BLADESTACK

Bridging the Gap Between Compliance and a Shared Services/Management Layer Implementation

CONTROL INHERITANCE

Control Inheritance

Demystifying the Shared Responsibility Model

Bridging the Gap Between Compliance and a Shared Services/Management Layer Implementation

Bhanu Jagasia

v 2.1, September 5, 2022

1. Table of Contents

2. Table of Figures

2.1. Notice of Non-Affiliation and Disclaimer

Bhanu Jagasia or bladestack.io is not affiliated, associated, authorized, endorsed by, or in any way officially connected with any of the companies and/or entities or any of their subsidiaries, as well as related names, marks, emblems and images are registered trademarks of those respective companies and/or entities.

The use in this whitepaper of trademarked names and/or images is strictly for editorial, educational and descriptive purposes, and no commercial claim to their use, or suggestion of sponsorship or endorsement, is made by Bhanu Jagasia or bladestack.io. Those words or terms that the author has reason to believe are trademarks are designated as such by the use of initial capitalization, where appropriate. However no attempt has been made to identify or designate all words or terms to which trademark or other proprietary rights may exist. Nothing contained herein is intended to express a judgment on, or affect the validity of legal status of, any word or term as a trademark, service mark, or other proprietary mark.

Please note that this whitepaper/guide is not endorsed by the National Institute of Science and Technology (NIST), Amazon Web Services (AWS), Microsoft, Google or any other organization. The whitepaper is merely an unofficial guide that Bhanu Jagasia has compiled to help organizations and industry to better understand cloud computing challenges with the shared responsibility model and security control inheritance.

This document is provided as a public service. Information, data, and software within this paper is “AS IS.” Bhanu Jagasia and/or bladestack.io makes no warranty of any kind, express, implied or statutory, including, without limitation, the implied warranty of merchantability, fitness for a particular purpose, non-infringement and data accuracy. Bhanu Jagasia and/or bladestack.io does not warrant or make any representations regarding the use of the software or the results thereof, including but not limited to the correctness, accuracy, reliability or usefulness of the software or hardware. You are solely responsible for determining the appropriateness of using and distributing the data/information and you assume all risks associated with its use, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and the unavailability or interruption of operation. Bhanu Jagasia and/or bladestack.io shall not be liable and you hereby release Bhanu Jagasia and/or bladestack.io from liability for any indirect, consequential, special, or incidental damages (including damages for loss of business profits, business interruption, loss of business information, and the like), whether arising in tort, contract, or otherwise, arising from or relating to the implementation or design (or the use of or inability to use this guidance), even if Bhanu Jagasia and/or bladestack.io has been advised of the possibility of such damages.

3. Abstract

The purpose of this whitepaper/guide is to help further clarify the shared responsibility, debunk common myths such as control inheritance being simple and straightforward, and help illuminate the many challenges hidden in plain sight, which may not present themselves until it's too late. Unclear control inheritance results in the consumption of time, costs, and energy due to the ambiguity of the controls and misperceived sense of security. Finally, the whitepaper aims to provide approachable shared services/management layer concepts which can be adopted for organizations looking to implement and deploy highly scalable services whilst maintaining compliance across a spectrum of services and applications.

Multiple occurrences plague the headline such as “Cloud Service Provider hacked, cloud is still not secure, data breach due to insecurity of the cloud!” and more, which ultimately can be traced back to the false sense of security provided by most shared responsibility models, which is essentially unclear control inheritance at its core.

The trend of continuing to provide new and innovative ways to offload responsibility to the cloud vendor is not slowing down anytime soon, and as more and more individuals, and corporations begin leveraging new innovative technologies, a greater risk of (but not limited to) breaches, misconfigurations, data leaks and vulnerability to cloud consumers presents itself. If control inheritance and the shared responsibility model is not further distilled and grey areas clearly revealed to the cloud consumer, the risk of another unnecessary breach exponentially increases. The Customer Responsibility Matrix (CRM) attempts to address this problem but fails short in many ways.

4. Target Audience

This whitepaper/guide is intended to serve a diverse audience of information system and information security professionals including:

- Individuals with information system, security, and/or risk management and oversight responsibilities (e.g., authorizing officials, chief information officers, senior information security officers, information system managers, information security managers);
- Individuals with information system development responsibilities (e.g., program managers, system designers and developers, information security engineers, systems integrators);
- Individuals with information security implementation and operational responsibilities (e.g., mission/business owners, information system owners, common control providers, information owners/stewards, system administrators, information system security officers);

- Individuals with information security assessment and monitoring responsibilities (e.g., auditors, Inspectors General, system evaluators, assessors, independent verifiers/validators, analysts, information system owners); and
- Commercial companies producing information technology products and systems, creating information security-related technologies, or providing information security services

- This space intentionally left blank -

5. Preface

Cloud security breaches consistently make news headlines. Yet, the stories of these breaches are often framed with vague explanations — a “misconfigured database” or mismanagement by an unnamed “third party.” The ambiguity that surrounds cloud computing can make securing the enterprise seem daunting. Concerns about security have led some CIOs to limit their organizational use of public cloud services.

However, the challenge exists not in the security of the cloud itself, but in the policies and technologies for security and control of the technology. In nearly all cases, it is the user, not the cloud provider, who fails to manage the controls used to protect an organization’s data. “CIOs need to ensure that their security teams are not holding back cloud initiatives with unsubstantiated cloud security worries,” says Jay Heiser, Vice President Analyst, Gartner. “Exaggerated fears can result in lost opportunity and inappropriate spending.”

CIOs must change their line of questioning from “Is the cloud secure?” to “Am I using the cloud securely?”

- **Through 2025, 90% of the organizations that fail to control public cloud use will inappropriately share sensitive data.**
  Cloud strategies usually lag behind cloud use. This leaves most organizations with a large amount of unsanctioned, and even unrecognized, public cloud use, creating unnecessary risk exposure. CIOs must develop a comprehensive enterprise strategy before cloud is implemented or risk the aftermath of an uncontrolled public cloud.

- **Through 2024, the majority of enterprises will continue to struggle with appropriately measuring cloud security risks.**
  Questions around the security of public cloud services are valid, but overestimating cloud risks can result in missed opportunities. Yet, while enterprises tended to overestimate cloud risk in the past, there’s been a recent shift — many organizations are now underestimating cloud risks. This can prove just as detrimental, if not more so, than an overestimation of risk. A well-designed risk management strategy, aligned with the overarching cloud strategy, can help organizations determine where public cloud use makes sense and what actions can be taken to reduce risk exposure.

- **Through 2025, 99% of cloud security failures will be the customer’s fault.**
  CIOs can combat this by implementing and enforcing policies on cloud ownership, responsibility and risk acceptance. They should also be sure to follow a life cycle approach to cloud governance and put in place central management and monitoring plans to cover the inherent complexity of multiload use.

6. Introduction

Acquiring a cloud service can be straightforward, we sign up with our preferred Cloud Service Provider (CSP) and soon thereafter are able to immediately deploy services and systems in a relatively simple matter; especially if you're using the services for personal matters. However, things can get tricky if you have been assigned or are responsible to leverage a CSP as part of your overall cloud strategy. Even at initial blush, with all the managed migration services and self-help migration tools at one's disposal, choosing and migrating to a CSP may seem simple and not overly complicated.

The real fun begins when you begin exploring all the new type of services the cloud vendor provides to you at your fingertips. From (but not limited to) managing and automating your entire Continuous Integration / Continuous Deployment (CI/CD) pipeline, to launching virtual hosts and/or assets only when your code demands it (serverless computing), to even simple services such as DNS and basic file storage, there are still many nuances to these services which require further unpacking.

If you're a commercial entity with a service offering which provides services to either the public or health sector and have services which run in the cloud, planning to migrate to the cloud, or just curious on how you may deploy your workloads in the most economical manner, chances are, you will be subjected to compliance regimes such as Federal Risk and Authorization Management Program (FedRAMP), Cybersecurity Maturity Model (CMMC), Health Insurance Portability and Accountability Act (HIPAA), which heavily relies on correct interpretation of the shared responsibility model and proper use of the Customer Responsibility Matrix (CRM). Even if you are not subject to any of these regulations – having systems and services in the cloud require a fundamental understanding of the shared responsibility matrix.

7. Abstraction

7.1. A Brief History of Abstraction

Abstraction in the cloud is simply a method to enable rapid deployment of applications, services, tools and or, data to reduce the cost and complexity of providing the consumer the underlying infrastructure and services which essentially simplifies operations. The goal of abstraction is to free up the business to focus on more strategic goals and allow businesses to use technology as a service rather than something that the business needs to build and manage themselves.

Today, there are more than a hundred types of "as a Service" offerings, each with their own levels of abstraction. The number does not begin to include all the different types of provider-specific service offerings provided by CSP's such Amazon Web Services (AWS), Microsoft Azure and, Google Cloud Platform (GCP). Complications begin when the consumer begins leveraging more and more abstracted services. The lines of the shared responsibility model begin to get more and more blurred. Eventually we all may be left at a position of second-guessing who is responsible for what.

7.2. Instruction Set Architecture

We begin our journey of abstraction with the Instruction Set Architecture (ISA), which is an abstract model of a computer. The ISA permitted multiple types of implementations that could vary in performance, physical size, and monetary cost (among other things). Just as the ISA served as the interface between software and hardware, a similar abstraction model concept applies to the cloud, with heights of abstraction reaching past the clouds. The ISA model was one of our initial forays into abstraction but as we will see, is certainly not the last.

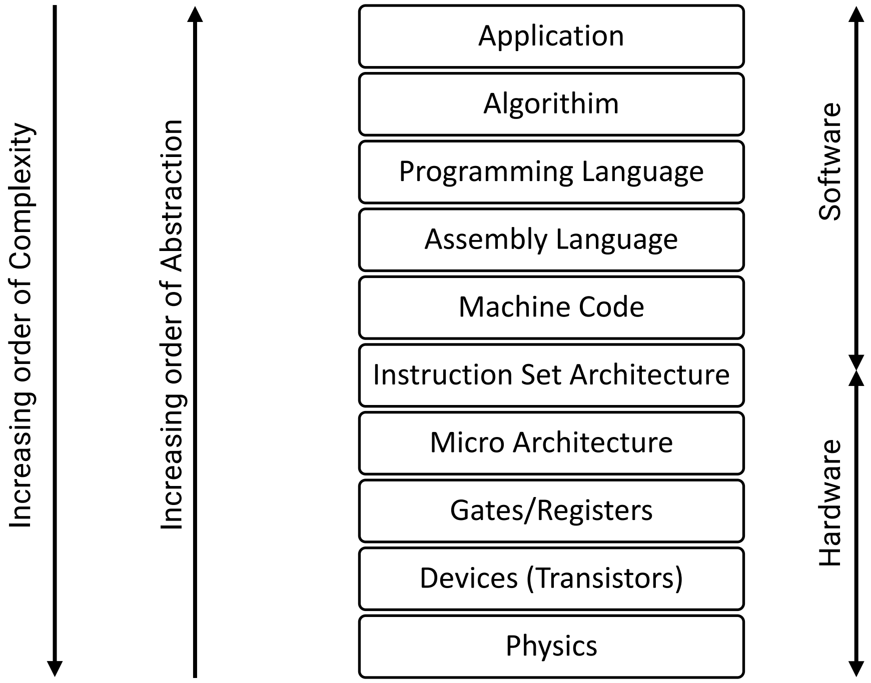

Figure 1 Instruction Set Architecture (ISA)

7.3. Open Systems Interconnection Model

Next came the Open Systems Interconnection Model (OSI Model) which most folks are more familiar with and is another conceptual model which lends its concepts to both Protocol Data Unit (PDU) and the Transmission Control Protocol/Internet Protocol (TCP/IP) stack models. The OSI Model characterizes and standardizes the communication functions of an either a telecommunication or computing system without regard to its underlying internal structure and technology. The model partitions a communication system into abstraction layers. The OSI model is one of the most recognized conceptual demonstrations of abstraction.

OSI Model			**Protocol data unit (PDU)**	**TCP/IP Stack**
Layers				
Host Layers	7	Application	Data	Application
	6	Presentation		
	5	Session		
	4	Transport	Segment, Datagram	Transport
Media Layers	3	Network	Packet	Internet
	2	Data Link	Frame	Network Access/Link
	1	Physical	Symbol	

Table 1 Open Systems Interconnection Model

The important idea to keep in mind about both the ISA and OSI models are that both are abstraction layers for digital systems, meaning the translation of the information/data is generally binary in format/nature (either zero or ones) where each bit is representative of two distinct amplitudes. In other words, the delineation between the abstraction layers are straightforward and there is generally no confusion between the responsibility of the layers within the stack. The physical layer only deals with the physical aspects of the transmission, whereas the data application layer contains the communications protocols and interface methods used in process-to-process communications across an Internet Protocol (IP) network.

To this day, abstraction continues to grow but not necessarily mature. In the cloud, a perimeter of responsibility exists between the cloud consumer and provider. The perimeter of shared responsibilities will vary depending on the type of services you choose to use.

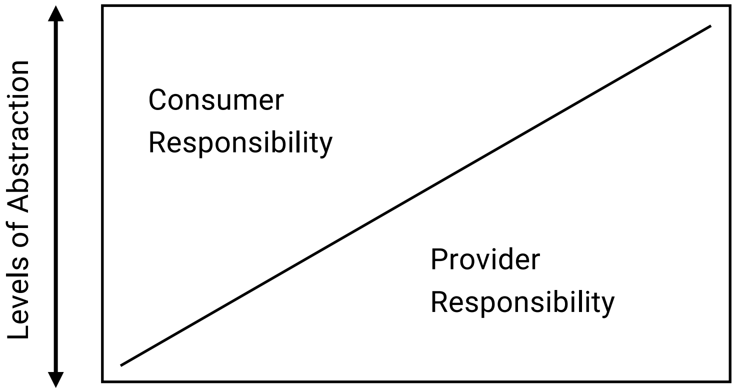

Figure 2 Basic Abstraction

For example, within Amazon Web Services (AWS), virtual instances provided through the label of Amazon Elastic Compute Cloud (EC2) were the initial layer of abstraction AWS introduced as part of their Infrastructure as a Service (IaaS). AWS EC2 is the service that allows AWS customers to launch virtual instances in the cloud. Customers retain responsibility of the guest operating system and above (middleware, applications, etc.) and the instances lifecycle. AWS retains the responsibility for managing the hardware and the hypervisor including their lifecycle. At the initial layer of abstraction, delineation of responsibility is again quite straightforward, but that was not always the case. When the cloud was a new and unfamiliar concept, many had incorrect assumptions and misconceptions with how the cloud functioned, what it meant, the advantages and so on.

