.

For example, securing the applications they wrote and defining data classification and protection controls are the responsibility of the customer. This may or may not progress down through the cloud computing stack, describing application and operating system controls and network capabilities. The underlying host infrastructure that includes hypervisors, storage components, redundancy, and scalability tools, and more will always be maintained by the provider. However, many OS, application, and networking controls may be abstracted by the cloud fabric that the provider supports. Those controls facilitate new models of risk-taking that might have been wholly unthinkable before the advent of PaaS and IaaS environments.

Another major trend that has occurred in the past several years is the gradual move to the shared fate model[1] of shared responsibility in the cloud. Leading providers are now hosting IoT platforms, payment processing for global financial organizations, and healthcare patient data processing and application integration. Organizations are increasingly dependent on cloud infrastructure, and cloud providers potentially have more exposure due to the proliferation of critical assets running in their environments and the continued evolution of cloud-specific threats.

To that end, there's a definitive need to ensure all parties are clearly informed as to where responsibility lies. As a result, cloud providers are more transparent than ever about what they're doing to defend their environments and protect workloads and data. Furthermore, there is more pressure for cloud providers to go deeper into what used to be mostly client areas of responsibility. For example, secure landing zones,[2] various guardrails, and frameworks indicate provider interest in helping the clients with their elements.

To address the types of threats we face in the cloud, cloud service providers have increasingly offered a growing array of capable security controls tenants can employ for prevention, detection, and response. For many organizations, the challenge may be where to start, and one sound concept is the idea of "trust minimization" across asset and service boundaries. For years, organizations have struggled with the concept of "least privilege" and reduction of access to only that needed for business reasons. In the cloud, leading service providers have done a good deal of this already to make rapid provisioning not only possible, but also secure by default.

One area where this is truer than ever is the realm of identity and access management (IAM), which can encompass authentication, authorization, role-based access control, privilege management, federation and single sign-on, and much more. A notoriously complex area of IT operations and policy definition on premises, most cloud infrastructure has readily acknowledged IAM as the lynchpin of cloud service and asset interaction, along with more focused security perimeters in defining trust boundaries. Having a single set of converged services that comprise IAM within a cloud infrastructure can help centralize security operations, as well as ongoing maintenance and monitoring.

One of the most significant cloud-driven shifts that has occurred in identity management is the advent of "machine identities" versus traditional "human identities." Machine identities, sometimes referred to as "non-people identities," are digital identities associated with computing resources that have access rights and control over other identities, compute services, and compute resources in a public cloud environment.

---

[1] "Demystifying 'shared Fate' - A New Approach To Understand Cybersecurity,"Forbes, Phil Venables, Anton Chuvakin, 19 April 2022. https://www.forbes.com/sites/googlecloud/2022/04/19/demystifying-shared-fate-a-new-approach-to-understand-cybersecurity/?sh=efa5995d6df2

[2] https://cloud.google.com/security-command-center/docs/concepts-secured-landing-zone-overview

These can often be broken down into four distinct categories:

- **Compute resources**—In the cloud, any compute resources, such as compute instances, cloud functions, and containers, can represent machine identities.
- **DevOps and engineering**—These include shared testing accounts, service accounts, and other technical accounts used for programmatic actions and deployments, which are often associated with elements of the DevOps pipeline like build tools, QA and testing platforms, and others.
- **Automation**—Deployment roles and account definitions, particularly for infrastructure as code (IaC) template deployments, are common in more automated cloud scenarios.
- **Cloud services identities**—Many distinct public cloud applications require identities that allow them to interact with other services and resources in the cloud environment.

Human identities, on the other hand, represent traditional interactive users and groups with defined sets of privileges for performing a variety of actions. These may be members of your own organization or external users with whom you collaborate, including those who interact with Google Cloud Platform (GCP)[3] resources via a range of different cloud interfaces. For security teams, it's important to know the use cases and context for any types of identities defined in the cloud. Human identities are usually reserved for administrators and hands-on engineers (and potentially some end users who need to interact with specific services), while machine identities are in place to facilitate cloud service and resource interactions and deployments.

The first element of any identity strategy in Google Cloud is the identity and access management (IAM) service.[4] IAM users are associated with credentials for making API calls to interact with cloud services. They only exist within the cloud environment itself. New IAM users have no permissions (an implicit "deny all" policy). This is a good thing, as permissions must be explicitly granted. This also can help with the common problem of over-allocating privileges to users and groups in the environment. IAM users can represent any asset/resource—an IAM user is a simple identity with associated permissions. This means that IAM users can be enabled for application access to Google resources too, not just as actual interactive user accounts. Once service-oriented users are created, they should be placed in defined groups, if warranted. Highly granular permissions models can be assigned easily through role definition, and the Google Cloud Policy Analyzer can easily assess any defined roles, principals, and groups to assist with privilege minimization or alert an organization when permissions are excessive based on usage analysis.

For larger enterprises, centralizing identity services across multi-account environments will be important. Within the Google Cloud, the centralized identity service for managing users, groups, policies, and role assignments across numerous accounts is known as

[3] https://cloud.google.com/gcp?hl=en

[4] https://cloud.google.com/iam

the Organization Policy Service. With this service, you can create policies that restrict and control how IAM is applied across a set of accounts and service implementations. The Organization Policy Service can actually control the entire account, group, and role life cycle with regard to policy application, and can do so for accounts that need to interact or have some relationship. A basic example of how this could be practical is in governing business unit (BU) account use (as users have totally different requirements, but still need some central control or billing) as well as governing and controlling DevOps and other team accounts (for the same reasons).

The Organization Policy Service[5] is the lynchpin of a multi-account blast radius limitation strategy in GCP. Creating a centralized policy model with clear constraints can allow security administrators to create different and "least privilege" policies for the appropriate accounts and assign them and/or revoke them easily.

Data Security in and for the Cloud

In the Cloud Security Alliance Top Threats to Cloud[6] research, organizations ranked data breaches and data exfiltration from cloud storage as major concerns for cloud deployments—no different from the major concerns to on-premise assets. Naturally, this also means that as part of the shared responsibility model, we have to enable controls in the cloud to protect data from exposure and attack. The good news is that we have more mature data security controls and products/services than ever before (more on this shortly). When storing sensitive personal information in the cloud, it's also imperative to choose a provider who can facilitate compliance to privacy regulations,[7] and has a global presence in the various regions needed to support these important regulatory requirements. Over time, it's likely that more and more region-specific privacy laws and requirements will come about too. That will necessitate choosing cloud provider partners that can keep pace with these changing controls and reporting needs.

There are many factors a mature organization needs to consider to adequately protect data today, and that applies for cloud deployments. This ranges from implementation of various controls to governance and process adaptation within cloud engineering and operations teams. A number of data security concepts change in a progressive cloud model. Some of the following are the most important to consider as you build and plan your cloud architecture and operations strategy:

- **Cloud provider SLAs and data availability/resiliency guarantees are now a part of your shared responsibility strategy.** Many SLAs for cloud storage uptime are at 99.5% and above, and service credits may be contractually guaranteed when these aren't met. This is a prime example of shifting some of the traditional responsibility of service uptime and integrity to the cloud provider. Being able to

[5] https://cloud.google.com/resource-manager/docs/organization-policy/overview

[6] "Top Threats to Cloud Computing," Cloud Security Alliance, https://cloudsecurityalliance.org/artifacts/top-threats-to-cloud-computing-pandemic-eleven-japanese-translation/

[7] https://cloud.google.com/privacy

share the risk by transferring some (not all) responsibility for data availability and resiliency to the provider could free some operational capacity to implement and maintain additional data security controls.

- **Secure transport of data is critical across certain data paths.** While secure transport of data[8] has always been important, creating a hybrid cloud architecture requires transport of data across the internet, an untrusted network. Fortunately, between dedicated connections, such as Dedicated Interconnect and industry-standard site-to-site encryption with IPSec, secure transfer of data is easy to accomplish in a hybrid cloud. Using third-party encryption gateways or network gateways also can help facilitate secure data transfer in a larger deployment.
- **Use of cloud native data security controls is likely a requirement.** There are plenty of data security options available in the cloud, both from providers and third parties. However, at least some of the cloud native controls are likely needed to facilitate implementation of encryption easily. Other cloud native services related to data security may be more affordable and easier to implement in the cloud, such as certificate management,[9] key management,[10] and secrets management.[11]
- **Emphasis on "Bring Your Own Key" and better encryption oversight will be paramount.** Having industry-leading encryption storage available through HSMs[12] (and even "Hold Your Own Key" such as Google Cloud EKM[13] for the right use cases) may facilitate better audit controls for keys and key access, as well as key lifecycle management. Given the increasing use of encryption as a core data security control in the cloud, flexibility in key generation, storage, and lifecycle management are "need to have" requirements for more organizations today.
- **A need for technology that works internally and in the cloud in some cases.** As a hybrid enterprise, you'll already have some data security controls in place in your internal environment. For a variety of reasons, you may need/desire to continue using products and services from third-party providers. Fortunately, an increasing number of providers have products and services readily available in the Google Cloud Marketplace.[14]

For most organizations, enabling full-disk volume encryption for workloads in PaaS and IaaS implementations is an easy and relatively low-cost option. Although not all these encryption types will truly support master boot record (MBR) encryption or granular recovery options, they really aren't intended for this anyway, as these options are usually for mobile devices that could be lost. Instead, volume encryption protects any

---

[8] https://cloud.google.com/blog/products/networking/tls-1-3-is-now-on-by-default-for-google-cloud-services

[9] https://cloud.google.com/certificate-authority-service

[10] https://cloud.google.com/security-key-management

[11] https://cloud.google.com/secret-manager

[12] https://cloud.google.com/kms/docs/hsm

[13] https://cloud.google.com/kms/docs/ekm

[14] https://cloud.google.com/marketplace

snapshots or replicas/backups taken automatically. In addition a key management and integration is usually vastly simplified within the native cloud provider environment.

Protecting data in motion is important for the cloud in two primary places. The first location is between the on-premises environment and cloud provider, where sensitive data may be passing constantly, as in the case of a hybrid cloud, or intermittently for other cloud deployments. The second location is internally within the cloud infrastructure, which then relies on point-to-point tunnels between workloads, data encryption, or both.

Managing, storing, and controlling encryption keys is a critical factor when using encryption in the cloud. Google Cloud Key Management[15] includes a managed HSM service[16] within GCP. Keys can be created in a region or imported from in-house key generation solutions. Numerous Google Cloud services are integrated with Cloud KMS, including workload services, storage services, and many more. In fact, all major storage types within GCP now support various forms of encryption, all of which can be integrated directly with Cloud KMS. Cloud KMS also includes an in-depth audit trail where all API requests and actions related to encryption configuration and key access are securely logged.

Cloud Security with Machine Learning and AI

A much more common use case for many organizations today is large-scale data processing. Many define big data as having three fundamental characteristics: volume on the order of terabytes to zettabytes, a variety of different data structure types (structured, semi-structured, and unstructured), and velocity (rapid capture, discovery, long-term retention, and analysis). On premises, there are numerous locations where data traditionally is housed, ranging from databases to file stores to Storage Area Networks (SANs). When organizations look to consolidate into a big data architecture, most data likely will be stored in an unstructured format. Structured data is often stored in a table format (or similar) and is usually referenced and maintained by relational databases or their modern equivalents. Unstructured data is data that doesn't fit well into a table or lacks a definite data model. A final concept often coupled with big data is analytics, which essentially means looking for patterns in data sets.

