SECURITY GUIDANCE

For Critical Areas of Focus In Cloud Computing v4.0

cloud security alliance®

The permanent and official location for Cloud Security Alliance's *Security Guidance for Critical Areas of Focus in Cloud Computing v4.0* is **https://cloudsecurityalliance.org/download/security-guidance-v4/**.

Official Study Guide for the CCSK™ Certificate of Cloud Security Knowledge

FOREWORD

Welcome to the fourth version of the Cloud Security Alliance's *Security Guidance for Critical Areas of Focus in Cloud Computing*. The rise of cloud computing as an ever-evolving technology brings with it a number of opportunities and challenges. With this document, we aim to provide both guidance and inspiration to support business goals while managing and mitigating the risks associated with the adoption of cloud computing technology.

The Cloud Security Alliance promotes implementing best practices for providing security assurance within the domain of cloud computing and has delivered a practical, actionable roadmap for organizations seeking to adopt the cloud paradigm. The fourth version of the *Security Guidance for Critical Areas of Focus in Cloud Computing* is built on previous iterations of the security guidance, dedicated research, and public participation from the Cloud Security Alliance members, working groups, and the industry experts within our community. This version incorporates advances in cloud, security, and supporting technologies; reflects on real-world cloud security practices; integrates the latest Cloud Security Alliance research projects; and offers guidance for related technologies.

The advancement toward secure cloud computing requires active participation from a broad set of globally-distributed stakeholders. CSA brings together this diverse community of industry partnerships, international chapters, working groups, and individuals. We are profoundly grateful to all who contributed to this release.

Please visit **cloudsecurityalliance.com** to learn how you can work with us to identify and promote best practices to ensure a secure cloud computing environment.

Best regards,

Luciano (J.R.) Santos
Executive Vice President of Research
Cloud Security Alliance

ACKNOWLEDGEMENTS

Lead Authors

Rich Mogull
James Arlen
Francoise Gilbert
Adrian Lane
David Mortman
Gunnar Peterson
Mike Rothman

Editors

John Moltz
Dan Moren
Evan Scoboria

CSA Staff

Jim Reavis
Luciano (J.R.) Santos
Hillary Baron
Ryan Bergsma
Daniele Catteddu
Victor Chin
Frank Guanco
Stephen Lumpe (Design)
John Yeoh

Contributors

On behalf of the CSA Board of Directors and the CSA Executive Team, we would like to thank all of the individuals who contributed time and feedback to this version of the CSA Security Guidance for Critical Areas of Focus in Cloud Computing. We value your volunteer contributions and believe that the devotion of volunteers like you will continue to lead the Cloud Security Alliance into the future.

LETTER FROM THE **CEO**

I am thrilled by this latest contribution to the community's knowledge base of cloud security best practices that began with Cloud Security Alliance's initial guidance document released in April of 2009. We hope that you will carefully study the issues and recommendations outlined here, compare with your own experiences and provide us with your feedback. A big thank you goes out to all who participated in this research.

Recently, I had the opportunity to spend a day with one of the industry experts who helped found Cloud Security Alliance. He reflected that for the most part CSA has completed its initial mission, which was to prove that cloud computing could be made secure and to provide the necessary tools to that end. Not only did CSA help make cloud computing a credible secure option for information technology, but today cloud computing has become the default choice for IT and is remaking the modern business world in very profound ways.

The resounding success of cloud computing and CSA's role in leading the trusted cloud ecosystem brings with it even greater challenges and urgency into our renewed mission. Cloud is now becoming the back end for all forms of computing, including the ubiquitous Internet of Things. Cloud computing is the foundation for the information security industry. New ways of organizing compute, such as containerization and DevOps are inseparable from cloud and accelerating our revolution. At Cloud Security Alliance, we are committed to providing you the essential security knowledge you need for this fast moving IT landscape and staying at the forefront of next-generation assurance and trust trends. We welcome your participation in our community, always.

Best regards,

Jim Reavis
Co-Founder & CEO
Cloud Security Alliance

TABLE OF **CONTENTS**

DOMAIN 1
Cloud Computing Concepts and Architectures

1.0 Introduction

This domain provides the conceptual framework for the rest of the Cloud Security Alliance's guidance. It describes and defines cloud computing, sets our baseline terminology, and details the overall logical and architectural frameworks used in the rest of the document.

There are many different ways of viewing cloud computing: It's a technology, a collection of technologies, an operational model, a business model, just to name a few. It is, at its essence, *transformative* and *disruptive*. It's also growing very, very quickly, and shows no signs of slowing down. While the reference models we included in the first version of this Guidance are still relatively accurate, they are most certainly no longer complete. And even this update can't possibly account for every possible evolution in the coming years.

Cloud computing offers tremendous potential benefits in *agility*, *resiliency*, and *economy*. Organizations can move faster (since they don't have to purchase and provision hardware, and everything is software defined), reduce downtime (thanks to inherent elasticity and other cloud characteristics), and save money (due to reduced capital expenses and better demand and capacity matching). We also see *security* benefits since cloud providers have significant economic incentives to protect customers.

However, these benefits only appear if you understand and adopt *cloud-native* models and adjust your architectures and controls to align with the features and capabilities of cloud platforms. In fact, taking an existing application or asset and simply moving it to a cloud provider without any changes will often reduce agility, resiliency, and even security, all while increasing costs.

The goal of this domain is to build the foundation that the rest of the document and its recommendations are based on. The intent is to provide a common language and understanding of cloud computing for security professionals, begin highlighting the differences between cloud and traditional computing, and help guide security professionals towards adopting cloud-native approaches that result in better security (and those other benefits), instead of creating more risks.

This domain includes 4 sections:

- Defining cloud computing
- The cloud logical model
- Cloud conceptual, architectural, and reference model
- Cloud security and compliance scope, responsibilities, and models

The Cloud Security Alliance isn't setting out to create an entirely new taxonomy or reference model. Our objective is to distill and harmonize existing models—most notably the work in **NIST Special Publication 800-145**, **ISO/IEC 17788 and ISO/IEC 17789**—and focus on what's most relevant to security professionals.

1.1 Overview

1.1.1 Defining Cloud Computing

Cloud computing is a new operational model and set of technologies for managing shared pools of computing resources.

It is a disruptive technology that has the potential to enhance collaboration, agility, scaling, and availability, as well as providing the opportunities for cost reduction through optimized and efficient computing. The cloud model envisages a world where components can be rapidly orchestrated, provisioned, implemented and decommissioned, and scaled up or down to provide an on-demand utility-like model of allocation and consumption.

NIST defines cloud computing as:

> Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

The ISO/IEC definition is very similar:

> Paradigm for enabling network access to a scalable and elastic pool of shareable physical or virtual resources with self-service provisioning and administration on-demand.

A (slightly) simpler way of describing cloud is that it takes a set of resources, such as processors and memory, and puts them into a big pool (in this case, using virtualization). Consumers ask for what they need out of the pool, such as 8 CPUs and 16 GB of memory, and the cloud assigns those resources to the client, who then connects to and uses them over the network. When the client is done, they can release the resources back into the pool for someone else to use.

A cloud can consist of nearly any computing resources, ranging from our "compute" examples of processors and memory to networks, storage, and higher level resources like databases and

applications. For example, subscribing to a customer-relations management application for 500 employees on a service shared by hundreds of other organizations is just as much cloud computing as launching 100 remote servers on a compute cloud.

> Definition: A *cloud user* is the person or organization requesting and using the resources, and the *cloud provider* is the person or organization who delivers it. We also sometimes use the terms *client* and *consumer* to refer to the cloud user, and *service* or simply *cloud* to describe the provider. **NIST 500-292** uses the term "cloud actor" and adds roles for cloud brokers, carriers, and auditors. ISO/IEC 17788 uses the terms cloud service customer, cloud service partner, and cloud service provider.

The key techniques to create a cloud are abstraction and orchestration. We abstract the resources from the underlying physical infrastructure to create our pools, and use orchestration (and automation) to coordinate carving out and delivering a set of resources from the pools to the consumers. As you will see, these two techniques create all the essential characteristics we use to define something as a "cloud."

This is the difference between cloud computing and traditional virtualization; virtualization abstracts resources, but it typically lacks the orchestration to pool them together and deliver them to customers on demand, instead relying on manual processes.

Clouds are *multitenant* by nature. Multiple different consumer constituencies share the same pool of resources but are *segregated* and *isolated* from each other. Segregation allows the cloud provider to divvy up resources to the different groups, and isolation ensures they can't see or modify each other's assets. Multitenancy doesn't only apply across different organizations; it's also used to divvy up resources between different units in a single business or organization.

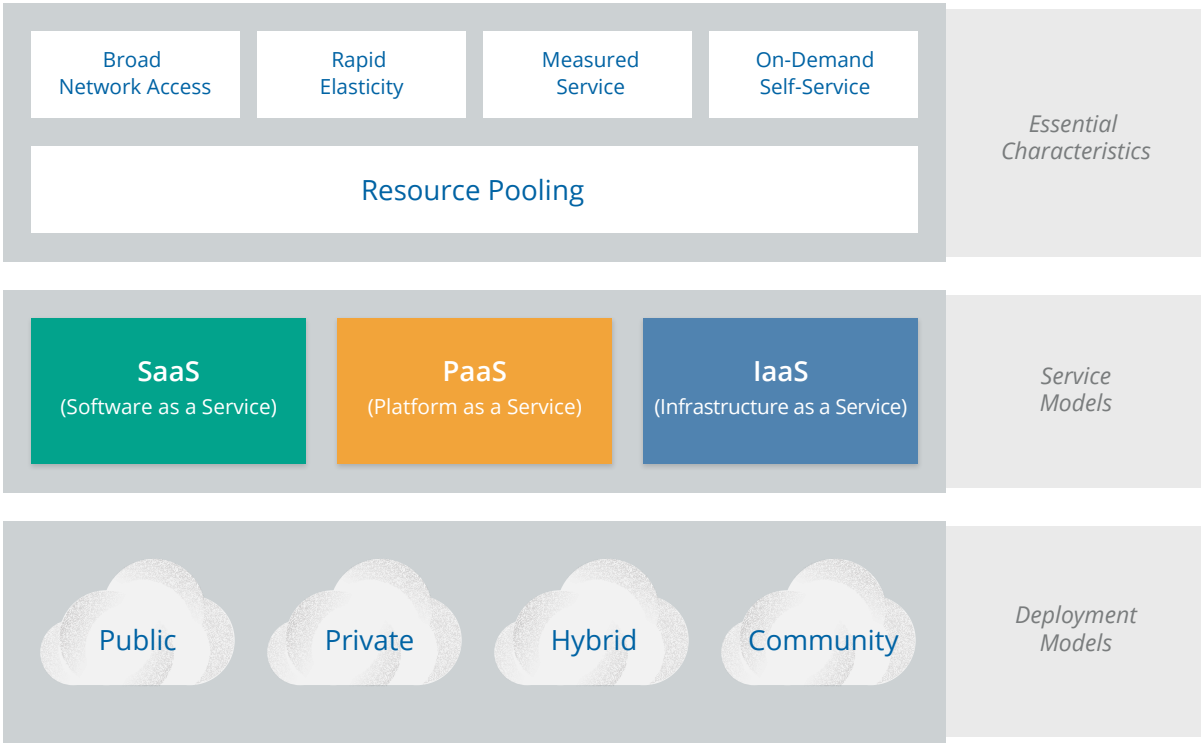
1.1.2 Definitional Model

The Cloud Security Alliance (CSA) uses the **NIST model for cloud computing** as its standard for defining cloud computing. The CSA also endorses the **ISO/IEC model** which is more in-depth, and additionally serves as a reference model. Throughout this domain we will reference both.

NIST's publication is generally well accepted, and the Guidance aligns with the NIST Working Definition of Cloud Computing (NIST 800-145) to bring coherence and consensus around a common language to focus on use cases rather than semantic nuances.

> It is important to note that this guide is intended to be broadly usable and applicable to organizations globally. While NIST is a U.S. government organization, the selection of this reference model should not be interpreted to suggest the exclusion of other points of view or geographies.

NIST defines cloud computing by describing five essential characteristics, three cloud service models, and four cloud deployment models. They are summarized in visual form and explained in detail on the following page.



1.1.2.1 Essential Characteristics

These are the characteristics that make a cloud a cloud. If something has these characteristics, we consider it cloud computing. If it lacks any of them, it is likely not a cloud.

- *Resource pooling* is the most fundamental characteristic, as discussed above. The provider abstracts resources and collects them into a pool, portions of which can be allocated to different consumers (typically based on policies).
- Consumers provision the resources from the pool using *on-demand self-service*. They manage their resources themselves, without having to talk to a human administrator.
- *Broad network access* means that all resources are available over a network, without any need for direct physical access; the network is not necessarily part of the service.
- *Rapid elasticity* allows consumers to expand or contract the resources they use from the pool (provisioning and deprovisioning), often completely automatically. This allows them to more closely match resource consumption with demand (for example, adding virtual servers as demand increases, then shutting them down when demand drops).
- *Measured service* meters what is provided, to ensure that consumers only use what they are allotted, and, if necessary, to charge them for it. This is where the term *utility computing* comes from, since computing resources can now be consumed like water and electricity, with the client only paying for what they use.

ISO/IEC 17788 lists six key characteristics, the first five of which are identical to the NIST characteristics. The only addition is *multitenancy*, which is distinct from resource pooling.

1.1.2.2 Service Models

NIST defines three *service models* which describe the different foundational categories of cloud services:

- *Software as a Service (SaaS)* is a full application that's managed and hosted by the provider. Consumers access it with a web browser, mobile app, or a lightweight client app.
- *Platform as a Service (PaaS)* abstracts and provides development or application platforms, such as databases, application platforms (e.g. a place to run Python, PHP, or other code), file storage and collaboration, or even proprietary application processing (such as machine learning, big data processing, or direct Application Programming Interfaces (API) access to features of a full SaaS application). The key differentiator is that, with PaaS, you don't manage the underlying servers, networks, or other infrastructure.
- *Infrastructure as a Service (IaaS)* offers access to a resource pool of fundamental computing infrastructure, such as compute, network, or storage.

We sometimes call these the "SPI" tiers.

ISO/IEC uses a more complex definition with a *cloud capabilities type* that maps closely to the SPI tiers (application, infrastructure, and platform capability types). It then expands into *cloud service categories* that are more granular, such as Compute as a Service, Data Storage as a Service, and then even includes IaaS/PaaS/SaaS.

These categories are somewhat permeable: Some cloud services span these tiers, others don't fall neatly into a single service model. Practically speaking, there's no reason to try and assign everything into these three broad categories, or even the more granular categories in the ISO/IEC model. This is merely a useful descriptive tool, not a rigid framework.

Both approaches are equally valid, but since the NIST model is more concise and currently used more broadly, it is the definition predominantly used in CSA research.

1.1.2.3 Deployment Models

Both NIST and ISO/IEC use the same four cloud deployment models. These are how the technologies are deployed and consumed, and they apply across the entire range of service models:

- *Public Cloud.* The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.
- *Private Cloud.* The cloud infrastructure is operated solely for a single organization. It may be managed by the organization or by a third party and may be located on-premises or off-premises.
- *Community Cloud.* The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g. mission, security requirements, policy, or compliance considerations). It may be managed by the organizations or by a third party and may be located on-premises or off-premises.
- *Hybrid Cloud.* The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or

proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds). Hybrid is also commonly used to describe a non-cloud data center bridged directly to a cloud provider.

Deployment models are defined based on the cloud user—that is, who uses the cloud. As the diagram below shows, the organization that owns and manages the cloud will vary even within a single deployment model.

	Infrastructure Managed By[1]	Infrastructure Owned By[2]	Infrastructure Located[3]	Accessible and Consumed By[4]
Public	Third-Party Provider	Third-Party Provider	Off-Premises	Untrusted
Private/ Community	Organization / Third-Party Provider	Organization / Third-Party Provider	On-Premises / Off-Premises	Trusted
Hybrid	Both Organization & Third-Party Provider	Both Organization & Third-Party Provider	Both On-Premises & Off-Premises	Trusted & Untrusted

[1] Management includes: governance, operations, security, compliance, etc...

[2] Infrastructure implies physical infrastructure such as facilities, compute network and storage equipment

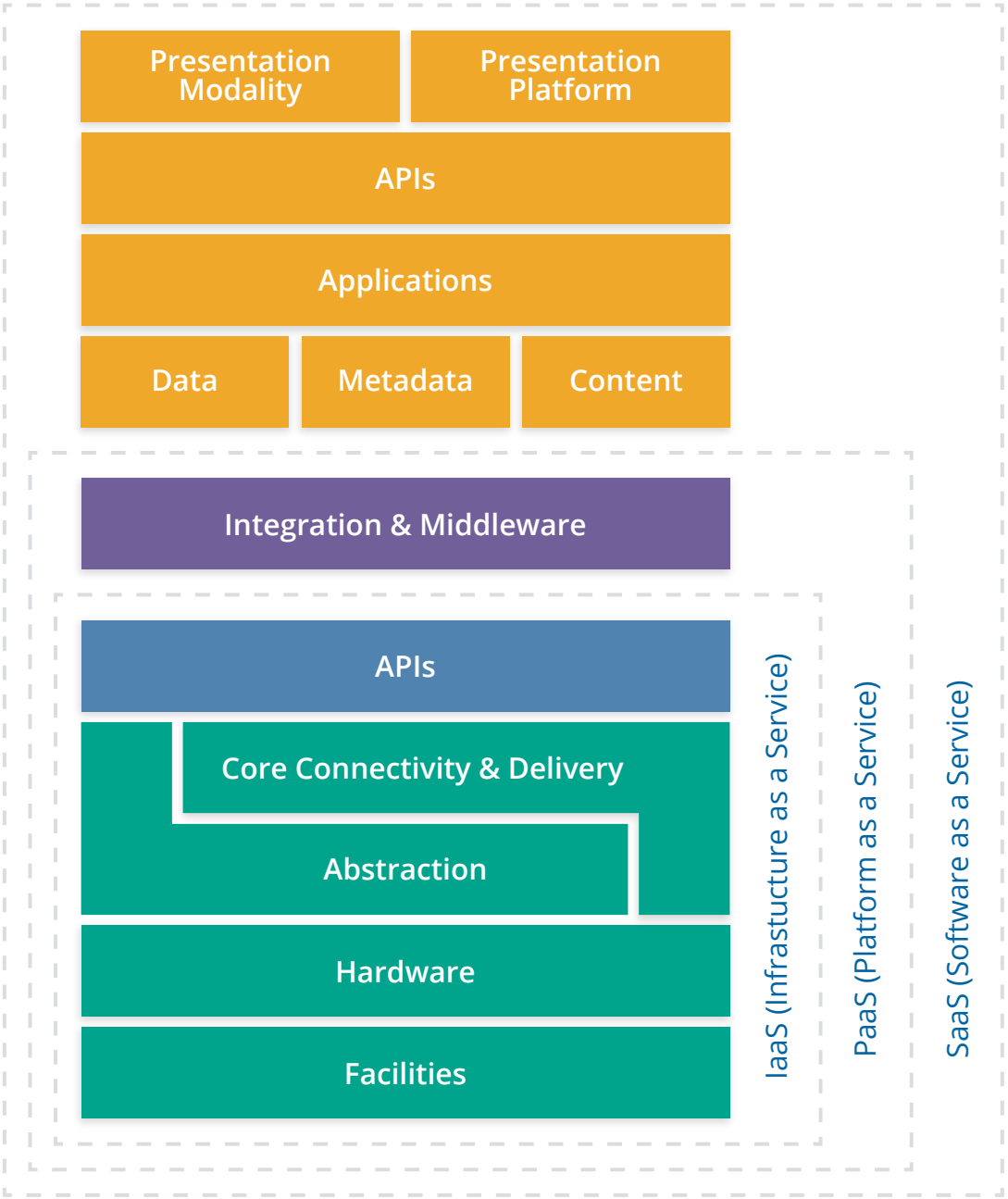
[3] Infrastructure location is both physical relative to an organization's management umbrella and speaks to ownership versus control

[4] Trusted consumers of service are those who are considered part of an organization's legal/contractual/ policy umbrella including employees, contractors, and business partners. Untrusted consumers are those that may be authorized to consume some/all services but are not logical extensions of the organization.

1.1.3 Reference and Architecture Models

These days there is a wide range of constantly evolving technological techniques for building cloud services, making any single reference or architectural model obsolete from the start. The objective of this section is to provide both some fundamentals to help security professionals make informed decisions as well as a baseline to understand more complex and emerging models. For an in-depth reference architectural model, we again recommend **ISO/IEC 17789** and **NIST 500-292**, which complement the NIST definition model.

One way of looking at cloud computing is as a stack where Software as a Service is built on Platform as a Service, which is built on Infrastructure as a Service. This is not representative of all (or even most) real-world deployments, but serves as a useful reference to start the discussion.



1.1.3.1 Infrastructure as a Service

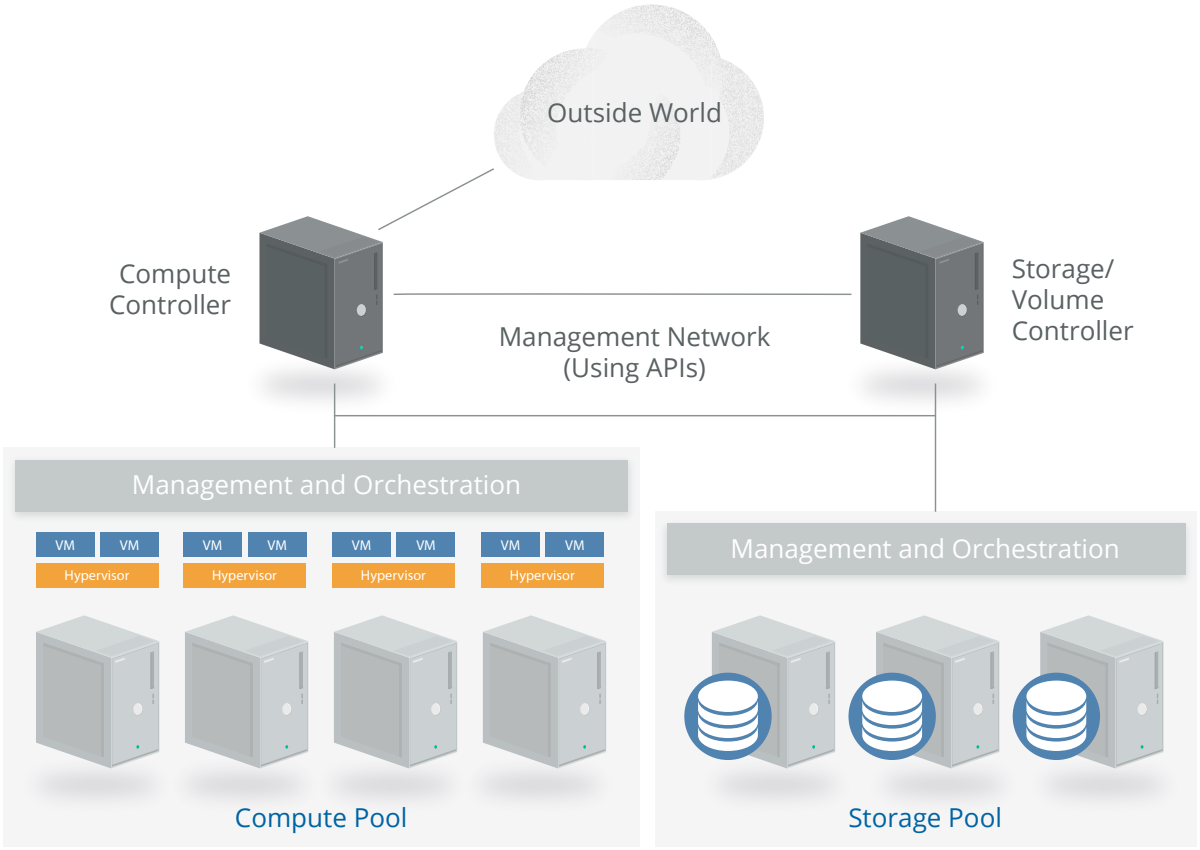
Physical facilities and infrastructure hardware form the foundation of IaaS. With cloud computing we abstract and pool these resources, but at the most basic level we always need physical hardware, networks, and storage to build on. These resources are pooled using abstraction and orchestration. Abstraction, often via virtualization, frees the resources from their physical constraints to enable pooling. Then a set of core connectivity and delivery tools (orchestration) ties these abstracted resources together, creates the pools, and provides the automation to deliver them to customers.

All this is facilitated using *Application Programming Interfaces*. APIs are typically the underlying communications method for components within a cloud, some of which (or an entirely different set) are exposed to the cloud user to manage their resources and configurations. Most cloud APIs these days use REST (Representational State Transfer), which runs over the HTTP protocol, making it extremely well suited for Internet services.

In most cases, those APIs are both remotely accessible and wrapped into a web-based user interface. This combination is the *cloud management plane*, since consumers use it to manage and configure the cloud resources, such as launching virtual machines (instances) or configuring virtual networks. From a security perspective, it is both the biggest difference from protecting physical infrastructure (since you can't rely on physical access as a control) and the top priority when designing a cloud security program. If an attacker gets into your management plane, they potentially have full remote access to your entire cloud deployment.

Thus IaaS consists of a facility, hardware, an abstraction layer, an orchestration (core connectivity and delivery) layer to tie together the abstracted resources, and APIs to remotely manage the resources and deliver them to consumers.

Here is a simplified architectural example of a compute IaaS platform:



This is a very simple diagram showing the compute and storage controllers for orchestration, hypervisors for abstraction, and the relationship between the compute and storage pools. It omits many components, such as the network manager.

A series of physical servers each run two components: a hypervisor (for virtualization) and the management/orchestration software to tie in the servers and connect them to the compute controller. A customer asks for an instance (virtual server) of a particular size and the cloud controller determines which server has the capacity and allocates an instance of the requested size.

The controller then creates a virtual hard drive by requesting storage from the storage controller, which allocates storage from the storage pool, and connects it to the appropriate host server and instance over the network (a dedicated network for storage traffic). Networking, including virtual network interfaces and addresses, is also allocated and connected to the necessary virtual network.

The controller then sends a copy of the server image into the virtual machine, boots it, and configures it; this creates an instance running in a virtual machine (VM), with virtual networking and storage all properly configured. Once this entire process is complete, the metadata and connectivity information is brokered by the cloud controller and available to the consumer, who can now connect to the instance and log in.

1.1.3.2 Platform as a Service

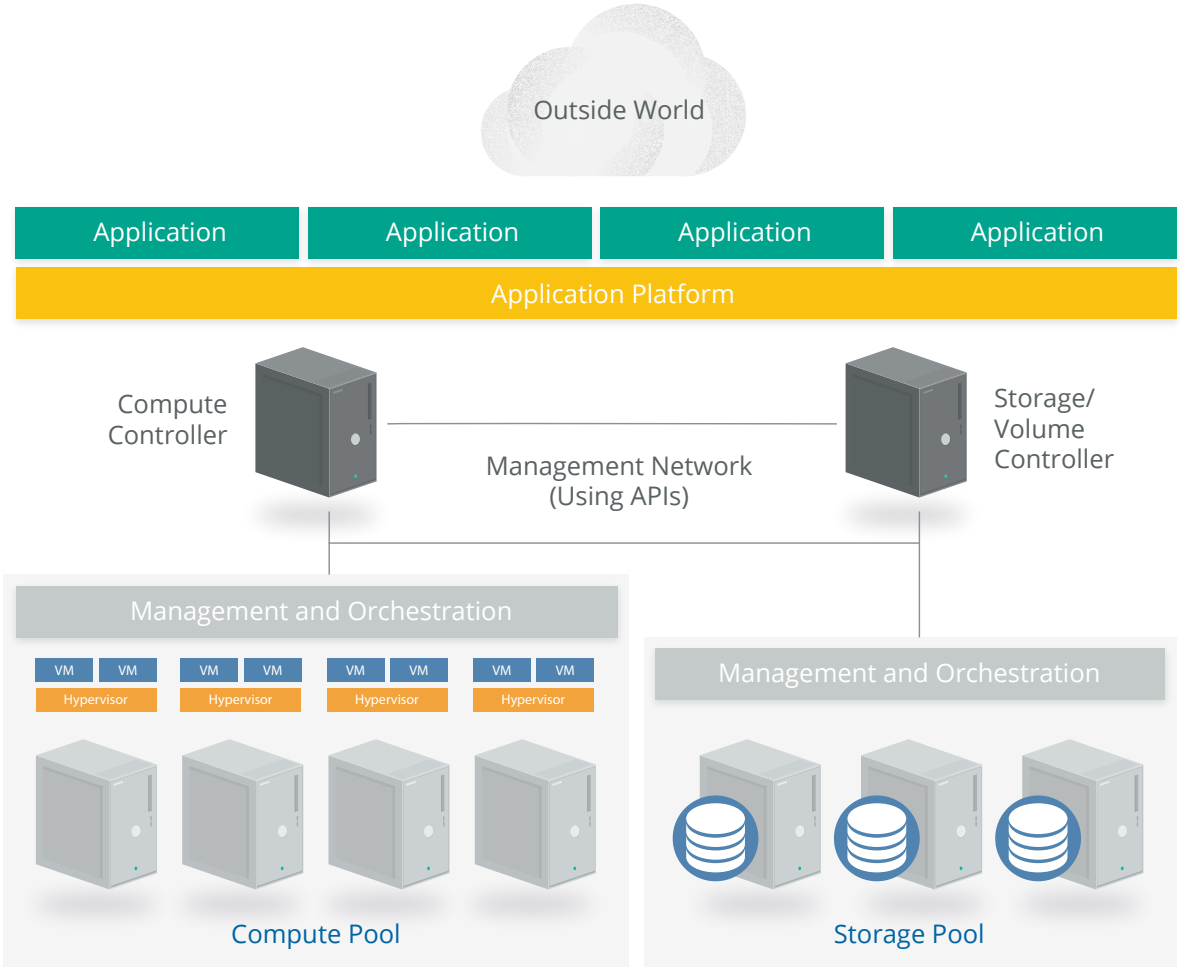
Of all the service models, **PaaS** is the hardest to definitively characterize due to both the wide range of PaaS offerings and the many ways of building PaaS services. PaaS adds an additional layer of integration with application development frameworks, middleware capabilities, and functions such as databases, messaging, and queuing. These services allow developers to build applications on the platform with programming languages and tools that are supported by the stack.

One option, frequently seen in the real world and illustrated in our model, is to build a platform on top of IaaS. A layer of integration and middleware is built on IaaS, then pooled together, orchestrated, and exposed to customers using APIs as PaaS. For example, a Database as a Service could be built by deploying modified database management system software on instances running in IaaS. The customer manages the database via API (and a web console) and accesses it either through the normal database network protocols, or, again, via API.

In PaaS the cloud user only sees the platform, not the underlying infrastructure. In our example, the database expands (or contracts) as needed based on utilization, without the customer having to manage individual servers, networking, patches, etc.

Another example is an application deployment platform. That's a place where developers can load and run application code without managing the underlying resources. Services exist for running nearly any kind of application in any language on PaaS, freeing the developers from configuring and building servers, keeping them up to date, or worrying about complexities like clustering and load balancing.

This simplified architecture diagram shows an application platform (PaaS) running on top of our IaaS architecture:



PaaS doesn't necessarily need to be built on top of IaaS; there is no reason it cannot be a custom-designed stand-alone architecture. The defining characteristic is that consumers access and manage the platform, not the underlying infrastructure (including cloud infrastructure).

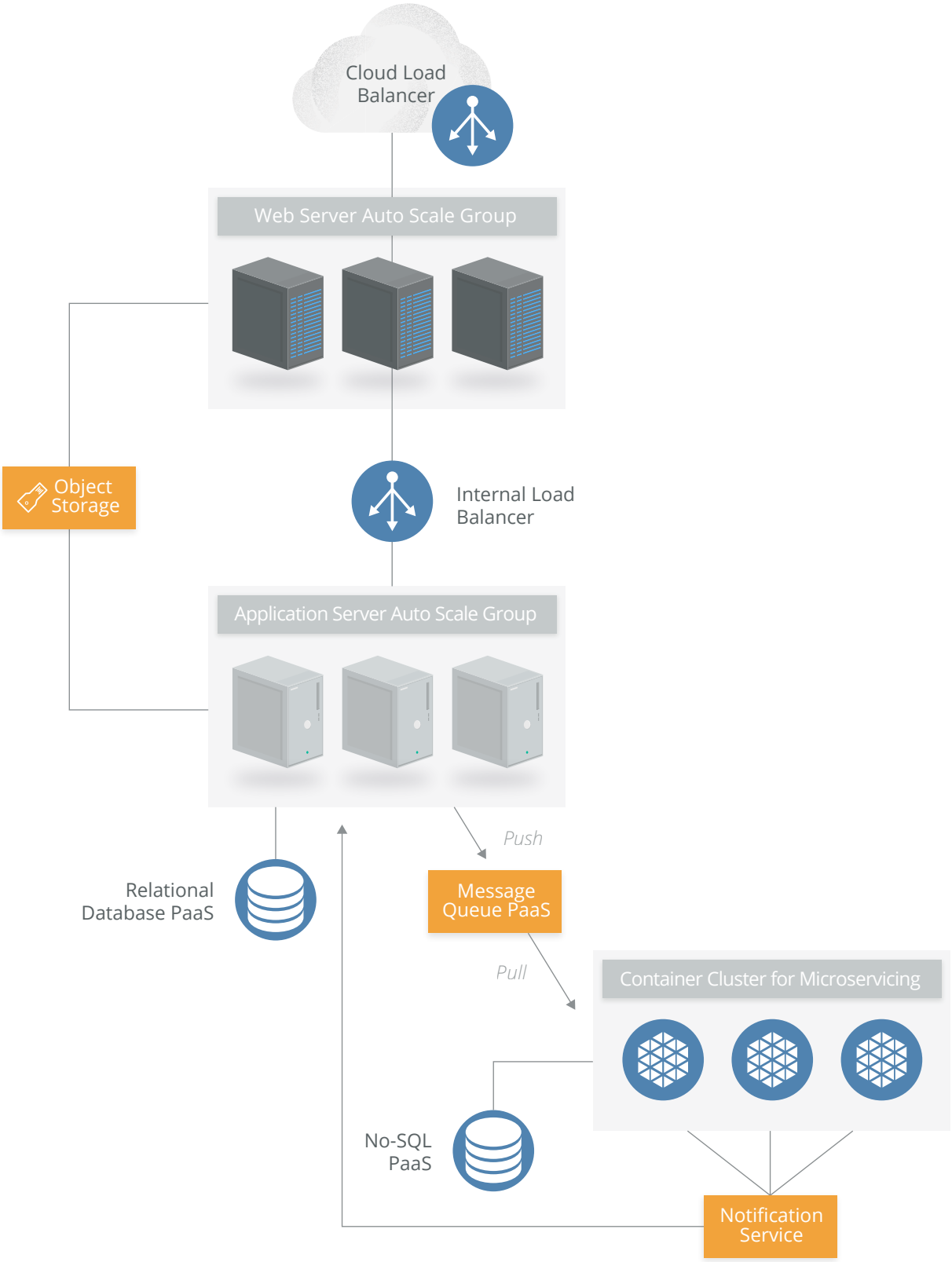
1.1.3.3 Software as a Service

SaaS services are full, multitenant applications, with all the architectural complexities of any large software platform. Many SaaS providers build on top of IaaS and PaaS due to the increased agility, resilience, and (potential) economic benefits.