A common initial misconception of the cloud is that if, for example, Microsoft Azure Government Cloud achieved FedRAMP accreditation and/or some other compliance regime, then any service or asset procured through the MS Azure Government Cloud is and will be preconfigured to meet the desired compliance/security standard out of the box – with no further configuration required. Many flocked out to deploy their respective applications on virtual instances provided by their CSP of choice and did not configure an absolute thing for the server/infrastructure which their application workloads ran on and were met with utter surprise and bewilderment when an advisor and/or hopefully not an competent auditor during an assessment pointed out both the customers misconfiguration and misunderstanding of the service.

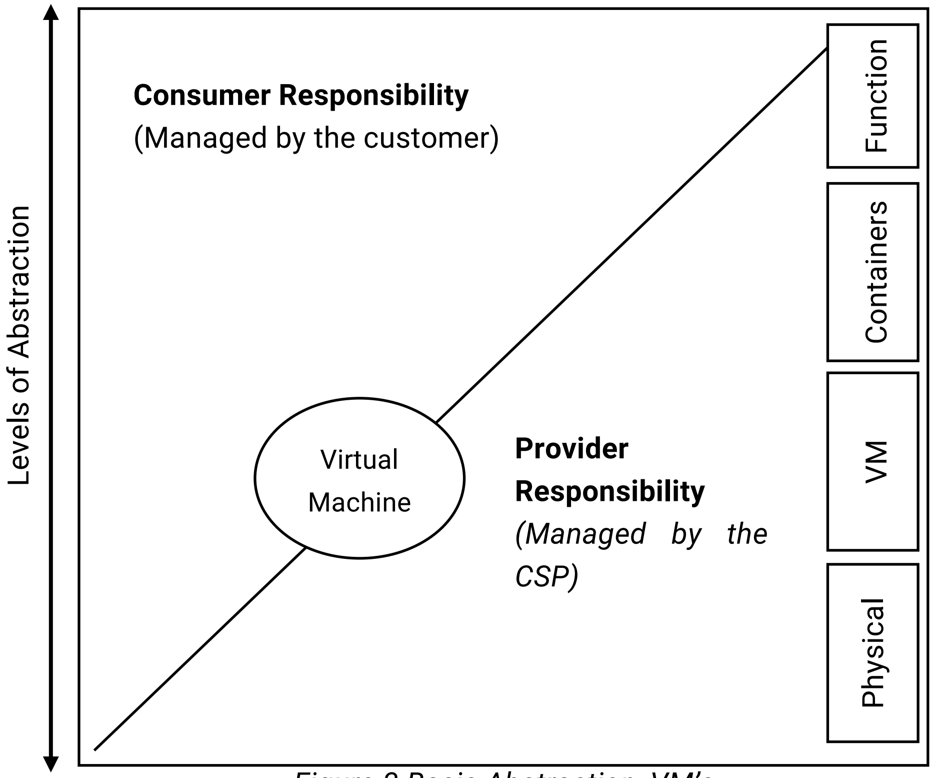

Figure 3 Basic Abstraction, VM's

Cloud computing is simply more than just procuring cloud services for applications or services to run on. Understanding the concept of what the customer is responsible for is another fundamental piece of cloud computing. In 2006, shared responsibility would continue to be somewhat straightforward during the early ages of AWS, Azure and other CSPs.

- This space intentionally left blank -

7.4. Containers & Microservices

As microservices began to emerge on the scenes of cloud computing, a new level of abstraction was introduced: containers. Containers themselves are not a new type of technology, but the rise of Docker a few years democratized access. Containers are essentially self-contained environments with boundaries which includes both your application and their respective dependencies such as libraries and software. Whereas an instance (or virtual machine) virtualizes a piece of hardware to allow the user to run dedicated operating systems, container technology virtualizes an operating system so that we can run separated application with different types of software dependencies.

Modern container-based solutions are usually implemented with two key logical pieces:

1. A container **controls plane** that is responsible for exposing the API and interfaces to define, deploy, and life cycle containers, also commonly referred to as the container orchestration layer.
2. A containers **data plane** that is responsible for providing capacity (as in CPU/Memory/Network/Storage) so that those containers can run and connect to a network. From a practical perspective, the compute resources are typically a Linux host or less often a Windows host where the containers launch and gain access to the network.

Several services exist to provide container abstraction, such as Amazon Elastic Container Service (ECS), Azure Container Instances and more recently Azure Kubernetes Service (AKS) and Amazon’s equivalent, Amazon Elastic Container Service for Kubernetes (EKS) – both based on Kubernetes. Kubernetes was originally developed by Google as an adjunct project of Google’s Borg project. Kubernetes has established itself as the de facto standard for container orchestration. Just like ECS or Azure Container Services, the aim for both these services are to free up customers from having to manage a containers control plane. In the past, customers would spin up instances and deploy/manage their own Kubernetes masters (masters are the name of the Kubernetes hosts running the control plane) on top of a virtualized abstraction.

The containers data plane is typically a fleet of instances which are managed by the customer. In this specific configuration, the containers control plane is managed by the CSP while the containers data plane is managed by the customer. In this configuration, the containers control data plane is managed by the CSP, while the containers data plane would be managed by the customer.

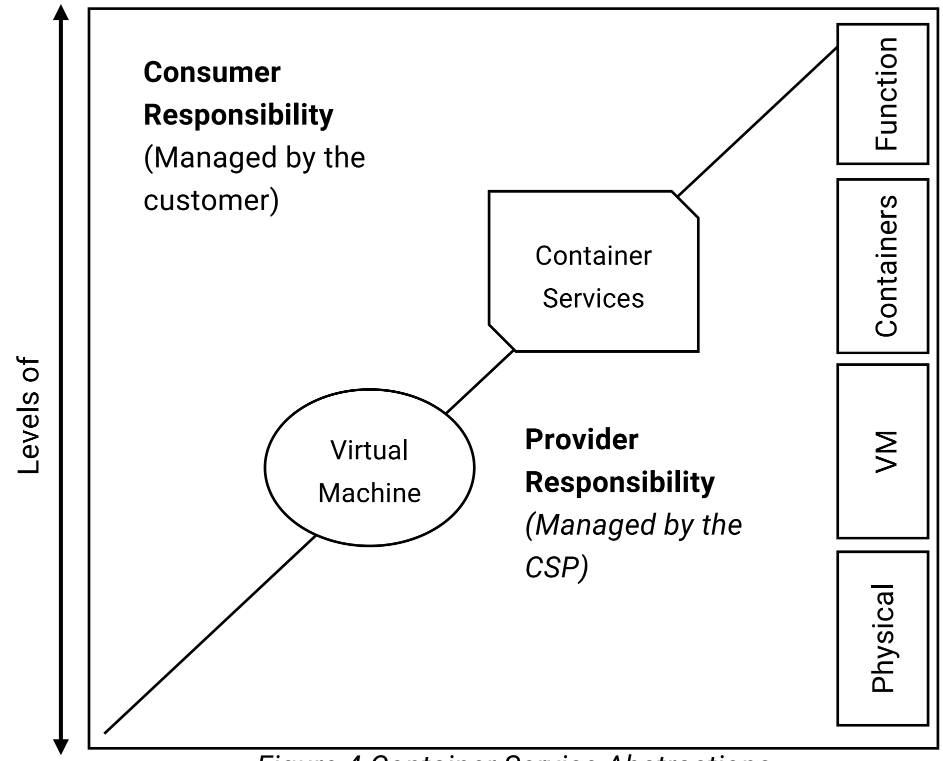

Figure 4 Container Service Abstractions

7.5. Serverless Functions

Another abstraction layer has been introduced by all three major CSP's, known as *serverless computing*. So, what is *serverless computing*? The short answer is:

> *"Anything providing resource pooling, rapid elasticity, and measured service (*as defined in the NIST Cloud Computing Definition) *in an opaque manner to the user*. "

The Cloud Native Computing Foundation (CNCF) defines serverless as:

> *"Where applications, bundled as one or more functions, are uploaded to a platform and then executed, scaled, and billed in response to the exact demand needed at the moment."*

The definition focuses on Function-as-a-Service (FaaS), which is further defined as:

> *"… code with functions that are triggered by events or HTTP requests. Developers deploy small units of code to the FaaS, which are executed as needed as discrete actions, scaling without the need to manage servers or any other underlying infrastructure."*

What does all this mean to the cloud consumer? Essentially, serverless allows developers to build and run applications and services without thinking about the servers executing the code.

Serverless services, or as mentioned above, FaaS providers, instrument this concept by allowing developers to upload the code while taking care of deploying, running, and scaling the resources, respectively. Thus, serverless can help create an environment that allows DevOp teams to focus on improving code, processes, and upgrade procedures, instead of provisioning, scaling, maintaining and similar administrative activities.

Instead of having to manage and run a full-blown OS instance to run your code, or having to track all software dependencies in a user-built container to run your code, serverless allows you to upload your code and have the CSP figure out how to run your code at scale. The key point for FaaS or serverless is that the cloud consumer does not have to manage the underlying infrastructure which the function is running on. No need to track the status of the physical hosts, no need to track the capacity of the fleet, no need to patch the OS where the function will be running. In a nutshell, no need to spend time and money on the undifferentiated heavy lifting.

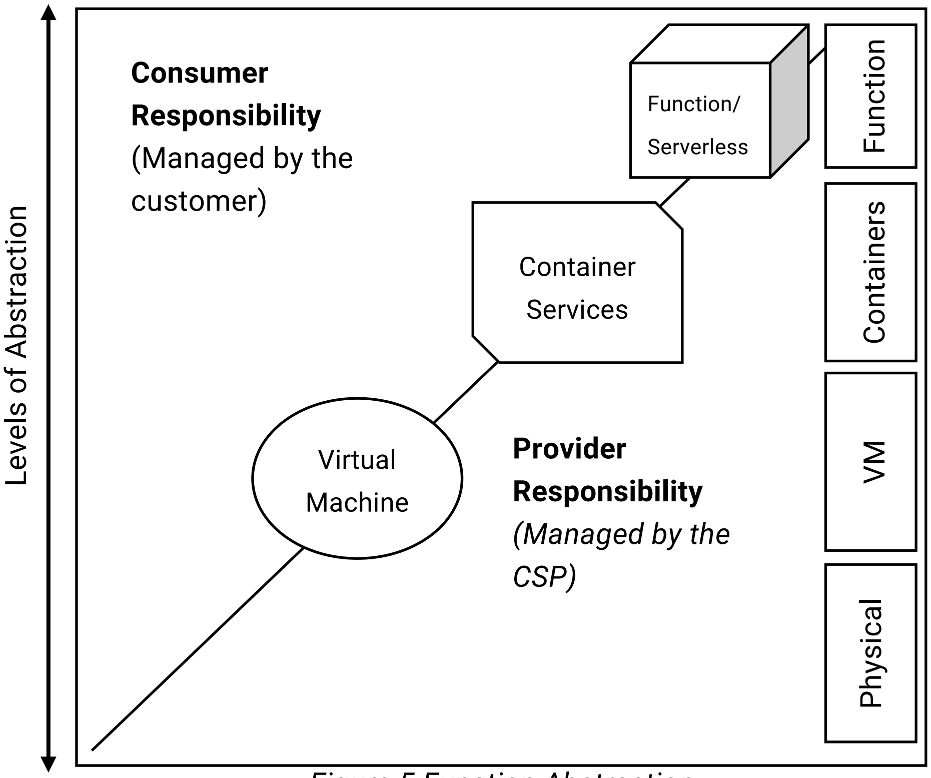

Figure 5 Function Abstraction

7.6. Back to Basics: Bare Metal

Also known as "no abstraction". Bare metal instances provide cloud consumers with direct access to the processor and memory of the underlying server. Bare metal instances are ideal for workloads that require access to hardware feature set (such as Intel VT-X), or for applications that need to run in non-virtualized environments for licensing or support requirements.

Bare metal instances allow cloud consumers to deploy applications that use physical hardware resources directly onto the CSP's infrastructure, and scale applications up and down in minutes. Just like other cloud provided instances, bare metals usually also provide support for different CSP services.

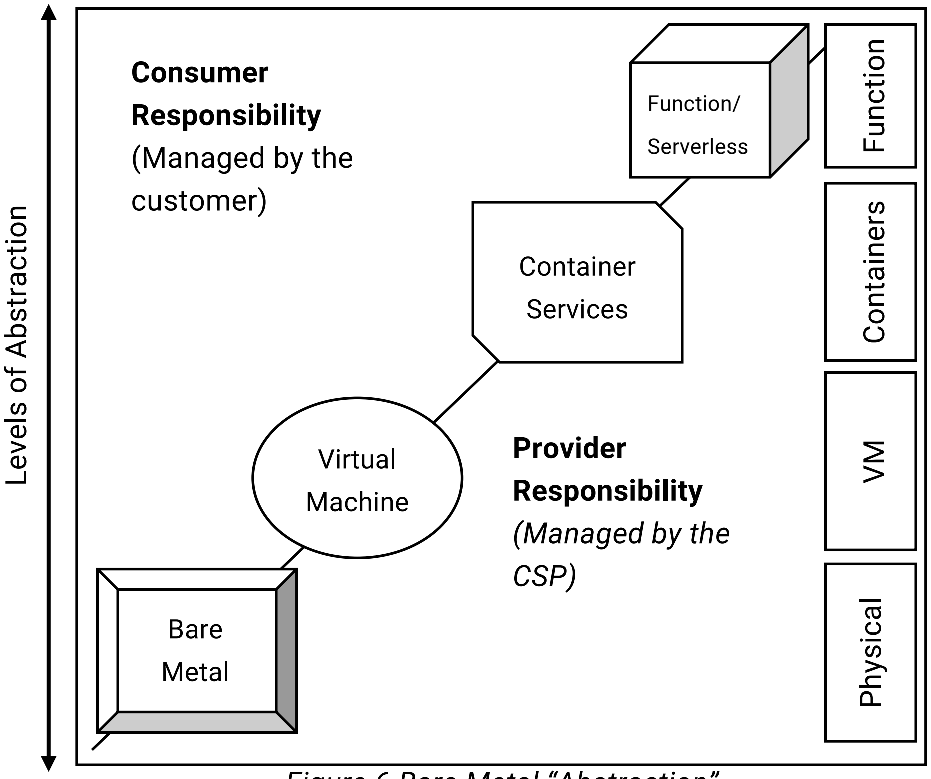

Figure 6 Bare Metal "Abstraction"

7.7. One Step Further

When we covered container abstraction, we identified there are two different fully managed containers control planes (controls plane and data plane) and specified there was not necessarily an option for the data plane. Some cloud customers were satisfied about being in full control of the instances. Other cloud consumers have been very outspoken and local that they wanted to get out of the business of managing the lifecycle of the data plane's infrastructure. Which brings us to clusterless/serverless method of running containers. Practically speaking, clusterless/serverless method of running containers pushes the containers data plane to fall into the 'provider space' responsibility. Which means the only compute unit is exposed to the user is the container abstraction, while the CSP will manage the data plane abstractions underneath.