What does "big data" mean to security? In a nutshell, security analytics has evolved to make use of machine learning (ML) algorithms with newer artificial intelligence (AI) models focused on analyzing larger and more diverse data sets for a vast array of business initiatives. For security teams, the focus might be on correlating data and looking for patterns that can help them detect malicious activity and potentially predict or plan for more security events and scenarios more effectively. Examples of ML/AI use cases that make sense for using security analytics include the following:

[15] https://cloud.google.com/security-key-management

[16] https://cloud.google.com/kms/docs/hsm

- **Threat intelligence analysis—**Threat intelligence data provides perspective on things like attacker sources, indicators of compromise, behavioral trends related to cloud account use, and attacks against various types of cloud services. Threat intelligence feeds can be aggregated, analyzed at scale using ML engines in the cloud, and processed for likelihood/predictability models. With attacks such as account hijacking and ransomware infections escalating, more rapid analysis of data and predictive intelligence could prove invaluable to security teams.
- **Security event management—**Log data and other events are being produced in enormous quantities, and security teams need to recognize specific indicators quickly, see patterns of events occurring, and spot events happening in the cloud environments. ML and AI could easily augment massive event data processing technology to build more intelligence detection and alerting tactics. Google Chronicle[17] is an excellent example of a massive scale event management engine that leverages AI and ML capabilities.
- **Fraud detection—**For financial services firms and insurers, fraud detection requires an enormous number of inputs and data types and many intensive types of processing. Text mining, database searches, social network analysis, and anomaly detection are coupled with predictive models at scale, and cloud AI and ML engines could likely help with this enormously. This could be extended to things like fraudulent use of cloud services, for example, a Google Gmail-based phishing attack from a hijacked account or payment fraud with ReCAPTCHA.[18]
- **Employee workforce risk reduction—**AI and ML models can be used to process and analyze data related to workforce activities in high-risk environments like manufacturing plants where accidents can prove dangerous or even fatal. AI algorithms could evaluate behavioral patterns noted before accidents occur and perform predictive scenarios to better improve safety procedures and prevent incidents.
- **Data classification and monitoring—**Based on known content types and patterns, AI-based cloud analysis engines can process all data uploaded and created in the cloud environment to classify and tag it based on predefined policies, then monitor for access. Google Cloud Data Loss Prevention[19] is an example of a service that uses AI methods for this purpose in Cloud Storage, Datastores, and BigQuery.

How can security teams leverage or consider implementing "big data" security solutions? There are several things they should undertake, in general. First, security teams need to develop a data repository that allows for unstructured data storage and rapid import of numerous and disparate data sources. One of the keys to leveraging big data" in a security team is taking advantage of many different data types, such as application logs and trends, virtualization and cloud platform logs, workflow and orchestration events, endpoint system events, and more. In a traditional SIEM

[17] https://chronicle.security/

[18] https://cloud.google.com/recaptcha-enterprise/docs/fraud-prevention

[19] https://cloud.google.com/dlp

implementation, most event data comes from firewall and network device logs, system logs, IDS/IPS events, vulnerability scans, and perhaps some applications.

The "big data" mentality often focuses on a broad scope collection of data into one large repository for analysis. That requires tools that can handle the collection and sorting of data into structured and unstructured formats. Next, this data needs to be analyzed, normalized, and followed by larger-scale correlation and reporting. Google's new Google Cloud Security AI Workbench[20] offers some large-scale insights into intelligence on vulnerabilities, malware, threat indicators, and behavioral threat actor profiles that could assist many enterprises in more rapidly and accurately gauging the risk landscape.

For large-scale data processing, the cloud makes a lot of sense. Many organizations are interested in generating and analyzing huge volumes of data to produce statistics, trends, and event behaviors at scale. Doing so in the cloud is usually more affordable and simpler than building a data analytics capability on premises. In addition to the security-oriented use cases described here, there are a vast number of business-specific cases organizations will likely pursue, and security teams will need to focus on the security of the data involved. In the cloud, native technologies are readily available (or even automatically applied like encryption[21]) to ensure data integrity, confidentiality, and ready availability as well.

Even with the numerous benefits of AI and ML, many in the security community are concerned about data privacy and security of these tools. One reason for this is that the data organizations upload into cloud service environments may require data protection controls such as encryption, transport security, tokenization, and obfuscation. Although most traditional data storage services in major cloud providers offer some or all of these, it's critical to ensure AI technologies and services can leverage existing encryption key management and usage models and controls that organizations may have deployed, so the data is not at risk of exposure. Aside from services in use, the geographic location of sensitive data used in ML and AI operations also is a major regulatory and compliance focus. Google AI services all support enterprise-grade capabilities such as data isolation, data protection, sovereignty, and compliance support.

As the use of cloud-based AI and ML services becomes more commonplace, risk management teams will undoubtedly continue to benefit from the rapid analytics processing of large data sets, removing many limitations of more manual risk management and risk analysis processes of the past.

[20] https://cloud.google.com/blog/products/identity-security/rsa-google-cloud-security-ai-workbench-generative-ai

[21] https://cloud.google.com/docs/security/encryption/default-encryption

Conclusion: Looking Ahead

As the types of available cloud services grow and organizations continue to deploy large PaaS and IaaS environments that employ numerous interconnected services, the range of cloud security controls needed and surface to protect also gets larger. To keep up with the array of different cloud services in use, security teams will need to learn and use more advanced controls and develop more dynamic and continuous processes for evaluating security conditions in their environments.

In 2023 and beyond, we see a variety of trends that will be likely to continue to grow including:

- **Major emphasis on data protection and privacy—**Especially for massive-scale data analytics and processing capabilities that exist across numerous accounts and regions
- **Continued focus on identity and access management—**Primarily for centralized monitoring and control of identities and privileged identity control and oversight
- **Continued work on configuring all cloud components and applications securely—** Typically done using a shared fate model
- **Continuous analysis of trust and privileges—**Within the cloud, aligning and focusing assets and workloads/applications based on a principle of least privilege and access minimization
- **Significant growth in ML and AI—**The security of data within AI and ML services—both for business use cases and security analytics—and environments will prove critical in defending against attacks that could pollute ML models. Cloud environments are ideally suited to help with this in all respects.

In all, these types of security controls and services are simply a natural evolution that reflects the nature of PaaS and IaaS software-defined cloud platforms and infrastructure. Security operations in large, distributed cloud environments will need to adapt to accommodate more dynamic deployments and changes, new services and workloads, and a significantly greater reliance on automation. In the next year and beyond, it's likely all these trends will grow and mature significantly.

Chapter 3

Navigating Cloud Security Challenges: Principles and Strategies for Cyber Defense

Written by Dr. Andre Alfred and Ryan Nicholson
Contributors: Jim Cheng, Jeremy Geib, Jacob Harlin, Joel Montano , Jakub Skoniecki

Microsoft

Introduction

Microsoft's cloud systems run at what we call "hyperscale," operating hundreds of datacenters, with thousands of products and millions of customers. Our security solutions generate more than 65 trillion signals per day, which we distill down using ML, AI, and other techniques to generate actionable signals. We leverage those signals to take preventive and protective actions across our infrastructure. There are so many complexities and emerging threats to defend against, whether they are nation-state actors or employees making mistakes that expose sensitive information.

Principle #1: There Is Always More to Learn

Threats evolve constantly, so it is important to have a governance function for cybersecurity that ensures that learning in the organization is happening when there is a security incident. Security must have a problem management function integrated with your engineering organization's processes to continuously improve the state of the environment. Treat critical findings from planned, internal security controls testing such as pentest findings, major code bugs, and red team activity as you would an incident from an external adversary. Creating a learning loop—one that includes a post-mortem process to prioritize and integrate resolving findings into your engineering backlogs as any other feature or bug—helps keep you on track.

Principle #2: Always Assume Breach

The "assume breach" principle is that an application never inherits trust between architectural layers, with the ability to monitor unexpected behavior in between those layers. This principle is not specific to cybersecurity, but a universal safety practice (reference) in any complex engineering system. Even if one tier faces compromise, the intent of this principle is to contain the blast radius and not allow actors to move laterally to more resources. Isolation of resources is the key architectural concept.

For example, leaked or phished credentials, a common type of security breach, cause isolated identities in your applications and services that are scoped down to only authenticate to specific resources and authorized only to their specific use case with the least privileged operations possible for the application to work. Although we always recommend two-factor authentication and phish-resistant credentials, there are also insider threat scenarios where even the appropriate person or machine accessing information must only access their authorized operations and data. We also still must think about protecting against secrets used for machine-to-machine authentication between application roles.

Principle #3: Each Architectural Layer Has Its Own Unique Challenges

Although we can have unique principles, each layer of infrastructure has its own unique engineering challenges to produce an effective defense-in-depth strategy.

Operational technology (OT) devices, such as controllers for power management systems in computer racks and cooling system controllers, are low powered and not sophisticated. They tend to require a passive monitoring strategy and will not have the compute and memory horsepower to run any type of real-time agents or integrate into an authentication system such as TACACS+. Even scanning the management interfaces of OT devices with tools such as Qualys or Nessus may crash them. Vendors that maintain these devices typically need special tools and access to your network either in person or via remote connectivity to provide maintenance and upgrades. OT equipment may not be upgradable at all if there is a firmware level issue that requires complete replacement. For example, replacing a cooling system is complicated and requires a significant capital investment. Network isolation for these environments provides a layer of protection from remote or local compromise as well as lateral movement to corporate networks. Treat this environment with the same rigor you would your corporate and production environments, but tailor the process to compensate for its unique challenges.

Principle #4: Validate Your Security Controls Continuously

Organizations continuously run safety drills, tests, and checks for vital systems and processes. Even a common fire drill for a company is an example of continuous validation and training for emergency response. Cybersecurity is no different. In today's world, the mental model of a cybersecurity breach should not prepare for "if" it happens but for "when" it happens through regular verification processes. Red team operations and tabletop exercises that include all parties from executives to analysts should be part of a regular operation that evaluates the processes. Also, do not limit this principle to your own organization. Apply it to your critical supply chain partners too. It is common to share sensitive data with a subcontractor. Do not let the first time you discuss joint cybersecurity incident response with those partners be during the incident itself.

These principles describe a continuous feedback system designed to help minimize your cybersecurity risk. Next, we will deep dive into how Microsoft Azure recommends you think about security governance processes for internal systems and suppliers and how you secure your code pipelines. We'll also look at how we think about detection depth in alerting, and how we continuously integrate our learning back into our systems and processes.

Security Governance in the Cloud

Governance for complex technological environments is difficult, but important to manage closely. Even if your organizational hierarchy is siloed, your cybersecurity posture is only as good as the weakest link or the lowest bar. Consistency is key and it's essential to review security requirements with engineering leaders on a regular basis. It is important for governance teams to realize that the security requests going into

engineering backlogs are actionable and negotiated. Posture management and problem management are part of the same continuous feedback learning loop, and the repair items discovered from that process get integrated back into the engineering strategies for our cloud products. We suggest mandatory attendance for weekly reviews at the executive levels of the company for security posture and gaps.

Diving into the Security Development Lifecycle

Development is the primary function of a technology company. But how do developers create software securely? How does compliance know that everyone is doing what they are supposed to? How can security ensure that engineers are integrating with and implementing security controls? Security development lifecycle (SDL) is the answer to all these questions. A good SDL not only focuses development work and streamlines compliance, but also eases security management and instills security into projects where and when it is most effective and efficient.

The aspects of SDL are well described in the Microsoft Security Development Lifecycle.[1] They are:

1. Provide training
2. Define security requirements
3. Define metrics and compliance reporting
4. Perform threat modeling
5. Establish design requirements
6. Define and use cryptography standards
7. Manage security risk or third-party components
8. Use approved tools
9. Perform static analysis security testing
10. Perform dynamic analysis security testing
11. Perform penetration testing
12. Establish a standard incident response process

Instead of rehashing these elements, let us dive into how to make those practices a reality and create technical backstops to provide additional assurances.