Most modern cloud applications (SaaS or otherwise) use a combination of IaaS and PaaS, sometimes across different cloud providers. Many also tend to offer public APIs for some (or all) functionality. They often need these to support a variety of clients, especially web browsers and mobile applications.

Thus all SaaS tends to have an application/logic layer and data storage, with an API on top. Then there are one or more presentation layers, often including web browsers, mobile applications, and public API access.

The simplified architecture diagram below is taken from a real SaaS platform, but generalized to remove references to the specific products in use:

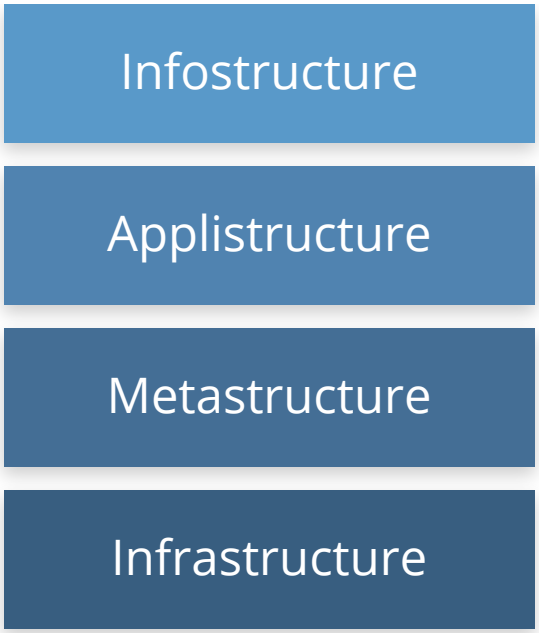


1.1.4 Logical Model

At a high level, both cloud and traditional computing adhere to a logical model that helps identify different layers based on functionality. This is useful to illustrate the differences between the different computing models themselves:

- *Infrastructure:* The core components of a computing system: compute, network, and storage. The foundation that everything else is built on. The moving parts.
- *Metastructure:* The protocols and mechanisms that provide the interface between the infrastructure layer and the other layers. The glue that ties the technologies and enables management and configuration.
- *Infostructure:* The data and information. Content in a database, file storage, etc.
- *Applistructure:* The applications deployed in the cloud and the underlying application services used to build them. For example, Platform as a Service features like message queues, artificial intelligence analysis, or notification services.

Different security focuses map to the different logical layers. Application security maps to applistructure, data security to infostructure, and infrastructure security to infrastructure.



The key difference between cloud and traditional computing is the metastructure. Cloud metastructure includes the management plane components, which are network-enabled and remotely accessible. Another key difference is that, in cloud, you tend to double up on each layer. Infrastructure, for example, includes both the infrastructure used to create the cloud as well as the virtual infrastructure used and managed by the cloud user. In private cloud, the same organization might need to manage both; in public cloud the provider manages the physical infrastructure while the consumer manages their portion of the virtual infrastructure.

As we will discuss later this has profound implications on who is responsible for, and manages, security.

These layers tend to map to different teams, disciplines, and technologies commonly found in IT organizations. While the most obvious and immediate security management differences are in metastructure, cloud differs extensively from traditional computing within each layer. The scale of the differences will depend not only on the cloud platform, but on *how* exactly the cloud user utilizes the platform.

For example, a cloud-native application that makes heavy utilization of a cloud provider's PaaS products will experience more applistructure differences than the migration of an existing application, with minimal changes, to Infrastructure as a Service.

1.2 Cloud Security Scope, Responsibilities, and Models

1.2.1 Cloud Security and Compliance Scope and Responsibilities

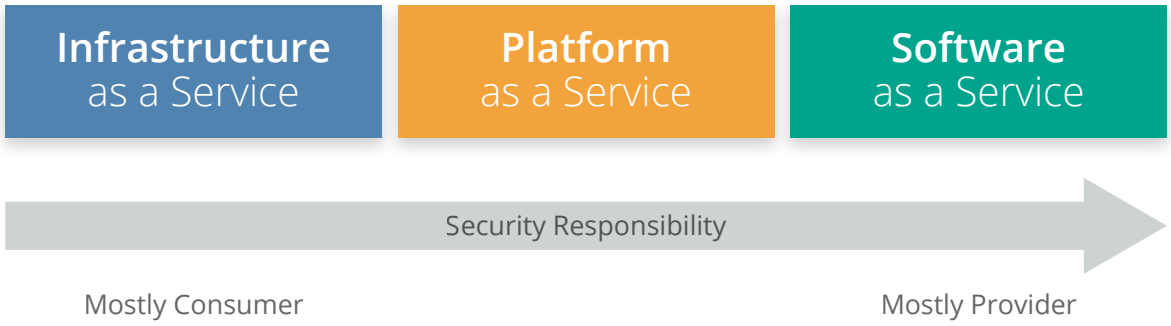
It might sound simplistic, but cloud security and compliance includes everything a security team is responsible for today, just in the cloud. All the traditional security domains remain, but the *nature of risks*, *roles and responsibilities*, and *implementation of controls* change, often dramatically.

Though the overall scope of security and compliance doesn't change, the pieces any given cloud actor is responsible for most certainly do. Think of it this way: Cloud computing is a shared technology model where different organizations are frequently responsible for implementing and managing different parts of the stack. As a result security responsibilities are also distributed across the stack, and thus across the organizations involved.

This is commonly referred to as the *shared responsibility model*. Think of it as a responsibility matrix that depends on the particular cloud provider and feature/product, the service model, and the deployment model.

At a high level, security responsibility maps to the degree of control any given actor has over the architecture stack:

- *Software as a Service:* The cloud provider is responsible for nearly all security, since the cloud user can only access and manage their use of the application, and can't alter how the application works. For example, a SaaS provider is responsible for perimeter security, logging/monitoring/auditing, and application security, while the consumer may only be able to manage authorization and entitlements.
- *Platform as a Service:* The cloud provider is responsible for the security of the platform, while the consumer is responsible for everything they implement on the platform, including how they configure any offered security features. The responsibilities are thus more evenly split. For example, when using a Database as a Service, the provider manages fundamental security, patching, and core configuration, while the cloud user is responsible for everything else, including which security features of the database to use, managing accounts, or even authentication methods.
- *Infrastructure as a Service:* Just like PaaS, the provider is responsible for foundational security, while the cloud user is responsible for everything they build on the infrastructure. Unlike PaaS, this places far more responsibility on the client. For example, the IaaS provider will likely monitor their perimeter for attacks, but the consumer is fully responsible for how they define and implement their virtual network security, based on the tools available on the service.



These roles are further complicated when using cloud brokers or other intermediaries and partners.

The most important security consideration is knowing exactly who is responsible for what in any given cloud project. It's less important if any particular cloud provider offers a specific security control, as long as you know precisely what they do offer and how it works. You can fill the gaps with your own controls, or choose a different provider if you can't close the controls gap. Your ability to do this is very high for IaaS, and less so for SaaS.

This is the essence of the security relationship between a cloud provider and consumer. What does the provider do? What does the consumer need to do? Does the cloud provider enable the consumer to do what they need to? What is guaranteed in the contract and service level agreements, and what is implied by the documentation and specifics of the technology?

This shared responsibility model directly correlates to two recommendations:

- *Cloud providers* should clearly document their internal security controls and customer security features so the cloud user can make an informed decision. Providers should also properly design and implement those controls.
- *Cloud users* should, for any given cloud project, build a responsibilities matrix to document who is implementing which controls and how. This should also align with any necessary compliance standards.

The Cloud Security Alliance provides two tools to help meet these requirements:

- The **Consensus Assessments Initiative Questionnaire (CAIQ)**. A standard template for cloud providers to document their security and compliance controls.
- The **Cloud Controls Matrix (CCM)**, which lists cloud security controls and maps them to multiple security and compliance standards. The CCM can also be used to document security responsibilities.

Both documents will need tuning for specific organizational and project requirements, but provide a comprehensive starting template and can be especially useful for ensuring compliance requirements are met.

1.2.2 Cloud Security Models

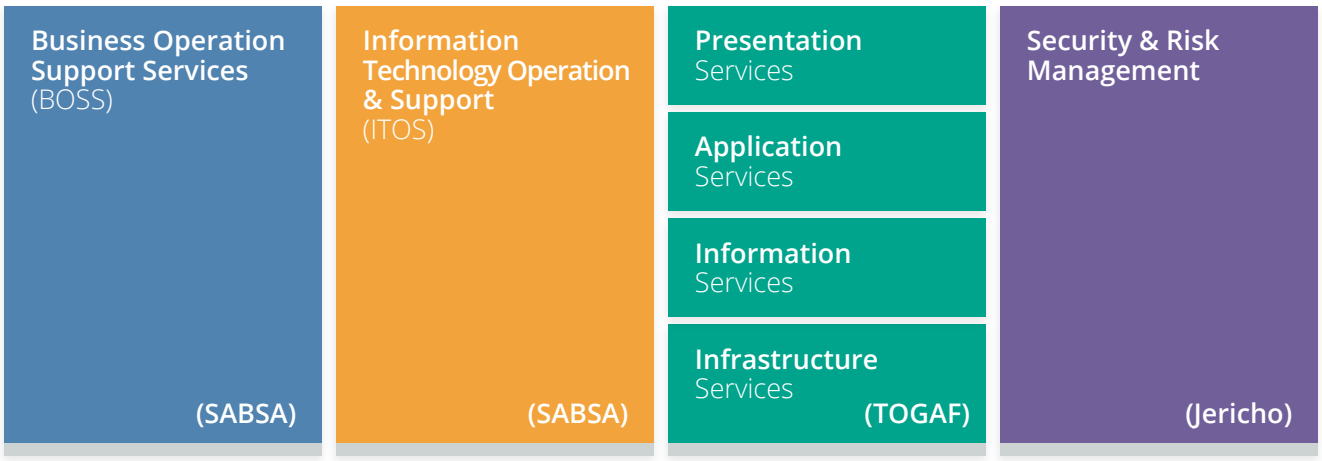
Cloud security models are tools to help guide security decisions. The term "model" can be used a little nebulously, so for our purposes we break out the following types:

- *Conceptual models or frameworks* include visualizations and descriptions used to explain cloud security concepts and principles, such as the CSA logical model in this document.
- *Controls models or frameworks* categorize and detail specific cloud security controls or categories of controls, such as the CSA CCM.
- *Reference architectures* are templates for implementing cloud security, typically generalized (e.g. an IaaS security reference architecture). They can be very abstract, bordering on conceptual, or quite detailed, down to specific controls and functions.
- *Design patterns* are reusable solutions to particular problems. In security, an example is IaaS log management. As with reference architectures, they can be more or less abstract or specific, even down to common implementation patterns on particular cloud platforms.

The lines between these models often blur and overlap, depending on the goals of the developer of the model. Even lumping these all together under the heading "model" is probably inaccurate, but since we see the terms used so interchangeably across different sources, it makes sense to group them.

The CSA has reviewed and recommends the following models:

- The **CSA Enterprise Architecture**
- The CSA **Cloud Controls Matrix**
- The NIST draft **Cloud Computing Security Reference Architecture (NIST Special Publication 500-299)**, which includes conceptual models, reference architectures, and a controls framework.
- **ISO/IEC FDIS 27017 Information technology – Security techniques – Code of practice for information security controls based on ISO/IEC 27002 for cloud services**.



Throughout this Guidance we also refer to other domain-specific models.

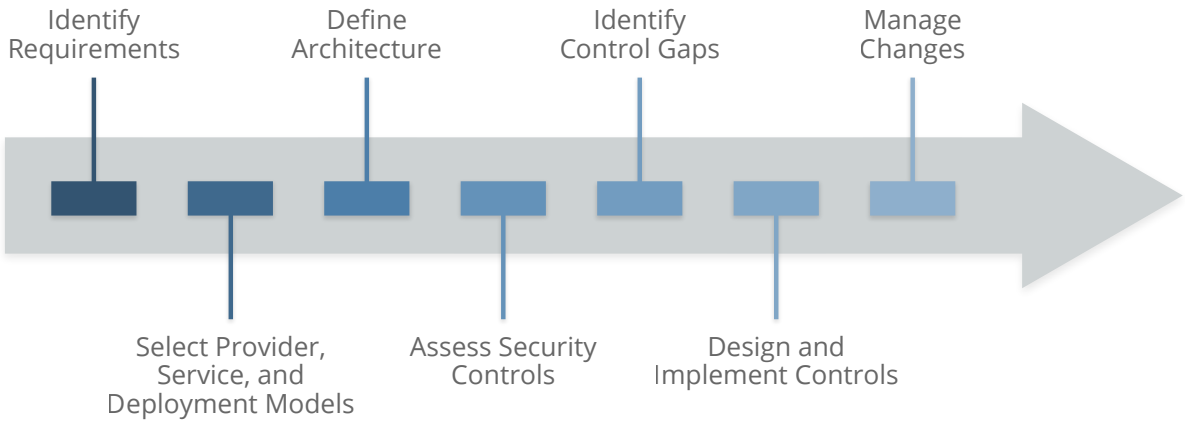
1.2.2.1 A Simple Cloud Security Process Model

While the implementation details, necessary controls, specific processes, and various reference architectures and design models vary greatly depending on the specific cloud project, there is a relatively straightforward, high-level process for managing cloud security:

- Identify necessary security and compliance requirements, and any existing controls.
- Select your cloud provider, service, and deployment models.
- Define the architecture.
- Assess the security controls.
- Identify control gaps.
- Design and implement controls to fill the gaps.
- Manage changes over time.

Since different cloud projects, even on a single provider, will likely leverage entirely different sets of configurations and technologies, each project should be evaluated on its own merits. For example, the security controls for an application deployed on pure IaaS in one provider may look very different than a similar project that instead uses more PaaS from that same provider.

The key is to identify requirements, design the architecture, and then identify the gaps based on the capabilities of the underlying cloud platform. That's why you need to know the cloud provider and architecture *before* you start translating security requirements into controls.



1.3 Areas of Critical Focus

The 13 other domains which comprise the remainder of the CSA Guidance highlight areas of concern for cloud computing and are tuned to address both the strategic and tactical security "pain points" within a cloud environment, and can be applied to any combination of cloud service and deployment model.

The domains are divided into two broad categories: governance and operations. The governance domains are broad and address strategic and policy issues within a cloud computing environment, while the operational domains focus on more tactical security concerns and implementation within the architecture.

1.3.1 Governing in the Cloud

Domain	Title	Description
2	Governance and Enterprise Risk Management	The ability of an organization to govern and measure enterprise risk introduced by cloud computing. Items such as legal precedence for agreement breaches, ability of user organizations to adequately assess risk of a cloud provider, responsibility to protect sensitive data when both user and provider may be at fault, and how international boundaries may affect these issues.
3	Legal Issues: Contracts and Electronic Discovery	Potential legal issues when using cloud computing. Issues touched on in this section include protection requirements for information and computer systems, security breach disclosure laws, regulatory requirements, privacy requirements, international laws, etc.
4	Compliance and Audit Management	Maintaining and proving compliance when using cloud computing. Issues dealing with evaluating how cloud computing affects compliance with internal security policies, as well as various compliance requirements (regulatory, legislative, and otherwise) are discussed here. This domain includes some direction on proving compliance during an audit.
5	Information Governance	Governing data that is placed in the cloud. Items surrounding the identification and control of data in the cloud, as well as compensating controls that can be used to deal with the loss of physical control when moving data to the cloud, are discussed here. Other items, such as who is responsible for data confidentiality, integrity, and availability are mentioned.

1.3.2 Operating in the Cloud

Domain	Title	Description
6	Management Plane and Business Continuity	Securing the management plane and administrative interfaces used when accessing the cloud, including both web consoles and APIs. Ensuring business continuity for cloud deployments.
7	Infrastructure Security	Core cloud infrastructure security, including networking, workload security, and hybrid cloud considerations. This domain also includes security fundamentals for private clouds.
8	Virtualization and Containers	Security for hypervisors, containers, and Software Defined Networks.
9	Incident Response, Notification and Remediation	Proper and adequate incident detection, response, notification, and remediation. This attempts to address items that should be in place at both provider and user levels to enable proper incident handling and forensics. This domain will help you understand the complexities the cloud brings to your current incident-handling program.
10	Application Security	Securing application software that is running on or being developed in the cloud. This includes items such as whether it's appropriate to migrate or design an application to run in the cloud, and if so, what type of cloud platform is most appropriate (SaaS, PaaS, or IaaS).
11	Data Security and Encryption	Implementing data security and encryption, and ensuring scalable key management.
12	Identity, Entitlement, and Access Management	Managing identities and leveraging directory services to provide access control. The focus is on issues encountered when extending an organization's identity into the cloud. This section provides insight into assessing an organization's readiness to conduct cloud-based Identity, Entitlement, and Access Management (IdEA).
13	Security as a Service	Providing third-party-facilitated security assurance, incident management, compliance attestation, and identity and access oversight.
14	Related Technologies	Established and emerging technologies with a close relationship to cloud computing, including Big Data, Internet of Things, and mobile computing.

1.4 Recommendations

- Understand the differences between cloud computing and traditional infrastructure or virtualization, and how *abstraction* and *automation* impact security.
- Become familiar with the NIST model for cloud computing and the CSA reference architecture.
- Use tools such as the CSA Consensus Assessments Initiative Questionnaire (CAIQ) to evaluate and compare cloud providers.
- Cloud providers should clearly document their security controls and features and publish them using tools like the CSA CAIQ.
- Use tools like the CSA Cloud Controls Matrix to assess and document cloud project security and compliance requirements and controls, as well as who is responsible for each.
- Use a cloud security process model to select providers, design architectures, identify control gaps, and implement security and compliance controls.

1.5 Credits

- Reference architecture and logical model based on the work of Christofer Hoff

DOMAIN 2
Governance and Enterprise Risk Management

2.0 Introduction

Governance and risk management are incredibly broad topics. This guidance focuses on how they change in cloud computing; it is not and should not be considered a primer or comprehensive exploration of those topics outside of cloud.

For security professionals, cloud computing impacts four areas of governance and risk management:

- *Governance* includes the policy, process, and internal controls that comprise how an organization is run. Everything from the structures and policies to the leadership and other mechanisms for management.

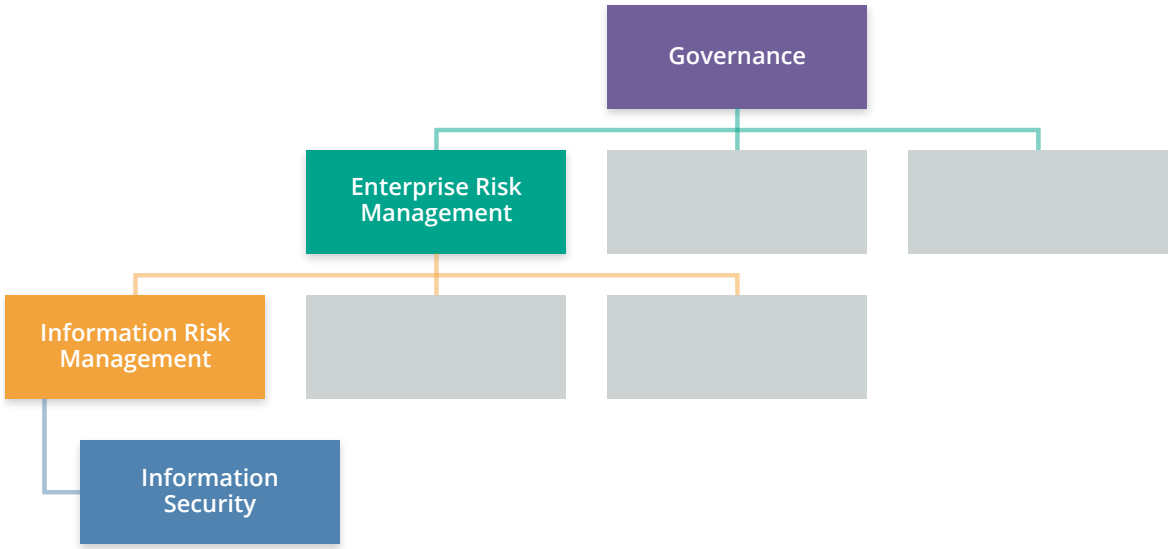
> For more information on governance please see
> * **ISO/IEC 38500:2015 - Information Technology - Governance of IT for the organization**
> * ISACA - COBIT - **A Business Framework for the Governance and Management of Enterprise IT**
> * **ISO/IEC 27014:2013 - Information Technology - Security techniques - Governance of information security**

- *Enterprise risk management* includes managing overall risk for the organization, aligned to the organization's governance and risk tolerance. Enterprise risk management includes all areas of risk, not merely those concerned with technology.

- *Information risk management* covers managing the risk to information, including information technology. Organizations face all sorts of risks, from financial to physical, and information is only one of multiple assets an organization needs to manage.

- *Information security* is the tools and practices to manage risk to information. Information security isn't the be-all and end-all of managing information risks; policies, contracts, insurance, and other mechanisms also play a role (including physical security for non-digital information). However, a—if not *the*—primary role of information security is to provide the processes and controls to protect electronic information and the systems we use to access it.

In a simplified hierarchy, information security is a tool of information risk management, which is a tool of enterprise risk management, which is a tool of governance. The four are all closely related but require individual focus, processes, and tools.



2.1: A Simplified Risk and Governance Hierarchy

Legal issues and compliance are covered in Domains 3 and 4, respectively. Information risk management and data governance are covered in Domain 5. Information security is essentially the rest of this guidance.

2.1 Overview

2.1.1 Governance

Cloud computing affects governance, since it either introduces a third party into the process (in the case of public cloud or hosted private cloud) or potentially alters internal governance structures in the case of self-hosted private cloud. The primary issue to remember when governing cloud computing is that *an organization can never outsource responsibility for governance*, even when using external providers. This is always true, cloud or not, but is useful to keep in mind when navigating cloud computing's concepts of shared responsibility models.

Cloud service providers try to leverage economies of scale to manage costs and enable capabilities. This means creating extremely standardized services (including contracts and service level agreements) that are consistent across all customers. Governance models can't necessarily treat cloud providers the same way they'd treat dedicated external service providers, which typically customize their offerings, including legal agreements, for each client.

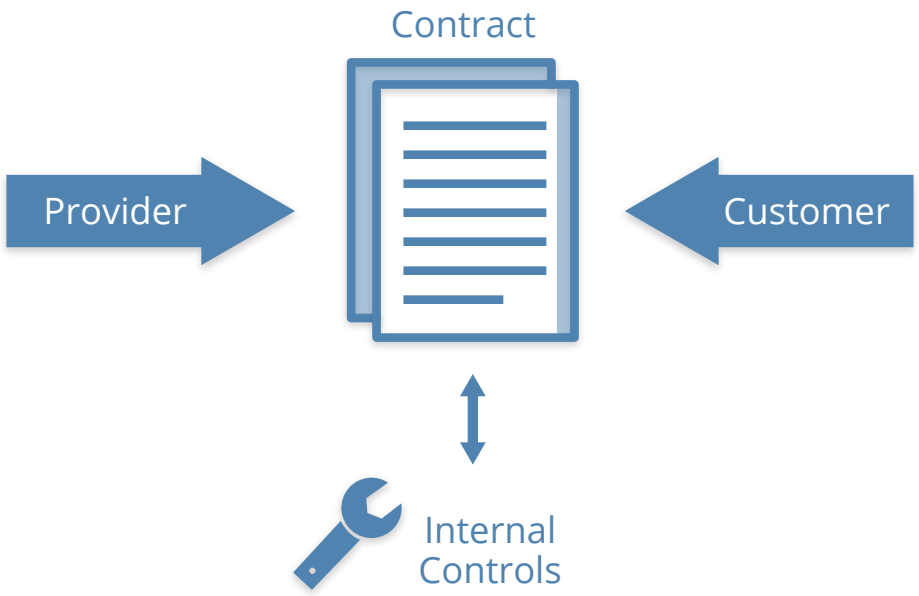
Cloud computing changes the *responsibilities* and mechanisms for implementing and managing governance. Responsibilities and mechanisms for governance are defined in the contract, as with

any business relationship. If the area of concern isn't in the contract, there are no mechanisms available to enforce, and there is a governance gap. Governance gaps don't necessarily exclude using the provider, but they do require the customer to adjust their own processes to close the gaps or accept the associated risks.

2.1.1.1 Tools of Cloud Governance

As with any other area, there are specific management tools used for governance. This list focuses more on tools for external providers, but these same tools can often be used internally for private deployments:

- *Contracts:* The primary tool of governance is the contract between a cloud provider and a cloud customer (this is true for public and private cloud). The contract is your only guarantee of any level of service or commitment—assuming there is no breach of contract, which tosses everything into a legal scenario. Contracts are the primary tool to extend governance into business partners and providers.



> Contracts define the relationship between providers and customers and are the primary tool for customers to extend governance to their suppliers.

- *Supplier (cloud provider) Assessments:* These assessments are performed by the potential cloud customer using available information and allowed processes/techniques. They combine contractual and manual research with third-party attestations (legal statements often used to communicate the results of an assessment or audit) and technical research. They are very similar to any supplier assessment and can include aspects like financial viability, history, feature offerings, third-party attestations, feedback from peers, and so on. More detail on assessments is covered later in this Domain and in Domain 4.

- *Compliance reporting:* Compliance reporting includes all the documentation on a provider's internal (i.e. self) and external compliance assessments. They are the reports from *audits of controls*, which an organization can perform themselves, a customer can perform on a provider (although this usually isn't an option in cloud), or have performed by a trusted third party. Third-party audits and assessments are preferred since they provide independent validation (assuming you trust the third party).

  Compliance reports are often available to cloud prospects and customers but may only be available under NDA or to contracted customers. This is often required by the firm that performed the audit and isn't necessarily something that's completely under the control of the cloud provider.

Assessments and audits should be based on existing standards (of which there are many). It's critical to understand the scope, not just the standard used. Standards like the SSAE 16 have a defined scope, which includes both *what* is assessed (e.g. which of the provider's services) as well as *which controls* are assessed. A provider can thus "pass" an audit that doesn't include any security controls, which isn't overly useful for security and risk managers. Also consider the transitive trust required to treat a third-party assessment as equivalent to the activities that you might undertake when doing your own assessment. Not all audit firms (or auditors) are created equal and the experience, history, and qualifications of the firm should be included in your governance decisions.

The **Cloud Security Alliance STAR Registry** is an assurance program and documentation registry for cloud provider assessments based on the CSA Cloud Controls Matrix and Consensus Assessments Initiative Questionnaire. Some providers also disclose documentation for additional certifications and assessments (including self-assessments).

2.1.2 Enterprise Risk Management

Enterprise Risk Management (ERM) is the overall management of risk for an organization. As with governance, the contract defines the roles and responsibilities for risk management between a cloud provider and a cloud customer. And, as with governance, you can never outsource your overall responsibility and accountability for risk management to an external provider.

> For more on risk management see
> * **ISO 31000:2009 - Risk management – Principles and guidelines**
> * **ISO/IEC 31010:2009 - Risk management – Risk assessment techniques**
> * [NIST Special Publication 800-37 Revision 1](updated June 5, 2014) (**http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-37r1.pdf**)

Risk management in cloud is based on the *shared responsibilities model* (which we most often discuss in reference to security). The cloud provider accepts some responsibility for certain risks, and the cloud customer is responsible for anything beyond that. This is especially evident as you evaluate differences between the service models, where the provider manages more risks in SaaS and the consumer more in IaaS. But, again, the cloud user is ultimately responsible for ownership of the risks; they only pass on some of the *risk management* to the cloud provider. This holds true even

with a self-hosted private cloud; in those situations an organizational unit is passing on some of their risk management to the internal cloud provider instead of an external party, and internal SLAs and procedures replace external contracts.

ERM relies on good contracts and documentation to know where the division of responsibilities and potential for untreated risk lie. Whereas governance is nearly exclusively focused on contracts, risk management can delve deeper into the technology and process capabilities of the provider, based on their documentation. For example, a contract will rarely define how network security is actually implemented. Review of the provider's documentation will provide much more information to help with an effective risk decision.

Risk tolerance is the amount of risk that the leadership and stakeholders of an organization are willing to accept. It varies based on asset and you shouldn't make a blanket risk decision about a particular provider; rather, assessments should align with the value and requirements of the assets involved. Just because a public cloud provider is external and a consumer might be concerned with shared infrastructure for some assets doesn't mean it isn't within risk tolerance for all assets. Over time this means that, practically speaking, you will build out a matrix of cloud services along with which types of assets are allowed in those services. Moving to the cloud doesn't change your risk tolerance, it just changes how risk is managed.

2.1.3The Effects of Service Model and Deployment Model

In considering the various options available not only in Cloud Service Providers but also in the fundamental delivery of cloud services, attention must be paid to how the Service and Deployment models affect the ability to manage governance and risk.

2.1.3.1 Service Models

Software as a Service (SaaS)

In the majority of cases, SaaS presents the most critical example of the need for a negotiated contract. Such a contract will protect the ability to govern or validate risk as it relates to data stored, processed, and transmitted with and in the application. SaaS providers tend to cluster at either end of the size/capability spectrum and the likelihood of a negotiated contract is much higher when dealing with a small SaaS provider. Unfortunately, many small SaaS providers are not able to operate at a level of sophistication that meets or exceeds customer governance and risk management capabilities. In concrete terms, the entire level of visibility into the actual operation of the infrastructure providing the SaaS application is limited to solely what is exposed in the user interface developed by the Cloud Provider.

Platform as a Service (PaaS)

Continuing through the Service Models, the level of detail that is available (and the consequential ability to self-manage governance and risk issues) increases. The likelihood of a fully negotiated contract is likely lower here than with either of the other service models. That's because the core driver for most PaaS is to deliver a single capability with very high efficiency.

PaaS is typically delivered with a rich API, and many PaaS providers have enabled the collection of

some of the data necessary to prove that SLAs are being adhered to. That said, the customer is still in the position of having to exercise a significant effort in determining whether contract stipulations are effectively providing the level of control or support required to enable governance or risk management.

Infrastructure as a Service (IaaS)

Infrastructure as a Service represents the closest that Cloud comes to a traditional data center (or even a traditional outsourced managed data center), and the good news is that the vast majority of existing governance and risk management activities that organizations have already built and utilize are directly transferable. There are, however, new complexities related to the underlying orchestration and management layers, as described in Domain 1, that enable the infrastructure which are often overlooked.

In many ways, the governance and risk management of that orchestration and management layer is consistent with the underlying infrastructure (network, power, HVAC, etc.) of a traditional data center. The same governance and risk management issues are present, but the exposure of those systems is sufficiently different that changes to the existing process are required. For example, controlling who can make network configuration changes shifts from accounts on individual devices to the cloud management plane.

2.1.3.2 Deployment Models

Cloud customers have a reduced ability to govern operations in a public cloud since the provider is responsible for the management and governance of their infrastructure, employees, and everything else. The customers also often have reduced ability to negotiate contracts, which impacts how they extend their governance model into the cloud. Inflexible contracts are a natural property of multitenancy: Providers can't necessarily adjust contracts and operations for each customer as everything runs on one set of resources, using one set of processes. Adapting for different customers increases costs, causing a trade-off, and often that's the dividing line between using public and private cloud. Hosted private cloud allows full customization, but at increased costs due to the loss of the economies of scale.

This doesn't mean you shouldn't try to negotiate your contract, but recognize that this isn't always possible; instead, you'll need to either choose a different provider (which may actually be less secure), or adjust your needs and use alternate governance mechanisms to mitigate concerns.

To use an analogy, think of a shipping service. When you use a common carrier/provider you don't get to define their operations. You put your sensitive documents in a package and entrust them to meet their obligations to deliver it safely, securely, and within the expected Service Level Agreement.

Public cloud isn't the only model that impacts governance; even private cloud will have an effect. If an organization allows a third party to own and/or manage the private cloud (which is very common), this is similar to how governance affects any outsourced provider. There will be shared responsibilities with obligations that are defined in the contract.

Although you will likely have more control over contractual terms, it's still important to ensure they cover the needed governance mechanisms. As opposed to a public provider—which has various incentives to keep its service well-documented and at particular standard levels of performance, functionality, and competitiveness—a hosted private cloud may only offer exactly what is in the contract, with everything else at extra cost. This *must* be considered and accounted for in negotiations, with clauses to guarantee that the platform itself remains up to date and competitive. For example, by requiring the vendor to update to the latest version of the private cloud platform within a certain time period of release and after *your* sign-off.

With a self-hosted private cloud governance will focus on internal service level agreements for the cloud users (business or other organizational units) and chargeback and billing models for providing access to the cloud.

When contemplating **hybrid cloud environments**, the governance strategy must consider the minimum common set of controls comprised of the Cloud Service Provider's contract and the organization's internal governance agreements. The cloud user is connecting either two cloud environments or a cloud environment and a data center. In either case the overall governance is the intersection of those two models. For example, if you connect your data center to your cloud over a dedicated network link you need to account for governance issues that will span both environments.

Since **community clouds** are a shared platform with multiple organizations, but are not public, governance extends to the relationships with those members of the community, not just the provider and the customer. It's a mix of how you would approach public cloud and hosted private cloud governance, where the overall tools of governance and contracts will have some of the economies of scale of a public cloud provider, but be tunable based on community consensus, as with a hosted private cloud. This also includes community membership relations, financial relationships, and how to respond when a member leaves the community.