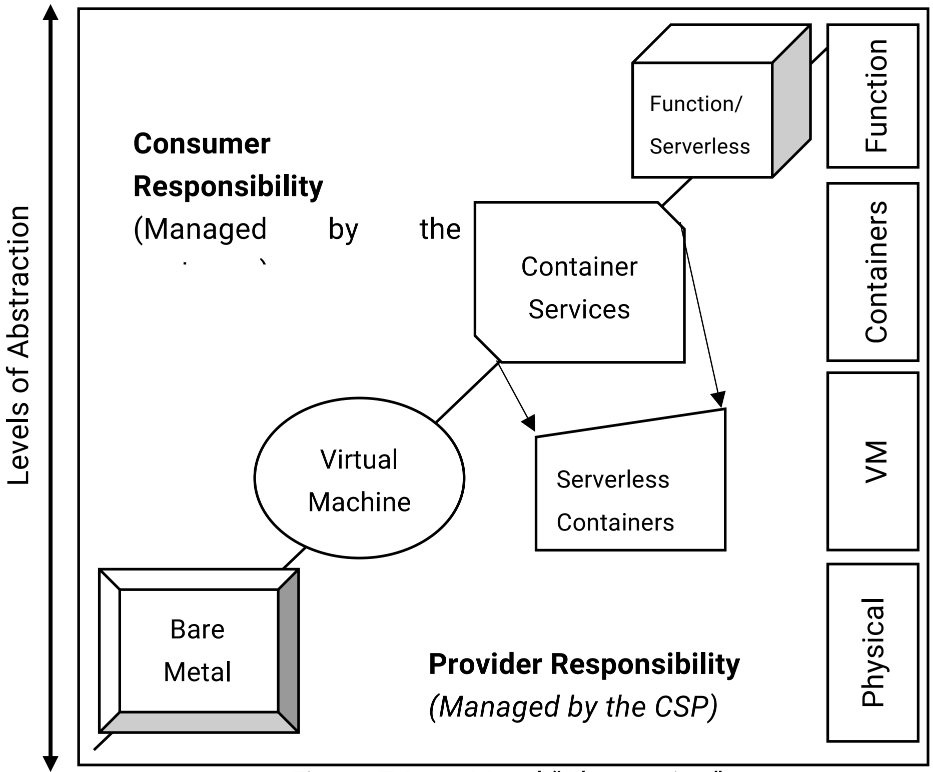

Figure 7 Bare Metal "Abstraction"

- This space intentionally left blank -

7.8. Concretizing the Abstract

Beginning with Virtual Machines in the cloud was a relatively simple concept to understand and accept back when Cloud Computing was fresh and new around 2006. The industry was already familiar with the responsibility between the host, guest operating systems and virtualization. To this extent, when organizations were questioned regarding implementation of their cloud workloads, there was little to no argument when customer responsibility was situated on virtual machines.

As discussed previously, the next levels of abstractions came from a variety of services in the last few years from microservices through containers, to cloud service provider services such as Elastic Beanstalk, AWS CloudTrail, managed databases, Content Delivery Services (CDN) and more. The abstracted services introduced new ways of interpreting the responsibility between customer(s) and CSP's. Even when Customer Responsibility Matrix's (CRM) were provided, most if not all CRM's did not go to the level of granularity required for organizations which were subjected to federal mandates or regulations to aide in properly configuring the leveraged systems and services.

As cloud security breaches began to occur at no fault of the CSP's, the misinformed majority are continuing to misinform others of the 'risks' of the cloud. When almost every cloud security breach has occurred due to a customer misconfiguration or non-configuration (running systems with all default configurations). Hundreds of online arguments have taken place between configurations and what needs to be done and what the customer should be provided out the gate, which is commonly referred to the as the 'baseline configuration'. To use NIST definition of a baseline configuration:

> *"A set of specifications for a system, or Configuration Item (CI) within a system, that has been formally reviewed and agreed on at a given point in time, and which can be changed only through change control procedures. The baseline configuration is used as a basis for future builds, releases, and/or changes."*

You can think of the baseline configuration as the system defaults. When you spin up/launch a service within the cloud and click 'next' without configuring anything and use the defaults for all options – you will ultimately be deploying the system/service with the default configuration baseline (with no options selected). However, all those options you just skipped are also your responsibility as the customer. So if you were to deploy a workload with all the default options and were given the option to configure multiple aspects of your service such as encryption, and your system is breached due to no encryption, you are and will be liable, as you were given the option, but ultimately did not configure encryption to be enabled. What we just described is essentially the Shared Responsibility Model.

8. Common Controls & The Cloud

Before we divulge too deep into the shared responsibility model, we need to take a slight detour into 'Common Controls' and how to interpret Common Controls in the Cloud. The Risk Management Framework (RMF) in combination with NIST Special Publication 800-53 introduced Common Controls. NIST defines common control as "a security control that is inheritable by one or more organizational information systems" and the revised Office of Management and Budget (OMB) Circular A-130 defines common control as a "security or privacy control that is inherited by multiple information systems or programs."

Common controls serve a very important purpose within the realm of information security compliance and operations. However, with the rapid proliferation of cloud-based information systems, there needs to be further clarity in the nomenclature as well as improved guidance regarding inheritance of common controls implemented within an organization versus controls implemented by an external entity such as a cloud service provider (CSP).

In a traditional IT environment, common controls were security controls that could be implemented centrally within an organization to support the security requirements for one or more organizational information systems. For example, consider a federal agency that implements 20 distinct information systems (general support systems and major applications). The physical security controls required by NIST SP 800-53 can be implemented centrally within the agency and support the security authorization of most or all that agency's information systems. Similarly, security controls related to security policies and procedures, security training and acquisition could be implemented very effectively as common controls within the agency. Implemented properly, common controls reduce the burden on individual system owners within an organization and enable implementation of the controls in a standardized, consistent, and cost-effective manner.

However, modern day information systems are much more complex. Many agencies are rapidly transitioning their existing information systems to leverage cloud services in the form of Infrastructure as a Service (IaaS), Platform as a Service (PaaS) or Software as a Services (SaaS). These cloud services are provisioned by CSPs external to the agency. Controls that are typically good candidates for allocation as common controls within an organization are often poor candidates for allocation as common controls when implemented by an external entity such as a CSP. Consider controls that implement security policies and procedures, physical security controls and training. When these controls are implemented by a CSP, it may not be appropriate to consider these controls as common controls that can be inherited by the agency information system using the cloud service.

While the above logic may appear obvious to security experts, we need to remember that for a typical federal information system categorized at the Moderate impact level (per Federal

Information Processing Standard 199), over 250 controls and control enhancements need to be selected and specified in accordance with NIST SP 800-53. While this is a daunting exercise for system owners in general, it can be made easier when an agency has identified and authorized common controls that can be inherited by other information systems within that agency.

When an agency information system leverages an IaaS/PaaS/SaaS cloud offering, things get more complicated. If the CSP is FedRAMP-authorized, it is very tempting for the system owner to assume that most, if not all, of the security controls implemented by the CSP can be inherited by the agency information system. As described above, though, this may lead to the inappropriate allocation of some controls as common controls that may put the agency information system at significant risk. A well-informed and security-savvy system owner may decide to evaluate each of the security controls implemented by the CSP to determine whether it can be inherited by the agency information system. However, this is a non-trivial exercise.

The FedRAMP body of guidance and templates seek to facilitate the authorization of CSPs to promote agency use of secure cloud services. There is little guidance for system owners that are trying to determine which controls implemented within the authorization boundary for the CSP can be inherited by the agency information system that is leveraging that CSP.

NIST SP 800-53, Rev 4 provides the three baselines for security controls based on the impact level (High, Moderate or Low) of an information system. After the system owner selects the appropriate impact level, what follows is the difficult task of tailoring the baseline security controls to align the controls with the specific conditions within the organization and the information system. The first step of tailoring is identifying and designating common controls. As pointed out above, this is a non-trivial exercise for system owners in general and is even more challenging for agency information systems that are utilizing cloud services.

The term common controls to include controls implemented by external entities (such as a CSP) adds more confusion than clarity. Controls provided by CSP’s makes it more difficult for system owners to differentiate between common controls implemented by providers within the agency from similar controls implemented by a CSP (and possibly not good candidates for inheritance).

9. Shared Responsibility Model

The shared responsibility model is a cloud security framework that dictates the security obligations of cloud computing. The shared responsibility model is one of the fundamental elements of a successful cloud deployment. A proper implementation of the shared responsibility model helps achieves several objectives such as taking advantage of the nature of the cloud, being efficient and economical with resources, clearly defined delineations between processes and people, and technology. The model below depicts the most known version of the Shared Responsibility model.

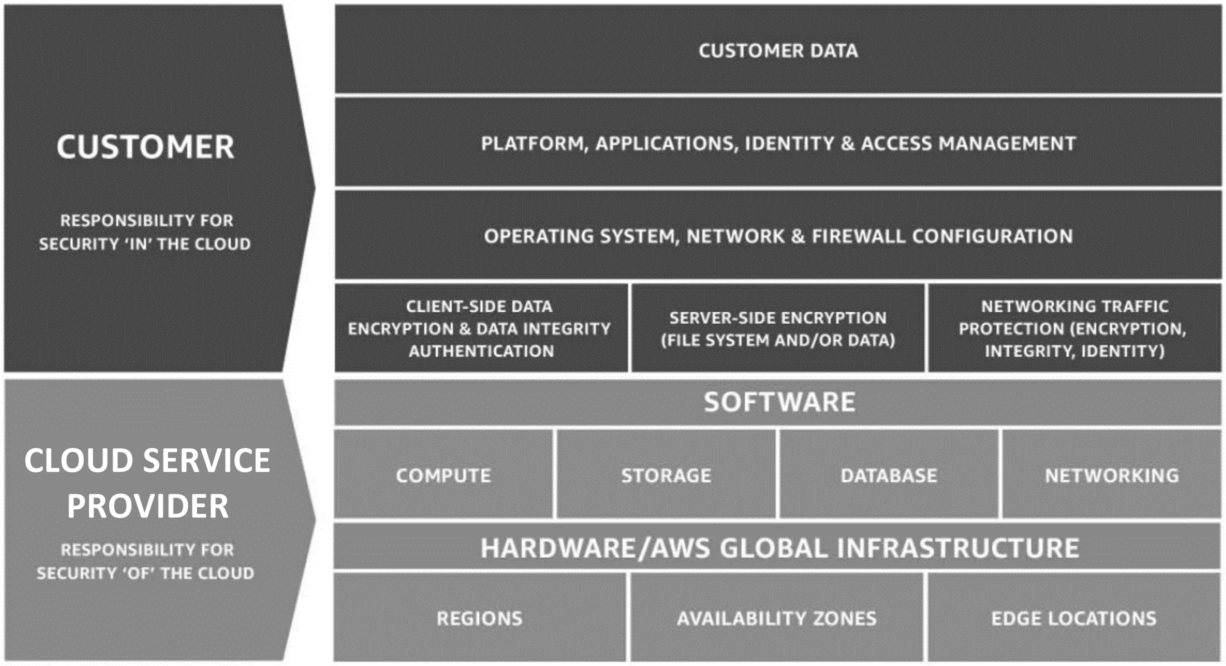
CUSTOMER
RESPONSIBILITY FOR SECURITY 'IN' THE CLOUD
CLOUD SERVICE PROVIDER
RESPONSIBILITY FOR SECURITY 'OF' THE CLOUD
CUSTOMER DATA
PLATFORM, APPLICATIONS, IDENTITY & ACCESS MANAGEMENT
OPERATING SYSTEM, NETWORK & FIREWALL CONFIGURATION
CLIENT-SIDE DATA ENCRYPTION & DATA INTEGRITY AUTHENTICATION
SERVER-SIDE ENCRYPTION (FILE SYSTEM AND/OR DATA)
NETWORKING TRAFFIC PROTECTION (ENCRYPTION, INTEGRITY, IDENTITY)
SOFTWARE
COMPUTE
STORAGE
DATABASE
NETWORKING
HARDWARE/AWS GLOBAL INFRASTRUCTURE
REGIONS
AVAILABILITY ZONES
EDGE LOCATIONS

Figure 8 Shared Responsibility Model

The shared model's intent is to help relieve the customer's operational burden as the CSP operates, manages, and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the service operates. The customer assumes responsibility and management of the guest operating system (including updates and security patches), other associated application software as well as the configuration of the CSP-provided security firewall.

In the shared responsibility model, customers need to carefully consider the services they choose as their responsibilities vary depending on the services used, the integration of those services into their cloud environment, and any applicable laws and regulations. The nature of the shared responsibility also provides the flexibility and customer control that permits the deployment. As shown in the figure above, the differentiation of responsibility is commonly referred to as Security "of" the Cloud versus Security "in" the Cloud.

When the Shared Responsibility Model was released, the model was relatively straightforward. The Cloud Service Provider (CSP) would be responsible for the "Security of the Cloud", meaning the CSP would be responsible for protecting and securing the infrastructure that runs all the services the CSP offers. This infrastructure would mostly be composed of the hardware, software, network, and the physical facilities that run and provide the cloud service.

- Cloud Service Provider (CSP) responsibility, "Security of the Cloud" – The CSP is responsible for protecting the infrastructure that runs all the services offered in the Cloud. The infrastructure is composed of the hardware, software, networking, and facilities that run the Cloud services.

- Customer responsibility, "Security in the Cloud" – Customer responsibility is determined by the Cloud services that a customer selects. The selection determines the amount of configuration work the customer must perform as part of their security responsibilities.