Identify and Explain Security Requirements

It can be complicated to clearly identify security requirements. You must align your particular use case and context with relevant technologies and your enterprise threat model, and bear in mind any relevant compliance regimens. A critical aspect of these requirements is that they must have clearly defined and implementable controls.

Controls can break down broadly into having three attributes: positive, procedural, and native (positive and procedural are terms loosely borrowed from Department of Defense airspace control and deconfliction).

- **Procedural** controls are what they sound like. They require adherence to a set of procedures. In security, these are referred to as "best-effort," as they will fail or go unsatisfied in inconsistent and occasionally unpredictable ways.

[1] https://www.microsoft.com/en-us/securityengineering/sdl/

- **Positive** controls are based on having real-time awareness and influence over what is happening.
- **Native** controls are what they sound like—controls that are already there. With the native control model, no adoption, onboarding, or button clicks by teams are needed, they are already in place.

These attributes have differing degrees of complexity, require different levels of effort to implement, and carry a variety of tradeoffs. These attributes are not mutually exclusive. The goal of making security operate with scale, speed, and surety is to enable controls to have all three attributes as much as possible. Ideally, all SDL controls include detailed guidance on requirements and implementation (procedure), the security governance function has deep insights into performance and non-conformance (positive), and the controls also are present and enabled by default (native). When at least two of these attributes are in place, it is much simpler to manage your security posture and satisfy compliance.

Creating SDL controls that contain these attributes can seem daunting, particularly with the wide variety of tech stacks, use cases, and contexts. So, let's explore some examples to see how it all works.

As specific controls can vary broadly based on context, we'll focus on some of the most common SDL controls: standards for design and development (NIST, PCI, etc.), system baselines or "golden" images (NIST, CIS), and threat modeling or risk assessments.

Coding Standards Are a Must

Develop (pun!) standards of practice for designing and developing your services. Such practices should include what and where cryptography should be used, specific allowed and disallowed libraries, and secure coding standards, such as avoiding integer overflows.[2] Once these standards are created, they must be communicated and implemented across development teams. Training sessions for secure development training are useful for this on their own, however, they work best when backed up with technical enforcement.

CodeQL[3] serves this function effectively. With CodeQL you can write several coding standards into rule sets to enforce your standards and consistently implement secure development.[4] CodeQL also handily fulfills the element of "static analysis" in the SDL above.

CodeQL (or a similar SAST tool) helps enforce secure development practices, detect variance, and identify otherwise unknown security issues in the codebase. You also can implement it at PR time to derive results and provide them to users in a manner more integrated into the development workflow. Of course, all this benefit also has some attendant costs and investments. To maximize benefit, build a robust infrastructure and workflow around rule tuning and tracking issue identification to ensure high signal/low noise, drive high fix rates, and keep suppressions to a minimum.

---

[2] CWE-190: Integer Overflow or Wraparound (4.11), mitre.org, https://cwe.mitre.org/data/definitions/190.html

[3] CodeQL, https://codeql.github.com/

[4] https://github.com/github/codeql/blob/main/go/ql/src/Security/CWE-190/AllocationSizeOverflow.ql

Think Golden Resources, Not Only Golden Images

Once upon a time, enterprises invested in "golden" images, or images that had received some security and configuration scrutiny and modifications, to ensure teams started off as securely as possible. In a more modern sense, this should be extended into the concept of "golden" resources. These resources include compute, deployment, and development infrastructure that have all received security hardening and modifications to ensure teams start as secure as possible. Efforts should be made to ensure that these resources have been secured and follow the principle of "secure by default."

Threat Modeling

Threat modeling is part art and part science. Done well and at the right time(s), it can prevent a number of security issues from ever leaving the whiteboard. To get started, focus on the big four questions (thanks to Shostack + Associates for formalizing these questions in their "Ultimate Beginner's Guide to Threat Modeling"):[5]

- What are we working on?
- What can go wrong?
- What are we going to do?
- Did we do a good job?

Although these questions may seem simple, they are best answered by a dedicated security resource working in collaboration with the engineering team. Engineering teams are fantastic at building new services and features, and a good security threat modeler is expert at identifying ways in which it can break in new and interesting ways.

A good SDL helps streamline this process by guiding teams down paved paths that avoid common pitfalls in design and implementation. For example, good cryptography and authentication controls reduce the likelihood that those aspects would be identified as "broken" or have easily identifiable weaknesses in a threat modeling session. SDL adherence reduces the chances that a team might adopt a security antipattern unknowingly, and that means that a threat modeling effort can focus on more engaging and less transactional security.

Continuous Assurance

To properly implement controls and security, you must maintain awareness of the control activities themselves and the disposition of the assets and resources in scope. Said differently: You can't address what you don't know. One of the critical aspects of SDL requirements and design activities is identifying how control tracking will be implemented. This insight should be dynamic and live as possible. A snapshot of the state tells you only what your status was at the time, not what it is now. With modern continuous integration and continuous delivery (CI/CD) and pipelines, the state of a given system can change rapidly, so timely information is vital.

Similarly, awareness of the population and disposition provides improved efficiency and effectiveness. Identification that a control is not having the desired impact or reach

[5] "The Ultimate Beginner's Guide to Threat Modeling," Shostack + Associates, https://shostack.org/resources/threat-modeling

early on can create a virtuous cycle where issues are identified and mitigated in faster iterations than what would be possible in a more static snapshot model.

Disposition and demographic tracking of your environment is vital. It is one thing to know that N% of a population is satisfying a control, it is another to be aware that the N% is a subset of a much larger X%. As stated earlier: You can't address what you don't know. The goal of a good continuous assurance program is the stance of "no surprises."

SDLs are a foundational element to a strong and resilient security program, but you must also be wary of potential pitfalls with improper implementation. Goodhart's law is very real in security. Put simply it states: When a measure becomes a target, it ceases to be a good measure. In practice, this means that if you focus your insights on how quickly a security ticket is closed, you'll wind up with rapidly closed tickets, but not necessarily faster issue remediation. If a threat modeling team's performance is measured on issues found, you will get a lot of issues but not necessarily improved value. A critical aspect of a continuous assurance effort is the avoidance of the dreaded watermelon—green on the outside and filled with red on the inside.

SDL Systems Not SDL Processes

Functionally, a process is serialized steps accomplished to produce an output. A system is a composition of organized things that maintain an interrelationship with each other. A system can also refer to underlying technical infrastructure. SDL can feel like a trudge to dev teams. Often this is related to controls and requirements not integrating with the work or workflow, an excess of manual processes, excessive time between steps, "surprise" security asks, obtuse processes, and vague or inconsistent requirements.

One way to approach a solution for this is to view SDL as a system, not just a process or checklist. For any SDL control, think about how it integrates with the people asked to perform it on a regular basis, evaluate the quality of the outcomes and the expectations, and be critical of it all. Ensure there is not wasted downtime between steps, and make sure real security value is being generated (avoid "check the box" mentality). Also ensure implementation requirements are clear, applicable, and feasible. Above all, never accept "because we've always done it that way" as an acceptable justification.

SDL must cut a third path through the tensions between security and product deliverables. As new controls enter review and adoption, or new compliance frameworks enter the product scope, think about the end user and how they work. Focus on embedding controls as natively as possible and give developers as much to say "yes" to as possible.

The more an SDL comes with baked-in solutions that are already in place in the infrastructure or are readily integrated into new work, the better. This approach makes it simpler for engineering teams to focus on delivering their work and not on figuring out how to roll their own public key infrastructure (PKI). Similarly, establishing SDL controls as axiomatic makes security and compliance more transparent and streamlined for all parties involved.

All that being said, if you can cheat and leverage a secure-by-default development infrastructure with ease of use, ease of insight/governance, and well documented SDL and compliance integrations, it always makes life easier.

Protect Your Pipelines

Pipelines are foundational to CI/CD and scale with velocity. At their most basic level, pipelines get your code where it needs to go, in the form it needs to be. Modern CI/CD systems rely on pipelines to complete a wide variety of tasks needed to pull in dependencies, compile code, perform scanning tasks, execute scripts, generate artifacts, and more. Fundamentally, they are a series of actions or steps performed to accomplish both the "integration" and "deployment" aspects of CI/CD. The "continuous" moniker comes from the pervasive use of automation for these pipeline actions. The pipeline process brings a slew of benefits but also carries some new risks to control for. Much like flying, it's a great benefit when the engineering and practice is done following strong guidance, requirements, and validation.

Cloud infrastructure relies on pipelines and, as such, they have become an increasingly common attack vector. From the attacker's perspective, compromise of a pipeline may provide access to target environments, secrets/keys, or even privileged service account credentials. In an unsecured pipeline compromise, few things are safe.

Some common pipeline attacks and vulnerabilities include:[6]

- **Supply chain attacks—**This is where an attacker seeks to compromise the target's supply chain and dependencies. This is an umbrella term for several attacks and methods and, honestly, it's worthy of several volumes to cover adequately. In a supply chain attack, a dependency could be compromised via a malicious contributor, an attacker could perform a dependency confusion attack, or traffic could be misdirected via typo squatting or DNS poisoning. A good starting point for where risks are can be found at the CNCF page on supply chain security.[7] Regardless of the modality used, the end goal is to get malicious code into the target environment.
- **Pipeline poisoning—**Pipeline poisoning is when an attacker seeks to compromise the build system (pipelines) by injecting malicious code into the configuration or actions of the pipeline. A successful compromise can result in an attacker's code being executed at elevated privileges and can result in further pivots and build and deployment compromise.
- **Artifact compromise—**In artifact compromise, an attacker seeks to compromise a resulting build artifact, often compiled code. This vector has several possibilities including potential loss of intellectual property (IP) if the attacker is able to exfiltrate the artifact or poisoning of the artifact itself, which could lead to further compromise of the environment and create additional attacker footholds or pivots. This attack can be used to move laterally in environments with strong separation between projects if artifacts can be cross published for consumption by other teams.

---

[6] "Top 10 CI/CD Security Risks," Github, https://github.com/cider-security-research/top-10-cicd-security-risks

[7] "Types of Supply Chain Compromise," Github, https://github.com/cncf/tag-security/blob/main/supply-chain-security/compromises/compromise-definitions.md

- **Build network compromise—**In this type of pipeline vulnerability, an attacker seeks to compromise or poison the build network, an often-neglected aspect of pipeline and CI/CD security. To accomplish its varied tasks and actions of building and deploying code, some form of compute is needed. Generally, a build network is made up of VMs or containers, which are often abstracted by the name "compute agents" or "build agents." If they are compromised, an attacker can influence builds and use the resulting network foothold to move further into the environment. Compromise can either come directly, in the event when a build network is publicly exposed, or through a side channel such as a tainted dependency pulled into a vulnerable compute agent.

Pipeline Protections 101

Pipelines are incredibly useful, and even better when established with security in mind. Although it may seem daunting to look at even a fraction of the types of attacks that target pipelines, there are several ways to mitigate those risks to help protect yourself and your enterprise from attackers.

Protect Your Ingestion

Generally speaking, code enters your pipeline in one of two ways: from your SCM users and from your dependencies. The first step in any pipeline security program is to build security around these two aspects. For users, we recommend the use and enforcement of strong, phishing-resistant credentials for access. One of the leading standards in this space is FIDO2, which provides strongly bound identity with origin binding and use of asymmetric cryptography. This means you'll have an extra layer of defense when one of your users gets phished.