2.1.3.3 Cloud Risk Management Trade-Offs

There are advantages and disadvantages to managing enterprise risk for cloud deployments. These factors are, as you would expect, more pronounced for public cloud and hosted private cloud:

- There is less physical control over assets and their controls and processes. You don't physically control the infrastructure or the provider's internal processes.
- There is a greater reliance on contracts, audits, and assessments, as you lack day-to-day visibility or management.
- This creates an increased requirement for proactive management of relationship and adherence to contracts, which extends beyond the initial contract signing and audits. Cloud providers also constantly evolve their products and services to remain competitive and these ongoing innovations might exceed, strain, or not be covered by existing agreements and assessments.
- Cloud customers have a reduced need (and associated reduction in costs) to manage risks that the cloud provider accepts under the shared responsibility model. You haven't

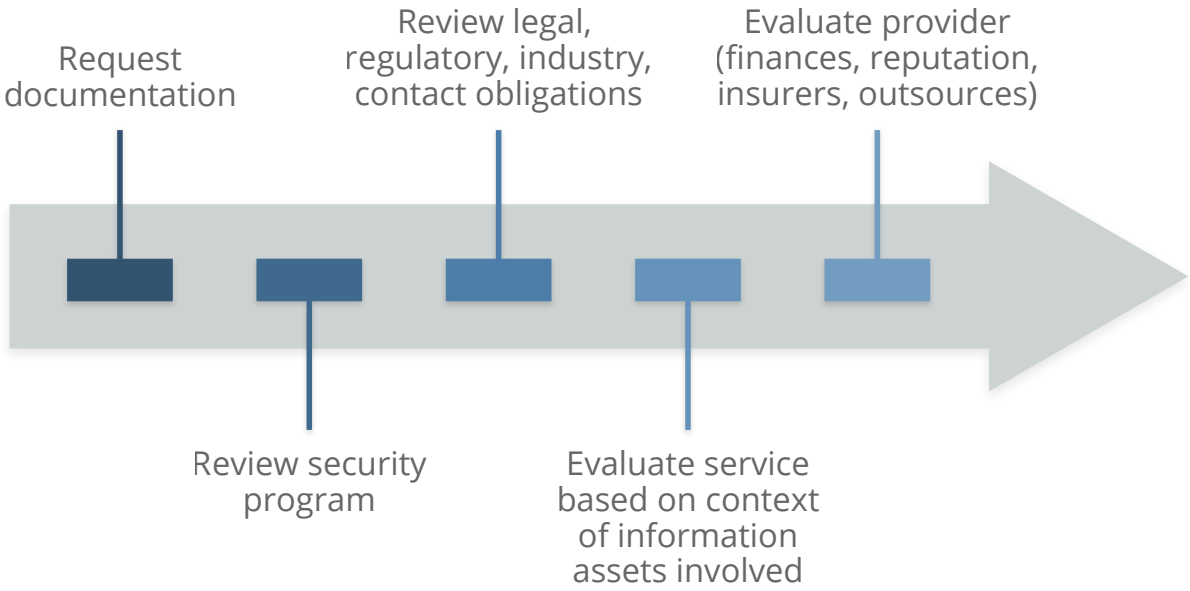
outsourced accountability for managing the risk, but you can certainly outsource the management of some risks.

2.1.3.4 Cloud Risk Management Tools

The following processes help form the foundation of managing risk in cloud computing deployments. One of the core tenents of risk management is that you can *manage*, *transfer*, *accept*, or *avoid* risks. But everything starts with a proper assessment.

The supplier assessment sets the groundwork for the cloud risk management program:

- Request or acquire documentation.
- Review their security program and documentation.
- Review any legal, regulatory, contractual, and jurisdictional requirements for both the provider and yourself. (See the Domain 3: Legal for more.)
- Evaluate the contracted service in the context of your information assets.
- Separately evaluate the overall provider, such as finances/stability, reputation, and outsourcers.



Supplier Assessment Process

Periodically review audits and assessments to ensure they are up to date:

- Don't assume all services from a particular provider meet the same audit/assessment standards. They can vary.
- Periodic assessments should be scheduled and *automated* if possible.

After reviewing and understanding what risks the cloud provider manages, what remains is residual risk. Residual risk may often be managed by controls that you implement (e.g. encryption). The availability and specific implementation of risk controls vary greatly across cloud providers, particular

services/features, service models, and deployment models. If, after all your assessments and the controls that you implement yourself there is still residual risk your only options are to transfer it, accept the risk, or avoid it.

Risk transfer, most often enabled by insurance, is an imperfect mechanism, especially for information risks. It can compensate some of the financial loss associated with a primary loss event, but won't help with a secondary loss event (like loss of customers)—especially an intangible or difficult to quantify loss, such as reputation damage. From the perspective of insurance carriers, cyber-insurance is also a nascent field without the depth of actuarial tables used for other forms of insurance, like those for fire or flooding, and even the financial compensation may not match the costs associated with the primary loss event. Understand the limits.

2.2 Recommendations

- Identify the shared responsibilities of security and risk management based on the chosen cloud deployment and service model. Develop a Cloud Governance Framework/Model as per relevant industry best practices, global standards, and regulations like CSA CCM, COBIT 5, NIST RMF, ISO/IEC 27017, HIPAA, PCI DSS, EU GDPR, etc.
- Understand how a contract affects your governance framework/model.
    - Obtain and review contracts (and any referenced documents) before entering into an agreement.
    - Don't assume that you can effectively negotiate contracts with a cloud provider—but this also shouldn't necessarily stop you from using that provider.
    - If a contract can't be effectively negotiated and you perceive an unacceptable risk, consider alternate mechanisms to manage that risk (e.g. monitoring or encryption).
- Develop a process for cloud provider assessments.
    - This should include:
        - Contract review.
        - Self-reported compliance review.
        - Documentation and policies.
        - Available audits and assessments.
        - Service reviews adapting to the customer's requirements.
        - Strong change-management policies to monitor changes in the organization's use of the cloud services.
    - Cloud provider re-assessments should occur on a scheduled basis and be automated if possible.
- Cloud providers should offer easy access to documentation and reports needed by cloud prospects for assessments.
    - For example, the CSA STAR registry.
- Align risk requirements to the specific assets involved and the risk tolerance for those assets.
- Create a specific risk management and risk acceptance/mitigation methodology to assess the risks of every solution in the space
- Use controls to manage residual risks.
    - If residual risks remain, choose to accept or avoid the risks.
- Use tooling to track approved providers based on asset type (e.g. linked to data classification), cloud usage, and management.

DOMAIN 3
Legal Issues, Contracts and Electronic Discovery

3.0 Introduction

This domain highlights some of the legal issues raised by moving data to the cloud; contracting with cloud service providers; and handling electronic discovery requests in litigation. Our overview here cannot address every potential legal situation. To address your specific issues, you should consult with legal counsel in the jurisdiction(s) in which you intend to operate and/or in which your customers reside. In addition, be aware that laws and regulations change frequently, and thus you should verify the relevancy of information contained in this domain before relying on it. Domain 3 is concerned primarily with the legal implications of public cloud computing and third party-hosted private clouds. Although this domain includes some aspects of data governance and audit/compliance, those topics are covered in more depth in domains 4 and 5.

Specific areas covered in this domain include the following:

- Legal issues
- Cloud service agreements (contracts)
- Third-party access to electronic documents stored in the cloud

3.1 Overview

3.1.1 Legal Frameworks Governing Data Protection and Privacy

Throughout the world, many countries have adopted legal frameworks requiring public and private organizations to safeguard the privacy of personal data and the security of information and computer systems. Most of these laws are based in part on the fair information privacy principles developed in the late 1960s and 1970s and later clarified and expanded in the Privacy and Security Guidelines of the Organization for Economic Cooperation and Development (OECD).

Under these laws, the data controller (typically the entity that has the primary relationship with an individual) is prohibited from collecting and processing personal data unless certain criteria are met. For example, if the data subject has consented to the collection and proposed uses of his or

her data, then the controller may collect and process data, according to the consent agreement. These laws define numerous obligations, such as confidentiality and security obligations, for the entities that access personal data. When entrusting a third party to process data on its behalf (a data processor), a data controller remains responsible for the collection and processing of that data. The data controller is required to ensure that any such third parties take adequate technical and organizational security measures to safeguard the data.

Despite a common theme, countries on all continents have developed data protection regimes that occasionally conflict with each other. As a result, cloud providers and cloud users operating in multiple regions struggle to meet compliance requirements.

In many cases, the laws of different countries might apply concurrently, in accordance with the following:

- The location of the cloud provider
- The location of the cloud user
- The location of the data subject
- The location of the servers
- The legal jurisdiction of the contract between parties, which may be different than the locations of any of the parties involved
- Any treaties or other legal frameworks between those various locations

> *Applicable legal requirements will vary tremendously based on the various jurisdictions and legal entities and frameworks involved.*

3.1.1.1 Common Themes

Many countries have adopted national or omnibus laws (applying to all categories of personal data) or sectoral laws (applying to specified categories of data) that are intended to protect the privacy of individuals.

3.1.1.2 Required Security Measures

These laws frequently contain provisions requiring the adoption of security measures, acknowledging that ensuring the security of personal data is essential to ensuring the protection of individual privacy. Concurrently, companies may also be expected to adopt reasonable technical, physical, and administrative measures in order to protect a wide range of data, including personal data, financial data, trade secrets, and other sensitive company data from loss, misuse or alteration.

3.1.1.3 Restrictions to Cross-border Data Transfers

Many countries prohibit or restrict the transfer of information out of their borders. In most cases, the transfer is permitted only if the country to which the data is transferred offers an "adequate level of protection" (as defined in the relevant national law) of personal information and privacy rights of affected individuals. The purpose of this adequacy requirement is to ensure the individuals whose data is transferred across borders will remain as protected as they were via policies afforded to them before the transfer of data.

Alternatively, the data importer and exporter may need to sign a contract ensuring the maintenance of privacy rights for data subjects. Depending on the country, the requirements for ensuring this adequate protection may be complex and stringent. In some cases, it may be necessary to obtain prior permission of the local Data Protection Commissioner before transferring data in or out of the country.

In addition, some countries are beginning to require that certain data be stored within their territory. This is the case, for example, with the new data localization laws of Russia and China, which require that specified personal data pertaining to individuals residing in their countries be stored within the country's borders.

3.1.1.4 Regional Examples

Below are examples of information privacy and security laws and legal frameworks in effect in numerous parts of the world.

ASIA PACIFIC REGION

Australia

In Australia, two key laws provide protection to consumers of cloud services: the Privacy Act of 1988 (Privacy Act) and the Australian Consumer Law (ACL). The Privacy Act includes 13 Australian Privacy Principles (APPs), which apply to all private sector and not-for-profit organizations with an annual turnover of more than AUD 3 million, all private health service providers, and some small businesses.

In February 2017, Australia amended its 1988 Privacy Act to require companies to notify affected Australian residents and the Australian Information Commissioner in the event of a breach of security. A breach of security must be reported if (a) there is unauthorized access or disclosure of personal information that would be likely to result in serious harm; or (b) personal information is lost in circumstances where unauthorized access or disclosure is likely to occur, and if it did occur, it would be likely to result in serious harm to any of the individuals to whom the information relates.

The ACL protects consumers from false or misleading contracts and poor conduct from providers, such as failed breach notifications. The Privacy Act can apply to Australian customers, even if the cloud service provider is based elsewhere, and even if other laws are stated in a contract.

China

Over the past few years, China has accelerated the pace of its adoption of legal structures to address the privacy and security of personal and company information. Its 2017 Cyber Security Law governs the operations of network operators and critical information infrastructure operators. In May 2017, proposed Measures on the Security of Cross-Border Transfers of Personal Information and Important Data were published by the Chinese government and are currently being evaluated for potential implementation.

The 2017 Cyber Security Law requires network operators to comply with a series of security requirements, including the design and adoption of information security measures; the formulation of cyber security emergency response plans; and assistance and support necessary to investigative authorities, where necessary, for protecting national security and investigating crimes. The law

requires providers of network products and services to inform users about known security defects and bugs and to report such defects and bugs to relevant authorities.

The Cyber Security Law imposes a series of security obligations to operators of critical information infrastructure, including internal organization, training, data backup; emergency response requirements, security inspections and annual assessments of cyber security risks; and reporting to relevant authorities. In addition, the law includes a data localization provision, which requires that personal information and other important data be stored within the territories of the People's Republic of China.

During the second quarter of 2017, China issued Draft Regulations on Cross Border Data Transfers to supplement the Cyber Security Law. These regulations would go beyond the working of the Cyber Security Law, and expand its scope. The draft regulations would impose new security review requirements on companies that contemplate sending data overseas. They would expand data localization requirements, and increase the categories of information that must be stored only on China's borders. In particular, they would to include personal information and important data collected by any network operators. The cybersecurity and privacy landscape as defined under the Cyber Security Law is in evolution, and has not yet stabilized.

Japan

In Japan, the Act on the Protection of Personal Information (APPI) requires the private sector to protect personal information and data securely. There are several other national laws, such as the Law on the Protection of Personal Information Held by Administrative Organs and laws pertaining to specific sectors, such as the healthcare industry. Profession-specific laws, such as the Medical Practitioners' Act; the Act on Public Health Nurses, Midwives and Nurses; and the Pharmacist Act, require registered health professionals to maintain the confidentiality of patient information.

Beginning in September 2017, amendments to the APPI law will limit the ability to transfer personal data to third parties, with prior consent of the data subject generally being required to transfer data to a third party. Consent to the transfer is not required if the country of destination has an established framework for the protection of personal information that meets the standard specified by the Personal Information Protection Commission.

Russia

The Russian data protection laws contain significant restrictions on data processing, including a requirement for consent for most forms of data processing. However, the most important aspect of the Russian legal framework regarding the handling of personal information is its data localization law. Since September 2015, companies are required to store personal data of Russian citizens within Russia. Roskomnadzor, the Russian Data Protection regulator, has commenced enforcement of the law and has already blocked access to one foreign social network, which did not have a physical presence in Russia, but operated a Russian language version of its website.

EUROPEAN UNION AND EUROPEAN ECONOMIC AREA

The European Union (EU) adopted the General Data Protection Regulation (GDPR) in 2016, which is binding on all EU member states, as well as members of the European Economic Area (EEA). The GDPR will become enforceable as of May 25, 2018. On that date, Directive 95/46/EC on the Protection of Personal Data, which had been the legal basis of the provisions of the national data protection laws of all EU and EEA member states, will be repealed.

The other important document governing aspects of the protection of personal data in the EU/EEA is Directive 2002/58/EC on Privacy and Electronic Communications. This directive is being phased out and a first draft of an E-Privacy Regulation, which would replace it, has been published and could enter into effect as of May 25, 2018, but delays are likely.

From a security standpoint, the Network Information Security Directive (NIS Directive) is paving the way to more stringent security requirements. Adopted in 2016, the NIS Directive requires EU/EEA member states to implement new information security laws for the protection of critical infrastructure and essential services by May 2018. Cloud service providers and some cloud users are likely to be affected by the national laws that will implement the overarching NIS Directive.

General Data Protection Regulation (GDPR)

The new GDPR is directly binding on any corporation that processes the data of EU citizens, and will be adjudicated by the data supervisory authorities or the courts of the member states that have the closest relationship with the individuals or the entities on both sides of the dispute.

Applicability: The GDPR applies to the processing of personal data in the context of the activities of an establishment of a controller or processor in the EU/EEA, regardless of whether the processing takes place in the EU/EEA or not. It also applies to the processing of personal data of data subjects who are in the EU/EEA by a controller or a processor not established in the EU/EEA if the processing relates to (a) the offering of goods or services irrespective of whether a payment by the data subject is required; or (b) the monitoring of the behavior of a data subject, when the behavior takes place within the EU/EEA.

Lawfulness: The processing of personal data is allowed only if (a) the data subject has freely given specific, informed and unambiguous indication of his/her consent to the processing of his/her personal data, or (b) the processing is authorized by a statutory provision.

Accountability Obligations: The GDPR has created numerous obligations for companies. For example,

the GDPR requires companies to keep records of their data processing activities. Certain categories of processing require a prior "Privacy Impact Assessment." Companies are expected to develop and operate their products and services in accordance with "privacy by design" and "privacy by default" principles.

Data Subjects' Rights: Data subjects have rights to information regarding the processing of their data: the right to object to certain uses of their personal data; to have their data corrected or erased; to be compensated for damages suffered as a result of unlawful processing; the right to be forgotten; and the right to data portability. The existence of these rights significantly affects cloud service relationships.

Cross-border Data Transfer Restrictions: The transfer of personal data outside the EU/EEA to a country that does not offer a similar range of protection of personal data and privacy rights is prohibited. To prove that it will be offering the "adequate level of protection" required, a company may use one of several methods, such as executing Standard Contractual Clauses (SCC), signing up to the EU-US Privacy Shield, obtaining certification of Binding Corporate Rules (BCRs), or complying with an approved industry Code of Conduct or approved certification mechanism. In rare cases, the transfer might be effected with the explicit, informed, consent of the data subject, or if other exceptions apply.

Breaches of Security: The GDPR requires companies to report that they have suffered a breach of security. The reporting requirements are risk-based, and there are different requirements for reporting the breach to the Supervisory Authority and to the affected data subjects. Breaches must be reported within 72 hours of the company becoming aware of the incident.

Discrepancies among Member States: There are numerous instances where each member state may adopt its own rules. For example, Germany requires that a Data Protection Officer be appointed if the company has more than nine employees.

Sanctions: Violations of the GDPR expose a company to significant sanctions. These sanctions may reach up to the greater of four percent of their global turnover or gross income, or up to EUR 20 million.

Network Information Security Directive (NIS Directive)

The NIS Directive entered into force in August 2016, requiring each EU/EEA member state to implement the Directive into its national legislation by May 2018. The NIS Directive establishes a framework to enable networks and information systems to resist, at a given level of confidence, actions that compromise the availability, authenticity, integrity, or confidentiality of stored, transmitted, or processed data, or the related services that are offered by or accessible through those networks and information systems.

The NIS Directive requires that member state's national laws impose network and information security requirements on operators of essential services, i.e., entities that provide a service essential for the maintenance of critical societal and/or economic activities; and where an incident to the network and information systems of that service would have significant disruptive effects on the provision of that service. The requirements to be implemented into national laws include the following:

- Taking technical and organizational measures to manage risks posed to the security of networks and information systems used in their operations;
- Taking appropriate measures to prevent and minimize the impact of incidents affecting the security of the networks and information systems used for the provision of such essential services, to facilitate the continuation of those services;
- Notifying, without undue delay, the competent authorities or agencies of incidents having a significant impact on the continuity of the essential services they provide;
- Providing information necessary to assess the security of their networks and information systems
- Providing evidence of the effective implementation of security policies, such as the results of a security audit.

The NIS Directive also requires that member state's national laws impose network and information security requirements on digital service providers, such as online market places (e.g., e-commerce platforms), cloud computing services; and online search engines. Digital service providers based outside the EU that provide services within the EU fall under the scope of the NIS Directive.

Member state's national laws will also have to require digital service providers to identify and take appropriate and proportionate technical and organizational measures to manage risks posed to the security of network and information systems they use, such as incident handling, business continuity management, monitoring, auditing and testing, and compliance with international standards.

Member State's national laws will have to require digital service providers to take measures to prevent and minimize the impact of incidents. They will be required to notify the competent authorities or agencies, without undue delay, of any incident having a substantial impact on the provision of a service, including sufficient information to enable the competent authority or agency to determine the significance of any cross-border impact. Where an operator of essential services relies on a third party digital service provider for a service that is essential, the operator will be required to notify any significant impact on the continuity of the essential services due to an incident affecting the digital service provider.

THE AMERICAS

Central and South America

Central and South American countries also are adopting data protection laws at a rapid pace. Each of these laws includes a security requirement and places on the data custodian the burden of ensuring the protection and security of personal data wherever the data are located, and especially when transferring to a third party.

For example, Argentina, Chile, Colombia, Mexico, Peru and Uruguay have passed data protection laws inspired mainly by the European directive 95/46/EC, and may include references to the APEC Privacy Framework. The federal data protection law of Mexico includes security breach disclosure provisions.

North America: United States

Due to its sectoral approach, the United States has hundreds of federal, state and local regulations, from the details of a written information security plan to the rules for disclosing security breaches. As a result, organizations that do business in the United States or collect or process personal or other information of individuals or companies located in the United States are often subject to several federal, state or local privacy or information security laws. The variety and complexity of these rules might be daunting both for providers or users of cloud services and for the service providers and subcontractors who participate in the provision of these services.

U.S. Federal Laws

Numerous federal laws and their related regulations—such as the Gramm-Leach-Bliley Act (GLBA), the Health Insurance Portability and Accountability Act of 1996 (HIPAA), and the Children's Online Privacy Protection Act of 1998 (COPPA)—contain provisions that pertain to the privacy and the security of personal information. Security-related provisions require companies to adopt reasonable security measures when processing personal data.

Most of these laws require companies to take precautions when hiring subcontractors and service providers. They may also hold organizations responsible for the acts of their subcontractors. For example, the security and privacy rules under GLBA and HIPAA require that covered organizations compel their subcontractors, in written contracts, to use reasonable security measures and comply with data privacy provisions.

U.S. State Laws

In addition to federal laws and regulations, most U.S. states have laws relating to data privacy and/or data security. These laws apply to any entity that collects or processes personal information (as narrowly defined in the applicable law) of individuals who reside in that state, regardless of where in the United States the data is stored.

Some state laws are elaborate. See, for example, the extensive requirements under Massachusetts' "Standards for the Protection of Personal Information of Residents of the Commonwealth," or 201 CMR 17.00. Other state laws are more general (see Washington State law RCW 19.255.020(2)(b) that assigns liability on the basis of compliance) and a small number of state laws reference other specific standards (such as the Payment Card Industry Data Security Standard, PCI-DSS, mentioned above). Most state laws that address information security issues require that the company have a written contract with the service provider with reasonable security measures. Numerous state laws also require companies to provide adequate privacy protections and security for personal data, and require their service providers to do the same.

Security Breach Disclosure Laws

Numerous federal security laws or rules, such as those applying to healthcare providers, as well as most state laws, require entities that have suffered a breach of security that compromised specified categories of data, such as PHI (patient health information), to promptly notify affected individuals,

and in many cases, state or federal agencies, of the occurrence of the breach of security.

Knowledge and understanding of these laws is critical for both cloud customers and cloud vendors, because breaches of security often trigger significant cost, including for example, the cost of responding to class action suits. Recent breaches of security have been reported to affect hundreds of millions of individuals, and the resulting legal costs and damages to be paid out by affected companies have also been significant.

Federal and State Agencies

In addition to specific laws and regulations, cloud providers and users should be aware of the "common law of privacy and security," the nickname given to the body of consent orders that have been published by federal and state government agencies at the conclusion of their investigations into security incidents and events.

For almost 20 years, U.S. government agencies, such as the Federal Trade Commission (FTC) and the State Attorneys General have used their power under Federal or state "unfair and deceptive practices" acts to conduct enforcement actions against companies whose privacy or security practices are inconsistent with claims made in their public disclosures, making their practices unfair or deceptive. The numerous consent decrees issued by the FTC in enforcement cases under **Section 5 of the FTC Act: Unfair or Deceptive Acts or Practices**— or by state attorneys general in similar cases under their states' unfair and deceptive practices act— at the conclusion of many of these security investigations provide important guidance on the views and objectives of the Federal or State agencies regarding the collection, use and protection of personal information.

3.1.2 Contracts and Provider Selection

Even if a specific activity is not regulated, cloud customers may have a contractual obligation to protect the personal information of their own clients, contacts or employees to ensure data is not used for secondary purposes, and is not disclosed to, or shared with, third parties. This obligation may stem, for example, from the Terms and Conditions and Privacy Statement that a company posts on its website, or from contracts that the company has executed with third parties. For example, a data processor may be bound by the terms of its Services Agreement to process personal data only for certain purposes. Alternatively, the company may have entered into contracts (such as service agreements) with its customers, in which it has made specific commitments to protect the data (personal data or company data), limit its use, insure its security, use encryption, etc. The organization must guarantee that, when data in its custody is hosted in the cloud, it will have the continued ability to meet the promises and commitments that it made in its privacy notice(s) or other contracts. Data in the cloud must be used only for the purposes for which it was collected.

If the privacy notice allows individual data subjects to have access to their personal data, and to have this information modified or deleted, the cloud service provider must also allow these access, modification, and deletion rights to be exercised to the same extent as it would in a non-cloud relationship.

When data or operations are transferred to a cloud, the responsibility for protecting and securing the data typically remains with the collector or custodian of that data, even if in some circumstances

this responsibility may be shared with others. Even when it relies on a third party to host or process its data, the custodian of the data remains liable for any loss, damage, or misuse of the data. It is therefore prudent, and may be required by law or regulation, that the data custodian and the cloud provider enter into a formal written agreement that clearly defines the roles, the expectations of the parties, and the allocation of the many responsibilities that are attached to the data at stake. Such an agreement should also clearly identify the permitted and prohibited uses of the data, and the measures to be taken if the data were stolen or compromised.

The laws, regulations, standards and the related best practices discussed above also require data custodians to ensure that these obligations will be fulfilled by conducting due diligence (before execution of the contract) or security audits (during performance of the contract).

3.1.2.1 Internal Due Diligence

Companies are the custodians of data entrusted to them. As seen above, numerous laws, regulations and contracts prohibit, restrict and limit disclosure and transfer of data to a third party. For example, health information protected under HIPAA cannot be transferred to a third party or "business associate" without imposing specific obligations on that associate. In addition, if data originates abroad, it is likely that there are significant obstacles to its transfer across borders into a country that does not provide "adequate protection" to privacy rights and personal data.

Before entering into a cloud computing arrangement, both the cloud service vendor and the cloud customer should evaluate respective practices, needs and restrictions to identify relevant legal barriers and compliance requirements. For example, a cloud customer should determine whether its business model allows for the use of cloud computing services, and under which conditions. The nature of its business might be such that it is restricted by law from relinquishing control of company data. A cloud vendor may find it prudent to evaluate in advance the cost of doing business in certain markets that might be subject to legal requirements with which the vendor is unfamiliar.

A cloud customer should investigate whether it has entered into any confidentiality agreements or data use agreements that might restrict the transfer of data to third parties, even if these third parties are service providers. If the company, or potential cloud customer, has signed a confidentiality agreement to protect personal information or trade secrets, this agreement probably prohibits hiring a subcontractor without prior permission of the data owner. A data use agreement to which the company is a party may require the consent of a customer if the company plans to subcontract the processing of the customer's data to a third party. That restriction would in most cases apply to transfers to a cloud service provider. Under these circumstances, moving data to a cloud without the prior permission of the customer (data owner) would cause a breach in the data use agreement with that customer.

In other circumstances, the data processed by the company might be so sensitive or confidential that it should not be transferred to a cloud service, or the transfer might require significant precautions. This might be the case, for example, for files that pertain to high stakes projects such as R&D (Research & Development) road maps, or plans for an upcoming IPO (Initial Public Offering), merger, or acquisition.

3.1.2.2 Monitoring, Testing, and Updating

The cloud environment is not static. It evolves and involved parties must adapt. Periodic monitoring, testing, and evaluation of cloud services are recommended in order to insure required privacy and security measures are followed. Without periodic testing of cloud services, control efficacy may be compromised in an undetected fashion.

In addition, the legal, regulatory, and technical landscape in which any company operates changes at a rapid pace. New security threats, new laws, and new compliance requirements must be addressed promptly. Both cloud clients and cloud providers must keep abreast of relevant legal, regulatory, contractual, and other requirements, and ensure that both their operations remain compliant with applicable laws and regulations, and that the security measures in place continue to evolve as new technologies emerge.

3.1.2.3 External Due Diligence

Before entering into any contract, a critical part of due diligence must be to request and review all relevant aspects of the operations of the other party—in this case, that of the proposed cloud provider or vendor. A purchaser of cloud services needs to ensure that it understands the particular application or service it is contemplating acquiring. The extent of the due diligence and the time invested in it will depend upon the circumstances. The process may take a day, a week or a month depending on the specific needs of the customer, the nature of the data to be processed, the sensitivity and intensity of the processing, and other factors that would make a particular operation routine or highly sensitive.

Thus, depending on the nature of the proposed project, the due diligence may involve evaluating the nature and completeness of the services provided, the reputation for quality or stability of the service, the availability of a certain level of support or maintenance, the responsiveness of customer service, the speed of the network, and the location of the data centers. Interviewing customers may provide valuable insight. Reviewing reports of litigation filed against cloud providers and conducting online searches to evaluate a vendor's reputation may also be eye-opening.

In most cases, the cloud customer will want to evaluate at least the applicable service level, end-user and legal agreements; privacy policies; security disclosures; and proof of compliance with applicable legal requirements (e.g., registration requirements) to ensure the conditions stated by the cloud provider are suitable for the customer's organization. Depending on the expected depth and intensity of the due diligence, issues to be investigated may include the following:

- Will the service be reliable and easy to use?
- How will the servers be used to process data?
- How will the service operate and be provided?
- Will data be collocated with other customers' data?
- How will data be protected from intrusion or disasters?
- How will the price evolve over time?
- Will the cloud vendor meet the company's computing and access needs?
- Will the cloud vendor remain in business for the next few years? What is its financial profile?
- What service levels will be offered?

- What security measures are used?
- What will happen in the event of a security breach?

Reviewing all terms and conditions of the cloud services agreement (including all annexes, schedules and appendices) is good due diligence for any new project. It is especially critical for cloud computing, as some vendor terms and conditions will be non-negotiable. In those instances, the customer needs to make an informed decision to use or not use a provider.

3.1.2.4 Contract Negotiations

Cloud contracts are intended to accurately describe the understanding of all parties. Numerous precautions and measures can be taken by the parties to reduce exposure to legal, commercial and reputational risk in connection with the use of cloud services.

The proposed contract should always be reviewed carefully, even if one is told that it is not negotiable. For one thing, it might actually be possible to negotiate changes. Even if it is not possible to do so, each purchaser of cloud services should understand the consequences and implications of the engagement it is making. A contract that cannot be negotiated is likely to lack some of the protections that the typical customer would need. In this case, the customer should weigh the risks from foregoing these protections against potential benefits.

3.1.2.5 Reliance on Third-Party Audits and Attestations

Audits and compliance are covered in more detail in Domain 4, but two considerations may affect contractual and legal/regulatory requirements. In cloud computing, third-party audits and attestations are frequently used to assure compliance with aspects of the cloud provider's infrastructure, allowing a customer to build their own compliant services on top of the cloud platform. It is critical for a provider to publish, and a customer to evaluate, the scope of the assessment, and which features and services are included in the assessment.

For example, a cloud provider's newest storage offering may not be HIPAA-compliant (and thus the provider may not be willing to sign a HIPAA Business Associate Agreement (BAA) covering it), even though many of its other service offerings are able to be used in a HIPAA-compliant fashion.

3.1.3 Electronic Discovery

U.S. rules around "discovery"—the process by which an opposing party obtains private documents for use in litigation—cover a wide range of potential documents. In particular, discovery need not be limited to documents known at the outset to be admissible as evidence in court; instead, discovery will apply to all documents reasonably calculated to lead to admissible evidence (evidence that is both relevant and probative). See **Rule 26, Federal Rules of Civil Procedure** (FRCP).

In recent years, many litigants have deleted, lost or modified evidence that was detrimental to their case. In these cases, the Federal Rules of Civil Procedure allow, among other penalties, money to be awarded to the side not responsible for the destruction; in some cases, the jury may be given

an instruction on an "adverse inference" (where a jury is instructed to assume that the destroyed evidence contains the worst possible information for the party that destroyed it). See Rule 37, FRCP. As a result of the ongoing litigation in this area, the FRCP have been changed to clarify the obligations of the parties, especially in the case of electronically stored information (ESI).

Since the cloud will become the repository of most ESI needed in litigation or an investigation, cloud service providers and their clients must carefully plan how they will be able to identify all documents that pertain to a case, in order to be able to fulfill the stringent requirements imposed by FRCP 26 with regard to ESI, and the state equivalents to these laws. In this regard, the cloud service client and provider need to consider the following issues in matters when a client is subject to a discovery request and potentially relevant data exists with the cloud provider.

3.1.3.1 Possession, Custody and Control

In most jurisdictions in the United States, a party's obligation to produce relevant information is limited to documents and data within its possession, custody or control. Hosting relevant data via a third party, such as a cloud provider, generally does not obviate a party's obligation to produce information. However, not all data hosted by a cloud provider may be in the control of a client (e.g., disaster recovery systems, or certain metadata created and maintained by the cloud provider to operate its environment). Distinguishing the data that is and is not available to the client may be in the interest of both the client and provider at the outset of a relationship. The obligations of the cloud service provider as cloud data handler with regard to the production of information in response to legal process is an issue left to each jurisdiction to resolve.

3.1.3.2 Relevant Cloud Applications and Environment

On occasion, an actual cloud application or environment could itself be relevant to resolving a dispute. In these circumstances, the application and environment will likely be outside the control of the client and require that a subpoena or other discovery process be served on the provider directly.

3.1.3.3 Searchability and E-Discovery Tools

In a cloud environment, a client may not be able to apply or use e-discovery tools that it uses in its own environment. Moreover, a client may not have the ability or administrative rights to search or access all the data hosted in the cloud. For example, where a client could access multiple employees' e-mail accounts on its own server at once, it may not have this ability with e-mail accounts hosted in the cloud. As such, clients need to account for the potential additional time and expense this limited access will cause. To the extent the customer is able to negotiate or supplement the cloud service agreement, this issue could be addressed ahead of time. Otherwise, the cloud customer may have no option other than to address issues on a case-by-case basis; and might therefore have to pay for additional services from the cloud provider.

3.1.3.4 Preservation

Depending on the cloud service and deployment model that a client is using, preservation in the cloud can be similar to preservation in other IT infrastructures, or it can be significantly more complex.

In the United States, a party is generally obligated to undertake reasonable steps to prevent the destruction or modification of data in its possession, custody or control that it knows, or reasonably should know, is relevant either to pending or reasonably anticipated litigation or a government investigation. (This is often referred to as a "litigation hold" on document destruction.) These concerns are addressed broadly by Federal Rule of Civil Procedure 37, though there are myriad jurisdictional rulings that apply to potential litigants. In the European Union, information preservation is governed under Directive 2006/24/EC of the European Parliament and of the Council of 15 March 2006. Japan, South Korea and Singapore have similar data protection initiatives. In South America, Brazil and Argentina have the Azeredo Bill and the Argentina Data Retention Law of 2004, Law No. 25.873, respectively.

3.1.3.5 Data Retention Laws and Record Keeping Obligations

In addition to data preservation obligations resulting from U.S. laws regarding e-discovery, companies need to be aware that data retention laws require covered entities to retain data for a certain period of time.