For example, a service providing computing resources, or 'instances', are categorized as Infrastructure as a Service (IaaS) and, as such, requires the customer to perform all of the necessary security configuration and management tasks. Customers that deploy instances are responsible for management of the guest operating system (including updates and security patches), any application software or utilities installed by the customer on the instances, and the configuration of the CSP-provided firewall for each instance.

For abstracted services, the CSP usually operates the infrastructure layer, the operating system, and platforms, and customers access the endpoints to store and retrieve data. Customers are responsible for managing their data (including encryption options), classifying their assets, and using any additional tools provided to apply the appropriate permissions.

To reiterate, the customer would be responsible for the "Security in the Cloud" which effectively meant if the consumer of the Cloud Service Offering (CSO) procured a relatively simple service such as virtual machine which provided compute services, the customer would be required to perform all of the necessary security configuration and management tasks for the virtual machine. Consumers would also be responsible for management of the guest operating system (including updates and security patches), any application software or utilities installed by the customer on the compute instances, the configuration, and so on and so forth. All compute configuration aspects within the Cloud would be managed by the Customer (cloud consumer) and therefore the customers responsibility.

The shared responsibility model also extends to IT controls. Just as the responsibility to operate the IT environment is shared between CSP and its customers, so is the management, operation and verification of IT controls shared. CSP's can help relieve customer burden of operating controls by managing those controls associated with the physical infrastructure deployed in the cloud environment that may previously have been managed by the customer. As every customer is deployed differently in the cloud, customers can take advantage of shifting management of certain IT controls to the cloud which results in a (new) distributed control environment. Customers then use the CSP provided control and compliance documentation available to them to perform their control evaluation and verification procedures as required.

As time and technology progressed, CSP's continued with more innovative ways to offload the customers responsibility to make the lives of the customer easier, however the transition of CSP's providing more innovative managed services had an impact of making the shared responsibility model and customer responsibility matrix more complex and challenging to interpret. CSP's began providing services that would manage the cloud consumers secrets, engines for artificial intelligence platforms, databases, fully managed container orchestration services and more. Overall, the introduction of the new services has been great for innovation and acceleration of deploying new applications and further increasing the speed of how quickly a cloud customer can deploy their new application/workload onto the cloud. As part of the movement of providing innovative managed services to cloud consumers, one key critical piece has been frequently overlooked – the shared responsibility model.

10. The Customer Responsibility Dilemma

Cloud consumers began consuming the innovative new services with the same false type of assumption when cloud computing was first adopted, that the customer would simply be able to begin leveraging the managed service in a turn-key manner where no additional configuration would be required. As we know now, the assumption for the most part is incorrect. Fully managed cloud services do exist, which provide clear shared responsibility model(s) and make a diligent effort to notify the customer of their responsibility. However, it would be unfair to state that the majority of the CSP's provide this level of tailored courtesy to all customers.

The Customer Responsibility Matrix seeks to help provide clarity to the controls which may be shared between the cloud consumer and the CSP – however in majority of the cases, the CRM is either an afterthought, or developed once but not diligently maintained and/or managed. In part, the inattention to the CRM is a result of the pace technology moves, which is not an excuse, but something we all need to consider and be more cognizant of. As we initially pointed out, cloud consumers are still puzzled and misinterpret what they themselves are responsible for within the cloud, especially if the service happens to be an innovative service which delivers a mixture of features – the customer can easily gain a false sense of security.

CSP’s do attempt to limit the risk exposure by limiting the baseline configuration of a system be in the most secure state, but most of the controversy is surrounded around what the CSP allows the cloud consumer to do. Security professionals know the weakest link is always the human factor. CSP’s have gotten more judicious and have begun including verification and acknowledgement messages when a cloud consumer attempts to configure a workload to be less secure.

Surprisingly, cloud consumers have also gotten much more astute with and comfortable with the cloud – Afterall we now many more innovative services being offered through the cloud built by the actual cloud consumers and not the actual CSP. The average day consumer has very little knowledge of the complexity and challenges which pose not only the CSP, but the cloud consumer which has built the workload

- This space intentionally left blank -

11. Shared Services/Management Layer

11.1. Multi-Application Stack

Companies migrating to or already in the cloud are beginning to implement their own flavor of the shared responsibility model within their organizations. Commonly referred to the CSP's "Shared Management Layer" or just the "Management Layer". The goal of the Shared Management Layer is like the Shared Responsibility model, where the cloud consumer themselves wish to abstract the cloud even further for their own purposes.

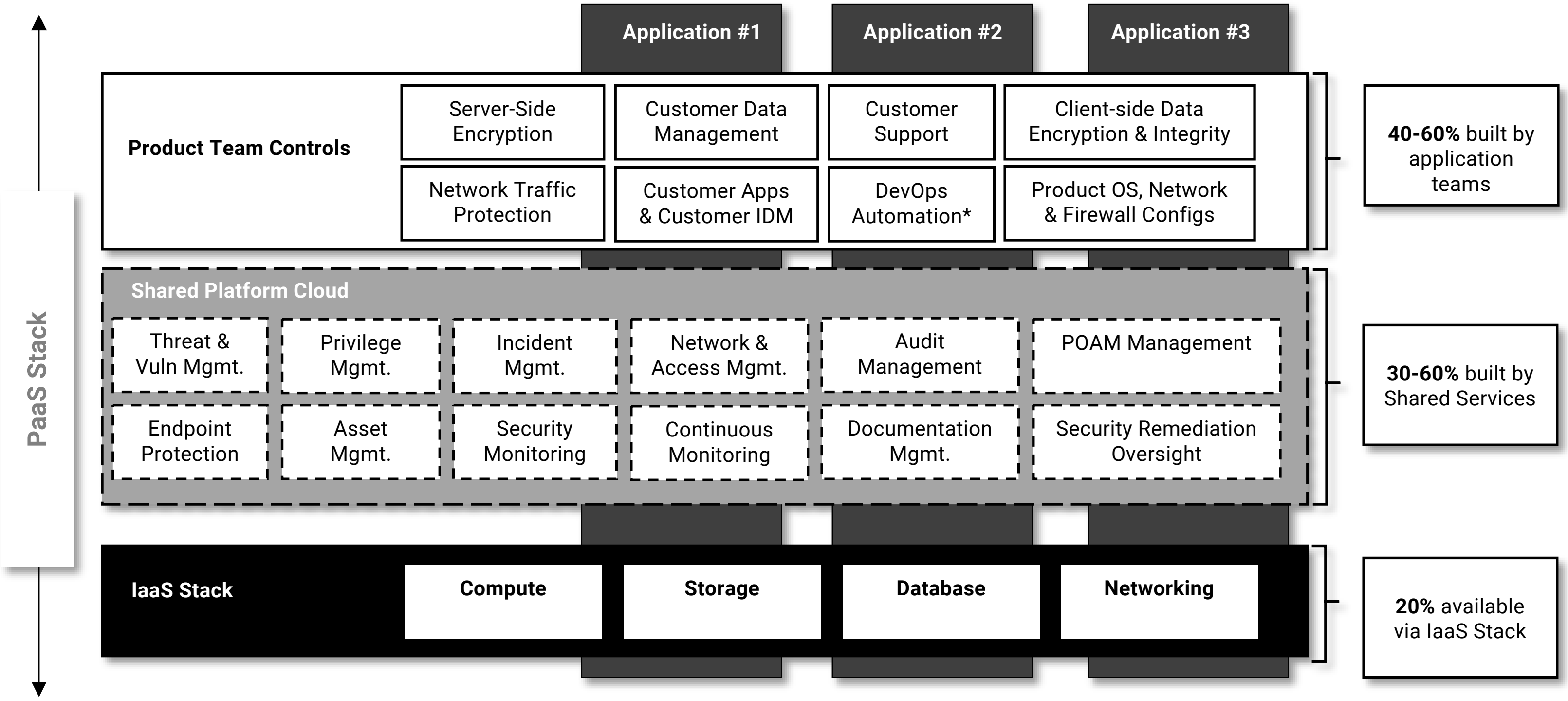

Figure 9 Example Multi Application System Stack

A common implementation of the model usually involves larger organizations with a suite of applications/services under one umbrella wishing to offload the individual application team(s) responsibility of management of their virtual infrastructure to a higher abstraction layer (the management layer). The offloading of responsibilities allows the companies individual application teams to focus on providing a better service and/or application as opposed to spending time administrating the system/service. The shared services approach can significantly reduce the economic operating footprint due the flexibility and nature of the shared services design. At first glance, the offloading and shared responsibility may seem straightforward and simple, however once we begin unpacking all the responsibilities individually, lines once again begin getting blurry.

The multi-application application stack represents a common conceptual overview of the Shared Serves/Management Layer within the cloud. The shared model can help relieve application teams' operational burden as abstracted Shared Services will support operate, manage and control the more administrative functions such as (but not limited to) vulnerability scanning, centralized logging, remote access, policy enforcement down to documentation. Within the Shared Services model, the application teams will generally assume responsibility and management of their individual products/service lines.

For most standard shared services deployment, a standard approach to shared functionality has been provided below in ***Error! Reference source not found.***. The functional overview at initial glance seems straightforward and easily consumed, however once we begin picking each layer off individually, it becomes apparent there is another level nuance that needs to be considered.

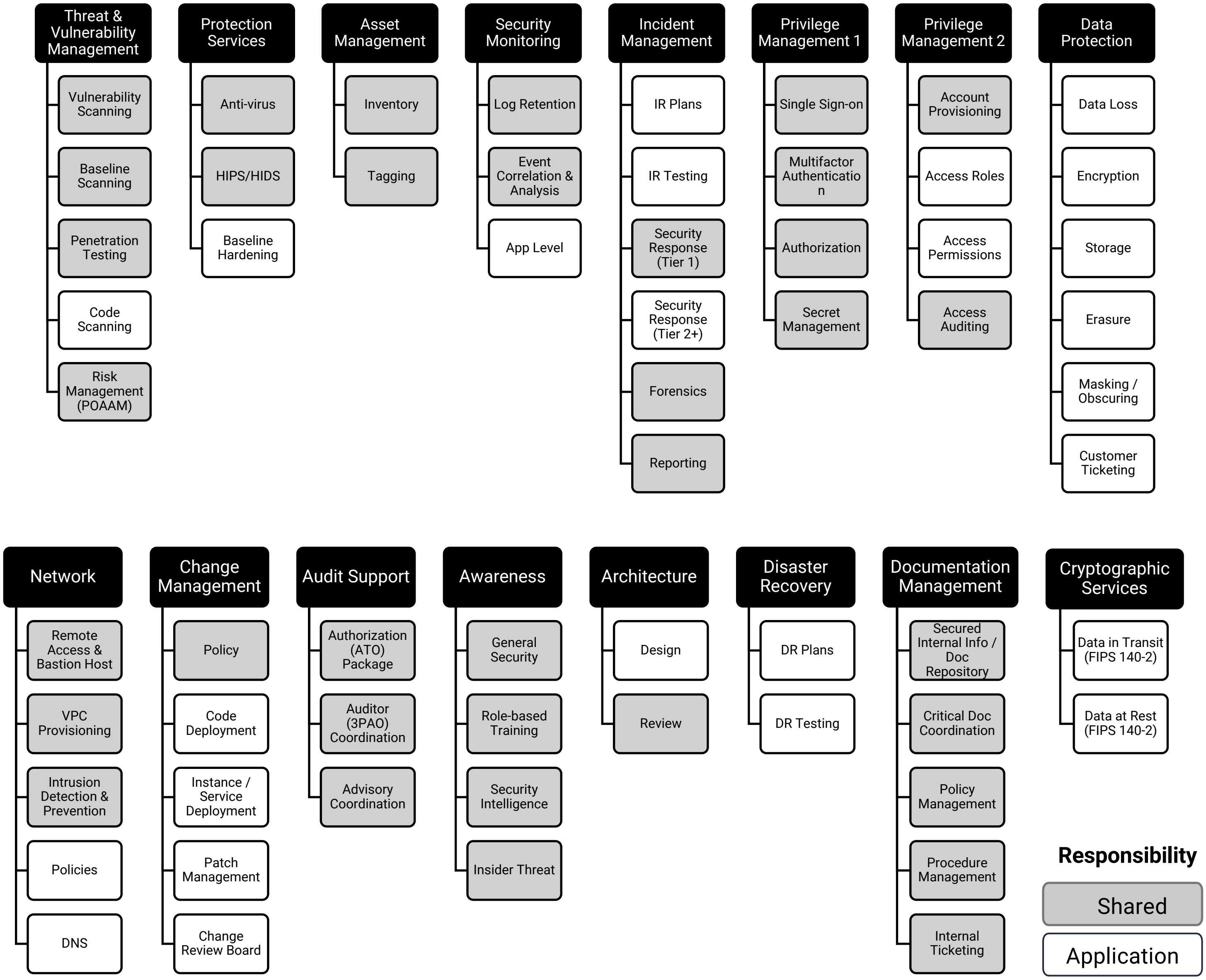

Figure 10 Shared Services – Functional Overview

Take for example, the category of Threat and Vulnerability Management and sub-category of Baseline Scanning within the conceptual diagram provided. The standard aspects such as centralized management and performing baseline configuration standard from the Shared Services plane makes complete sense. Deploy the vulnerability management tool/service on the shared plane and scan across all your multiple systems and services. However, once we begin peeling back the layers, questions such as the following:

1. *How are baselines established – what type of nuance is involved with providing baseline configuration scans to the application teams?*
2. *Is the Shared Services team expected to know the baseline standard for each individual application?*
3. *Is a report provided back to the product team for consumption?*
4. *Do the application teams provide their approved baseline configurations through an IT Service Management Tool or some other process for the Shared Services team consumption?*
5. *How do the application teams report deviations from the configuration baseline?*
6. *If Shared Services are identifying baseline configuration deficiencies, how are deficiencies reported back? Who is responsible for addressing deficiencies?*
7. *How are changes, deviations, updates to the baseline funneled back into the Plan of Actions & Milestones (POAM) & more...*

All of the questions can certainly be addressed by proper processes and procedures however there is an important detail which needs to be better clarified – the *Shared Responsibility* between the application teams and Shared Services – who does what, when and how? In most cases, a clear delineation of responsibility is not defined and unless the responsibility is properly documented and supplemented with adequate processes and procedures – the Shared Responsibility can become open-ended.