Use of FIDO2 security tokens also creates the opportunity to make use of commit signing. Commit signing has been around for a while and has had some fairly clunky implementations. Thankfully, GitHub has simplified a lot of that headache. With commit signing, a user registers their key with the repository service and can use their private key to sign their commits in a developer-friendly workflow. With proper branch protections in place, you can prevent any unsigned or improperly signed code from being merged. This in turn not only gives you strong assurances that the commits have not been manipulated by a third party, but also provides strong assurances on proof of presence. Two security benefits with a single adoption.

Manage Your Pipeline Permissions

Permissions management pre-dates computing and it's something that is not going away any time soon. Although it can be an intensive and heavily manual "toil" task, there are a wide variety of ways and tools to make it easier.

The first and most basic way to make it easier for yourself is to limit who gets permissions to create, edit, or delete pipelines in the first place. This establishes a simpler span of control, reduces the ways attackers can gain access, and makes

management far simpler. If that does not work for your use case, look at just-in-time (JIT) permissions systems. With JIT, you make a request for access to the system, someone other than you approves the request, and you get the needed permissions for a limited time before they expire. JIT can help reduce the risk of permissions creep in users and still maintain a high degree of flexibility for who does what and when. Lastly, look at establishing re-authentication using FIDO2 for privileged actions. An external attacker may compromise credentials and get into your SCM a variety of ways, but if they lack the needed registered hardware token and pin, nefariously manipulating pipelines will be that much harder.

One additional identity element to be mindful of with your pipelines is the security of your service accounts and machine identities. A huge reason to use pipelines is the power of automation, and, depending on the tools you are using, the automation you use in the pipeline may have its own identity you have to consider for security.

What does this mean? You have a whole separate identity not tied to a human that is often granted very wide-ranging permissions. The solution for this can be very straightforward: Don't give machine identities (MI) broad permissions and ensure there are strong controls over how the MI can be accessed by users. MIs (sometimes called service principles or service accounts) should be limited in their scope of duties, functions, and permissions.

Secure Your Supply Chain

Entire books have been written about supply chain security, and rightfully so. It is an extremely broad and deep subject area that is rife with complexities, dependencies (pun!), and participants, and, to top it all off, it's highly dynamic. The summation of most of the literature and zeitgeist is this: Third-party and open source software are fantastically useful and filled with risk, so you need a multidimensional approach to securing them.

A good first step for securing outside code is knowing where it came from and broadly what is inside. These are the concepts of provenance, where the artifact came from (and occasionally who the contributors were), and pedigree, the lineage or history of the artifact. This is where software bill of materials (SBOMs) and signed code come into play. SBOMs are bills of materials or an index of the code, packages, and libraries, and attestations for its name and version. "Signed" is the attribute of the code having been cryptographically signed by a relevant group (e.g. package maintainer, vendor, etc.). Together, this information provides provenance and pedigree assertions.

These provenance assertions are useful for several reasons. A proper validation and catalog of them aids dramatically in security governance, vulnerability management, and blast radius awareness. It aids in protecting against common dependency confusion attacks that leverage DNS poisoning and allows for more quickly tracking and identifying what software is used where. As part of the provenance checks, vulnerability analysis tools are also useful. These provide an additional validation check on the SBOM and increase the transparency of what code and risks you are consuming. In addition to the SBOM, these scans provide another check and validation on what software is present

(though your mileage may vary as scans can be subverted or defeated) as shown in the Github example.[8] To make the use and consumption of third-party code easier and more secure, Microsoft has contributed to the community the Secure Supply Chain Consumption Framework (S2C2F).[9]

These assertions, when taken in total, help reduce risk from common attack vectors and methods. However, these provenance assertions are not necessarily security assurances. The key difference is that they track where the software came from and what packages comprise it, not necessarily that it is safe. As said earlier: This is a complex and dynamic space. Security is a journey and, as a community and industry, we've really only just started down the path to a more secure supply chain.

Inventory Management

There is a reason the NIST Cyber Security Framework starts with "Identify." Just as the adage about "you cannot manage what you cannot measure" is true, so is its cyber equivalent: You cannot protect what you don't know exists.

Inventory is often a topic of tension in security. It is frequently relegated to the realm of "someone else's problem." The fact of the matter is that inventory is fundamentally a security matter. Securing anything, whether it's your home or your cloud, must begin with knowing what is where. If you don't know about it, you can't secure it, and you won't respond well if there is an issue.

There are two axioms for inventory. The first is that it is very easy to lose track of where things are and what those things are. Even technically sophisticated firms can falter. The second is that the best time to build a high-fidelity inventory is before you need it.

Cloud Changes the Game

Modern large-scale applications are frequently comprised of a multitude of microservices, or just smaller macroservices. This architecture enabled a revolution in scalable, durable, and semi-autonomous operations for large or distributed applications. Traditional static approaches for inventory typically do not scale to this model. This trend has coincided with the use of compute in new and interesting ways, such as containers and ephemeral compute (nothing is really serverless). Although these modalities have enabled an impressive wave of auto-scaling and semi-autonomous automation, they can create significant conceptual difficulties for inventory.

With cloud infrastructure, few (if any) things are static. Compute, addresses, even naming conventions can be dynamic, and tracking all the shifts and changes presents its own unique challenges. In some cases, firms take it upon themselves to solve this problem. Netflix's Edda[10] is one example. To ensure security and availability, they built their own service to track where their resources are. It works by polling a multitude of cloud APIs to derive a picture of where all their resources are and their state.

---

[8] "Malicious Compliance: Reflections on Trusting Container Image Scanners," Github, https://github.com/bgeesaman/malicious-compliance#TBD

[9] "Secure Supply Chain Consumption Framework," Github, https://github.com/ossf/s2c2f/blob/main/specification/Secure_Supply_Chain_Consumption_Framework (S2C2F).pdf

[10] https://netflix.github.io/edda/

The ability to dynamically query a dynamic environment has its own difficulties, however, the result of such a system is the improved ability to respond to outages, incidents, and issues; improved focus for security teams; reduced toil for developers; and more efficient resource allocation.

One of the fantastic benefits of a cloud infrastructure is that the state, status, and disposition of everything is always tracked, so the data all exists. The hard work comes in collecting, collating, and comprehending that data to derive a useful and usable inventory system.

Inside Inventory

Inventory should start at the code repository and extend through to the deployed compute resource. Throughout the inventory life cycle, several attributes should be tracked and collated. Code, libraries and packages, artifacts, code owners, resource owners, management groups, pipelines, component micro/services, resource types, resource locations, and more should all be visible, dynamically updated, and available for querying.

Services like Azure Resource Graph go a long way to enable this functionality by providing the ability to query resources and dispositions. However, the ability of a tool to work is dependent on the discipline of the users and quality of the input. "Garbage in, garbage out" as the saying goes. Inventory starts with organizational culture. If teams follow a disciplined culture of tagging and organizing the code, pipelines, resources, and services, the quality of any inventory system will be dramatically improved. Although we have made significant advances in machine learning, most models would still struggle to discern the dozens of services, owners, and functionality of a massive monorepo that lacks metadata and structure.

Maintain Awareness

Inventory is usually conceptually focused on what is supposed to be there. The corollary of "what is there" is a critical component to a robust inventory system. Systems fail, networks lag, data gets corrupted. Good engineering requires accounting for failure, and inventory systems are the same. Regular querying and discovery scans of the environment to validate inventory aids incident response and service durability and provides protection against wallet attacks (DDoS by cloud bill).

Supplier Cybersecurity Governance

Supply chain security has become a critical topic for many organizations with a rapid increase of attacks specifically targeting software, hardware, and services that are developed, hosted, and/or managed by external suppliers.

Threat Landscape

According to a survey conducted by IBM Security X-Force, 62% of organizations were hit by supply chain attacks in 2021.[11] The Target data breach in 2013 was a perfect example of this. The attackers were able to exploit third-party systems (used by an HVAC firm to access Target's network for remote monitoring of HVAC energy consumption and temperatures) to exfiltrate payment information, which impacted more than 41 million customers. The recent MOVEit (a popular file transfer service) breach was another perfect example of threat actors breaching a supplier's environment (through exploiting a SQL injection vulnerability) to steal a large amount of data uploaded by customers, without even coming into the customer's network.

Challenges and Strategies

Ensuring adequate security oversight in your suppliers that allows you to exercise identify, detect, protect, respond, and recover capability is obviously more difficult than performing the same tasks in your own organizations. Legal, operational process, and technical execution should all play a part in the supplier security governance. At the high level, you should start building your supplier cybersecurity management framework by following the best practices from NIST SP800-161 Cybersecurity Supply Chain Risk Management Practices for Systems and Organizations, as well as the NIST Software Supply Chain Security Guidance mandated by the Executive Order 14028. A comprehensive supplier cybersecurity security governance should include the following aspects:

- **Understanding the supply chain—**Organizations must have a comprehensive understanding of their supply chain, including all entities, processes, and dependencies involved. This includes suppliers, contractors, and subcontractors, as well as the flow of components, software, and information.
- **Risk assessment—**A thorough risk assessment process should be implemented to identify potential vulnerabilities and threats within the supply chain. This involves assessing the trustworthiness and security practices of suppliers, evaluating the potential impact of risks, and prioritizing mitigation efforts.
- **Establishing security requirements—**Organizations should clearly define their security requirements and expectations for suppliers. This includes specifying security controls, testing and evaluation criteria, and contractual obligations related to cybersecurity.
- **Supplier evaluation and selection—**When engaging with suppliers, organizations should conduct due diligence to evaluate their cybersecurity practices. This may involve assessing their security posture, certifications, incident response capabilities, and adherence to industry standards and best practices.
- **Communication and information sharing—**Effective communication and information sharing are essential for managing supply chain risks. Organizations

[11] "62% of Surveyed Organizations Hit By Supply Chain Attacks in 2021," Security Intelligence, https://securityintelligence.com/articles/62-of-surveyed-organizations-hit-by-supply-chain-attacks-in-2021/

should establish clear lines of communication with suppliers, regularly exchange information on threats and vulnerabilities, and collaborate on incident response and mitigation efforts.

- **Monitoring and auditing—**Continuous monitoring and auditing of the supply chain are critical to detect and respond to emerging risks. This includes monitoring supplier performance, conducting security assessments, and implementing mechanisms for early warning of potential threats.
- **Incident response and recovery—**Organizations should have robust incident response plans in place to address supply chain disruptions and cybersecurity incidents. This includes clearly defining roles and responsibilities, establishing communication channels, and coordinating with suppliers to minimize the impact of incidents.
- **Training and awareness—**Building a culture of cybersecurity awareness and promoting training and education programs within the organization and across the supply chain are essential. This helps ensure all stakeholders understand their roles and responsibilities in managing supply chain risks.
- **Continuous improvement—**Supply chain risk management is an ongoing process that requires continuous improvement. Organizations should regularly reassess risks, update security requirements, and incorporate lessons learned from incidents and audits into their practices.

Tactical Best Practices

In the cloud world at the tactical level, the focus is obviously on the software/service supply chain, such as cloud platforms and services from external providers, as well as application code and libraries sourced from third parties.

Ensure your enterprise's software development life cycle (SDLC) or process includes a set of security controls to govern the third-party cloud services and the software components (including both proprietary and open source software) where your applications have dependencies. Define gating criteria to prevent vulnerable or malicious components being integrated and deployed into your cloud environment.