Costs and Storage: Preservation can require that large volumes of data be retained for extended periods. Customers should consider these questions and determine the risk tolerated before moving to the cloud:

- What are the ramifications of retaining data under the service level agreement (SLA)?
- What happens if the preservation requirements outlast the terms of the SLA?
- If the client preserves the data in place, who pays for the extended storage, and at what cost?
- Does the client have the storage capacity under its SLA?
- Can the client effectively download the data in a forensically sound manner so it can be preserved off-line or near-line?

Scope of Preservation: A requesting party is entitled only to data hosted in the cloud that contains, or is reasonably calculated to lead to, relevant, probative information for the legal issue at hand. The party is not entitled to all the data in the cloud or in the application. (The issue of precise limits is likely to be resolved in litigation.) However, if the client does not have the ability to preserve only the relevant information or data in a granular way, it may be required to over-preserve in order to secure reasonable preservation, depending on the investigation. The over-preserved information is then examined for a determination of what must and must not be turned over as part of the discovery process. This process, referred to as a document review or privilege review, may be undertaken by client paid attorney staff or, in some cases, by emerging expert systems. How to sort the ever-more-voluminous quantities of information that may be produced by discovery is an ongoing area of both legal and technical research.

Dynamic and Shared Storage: The burden of preserving data in the cloud may be relatively modest if the client has space to hold it in place, if the data is relatively static, and if the people with access are limited and know to preserve the data. However, in a cloud environment that programmatically modifies or purges data, or one where the data is shared with people unaware of the need to preserve, preservation can be more difficult. After a client determines that such data is relevant and needs to be preserved, the client may need to work with the provider to determine a reasonable way to preserve such data.

3.1.3.6 Collection

Because of the potential lack of administrative control a client has over its data in the cloud, collection from the cloud can be more difficult, more time-consuming and more expensive than from behind a client's firewall. In particular, a client may not have the same level of visibility across its cloud data, and it may have more difficulty comparing the data it has collected with the data in the cloud to determine that export was reasonably complete and accurate.

Access and Bandwidth: In most cases, a client's access to its data in the cloud will be determined by its SLA. This may limit its ability to collect large volumes of data quickly and in a forensically sound manner (i.e., with all reasonably relevant metadata preserved). Clients and cloud providers should consider this issue at the outset of their relationship, and establish a protocol (and cost) for extraordinary access in the case of litigation. Absent these agreements, clients are responsible for the extra time and cost implicated by collection in the cloud when making representations to requesting parties and courts. Note that FRCP 26(b)(2)(B) excuses a litigant who is able to show that the information requested is not reasonably accessible.

However, a court may nonetheless order discovery from such sources if the requesting party is able to show why this information is needed and may not be obtained otherwise.

In a related issue, a client's right of access may provide them access to a full range of data, but not provide them the degree of functionality that would best assist them in a given situation. For example, a client may have access to three years of retail transactional data, but may only be able to download data two weeks at a time because of functionality constraints. Moreover, a client may not only have view of limited metadata. It is prudent for a client to learn what is possible with the tools available before it becomes necessary to use them as a part of active litigation.

Forensics: Bit-by-bit imaging of a cloud data source is generally difficult or impossible. For obvious security reasons, providers are reluctant to allow access to their hardware, particularly in a multi-tenant environment where a client could gain access to other clients' data. Even in a private cloud, forensics may be extremely difficult, and clients may need to notify opposing counsel or the courts of these limitations. (Again, FRCP 26(b)(2)(B) may provide relief from such undue burdens.) Luckily, this type of forensic analysis is rarely warranted in cloud computing, because the environment often consists of a structured data hierarchy or virtualization that does not provide significant additional relevant information in a bit-by-bit analysis.

Reasonable Integrity: A client subject to a discovery request should undertake reasonable steps to validate that its collection from its cloud provider is complete and accurate, especially where ordinary business procedures for the request are unavailable and litigation-specific measures are being used to obtain the information. This process is separate from verifying that the data stored in the cloud is accurate, authenticated or admissible.

Limits to Accessibility: Due to differences in how data is stored, and the access rights and privileges available to the owner of the data, there are cases where a cloud customer may not be able to access all their data stored in a cloud. The cloud customer and cloud provider may have to analyze the request for information and the pertinent data structures for relevance, materiality,

proportionality, or accessibility, when responding to a discovery request.

3.1.3.7 Direct Access

Outside the cloud environment, a requesting party's direct access to a responding party's IT environment is not generally favored. (It does happen from time to time. In fact, some courts have been willing to allow no-notice seizures of IT equipment for the purpose of evidence preservation in civil cases, including employment disputes.) In the cloud environment, it is even less favored and may be impossible as a forensic analysis may be impossible. Some cloud providers may not be able to provide direct access, because the hardware and facilities are outside its possession, custody or control, and a requesting party would need to negotiate directly with the provider for such access.

3.1.3.8 Native Production

Cloud service providers often store data in highly proprietary systems and applications that clients do not control. Generally, ESI is expected to be produced in standard formats (such as PDF for electronic documents), unless information lost by conversion (such as metadata) is relevant to the dispute. Data in its cloud-native format may be useless to the requesting party. In these circumstances, it may be best for all concerned—requesting party, producing party and provider—that the relevant information be exported using standard protocols within the cloud environment, with due care given to preserving relevant information.

3.1.3.9 Authentication

Authentication in this context refers to forensic authentication of data admitted into evidence. (This should not be confused with user authentication, which is a component of Identity Management.) Storing data in the cloud does not affect the authentication of data to determine if it should be admitted into evidence. The question is whether the document is what it purports to be. For example, an e-mail is no more or less authentic because it was stored behind a company's firewall or was stored in the cloud; the question is whether it was stored with integrity, such that the court can trust that it has not been altered since it was created. Absent other evidence, such as tampering or hacking, documents should not be considered more or less admissible or credible merely because they were created or stored in the cloud.

3.1.3.10 Cooperation Between Provider and Client in E-Discovery

It is in the best interests of both providers and clients to consider the complications caused by discovery at the beginning of their relationship and to account for it in their SLAs. Providers may want to consider designing their cloud offerings to include discovery services to attract clients ("Discovery by Design"). In any event, clients and providers should consider including an agreement to reasonably cooperate with each other in the event of discovery requests against either.

3.1.3.11 Response to a Subpoena or Search Warrant

Should a cloud service provider receive, from a third party, a request to provide information; this may be in the form of a subpoena, a warrant, or a court order in which access to the client data is

demanded. The client may want to have the ability to fight the request for access in order to protect the confidentiality of their data. To this end, the cloud service agreement should require the cloud service provider to notify the customer that a subpoena was received and give the company time to fight the request for access.

The cloud service provider might be tempted to reply to the request by opening its facilities and providing the requester with whatever they request. Before doing so, the cloud service provider should ensure, in consultation with counsel, that the request is legal and solid. The cloud service provider should carefully analyze the request before disclosing information in its custody, and consider whether it can meet its obligations to its clients when releasing information. In some cases, a provider may be better able to serve the needs of its clients by fighting an overbroad or otherwise problematic demand for information.

3.1.3.12 More Information

For more reading on discovery and electronically stored information, there are a wide variety of sources. One that may be of interest is the Sedona Conference, a nonprofit, research and educational institute that has for several years made influential recommendations about the handling of ESI, which have in turn shaped this emerging area of law. Note, however, that their recommendations do not themselves carry the force of law.

3.2 Recommendations

- Cloud customers should understand the relevant legal and regulatory frameworks, as well as contractual requirements and restrictions that apply to the handling of their data or data in their custody, and the conduct of their operations, before moving systems and data to the cloud.
- Cloud providers should clearly and conspicuously disclose their policies, requirements and capabilities, including all terms and conditions that apply to the services they provide.
- Cloud customers should conduct a comprehensive evaluation of a proposed cloud service provider before signing a contract, and should regularly update this evaluation and monitor the scope, nature and consistency of the services they purchase.
- Cloud providers should publish their policies, requirements and capabilities to meet legal obligations for customers, such as electronic discovery.
- Cloud customers should understand the legal implications of using particular cloud providers and match those to their legal requirements.
- Cloud customers should understand the legal implications of where the cloud provider physically operates and stores information.
- Cloud customer should decide whether to choose where their data will be hosted, if the option is available, to comply with their own jurisdictional requirements.
- Cloud customers and providers should have a clear understanding of the legal and technical requirements to meet any electronic discovery requests.
- Cloud customers should understand that click-through legal agreements to use a cloud service do not negate requirements for a provider to perform due diligence.

DOMAIN 4
Compliance and Audit Management

4.0 Introduction

Organizations face new challenges as they migrate from traditional data centers to the cloud. Delivering, measuring, and communicating compliance with a multitude of regulations across multiple jurisdictions are among the largest of these challenges. Customers and providers alike need to understand and appreciate the jurisdictional differences and their implications on existing compliance and audit standards, processes, and practices. The distributed and virtualized nature of cloud computing requires significant adjustment from approaches based on definite and physical instantiations of information and processes.

In addition to providers and customers, regulators and auditors are also adjusting to the new world of cloud computing. Few existing regulations were written to account for virtualized environments or cloud deployments. A cloud user can be challenged to show auditors that the organization is in compliance. Understanding the interaction of cloud computing and the regulatory environment is a key component of any cloud strategy. Cloud customers, auditors, and providers must consider and understand the following:

- Regulatory implications for using a particular cloud service or provider, giving particular attention to any cross-border or multi-jurisdictional issues when applicable.
- Assignment of compliance responsibilities between the provider and customer, including indirect providers (i.e., the cloud provider of your cloud provider). This includes the concept of compliance inheritance where a provider may have parts of their service certified as compliant which removes this from the audit scope of the customer, but the customer is still responsible for the compliance of everything they build on top of the provider.
- Provider capabilities for demonstrating compliance, including document generation, evidence production, and process compliance, in a timely manner.

Some additional cloud-specific issues to pay particular attention to include:

- The role of provider audits and certifications and how those affect customer audit (or assessment) scope.
- Understanding which features and services of a cloud provider are within the scope of which

audits and assessments.
- Managing compliance and audits over time.
- Working with regulators and auditors who may lack experience with cloud computing technology.
- Working with providers who may lack audit and or regulatory compliance experience.

4.1 Overview

Achieving and maintaining compliance with a plethora of modern regulations and standards is a core activity for most information security teams and a critical tool of governance and risk management. So much so that the tools and teams in this realm have their own acronym: GRC, for governance, risk, and compliance. Although very closely related with audits — which are a key mechanism to support, assure, and demonstrate compliance — there is more to compliance than audits and more to audits than using them to assure regulatory compliance. For our purposes:

- Compliance validates awareness of and adherence to corporate obligations (e.g., corporate social responsibility, ethics, applicable laws, regulations, contracts, strategies and policies). The compliance process assesses the state of that awareness and adherence, further assessing the risks and potential costs of non-compliance against the costs to achieve compliance, and hence prioritize, fund, and initiate any corrective actions deemed necessary.
- Audits are a key tool for proving (or disproving) compliance. We also use audits and assessments to support non-compliance risk decisions.

This section discusses these interrelated domains individually to better focus on the implications cloud computing has on each.

4.1.1 Compliance

Information technology in the cloud (or anywhere really) is increasingly subject to a plethora of policies and regulations from governments, industry groups, business relationships, and other stakeholders. Compliance management is a tool of governance; it is how an organization assesses, remediates, and proves it is meeting these internal and external obligations.

Regulations, in particular, typically have strong implications for information technology and its governance, especially in terms of monitoring, management, protection, and disclosure. Many regulations and obligations require a certain level of security, which is why information security is so deeply coupled with compliance. Security controls are thus an important tool to assure compliance, and evaluation and testing of these controls is a core activity for security professionals. This includes assessments even when performed by dedicated internal or external auditors.

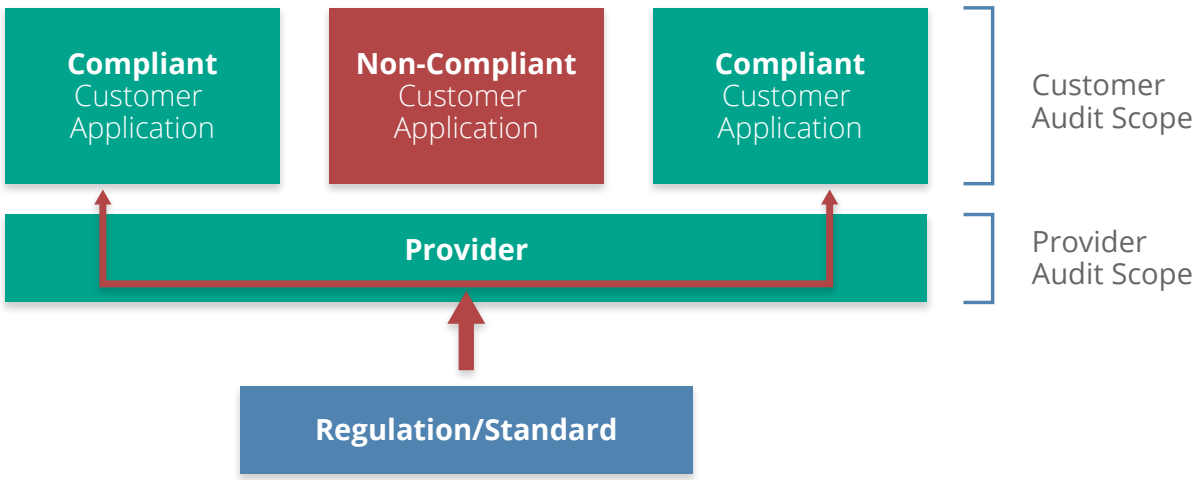
4.1.1.1 How Cloud Changes Compliance

As with security, compliance in the cloud is a shared responsibility model. Both the cloud provider and customer have responsibilities, but the customer is *always ultimately responsible for their own compliance*. These responsibilities are defined through contracts, audits/assessments, and specifics of the compliance requirements.

Cloud customers, particularly in public cloud, must rely more on third-party attestations of the provider to understand their compliance alignment and gaps. Since public cloud providers rely on economies of scale to manage costs they often will not allow customers to perform their own audits. Instead, similar to financial audits of public companies, they engage with a third-party firm to perform audits and issue attestations. Thus the cloud customer doesn't typically get to define the scope or perform the audit themselves. They will instead need to rely on these reports and attestations to determine if the service meets their compliance obligations.

Many cloud providers are certified for various regulations and industry requirements, such as PCI DSS, SOC1, SOC2, HIPAA, best practices/frameworks like CSA CCM, and global/regional regulations like the EU GDPR. These are sometimes referred to as *pass-through audits*. A pass-through audit is a form of *compliance inheritance*. In this model all or some of the cloud provider's infrastructure and services undergo an audit to a compliance standard. The provider takes responsibility for the costs and maintenance of these certifications. Provider audits, including pass-through audits, need to be understood within their limitations:

- They certify that the *provider* is compliant.
- It is still the responsibility of the customer to *build compliant applications and services on the cloud*.
- This means the provider's infrastructure/services are not within scope of a customer's audit/assessment. But everything the customer builds themselves is still within scope.
- The customer is still ultimately responsible for maintaining the compliance of what they build and manage. For example, if an IaaS provider is PCI DSS-certified, the customer can build their own PCI-compliant service on that platform and the provider's infrastructure and operations should be outside the *customer's* assessment scope. However, the customer can just as easily run afoul of PCI and fail their assessment if they don't design their own application running in the cloud properly.



> *With compliance inheritance the cloud provider's infrastructure is out of scope for a customer's compliance audit, but everything the customer configures and builds on top of the certified services is still within scope.*

Cloud compliance issues aren't merely limited to pass-through audits; the nature of cloud also creates additional differentiators.

Many cloud providers offer globally distributed data centers running off a central management console/platform. It is still the customer's responsibility to manage and understand where to deploy data and services and still maintain their legal compliance across national and international jurisdictions.

Organizations have the same responsibility in traditional computing, but the cloud dramatically reduces the friction of these potentially international deployments, e.g., a developer can potentially deploy regulated data in a non-compliant country without having to request an international data center and sign off on multiple levels of contracts, should the proper controls not be enabled to prevent this.

Not all features and services within a given cloud provider are necessarily compliant and certified/audited with respect to all regulations and standards. It is incumbent on the cloud provider to communicate certifications and attestations clearly, and for customers to understand the scopes and limitations.

4.1.2 Audit Management

Proper organizational governance naturally includes audit and assurance. Audits must be independently conducted and should be robustly designed to reflect best practice, appropriate resources, and tested protocols and standards. Before delving into cloud implications we need to define the scope of audit management related to information security.

Audits and assessments are mechanisms to document compliance with internal or external requirements (or identify deficiencies). Reporting needs to include a compliance determination, as well as a list of identified issues, risks, and remediation recommendations. Audits and assessments aren't limited to information security, but those related to information security typically focus on evaluating the effectiveness of security management and controls. Most organizations are subject to a mix of internal and external audits and assessments to assure compliance with internal and external requirements.

All audits have variable scope and statement of applicability, which defines what is evaluated (e.g., all systems with financial data) and to which controls (e.g., an industry standard, custom scope, or both). An attestation is a legal statement from a third party, which can be used as their statement of audit findings. Attestations are a key tool when evaluating and working with cloud providers since the cloud customer does not always get to perform their own assessments.

Audit management includes the management of all activities related to audits and assessments, such as determining requirements, scope, scheduling, and responsibilities.

4.1.2.1 How Cloud Changes Audit Management

Some cloud customers may be used to auditing third-party providers, but the nature of cloud computing and contracts with cloud providers will often preclude things like on-premises audits. Customers should understand that providers can (and often should) consider on-premises audits a

security risk when providing multitenant services. Multiple on-premises audits from large numbers of customers present clear logistical and security challenges, especially when the provider relies on shared assets to create the resource pools.

Customers working with these providers will have to rely more on third-party attestations rather than audits they perform themselves. Depending on the audit standard, actual results may only be releasable under a nondisclosure agreement (NDA), which means customers will need to enter into a basic legal agreement before gaining access to attestations for risk assessments or other evaluative purposes. This is often due to legal or contractual requirements with the audit firm, not due to any attempts and obfuscation by the cloud provider.

Cloud providers should understand that customers still need assurance that the provider meets their contractual and regulatory obligations, and should thus provide rigorous third-party attestations to prove they meet their obligations, especially when the provider does not allow direct customer assessments. These should be based on industry standards, with clearly defined scopes and the list of specific controls evaluated. Publishing certifications and attestations (to the degree legally allowed) will greatly assist cloud customers in evaluating providers. The Cloud Security Alliance STAR Registry offers a central repository for providers to publicly release these documents.

Some standards, such as SSAE 16, attest that documented controls work as designed/required. The standard doesn't necessarily define the *scope of controls*, so both are needed to perform a full evaluation. Also, attestations and certifications don't necessarily apply equally to all services offered by a cloud provider. Providers should be clear about which services and features are covered, and it is the responsibility of the customer to pay attention and understand the implications on their use of the provider.

Certain types of customer technical assessments and audits (such as a vulnerability assessment) may be limited in the provider's terms of service, and may require permission. This is often to help the provider distinguish between a legitimate assessment and an attack.

It's important to remember that attestations and certifications are point-in-time activities. An attestation is a statement of an "over a period of time" assessment and may not be valid at any future point. Providers must keep any published results current or they risk exposing their customers to risks of non-compliance. Depending on contracts, this could even lead to legal exposures to the provider. Customers are also responsible for ensuring they rely on current results and track when their providers' statuses change over time.

Artifacts are the logs, documentation, and other materials needed for audits and compliance; they are the evidence to support compliance activities. Both providers and customers have responsibilities for producing and managing their respective artifacts.

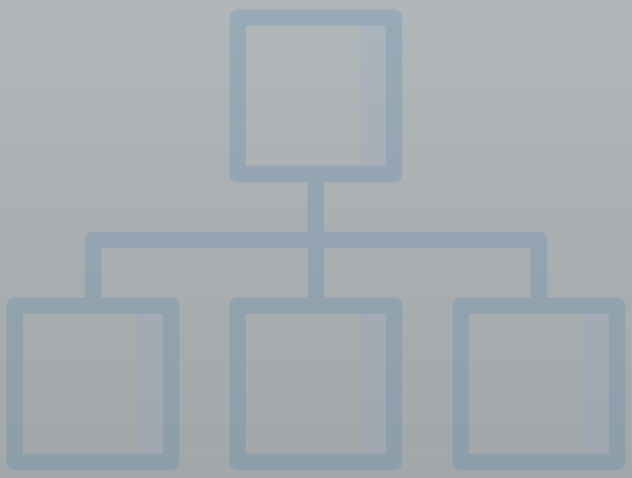
Collecting and maintaining artifacts of compliance will change when using a cloud provider.

Customers are ultimately responsible for the artifacts to support their own audits, and thus need to know what the provider offers, and create their own artifacts to cover any gaps. For example, by building more robust logging into an application since server logs on PaaS may not be available.

4.2 Recommendations

- Compliance, audit, and assurance should be continuous. They should not be seen as merely point-in-time activities, and many standards and regulations are moving more towards this model. This is especially true in cloud computing, where both the provider and customer tend to be in more-constant flux and are rarely ever in a static state.
- Cloud providers should:
  - Clearly communicate their audit results, certifications, and attestations with particular attention to:
    - The scope of assessments.
    - Which specific features/services are covered in which locations and jurisdictions.
    - How customers can deploy compliant applications and services in the cloud.
    - Any additional customer responsibilities and limitations.
  - Cloud providers must maintain their certifications/attestations over time and proactively communicate any changes in status.
  - Cloud providers should engage in continuous compliance initiatives to avoid creating any gaps, and thus exposures, for their customers.
  - Provide customers commonly needed evidence and artifacts of compliance, such as logs of administrative activity the customer cannot otherwise collect on their own.
- Cloud customers should:
  - Understand their full compliance obligations before deploying, migrating to, or developing in the cloud.
  - Evaluate a provider's third-party attestations and certifications and align those to compliance needs.
  - Understand the scope of assessments and certifications, including both the controls and the features/services covered.
  - Attempt to select auditors with experience in cloud computing, especially if pass-through audits and certifications will be used to manage the customer's audit scope.
  - Ensure they understand what artifacts of compliance the provider offers, and effectively collect and manage those artifacts.
    - Create and collect their own artifacts when the provider's artifacts are not sufficient.
  - Keep a register of cloud providers used, relevant compliance requirements, and current status. The Cloud Security Alliance Cloud Controls Matrix can support this activity.

DOMAIN 5
Information Governance



5.0 Introduction

The primary goal of information security is to protect the fundamental data that powers our systems and applications. As companies transition to cloud computing, the traditional methods of securing data are challenged by cloud-based architectures. Elasticity, multi-tenancy, new physical and logical architectures, and abstracted controls require new data security strategies. In many cloud deployments, users even transfer data to external — or even public — environments in ways that would have been unthinkable only a few years ago.

Managing information in the era of cloud computing is a daunting challenge that affects all organizations and requires not merely new technical protections but new approaches to fundamental governance. Although cloud computing has at least some effect on all areas of information governance, it particularly impacts compliance, privacy, and corporate policies due to the increased complexity in working with third parties and managing jurisdictional boundaries.

Definition of information/data governance:

> *Ensuring the use of data and information complies with organizational policies, standards and strategy — including regulatory, contractual, and business objectives.*

Our data is always subject to a range of requirements: some placed on us by others — like regulatory agencies or customers and partners — others that are self-defined based on our risk tolerance or simply how we want to manage operations. Information governance includes the corporate structures and controls we use to ensure we handle data in accordance with our goals and requirements.

There are numerous aspects of having data stored in the cloud that have an impact on information and data governance requirements.

- *Multitenancy:* Multitenancy presents complicated security implications. When data is stored in the public cloud, it's stored on shared infrastructure with other, untrusted tenants. Even in a private cloud environment, it is stored and managed on infrastructure that's shared across different business units, which likely have different governance needs.

- *Shared security responsibility:* With greater sharing of environments comes greater shared security responsibilities. Data is now more likely to be owned and managed by different teams or even organizations. So, it's important to recognize the difference between data custodianship and data ownership.
    - *Ownership*, as the name says, is about who owns the data. It's not always perfectly clear. If a customer provides you data, you might own it or they might still legally own it, depending on law, contracts, and policies. If you host your data on a public cloud provider you should own it, but that might again depend on contracts.
    - *Custodianship* refers to who is managing the data. If a customer gives you their personal information and you don't have the rights to own it, you are merely the custodian. That means you can only use it in approved ways. If you use a public cloud provider, they, likewise, become the custodian of the data, although you likely also have custodial responsibility depending on what controls you implement and manage yourself. Using a provider doesn't obviate your responsibility. Basically, the owner defines the rules (sometimes indirectly through regulation) and the custodian implements the rules. The lines and roles between owner and custodian are impacted by cloud infrastructure, particularly in the case of public cloud.

By hosting customer data in the cloud, we are introducing a third party into the governance model, the cloud provider.

- *Jurisdictional boundaries and data sovereignty:* Since cloud, by definition, enables broad network access, it increases the opportunities to host data in more locations (jurisdictions) and reduces the friction in migrating data. Some providers may not be as transparent about the physical location of the data, while in other cases additional controls may be needed to restrict data to particular locations.
- *Compliance, regulations, and privacy policies:* All of these may be impacted by cloud due to the combination of a third-party provider and jurisdictional changes, e.g., your customer agreement may not allow you to share/use data on a cloud provider, or may have certain security requirements (like encryption).
- *Destruction and removal of data:* This ties in to the technical capabilities of the cloud platform. Can you ensure the destruction and removal of data in accordance with policy?

When migrating to cloud, use it as an opportunity to revisit information architectures. Many of our information architectures today are quite fractured as they were implemented over sometimes decades in the face of ever-changing technologies. Moving to cloud creates a green field opportunity to reexamine how you manage information and find ways to improve things. Don't lift and shift existing problems.

5.1 Overview

Data/information governance means ensuring that the use of data and information complies with organizational policies, standards, and strategy. This includes regulatory, contractual, and business requirements and objectives. Data is different than information, but we tend to use them interchangeably. Information is data with value. For our purposes, we use both terms to mean the same thing since that is so common.

5.1.1 Cloud Information Governance Domains

We will not cover all of data governance, but we'll focus on where hosting in the cloud affects data governance. Cloud computing affects most data governance domains:

- *Information Classification.* This is frequently tied to compliance and affects cloud destinations and handling requirements. Not everyone necessarily has a data classification program, but if you do you need to adjust it for cloud computing.
- *Information Management Policies.* These tie to classification and the cloud needs to be added if you have them. They should also cover the different SPI tiers, since sending data to a SaaS vendor versus building your own IaaS app is very different. You need to determine what is allowed to go where in the cloud? Which products and services? With what security requirements?
- *Location and Jurisdiction Policies.* These have very direct cloud implications. Any outside hosting must comply with locational and jurisdictional requirements. Understand that internal policies can be changed for cloud computing, but legal requirements are hard lines. (See the Legal Domain for more information on this.) Make sure you understand that treaties and laws may create conflicts. You need to work with your legal department when handling regulated data to ensure you comply as best you can.
- *Authorizations.* Cloud computing requires minimal changes to authorizations, but see the data security lifecycle to understand if the cloud impacts.
- *Ownership.* Your organization is always responsible for data and information and that can't be abrogated when moving to the cloud.
- *Custodianship.* Your cloud provider may become custodian. Data hosted but properly encrypted is still under custodianship of the organization.
- *Privacy.* Privacy is a sum of regulatory requirements, contractual obligations, and commitments to customers (e.g. public statements). You need to understand the total requirements and ensure information management and security policies align.
- *Contractual controls.* This is your legal tool for extending governance requirements to a third party, like a cloud provider.
- *Security controls.* Security controls are the tool to implement data governance. They change significantly in cloud computing. See the Data Security and Encryption domain.

5.1.2 The Data Security Lifecycle

Although Information Lifecycle Management is a fairly mature field, it doesn't map well to the needs of security professionals. The Data Security Lifecycle is different from Information Lifecycle

Management, reflecting the different needs of the security audience. This is a summary of the lifecycle, and a complete version is available at http://www.securosis.com/blog/data-security-lifecycle-2.0. It is simply a tool to help understand the security boundaries and controls around data. It's not meant to be used as a rigorous tool for all types of data. It's a modeling tool to help evaluate data security at a high level and find focus points.

The lifecycle includes six phases from creation to destruction. Although it is shown as a linear progression, once created, data can bounce between phases without restriction, and may not pass through all stages (for example, not all data is eventually destroyed).

Create. Creation is the generation of new digital content, or the alteration/updating/modifying of existing content.

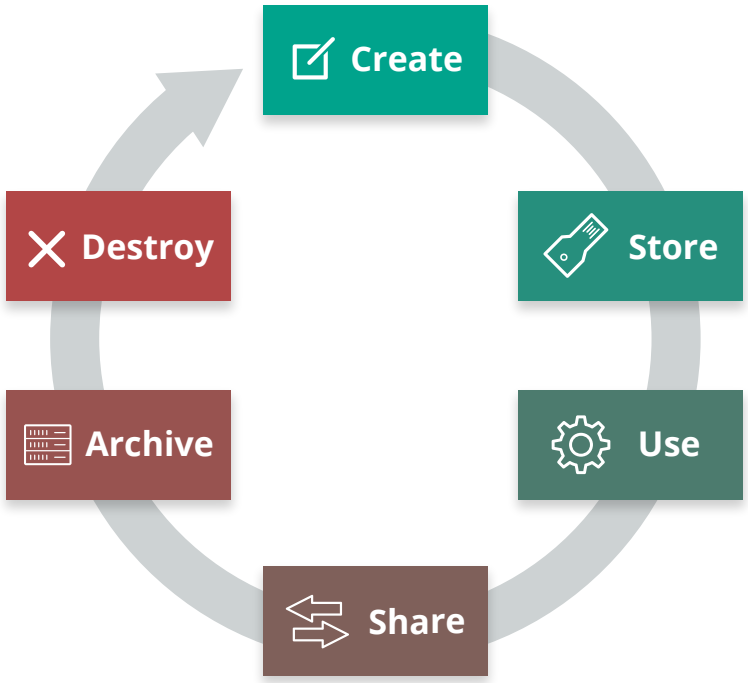
Store. Storing is the act committing the digital data to some sort of storage repository and typically occurs nearly simultaneously with creation.

Use. Data is viewed, processed, or otherwise used in some sort of activity, not including modification.

Share. Information is made accessible to others, such as between users, to customers, and to partners.

Archive. Data leaves active use and enters long-term storage.

Destroy. Data is permanently destroyed using physical or digital means (e.g., cryptoshredding).

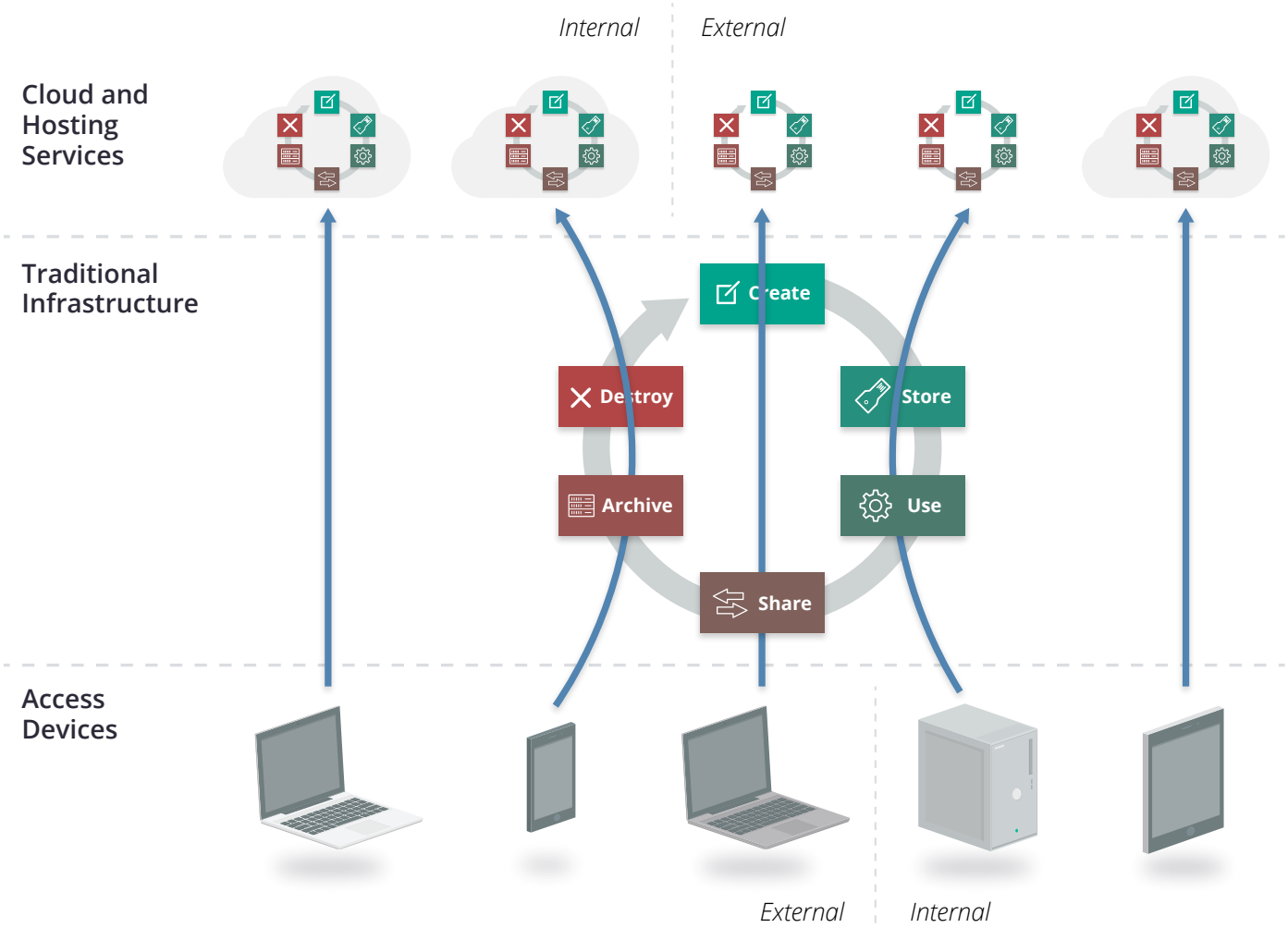


The Data Security Lifecycle

5.1.2.1 Locations and Entitlements

The lifecycle represents the phases information passes through but doesn't address its location or how it is accessed.

Locations: This can be illustrated by thinking of the lifecycle not as a single, linear operation, but as a series of smaller lifecycles running in different operating environments. At nearly any phase data can move into, out of, and between these environments.