From a legal liability standpoint, an open-ended shared responsibility can have implications and ramifications on who is ultimately responsible. There are however usually legal clauses within a body of a CSP’s contract which covers potential liabilities. However, the clauses do not relinquish any entity from requiring developing a more comprehensive Customer Responsibility Model and better defining their adaptation of the Shared Responsibility Model.

The open-ended approach to the Shared Responsibility usually boils down to “if you manage it, you are responsible” with “it” being whatever aspect of the cloud system you are managing. However, the approach has the same challenge with nuance as previously discussed. Building on the shared services example, if you are responsible for managing the system configuration baseline as the product team, you still have *shared responsibility* with the shared services layer to carry out functions to fully satisfy the full function for configuration baseline management.

11.2. Multi-Application Shared Services Matrices

In the following sections we will have a basic framework laid out on how to approach an centralized Shared Services model based off the conceptual model in *Figure 9 Example Multi Application System Stack* and the shared responsibilities between the Shared Services team and the Application teams depicted in ***Error! Reference source not found.***. It is important to understand the standard responsibilities across both the Shared functions and Application functions. More importantly, it is critical to understand that the standard framework is not without its drawbacks – for example, the basic principles assume an somewhat standard application and shared services deployment – and in most cases the guidance can guide an organization on approach. However, not all systems and services are created equally, and the shared model may fall apart depending on the service offered or how a business operates.

As most of my work currently is involved with FedRAMP – we will find a lot of references to the FedRAMP framework as part of the guide. However, the type of cybersecurity framework necessarily do not apply as the conceptual models and functional matrices is based off the fundamental security principles of Confidentiality, Integrity and Availability (CIA) and the Risk Management Framework (RMF).

11.2.1. Threat & Vulnerability Management: Baseline Scanning

Overview Centralized management and performance of baseline configuration scanning against established hardening benchmarks	**Tools and Systems** • Vulnerability Scanning Mechanism
Responsibility **Shared Services Layer** • Conduct and configure baseline scans of all systems within the product environments to identify potential vulnerabilities and baseline deviations • Customize Center for Information Security (CIS) and DISA Security Technical Implementation Guide (STIG) benchmarks • Escalate baseline deficiencies for appropriate individuals to validate • Compile baseline deficiencies to track remediation **Application Teams** • Identify enhancements to baseline configurations • Receive baseline finding reports to validate and provide remediation responses • Perform mitigation of baseline issues commensurate with severity risk • Request ad-hoc scans	**Processes** • Continuous Monitoring • Configuration Management • NIST 800-128 Security-Focused Configuration Management
	Key Controls • CM-2 Baseline Configuration • CM-6 Configuration Settings

Risks and Caveats - Feed appropriate results into POA&M management for tracking. - Level of effort by Application Teams to identify POA&M line items and provide mitigation statements to all false positives identified. - Ability to efficiently create, test, and deploy CIS and STIG benchmarks through a product environment.	**Considerations** - Ability for teams to programmatically initiate scans and obtain reports

11.2.2. Threat & Vulnerability Management: Penetration Testing

Overview Facilitation of required internal and external penetration testing of the authorization boundary and associated environments	**Tools and Systems** Third Party Assessment Organizations (3PAO)
Responsibility **Shared Services Layer** - Schedule 3rd party entity (i.e. 3PAO) for internal and external penetration testing aligned with FedRAMP requirements on an annual basis - Coordinate with application teams to obtain required application information - Coordinate execution of tests to minimize impact to environments - Escalate findings for appropriate individuals to validate - Compile findings to track remediation **Application Teams** - Receive findings to validate and provide remediation responses - Perform mitigation of issues commensurate with severity risk - Request coordination to perform an out-of-cycle, ad-hoc, or special penetration testing of the product environment with established vendor.	**Processes** - Incident Response - NIST 800-115 Information Security Testing and Assessment
	Key Controls - CA-8 Penetration Testing
Risks and Caveats - Application Teams responsible for penetration testing of their product components that are outside the scope of FedRAMP requirements and scheduled testing cycles. This include testing required as part of company SDLC practices or ad-hoc requests. Additional fees involved.	**Considerations** - Third parties performing internal or external penetration testing which are not the selected 3PAO

11.2.3. Threat & Vulnerability Management: Risk Management (POA&M)

Overview Identification, tracking, and reporting of security weaknesses or deficiencies. Includes development of a plan of action and milestones (POAM) to document planned actions to reduce or eliminate vulnerabilities; management of monthly federal reporting requirements to assist agencies in monitoring the progress of corrective efforts.	**Tools and Systems** • Governance, Risk and Compliance (GRC) tool (If applicable) • All Vulnerability Management Tools
Responsibility **Shared Services Layer** • Maintain a central register of security weaknesses with restricted access by product • Identify, classify, and track security deficiencies on a consistent basis • Import findings detected by vulnerability management tools and other mechanisms • Assign issues to product stakeholders to validate and provide remediation response • Compile and submit monthly POAM reports to the federal program team **Application Teams** • Monitor tracked deficiencies with named staff; add findings; run reports • Establish and submit remediation plans, drive remediation activities • Update status and notify when remediated for evaluation • Validate monthly POAM submissions	**Processes** • NIST 800-115 Information Security Testing and Assessment • Internal audits • Continuous Monitoring
	Key Controls • CA-5 Plan of Action and Milestones • CA-7 Continuous Monitoring • RA-3 Risk Assessment
Risks and Caveats • Inability to maintain and drive action with Continuous Monitoring and POA&M items can lead to poor 3PAO reviews and status reports with FedRAMP PMO. Major issues could be directed to any agency Office of Inspector General that is participating with such CSP solutions. • PAO&M management requires dedicated resource(s) to properly manage and maintain the level of information that goes into identifying, managing, and reporting of POA&M items.	**Considerations** • Inclusion of application specific vulns/risks outside the immediate purview of the Shared Services team • Identification of additional risk feeds beyond vulnerability mgmt. tools and required scanning • Selection of centralized GRC tool

11.2.4. Protection Services: Anti-Virus

Overview Centralized management and scanning for both signatures based and non-signature-based malware on endpoints	**Tools and Systems** • Endpoint Protection Mechanism/Utility • Security information and event management (SIEM) system
Responsibility **Shared Services Layer** • Implement malware scan and detection profiles on endpoint systems • Receive and analyze alerts within security monitoring system; escalate as appropriate • Compile deficiencies to track remediation • Identify systems lacking required agents and policies **Application Teams** • Deploy required agents on applicable end points and register with centralized management services to receive policies • Receive alerts to validate potential issues and provide responses • Request ad-hoc scans • Request policy adjustments	**Processes** • Configuration Management • Continuous Monitoring • Incident Response
	Key Controls • SI-2 Flaw Remediation • SI-3 Malicious Code Protection • SI-7 Software, Firmware, and Information Integrity
Risks and Caveats • Scan Product instances for signature and non-signature-based vulnerabilities. • Excludes Boundary UTM requirements	**Considerations** • System and application design, functionality and capabilities may potentially require application team responsibility

11.2.5. Protection Services: HIPS/HIDS

Overview Centralized management and detection of potentially unauthorized system changes	**Tools and Systems** • Endpoint Protection Mechanism/Utility • Security information and event management (SIEM) system
Responsibility **Shared Services Layer** • Implement detection profiles on endpoint systems • Receive and analyze alerts within security monitoring system; escalate as appropriate • Compile deficiencies to track remediation • Identify systems lacking required agents and policies **Application Teams** • Deploy required agents on applicable end points and register with centralized management services to receive detection policies • Receive alerts to validate potential issues and provide responses • Request policy adjustments specific to application details	**Processes** • Configuration Management • Continuous Monitoring • Incident Response • NIST 800-94 Intrusion Detection and Prevention Systems (IDPS)
	Key Controls • SI-2 Flaw Remediation • SI-3 Malicious Code Protection • SI-7 Software, Firmware, and Information Integrity
Risks and Caveats • Detection policies may create performance issues. • Protection policies (Active blocking) may cause production issues and requires significant learning / testing prior to production release	**Considerations** • System and application design, functionality and capabilities may potentially require application team responsibility

11.2.6. Asset Management: Inventory

Overview Record keeping of an accurate inventory of assets to include endpoints, network devices, applications, and cloud services	**Tools and Systems** • Asset Management Tool • Configuration Management Database (CMDB)
Responsibility **Shared Services Layer** • Generate a point-in-time view of all assets • Categorize assets • Track asset ownership • Identify potentially missing assets or classes of assets • Compile and submit monthly asset inventory lists **Application Teams** • Review and reconcile monthly asset inventory lists • Manage application-specific system inventory	**Processes** • Continuous Monitoring
	Key Controls • CM-8 Information System Component Inventory
Risks and Caveats • Asset inventory dependent on appropriate tagging of resources	**Considerations** • In a model where application teams provision assets for customers (in the fashion of providing managed services as part of service offering) – asset inventory aggregated or provisioned by the individual application teams may include sensitive asset data

11.2.7. Asset Management: Tagging

Overview Standardization of system tags to facilitate auditing, financial reporting, and policy enforcement.	**Tools and Systems** • Automate Tagging
Responsibility **Shared Services Layer** • Identify resources owned by specific individuals or teams • Identify resources lacking minimally required tags • Perform financial and utilization analysis • Enforce policies based on tag values **Application Teams** • Generate reports based on groups of resources identified by tags • Create custom tags and reports • Enforce policies based on tag values • Reference tags to automate the operational elements (Infrastructure as Code)	**Processes** • Continuous Monitoring • Tagging Policies & Governance Process
	Key Controls • CM-8 Information System Component Inventory
Risks and Caveats • Remediation of Untagged Resources	**Considerations** • Focus on Required & Conditionally Required Tags • Integration with Authoritative Data Source

11.2.8. Security Monitoring: Log Retention

Overview Storage and protection of security relevant logs for defined time retention periods should they need to be recalled for investigative purposes	**Tools and Systems** • Auditing Configurations • Security information and event management (SIEM) system
Responsibility **Shared Services Layer** • Ingest *defined* security logs from application teams • Store security logs in a non-repudiated manner • Store logs online (for immediate access) for a minimum of 92 days • Store logs offline for a minimum of 1 year • Be alerted to logging failures • Dynamically grow storage capacity • Restrict access to logs based on role & product environment **Application Teams** • Retrieve logs specific to each individual product's environment based on role • Perform queries to search for specific log activity	**Processes** • Data Retention • Incident Response • Continuous Monitoring
	Key Controls • AU-4 Audit Storage Capacity • AU-5 Response to Audit Processing Failures • AU-9 Protection of Audit Information • AU-11 Audit Record Retention
Risks and Caveats • Requires definition of *security relevant* logs from application teams • Storage of additional logs identified by application teams that are not defined as "security relevant"	**Considerations** • Varying levels of National Archives and Records Administration (NARA) record retentions may pose challenges depending on system design and architecture

11.2.9. Security Monitoring: Event Correlation & Analysis

<table>
<tr>
<td><b>Overview</b>
Analysis of security-relevant event data to identify relationships and application of security- based analytics to perform actions</td>
<td><b>Tools and Systems</b>
• Security information and event management (SIEM) system</td>
</tr>
<tr>
<td rowspan="2"><b>Responsibility</b>
<b>Shared Services Layer</b>
• Ingest <i>defined</i> security logs from application teams and correlate in a meaningful way – both across all product environments and individually
• Generate alerts about potential security threats and key activities for review
• Create additional rules for detection and alerting based on team-defined criteria
• Store, at a minimum, logs online for 92 days and generated events for 1 year
• Utilize tool to perform initial investigations
<b>Application Teams</b>
• Receive reports about security activity within my product environment for investigation
• Work with Shared Services to create additional alerts specific to my product</td>
<td><b>Processes</b>
• Incident Response
• Continuous Monitoring</td>
</tr>
<tr>
<td><b>Key Controls</b>
• CA-7 Continuous Monitoring
• CM-4 Security Impact Analysis
• RA-5 Vulnerability Scanning
• SI-4 Information System Monitoring</td>
</tr>
<tr>
<td><b>Risks and Caveats</b>
• Requires definition of <i>security relevant</i> logs. System cannot ingest any type of logs
• SIEM systems can be expensive, need to be mindful of log volume</td>
<td><b>Considerations</b>
• Individual applications teams may leverage independent SIEM systems for application rules that may be outside of system configuration baseline.</td>
</tr>
</table>

11.2.10. Incident Management: Security Response (Tier 1)

Overview Provide initial triage, alerting, and escalation services for both potential and actual security incidents	**Tools and Systems** • Security information and event management (SIEM) system • Incident Response Platform • IT Service Management (ITSM) Tool
Responsibility **Shared Services Layer** • Receive alerts 24x7 on suspected critical security incidents requiring timely response • Conduct initial triage (where applicable) on potential incidents and obtain relevant facts • Identify and engage named product escalation contacts for further investigation • Facilitate resolution activities, including war rooms • Determine handoff to Tier2+ response capabilities **Application Teams** • Receive escalations from Shared Services involving suspect security incidents impacting my product • Start additional investigation as required	**Processes** • Incident Response
	Key Controls • IR-4 Incident Handling • IR-5 Incident Monitoring • IR-8 Incident Response Plan
Risks and Caveats • Only security incidents, not product application failures • Limited extent of Shared Services reach into product areas • Application Teams to define application specific alerting logic	**Considerations** • Incident scenarios for Shared Services Team response need to be defined • Individual applications may leverage independent SIEM systems for application rules –may require additional security response handling. • Retainer for IR experts (internal/external) when additional help required.

11.2.11. Incident Management: Forensics

Overview Provide a set of tools and an appropriate environment to facilitate additional investigation requiring forensic analysis	**Tools and Systems** • 3rd Party Incident Response Services • Forensic Tools & Utilities
Responsibility **Shared Services Layer** • Provision access to forensic-based toolsets for specific users to support investigations into potential compromises • Provide a "quarantine" area to temporarily locate suspected systems that require additional inspection • Engage IR experts and provide access to conduct forensic reviews **Application Teams** • Identify when a forensic investigation is required • Transfer suspected systems to a "quarantine" area for additional technical review • Access forensic-based toolsets to further investigate the incident	**Processes** • Incident Response • Third Party Engagement
	Key Controls • IR-4 Incident Handling • IR-7 Incident Response Assistance
Risks and Caveats • Limited extent of Shared Services reach into product areas • Limiting reach of forensic toolset into the specific product VPC requiring access • Establishing Operational Level Agreements (OLAs) with unique CSP groups & services which provide such services	**Considerations** • When performing forensic based investigations, what is done Product Cloud (VPC/VNET) vs "Quarantine/Sandboxed" environment. • Retainer for IR experts (internal/external) when additional help required.