The software supply chain security controls should at least include the following aspects:

- Properly manage a SBOM by identifying the upstream dependencies required for the cloud service/resource development, build, integration, and deployment phase.
- Inventory and track the in-house and third-party software components for known vulnerability when there is a fix available in the upstream.
- Assess the vulnerabilities and malware in the software components using static and dynamic application testing for unknown vulnerabilities.
- Ensure the vulnerabilities and malware are mitigated using the appropriate approach. This may include source code local or upstream fix, feature exclusion, and/or applying compensating controls if direct mitigation is not available.

If closed source third-party components are used in your cloud environment, you may only have limited visibility to its security posture. You should consider using additional cloud native and non-native controls such as access control, network isolation, and endpoint security to minimize the impact and reduce the blast radius if there is a malicious activity or vulnerability associated with the component.

If you use cloud platforms such as GitHub to manage your code repository, many of them also come with native security features to allow you to enforce the vulnerability scanning throughout your application build, test, and deployment process.

Supply Chain Cybersecurity Posture Management

A critical lever in delivering a secure Microsoft cloud is securing our extensive network of third-party suppliers that enable Azure to operate at scale. There are suppliers at every level of the cloud stack, from physical devices powering our data centers to support staff working on our services. To maximize our return on security investment (ROSI), a strategic roadmap of activities is required to make impactful progress in the holistic security posture of our supply chain. Our approach involves an assess-triage-mitigate loop that operates on a "trust but verify" framework and enables us to work with suppliers in a synergistic capacity. Microsoft is not here to audit our partners. Rather we collaborate on effective solutions to address any gaps identified by analyzing the various signals in our program.

Addressing Scale

Scale is an issue at both extremes. Organizations such as Microsoft conduct critical business with suppliers at varying stages in their cybersecurity maturity journey. A broad brush-stroke approach to supply chain cybersecurity will miss one or both extremes: Too advanced and your smaller-scale suppliers are missing foundational pieces of their program, too generic and your larger-scale and more mature suppliers lack the incentive to collaborate. Knowing this, focusing on strategically relevant suppliers will enable you to operate a meaningful supply chain cybersecurity program. You classify these suppliers as your "needle movers." With a finite number of resources at your disposal:

- With whom do you partner to have the greatest impact on your supply chain cybersecurity?
- Which suppliers are carrying or have access to critical workloads?
- What threat intelligence signals have the greatest impact on your environment?
- Have there been recent incidents and breaches involving suppliers?
- What are the priorities of the business?

Clarifying Scope

Before we dive into the process, it's important to clarify that this scope is a layer above procurement and compliance. It is non-negotiable that suppliers achieve the minimum bar defined by the organization and take the proper steps to remain in good standing

in order to conduct business. Our scope is meant to take the next step in elevating the cybersecurity maturity bar of the ecosystem and to move beyond the minimum requirements and into concerted efforts to protect our users and customers.

Assess

As we mentioned in the security governance section, your cybersecurity posture is only as good as the weakest link. As an organization, we must do our diligence in securing our supply chain without having direct decision-making capabilities. The first step in doing so is establishing a cybersecurity baseline for suppliers. Through a series of assessments, the organization can establish the current cybersecurity posture of the supply chain, identify trends and gaps across the ecosystem, and use these signals to develop actionable mitigation plans for suppliers.

Part of the initial assessment can take the form of "outside-looking-in" through external attack surface analysis and cooperative threat intelligence. These signals form the external view of the assessment. To complete the full picture, an organization must collaborate with suppliers to conduct self-attested assessments of the supplier's environment. The scale and fidelity of these assessments vary on the strategic plan set in place—your "needle movers" participate in in-depth assessments covering several domains from a framework such as CMMC or NIST. These engagements are an involved process where both parties engage in a series of activities focused on the cybersecurity procedures and policies in place. The broader supply chain can participate in less tightly scoped assessments, offering brief questionnaires that enable the supplier to input, at a high-level, highlights of their cybersecurity program. Humility and empathy are critical when approaching your suppliers. You are not there to audit their environments, but instead to have an open dialogue about the current state of the program and ways to jointly address areas of need. Microsoft approached such conversations by sharing our own assessment findings. We identified areas of excellence and improvement within our first-party environment and started with a dialogue of improvement, collaboration, and learning.

Triage

When the assessments conclude and you have identified a baseline, it is critical to triage and prioritize gaps identified across the supply chain. As many of us know, there will be more findings than what our organizations can realistically address. Again, the "needle movers" mantra will guide decision making throughout this process. Which gaps are a) uniform across several of our suppliers or b) high-impact critical findings needing to be addressed? Your decisions here should be driven by a high ROSI horizontally (across the supply chain) or vertically (high-impact areas).

Mitigate

Once the priority gaps are identified, we must now find a collaborative approach to mitigating these risks. Microsoft is in the privileged position of offering several security tools and solutions that cover many domains of cybersecurity. That said, Microsoft

approaches this part of the process with a vendor-agnostic mindset. We are here to protect our cloud, so we empower our suppliers to do so in the most efficient and cost-effective manner possible. We assembled a playbook of security solutions mapped to the domains in our assessments. They are not catch-all solutions, but once you have a repository of solutions, it is much easier to scale across a broad range of suppliers, each with slightly different needs.

In addition to security products and services, Microsoft engages in security readiness exercises for domains, such as incident response. These exercises can improve cybersecurity awareness and involvement with a supplier's organization but also contribute to the cross-org partnership necessary for this type of program to succeed. We take an active role in these engagements and participate as both a player and host, providing "real-world" scenarios that mirror supply-chain incidents closely.

Learn and Revisit

This program is cyclical. You must regularly reassess priorities and improve upon your mitigation playbooks and strategies. Your "needle movers" may shift based on emerging threat indicators or changing priorities of your business. The important mindset to maintain throughout the maturation of your program is that you are not looking for faults but rather for areas of opportunity. We avoid a "me vs. you" mentality during the assessments and mitigation efforts because the job is not to dictate what needs to be improved but rather identify areas the collective can jointly work on.

Making the Best of Your Detections

The "detect" function in the NIST Cybersecurity Framework focuses on the timely identification of cybersecurity events. That function should be approached layer by layer to ensure comprehensive coverage of the detection capabilities to address each layer's unique challenges and characteristics.

Layer-by-Layer Approach

The layer-by-layer approach usually includes the following aspects:

- **Network layer—**This includes analyzing network packets, detecting anomalies, and identifying known attack signatures.
- **Endpoint layer—**This involves implementing host-based intrusion detection systems (HIDS) or endpoint detection and response (EDR) solutions to detect unauthorized access, malware infections, or suspicious behaviors at the endpoints (including servers, workstations, smartphones, IoT devices, and any devices that are assigned with IPs).
- **Application layer—**This involves application-specific detection mechanisms such as web application firewalls (WAFs) or runtime application self-protection (RASP) solutions to identify vulnerabilities and indication of compromise and monitor application-level activities for anomalies.

- **Data layer—**The data layer focuses on detecting unauthorized access, data breaches, or data exfiltration using tools such as data loss prevention (DLP) solutions to monitor data flows, identify sensitive data, and detect any attempts to access or transfer data in violation of security policies.
- **User layer—**This involves monitoring end user activities and behaviors to detect potential insider threats, compromised accounts, or unauthorized access. User behavior analytics (UBA) solutions can be used to establish baseline behavior patterns and identify deviations that may indicate suspicious or malicious activities.
- **Cloud layer—**The detection capabilities extend to cover public cloud and third-party layers. This layer includes implementing cloud security monitoring tools such as CWPP (Cloud Workload Protection Platform), CSPM (Cloud Security Posture Management), and CNAPP (Cloud-Native Application Protection Platform) to perform regular audits/monitoring of cloud environments. Be mindful when you operate in a multicloud environment. You need to have a unified detection approach (such as log data normalization and aggregation) to align the signals from different clouds into a centralized SIEM platform for correlation and analysis.

It is important to note this layer-by-layer approach is not strictly linear or exclusive. There can be overlaps and interdependencies between layers. For example, threat intelligence feeds and security information and event management (SIEM) systems can provide cross-layer visibility and correlation of events.

The layer-by-layer approach in the "detect" function can be further enhanced by integrating the MITRE ATT&CK® framework, which provides a base of adversary tactics, techniques, and procedures (TTPs) targeting each layer. This can help organizations in their detection efforts across different layers using tailored and optimized setup in tools such as SIEM, XDR, firewall, DLP, and other security solutions.

Logging Strategies and Detection Authoring

Don't forget a robust detection capability requires effective logging strategies and detection authoring.

Effective logging strategies involve enabling adequate logging for security detection while minimizing noise, implementing centralized log storage for correlation and analysis by SIEM and other anomaly analysis tools, and establishing log retention for future post-mortem purposes. By fine-tuning logging settings, organizations can capture relevant security events while filtering out unnecessary noise. Centralized log storage enables comprehensive visibility and analysis of logs, facilitating the detection of patterns and anomalies. Additionally, appropriate log retention ensures valuable data for post-incident reviews and continuous improvement.

Detection authoring across services involves creating and customizing detection rules and mechanisms for various systems and services. It empowers security teams to tailor detection mechanisms to the unique characteristics and vulnerabilities of different components within the environment. This approach ensures that detection rules are

optimized to monitor and identify potential threats specific to each service. It encourages collaboration between teams responsible for different services to eliminate possible blind spots in detection of anomalies of a service. Lastly, continuously updating and refining detection rules also should be part of the detection authoring routines. These approaches strengthen the organization's ability to detect and respond to potential security incidents across their entire IT ecosystem.

Together, effective logging strategies and detection authoring provide a robust foundation for proactive threat detection and response. They enable organizations to leverage valuable insights from logs and continuously refine detection rules to stay ahead of emerging threats.

Handling Miss Detection

Reviewing miss detections in security incidents is vital to improve detection capabilities. By analyzing incidents, identifying gaps, and determining root causes, organizations can develop targeted remediation plans. This process involves a thorough examination of the incident, including events, attack vectors, and impact. The focus is on identifying missed detections and failures in existing mechanisms, such as outdated rules or misconfigurations. Root cause analysis helps uncover underlying factors contributing to the misses. Remediation actions include updating rules, enhancing monitoring tools, and integrating threat intelligence. The process fosters continuous learning and a culture of improvement. Lessons learned are documented and applied to future strategies, strengthening detection capabilities and reducing future misses. This iterative approach helps organizations stay ahead of evolving threats and enhances their overall cybersecurity defenses.

Creating a Learning Loop for Secure Operations

Approaching security with a growth mindset, every potential failure can be reframed as an opportunity for continuous improvement. This is the focus of the last phase of the SDLC in which findings from the response processes are analyzed, categorized, and prioritized for proactive investment. For continuous learning to happen successfully, it is critical to ensure that the focus of the feedback loop is on how the organization can improve and not on assigning blame to an individual for the current failure.

Analysis

Utilizing a process referred to as variant hunting, we seek to rapidly apply information about a new defect across the organization. There are two types of variant hunting:

- Horizontal variant hunting is performed across the organization to answer the question "Is this defect present in any additional products or services?"
- Vertical variant hunting focuses on the affected product or service and seeks to answer the question "Are there similar defects present in the same product or service?"

Variant hunting is often performed with a combination of manual reviews and rapidly prototyped detections. When performed holistically, variant hunting is a key method to reduce the overall defect density of your products and services.

Categorization and Taxonomy

During the post-incident response phase, defects are categorized into the appropriate contributing root causes, antipatterns, and SDL escapes. The classification and categorization of defects into a standard taxonomy allows the organization to develop a heatmap of failures, identifying potential areas of systemic risk that can be prioritized for proactive investments. The organization's taxonomy should be robust enough to cover both the "what" (antipatterns, contributing root causes) as well as the "how" (SDL escapes), painting a full picture of the class of the defect and how it *should* have been prevented. As with all aspects of the process, no categorization system is perfect, and organizations must develop their taxonomy to grow and adapt over time.