Data is accessed and stored in multiple locations, each with its own lifecycle.

Due to all the potential regulatory, contractual, and other jurisdictional issues, it is extremely important to understand both the logical and physical locations of data.

Entitlements: When users know where the data lives and how it moves, they need to know who is accessing it and how. There are two factors here:

- Who accesses the data?
- How can they access it (device and channel)?

Data today is accessed using a variety of different devices. These devices have different security characteristics and may use different applications or clients.

5.1.2.2 Functions, Actors, and Controls

The next step identifies the functions that can be performed with the data, by a given actor (person or system) and a particular location.

Functions: There are three things we can do with a given datum:

- *Read.* View/read the data, including creating, copying, file transfers, dissemination, and other exchanges of information.
- *Process.* Perform a transaction on the data; update it; use it in a business processing transaction, etc.
- *Store.* Hold the data (in a file, database, etc.).

The table below shows which functions map to which phases of the lifecycle:

	Create	Store	Use	Share	Archive	Destroy
Read	X	X	X	X	X	X
Process	X		X			
Store		X			X	

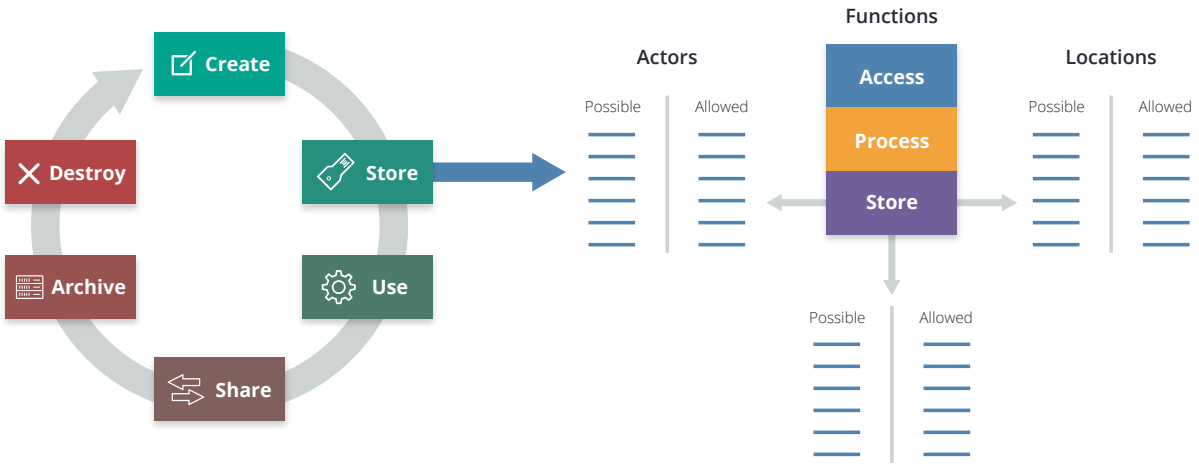
Table 1—Information Lifecycle Phases

An actor (person, application, or system/process, as opposed to the access device) performs each function in a location.

Controls: A control restricts a list of possible actions down to allowed actions. The table below shows one way to list the possibilities, which the user then maps to controls.

Function		Action		Location	
Possible	Allowed	Possible	Allowed	Possible	Allowed

Mapping the lifecycle to functions and controls.



Mapping the lifecycle to functions and controls.

5.2 Recommendations

- Determine your governance requirements for information before planning a transition to cloud. This includes legal and regulatory requirements, contractual obligations and other corporate policies. Your corporate policies and standards may need to be updated to allow a third party to handle data.
- Ensure information governance policies and practices extend to the cloud. This will be done through contractual and security controls.
- When needed, use the data security lifecycle to help model data handling and controls.
- Instead of lifting and shifting existing information architectures take the opportunity of the migration to the cloud to re-think and re-structure what is often the fractured approach used in existing infrastructure. Don't bring bad habits.

DOMAIN 6
Management Plane and Business Continuity

6.0 Introduction

The management plane is the single most significant security difference between traditional infrastructure and cloud computing. This isn't all of the metastructure (defined in Domain 1) but is the interface to connect with the metastructure and configure much of the cloud.

We always have a management plane, the tools and interfaces we use to manage our infrastructure, platforms, and applications, but cloud abstracts and centralizes administrative management of resources. Instead of controlling a data center configuration with boxes and wires, it is now controlled with API calls and web consoles.

Thus, gaining access to the management plane is like gaining unfettered access to your data center, unless you put the proper security controls in place to limit who can access the management plane and what they can do within it.

To think about it in security terms, the management plane consolidates many things we previously managed through separate systems and tools, and then makes them Internet-accessible with a single set of authentication credentials. This isn't a net loss for security — there are also gains — but it is most definitely different, and it impacts how we need to evaluate and manage security.

Centralization also brings security benefits. There are no hidden resources, you always know where everything you own is at all times, and how it is configured. This is an emergent property of both broad network access and metered service. The cloud controller always needs to know what resources are in the pool, out of the pool, and where they are allocated.

This doesn't mean that all the assets you put into the cloud are equally managed. The cloud controller can't peer into running servers or open up locked files, nor understand the implications of your specific data and information.

In the end, this is an extension of the shared responsibility model discussed in Domain 1 and throughout this Guidance. The cloud management plane is responsible for managing the assets of the resource pool, while the cloud user is responsible for how they configure those assets, and for

the assets they deploy into the cloud.

- The cloud provider is responsible for ensuring the management plane is secure and necessary security features are exposed to the cloud user, such as granular entitlements to control what someone can do even if they have management plane access.
- The cloud user is responsible for properly configuring their use of the management plane, as well as for securing and managing their credentials.

6.0.1 Business Continuity and Disaster Recovery in the Cloud

Business Continuity and Disaster Recovery (BC/DR) is just as important in cloud computing as it is for any other technology. Aside from the differences resulting from the potential involvement of a third-party provider (something we often deal with in BC/DR), there are additional considerations due to the inherent differences when using shared resources.

The three main aspects of BC/DR in the cloud are:

- Ensuring continuity and recovery within a given cloud provider. These are the tools and techniques to best architect your cloud deployment to keep things running if either what you deploy breaks, or a portion of the cloud provider breaks.
- Preparing for and managing cloud provider outages. This extends from the more constrained problems that you can architect around within a provider to the wider outages that take down all or some of the provider in a way that exceeds the capabilities of inherent DR controls.
- Considering options for portability, in case you need to migrate providers or platforms. This could be due to anything from desiring a different feature set to the complete loss of the provider if, for example, they go out of business or you have a legal dispute.

6.0.1.1 Architect for Failure

Cloud platforms can be incredibly resilient, but single cloud assets are typically less resilient than in the case of traditional infrastructure. This is due to the inherently greater fragility of virtualized resources running in highly-complex environments.

This mostly applies to compute, networking, and storage, since those allow closer to raw access, and cloud providers can leverage additional resiliency techniques for their platforms and applications that run on top of IaaS.

However, this means that cloud providers tend to offer options to improve resiliency, often beyond that which is attainable (for equivalent costs) in traditional infrastructure. For example, by enabling multiple "zones," where you can deploy virtual machines within an auto-scaled group that encompasses physically distinct data centers for high-availability. Your application can be balanced across zones so that if an entire zone goes down your application still stays up. This is quite difficult to implement in a traditional data center, where it typically isn't cost-effective to build multiple, isolated physical zones across which you can deploy a cross-zone, load-balanced application with automatic failover.

But this extra resiliency is only achievable if you architect to leverage these capabilities. Deploying your application all in one zone, or even on a single virtual machine in a single zone, is likely to be less resilient than deploying on a single, well-maintained physical server.

This is why "lift and shift" wholesale migration of existing applications without architectural changes can reduce resiliency. Existing applications are rarely architected and deployed to work with these resiliency options, yet straight-up virtualization and migration without changes can increase the odds of individual failures.

The ability to manage is higher with IaaS and much lower with SaaS, just like security. For SaaS, you rely on the cloud provider keeping the entire application service up. With IaaS, you can architect your application to account for failures, putting more responsibility in your hands. PaaS, as usual, is in the middle — some PaaS may have resiliency options that you can configure, while other platforms are completely in the hands of the provider.

Overall, a risk-based approach is key:

- Not all assets need equal continuity.
- Don't drive yourself crazy by planning for full provider outages just because of the perceived loss of control. Look at historical performance.
- Strive to design for RTOs and RPOs equivalent to those on traditional infrastructure.

6.1 Overview

6.1.1 Management Plane Security

The management plane refers to the interfaces for managing your assets in the cloud. If you deploy virtual machines on a virtual network the management plane is how you launch those machines and configure that network. For SaaS, the management plane is often the "admin" tab of the user interface and where you configure things like users, settings for the organization, etc.

The management plane controls and configures the metastructure (defined in Domain 1), and is also part of the metastructure itself. As a reminder, cloud computing is the act of taking physical assets (like networks and processors) and using them to build resource pools. Metastructure is the glue and guts to create, provision, and deprovision the pools. The management plane includes the interfaces for building and managing the cloud itself, but also the interfaces for cloud users to manage their own allocated resources of the cloud.

The management plane is a key tool for enabling and enforcing separation and isolation in multitenancy. Limiting who can do what with the APIs is one important means for segregating out customers, or different users within a single tenant. Resources are in the pool, out of the pool, and where they are allocated.

6.1.1.1 Accessing the Management Plane

APIs and web consoles are the way the management plane is delivered. Application Programming Interfaces allow for programmatic management of the cloud. They are the glue that holds the cloud's components together and enables their orchestration. Since not everyone wants to write programs to manage their cloud, web consoles provide visual interfaces. In many cases web consoles merely use the same APIs you can access directly.

Cloud providers and platforms will also often offer Software Development Kits (SDKs) and Command Line Interfaces (CLIs) to make integrating with their APIs easier.

- *Web consoles* are managed by the provider. They can be organization-specific [typically using Domain Name Server (DNS) redirection tied to federated identity]. For example, when you connect to your cloud file-sharing application you are redirected to your own "version" of the application after you log in. This version will have its own domain name associated with it, which allows you to integrate more easily with federated identity (e.g. instead of all your users logging in to "application.com" they log into "your-organization.application.com").

As mentioned, most web consoles offer a user interface for the same APIs that you can access directly. Although, depending on the platform or provider's development process, you may sometimes encounter a mismatch where either a web feature or an API call appear on one before the other.

APIs are typically **REST** for cloud services, since REST is easy to implement across the Internet. REST APIs have become the standard for web-based services since they run over HTTP/S and thus work well across diverse environments.

These can use a variety of authentication mechanisms, as there is no single standard for authentication in REST. HTTP request signing and OAuth are the most common; both of these leverage cryptographic techniques to validate authentication requests.

You still often see services that embed a password in the request. This is less secure and at higher risk for credential exposure. It's most often seen in older or poorly-designed web platforms that built their web interface first and only added consumer APIs later. If you do encounter this, you need to use dedicated accounts for API access if possible, in order to reduce the opportunities for credential exposure.

6.1.1.2 Securing the Management Plane

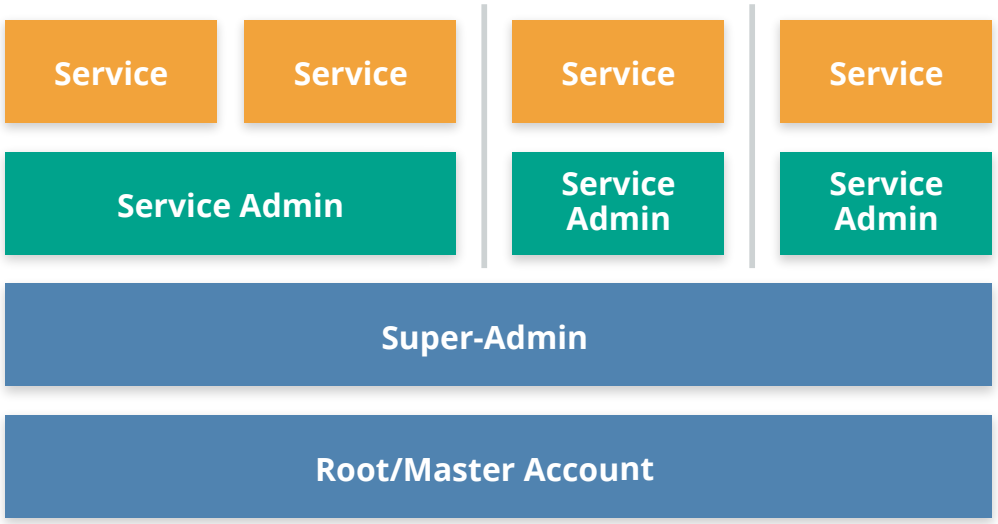
Identity and Access Management (IAM) includes identification, authentication, and authorizations (including access management). This is how you determine who can do what within your cloud platform or provider.

The specific options, configurations, and even concepts vary heavily between cloud providers and platforms. Each has their own implementation and may not even use the same definitions for things like "groups" and "roles."

No matter the platform or provider there is always an account owner with super-admin privileges to manage the entire configuration. This should be enterprise-owned (not personal), tightly locked down, and nearly never used.

Separate from the account-owner you can usually create super-admin accounts for individual admin use. Use these privileges sparingly; this should also be a smaller group since compromise or abuse of one of these accounts could allow someone to change or access essentially everything and anything.

Your platform or provider may support lower-level administrative accounts that can only manage parts of the service. We sometimes call these "service administrators" or "day to day administrators". These accounts don't necessarily expose the entire deployment if they are abused or compromised and thus are better for common daily usage. They also help compartmentalize individual sessions, so it isn't unusual to allow a single human administrator access to multiple service administrator accounts (or roles) so they can log in with just the privileges they need for that particular action instead of having to expose a much wider range of entitlements.



Examples of baseline cloud management plane user accounts including super-administrators and service administrators.

Both providers and consumers should consistently only allow the least privilege required for users, applications, and other management plane usage.

All privileged user accounts should use multi-factor authentication (MFA). If possible, *all* cloud accounts (even individual user accounts) should use MFA. It's one of the single most effective security controls to defend against a wide range of attacks. This is also true regardless of the service model: MFA is just as important for SaaS as it is for IaaS.

(See the IAM domain for more information on IAM and the role of federation and strong authentication, much of which applies to the cloud management plane.)

6.1.1.3 Management Plane Security When Building/Providing a Cloud Service

When you are responsible for building and maintaining the management plane itself, such as in a private cloud deployment, that increases your responsibilities. When you consume the cloud you only configure the parts of the management plane that the provider exposes to you, but when you are the cloud provider you obviously are responsible for everything.

Delving into implementation specifics is beyond the scope of this Guidance, but at a high level there are five major facets to building and managing a secure management plane:

- *Perimeter security:* Protecting from attacks against the management plane's components itself, such as the web and API servers. It includes both lower-level network defenses as well as higher-level defenses against application attacks.
- *Customer authentication:* Providing secure mechanisms for customers to authenticate to the management plane. This should use existing standards (like OAuth or HTTP request signing) that are cryptographically valid and well documented. Customer authentication should support MFA as an option or requirement.
- *Internal authentication and credential passing:* The mechanisms your own employees use to connect with the non-customer-facing portions of the management plane. It also includes any translation between the customer's authentication and any internal API requests. Cloud providers should always mandate MFA for cloud management authentication.
- *Authorization and entitlements:* The entitlements available to customers and the entitlements for internal administrators. Granular entitlements better enable customers to securely manage their own users and administrators. Internally, granular entitlements reduce the impact of administrators' accounts being compromised or employee abuse.
- *Logging, monitoring, and alerting:* Robust logging and monitoring of administrative is essential for effective security and compliance. This applies both to what the customer does in their account, and to what employees do in their day-to-day management of the service. Alerting of unusual events is an important security control to ensure that monitoring is actionable, and not merely something you look at after the fact. Cloud customers should ideally be able to access logs of their own activity in the platform via API or other mechanism in order to integrate with their own security logging systems.

6.1.2 Business Continuity and Disaster Recovery

Like security and compliance, BC/DR is a shared responsibility. There are aspects that the cloud provider has to manage, but the cloud customer is also ultimately responsible for how they use and manage the cloud service. This is especially true when planning for outages of the cloud provider (or parts of the cloud provider's service).

Also similar to security, customers have more control and responsibility in IaaS, less in SaaS, with PaaS in the middle.

BC/DR must take a risk-based approach. Many BC options may be cost prohibitive in the cloud, but may also not be necessary. This is no different than in traditional data centers, but it isn't unusual

to want to over-compensate when losing physical control. For example, the odds of a major IaaS provider going out of business or changing their entire business model are low, but this isn't all that uncommon for a smaller venture-backed SaaS provider.

- Ask the provider for outage statistics over time since this can help inform your risk decisions.
- Remember that capabilities vary between providers and should be included in the vendor selection process.

6.1.2.1 Business Continuity Within the Cloud Provider

When you deploy assets into the cloud you can't assume the cloud will always be there, or always work the way you expect. Outages and issues are no more or less common than with any other technology, although the cloud can be overall more resilient when the provider includes mechanisms to better enable building resilient applications.

This is a key point we need to spend a little more time on: As we've mentioned in a few places the very nature of virtualizing resources into pools typically creates less resiliency for any single asset, like a virtual machine. On the other hand, abstracting resources and managing everything through software opens up flexibility to more easily enable resiliency features like durable storage and cross-geographic load balancing.

There is a huge range of options here, and not all providers or platforms are created equal, but you shouldn't assume that "the cloud" as a general term is more or less resilient than traditional infrastructure. Sometimes it's better, sometimes it's worse, and knowing the difference all comes down to your risk assessment and how you use the cloud service.

This is why it is typically best to re-architect deployments when you migrate them to the cloud. Resiliency itself, and the fundamental mechanisms for ensuring resiliency, change. Direct "lift and shift" migrations are less likely to account for failures, nor will they take advantage of potential improvements from leveraging platform or service specific capabilities.

The focus is on understanding and leveraging the platform's BC/DR features. Once you make the decision to deploy in the cloud you then want to optimize your use of included BC/DR features before adding on any additional capabilities through third-party tools.

BC/DR must account for the entire logical stack:

- *Metastructure:* Since cloud configurations are controlled by software, these configurations should be backed up in a restorable format. This isn't always possible, and is pretty rare in SaaS, but there are tools to implement this in many IaaS platforms (including third-party options) using *Software-Defined Infrastructure*.

- *Software-Defined Infrastructure* allows you to create an infrastructure template to configure all or some aspects of a cloud deployment. These templates are then translated natively by the cloud platform or into API calls that orchestrate the configuration.

This should include controls such as IAM and logging, not merely architecture, network design, or service configurations.

- *Infrastructure:* As mentioned, any provider will offer features to support higher availability than can comparably be achieved in a traditional data center for the same cost. But these only work if you adjust your architecture. "Lifting and shifting" applications to the cloud without architectural adjustments or redesign will often result in lower availability.

  Be sure and understand the cost model for these features, especially for implementing them across the provider's physical locations/regions, where the cost can be high. Some assets and data must be converted to work across cloud locations/regions, for example, custom machine images used to launch servers. These assets must be included in plans.

- *Infostructure:* Data synchronization is often one of the more difficult issues to manage across locations, even if the actual storage costs are manageable. This is due to the size of data sets (vs. an infrastructure configuration) and keeping data in sync across locations and services, something that's often difficult even in a single storage location/system.

- *Applistructure:* Applistructure includes all of the above, but also the application assets like code, message queues, etc. When a cloud user builds their own cloud applications they're usually built on top of IaaS and/or PaaS, so resiliency and recovery are inherently tied to those layers. But Applistructure includes the full range of everything in an application.

  Understand PaaS limitations and lock-ins, and plan for the outage of a PaaS component. Platform services include a range of functions we used to manually implement in applications, everything from authentication systems to message queues and notifications. It isn't unusual for modern applications to even integrate these kinds of services from multiple different cloud providers, creating an intricate web.

  Discussing availability of the component/service with your providers is reasonable. For example, the database service from your infrastructure provider may not share the same performance and availability as their virtual machine hosting.

  When real-time switching isn't possible, design your application to gracefully fail in case of a service outage. There are many automation techniques to support this. For example, if your queue service goes down, that should trigger halting the front end so messages aren't lost.

  Downtime is always an option. You don't always need perfect availability, but if you do plan to accept an outage you should at least ensure you fail gracefully, with emergency downtime notification pages and responses. This may be possible using static stand-by via DNS redirection.

  "Chaos Engineering" is often used to help build resilient cloud deployments. Since everything cloud is API-based, Chaos Engineering uses tools to selectively degrade portions of the cloud to continuously test business continuity.

  This is often done in production, not just test environment, and forces engineers to assume

failure instead of viewing it as only a possible event. By designing systems for failure you can better absorb individual component failures.

6.1.2.2 Business Continuity for Loss of the Cloud Provider

It is always possible that an entire cloud provider or at least a major portion of its infrastructure (such as one specific geography) can go down. Planning for cloud provider outages is difficult, due to the natural lock-in of leveraging a provider's capabilities. Sometimes you can migrate to a different portion of their service, but in other cases an internal migration simply isn't an option, or you may be totally locked in.

Depending on the history of your provider, and their internal availability capabilities, accepting this risk is often a legitimate option.

Downtime may be another option, but it depends on your recovery time objectives (RTO). However, some sort of static stand-by should be available via DNS redirection. Graceful failure should also include failure responses to API calls, if you offer APIs.

Be wary of selecting a secondary provider or service if said service may also be located or reliant on the same provider. It doesn't do you any good to use a backup storage provider if said provider happens to be based on the same infrastructure provider.

Moving data between providers can be difficult, but might be easy compared to moving metastructure, security controls, logging, and so on, which may be incompatible between platforms.

SaaS may often be the biggest provider outage concern, due to total reliance on the provider. Scheduled data extraction and archiving may be your only BC option outside of accepting downtime. Extracting and archiving to another cloud service, especially IaaS/PaaS, may be a better option than moving it to local/on-premises storage. Again, take a risk-based approach that includes the unique history of your provider.

Even if you have your data, you must have an alternate application that you know you can migrate it into. If you can't use the data, you don't have a viable recovery strategy.

Test, test, and test. This may often be easier than in a traditional data center because you aren't constrained by physical resources, and only pay for use of certain assets during the life of the test.

6.1.2.3 Business Continuity For Private Cloud and Providers

This is completely on the provider's shoulders, and BC/DR includes everything down to the physical facilities. RTOs and RPOs will be stringent, since if the cloud goes down, everything goes down.

If you are providing services to others, be aware of contractual requirements, including data residency, when building your BC plans. For example, failing over to a different geography in a different legal jurisdiction may violate contracts or local laws.

6.2 Recommendations

- Management plane (metastructure) security
  - Ensure there is strong perimeter security for API gateways and web consoles.
  - Use strong authentication and MFA.
  - Maintain tight control of primary account holder/root account credentials and consider dual-authority to access them.
    - Establishing multiple accounts with your provider will help with account granularity and to limit blast radius (with IaaS and PaaS).
  - Use separate super administrator and day-to-day administrator accounts instead of root/primary account holder credentials.
  - Consistently implement least privilege accounts for metastructure access.
    - This is why you separate development and test accounts with your cloud provider.
  - Enforce use of MFA whenever available.
- Business continuity
  - Architecture for failure.
  - Take a risk-based approach to everything. Even when you assume the worst, it doesn't mean you can afford or need to keep full availability if the worst happens.
  - Design for high availability within your cloud provider. In IaaS and PaaS this is often easier and more cost effective than the equivalent in traditional infrastructure.
    - Take advantage of provider-specific features.
    - Understand provider history, capabilities, and limitations.
    - Cross-location should always be considered, but beware of costs depending on availability requirements.
      - Also ensure things like images and asset IDs are converted to work in the different locations.
    - Business Continuity for metastructure is as important as that for assets.
  - Prepare for graceful failure in case of a cloud provider outage.
    - This can include plans for interoperability and portability with other cloud providers or a different region with your current provider.
  - For super-high-availability applications, start with cross-location BC before attempting cross-provider BC.
  - Cloud providers, including private cloud, must provide the highest levels of availability and mechanisms for customers/users to manage aspects of their own availability.

DOMAIN 7
Infrastructure Security

7.0 Introduction

Infrastructure security is the foundation for operating securely in the cloud. "Infrastructure" is the glue of computers and networks that we build everything on top of. For the purposes of this Guidance we start with compute and networking security, which also encompass workload and hybrid cloud. Although storage security is also core to infrastructure, it is covered in full depth in Domain 11: Data Security and Encryption. This domain also includes the fundamentals for private cloud computing. It does not include all the components of traditional data center security that are already well covered by existing standards and guidance.

Infrastructure security encompasses the lowest layers of security, from physical facilities through the consumer's configuration and implementation of infrastructure components. These are the fundamental components that everything else in the cloud is built from, including compute (workload), networking, and storage security.

For purposes of the CSA Guidance we are focusing on cloud-specific aspects of infrastructure security. There are already incredibly robust bodies of knowledge and industry standards for data center security that cloud providers and private cloud deployments should reference. Consider this Guidance a layer on top of those extensive and widely available materials. Specifically, this Domain discusses two aspects: cloud considerations for the underlying infrastructure, and security for virtual networks and workloads.

7.1 Overview

In cloud computing there are two macro layers to infrastructure:

- The fundamental resources pooled together to create a cloud. This is the raw, physical and logical compute (processors, memory, etc.), networks, and storage used to build the cloud's resource pools. For example, this includes the security of the networking hardware and software used to create the network resource pool.

- The virtual/abstracted infrastructure managed by a cloud user. That's the compute, network, and storage assets that they use from the resource pools. For example, the security of the virtual network, as defined and managed by the cloud user.

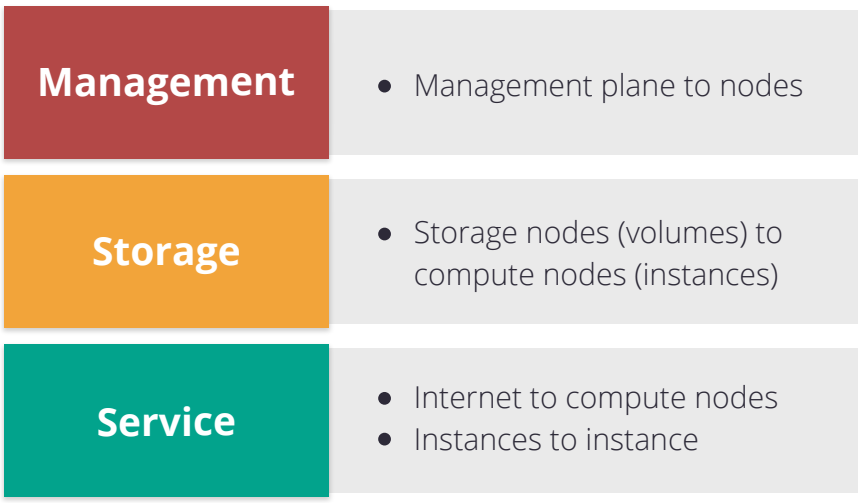
The information and advice in this domain primarily focuses on the second macro layer, infrastructure security for the cloud user. Infrastructure security that's more fundamental for cloud providers, including those who manage private clouds, is well aligned with existing security standards for data centers.

7.2 Cloud Network Virtualization

All clouds utilize some form of virtual networking to abstract the physical network and create a network resource pool. Typically the cloud user provisions desired networking resources from this pool, which can then be configured within the limits of the virtualization technique used. For example, some cloud platforms only support allocation of IP addresses within particular subnets, while others allow the cloud user the capability to provision entire Class B virtual networks and completely define the subnet architecture.

If you are a cloud provider (including managing a private cloud), physical segregation of networks composing your cloud is important for both operational and security reasons. We most commonly see at least three different networks which are isolated onto dedicated hardware since there is no functional or traffic overlap:

- The service network for communications between virtual machines and the Internet. This builds the network resource pool for the cloud users.
- The storage network to connect virtual storage to virtual machines.
- A management network for management and API traffic.



Common networks underlying IaaS.

This isn't the only way to build out a private cloud network architecture, but it is a common baseline, especially for private clouds that don't deal with the massive scale of public cloud providers but still need to balance performance and security.

There are two major categories of network virtualization commonly seen in cloud computing today:

- *Virtual Local Area Networks (VLANs):* VLANs leverage existing network technology implemented in most network hardware. VLANs are extremely common in enterprise networks, even without

cloud computing. They are designed for use in single-tenant networks (enterprise data centers) to separate different business units, functions, etc. (like guest networks). VLANs are not designed for cloud-scale virtualization or security and shouldn't be considered, on their own, an effective security control for isolating networks. They are also never a substitute for physical network segregation.

- *Software Defined Networking (SDN):* A more complete abstraction layer on top of networking hardware, SDNs decouple the network control plane from the data plane (you can **read more on SDN principles at this Wikipedia entry**). This allows us to abstract networking from the traditional limitations of a LAN.

  There are multiple implementations, including standards-based and proprietary options. Depending on the implementation, SDN can offer much higher flexibility and isolation. For example, multiple segregated overlapping IP ranges for a virtual network on top of the same physical network. Implemented properly, and unlike standard VLANs, SDNs provide effective security isolation boundaries. SDNs also typically offer software definition of arbitrary IP ranges, allowing customers to better extend their existing networks into the cloud. If the customer needs the 10.0.0.0/16 CIDR (Classless Inter-Domain Routing) range, an SDN can support it, regardless of the underlying network addressing. It can typically even support multiple customers using the same internal networking IP address blocks.

  On the surface, an SDN may look like a regular network to a cloud user, but being a more complete abstraction will function very differently beneath the surface. The underlying technologies and the management of the SDN will look nothing like what the cloud user accesses, and will have quite a bit more complexity. For example, an SDN may use packet encapsulation so that virtual machines and other "standard" assets don't need any changes to their underlying network stack. The virtualization stack takes packets from standard operating systems (OS) connecting through a virtual network interface, and then encapsulates the packets to move them around the actual network. The virtual machine doesn't need to have any knowledge of the SDN beyond a compatible virtual network interface, which is provided by the hypervisor.

7.3 How Security Changes With Cloud Networking

The lack of direct management of the underlying physical network changes common network practices for the cloud user and provider. The most commonly used network security patterns rely on control of the physical communication paths and insertion of security appliances. This isn't possible for cloud customers, since they only operate at a virtual level.

Traditional Network Intrusion Detection Systems, where communications between hosts are are mirrored and inspected by the virtual or physical Intrusion Detection Systems will not be supported in cloud environments; customer security tools need to rely on an in-line virtual appliance, or a software agent installed in instances. This creates either a chokepoint or increases processor overhead, so be sure you really need that level of monitoring before implementing. Some cloud providers may offer some level of built-in network monitoring (and you have more options with private cloud platforms) but this isn't typically to the same degree as when sniffing a physical network.

7.3.1 Challenges of Virtual Appliances

Since physical appliances can't be inserted (except by the cloud provider) they must be replaced by virtual appliances if still needed, and if the cloud network supports the necessary routing. This brings the same concerns as inserting virtual appliances for network monitoring:

- Virtual appliances thus become bottlenecks, since they cannot fail open, and must intercept all traffic.
- Virtual appliances may take significant resources and increase costs to meet network performance requirements.
- When used, virtual appliances should support auto-scaling to match the elasticity of the resources they protect. Depending on the product, this could cause issues if the vendor does not support elastic licensing compatible with auto-scaling.
- Virtual appliances should also be aware of operating in the cloud, as well as the ability of instances to move between different geographic and availability zones. The *velocity* of change in cloud networks is higher than that of physical networks and tools need to be designed to handle this important difference.
- Cloud application components tend to be more distributed to improve resiliency and, due to auto-scaling, virtual servers may have shorter lives and be more prolific. This changes how security policies need to be designed.
    - This induces that very high rate of change that security tools must be able to manage (e.g., servers with a lifespan of less than an hour).
    - IP addresses will change far more quickly than on a traditional network, which security tools must account for. Ideally they should identify assets on the network by a unique ID, not an IP address or network name.
    - Assets are less likely to exist at static IP addresses. Different assets may share the same IP address within a short period of time. Alerts and the Incident Response lifecycle may have to be modified to ensure that the alert is actionable in such a dynamic environment. Assets within a single application tier will often be located on multiple subnets for resiliency, further complicating IP-based security policies. Due to auto-scaling, assets may also be ephemeral, existing for hours or even minutes. On the upside, cloud architectures skew towards fewer services per server, which improves your ability to define restrictive firewall rules. Instead of a stack of services on a single virtual machine — as on physical servers where you need to maximize the capital investment in the hardware — it is common to run a much smaller set of services, or even a single service, on a virtual machine.

7.3.2 SDN Security Benefits

On the positive side, software-defined networks enable new types of security controls, often making it an overall gain for network security:

- Isolation is easier. It becomes possible to build out as many isolated networks as you need without constraints of physical hardware. For example, if you run multiple networks with the same CIDR address blocks, there is no logical way they can directly communicate, due

to addressing conflicts. This is an excellent way to segregate applications and services of different security contexts. We discuss this microsegregation in more detail below.

- SDN firewalls (e.g., security groups) can apply to assets based on more flexible criteria than hardware-based firewalls, since they aren't limited based on physical topology. (Note that this is true of many types of software firewalls, but is distinct from hardware firewalls). SDN firewalls are typically policy sets that define ingress and egress rules that can apply to single assets or groups of assets, regardless of network location (within a given virtual network). For example, you can create a set of firewall rules that apply to any asset with a particular tag. Keep in mind this gets slightly difficult to discuss, since different platforms use different terminology and have different capabilities to support this kind of capability, so we are trying to keep things at a conceptual level.
    - Combined with the cloud platform's orchestration layer, this enables very dynamic and granular combinations and policies with less management overhead than the equivalent using a traditional hardware or host-based approach. For example, if virtual machines in an auto-scale group are automatically deployed in multiple subnets and load balanced across them, then you can create a firewall ruleset that applies to these instances, regardless of their subnet or IP address. It is a key enabling feature of secure cloud networks that use architectures quite differently from traditional computing.
    - Default deny is often the starting point, and you are required to open connections from there, which is the opposite of most physical networks.
        - Think of it as the granularity of a host firewall with the better manageability of a network appliance. Host firewalls have two issues: They are difficult to manage at scale, and if the system they are on is compromised, they are easy to alter and disable. On the other hand, it is cost-prohibitive to route all internal traffic, even between peers on a subnet, through a network firewall. Software firewalls, such as security groups, are managed outside a system yet apply to each system, without additional hardware costs or complex provisioning needed. Thus, it is trivial to do things like isolate every single virtual machine on the same virtual subnet.
        - As briefly mentioned above, firewall rules can be based on other criteria, such as tags. Note, that while the potential is there, the actual capabilities depend on the platform. Just because a cloud network is SDN-based doesn't mean it actually conveys any security benefits.
        - Many network attacks are eliminated by default (depending on your platforms), such as ARP spoofing and other lower level exploits, beyond merely eliminating sniffing. This is due to the inherent nature of the SDN and application of more software-based rules and analysis in moving packets.
        - It is possible to encrypt packets as they are encapsulated.
        - As with security groups, other routing and network design can be dynamic and tied to the cloud's orchestration layer, such as bridging virtual networks or connecting to internal PaaS services.
        - Additional security functions can potentially be added natively.