11.2.12. Incident Management: Reporting

Overview Track and document details about security incidents and remediation activities that impact the authorization boundary and associated systems/data; reporting to authorities	**Tools and Systems** • IT Service Management (ITSM) Tool • Governance, Risk and Compliance (GRC) tool (If applicable)
Responsibility **Shared Services Layer** • Document, track, and report on security incidents across all products • Capture sensitive details about security incidents in central ticketing system • Coordinate root-cause analysis • Generate reports about security incidents for required incident reporting • Gain insight into incidents and evaluate for mandatory notification • Report both actual and suspect security incidents to defined authorities as mandated **Application Teams** • Capture sensitive details about security incidents impacting my product • View and update security incidents specific to my product • Leverage tracking repository to assist with reporting requirements	**Processes** • Incident Response • Continuous Monitoring
	Key Controls • IR-6 Incident Reporting • AU-7 Audit Reduction and Report Generation
Risks and Caveats • All sensitive details about security incidents must remain within the authorization boundary • Mandatory government and customer reporting to be captured within incident response plans	**Considerations** • Consider integration, if applicable, if an existing GRC tool provide similar capabilities

11.2.13. Privilege Management: Single Sign-On & MFA

Overview Centralized authentication service allowing for one (1) set of credentials for each CSP user to access all environments.	**Tools and Systems** • Directory Services • Multi-Factor Authentication • Ephemeral keys
Responsibility **Shared Services Layer** • Identify and authenticate the internal (i.e. CSP) users requiring access to the management and product environments • Register MFA credentials and assign to staff • Provide ephemeral key-based credentialing system for access to/within product environments • Determine when and where a credential was used and the granting/revocation of roles and permissions **Application Teams** • Use a single set of credentials to access both the management and product environments • Leverage ephemeral key-based credentials to reduce the need for key based management • Application teams must implement SAML assertion capabilities within the web application to allow for customer sign-on authentication	**Processes** • Staff On-boarding
	Key Controls • IA-2 Identification and Authentication (Organizational Users) • IA-4 Identifier Management • IA-5 Authenticator Management • IA-11 Re-Authentication
Risks and Caveats • Customer sign-on & MFA not included (i.e. PIV & CAC)	**Considerations** • Unique product requirements for system/app sign-on • Incident Response triggers for failed logins

11.2.14. Privilege Management: Authorization

Overview Centralized authorization service allowing for role-based access control	**Tools and Systems** • Directory Services
Responsibility **Shared Services Layer** • Create internal (i.e. CSP) user and admin roles users for management and product environments • Add/revoke user access to roles and privileges in the management platform • Audit granting and revocation of roles and permissions • Delegate aspects of privilege model to application teams to facilitate control access to their assets **Application Teams** • Define roles and associated privileges for access within the product environments • Identify named users to access product environment • Request the addition/removal of users to/from roles • Adjust privileges and assignments for select roles	**Processes** • Staff On-boarding • Staff Off-boarding
	Key Controls • IA-2 Identification and Authentication (Organizational Users) • IA-4 Identifier Management • AC-2 Account Management • AC-3 Access Enforcement • AC-6 Least Privilege
Risks and Caveats • Customer sign-on & MFA not included only applies to CSP staff • Lack of notification when staff depart may result in ex-staff retaining access for a period	**Considerations** • Unique product requirements for system/app sign-on • Incident Response triggers for failed authorizations • Do customers have privileged account/access management responsibilities?

11.2.15. Privilege Management: Secret Management

Overview Centralized management of storing and sharing secrets including passwords, keys, tokens etc.	**Tools and Systems** • Secrets Manager
Responsibility **Shared Services Layer** • Securely store secrets and delegate access to other users • Revoke service users and select service keys • Centrally manage retention and expiration policies **Application Teams** • Store keys and secrets in a secured service • Manage access to keys and secrets • Retrieve and rotate keys and secrets • Use current credentials without needing to know if/when credentials have been revoked or rotated	**Processes** • Key Management Plan • FIPS 140-2 Validation
	Key Controls • SC-12 Cryptographic Key Establishment and Management • SC-13 Cryptographic Protection • SC-28 Protection of Information at Rest
Risks and Caveats • Data sensitivity requirements (such as DoD) may have specific secrets requirements. Depending on the implementation /system design, secrets may not reside alongside with other secrets`	**Considerations** • Potential for customer key management and corresponding requirements based on service offering

11.2.16. Privilege Management: Account Provisioning

Overview

Centralized provisioning and deprovisioning of accounts to ensure only authorized and approved staff obtain and retain access

Tools and Systems

- Directory Services
- Corporate Directory Services
- IT Service Management (ITSM) Tool

Responsibility

Shared Services Layer

- Create internal (i.e. CSP) user accounts to access environments
- Enforce account approval workflows
- Review, validate, and document individual requirements prior to issuing a user account
- Disable and delete user access accounts; receive departing staff notifications from company directory to evaluate
- Audit creation and deletion of accounts

Application Teams

- Identify named users to access the environment
- Request the addition/removal of users within the environment
- Request roles to be assigned to user accounts

Processes

- Staff On-boarding
- Staff Off-boarding

Key Controls

- AC-2 Account Management

Risks and Caveats

- Required checks must be successfully completed and documented before account created
- Customer sign-on & MFA not included only applies to CSP staff
- Lack of notification when staff depart may result in ex-staff retaining access for a period

Considerations

- Validate required checks (including HR) before staff issued account
- Notification from Corporate AD/Directory Services when corresponding staff depart based on reference account info between environments.

11.2.17. Privilege Management: Access Auditing

Overview Auditing activities to ensure only the appropriate users, have the appropriate access, to the appropriate systems	**Tools and Systems** • Security information and event management (SIEM) system • Cloud Platform Object Storage
Responsibility **Shared Services Layer** • Coordinate routine audits of active users and privileged access across all environments • Generate automated reports of users, roles, permissions, and access attempts • Review access permissions with a purpose-built system to identify potential issues • Receive alerts of changes to privileged access • Ensure all deployed audit mechanisms within each application teams boundary correlates with baseline configuration, identify and report on deltas **Application Teams** • Review access logs and potential issues related to the product environment • Validate appropriateness of user access and make changes as appropriate • Receive alerts of user changes to product admin roles • Ensure all hosts within the application/system boundary have appropriate agents/mechanisms in place deployed for proper reporting	**Processes** • Incident Response • Continuous Monitoring
	Key Controls • AU-2 Audit Events • AU-6 Audit Review, Analysis, and Reporting • AU-12 Audit Generation
Risks and Caveats • Customer users and access not included	**Considerations** • Access systems within product environment not managed by Shared Services may require special handling by application teams

11.2.18. Network: Remote Access/Bastion Host

Overview Centralized remote access to all environments to perform administrative tasks	**Tools and Systems** • Remote Desktop Services • Multi-Factor Authentication (MFA)
Responsibility **Shared Services Layer** • Provide a single and hardened point of entry for remote access into all environments • Restrict remote access to enforce origination from pre-authorized networks (i.e. CSP Corp) • Enforce policies based on user role (i.e. restrict environment "jump-to" capabilities) • Provide remote access to all systems and consoles including Windows, Linux, and AWS Console • Monitor "human" traffic accessing product environment **Application Teams** • Gain access to the product environment and approved systems to conduct remote management activities • Transfer files into the environment	**Processes** • Staff On-boarding
	Key Controls • AC-17 Remote Access
Risks and Caveats • Application Teams looking to circumvent the use of the bastion host due to connectivity or personal inconvenience. • User may be required to be on Corporate network (office or VPN) to attempt access to bastion host	**Considerations** • Remote access to instances via bastion host vs. AWS console access via Broker • Process to transfer of files into the environment

11.2.19. Network: VPC Provisioning

Overview
Deployment of segregated virtual networks that serve as the network infrastructure base to deploy product components to deliver customer-facing services

Tools and Systems

- Infrastructure as Code (IaC) Deployment Orchestration Tools

Responsibility
Shared Services Layer

- Provision segregated VPCs to deploy products and supporting services
- Provide interconnectivity (i.e. peering) between VPCs for consumption of services
- Allocate network space
- Create baseline network interfaces and security groups
- Advise on best practices and coordinate 3rd party technical advice/assistance for securing and deploying components into VPC

Application Teams

- Utilize provided VPC/VNETs for deployment of components
- Coordinate with Shared Services team to address specific deployment requirements related to VPC architecture and infrastructure

Processes

- Change Management

Key Controls

- CM-3 Configuration Change Control
- CM-6 Configuration Settings
- CM-8 Information System Component Inventory

Risks and Caveats

- Application Teams' requirements around connectivity/VPC/VNET design which may:
    - 1) break security model;
    - 2) Might be unapproved/untested design against best practices;
    - 3) forces exceeding operational overhead and;
    - 4) exceeds implied or explicit limits (IP space allocation, cloud platform limits, etc.)

Considerations

- Specific, non-standard requirements from the Application Teams if any
- Direct connect, if required, will take a significant amount of time due to physical interconnects (est. 2 months for link activation + ~1 month for configuration on usual)

11.2.20. Network: Intrusion Detection & Prevention

Overview Identification, logging, and prevention of threats occurring across network boundaries and within the network	**Tools and Systems** • Unified Threat Management Tool • Endpoint Protection • Advanced Threat Protection (ATP) Services Tools & Appliances
Responsibility **Shared Services Layer** • Inspect all network traffic traversing internet and authorization boundaries • Perform deep packet inspection • Decrypt and re-encrypt boundary traffic to enable inspection • Apply detection and prevention policies to mitigate potential threats • Leverage endpoint agents to identify potential network threats within environments • Receive and analyze alerts within security monitoring system; escalate as appropriate **Application Teams** • Deploy required agents on applicable end points and register with centralized management services to obtain intrusion detection policies • Receive alerts to validate potential issues and provide responses • Request policy adjustments specific to application details	**Processes** • Continuous Monitoring • Incident Response • NIST 800-94 Intrusion Detection and Prevention Systems (IDPS) • NIST 800-41 R1 Firewalls and Firewall Policy
	Key Controls • SC-5 Denial of Service Protection • SC-7 Boundary Protection • SI-4 Information System Monitoring
Risks and Caveats • Inspection of Management traffic vs Customer data plane. To meet requirements, ALL traffic at the boundary is to be inspected, not selective traffic flows. This means breaking apart encrypted streams for inspection, which could have a crippling performance impact on some products.	**Considerations** • Technology selection may pose a challenge depending on type of service offering/system design & deployment • Architecture - Centralized traffic inspection for all environments or per product/VPC environment • Scanning all boundary traffic (including customer data plane) vs selective traffic flows (i.e. management traffic)

11.2.21. Network: DNS

Overview Distributed management of registrations to include hostnames, subdomains, and delegations within internal and external domain name services (DNS)	**Tools and Systems** • AWS Route 53 DNS (internal) • CSP Corporate DNS servers (external)
Responsibility **Shared Services Layer** • Provision supported *internal* DNS services within product environments • Delegate *internal* DNS administration to named product administrators **Application Teams** • Engage Corporate IT to purchase and/or manage *external* DNS registrations and domains • Manage *internal* DNS registrations within subdomains • Align with established naming conventions • Implement *internal* DNS solutions to address unique application needs	**Processes** • Configuration Management
	Key Controls • SC-20 Secure Name / Address Resolution Service (Auth Source) • SC-21 Secure Name / Address Resolution Service (Resolver) • SC-22 Architecture and Provisioning for Name /Address Resolution Service
Risks and Caveats • Application Teams may require separate DNS systems/services to support application delivery requirements beyond scope of Corporate IT and Shared Services.	**Considerations** • External entries may require name obfuscation • Internal subdomain naming convention (i.e. product1.fedcloud.com)

11.2.22. Change Management: Policy

Overview Organizational policy oversight for any changes impacting components within the authorization boundary, including product offerings	**Tools and Systems** • IT Service Management (ITSM) Tool • Collaborative Information Management
Responsibility **Shared Services Layer** • Establish change management policy governing the entire environment • Identify criteria and requirements that must be met for changes • Make recommendations on how to meet change management requirements • Update and maintain policy documentation • Disseminate change management policy to stakeholders and users **Application Teams** • Access and review change management policy • Implement supporting processes and toolsets to align changes with the policy • Enforce adherence to the policy • Demonstrate adherence to the policy through change artifacts	**Processes** • Shared Services Organizational Policy
	Key Controls • CM-1 Configuration Management Policy and Procedures • All “-1” policy controls in all control families
Risks and Caveats • Application Teams are responsible for DevOps within their environment. They will need to adhere to the mandatory policy requirements governing the Shared Services. This may require different change management approaches and toolsets used in previous environments (i.e. non-Shared Services). Additional burden on application teams to demonstrate compliance to policy.	**Considerations** • Movement of product binaries from Corporate environment into Shared Services and required inspections.

11.2.23. Audit Support: Authorization (ATO) Package

Overview

Submission and inquiry support of all required documents that constitute the authorization package for government acceptance. Required documents include, but not limited to: System Security Plan (SSP), Workbooks, Policies, Plans, and system descriptions.

Tools and Systems

- Governance, Risk and Compliance (GRC) tool (If applicable)
- Secure Document Repo
- Collaborative Information Management

Responsibility

Shared Services Layer

- Submit formal documentation authorization packages for auditor and government/agency reviews
- Participate in the auditor and government review process and coordinate responses
- Address questions related to shared controls and where possible, product environments
- Engage appropriate product stakeholders to address questions as required
- Monitor and report on overall review progress

Application Teams

- Provide and validate information within the authorization package
- Identify and coordinate key personnel to address product specific questions
- Track progress on review progress and potential risks

Processes

- Risk Management Framework (RMF) life cycle program
- Review & Maintenance

Key Controls

- CA-2 Security Assessments
- CA-6 Security Authorization

Risks and Caveats

- Products not ready for package submission during planned cycles may be deferred to a subsequent cycle to prevent delays to other products in the cycle.