Closing the Loop

Organizations must create repair items and track them to completion to prevent the recurrence of defects in the environment. To be considered complete, all the contributing root causes and escapes identified during the classification process should have repair items associated with them that clearly define the acceptance criteria for the repair item, including who is accountable for delivering the repair item and the expected delivery date. Reviewing the repair items in an open forum with individuals representing many different parts of the organization may help to remove bias and myopic perspectives on what is considered an appropriate repair. To drive accountability within the organization, it's critical to establish a system that empowers leadership to view the current state of all repair items. This can be accomplished through whatever means is typical for the organization (e.g., reporting, dashboards, etc.) as long as it is complete, accurate, and timely.

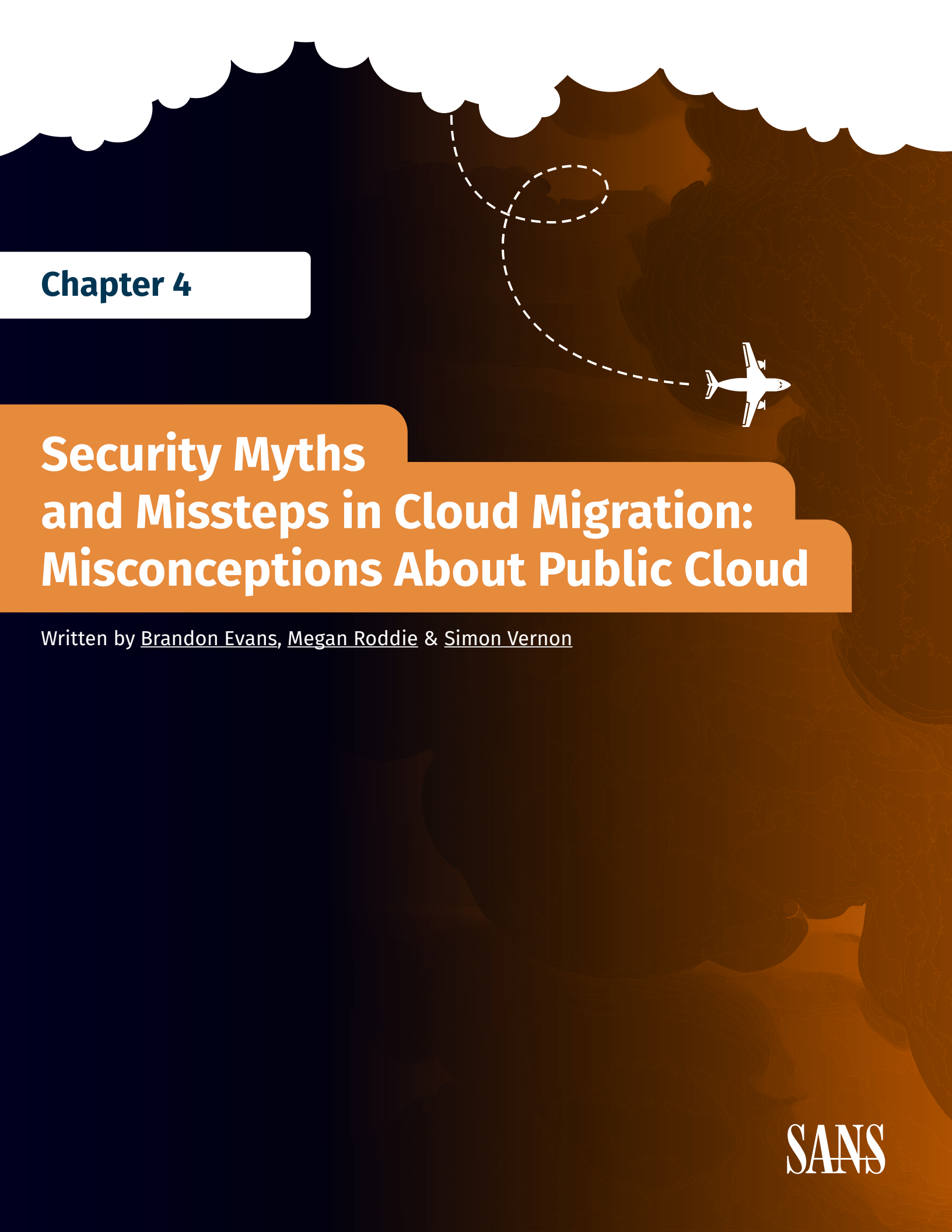
Conclusion

We've discussed a wide range of topics on how to manage security in the cloud. An effective way to govern your security posture is to assume breach between layers and that there will always be something new to learn. It's fundamental to employ problem-management techniques that incorporate your learnings from simulated and real incidents using blame-free post-mortems. Leverage modern cloud native tools to continuously validate your security posture in all your resource categories where possible. Cybersecurity is a rewarding career and one of the world's most important professions. Take it from us that we still learn and improve every single day at Microsoft, and we recommend that you do the same.

Chapter 4

Security Myths and Missteps in Cloud Migration: Misconceptions About Public Cloud

Written by Brandon Evans, Megan Roddie & Simon Vernon



SANS

Introduction

Public cloud computing has revolutionized the IT industry by providing on-demand access to scalable and flexible computing resources, storage, and applications. There are many organizations that still have concerns and misconceptions that prevent them from investing in the cloud. These concerns often arise from comparison to traditional on-premises infrastructure investments.

One of the most common concerns is security, and many organizations worry that their data and applications will not be as secure in the public cloud as they are in their own on-premises data centers. This concern is often based on the misconception that the cloud provider is solely responsible for security. This misconception is assumed to be true and likely weighs heavily on the minds of decision makers when it comes to determining cloud security and cloud adoption.

Another common misconception is that investing in the public cloud will be more expensive than investing in on-premises infrastructure. While it is true that cloud computing can come with a range of costs, such as data transfer, compute, and storage fees, these are not always clearly understood and the costing process for estimation of consumption is complicated.

Compliance and regulatory concerns are also a common barrier to investing in the public cloud. Many organizations worry their data and applications will not comply with industry regulations or that they will lose control over their data. However, cloud providers typically have compliance certifications and attestations for many industries, including healthcare, finance, and government. Cloud providers also offer tools and services to help organizations adapt current security models, then provide ongoing assessment and compliance reporting.

Organizations have several primary security concerns when it comes to adopting public cloud computing. These include:

- **Data breaches—**Organizations are concerned about the potential for data breaches in the public cloud. This concern arises from the perception that their data will be less secure in the cloud than it would be in their own on-premises data centers. In reality, cloud providers often have more resources and expertise to secure their infrastructure than many organizations have in-house based on their capability to crowdsource and identify security trends.
- **Unauthorized access—**Organizations are worried about the possibility of unauthorized access to their data and applications in the cloud. This concern is often based on the misconception that cloud providers are solely responsible for security. However, in the shared responsibility model, both the cloud provider and the customer share responsibility for security. It is essential for organizations to properly configure and secure their applications and data in the cloud.

- **Compliance—**Organizations are concerned about maintaining compliance with industry and government regulations in the public cloud. This is especially true for highly regulated industries such as healthcare, finance, and government. Cloud providers typically have compliance certifications and attestations for many industries, but organizations still need to ensure that their applications and data are properly secured and compliant.
- **Data loss—**Organizations worry about the possibility of losing their data in the public cloud. This can occur due to human error, hardware failure, or other factors. To mitigate this risk, organizations need to have established and tested backup and recovery processes in place.
- **Account hijacking—**Organizations are concerned about the possibility of their cloud provider accounts being hijacked by attackers. This could result in unauthorized access to data and applications, as well as other security breaches. To prevent account hijacking, organizations need to implement strong authentication and access controls, as well as regularly monitor their cloud provider accounts for suspicious activity.
- **Shared responsibility model—**Organizations need to understand the shared responsibility model and take appropriate steps to secure their applications and data in the public cloud. This includes proper configuration, access control, monitoring, and backup and recovery processes.

When it comes to pricing, return on investment (ROI), and financial concerns of adopting public cloud computing, the following should be considered:

- **Cost of migration—**One of the first pricing concerns organizations face when considering public cloud adoption is the cost of migration. Migrating to the public cloud can involve significant upfront costs, including the cost of redesigning applications, refactoring code, and migrating data. Organizations need to carefully assess the cost of migration and develop a comprehensive migration plan that includes cost estimates for each step of the process. There are options available through cloud brokers to acquire the necessary skills and consultants to improve adoption strategy and rein in costs.
- **Ongoing costs and hidden costs—**These costs can include data storage, compute usage, and network bandwidth usage, among others. While the pay-as-you-go pricing model of public cloud computing can be more cost-effective than on-premises infrastructure over time, it can be difficult for organizations to predict these costs and manage their budgets accordingly. It is important for organizations to carefully review their contracts with cloud providers and understand all the potential costs associated with their services.
- **Vendor lock-in—**Another pricing concern is the potential for vendor lock-in. Organizations worry that once they adopt a public cloud provider, they will be locked into using that provider's services and will be unable to switch to another provider without incurring significant costs. This can limit an organization's ability to negotiate pricing or take advantage of more cost-effective services from other providers.

- **Return on investment—**Finally, organizations worry about the ROI of public cloud adoption. Although there are many potential benefits of cloud computing, including increased agility, scalability, and security, organizations need to carefully assess the potential ROI of these benefits against the costs of adoption. They also need to consider the potential ROI of alternative approaches, such as on-premises infrastructure or hybrid cloud solutions.

There is a need to carefully consider the pricing and financial implications of public cloud adoption before making the switch. Enterprises need to assess the costs of migration, ongoing costs, potential for vendor lock-in, hidden costs, and potential ROI to ensure they are making the best financial decision for their organization.

Issues of Cloud Exit Strategies

While public cloud adoption offers numerous benefits, there are scenarios where organizations might consider transitioning back to on-premises data centers. This is often the result of poor planning, short-sighted business strategy, or architectural issues stemming from lack of research and incorrect design decisions.

Backing out of a cloud adoption strategy, even at an early stage, can be a very complex operation. There can be several technical difficulties to face. One of the major challenges is extracting and migrating data from the cloud platform. Depending on the volume and complexity of the data, this process can be time-consuming and resource-intensive. Organizations need to plan and execute a comprehensive data migration strategy that ensures data integrity to minimize downtime during the transition. This may involve using data migration tools, implementing data transformation processes, and ensuring compatibility between the source and target environments.

Applications built specifically for the cloud platform may need to be refactored or modified to ensure compatibility with the new environment. This can be a complex process, requiring adjustments to application architecture, code, and dependencies. The level of effort involved in application refactoring will depend on the extent to which the applications are tightly coupled with the cloud provider's proprietary services and features.

Cloud platforms often have vendor-specific technologies and services that organizations may have become dependent upon. When transitioning away from the cloud, organizations will need to identify alternative solutions or technologies that can provide similar functionality and capabilities. This may require evaluating and selecting new tools, frameworks, or platforms, and adapting or rewriting parts of the applications to integrate with the new technologies. Applications optimized for the cloud platform may not perform optimally in a different environment, so organizations need to assess and optimize their applications to ensure they meet performance expectations and efficiently utilize resources in the new infrastructure. This may involve tuning application configurations, adjusting resource allocation, and conducting thorough testing to validate performance under different conditions.

Cloud platforms typically provide robust network infrastructure and connectivity options, and organizations need to ensure they have suitable network setups in their new environment to maintain performance, security, and availability. This may involve setting up dedicated network connections and load balancers, configuring firewalls, and optimizing network routing. Moving away from global elastic network architecture presents issues when determining on-premises scaling, elasticity, and network address space allocations.