7.3.3 Microsegmentation and the Software Defined Perimeter

Microsegmentation (also sometimes referred to as *hypersegregation*) leverages virtual network topologies to run more, smaller, and more isolated networks without incurring additional hardware costs that historically make such models prohibitive. Since the entire networks are defined in software without many of the traditional addressing issues, it is far more feasible to run these multiple, software-defined environments.

A common, practical example leveraging this capability is running most, if not all, applications on their own virtual network and only connecting those networks as needed. This dramatically reduces the *blast radius* if an attacker compromises an individual system. The attacker can no longer leverage this foothold to expand across the entire data center.

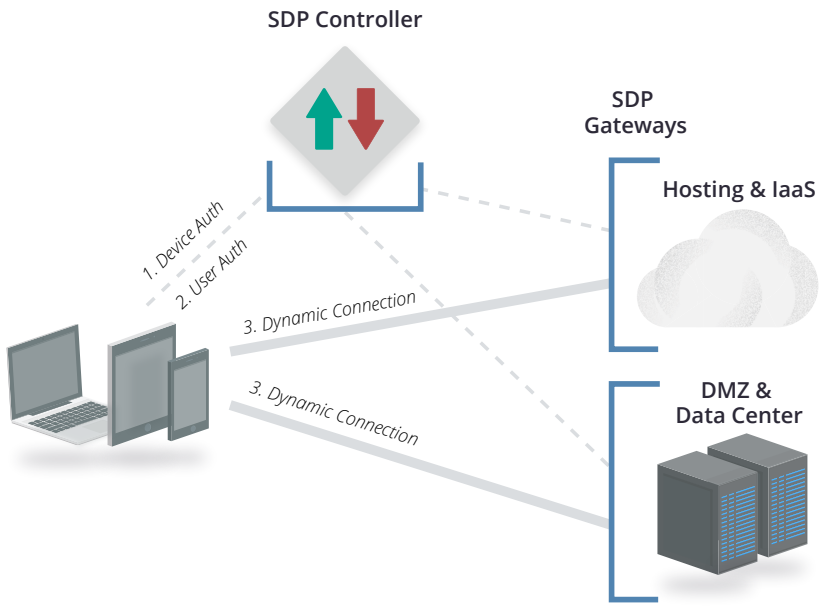
Although there are no increases in capital expenses since cloud microsegmentation is based on software configurations, it *can* increase operational expenses in managing multiple overlapping networks and connectivity.

The CSA **Software Defined Perimeter Working Group** has developed a model and specification that combines device and user authentication to dynamically provision network access to resources and enhance security. SDP includes three components:

- An SDP client on the connecting asset (e.g. a laptop).
- The SDP controller for authenticating and authorizing SDP clients and configuring the connections to SDP gateways.
- The SDP gateway for terminating SDP client network traffic and enforcing policies in communication with the SDP controller.

Network security decisions can thus be made on a wider range of criteria than just IP packets. Especially combined with SDNs this potentially offers greater flexibility and security for evolving network topologies.

More information on SDP is available from the CSA at **https://cloudsecurityalliance.org/group/software-defined-perimeter/#_overview**



Common networks underlying IaaS.

7.3.4 Additional Considerations for Cloud Providers or Private Clouds

Providers must maintain the core security of the physical/traditional networks that the platform is built on. A security failure at the root network will likely compromise the security of all customers. And this security must be managed for arbitrary communications and multiple tenants, some of which must be considered adversarial.

It is absolutely critical to maintain segregation and isolation for the multitenant environment. There will thus be additional overhead to properly enable, configure, and maintain the SDN security controls. While an SDN is more likely to provide needed isolation once it is up and running, it is important to take the extra time to get everything set up properly in order to handle potentially hostile tenants. We aren't saying your users are necessarily hostile, but it is safe to assume that, at some point, something on the network will be compromised and used to further an attack.

Providers must also expose security controls to the cloud users so they can properly configure and manage their network security.

Finally, providers are responsible for implementing perimeter security that protects the environment, but minimizes impact on customer workloads, for example, Distributed Denial of Service Protection (DDoS) and baseline IPS to filter out hostile traffic before it affects the cloud's consumers. Another consideration is to ensure that any potentially sensitive information is scrubbed when a virtual instance is released back to the hypervisor, to ensure the information is not able to be read by another customer when the drive space is provisioned.

7.3.5 Hybrid Cloud Considerations

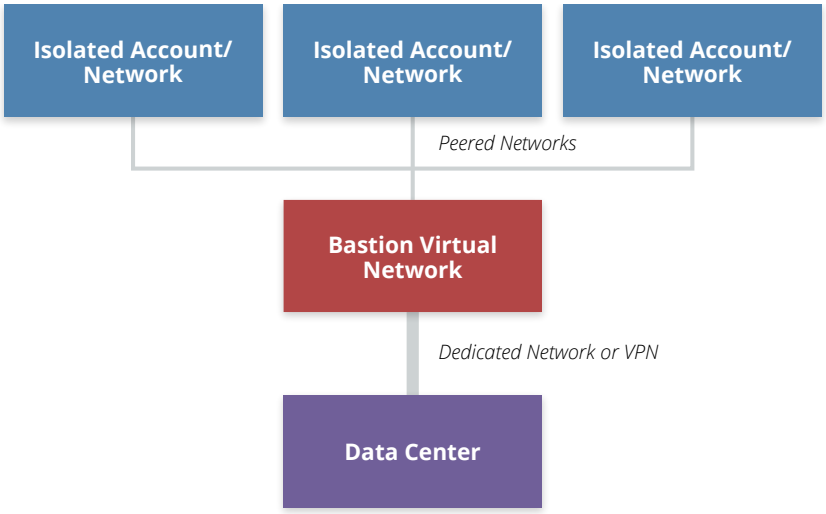
As mentioned in Domain 1, hybrid clouds connect an enterprise private cloud or data center to a public cloud provider, typically using either a dedicated Wide Area Network (WAN) link or VPN. Ideally the hybrid cloud will support arbitrary network addressing to help seamlessly extend the cloud user's network. If the cloud uses the same network address range as your on-premises assets, it is effectively unusable.

The hybrid connection may reduce the security of the cloud network if the private network isn't at an equivalent security level. If you run a flat network in your data center, with minimal segregation from your employees' systems, someone could compromise an employee's laptop and then use that to scan your entire cloud deployment over the hybrid connection. A hybrid connection shouldn't effectively flatten the security of both networks. Separation should be enforced via routing, access controls, and even firewalls or additional network security tools between the two networks.

For management and security reasons it is typically preferable to minimize hybrid connections. Connecting multiple disparate networks is complex, especially when one of those networks is software-defined and the other limited by hardware. Hybrid connections are often still necessary, but don't assume they are needed. They may increase routing complexity, can reduce the ability to run multiple cloud networks with overlapping IP ranges, and complicate security on both sides, due to the need to harmonize security controls.

One emerging architecture for hybrid cloud connectivity is "bastion" or "transit" virtual networks:

- This scenario allows you to connect multiple, different cloud networks to a data center using a single hybrid connection. The cloud user builds a dedicated virtual network for the hybrid connection and then peers any other networks through the designated bastion network.
- Second-level networks connect to the data center through the bastion network, but since they aren't peered to each other they can't talk to each other and are effectively segregated. Also, you can deploy different security tools, firewall rulesets, and Access Control Lists in the bastion network to further protect traffic in and out of the hybrid connection.



"Bastion" or "Transit" networks for more-flexible hybrid cloud architectures.

7.4 Cloud Compute and Workload Security

A workload is a unit of processing, which can be in a virtual machine, a container, or other abstraction. Workloads always run somewhere on a processor and consume memory. Workloads include a very diverse range of processing tasks, which range from traditional applications running in a virtual machine on a standard operating system, to GPU- or FPGA-based specialized tasks. Nearly every one of these options is supported in some form in cloud computing.

It's important to remember that every cloud workload runs on a hardware stack, and the integrity of this hardware is absolutely critical for the cloud provider to maintain. Different hardware stacks also support different execution isolation and chain of trust options. This can include hardware-based supervision and monitoring processes that run outside the main processors, secure execution environments, encryption and key management enclaves, and more. The range and rapidly changing nature of these options exceeds our ability to provide proscriptive guidance at this time, but in a general sense there are potentially very large gains in security by selecting and properly leveraging hardware with these advanced capabilities.

There are multiple compute abstraction types, each with differing degrees of segregation and isolation:

- *Virtual machines:* Virtual machines are the most-well known form of compute abstraction, and are offered by all IaaS providers. They are commonly called instances in cloud computing since they are created (or cloned) off a base image. The Virtual Machine Manager (hypervisor) abstracts an operating system from the underlying hardware. Modern hypervisors can tie into underlying hardware capabilities now commonly available on standard servers (and workstations) to reinforce isolation while supporting high-performance operations.

  Virtual machines are potentially open to certain memory attacks, but this is increasingly difficult due to ongoing hardware and software enhancements to reinforce isolation. VMs on modern hypervisors are generally an effective security control, and advances in hardware isolation for VMs and secure execution environments continue to improve these capabilities.

- *Containers:* Containers are code execution environments that run within an operating system (for now), sharing and leveraging resources of that operating system. While a VM is a full abstraction of an operating system, a container is a constrained place to run segregated processes while still utilizing the kernel and other capabilities of the base OS. Multiple containers can run on the same virtual machine or be implemented without the use of VMs at all and run directly on hardware. The container provides code running inside a restricted environment with only access to the processes and capabilities defined in the container configuration. This allows containers to launch incredibly rapidly, since they don't need to boot an operating system or launch many (sometimes any) new services; the container only needs access to already-running services in the host OS and some can launch in milliseconds.

  Containers are newer, with differing isolation capabilities that are very platform-dependent. They are also evolving quickly with different management systems, underlying operating systems, and container technologies. We cover containers in more depth in Domain 8.

- *Platform-based workloads:* This is a more complex category that covers workloads running on a shared platform that aren't virtual machines or containers, such as logic/procedures running on a shared database platform. Imagine a stored procedure running inside a multitenant database, or a machine-learning job running on a machine-learning Platform as a Service. Isolation and security are totally the responsibility of the platform provider, although the provider may expose certain security options and controls.

- *Serverless computing:* Serverless is a broad category that refers to any situation where the cloud user doesn't manage any of the underlying hardware or virtual machines, and just accesses exposed functions. For example, there are serverless platforms for directly executing application code. Under the hood, these still utilize capabilities such as containers, virtual machines, or specialized hardware platforms. From a security perspective, serverless is merely a combined term that covers containers and platform-based workloads, where the cloud provider manages all the underlying layers, including foundational security functions and controls.

7.4.1 How Cloud Changes Workload Security

Any given processor and memory will nearly always be running multiple workloads, often from different tenants. Multiple tenants will likely share the same physical compute node, and there is a range of segregation capabilities on different hardware stacks. The burden to maintain workload isolation is on the cloud provider and should be one of their top priorities.

In some environments dedicated/private tenancy is possible, but typically at a higher cost. With this model only designated workloads run on a designated physical server. Costs increase in public cloud as a consumer since you are taking hardware out of the general resource pool, but also in private cloud, due to less efficient use of internal resources.

Cloud users rarely get to control where a workload physically runs, regardless of deployment model, although some platforms do support designating particular hardware pools or general locations to support availability, compliance, and other requirements.

7.4.2 Immutable Workloads Enable Security

Auto-scaling and containers, by nature, work best when you run instances launched dynamically based on an image; those instances can be shut down when no longer needed for capacity without breaking an application stack. This is core to the elasticity of compute in the cloud. Thus, you no longer patch or make other changes to a running workload, since that wouldn't change the image, and, thus, new instances would be out of sync with whatever manual changes you make on whatever is running. We call these virtual machines *immutable*.

To reconfigure or change an immutable instance you update the underlying image, and then rotate the new instances by shutting down the old ones and running the new ones in their place.

There are degrees of immutable. The pure definition is fully replacing running instances with a new image. However, some organizations only push new images to update the operating system and use alternative deployment techniques to push code updates into running virtual machines. While technically not completely immutable, since the instance changes, these pushes still happen completely through automation and no one ever manually logs in to running systems to make local changes.

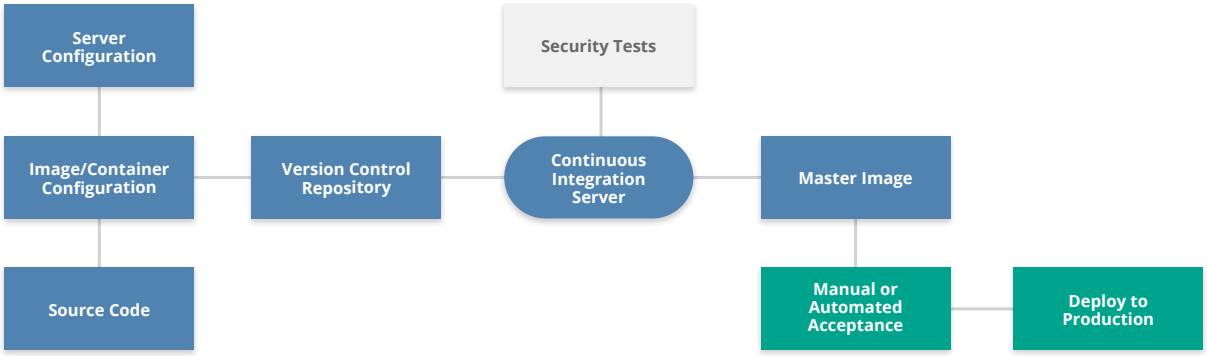
Immutable workloads enable significant security benefits:

- You no longer patch running systems or worry about dependencies, broken patch processes, etc. You replace them with a new gold master.
- You can, and should, disable remote logins to running workloads (if logins are even an option). This is an operational requirement to prevent changes that aren't consistent across the stack, which also has significant security benefits.
- It is much faster to roll out updated versions, since applications must be designed to handle individual nodes going down (remember, this is fundamental to any auto-scaling). You are less constrained by the complexity and fragility of patching a running system. Even if something breaks, you just replace it.

- It is easier to disable services and whitelist applications/processes since the instance should never change.
- Most security testing can be managed during image creation, reducing the need for vulnerability assessment on running workloads since their behavior should be completely known at the time of creation. This doesn't eliminate all security testing for production workloads, but it is a means of offloading large portions of testing.

Immutable does add some requirements:

- You need a consistent image creation process and the automation to support updating deployments. These new images must be produced on a regular basis to account for patch and malware signature updates.
- Security testing must be integrated into the image creation and deployment process, including source code tests and, if using virtual machines or standard containers, vulnerability assessments.
- Image configurations need mechanisms to disable logins and restrict services before deploying the images and using them for production virtual machines.
- You may want a process, for some workloads, to enable logins to workloads that aren't actively in the application stack for troubleshooting. This could be a workload pulled from the group but allowed to continue to run in isolation. Alternatively (and often preferred), send sufficiently detailed logs to an external collector so that there is never a need to log in.
- There will be increased complexity to manage the service catalog, since you might create dozens, or even hundreds, of images on any given day.



A deployment pipeline for creating images for immutable virtual machines or containers.

7.4.3 The Impact of Cloud on Standard Workload Security Controls

Some standard workload controls aren't as viable in cloud workloads (e.g. running antivirus inside some container types). Others aren't necessarily needed or need deep modification to maintain effectiveness in cloud computing:

- You may lose the ability to run software agents for non-VM based workloads, such as those running in "serverless" provider-managed containers.
- "Traditional" agents may impede performance more heavily in cloud. Lightweight agents with lower compute requirements allow better workload distribution and efficient use of resources. Agents not designed for cloud computing may assume underlying compute capacity that isn't aligned with how the cloud deployment is designed. The developers on a given project might assume they are running a fleet of lightweight, single-purpose virtual machines. A security agent not attuned to this environment could significantly increase processing overhead, requiring larger virtual machine types and increasing costs.
- Agents that operate in cloud environments also need to support dynamic cloud workloads and deployment patterns like auto-scaling. They can't rely (on the agent or in the management system) on static IP addressing. While some cloud assets run on static IP addresses, it is far more common for the cloud to dynamically assign IP addresses at run time to enable elasticity. Thus, the agent must have the ability to discover the management/control plane and use that to determine what kind of workload it is running on and where.
- The management plane of the agent must itself also operate at the speed of auto-scaling and support elasticity (e.g., be able to keep up with incredibly dynamic IP addressing, such as the same address used by multiple workloads within a single hour). Traditional tools aren't normally designed for this degree of velocity, creating the same issue as we discussed with network security and firewalls.
- Agents shouldn't increase attack surface due to communications/networking or other requirements that increase the attack surface. While this is always true, there is a greater likelihood of an agent becoming a security risk in cloud for a few reasons:
    - We have a greater ability to run immutable systems, and an agent, like any piece of software, opens up additional attack surface, especially if it ingests configuration changes and signatures that could be used as an attack vector.
    - In cloud we also tend to run fewer different services with a smaller set of networking ports on any given virtual machine (or container), as compared to a physical server. Some agents require opening up additional firewall ports, which increases the network attack surface.
    - This doesn't mean agents always create new security risks, but the benefits need to be balanced before simply assuming the security upside.
- File integrity monitoring can be an effective means of detecting unapproved changes to running immutable instances. Immutable workloads typically require fewer additional security tools, due to their hardened nature. They are locked down more than the usual servers and tend to run a smaller set of services. File integrity monitoring, which tends to be very lightweight, can be a good security control for immutable workloads since you should essentially have zero false positives by their unchanging nature.
- Long-running VMs that still run standard security controls may be isolated on the network, changing how they are managed. You might experience difficulty in connecting your

management tool to a virtual machine running in a private network subnet. While you can technically run the management tool in the same subnet, this could increase costs significantly and be more difficult to manage.
- Cloud workloads running in isolation are typically less resilient than on physical infrastructure, due to the abstraction. Providing disaster recovery for these is extremely important.

7.4.4 Changes to Workload Security Monitoring and Logging

Security logging/monitoring is more complex in cloud computing:

- IP addresses in logs won't necessarily reflect a particular workflow since multiple virtual machines may share the same IP address over a period of time, and some workloads like containers and serverless may not have a recognizable IP address at all. Thus, you need to collect some other unique identifiers in the logs to be assured you know what the log entries actually refer to. These unique identifiers need to account for ephemeral systems, which may only be active for a short period of time.
- Logs need to be offloaded and collected externally more quickly due to the higher velocity of change in cloud. You can easily lose logs in an auto-scale group if they aren't collected before the cloud controller shuts down an unneeded instance.
    - Logging architectures need to account for cloud storage and networking costs. For example, sending all logs from instances in a public cloud to on-premises Security Information and Event Management (SIEM) may be cost prohibitive, due to the additional internal storage and extra Internet networking fees.

7.4.5 Changes to Vulnerability Assessment

Vulnerability assessments in cloud computing need to account for both architectural and contractual limitations:

- The cloud owner (public or private) will typically require notification of assessments and place limits on the nature of assessments. This is because they may be unable to distinguish an assessment from a real attack without prior warning.
- Default deny networks further limit the potential effectiveness of an automated network assessment, just as any firewall would. You either need to open up holes to perform the assessment, use an agent on the instance to perform the assessment, or assess knowing that a lot of tests are blocked by the firewall rules.
- Assessments can be run during the image creation process for immutable workloads. Since these aren't in production, and the process is automated, they can run with fewer network restrictions, thus increasing the assessment surface.
- Penetration testing is less affected since it still uses the same scope as an attacker. We cover penetration testing in more detail in Domain 10.

7.4.6 Cloud Storage Security

Although part of infrastructure, we cover storage and data security in much more depth in Domain 11.

7.5 Recommendations

- Know the infrastructure security of your provider or platform.
  - In the shared security model, the provider (or whoever maintains the private cloud platform) has the burden of ensuring the underlying physical, abstraction, and orchestration layers of the cloud are secure.
  - Review compliance certifications and attestations.
    - Check industry-standard and industry-specific compliance certifications and attestations on a regular basis for having the assurance that your provider is following cloud infrastructure best-practices and regulations.
- Network
  - Prefer SDN when available.
  - Use SDN capabilities for multiple virtual networks and multiple cloud accounts/segments to increase network isolation.
    - Separate accounts and virtual networks dramatically limit blast radius compared to traditional data centers.
  - Implement default deny with cloud firewalls.
  - Apply cloud firewalls on a per-workload basis as opposed to a per-network basis.
  - Always restrict traffic between workloads in the same virtual subnet using a cloud firewall (security group) policy whenever possible.
  - Minimize dependency on virtual appliances that restrict elasticity or cause performance bottlenecks.
- Compute/workload
  - Leverage immutable workloads whenever possible.
    - Disable remote access.
    - Integrate security testing into image creation.
    - Alarm with file integrity monitoring.
    - Patch by updating images, not patching running instances.
    - Choose security agents that are cloud-aware and minimize performance impact, if needed.
  - Maintain security controls for long-running workloads, but use tools that are cloud aware.
  - Store logs external to workloads.
  - Understand and comply with cloud provider limitations on vulnerability assessments and penetration testing.

DOMAIN 8
Virtualization and Containers

8.0 Introduction

Virtualization isn't merely a tool for creating virtual machines—it's the core technology for enabling cloud computing. We use virtualization all throughout computing, from full operating virtual machines to virtual execution environments like the Java Virtual Machine, as well as in storage, networking, and beyond.

Cloud computing is fundamentally based on pooling resources and virtualization is the technology used to convert fixed infrastructure into these pooled resources. Virtualization provides the abstraction needed for resource pools, which are then managed using orchestration.

As mentioned, virtualization covers an extremely wide range of technologies; essentially any time we create an abstraction, we're using virtualization. For cloud computing we tend to focus on those specific aspects of virtualization used to create our resource pools, especially:

- Compute
- Network
- Storage
- Containers

The aforementioned aren't the only categories of virtualization, but they are the ones most relevant to cloud computing.

Understanding the impacts of virtualization on security is fundamental to properly architecting and implementing cloud security. Virtual assets provisioned from a resource pool may look just like the physical assets they replace, but that look and feel is really just a tool to help us better understand and manage what we see. It's also a useful way to leverage existing technologies, like operating systems, without having to completely rewrite them from scratch. Underneath, these virtual assets work completely differently from the resources they are abstracted from.

8.1 Overview

At its most basic, virtualization abstracts resources from their underlying physical assets. You can virtualize nearly anything in technology, from entire computers to networks to code. As mentioned in the introduction, cloud computing is fundamentally based on virtualization: It's how we abstract resources to create pools. Without virtualization, there is no cloud.

Many security processes are designed with the expectation of physical control over the underlying infrastructure. While this doesn't go away with cloud computing, virtualization adds two new layers for security controls:

- *Security of the virtualization technology itself*, e.g., securing a hypervisor.
- *Security controls for the virtual assets*. In many cases, this must be implemented differently than it would be in the corresponding physical equivalent. For example, as discussed in Domain 7, virtual firewalls are not the same as physical firewalls, and mere abstraction of a physical firewall into a virtual machine still may not meet deployment or security requirements.

Virtualization security in cloud computing still follows the shared responsibility model. The cloud provider will always be responsible for securing the physical infrastructure and the virtualization platform itself. Meanwhile, the cloud customer is responsible for properly implementing the available virtualized security controls and understanding the underlying risks, based on what is implemented and managed by the cloud provider. For example, deciding when to encrypt virtualized storage, properly configuring the virtual network and firewalls, or deciding when to use dedicated hosting vs. a shared host.

Since many of these controls touch upon other areas of cloud security, such as data security, we try to focus on the virtualization-specific concerns in this domain. The lines aren't always clear, however, and the bulk of cloud security controls are covered more deeply in the other domains of this Guidance. Domain 7: Infrastructure Security focuses extensively on virtual networks and workloads.

8.1.1 Major Virtualization Categories Relevant to Cloud Computing

8.1.1.1 Compute

Compute virtualization abstracts the running of code (including operating systems) from the underlying hardware. Instead of running directly on the hardware, the code runs on top of an abstraction layer that enables more flexible usage, such as running multiple operating systems on the same hardware (virtual machines). This is a simplification, and we recommend further research into virtual machine managers and hypervisors if you are interested in learning more.

Compute most commonly refers to virtual machines, but this is quickly changing, in large part due to ongoing technology evolution and adoption of containers.

Containers and certain kinds of serverless infrastructure also abstract compute. These are different abstractions to create code execution environments, but they don't abstract a full operating system

as a virtual machine does. (Containers are covered in more detail below.)

Cloud Provider Responsibilities

The primary security responsibilities of the cloud provider in compute virtualization are to enforce *isolation* and maintain a *secure virtualization infrastructure*.

- *Isolation* ensures that compute processes or memory in one virtual machine/container should not be visible to another. It is how we separate different tenants, even when they are running processes on the same physical hardware.
- The cloud provider is also responsible for securing the *underlying infrastructure and the virtualization technology* from external attack or internal misuse. This means using patched and up-to-date hypervisors that are properly configured and supported with processes to keep them up to date and secure over time. The inability to patch hypervisors across a cloud deployment could create a fundamentally insecure cloud when a new vulnerability in the technology is discovered.

Cloud providers should also support secure use of virtualization for cloud users. This means creating a secure chain of processes from the image (or other source) used to run the virtual machine all the way through a boot process with security and integrity. This ensures that tenants cannot launch machines based on images that they shouldn't have access to, such as those belonging to another tenant, and that a running virtual machine (or other process) is the one the customer expects to be running.

In addition, cloud providers should assure customers that volatile memory is safe from unapproved monitoring, since important data could be exposed if another tenant, a malicious employee, or even an attacker is able to access running memory.

Cloud User Responsibilities

Meanwhile, the primary responsibility of the cloud user is to properly implement the security of whatever it deploys within the virtualized environment. Since the onus of compute virtualization security is on the provider, the customer tends to have only a few security options relating directly to the virtualization of the workload. There is quite a bit more to securing workloads, and those are covered in Domain 7.

That said, there are still some virtualization-specific differences that the cloud user can address in their security implementation. Firstly, the cloud user should take advantage of the security controls for managing their virtual infrastructure, which will vary based on the cloud platform and often include:

- *Security settings, such as identity management, to the virtual resources*. This is not the identity management within the resource, such as the operating system login credentials, but the identity management of who is allowed to access the cloud management of the resource—for example, stopping or changing the configuration of a virtual machine. See Domain 6 for specifics on management plane security.

- *Monitoring and logging*. Domain 7 covers monitoring and logging of workloads, including how to handle system logs from virtual machines of containers, but the cloud platform will likely offer additional logging and monitoring at the virtualization level. This can include the status of a virtual machine, management events, performance, etc.
- *Image asset management*. Cloud compute deployments are based on master images—be it a virtual machine, container, or other code—that are then run in the cloud. This is often highly automated and results in a larger number of images to base assets on, compared to traditional computing master images. Managing these—including which meet security requirements, where they can be deployed, and who has access to them—is an important security responsibility.
- *Use of dedicated hosting*, if available, based on the security context of the resource. In some situations you can specify that your assets run on hardware dedicated only to you (at higher cost), even on a multitenant cloud. This may help meet compliance requirements or satisfy security needs in special cases where sharing hardware with another tenant is considered a risk.

Secondly, the cloud user is also responsible for security controls within the virtualized resource:

- This includes all the standard security for the workload, be it a virtual machine, container, or application code. These are well covered by standard security, best practices and the additional guidance in Domain 7.
- Of particular concern is ensuring deployment of only secure configurations (e.g., a patched, updated virtual machine image). Due to the automation of cloud computing it is easy to deploy older configurations that may not be patched or properly secured.

Other general compute security concerns include:

- Virtualized resources tend to be more ephemeral and change at a more rapid pace. Any corresponding security, such as monitoring, must keep up with the pace. Again, the specifics are covered in more depth in Domain 7.
- Host-level monitoring/logging may not be available, especially for serverless deployments. Alternative log methods may need to be implemented. For example, in a serverless deployment, you are unlikely to see system logs of the underlying platform and should offset by writing more robust application logging in to your code.

8.1.2 Network

There are multiple kinds of virtual networks, from basic VLANs to full Software-Defined Networks. As a core of cloud infrastructure security these are covered both here and in Domain 7.

To review, most cloud computing today uses SDN for virtualizing networks. (VLANs are often not suitable for cloud deployments since they lack important isolation capabilities for multitenancy.)

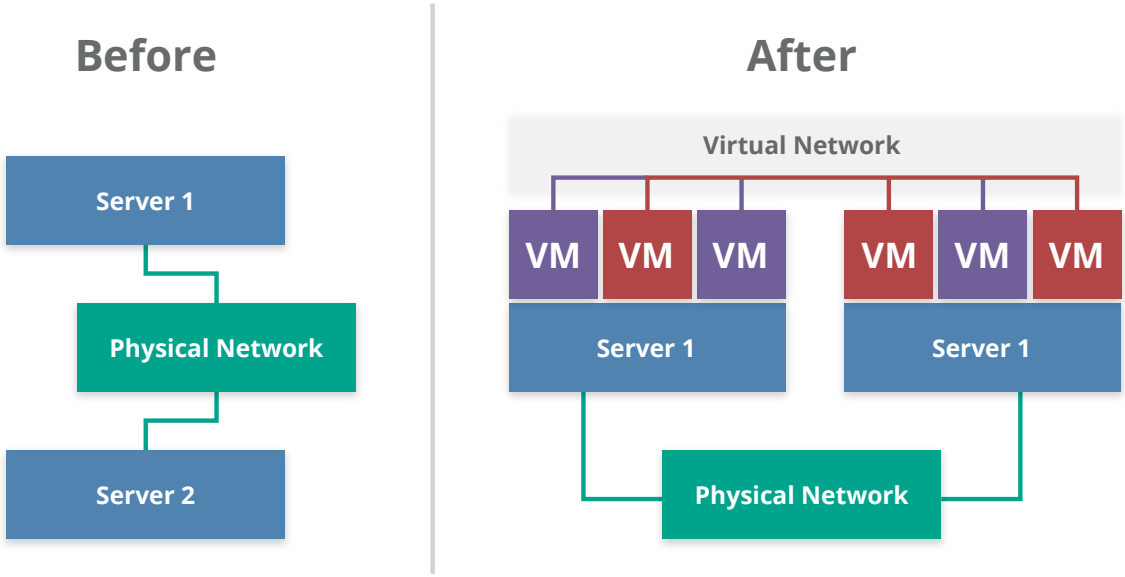
SDN abstracts the network management plane from the underlying physical infrastructure, removing many typical networking constraints. For example, you can overlay multiple virtual

networks, even ones that completely overlap their address ranges, over the same physical hardware, with all traffic properly segregated and isolated. SDNs are also defined using software settings and API calls, which supports orchestration and agility.

Virtual networks are quite different than physical networks. They run on physical networks, but abstraction allows for deep modification on networking behavior in ways that impact many security processes and technologies.

8.1.2.1 Monitoring and Filtering

In particular, monitoring and filtering (including firewalls) change extensively due to the differences in how packets move around the virtual network. Resources may communicate on a physical server without traffic crossing the physical network. For example, if two virtual machines are located on the same physical machine there is no reason to route network traffic off the box and onto the network. Thus, they can communicate directly, and monitoring and filtering tools inline on the network (or attached to the routing/switching hardware) will never see the traffic.



Virtual networks move packets in software and monitoring can't rely on sniffing the physical network connections.

To compensate, you can route traffic to a virtual network monitoring or filtering tool on the same hardware (including a virtual machine version of a network security product). You can also bridge all network traffic back out to the network, or route it to a virtual appliance on the same virtual network. Each of these approaches has drawbacks since they create bottlenecks and less-efficient routing.

The cloud platform/provider may not support access for direct network monitoring. Public cloud providers rarely allow full packet network monitoring to customers, due to the complexity (and cost). Thus, you can't assume you will ever have access to raw packet data unless you collect it yourself in the host, or using a virtual appliance.

With public cloud in particular, some communications between cloud services will occur on the

provider's network; customer monitoring and filtering of that traffic isn't possible (and would create a security risk for the provider). For example, if you connect a serverless application to the cloud provider's object storage, database platform, message queue, or other PaaS product, this traffic would run natively on the provider's network, not necessarily within the customer-managed virtual network. As we move out of simple infrastructure virtualization, the concept of a customer-managed network begins to fade.

However, all modern cloud platforms offer built-in firewalls, which may offer advantages over corresponding physical firewalls. These are software firewalls that may operate within the SDN or the hypervisor. They typically offer fewer features than a modern, dedicated next-generation firewall, but these capabilities may not always be needed due to other inherent security provided by the cloud provider.

8.1.2.2 Management Infrastructure

Virtual networks for cloud computing always support remote management and, as such, securing the management plane/metastructure is critical. At times it is possible to create and destroy entire complex networks with a handful of API calls or a few clicks on a web console.

Cloud Provider Responsibilities

The cloud provider is primarily responsible for building a secure network infrastructure and configuring it properly. The absolute top security priority is segregation and isolation of network traffic to prevent tenants from viewing another's traffic. This is the most foundational security control for any multitenant network.

The provider should disable packet sniffing or other metadata "leaks" that could expose data or configurations between tenants. Packet sniffing, even within a tenant's own virtual networks, should also be disabled to reduce the ability of an attacker to compromise a single node and use it to monitor the network, as is common on non-virtualized networks. Tagging or other SDN-level metadata should also not be exposed outside the management plane or a compromised host could be used to span into the SDN itself.

All virtual networks should enable built-in firewall capabilities for cloud users without the need for host firewalls or external products. The provider is also responsible for detecting and preventing attacks on the underlying physical network and virtualization platform. This includes perimeter security of the cloud itself.

Cloud User Responsibilities

Cloud users are primarily responsible for properly configuring their deployment of the virtual network, especially any virtual firewalls.

Network architecture can play a larger role in virtual network security since we aren't constrained by physical connections and routing. Since virtual networks are software constructs, the use of multiple, separate virtual networks may offer extensive compartmentalization advantages not possible

on a traditional physical network. You can run every application stack in its own virtual network, which dramatically reduces the attack surface if a malicious actor gains a foothold. An equivalent architecture on a physical network is cost prohibitive.