Considerations

- Internal approval process for package submission

11.2.24. Audit Support: Auditor Coordination

Overview Engagement and coordination with external third-party assessment organization (3PAO) to perform readiness assessments, annual audits, and other inspections as required.	**Tools and Systems** • Governance, Risk and Compliance (GRC) tool (If applicable) • IT Service Management (ITSM) Tool • Secure Document Repo
Responsibility **Shared Services Layer** • Select, contract, and partner with preferred audit organization to provide comprehensive assessment and audit services across all environments • Schedule audits to be performed at required cycles • Capture, assign, and track audit evidence requests for appropriate parties to collect • Provide evidence repository to share with auditor • Monitor and report on overall audit progress **Application Teams** • Obtain and submit requested audit evidence • Schedule appropriate stakeholders to address questions • Track progress on audit activities and potential risks	**Processes** • Continuous Monitoring • Third Party Engagement
	Key Controls • AU-9 Protection of Audit Information • CA-2 Security Assessments
Risks and Caveats • Teams attempting to leverage 3rd party audit services outside of contracted partner will significantly increase audit costs, disrupt audit synergies, and increase level of effort to perform audit activities. • Lack of Shared Services internal compliance staff will contribute to audit delays	**Considerations** • Shared Services introduces multiple complexities – ensure system design and shared models are depicted in a manner to provide clear and adequate scoping. Improper depiction of shared services can result in scope creep.

11.2.25. Audit Support: Advisory Coordination

Overview Engagement and coordination with external advisory services specializing in FedRAMP to provide guidance and assistance to meet requirements.	**Tools and Systems** • IT Service Management (ITSM) Tool • Secure Document Repo
Responsibility **Shared Services Layer** • Select advisory organizations as preferred partners and manage overall relationship • Obtain comprehensive guidance across all environments to include associated dependencies • Obtain external guidance for complex and challenging issues on behalf of any team • Schedule and coordinate assessments • Contract for additional services as required • Coordinate the sharing of information between parties • Provide oversight for all engagements and communication with advisory **Application Teams** • Escalate questions to the Shared Services team for review to determine if external advisory referral is required • Engage advisory service once level of engagement agreed and established	**Processes** • Third Party Engagement
	Key Controls • AU-9 Protection of Audit Information • CA-2 Security Assessments
Risks and Caveats • Teams leveraging 3rd party advisory services outside of contracted partners may increase costs and efforts due to lack of synergies. • Audit (3PAO) and Advisory organizations must be different companies.	**Considerations** • Resource forecasting; ensure proper project planning is in place prior to beginning engagements with 3rd party advisors as timelines and business objectives may be impacted

11.2.26. Awareness: General Security

Overview Administration of relevant security awareness training for staff accessing any environment	**Tools and Systems** • Learning Management System (LMS) • Governance, Risk and Compliance (GRC) tool (If applicable) • IT Service Management (ITSM) Tool
Responsibility **Shared Services Layer** • Identify staff with access to any environment • Validate appropriateness of security training modules and enhance as required • Issue security training modules to staff upon environment On-boarding and based on schedule requirements • Track and enforce completion **Application Teams** • Complete training requirements • Review training completion records	**Processes** • Staff On-boarding • Access Audit
	Key Controls • AT-2 Security Awareness Training • AT-4 Security Training Records
Risks and Caveats • Staff access may be suspended if training not successfully completed within the requisite timeframe. • Corporate Training module may not be company-administered within the requisite timeframe and require out-of-cycle coordination.	**Considerations** • Ability to integrate with Corporate Training (LMS) system for training delivery, completion, and tracking.

11.2.27. Awareness: Role-Based Training

Overview Administration of role-based security training for staff accessing any environment	**Tools and Systems** • Learning Management System (LMS) • Governance, Risk and Compliance (GRC) tool (If applicable) • IT Service Management (ITSM) Tool • Additional training modules
Responsibility **Shared Services Layer** • Establish key security-based roles (i.e. VPC admin, instance admin, power user) commonly leveraged across all products. • Identify staff with access to any environment and their associated roles • Validate adequacy and/or coordinate development of role-based training content • Issue training modules to staff upon environment On-boarding and based on schedule requirements • Track and enforce completion **Application Teams** • Identify roles associated with named individuals with environment access • Develop and administer training content specific to roles within product environment • Review and maintain training completion records.	**Processes** • Staff On-boarding • Access Audit
	Key Controls • AT-3 Role-Based Security Training • AT-4 Security Training Records
Risks and Caveats • Staff access may be suspended if training not successfully completed within the requisite timeframe. • Standard modules issued by Corporate Training do not typically cover role-based training may need to be developed separately – both by Shared Services and Application Teams.	**Considerations** • Ability to integrate with Learning Management System (LMS) system for training delivery, completion, and tracking.

11.2.28. Awareness: Security Intelligence

Overview

Distribution of security intelligence notifications relevant to the threat landscape and technologies implemented within the environment.

Tools and Systems

- Industry Alerts (US CERT, SANS, etc.…)
- Vendor-specific alerts (AWS, Azure etc.…)
- CSP Corporate Email & Distribution Lists

Responsibility

Shared Services Layer

- Identify relevant technologies in use across all products through asset management
- Subscribe and receive associated security alert and vulnerability notifications
- Disseminate notifications to relevant stakeholders
- Determine if additional review and action planning may be warranted

Application Teams

- Validate technology lists for alignment with notifications
- Receive notifications to review for impact and provide responses
- Perform mitigations commensurate with severity risk

Processes

- Asset Management
- Vulnerability Management

Key Controls

- SI-5 Security Alerts, Advisories, and Directives
- SI-2 Flaw Remediation

Risks and Caveats

- Specific application teams may have customers with special data requirements, action upon sensitive data alerts may not be suitable for a shared service ITSM & may require a single-tenant deployment
- Potential for insecure sensitive data dissemination
- Proper data classification requirements

Considerations

- Process to enroll and distribute notifications to stakeholders and customization of intelligence feeds
- Process to ensure all enrolled stakeholders are continually reviewed

11.2.29. Awareness: Insider Threat

Overview Administration of insider threat training for staff accessing any environment	**Tools and Systems** • Learning Management System (LMS) • Governance, Risk and Compliance (GRC) tool (If applicable) • IT Service Management (ITSM) Tool •
Responsibility **Shared Services Layer** • Identify staff with access to any environment • Validate appropriateness of security training modules and enhance as required • Issue security training modules to staff upon environment On-boarding and based on schedule requirements • Track and enforce completion • Report potential indicators of activity to the appropriate channel **Application Teams** • Complete training requirements • Review training completion records • Report potential indicators of activity to the appropriate channel	**Processes** • Staff On-boarding • Access Audit • Incident Reporting
	Key Controls • AT-2 Security Awareness Training • AT-4 Security Training Records
Risks and Caveats • Staff access may be suspended if training not successfully completed within the requisite timeframe. • Corporate Training module may not be company-administered within the requisite timeframe and require out-of-cycle coordination. • Corporate Training modules may not include adequate topic training and may need to be obtained separately.	**Considerations** • Integration with Corporate Training (LMS) system for training delivery, completion, and tracking. • coordination with Corporate Security (GSO) to integrate with company Insider Threat Program.

11.2.30. Architecture: Architecture Review

Overview Inspection of proposed architectures to help ensure the solution aligns with established compliance requirements and best practices	**Tools and Systems** • Architectural Diagramming Tools • ITSM
Responsibility **Shared Services Layer** • Review product architecture and determine impact on authorization boundary • Identify cloud services that are not available/authorized for use in the environment • Provide guidance on how to maintain/obtain compliance • Provide guidance on how to run services in a secure, cost effective manner • Ensure common best practices are designed for the environment • Engage 3rd party resources (advisory or CSP) for additional review and guidance **Application Teams** • Obtain advice on how to design services that are secure, have less risk in obtaining compliance and can be operated securely and cost effectively • Design and implement architecture aligned with compliance requirements	**Processes** • Security Design Life Cycle • Change Control Board • Product On-boarding
	Key Controls • SA-3 System Development Life Cycle • SA-8 Security Engineering Principles • SA-9 External Information System Services • CM-4 Security Impact Analysis • CA-3 System Interconnections
Risks and Caveats • To meet requirements, products may require significant architecture changes due to unavailability of inability to leverage specific CSP services • Data sovereignty risk based on potentially distributed application or shared services data which may span across globally disparate regions	**Considerations** • Data sensitivity requirements (such as DoD) may have specific secrets requirements. Depending on the implementation /system design, secrets may not reside alongside with other secrets (cryptographic erase)

11.2.31. Documentation Management: Secure Internal Data

Overview Establishment of secure documentation repositories to store sensitive data (including ATO documents, evidence artifacts, environment details, runbooks, etc.…) for reference, internal sharing, and collaboration with authorized third parties	**Tools and Systems** • Secure Document Repo • Collaborative Information Management
Responsibility **Shared Services Layer** • Establish central team collaboration sites for environment specific information • Establish document repositories to facilitate secured storage and authorized sharing, both internally and externally • Assign and restrict access based on Shared Services-wide or product-only content • Exchange sensitive documents with authorized 3rd parties and named individuals • Monitor access to documentation repositories **Application Teams** • Access Shared Services-wide knowledge bases and document repositories • Create environment knowledge bases and document repositories containing sensitive information that are only accessible by the product team and Shared Services admins • Exchange sensitive documents with pre-authorized 3rd parties and named individuals • Transfer files into the environment	**Processes** • Staff On-boarding • Access Control
	Key Controls • AC-21 Information Sharing • AC-3 Access Enforcement • AU-9 Protection of Audit Information
Risks and Caveats • Not designed to store customer data to conduct routine product operations • Sensitive files cannot be transferred out of the environment	**Considerations** • Data Loss Prevention (DLP) detection may be inappropriate document sharing or exfiltration

11.2.32. Documentation Management: Critical Documentation Coordination

Overview Management, maintenance, development assistance, and compilation of all mandatory documents that are required to achieve and maintain compliance as part of an authorization package. Required documents include, but not limited to: System Security Plan (SSP), Workbooks, Policies, Plans, and system descriptions.	**Tools and Systems** • Secure Document Repo • Collaborative Information Management • GRC tool • ITSM
Responsibility **Shared Services Layer** • Create and maintain all required documents that constitute the authorization package and scheduled checkpoints • Build documents to maximize inheritance and consistency from shared controls and other documentation sets • Coordinate with Application Teams to update documentation specific to their environment • Schedule documentation reviews on a periodic basis. • Address Continuous Monitoring documentation requirements, to include POAMs **Application Teams** • Provide required information to be captured within documentation sets • Validate documents for accuracy and correct as appropriate • Assert to the fairness, suitability, and operational effectiveness of documents	**Processes** • Change Management • Risk Management Framework (RMF) life cycle program • Review & Maintenance
	Key Controls • AC-21 Information Sharing • AC-3 Access Enforcement
Risks and Caveats • Specific applications may have varying level of data requirements; resulting in Role-Based Access Control (RBAC) implementation not sufficient. Multi-tenancy model for documentation	**Considerations** • Standardized nomenclature for all documentation, enforced policies must be in place to ensure minimal to zero documentation configuration drift

11.2.33. Documentation Management: Policy Management

Overview

Establishment and maintenance of all Information Security Policy addressing all required controls, applicable to all tenants within the environment

Tools and Systems

- Secure Document Repo
- Collaborative Information Management

Responsibility

Shared Services Layer

- Develop all required policies which govern the entire environment including product
- Update and obtain approvals for policy changes as required
- Notify stakeholders and consult on impact of policy changes
- Publish policies for review

Application Teams

- Access, review, and acknowledge policies
- Adhere to and implement policy requirements
- Request clarification on policy requirements

Processes

- Change Management

Key Controls

- All “-1” policy controls in all control families

Risks and Caveats

- Product team adherence to the mandatory policy requirements governing the entire Shared Services. This may require different approaches than established in previous environments (i.e. non-Shared Services). Additional burden on application teams to conform and demonstrate compliance to policy.
- Inability to support individual product policy deviations due to enforcement, maintenance, inheritance, and supportability complexities

Considerations

- Level of alignment with Corporate Services, Shared Services, and Product procedures

11.2.34. Documentation Management: Procedure Management

Overview

Establishment and maintenance of all Information Security Procedure addressing all required controls, applicable to all tenants within the environment

Tools and Systems

- Secure Document Repo
- Collaborative Information Management

Responsibility

Shared Services Layer

- Develop all required procedures for the entire environment including product
- Update and obtain approvals for procedure changes as required
- Notify stakeholders and consult on impact of procedure changes
- Publish procedures for review

Application Teams

- Access, review, and acknowledge procedures
- Adhere to and implement procedure requirements
- Request clarification on procedure requirements

Processes

- Change Management

Key Controls

- All "-1" procedures controls in all control families

Risks and Caveats

- Product team adherence to procedures governing the entire Shared Services. This may require different approaches than established in previous environments (i.e. non-Shared Services). Additional burden on application teams to conform and demonstrate compliance to procedures.
- Inability to support individual product procedural deviations due to enforcement, maintenance, inheritance, and supportability complexities

Considerations

- Level of alignment with Corporate Services, Shared Services, and Product procedures

11.2.35. Documentation Management: Internal Ticketing

Overview Documentation and tracking of important activities, workflows, and sensitive data	**Tools and Systems** • IT Service Management (ITSM) Tool
Responsibility **Shared Services Layer** • Assign and track resolution of support requests initiated by application teams • Capture and track resolution of incidents impacting all environments • Capture, track, and approve (as required) change requests impacting the environment • Assign and track requests for audit artifacts to product stakeholders • Report on all changes and requests • Restrict access based on Shared Services-wide or product-only content **Application Teams** • Document incident details impacting the product environment • Track customer requests containing PII and associate to non-Shared Services redacted requests • Submit change requests and access requests • Monitor and respond to audit requests	**Processes** • Change Management • Incident Response • Staff On-boarding • Access Request
	Key Controls • AC-21 Information Sharing • AC-3 Access Enforcement • AU-9 Protection of Audit Information
Risks and Caveats • Data sensitivity requirements may impact how shared services ITSM tooling may need to be implemented	**Considerations** • Use of internal ticketing system to capture external product customer tickets (entered by CSP staff) as part of support processes. • Change request workflows, integration with CI/CD pipelines • Interaction between GRC tool and IT Service Management (ITSM) Tool to capture audit evidence

11.3. Multi-Cloud Strategies

Before we delve too deep into multi-cloud workloads and shared services, a common doubt we have seen is “Why multi-cloud” and why any organization would purposefully choose a multi-cloud deployment. At the surface, it may sound more rational to remain with a single cloud provider and not shift to another cloud by adding another complexity of layer by augmenting your cloud workload with yet another cloud. However, we will soon discover why an organization may choose a multi-cloud strategy from out the gate.