Once you have adopted a cloud strategy and progressed to implementation, backing out is unlikely to be the best idea.

It's important for organizations to thoroughly plan and strategize a potential exit from the cloud platform—one that involves stakeholders from different departments and considers both technical and business implications. The focus should be whether exit is the correct decision and not a reaction to a failure of on-boarding. External experts or consultants also can provide valuable guidance and support throughout the decision-making process. By addressing technical difficulties proactively and systematically, organizations can either successfully decide to exit from public cloud platforms or re-engineer to use the extensive capabilities of public cloud offerings.

Security and Cloud Implementations

When adopting cloud strategies, organizations need to address compliance and regulation concerns to ensure they can meet regulatory requirements and maintain the security of their data. The following are steps that organizations can take to address these concerns:

- **Understand the regulatory landscape—**The first step is to understand the regulatory landscape and the compliance requirements that apply to the organization. This includes industry-specific regulations as well as regional and national regulations. Organizations need to have a clear understanding of the requirements they need to meet and the potential risks associated with noncompliance.
- **Select cloud brokers and consultants with appropriate certifications—**Organizations should choose cloud providers that have appropriate certifications and accreditations for their industry and regulatory requirements. For example, if an organization is in the healthcare industry, they should choose a cloud provider that has achieved HIPAA compliance. Organizations also should ensure their cloud provider is transparent about their compliance and security practices.
- **Implement security and compliance controls—**Organizations should implement security and compliance controls that are appropriate for their regulatory requirements. This includes controls such as access controls, data encryption, and audit logging. They also should regularly monitor their systems and applications to ensure compliance.

- **Develop a cloud governance framework—**Organizations should develop a cloud governance framework that outlines the policies, procedures, and controls for managing their cloud environment. This framework should include guidelines for selecting and using cloud services, managing cloud service providers (CSPs), and monitoring compliance with regulatory requirements.
- **Regularly conduct compliance audits—**Finally, organizations should regularly conduct compliance audits to ensure their cloud environment is meeting regulatory requirements. These audits should be conducted by a qualified third-party auditor and should include a review of the cloud provider's controls as well as the organization's own controls.

By taking these steps, organizations can ensure their cloud strategy is compliant with regulatory requirements and can maintain the security of their data.

There is also the lack of customization concern, as many organizations don't comprehend the vast complexity of cloud services based on millions of use cases generated by other tenants, effectively crowd-generating functionality, which the CSPs then supply. There should be a level of understanding that organizations can leverage various customization options available in the cloud. Cloud providers often offer a wide range of services and configurations that can be tailored to meet specific business requirements. Additionally, organizations also can explore options, such as hybrid cloud deployments, where they can retain certain components on premises while leveraging cloud services for other aspects. This hybrid approach provides more flexibility and customization options, allowing organizations to strike a balance between control and leveraging the benefits of the cloud.

It's important, though, for organizations to engage with cloud providers and consultants/brokers during the planning and design phase to discuss their customization needs and ensure that the chosen cloud solution aligns with their specific requirements. Additionally, working closely with cloud providers, specialist architects, and business analysts to leverage their expertise can help organizations find innovative solutions and address any customization concerns effectively.

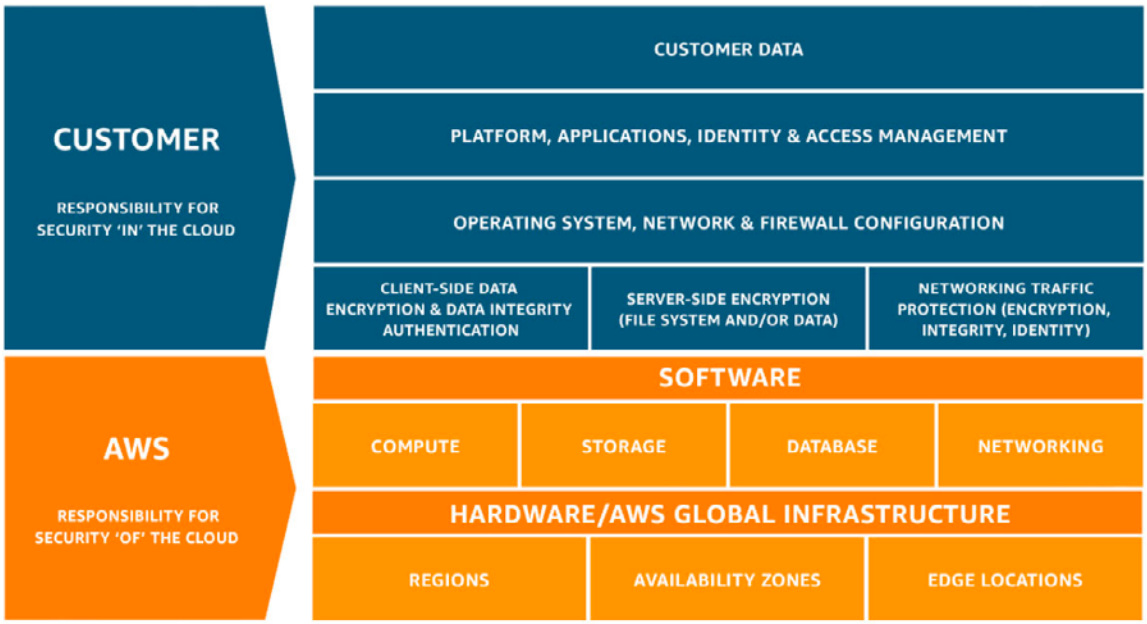
The Shared Responsibility Model

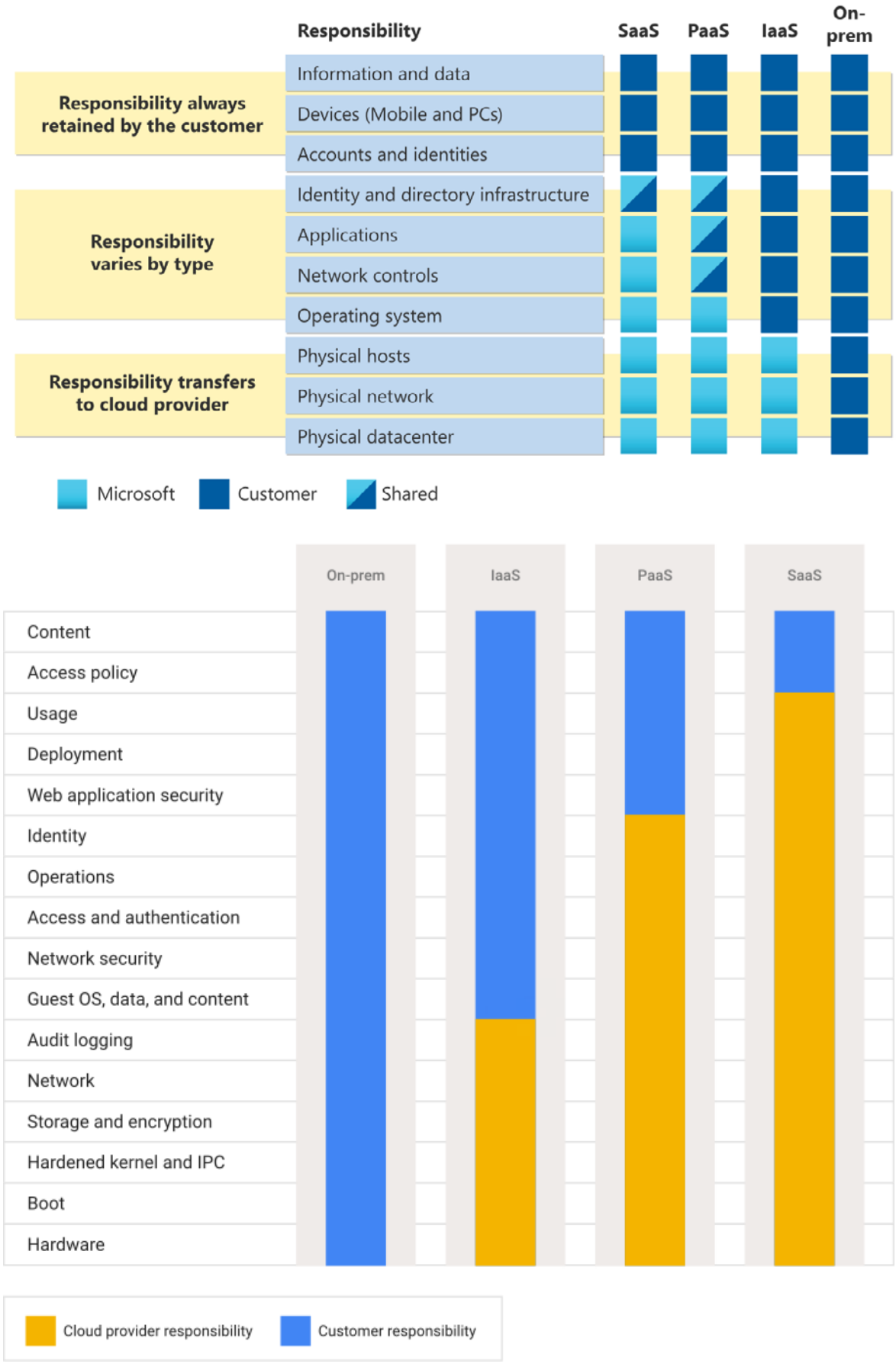
Despite being a fundamental and commonly discussed topic, the shared responsibility model is still largely misunderstood by security professionals. Cloud sceptics might point to a breach that happens to an organization in the cloud and use that as a reason to go on premises. When a cloud-based organization fails a compliance audit, they might argue that the cloud provider misrepresented their certifications. However, those who understand the shared responsibility model would never make these claims. Without understanding this model, cloud customers will continue failing to apply the necessary controls to keep their data, infrastructure, and users safe.

The shared responsibility model is not a buzzword cloud providers use to deflect from their security obligations. Some form of this model must exist for the cloud to function. A cloud provider does not and cannot know their customer's specific security needs and requirements. As a result, the organization must define these requirements and codify them using the appropriate cloud configuration settings. The cloud provider's burden is to ensure these settings are properly enforced as they are represented.

Dividing this responsibility can potentially improve security. The cloud provider specializes in developing general-purpose services, while the average organization would prefer to use these types of services to enable their core competency. So, for example, the teams developing Amazon S3, Azure Storage, and Google Cloud Storage have thought through many abuse cases and implemented many mitigations for storage solutions. It is unlikely that another organization would be able to dedicate the same amount of effort toward designing a service like this, let alone test it at the same scale. This paper will get into more concrete examples of this in the next section, but it is self-evident. Organizations use third-party vendors for the same reason. The built-in cloud services have the added benefit of seamlessly integrating with other services in the same cloud.

Athough many understand this concept in theory, they might not understand exactly which responsibilities are on which side of the model. Each of the Big 3 cloud providers have diagrams attempting to define this demarcation (see Figures 1, 2, and 3).





Figures 1-3. Shared Responsibility Model Illustrations for AWS, Azure, and Google Cloud Respectively

All of these diagrams clearly place the following into the cloud provider's domain:

- **Data centers and hardware—**The cloud provider must construct the building, recover from fires and earthquakes, hire security guards, prevent hardware theft, and more. This is the most obvious of their responsibilities as the cloud customer has no control over these aspects.
- **Physical networking—**This must be secured by the cloud provider for the same reason as above.
- **Cloud service security—**Although the customer is responsible for using cloud services appropriately, the cloud provider must remediate security issues in the service software itself. If a customer discovers a vulnerability in one of these services, they can report it to the cloud provider through their bug bounty program, but they are not typically going to create and ship the code that fixes the issue.