Immutable networks can be defined on some cloud platforms using software templates, which can help enforce known-good configurations. The entire known-good state of the network can be defined in a template, instead of having to manually configure all the settings. Aside from the ability to create multiple networks with a secure baseline, these can also be used to detect, and in some cases revert, deviations from known-good states.

The cloud user is, again, responsible for proper rights management and configuration of exposed controls in the management plane. When virtual firewalls and/or monitoring don't meet security needs, the consumer may need to compensate with a virtual security appliance or host security agent. This falls under cloud infrastructure security and is covered in depth in Domain 7.

8.1.2.3 Cloud Overlay Networks

Cloud overlay networks are a special kind of WAN virtualization technology for created networks that span multiple "base" networks. For example, an overlay network could span physical and cloud locations or multiple cloud networks, perhaps even on different providers. A full discussion is beyond the scope of this Guidance and the same core security recommendations apply.

8.1.3 Storage

Storage virtualization is already common in most organizations—Storage Area Network (SAN) and Network-Attached Storage (NAS) are both common forms of storage virtualization—and storage security is discussed in more detail in Domain 11.

Most virtualized storage is durable and keeps multiple copies of data in different locations so that drive failures are less likely to result in data loss. Encrypting those drives reduces the concern that swapping out a drive, which is a very frequent activity, could result in data exposure.

However, this encryption doesn't protect data in any virtualized layers; it only protects the data at physical storage. Depending on the type of storage the cloud provider may also (or instead) encrypt it at the virtualization layer, but this may not protect customer data from exposure to the cloud provider. Thus, any additional protection should be provided using the advice in Domain 11.

8.1.4 Containers

Containers are highly portable code execution environments. To simplify, a virtual machine is a complete operating system, all the way down to the kernel. A container, meanwhile, is a virtual execution environment that features an isolated user space, but uses a shared kernel. A full discussion is beyond the scope of this guidance and **you can read more about software containers at this Wikipedia entry**.

Such containers can be built directly on top of physical servers or run on virtual machines. Current implementations rely on an existing kernel/operating system, which is why they can run inside a virtual machine even if nested virtualization is not supported by the hypervisor. (Software containers rely on a completely different technology for hypervisors.)

Software container systems always include three key components:

- The execution environment (the container).
- An orchestration and scheduling controller (which can be a collection of multiple tools).
- A repository for the container images or code to execute.
- Together, these are the place to run things, the things to run, and the management system to tie them together.

Regardless of the technology platform, container security includes:

- *Assuring the security of the underlying physical infrastructure (compute, network, storage)*. This is no different than any other form of virtualization, but it now extends into the underlying operating system where the container's execution environment runs.
- *Assuring the security of the management plane*, which in this case are the orchestrator and the scheduler.
- *Properly securing the image repository*. The image repository should be in a secure location with appropriate access controls configured. This is both to prevent loss or unapproved modification of container images and definition files, as well as to forestall leaks of sensitive data through unapproved access to the files. Containers run so easily that it's also important that images are only able to deploy in the right security context.
- *Building security into the tasks/code running inside the container*. It's still possible to run vulnerable software inside a container and, in some cases this could expose the shared operating system or data from other containers. For example, it is possible to configure some containers to allow not merely access to the container's data on the file system but also root file system access. Allowing too much network access is also a possibility. These are all specific to the particular container platform and thus require securely configuring both the container environment *and* the images/container configurations themselves.

Containers are rapidly evolving, which complicates some aspects of security, but doesn't mean that they are inherently insecure.

Containers don't necessarily provide full security isolation, but they do provide task segregation. That said, virtual machines typically do provide security isolation. Thus you can put tasks of equivalent security context on the same set of physical or virtual hosts in order to provide greater security segregation.

Container management systems and image repositories also have different security capabilities, based on which products you use. Security should learn and understand the capabilities of the products they need to support. Products should, at a minimum, support role-based access controls and strong authentication. They should also support secure configurations, such as isolating file system, process, and network access.

A deep understanding of container security relies on a deep understanding of operating system internals, such as namespaces, network port mapping, memory, and storage access.

Different host operating systems and container technologies offer different security capabilities. This assessment should be included in any container platform selection process.

One key area to secure is which images/tasks/code are allowed into a particular execution environment. A secure repository with proper container management and scheduling will enable this.

8.2 Recommendations

- Cloud providers should:
  - Inherently secure any underlying physical infrastructure used for virtualization.
  - Focus on assuring security isolation between tenants.
  - Provide sufficient security capabilities at the virtualization layers to allow cloud users to properly secure their assets.
  - Strongly defend the physical infrastructure and virtualization platforms from attack or internal compromise.
  - Implement all customer-managed virtualization features with a secure-by-default configuration.
  - Specific priorities:
    - Compute
      - Use secure hypervisors and implement a patch management process to keep them up to date.
      - Configure hypervisors to isolate virtual machines from each other.
      - Implement internal processes and technical security controls to prevent admin/non-tenant access to running VMs or volatile memory.
    - Network
      - Implement essential perimeter security defenses to protect the underlying networks from attack and, wherever possible, to detect and prevent attacks against consumers at the physical level, as well as at any virtual network layers that they can't directly protect themselves.
      - Assure isolation between virtual networks, even if those networks are all controlled by the same consumer.
        - Unless the consumer deliberately connects the separate virtual networks.
      - Implement internal security controls and policies to prevent both modification of consumer networks and monitoring of traffic without approval or outside contractual agreements.
    - Storage
      - Encrypt any underlying physical storage, if it is not already encrypted at another level, to prevent data exposure during drive replacements.
      - Isolate encryption from data-management functions to prevent unapproved access to customer data.

- Cloud users should:
  - Ensure they understand the capabilities offered by their cloud providers as well as any security gaps.
  - Properly configure virtualization services in accordance with the guidance from the cloud provider and other industry best practices.
    - The bulk of fundamental virtualization security falls on the cloud provider, which is why most of the security recommendations for cloud users are covered in the other domains of this Guidance.
  - For containers:
    - Understand the security isolation capabilities of both the chosen container platform and underlying operating system then choose the appropriate configuration.
    - Use physical or virtual machines to provide container isolation and group containers of the same security contexts on the same physical and/or virtual hosts.
    - Ensure that only approved, known, and secure container images or code can be deployed.
    - Appropriately secure the container orchestration/management and scheduler software stack(s).
    - Implement appropriate role-based access controls and strong authentication for all container and repository management.

DOMAIN 9
Incident Response

9.0 Introduction

Incident Response (IR) is a critical facet of any information security program. Preventive security controls have proven unable to completely eliminate the possibility that critical data could be compromised. Most organizations have some sort of IR plan to govern how they will investigate an attack, but as the cloud presents distinct differences in both access to forensic data and governance, organizations must consider how their IR processes will change.

This domain seeks to identify those gaps pertinent to IR that are created by the unique characteristics of cloud computing. Security professionals may use this as a reference when developing response plans and conducting other activities during the preparation phase of the IR lifecycle. This domain is organized in accord with the commonly accepted Incident Response Lifecycle as described in the National Institute of Standards and Technology Computer Security Incident Handling Guide (NIST 800-61rev2 08/2012) [1]. Other international standard frameworks for incident response include ISO/IEC 27035 and the ENISA Strategies for incident response and cyber crisis cooperation.

After describing the Incident Response Lifecycle, as laid out in NIST 800-61rev2, each subsequent section addresses a phase of the lifecycle and explores the potential considerations for responders as they work in a cloud environment.

9.1 Overview

9.1.1 Incident Response Lifecycle

The Incident Response Lifecycle is defined in the NIST 800-61rev2 document. It includes the following phases and major activities:

The Incident Response Lifecycle

- Preparation: "Establishing an incident response capability so that the organization is ready to respond to incidents."
  - Process to handle the incidents.
  - Handler communications and facilities.
  - Incident analysis hardware and software.
  - Internal documentation (port lists, asset lists, network diagrams, current baselines of network traffic).
  - Identifying training.
  - Evaluating infrastructure by proactive scanning and network monitoring, vulnerability assessments, and performing risk assessments.
  - Subscribing to third-party threat intelligence services.

- Detection and Analysis
  - Alerts [endpoint protection, network security monitoring, host monitoring, account creation, privilege escalation, other indicators of compromise, SIEM, security analytics (baseline and anomaly detection), and user behavior analytics].
  - Validate alerts (reducing false positives) and escalation.
  - Estimate the scope of the incident.
  - Assign an Incident Manager who will coordinate further actions.
  - Designate a person who will communicate the incident containment and recovery status to senior management.
  - Build a timeline of the attack.
  - Determine the extent of the potential data loss.
  - Notification and coordination activities.

- Containment, Eradication and Recovery
  - *Containment:* Taking systems offline. Considerations for data loss versus service availability. Ensuring systems don't destroy themselves upon detection.
  - *Eradication and Recovery:* Clean up compromised devices and restore systems to normal operation. Confirm systems are functioning properly. Deploy controls to prevent similar incidents.
  - Documenting the incident and gathering evidence (chain of custody).

- Post-mortem
    - What could have been done better? Could the attack have been detected sooner? What additional data would have been helpful to isolate the attack faster? Does the IR process need to change? If so, how?

9.1.2 How the Cloud Impacts IR

Each of the phases of the lifecycle is affected to different degrees by a cloud deployment. Some of these are similar to any incident response in an outsourced environment where you need to coordinate with a third party. Other differences are more specific to the abstracted and automated nature of cloud.

9.1.2.1 Preparation

When preparing for cloud incident response, here are some major considerations:

- *SLAs and Governance*: Any incident using a public cloud or hosted provider requires an understanding of service level agreements (SLAs), and likely coordination with the cloud provider. Keep in mind that, depending on your relationship with the provider, you may not have direct points of contact and might be limited to whatever is offered through standard support. A custom private cloud in a third-party data center will have a very different relationship than signing up through a website and clicking through a license agreement for a new SaaS application.

Key questions include: What does your organization do? What is the cloud service provider (CSP) responsible for? Who are the points of contact? What are the response time expectations? What are the escalation procedures? Do you have out-of-band communication procedures (in case networks are impacted)? How do the hand-offs work? What data are you going to have access to?

Be sure to test the process with the CSP if possible. Validate that escalations and roles/responsibilities are clear. Ensure the CSP has contacts to notify you of incidents they detect, and that such notifications are integrated into your process. For click-through services, notifications will likely be sent to your registration email address; these should be controlled by the enterprise and monitored continuously. Ensure that you have contacts, including out-of-band methods, for your CSP and that you test them.

- *IaaS/PaaS vs. SaaS:* In a multitenant environment, how can data specific to your cloud be provided for investigation? For each major service you should understand and document what data and logs will be available in an incident. Don't assume you can contact a provider after the fact and collect data that isn't normally available.
- *"Cloud jump kit:"* These are the tools needed to investigate in a remote location (as with cloud-based resources). For example, do you have tools to collect logs and metadata from the cloud platform? Do you have the ability to interpret the information? How do you obtain images of running virtual machines and what kind of data do you have access to: disk storage or volatile memory?

- *Architect the cloud environment for faster detection, investigation, and response (containment and recoverability)*. This means ensuring you have the proper configuration and architecture to support incident response:
    - Enable instrumentation, such as cloud API logs, and ensure that they feed to a secure location that's available to investigators in case of an incident.
    - Utilize isolation to ensure that attacks cannot spread and compromise the entire application.
    - Use immutable servers when possible. If an issue is detected, move workloads from compromised device onto a new instance in a known-good state. Employ a greater focus on file integrity monitoring and configuration management.
    - Implement application stack maps to understand where data is going to reside in order to factor in geographic differences in monitoring and data capture.
    - It can be very helpful to perform threat modeling and tabletop exercises to determine the most effective means of containment for different types of attacks on different components in the cloud stack.
    - This should include differences between responses for IaaS/PaaS/SaaS.

9.1.2.2 Detection and Analysis

Detection and analysis in a cloud environment may look nearly the same (for IaaS) and quite different (for SaaS). In all cases, the monitoring scope must cover the cloud's management plane, not merely the deployed assets.

You may be able to leverage in-cloud monitoring and alerts that can kick off an automated IR workflow in order to speed up the response process. Some cloud providers offer these features for their platforms, and there are also some third-party monitoring options available. These may not be security-specific: Many cloud platforms (IaaS and possibly PaaS) expose a variety of real-time and near-real-time monitoring metrics for performance and operational reasons. But security may also be able to leverage these for security needs.

Cloud platforms also offer a variety of logs, which can sometimes be integrated into existing security operations/monitoring. These could range from operational logs to full logging of all API calls or management activity. Keep in mind that they are not available on all providers; you tend to see them more with IaaS and PaaS than SaaS. When log feeds aren't available you may be able to use the cloud console as a means to identify environment/configuration changes.

Data sources for cloud incidents can be quite different from those used in incident response for traditional computing. There is significant overlap, such as system logs, but there are differences in terms of how data can be collected and in terms of new sources, such as feeds from the cloud management plane.

As mentioned, cloud platform logs may be an option, but they are not universally available. Ideally they should show all management-plane activity. It's important to understand what is logged and the gaps that could affect incident analysis. Is all management activity recorded? Do they include automated system activities (like auto-scaling) or cloud provider management activities? In the case of a serious incident, providers may have other logs that are not normally available to customers.

One challenge in collecting information may be limited network visibility. Network logs from a cloud provider will tend to be flow records, but not full packet capture.

Where there are gaps you can sometimes instrument the technology stack with your own logging. This can work within instances, containers, and application code in order to gain telemetry important for the investigation. Pay particular attention to PaaS and serverless application architectures; you will likely need to add custom application-level logging.

External threat intelligence may also be useful, as it is with on-premises incident response, in order to help identify indicators of compromise and to get adversary information.

Be aware that there are potential challenges when the information that is provided by a CSP faces chain of custody questions. There are no reliable precedents established at this point.

Forensics and investigative support will also need to adapt, beyond understanding changes to data sources.

Always factor in what the CSP can provide and whether it meets chain of custody requirements. Not every incident will result in legal action, but it's important to work with your legal team to understand the lines and where you could end up having chain of custody issues.

There is a greater need to automate many of the forensic/investigation processes in cloud environments, because of their dynamic and higher-velocity nature. For example, evidence could be lost due to a normal auto-scaling activity or if an administrator decides to terminate a virtual machine involved in an investigation. Some examples of tasks you can automate include:

- Snapshotting the storage of the virtual machine.
- Capturing any metadata at the time of alert, so that the analysis can happen based on what the infrastructure looked like at that time.
- If your provider supports it, "pausing" the virtual machine, which will save the volatile memory state.

You can also leverage the capabilities of the cloud platform to determine the extent of the potential compromise:

- Analyze network flows to check if network isolation held up. You can also use API calls to snapshot the network and the virtual firewall rules state, which could give you an accurate picture of the entire stack at the time of the incident.
- Examine configuration data to check if other similar instances were potentially exposed in the same attack.
- Review data access logs (for cloud-based storage, if available) and management plane logs to see if the incident affected or spanned into the cloud platform.
- Serverless and PaaS-based architectures will require additional correlation across the cloud platform and any self-generated application logs.

9.1.2.3 Containment, Eradication and Recovery

Always start by ensuring the cloud management plane/metastructure is free of an attacker. This will often involve invoking break-glass procedures to access the root or master credentials for the cloud account, in order to ensure that attacker activity isn't being masked or hidden from lower-level administrator accounts. Remember: You can't contain an attack if the attacker is still in the management plane. Attacks on cloud assets, such as virtual machines, may sometimes reveal management plane credentials that are then used to bridge into a wider, more serious attack.

The cloud often provides a lot more flexibility in this phase of the response, especially for IaaS. Software-defined infrastructure allows you to quickly rebuild from scratch in a clean environment, and, for more isolated attacks, inherent cloud characteristics—such as auto-scale groups, API calls for changing virtual network or machine configurations, and snapshots—can speed quarantine, eradication, and recovery processes. For example, on many platforms you can instantly quarantine virtual machines by moving the instance out of the auto-scale group, isolating it with virtual firewalls, and replacing it.

This also means there's no need to immediately "eradicate" the attacker before you identify their exploit mechanisms and the scope of the breach, since the new infrastructure/instances are clean; instead, you can simply isolate them. However, you still need to ensure the exploit path is closed and can't be used to infiltrate other production assets. If there is concern that the management plane is breached, be sure to confirm that the templates or configurations for new infrastructure/applications have not been compromised.

That said, these capabilities are not always universal: With SaaS and some PaaS you may be very limited and will thus need to rely more on the cloud provider.

9.1.2.4 Post-mortem

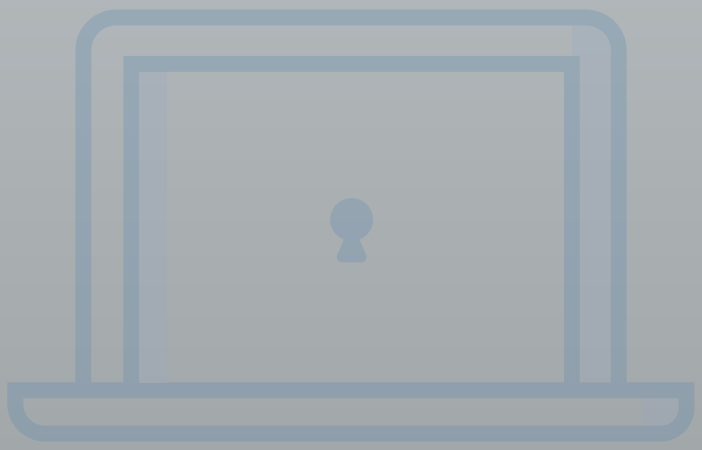
As with any attack, work with the internal response team and provider to figure what worked and what didn't, then pinpoint any areas for improvement. Pay particular attention to the limitations in the data collected and figure out how to address the issues moving forward.

It is hard to change SLAs, but if the agreed-upon response time, data, or other support wasn't sufficient, go back and try to renegotiate.

9.2 Recommendations

- SLAs and setting expectations around what the customer does versus what the provider does are the most important aspects of incident response for cloud-based resources. Clear communication of roles/responsibilities and practicing the response and hand-offs are critical.
- Cloud customers must set up proper communication paths with the provider that can be utilized in the event of an incident. Existing open standards can facilitate incident communication.
- Cloud customers must understand the content and format of data that the cloud provider will supply for analysis purposes and evaluate whether the available forensics data satisfies legal chain of custody requirements.
- Cloud customers should also embrace continuous and serverless monitoring of cloud-based resources to detect potential issues earlier than in traditional data centers.
    - Data sources should be stored or copied into locations that maintain availability during incidents.
    - If needed and possible, they should also be handled to maintain a proper chain of custody.
- Cloud-based applications should leverage automation and orchestration to streamline and accelerate the response, including containment and recovery.
- For each cloud service provider used, the approach to detecting and handling incidents involving the resources hosted at that provider must be planned and described in the enterprise incident response plan.
- The SLA with each cloud service provider must guarantee support for the incident handling required for the effective execution of the enterprise incident response plan. This must cover each stage of the incident handling process: detection, analysis, containment, eradication, and recovery.
- Testing will be conducted at least annually or whenever there are significant changes to the application architecture. Customers should seek to integrate their testing procedures with that of their provider (and other partners) to the greatest extent possible.

DOMAIN 10
Application Security



10.0 Introduction

Application security encompasses an incredibly complex and large body of knowledge: everything from early design and threat modeling to maintaining and defending production applications. Application security is also evolving at an incredibly rapid pace as the practice of application development continues to progress and embrace new processes, patterns, and technologies. Cloud computing is one of the biggest drivers of these advancements and that results in corresponding pressure to evolve the state of application security, in order to ensure that this progress continues as safely as possible.

This section of the guidance is intended for software development and IT teams who want to securely build and deploy applications in cloud computing environments, specifically PaaS and IaaS. (Many of the techniques in this section are used to underpin secure SaaS applications as well.) It focuses on:

- How application security differs in cloud computing.
- Reviewing secure software development basics and how those change in the cloud.
- Leveraging cloud capabilities for more secure cloud applications.

We can't cover all possible development and deployment options—even just the ones directly related to cloud computing—so the goal is to focus on significant areas that should help guide security in the majority of situations. This domain also introduces security fundamentals for DevOps, which is rapidly emerging as a dominant force in cloud-based application development.

Cloud computing mostly brings security benefits to applications, but as with most areas of cloud technology, it does require commensurate changes to existing practices, processes, and technologies that were not designed to operate in the cloud. At a high level, this balance of opportunities and challenges includes:

Opportunities

- *Higher baseline security.* Cloud providers, especially major IaaS and PaaS providers, have

significant economic incentives to maintain higher baseline security than most organizations. In a cloud environment, major baseline security failures completely undermine the trust that a public cloud provider needs in order to maintain relationships with its customer base. Cloud providers are also subject to a wider range of security requirements in order to meet all the regulatory and industry compliance baselines needed to attract customers from those verticals. These combine to strongly motivate cloud providers to maintain extremely high levels of security.

- *Responsiveness.* APIs and automation provide extensive flexibility to build more responsive security programs at a lower cost than in traditional infrastructure. For example, changing firewall rules or deploying new servers with updated code can be handled with a few API calls or through automation.
- *Isolated environments.* Cloud applications can also leverage virtual networks and other structures, including PaaS, for hyper-segregated environments. For example, it is possible, at no additional cost, to deploy multiple application stacks on entirely separate virtual networks, eliminating the ability for an attacker to use one compromised application to attack others behind the perimeter firewalls.
- *Independent virtual machines.* Security is further enhanced by the use of micro-service architectures. Since cloud doesn't require the consumer to optimize the use of physical servers, a requirement that often results in deploying multiple application components and services on a single system, developers can instead deploy more, smaller virtual machines, each dedicated to a function or service. This reduces the attack surface of the individual virtual machines and supports more granular security controls.
- *Elasticity.* Elasticity enables greater use of immutable infrastructure. When using elasticity tools like auto-scale groups, each production system is launched dynamically based on a baseline image, and may be automatically deprovisioned without human interaction. Thus, core operational requirements mean you never want to allow an administrator to log into a system and make changes, since they will be lost during a normal auto-scale activity. This enables the use of immutable servers, where remote administration is completely disabled. We describe immutable servers and infrastructure in more detail in Domain 7.
- *DevOps.* DevOps is a new application development methodology and philosophy focused on automation of application development and deployment. DevOps opens up many opportunities for security to improve code hardening, change management, and production application security, and even to enhance security operations in general.
- *Unified interface.* A unified interface (management interface and APIs) for infrastructure and application services (when using PaaS) provides a more comprehensive view and better management compared to the traditional disparate systems and devices (load balancers, servers, network devices, firewalls, ACLs, etc.), which are often managed by different groups. This creates opportunities to reduce security failures due to lack of communication or full-stack visibility.

Challenges

- *Limited detailed visibility.* Visibility and the availability of monitoring and logging are impacted, requiring new approaches to gathering security-related data. This is especially true when using PaaS, where commonly available logs, such as system or network logs, are often no longer accessible to the cloud user.

- *Increased application scope.* The management plane/metastructure security directly affects the security of any applications associated with that cloud account. Developers and operations will also likely need access to the management plane, as opposed to always going through a different team. Data and sensitive information is also potentially exposable within the management plane. Lastly, modern cloud applications often connect with the management plane to trigger a variety of automated actions, especially when PaaS is involved. For all those reasons, management plane security is now within scope of the application's security and a failure on either side could bridge into the other.
- *Changing threat models.* The cloud provider relationship and the shared security model will need to be included in the threat model, as well as in any operational and incident response plans. Threat models also need to adapt to reflect the technical differences of the cloud provider or platform in use.
- *Reduced transparency.* There may be less transparency as to what is going on within the application, especially as it integrates with external services. For example, you rarely know the entire set of security controls for an external PaaS service integrated with your application.

Overall, there will be changes to application security due to the shared security model. Some of these are directly tied to governance and operations, but there are many more in terms of how you think and plan for the application's security.

10.1 Overview

Due to the broad nature of application security and the many different skill sets and roles involved in an effective application security program, this domain is broken into the following major areas:

- *The Secure Software Development Lifecycle (SSDLC):* How cloud computing affects application security, from design to deployment.
- *Design and Architecture:* Trends in designing applications for cloud computing that affect and can even improve security.
- *DevOps and Continuous Integration/Continuous Deployment (CI/CD):* DevOps and CI/CD are very frequently used in both the development and deployment of cloud applications, and are quickly becoming the dominant models. They bring new security considerations, and again, opportunities to improve security over more manual development and deployment patterns like waterfall.

10.1.1 Introduction to the Secure Software Development Lifecycle and Cloud Computing

The SSDLC describes a series of security activities during all phases of application development, deployment, and operations. There are multiple frameworks used in the industry, including:

- Microsoft's Security Development Lifecycle
- NIST 800-64
- ISO/IEC 27034
- Other organizations, including Open Web Application Security Project (OWASP) and a variety of

application security vendors, also publish their own lifecycle and security activities guidance.

Due to the range of frameworks and differences in terminology, the Cloud Security Alliance breaks these into larger "meta-phases" to help describe the relatively standard set of activities seen across the frameworks. These aren't meant to replace the formal methodologies, but merely provide a descriptive model that we can use to address the major activities, independent of whatever lifecycle an organization will standardize on.

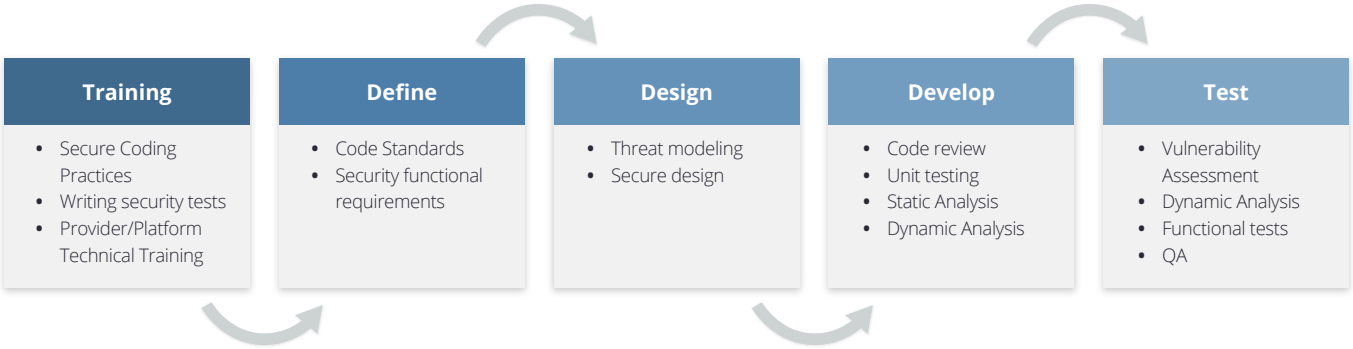
- *Secure Design and Development:* From training and developing organization-wide standards to actually writing and testing code.
- *Secure Deployment:* The security and testing activities when moving code from an isolated development environment into production.
- *Secure Operations:* Securing and maintaining production applications, including external defenses such as Web Application Firewalls (WAF) and ongoing vulnerability assessments.

Cloud computing affects every phase of the SSDLC, regardless of which particular SSDLC you use. This is a direct result of the abstraction and automation of cloud computing, combined with (in the public cloud) a greater reliance on an external provider. Specifically:

- The shared responsibility model means there is always some reliance on the cloud provider for some aspects of security, even in a very bare-bones IaaS-based application. The more you adopt PaaS and provider-specific features, the greater the split in security responsibilities. It could be as simple as using a cloud load balancer, which the provider is completely responsible for keeping secure but the cloud user is responsible for configuring and using properly.
- There are large changes in visibility and control, as discussed in nearly every domain of this Guidance. When running mostly on IaaS it might just be a lack of network logs, but as you move into PaaS it may mean a loss of server and service logs. And, it will all vary based on provider and technology.
- Different cloud providers have different capabilities in terms of features, services, and security, which must be accounted for in the overall application security plan.
- The management plane and metastructure may now be within the application security scope, especially when the application components communicate directly with the cloud service.
- There are new and different architectural options, especially, again, as you consume PaaS.
- The rise and impact of DevOps, which we cover later in this Domain.

10.1.2 Secure Design and Development

There are five main phases in secure application design and development, all of which are affected by cloud computing:



Secure application design and development phases

Training: Three different roles will require two new categories of training. Development, operations, and security should all receive additional training on cloud security fundamentals (which are not provider specific), as well as appropriate technical security training on any specific cloud providers and platforms used on their projects. There is typically greater developer and operations involvement in directly architecting and managing the cloud infrastructure, so baseline security training that's specific to the tools they will use is essential.

Define: The cloud user determines the approved architectures or features/tools for the provider, security standards, and other requirements. This might be tightly coupled to compliance requirements, listing, for example, what kind of data is allowed onto which cloud services (including individual services within a larger provider). At this step the deployment processes should also be defined, although that is sometimes finalized later in a project. Security standards should include the initial entitlements for who is allowed to manage which services in the cloud provider, which is often independent of the actual application architecture. It should also include pre-approved tools, technologies, configurations, and even design patterns.

Design: During the application design process, especially when PaaS is involved, the focus for security in cloud is on architecture, the cloud provider's baseline capabilities, cloud provider features, and automating and managing security for deployment and operations. We find that there are often significant security benefits to integrating security into the application architecture since there are opportunities to leverage the provider's own security capabilities. For example, inserting a serverless load balancer or message queue could completely block certain network attack paths. This is also where you perform threat modeling, which must also be cloud and provider/platform specific.

Develop: Developers may need a development environment with administrative access to the cloud management plane so that they can configure networks, services, and other settings. This should never be a production environment or hold production data. Developers will also likely use a CI/CD pipeline, which must be secured—especially the code repository. If PaaS is used, then

developers should build logging into their application to compensate, as much as possible, for any loss of network, system, or service logs.

Test: Security testing should be integrated into the deployment process and pipeline. Testing tends to span this and the Secure Deployment phase, but leans towards security unit tests, security functional tests, Static Application Security Testing (SAST), and Dynamic Application Security Testing (DAST). Due to the overlap, we cover the cloud considerations in more depth in the next section. Organizations should also rely more on automated testing in cloud. Infrastructure is more often in scope for application testing due to "infrastructure as code," where the infrastructure itself is defined and implemented through templates and automation. As part of security testing, consider requiring flagging features for security-sensitive capabilities that may require deeper security review, such as authentication and encryption code.

10.1.3 Secure Deployment

Since deployment automation tends to be more prominent in cloud environments, it often includes certain security activities that could also be implemented in the Design and Development phase. Automated security testing is very frequently integrated into the deployment pipeline and performed outside of direct developer control. This is in and of itself a departure from many on-premises development efforts, but the testing itself also needs to be adapted for cloud computing.

There are multiple kinds of application security tests that could be potentially integrated into development and deployment:

Code Review: This is a manual activity that's not necessarily integrated into automated testing, but the CI/CD pipeline might impose a manual gate. The review itself doesn't necessarily change for cloud, but there are specific areas that need additional attention. Any application communication with the management plane (e.g., API calls to the cloud service, some of which can alter the infrastructure) should be scrutinized, especially early in the project. Aside from looking at the code itself, the security team can focus on ensuring that only the least privilege entitlements are enabled for that part of the application and then validate them with the management plane configuration. Anything related to authentication and encryption is also important for additional review. The deployment process can then be automated to notify security if there are any modifications to these portions of code that might require manual approval, or just after-the-fact change review.

Unit testing, regression testing, and functional tests: These are the standard tests used by developers in their normal processes. Security testing can and should be integrated in these to ensure that the security features in the application continue to function as expected. The tests themselves will likely need to be updated to account for running in the cloud, including any API calls.

Static Application Security Testing (SAST): On top of the normal range of tests, these should ideally incorporate checks on API calls to the cloud service. They should also look for any static embedded credentials for those API calls, which is a growing problem.

Dynamic Application Security Testing (DAST): DAST tests running applications and includes tests

such as web vulnerability testing and fuzzing. Due to the terms of service with the cloud provider DAST may be limited and/or require pre-testing permission from the provider. With cloud and automated deployment pipelines it is possible to stand up entirely functional test environments using infrastructure as code and then perform deep assessments before approving changes for production.

10.1.3.1 Impact on Vulnerability Assessment

Vulnerability assessment can be integrated into CI/CD pipelines and implemented in cloud fairly easily, but it nearly always requires compliance with the provider's terms of service.

There are two specific patterns we commonly see. The first is running full assessments against images or containers as part of the pipeline in a special testing area of the cloud (a segment of a virtual network) that you define for this purpose. The image is only approved for production deployments if it passes this test. We see a similar pattern used to test entire infrastructures by building a test environment using infrastructure as code.

In both cases production is tested less, or not at all, since it should be immutable and exactly resemble the test environment (both are based on the same definition files). Organizations can also use host-based vulnerability assessment tools, which run locally in a virtual machine and thus do not require coordination with or permission of the cloud provider.

10.1.3.2 Impact on Penetration Testing

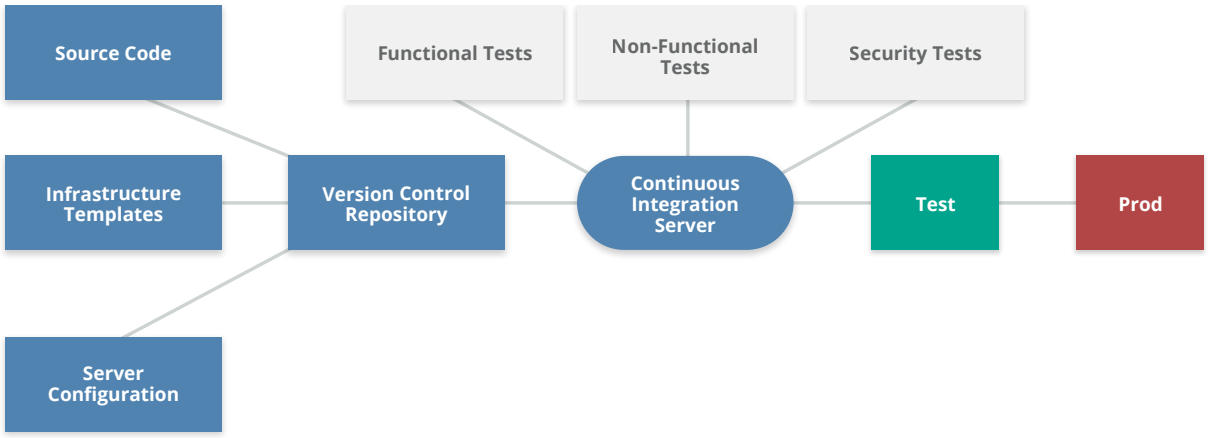
As with vulnerability assessment there will almost certainly be limits on performing penetration tests without the permission of the cloud provider. The CSA recommends adapting penetration testing for cloud using the following guidelines:

- Use a testing firm that has experience on the cloud provider where the application is deployed.
- Include developers and cloud administrators within the scope of the test. Many cloud breaches attack those who maintain the cloud, not the application on the cloud itself. This includes the cloud management plane.
- If the application is a multitenant app, then allow the penetration testers authorized access as a tenant to see if they can compromise the tenancy isolation and use their access to break into another tenant's environment or data.