Usually if you're a small organization, you will not want to think about multi-cloud strategies and instead focus on optimizing your single cloud workload, your account probably won't be shut down, the cost you save optimizing across clouds won't pay for the overhead (initially), etc. However, if the trend of multi-cloud deployments continues, I predict the time will come when multi-cloud becomes valuable.

If the organization plans on 1) being a big company or 2) being acquired (basically anything besides choosing to stay a small private company), organizations should be mentally prepared that multi-cloud is coming for you whether you want it or not.

From a Global 2000 standpoint, most large companies existed prior to the cloud existing (in a real, marketable, business-ready form). These companies are usually coming from an extensive physical footprint, possibly multiple datacenters. So, the companies will begin with adopting a single cloud. Systems are complex and therefore there is at least a multi-year period where these organizations are "multi-cloud" (or hybrid you may say) across cloud + physical environments. The multi-cloud strategy encourages these companies to look for tools that can benefit both (such as Terraform or Vault).

But Global 2000s also acquire companies. Acquisitions are key to their growth strategies. Acquisitions are not usually contingent on what cloud platform you chose, so the dev/ops groups get whatever corporate development brings into the company. Surprise! Your company just acquired a company that is all-in on another cloud platform. You now have a choice: either spend a lot of time/energy migrating those workloads to your systems or spend time/energy on supporting both. In most cases the organizations will choose the latter. At this point, the organizations are now unexpectedly and forcibly become multi-cloud. Companies that have spent time preparing for this take it with ease, no problem; you have built the process and technologies to support any workload. Companies that went all-in on one cloud struggle and have a lot of pain ahead.

From a business standpoint, if you are a big company and acquisitions are not part of your growth strategy (or you are focused on a single cloud). If you are a large company, your IT spend is going to be considerable. In most cases, a very easy $1M+ per year and most larger organizations are spending orders of magnitude more. The significant capital spend motivates vendors. If you pay Cloud A $500K per month, Cloud B will send some suits knocking on your door offering you the same resources for $400K per month guaranteed for 3 years. Cloud C is going to just give you millions of dollars in credit to "try" their platform. Clouds know once you have workloads on their systems, you usually do not move off too easily. From a top-down executive perspective, it is hard to say no to this.

Back to the technology aspect, large companies run a lot of software and that software may have specific requirements. The most common case for multi-cloud we see early on is "we're 90% cloud A but 10% cloud B because cloud B software runs better there." The most obvious example: Active Directory. AD is easily the most common onramp onto Azure we see, it is so easy to run AD on Azure (relatively) and almost all large companies are built on AD systems.

Another technical choice: better high-level services. Certain clouds have much better high-level services than others regarding data processing, machine learning, etc. So sometimes specific teams (for example teams building ML models) may be motivated to use a certain cloud even if the built model will be run on a different cloud. This comes back to the question of: are you going to force all your dev teams to use your one true cloud? Or are you going to let them run their dev workloads (at least, if not prod!) on others? If the latter, how are you going to do access control, resource management, budgeting, etc.? The conversation opens a big can of worms that pushes the organization down the path of multi-cloud processes and tooling, again, even if its non-production.

A common justification for a multi-cloud deployment is also "vendor lock-in". “Vendor lock-in” is real. But it is not as huge of a thing as people claim. We work with an organization which is 99% on one cloud. The organization has a full plan (technical to human) to migrating to a specific 2nd cloud in approx. 6 months. Did the organization plan to? Not at all. But the organization will execute for two reasons. One, if the organization acquires a company (they have not yet), they can support multi-cloud since their processes are built around it. Two, if their current cloud starts hard negotiating their reserved pricing, they have leverage, they can move.

And finally, just a real quick point: a common confusion is multi-cloud vs. multi-vendor services. The latter is way more common, especially at smaller companies. Tools such as Terraform are often touted as "multi-cloud" and people ask questions like "Why would I use Terraform if I use only AWS?" And the easy answer to that is tools such as Terraform allow you to manage anything with an API as code. For example: do you want to manage DNS, or CDN, or DBs, etc. (that maybe are not on your current Cloud Service Platform) as code? Terraform is a mechanism/tool which provides you the ability to learn one config language/workflow to make that happen, even if 100% of your compute is on one provider. From a non-technical standpoint, this helps your organization start learning non-vendor-specific tooling, which better prepares you from a human standpoint for the future noted above.

We believe the #1 value of multi-cloud is organizational: you build your core infra/app lifecycle processes (dev, build, deploy, monitor, etc.) around a technology-agnostic stance. As technologies shift, other clouds become important, new paradigms emerge, etc. your organization is likely more prepared to experience that change.

- This space intentionally left blank -

11.4. Hybrid/Cross Cloud Shared Services

Which brings us to an even more complex implementation of the shared services model. Which is a hybrid multi-cloud deployment leveraging existing shared services from usually either an existing on-premises due to growth or another IaaS. The multi-cloud deployment is usually leveraged for various reasons, such as (but not limited to), the CSP wanting to either offer their system/service on multiple platforms, leverage the multiple different types of services offered through the different IaaS cloud providers and or simply have an adapted hybrid model due to cloud adoption and growth.

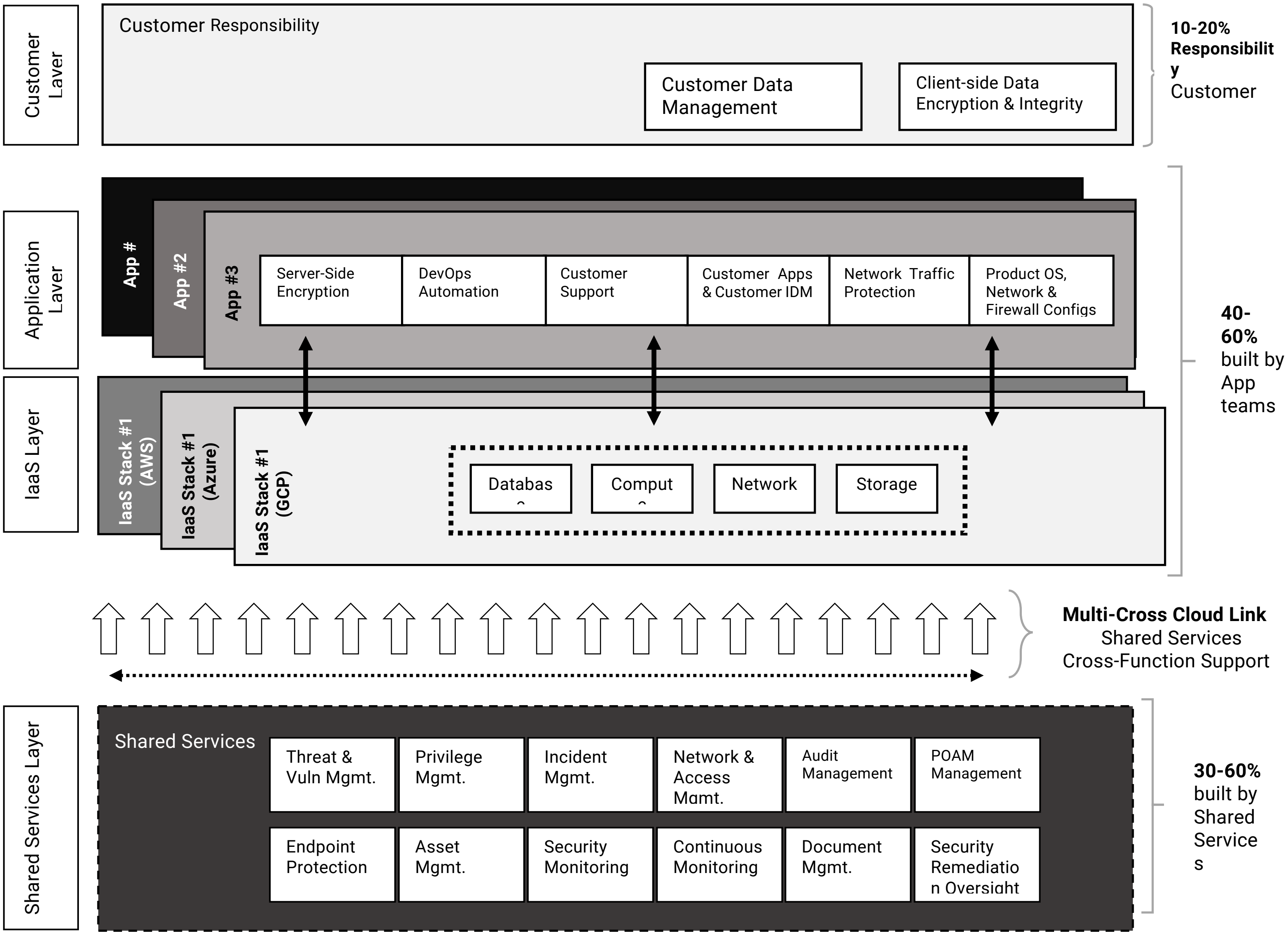

Figure 11 Multi-Cloud Shared Services

Within the multi-cloud Shared Services Model, the Shared Services layer is most commonly swapped with the IaaS layer from the previous model. This results in the shared services residing within either a preferred vendor CSP IaaS or existing on-premises system. In most cases, multi-cloud shared services deployments are leveraged by larger organizations as they usually have significant existing physical footprint, as depicted in *Figure 11 Multi-Cloud Shared Services*. The important takeaway is that regardless of how the shared services model is implemented, it is critical to understand the varying responsibilities between the different layers of the entire cloud fabric.

12. Conclusion

We started with the basic fundamentals with abstraction from the Instruction Set Architecture (ISA) all the way up the technology stack to the present-day cloud. We discussed the issues and challenges which arise from the use of outdated terminology such as ‘common controls’ and emphasized that full inheritance is only applicable in only very clearly defined areas of responsibility. We followed the discussion to highlight the shortcomings of the current state of most Customer Responsibility Matrices and the lack of thought and critical analysis given in identifying responsibilities. We proposed a framework for shared services within the cloud for architects and the industry to consider. The framework for Shared Services is not new; however, I believe the emphasis on the further clarifying customer responsibility is. I hope the proposed framework at a minimum, highlights the nuance which is involved in properly implementing a fully defined shared services model within the cloud, with responsibilities fully flushed out.

13. Glossary

1. **AD** – Active Directory
2. **AKS** - Azure Kubernetes Service
3. **API** – Application Programming Interface
4. **AWS –** Amazon Web Services
5. **CAC** – Common Access Card
6. **CDN** - Content Deliver Services
7. **CI** - Configuration Item
8. **CI/CD** – Continuous Integration / Continuous Deployment
9. **CIS** – Center for Internet Security
10. **CMMC** – Cybersecurity Maturity Model Certification
11. **CNCF** - Cloud Native Computing Foundation
12. **CPU** – Central Processing Unit
13. **CRM** – Customer Responsibility Matrix
14. **CSO** - Cloud Service Offering
15. **CSP** – Cloud Service Provider
16. **DB** – Database
17. **DISA** – Defense Information Systems Agency
18. **DNS** – Domain Name System
19. **ECS** - Elastic Container Service
20. **EKS** - Elastic Container Service For Kubernetes
21. **FedRAMP** – Federal Risk and Authorization Management Program
22. **FICAM** – Federal Identity, Credential, and Access Management
23. **GCP** – Google Cloud Platform
24. **GRC** – Governance, Risk & Compliance
25. **HIDS** – Host-Based Intrusion Detection System
26. **HIPAA** – Health Insurance Portability and Accountability Act
27. **HIPS** – Host-Based Intrusion Prevention System
28. **HR** – Human Resources
29. **HTTP** - Hyper Transport Text Protocol
30. **IDM** – Identification Management
31. **IR** – Incident Response
32. **ISA** – Instruction Set Architecture
33. **IT** - Information Technology
34. **ITSM** – Information Technology Service Management
35. **MFA** – Multi-factor Authentication
36. **ML** – Machine Learning
37. **MS** – Microsoft
38. **NIST** – National Institute of Standards and Technology
39. **OS** - Operating System

40. **OSI** – Open System Interconnection Model
41. **PDU** - Protocol Data Unit
42. **PII** – Personally Identifiable Information
43. **PIV** – Personal Identity Verification
44. **POA&M/POAM** – Plans of Actions & Milestones
45. **RBAC** – Role Based Access Control
46. **SAML** – Security Assertion Markup Language
47. **SDLC** – Software/System/Security Development Lifecycle
48. **SIEM** – Security Information & Events Management
49. **STIG** – Security Technical Implementation Guide
50. **TCP/IP** - Transmission Control Protocol / Internet Protocol
51. **UTM** – Unified Threat Management
52. **VM** - Virtual Machine
53. **VPC** – Virtual Private Cloud
54. **WAF** – Web Application Firewall

14. References

[1] AWS Architecture Blog, Compute Abstraction on AWS: A Visual Story, Massimo Re Ferre, September 2018

[2] What is Serverless? A Definition, Srinath Perera, December 2018

[3] Amazon Web Services, Shared Responsibility Model, May 2020

[4] Rethinking Common Controls for Cloud-Based Federal Information Systems, Sarbari Gupta, July 2019

[5] Why Multicloud, Mitchell Hashimoto, HashiCorp,

[6] Gartner: Is the Cloud Secure, Kasey Panetta

[7] Center for Internet Security: Shared Responsibility for Cloud Security: What You Need to Know

[8] National Institute of Standards and Technology, Security and Privacy Controls for Federal Information Systems and Organizations

15. Contact

Bhanu Jagasia
571.269.4272
bjagasia@bladestack.io

Our Mission

When it comes to securing your businesses’ future, leading cloud infrastructure providers, SaaS providers, and enterprises turn to the cyber-samurais at bladestack.io. We are the cybersecurity advisor that combines extensive cloud expertise, technology, and innovative approaches to empower our clients to use security and compliance to their advantage.

Our Vision

To become the most absurdly technical global cyber leader in Information Security, compliance, and risk management services through simplifying cybersecurity for our customers

bladestack.io

📍 7902 Tysons One Place, Suite 2001
McLean, VA

☎ t1. 571.269.4272
t2. 540.326.6929

✉ info@bladestack.io

🌐 www.bladestack.io