Many other considerations depend on whether the customer is using IaaS, PaaS, or SaaS. Each service type gives the customer more control than the next, but that also comes with increased responsibility.

Simply put, the CSP is responsible for securing the tools you use to build workloads in the cloud and giving you the security configuration options you need to meet your specific security requirements. You are responsible for using these configuration options appropriately and for ensuring that the workloads you deploy are secure.

It should be noted that, although this model is necessary and mostly workable, there are still challenges. Anton Chuvakin of Google Cloud, who also contributed to this e-book, has detailed many of these challenges in this August 2022 article.[1] He echoed that customers often "assume that the CSP does more than it really does." Even if the documentation is clear, if the user does not read it, they will be unaware of their responsibilities. Though this is a common occurrence and a failure on the customer's part, we would argue that the cloud providers should feel somewhat responsible for effectively educating their customers. It is unreasonable to ask the customer to read thousands of pages of documentation. CSPs also should give security guidance through other mediums with which their customers are more likely to engage. They should deliver articles, videos, and even warnings to their customers when they are about to do something that is potentially insecure on their platform. We especially appreciate that they helped meet this responsibility by contributing to this e-book!

[1] https://medium.com/anton-on-security/where-does-shared-responsibility-model-for-security-breaks-in-the-real-world-970f7dad56f4

The Benefits of Public Cloud

Cost Savings

In the first section, we discussed concerns regarding the costs associated with a cloud migration. A lot of these issues arise, however, due to a misunderstanding of pricing models. Pricing in the cloud is much different than what we've experienced in on-premises environments. It's important to understand how you are going to be charged for usage before performing your migration because, if done properly, you can actually reduce costs over your on-premises environment. It is very dependent on how you configure and leverage your environment.

In an on-premises environment, when you purchase hardware, you must estimate the resources you need. Often the smart move is to overestimate, but if you have periods of reduced usage or were incorrect with your estimates, you are then paying for those unused resources. With the cloud, you are only going to pay for the resources you use. For example, for storage, you're charged for the amount of data you are storing, rather than paying extensive costs up front for physical drives that might not be fully taken advantage of. With cloud-hosted infrastructure, scalability, which will be discussed in the next section, allows your systems to only use the resources needed at a given moment, rather than purchasing the maximum hardware requirements even for periods of low usage.

Another consideration that highlights the importance of properly planning a migration is region-based services. Most cloud providers allocate resources by region and there are associated costs with cross-region data transfer. If you strategically consolidate your resources to a limited number of regions, you will be able to transfer data within the region for free in most cases, reducing costs associated with data transfer. As such, it's important to plan out in which region(s) you are going to deploy your resources so you don't incur unnecessary costs by distributing your infrastructure too widely.

In the section on techniques to address some of the concerns with cloud migration, we will have some strategies for managing cloud costs that allow you to take advantage of cost savings rather than risking increased costs compared to an on-premises deployment.

Scalability and Flexibility

A major benefit of the cloud is the scalability and flexibility provided. We mentioned from a cost-savings perspective how a "pay for what you use" model is beneficial, but the scalability and flexibility benefits extend beyond that.

As we said, often when acquiring on-premises hardware, you are estimating what you expect you'll need from a resource perspective. Underestimate, and you're going to encounter resource constraints. Overestimate, and you're paying for unused resources. Even if you can get a good estimate for a given point in time, resource requirements vary in many situations. Consider online retail businesses, for example. Many retailers

experience an influx of traffic to their e-commerce sites around holidays or with promotional offers. As such, when acquiring hardware for their web server, they need to plan for those peak seasons. On most days, however, the traffic is likely not going to reach that level, so they are not able to take advantage of the hardware investment. In a cloud environment, however, autoscaling allows your web server to adapt to the amount of traffic. This goes for any servers or devices you migrate to the cloud: virtual machines, serverless applications, firewalls, and more.

Similarly, storage is scalable and flexible. As a customer, you essentially have unlimited storage space, assuming you're willing to pay the associated costs. No longer are the days of storage space alerts with frustrated attempts to decide what you can get rid of to make space for new data.

Improved Security

Earlier we discussed the shared responsibility model. One of the advantages of moving to the cloud is that this model puts part of the responsibility for the security of your resources and infrastructure on the CSP. Regardless of the cloud model you are leveraging (IaaS, PaaS, or SaaS), at least some of the security controls fall to the provider. For example, even in IaaS where the most responsibility is in the customer's hands, the hardware security is on the provider, removing the need to secure physical devices in your control. In the other models, PaaS and SaaS, even more responsibility is taken out of the hands of the customer with the CSP being responsible for tasks such as operating system patching and network security controls.

Often CSPs would prefer to default to more secure configurations and will look for ways to reduce the risk of customers accidentally creating vulnerable resources. For example, when AWS realized the widespread data loss being caused by public S3 buckets, they made a change so that newly created buckets would be private by default instead of public. Now, if you set a bucket to public, there will be numerous warnings indicating the risk of such an action.

Even for the responsibilities that do fall to the customer, cloud environments improve the ease of securing your resources. All three big CSPs provide various capabilities and controls focused on security. Their identity and access management (IAM) solutions provide controls for ensuring that access is limited to those who need it on a granular level. Encryption capabilities allow for customers to secure their data without the requirement of deploying their own key management system. Network security groups provide an additional layer of control to IAM to ensure that access is restricted where required.

On top of the aforementioned solutions provided alongside many of the resources and services, security tools themselves are more readily and easily deployable in the cloud. Outside of cost considerations, it is much more trivial to deploy a firewall in the cloud than to acquire and deploy a firewall appliance in an on-premises network.

The last benefit worth highlighting here is logging capabilities. There is a lot of logging provided by the cloud providers by default. Customers also are provided with a variety

of methods of reviewing and/or exporting these logs. Logging is key to both detect and respond to security incidents, so the ability to have logs available when the need arises, and the ability to centralize the logs, is a significant benefit.

Increased Productivity and Efficiency

To finalize this section, we are going to look at the productivity and efficiency benefits resulting from a cloud migration. First, the process for deploying applications, services, and resources is much more efficient. For an on-premises deployment, spinning up a new server requires procuring hardware and coordination with multiple teams, bottlenecking the ability to perform actual work related to the business need for it. By removing these requirements, working with new applications and services is exponentially quicker.

CSPs offer numerous services that allow you to migrate capabilities to new technologies designed for improved productivity and efficiency. For example, serverless functions provide a method for automation to speed up workflows. Another example is infrastructure-as-code, such as Azure's ARM templates. With this capability, code can be used to create redeployable environments, avoiding the need to reapply numerous configurations for the same use case.

In recent years, the increased number of remote employees and distributed offices created a challenge related to the efficiency of sharing data and workloads across regions. However, the cloud assists in addressing this issue. Since cloud resources are accessible globally, it is much simpler to connect employees to your infrastructure than with an on-premises deployment. Additionally, by leveraging region-based resources to strategically place resources close to the users requiring access or through features like AWS's Transfer Acceleration, you can increase the efficiency even further.

Best Practices for Securing Data in the Cloud

As organizations make a move toward the cloud, threat actors also are adapting their operations to target cloud environments. It's important that before moving your data to the cloud, the correct controls are in place to secure it. As mentioned in the discussion on improved security in the cloud, the cloud vendors provide customers with a variety of controls that can be used to secure their data.

Although the tools available are numerous, there are some key considerations you can consider that will provide a big impact on protecting cloud environments.

Strategies for Managing Cloud Costs

We've discussed that the cloud can either result in increased costs or cost savings depending on how you manage your costs. There are various ways to attempt to reduce how much you'll spend, and a lot of that is understanding how and for what you will be charged.

Amazon, Google, and Microsoft all provide pricing calculators to help you estimate costs associated with the various software and services you are interested in. These tools can be used when you've planned out your requirements for a cloud architecture and will give rough estimates for the price range you're looking at monthly.

There are many costs for interlinked services and resources you might not be expecting without clearly reviewing the associated costs. For example, spinning up an EC2 VM does not just involve the hourly cost of running the machine. You'll be paying for the hard drive, the snapshots, and potentially additional services as required, such as elastic IPs and encryption capabilities. It is important to understand every aspect of a cloud deployment and how you are going to use it to know what the cloud vendor may charge you for.

There are a few tricks that can help reduce costs when managing a cloud deployment. First, consider the regions in which you are deploying resources and the ingress/egress costs associated with the configuration. Although some cloud services are global, many are region-based, and data transfer costs are typically based on whether the data leaves the region or not. By grouping resources that need to transfer data between each other into the same region, egress transfer costs are reduced.

Another way to reduce costs is to regularly review your resources to ensure no unnecessary storage charges are being incurred. For example, snapshot storage can add up quickly if snapshots are being automatically, regularly generated for virtual machines. If you don't review the snapshots, you might end up with some from five years ago that are no longer relevant to your operations yet are being charged to your budget.

Conclusion

In this discussion, we explored various concerns that organizations have when it comes to adopting public cloud strategies. We began by highlighting the misconceptions and concerns that prevent organizations from investing in the public cloud, such as security, pricing, compliance, and lack of control.

Public cloud platforms have a global presence, offering data centers in multiple regions worldwide. This enables organizations to reach customers and users across the globe with reduced latency and improved performance. Additionally, cloud-based services can be accessed remotely from any location, facilitating collaboration and enabling remote work capabilities. Cloud providers typically have robust infrastructure and high availability guarantees, often offering service level agreements (SLAs) to ensure uptime and reliability. They have redundant systems, backup mechanisms, and disaster recovery solutions in place to minimize downtime and data loss. This level of reliability is challenging for many organizations to achieve with their on-premises infrastructure.

Although security concerns were mentioned earlier, it's important to note that public cloud providers invest heavily in security measures and compliance certifications. They have dedicated security teams, advanced threat detection systems, and industry-

standard security practices in place. Cloud providers also ensure compliance with various regulations, making it easier for organizations to meet their compliance obligations. Plus, they provide assurance and regular updates relying on SOC reports to support visibility of operational practices and security measurements.

Despite the concerns and challenges, public cloud adoption offers organizations scalability, flexibility, cost-effectiveness, global accessibility, reliability, security, compliance, innovation, and reduced maintenance burdens. By carefully addressing the potential concerns and leveraging the benefits of public cloud, organizations can harness the power of cloud computing to drive their digital transformation, enhance operational efficiency, and remain competitive in today's rapidly evolving business landscape.

Exit from cloud computing should be a last resort. The benefits outweigh the negatives extensively as new capabilities, features, and security measures are continually upgraded and improved. Furthermore, organizations may miss out on the innovation opportunities offered by the cloud. Cloud platforms provide access to a wide range of advanced services and technologies, such as AI, machine learning, and big data analytics, which can drive business growth and competitive advantage. By exiting the cloud, organizations may lose the ability to leverage these innovative capabilities.

Complexity does pertain to vulnerability in the context of cybersecurity, as intricate and convoluted systems often introduce more potential points of failure and make it harder to identify and address security vulnerabilities. With extensive knowledge and experience provided by the public cloud platforms, qualified and skilled engineers, security consultants, and architects, a robust and secure ecosystem can be attained with far greater flexibility than on-premises platforms.

SANS

Access more free educational content from SANS at:

SANS Reading Room

Checkout upcoming and on demand webcasts:

SANS Webcasts

CLOUD SECURITY:

Making Cloud Environments a Safer Place

aws Google Cloud Microsoft