10.1.3.3 Deployment Pipeline Security

CI/CD pipelines can enhance security through support of immutable infrastructure (fewer manual changes to production environments), automating security testing, and extensive logging of application and infrastructure changes when those changes run through the pipeline. When configured properly, logs can track every code, infrastructure, and configuration change and tie them back to whoever submitted the change and whoever approved it; they will also include any testing results.

The pipeline itself needs to be tightly secured. Consider hosting pipelines in a dedicated cloud environment with very limited access to the cloud or the infrastructure hosting the pipeline components.



A continuous deployment pipeline.

10.1.3.4 Impact of Infrastructure as Code and Immutable

In multiple places we refer to infrastructure as code. Due to the virtual and software-defined nature of cloud it is often possible to define entire environments using templates that are translated by tools (either the provider's or third party) into API calls that automatically build the environment. A basic example is building a server configuration from a template. More complex implementations can build entire cloud application stacks, down to the network configuration and identity management.

Since these environments are built automatically from a set of source file definitions, they can also be immutable. If the system or environment is built automatically from a template, likely from a CI/CD pipeline, then any changes made in production will be overwritten by the next code or template change. The production environment can thus be locked down much more tightly than is normally possible in a non-cloud application deployment, where much of the infrastructure is configured manually to a specification. When security is properly engaged, the use of infrastructure as code and immutable deployments can significantly improve security.

10.1.4 Secure Operations

When an application is deployed into production, security activities continue. Many of these are covered in other areas throughout this Guidance, especially in Domain 7 (Infrastructure), Domain 8 (Containers), Domain 11 (Data), and Domain 12 (Identity and Access Management). This section contains additional guidance that more directly applies to applications:

- The management plane for production environments should be much more tightly locked down than those for development. As previously mentioned, if the application directly accesses the management plane for the environment where it is hosted, then those privileges should be scoped to the least possible required. We recommend using multiple sets of credentials for each application service in order to further compartmentalize entitlements.
- Even when using immutable infrastructure, the production environment should still be actively monitored for changes and deviations from approved baselines. This can and should be

automated through code (or tools) that make API calls to the cloud in order to regularly assess configuration state.

On some cloud platforms it may be possible to use built-in assessment and configuration management features. It may also be possible to automatically remediate unapproved changes, depending on the platform and the nature of the change. For example, code can automatically revert any firewall rule changes that weren't approved by security.

- Even after deployment, and even using immutable infrastructure, don't neglect ongoing application testing and assessment. In public cloud scenarios, this will likely require coordination with or permission of the cloud provider to avoid violating terms of service, just as with any other vulnerability assessment.
- Change management doesn't just include the application, but also any infrastructure and the cloud management plane.

For information on incident response, see Domain 9; for more on business continuity and management plane security, see Domain 6.

10.1.5 How Cloud Impacts Application Design and Architectures

The very nature of cloud is already creating changes in preferred application designs, architectures, and patterns. Some of these have nothing directly to do with security, but the following trends offer opportunities to reduce common security issues:

- *Segregation by default:* Applications can easily be run in their own isolated cloud environments. Depending on the provider, this could be a separate virtual network or account/sub-account. Using accounts or sub-account structures offers the benefit of enabling management plane segregation. The organization can open up wider rights in development accounts while running highly-restrictive production accounts.
- *Immutable infrastructure:* As mentioned, immutable infrastructure is becoming increasingly common in cloud, for operational reasons. Security can extend these benefits by disabling remote logins to immutable servers/containers, adding file integrity monitoring, and integrating immutable techniques into incident recovery plans.
- *Increased use of micro-services:* In cloud computing, it is easier to segregate out different services onto different servers (or containers), since, for one thing, you no longer need to maximize utilization of physical servers and, for another, auto-scale groups can assure application scalability even when using fleets of smaller computer nodes for workloads. Since each node does less, it's easier to lock down and minimize the services running on it. While this improves the security of each workload (when used correctly), it does add some overhead to secure the communications between all the micro-services and ensure that any service discovery, scheduling, and routing is also configured securely.
- *PaaS and "serverless" architectures:* With PaaS and "serverless" setups (running workloads directly on the cloud provider's platform, where you don't manage the underlying services and operating system) there is great potential for dramatically reducing the attack surface. This is only if the cloud provider takes responsibility for the security of the platform/serverless setup and meets your

requirements.

Serverless can bring a few advantages. First, there are large economic incentives for the provider to maintain extremely high security levels and keep their environment up to date. This removes the day-to-day responsibility for keeping these secure from the cloud user, but never obviates their ultimate accountability for security. Working with a trusted cloud provider with a strong track record is critical.

Next, the serverless platforms may run on the provider's network with communications to the consumer's components through API or HTTPS traffic. This removes direct network attack paths, even if an attacker compromises a server or container. The attacker is limited to attempting API calls or HTTPS traffic and can't port scan, identify other servers, or use other common techniques.

- *Software-defined security:* Security teams can leverage all the same tools and technologies to automate many security operations, even integrating them with the application stack. Some examples include automating cloud incident response, automating dynamic changes to entitlements, and remediation of unapproved infrastructure changes.
- *Event driven security:* Certain cloud providers support event-driven code execution. In these cases, the management plane detects various activities—such as a file being uploaded to a designated object storage location or a configuration change to the network or identity management—which can in turn trigger code execution through a notification message, or via serverless hosted code. Security can define events for security actions and use the event-driven capabilities to trigger automated notification, assessment, remediation, or other security processes.

10.1.6 Additional Considerations for Cloud Providers

Cloud providers of all service models need to pay extra attention to certain aspects of their application services that could cause very significant problems for their customers if there are security issues:

- APIs and web services need to be extensively hardened and assume attacks from both authenticated and unauthenticated adversaries. This includes using industry-standard authentication designed specifically for APIs.
- APIs should be monitored for abuse and unusual activity.
- The service should undergo extensive design and testing to prevent attacks or inappropriate/accidental cross-tenant access.

10.1.7 The Rise and Role of DevOps

DevOps refers to the deeper integration of development and operations teams through better collaboration and communications, with a heavy focus on automating application deployment and infrastructure operations. There are multiple definitions, but the overall idea consists of a culture, philosophy, processes, and tools.

At the core is the combination of Continuous Integration and/or Continuous Delivery (CI/CD) through automated deployment pipelines, and the use of programmatic automation tools to better manage infrastructure. DevOps is not exclusive to cloud, but as discussed it is highly attuned to cloud and is growing to become the dominant cloud application delivery model.

10.1.7.1 Security Implications and Advantages

- *Standardization:* With DevOps, anything that goes into production is created by the CI/CD pipeline on approved code and configuration templates. Dev/Test/Prod are all based on the exact same source files, which eliminates any deviation from known-good standards.
- *Automated testing:* As discussed, a wide variety of security testing can be integrated into the CI/CD pipeline, with manual testing added as needed to supplement.
- *Immutable:* CI/CD pipelines can produce master images for virtual machines, containers, and infrastructure stacks very quickly and reliably. This enables automated deployments and immutable infrastructure.
- *Improved auditing and change management:* CI/CD pipelines can track everything, down to individual character changes in source files that are tied to the person submitting the change, with the entire history of the application stack (including infrastructure) stored in a version control repository. This offers considerable audit and change-tracking benefits.
- *SecDevOps/DevSecOps and Rugged DevOps:* These two terms are emerging to describe the integration of security activities into DevOps. SecDevOps/DevSecOps sometimes refers to the use of DevOps automation techniques to improve security operations. Rugged DevOps refers to integration of security testing into the application development process to produce harder, more secure, and more resilient applications.

10.2 Recommendations

- Understand the security capabilities of your cloud providers. Not merely their baseline, but the various platforms and services.
- Build security into the initial design process. Cloud deployments are more often greenfield, creating new opportunities to engage security early.
- Even if you don't have a formal SDLC, consider moving to continuous deployment and automating security into the deployment pipeline.
- Threat modeling, SAST, and DAST (with fuzzing) should all be integrated. Testing should be configured to work in the cloud environment, but also to test for concerns specific to cloud platforms, such as stored API credentials.
- Understand the new architectural options and requirements in the cloud. Update your security policies and standards to support them, and don't merely attempt to enforce existing standards on an entirely different computing model.
- Integrate security testing into the deployment process.
- Use software-defined security to automate security controls.
- Use event-driven security, when available, to automate detection and remediation of security issues.
- Use different cloud environments to better segregate management plane access and provide developers the freedom they need to configure development environments, while also locking down production environments.

DOMAIN 11
Data Security and Encryption

11.0 Introduction

Data security is a key enforcement tool for information and data governance. As with all areas of cloud security, its use should be risk-based since it is not appropriate to secure everything equally.

This is true for data security overall, regardless of whether the cloud is involved. However, many organizations aren't as accustomed to trusting large amounts of their sensitive data—if not all of it—to a third party, or mixing all their internal data into a shared resource pool. As such, the instinct may be to set a blanket security policy for "anything in the cloud" instead of sticking with a risk-based approach, which will be far more secure and cost effective.

For example, encrypting everything in SaaS because you don't trust that provider at all likely means that you shouldn't be using the provider in the first place. But encrypting everything is not a cure-all and may lead to a false sense of security, e.g., encrypting data traffic without ensuring the security of the devices themselves.

By some perspectives information security is data security, but for our purposes this domain will focus on those controls related to securing the data itself, of which encryption is one of the most important.

11.1 Overview

11.1.1 Data Security Controls

Data security controls tend to fall into three buckets. We cover all of these in this section:

- Controlling what data goes into the cloud (and where).
- Protecting and managing the data in the cloud. The key controls and processes are:
  - Access controls
  - Encryption
  - Architecture

- Monitoring/alerting (of usage, configuration, lifecycle state, etc.)
- Additional controls, including those related to the specific product/service/platform of your cloud provider, data loss prevention, and enterprise rights management.

- Enforcing information lifecycle management security.
  - Managing data location/residency.
  - Ensuring compliance, including audit artifacts (logs, configurations).
  - Backups and business continuity, which are covered in Domain 6.

11.1.2 Cloud Data Storage Types

Since cloud storage is virtualized it tends to support different data storage types than used in traditional storage technologies. Below the virtualization layer these might use well-known data storage mechanisms, but the cloud storage virtualization technologies that cloud users access will be different. These are the most common:

Object storage: Object storage is similar to a file system. "Objects" are typically files, which are then stored using a cloud-platform specific mechanism. Most access is through APIs, not standard file sharing protocols, although cloud providers may also offer front-end interfaces to support those protocols.

Volume storage: This is essentially a virtual hard drive for instances/virtual machines.

Database: Cloud platforms and providers may support a variety of different kinds of databases, including existing commercial and open source options, as well as their own proprietary systems. Proprietary databases typically use their own APIs. Commercial or open source databases are hosted by the provider and typically use existing standards for connections. These can be relational or non-relational—the latter includes NoSQL and other key/value storage systems, or file system-based databases (e.g. HDFS).

Application/platform: Examples of these would be a content delivery network (CDN), files stored in SaaS, caching, and other novel options.

Most cloud platforms also use redundant, durable storage mechanisms that often utilize *data dispersion* (sometimes also known as data *fragmentation of bit splitting*). This process takes chunks of data, breaks them up, and then stores multiple copies on different physical storage to provide high durability. Data stored in this way is thus physically dispersed. A single file, for example, would not be located on a single hard drive.

11.1.3 Managing Data Migrations to the Cloud

Before securing the data in the cloud, most organizations want some means of managing what data is stored in private and public cloud providers. This is often essential for compliance as much or more than for security.

To start, define your policies for which data types are allowed and where they are allowed, then tie these to your baseline security requirements. For example, "Personally Identifiable Information (PII) is allowed on x services assuming it meets y encryption and access control requirements."

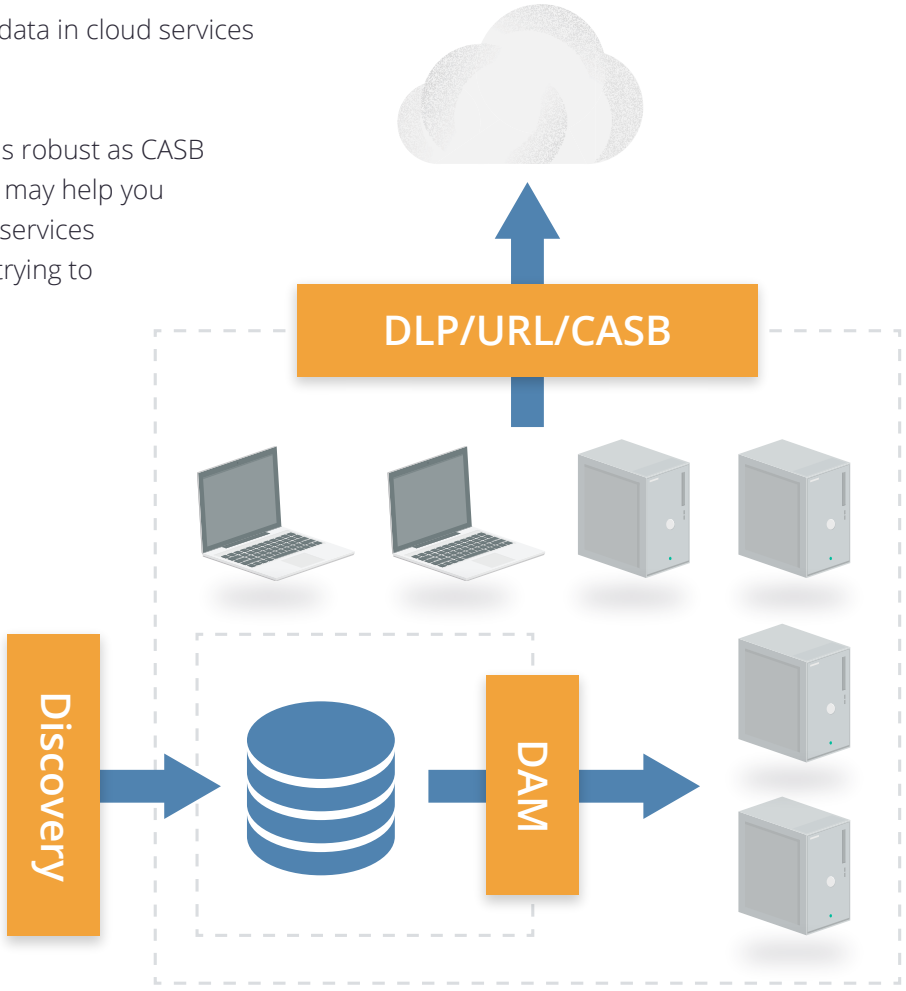
Then identify your key data repositories. Monitor them for large migrations/activity using tools such as Database Activity Monitoring and File Activity Monitoring. This is essentially building an "early warning system" for large data transfers, but it's also an important data security control to detect all sorts of major breaches and misuse scenarios.

To detect actual migrations, monitor cloud usage and any data transfers. You can do this with the help of the following tools:

CASB: Cloud Access and Security Brokers (also known as Cloud Security Gateways) discover internal use of cloud services using various mechanisms such as network monitoring, integrating with an existing network gateway or monitoring tool, or even by monitoring DNS queries. After discovering which services your users are connecting to, most of these products then offer monitoring of activity on approved services through API connections (when available) or inline interception (man in the middle monitoring). Many support DLP and other security alerting and even offer controls to better manage use of sensitive data in cloud services (SaaS/PaaS/and IaaS).

URL filtering: While not as robust as CASB a URL filter/web gateway may help you understand which cloud services your users are using (or trying to use).

DLP: If you monitor web traffic (and look inside SSL connections) a Data Loss Prevention (DLP) tool may also help detect data migrations to cloud services. However, some cloud SDKs and APIs may encrypt portions of data and traffic that DLP tools can't unravel, and thus they won't be able to understand the payload.



Managing data migrations to the cloud.

11.1.3.1 Securing Cloud Data Transfers

Ensure that you are protecting your data as it moves to the cloud. This necessitates understanding your provider's data migration mechanisms, as leveraging provider mechanisms is often more secure and cost effective than "manual" data transfer methods such as Secure File Transfer Protocol (SFTP). For example, sending data to a provider's object storage over an API is likely much more reliable and secure than setting up your own SFTP server on a virtual machine in the same provider.

There are a few options for in-transit encryption depending on what the cloud platform supports. One way is to encrypt before sending to the cloud (client-side encryption). Network encryption (TLS/SFTP/etc.) is another option. Most cloud provider APIs use Transport Layer Security (TLS) by default; if not, pick a different provider, since this is an essential security capability. Proxy-based encryption may be a third option, where you place an encryption proxy in a trusted area between the cloud user and the cloud provider and the proxy manages the encryption before transferring the data to the provider.

In some instances you may have to accept public or untrusted data. If you allow partners or the public to send you data, ensure you have security mechanisms in place to sanitize it before processing or mixing it with your existing data. Always isolate and scan this data before integrating it.

11.1.4 Securing Data in the Cloud

Access controls and encryption are the core data security controls across the various technologies.

11.1.4.1 Cloud Data Access Controls

Access controls should be implemented with a minimum of three layers:

- *Management plane:* These are your controls for managing access of users that directly access the cloud platform's management plane. For example, logging in to the web console of an IaaS service will allow that user to access data in object storage. Fortunately, most cloud platforms and providers start with default deny access control policies.
- *Public and internal sharing controls:* If data is shared externally to the public or partners that don't have direct access to the cloud platform, there will be a second layer of controls for this access.
- *Application level controls:* As you build your own applications on the cloud platform you will design and implement your own controls to manage access.

Options for access controls will vary based on cloud service model and provider-specific features. Create an entitlement matrix based on platform-specific capabilities. An entitlement matrix documents which users, groups, and roles should access which resources and functions.

Entitlement	Super-Admin	Service-Admin	Storage-Admin	Dev	Security-Audit	Security-Admin
Volume Describe	X	X		X	X	X
Object Describe	X		X	X	X	X
Volume Modify	X	X		X		X
Read Logs	X				X	X

Frequently (ideally, continuously) validate that your controls meet your requirements, paying particular attention to any public shares. Consider setting up alerts for all new public shares or for changes in permissions that allow public access.

Fine-Grained Access Controls and Entitlement Mappings

The depth of potential entitlements will vary greatly from technology to technology. Some databases may support row-level security, others little more than broad access. Some will allow you to tie entitlements to identity and enforcement mechanisms built into the cloud platform, while others rely completely on the storage platform itself merely running in virtual machines.

It's important to understand your options, map them out, and build your matrix. This applies to more than just file access, of course; it also applies to databases and all your cloud data stores.

11.1.4.2 Storage (At-Rest) Encryption and Tokenization

Encryption options vary tremendously based on service model, provider, and application/deployment specifics. Key management is just as essential as encryption, and is thus covered in a subsequent section.

Encryption and tokenization are two separate technologies. Encryption protects data by applying a mathematical algorithm that "scrambles" the data, which then can only be recovered by running it through an unscrambling (decryption) process with a corresponding key. The result is a blob of ciphertext. Tokenization, on the other hand, takes the data and replaces it with a random value. It then stores the original and the randomized version in a secure database for later recovery.

Tokenization is often used when the *format* of the data is important (e.g. replacing credit card numbers in an existing system that requires the same format text string). Format Preserving Encryption encrypts data with a key but also keeps the same structural format as tokenization, but it may not be as cryptographically secure due to the compromises.

There are three components of an encryption system: data, the encryption engine, and key management. The data is, of course, the information that you're encrypting. The engine is what performs the mathematical process of encryption. Finally, the key manager handles the keys for the encryption. The overall design of the system focuses on where to put each of these components.

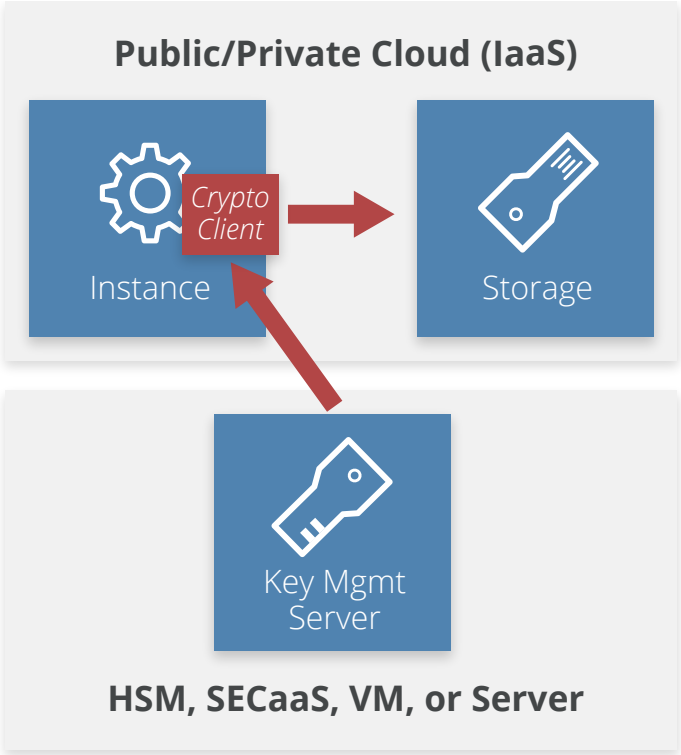
When designing an encryption system, you should start with a threat model. For example, do you trust a cloud provider to manage your keys? How could the keys be exposed? Where should you locate the encryption engine to manage the threats you are concerned with?

IaaS Encryption

IaaS volumes can be encrypted using different methods, depending on your data.

Volume storage encryption

- *Instance-managed encryption:* The encryption engine runs within the instance, and the key is stored in the volume but protected by a passphrase or keypair.
- *Externally managed encryption:* The encryption engine runs in the instance, but the keys are managed externally and issued to the instance on request.



Externally managed volume encryption

Object and file storage

- *Client-side encryption:* When object storage is used as the back-end for an application (including mobile applications), encrypt the data using an encryption engine embedded in the application or client.
- *Server-side encryption:* Data is encrypted on the server (cloud) side after being transferred in. The cloud provider has access to the key and runs the encryption engine.
- *Proxy encryption:* In this model, you connect the volume to a special instance or appliance/software, and then connect your instance to the encryption instance. The proxy handles all crypto operations and may keep keys either onboard or externally.

PaaS Encryption

PaaS encryption varies tremendously due to all the different PaaS platforms.

- *Application layer encryption:* Data is encrypted in the PaaS application or the client accessing the platform.
- *Database encryption:* Data is encrypted in the database using encryption that's built in and is supported by a database platform like Transparent Database Encryption (TDE) or at the field level.
- *Other:* These are provider-managed layers in the application, such as the messaging queue. There are also IaaS options when that is used for underlying storage.

SaaS Encryption

SaaS providers may use any of the options previously discussed. It is recommended to use per-customer keys when possible, in order to better enforce multitenancy isolation. The following options are for SaaS consumers:

- *Provider-managed encryption:* Data is encrypted in the SaaS application and generally managed by the provider.
- *Proxy encryption:* Data passes through an encryption proxy before being sent to the SaaS application.

11.1.4.3 Key Management (Including Customer-Managed Keys)

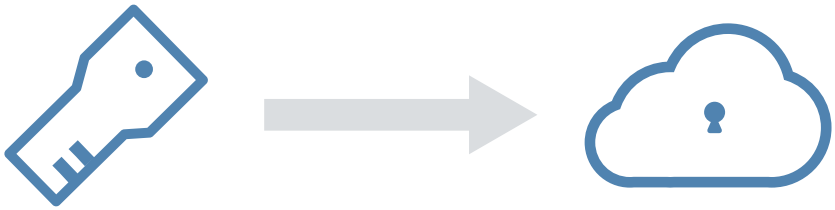
The main considerations for key management are performance, accessibility, latency, and security. Can you get the right key to the right place at the right time while also meeting your security and compliance requirements?

There are four potential options for handling key management:

- *HSM/appliance:* Use a traditional hardware security module (HSM) or appliance-based key manager, which will typically need to be on-premises, and deliver the keys to the cloud over a dedicated connection.
- *Virtual appliance/software:* Deploy a virtual appliance or software-based key manager in the cloud.
- *Cloud provider service:* This is a key management service offered by the cloud provider. Before selecting this option, make sure you understand the security model and SLAs to understand if your key could be exposed.
- *Hybrid:* You can also use a combination, such as using a HSM as the root of trust for keys but then delivering application-specific keys to a virtual appliance that's located in the cloud and only manages keys for its particular context.

Customer-Managed Keys

A customer-managed key allows a cloud customer to manage their own encryption key while the provider manages the encryption engine. For example, using your own key to encrypt SaaS data within the SaaS platform. Many providers encrypt data by default, using keys completely in their control. Some may allow you to substitute your own key, which integrates with their encryption system. Make sure your vendor's practices align with your requirements.



Customer managed keys.

Some providers may require you to use a service within the provider to manage the key. Thus, although the key is customer-managed, it is still potentially available to provider. This doesn't necessarily mean it is insecure: Since the key management and data storage systems can be separated, it would require collusion on the part of multiple employees at the provider to potentially compromise data. However, keys and data could still be exposed by a government request, depending on local laws. You may be able to store the keys externally from the provider and only pass them over on a per-request basis.

11.1.5 Data Security Architectures

Application architecture impacts data security. The features your cloud provider offers can reduce the attack surface, but make sure to demand strong metastructure security. For example, gap networks by using cloud storage or a queue service that communicates on the provider's network, not within your virtual network. That forces attackers to either fundamentally compromise the cloud provider or limit themselves to application-level attacks, since network attack paths are closed.

An example would be using object storage for data transfers and batch processing, rather than SFTP-ing, to static instances. Another is message queue gapping—run application components on different virtual networks that are only bridged by passing data through the cloud provider's message queue service. This eliminates network attacks from one portion of the application to the other.

11.1.6 Monitoring, Auditing, and Alerting

These should tie into overall cloud monitoring. (See Domains 3, 6, and 7.) Identify (and alert about) any public access or entitlement changes on sensitive data. Use tagging to support alerting, when it's available.

You'll need to monitor both API and storage access, since data may be exposed through either—in other words, accessing data in object storage via an API call or via a public sharing URL. Activity monitoring, including Database Activity Monitoring, may be an option. Make sure to store your logs in a secure location, like a dedicated logging account.

11.1.7 Additional Data Security Controls

11.1.7.1 Cloud Platform/Provider-Specific Controls

A cloud platform or provider may have data security controls that are not covered elsewhere in this domain. Although typically they will be some form of access control and encryption, this Guidance can't cover all possible options.

11.1.7.2 Data Loss Prevention

Data Loss Prevention (DLP) is typically a way to monitor and protect data that your employees access via monitoring local systems, web, email, and other traffic. It is not typically used within data centers, and thus is more applicable to SaaS than PaaS or IaaS, where it is typically not deployed.

- *CASB:* Some CASBs include basic DLP features for the sanctioned services they protect. For

example, you could set a policy that a credit card number is never stored in a particular cloud service. The effectiveness depends greatly on the particular tool, the cloud service, and how the CASB is integrated for monitoring. Some CASB tools can also route traffic to dedicated DLP platforms for more robust analysis than is typically available when the CASB offers DLP as a feature.
- *Cloud provider feature:* The cloud provider themselves may offer DLP capabilities, such as a cloud file storage and collaboration platform that scans uploaded files for content and applies corresponding security policies.

11.1.7.3 Enterprise Rights Management

As with DLP, this is typically an employee security control that isn't always as applicable in cloud. Since all Digital Rights Management (DRM)/Enterprise Rights Management (ERM) is based on encryption, existing tools may break cloud capabilities, especially in SaaS.

- *Full DRM:* This is traditional, full digital rights management using an existing tool. For example, applying rights to a file before storing it in the cloud service. As mentioned, it may break cloud provider features, such as browser preview or collaboration, unless there is some sort of integration (which is rare at the time of this writing).
- *Provider-based control:* The cloud platform may be able to enforce controls very similar to full DRM by using native capabilities. For example, user/device/view versus edit: a policy that only allows certain users to view a file in a web browser, while other users can download and/or edit the content. Some platforms can even tie these policies to specific devices, not just on a user level.

11.1.7.4 Data Masking and Test Data Generation

These are techniques to protect data used in development and test environments, or to limit real-time access to data in applications.

- *Test data generation:* This is the creation of a database with non-sensitive test data based on a "real" database. It can use scrambling and other randomization techniques to create a data set that resembles the source in size and structure but lacks sensitive data.
- *Dynamic masking:* Dynamic masking rewrites data on the fly, typically using a proxy mechanism, to mask all or part of data delivered to a user. It is usually used to protect some sensitive data in applications, for example masking out all but the last digits of a credit card number when presenting it to a user.

11.1.8 Enforcing Lifecycle Management Security

- *Managing data location/residency:* At certain times, you'll need to disable unneeded locations. Use encryption to enforce access at the container or object level. Then, even if the data moves to an unapproved location, the data is still protected unless the key moves with it.
- *Ensuring compliance:* You don't merely need to implement controls to maintain compliance, you need to document and test those controls. These are "artifacts of compliance;" this includes any audit artifacts you will have.
- *Backups and business continuity:* See Domain 6.

11.2 Recommendations

- Understand the specific capabilities of the cloud platform you are using.
- Don't dismiss cloud provider data security. In many cases it is more secure than building your own, and comes at a lower cost.
- Create an entitlement matrix for determining access controls. Enforcement will vary based on cloud provider capabilities.
- Consider CASB to monitor data flowing into SaaS. It may still be helpful for some PaaS and IaaS, but rely more on existing policies and data repository security for those types of large migrations.
- Use the appropriate encryption option based on the threat model for your data, business, and technical requirements.
- Consider use of provider-managed encryption and storage options. Where possible, use a customer-managed key.
- Leverage architecture to improve data security. Don't rely completely on access controls and encryption.
- Ensure both API and data-level monitoring are in place, and that logs meet compliance and lifecycle policy requirements
- Standards exist to help establish good security and the proper use of encryption and key management techniques and processes. Specifically, NIST SP-800-57 and ANSI X9.69 and X9.73.

DOMAIN 12
Identity, Entitlement, and Access Management

12.0 Introduction

Identity, entitlement, and access management (IAM) are deeply impacted by cloud computing. In both public and private cloud, two parties are required to manage IAM without compromising security. This domain focuses on what needs to change in identity management for cloud. While we review some fundamental concepts, the focus is on how cloud changes identity management, and what to do about it.

Cloud computing introduces multiple changes to how we have traditionally managed IAM for internal systems. It isn't that these are necessarily new issues, but that they are bigger issues when dealing with the cloud.

The key difference is the relationship between the cloud provider and the cloud user, even in private cloud. IAM can't be managed solely by one or the other and thus a trust relationship, designation of responsibilities, and the technical mechanics to enable them are required. More often than not this comes down to federation. This is exacerbated by the fact that most organizations have many (sometimes hundreds) of different cloud providers into which they need to extend their IAM.

Cloud also tends to change faster, be more distributed (including across legal jurisdictional boundaries), add to the complexity of the management plane, and rely more (often exclusively) on broad network communications for everything, which opens up core infrastructure administration to network attacks. Plus, there are extensive differences between providers and between the different service and deployment models.

> This domain focuses primarily on IAM between an organization and cloud providers or between cloud providers and services. It does not discuss all the aspects of managing IAM within a cloud application, such as the internal IAM for an enterprise application running on IaaS. Those issues are very similar to building similar applications and services in traditional infrastructure.

12.0.1 How IAM is Different in the Cloud

Identity and access management is always complicated. At the heart we are mapping some form of an entity (a person, system, piece of code, etc.) to a verifiable identity associated with various attributes (which can change based on current circumstances), and then making a decision on what they can or can't do based on entitlements. Even when you control the entire chain of that process, managing it across disparate systems and technologies in a secure and verifiable manner, especially at scale, is a challenge.

In cloud computing, the fundamental problem is that multiple organizations are now managing the identity and access management to resources, which can greatly complicate the process. For example, imagine having to provision the same user on dozens—or hundreds—of different cloud services. Federation is the primary tool used to manage this problem, by building trust relationships between organizations and enforcing them through standards-based technologies.

Federation and other IAM techniques and technologies have existed since before the first computers (just ask a bank or government), and over time many organizations have built patchworks and silos of IAM as their IT has evolved. Cloud computing is a bit of a forcing function since adopting cloud very quickly pushes organizations to confront their IAM practices and update them to deal with the differences of cloud. This brings both opportunities and challenges.

At a high level, the migration to cloud is an opportunity to build new infrastructure and processes on modern architectures and standards. There have been tremendous advances in IAM over the years, yet many organizations have only been able to implement them in limited use cases due to budget and legacy infrastructure constraints. The adoption of cloud computing, be it a small project or an entire data center migration, means building new systems on new infrastructure that are generally architected using the latest IAM practices.

These shifts also bring challenges. Moving to federation at scale with multiple internal and external parties can be complex and difficult to manage due to the sheer mathematics of all the variables involved. Determining and enforcing attributes and entitlements across disparate systems and technologies bring both process and technical issues. Even fundamental architectural decisions may be hampered by the wide variation in support among cloud providers and platforms.

IAM spans essentially every domain in this document. This section starts with a quick review of some core terminology that not all readers may be familiar with, then delves into the cloud impacts firstly on identity, then on access and entitlement management.

12.1 Overview

IAM is a broad area of practice with its own lexicon that can be confusing for those who aren't domain specialists, especially since some terms have different meanings in different contexts (and are used in areas outside IAM). Even the term "IAM" is not universal and is often referred to as *Identity Management (IdM)*.

Gartner defines IAM as **"the security discipline that enables the right individuals to access the right resources at the right times for the right reasons."** Before we get into the details, here are the high level terms most relevant to our discussion of IAM in cloud computing:

- *Entity:* the person or "thing" that will have an identity. It could be an individual, a system, a device, or application code.
- *Identity:* the unique expression of an entity within a given namespace. An entity can have multiple digital identities, such as a single individual having a work identity (or even multiple identities, depending on the systems), a social media identity, and a personal identity. For example, if you are a single entry in a single directory server then that is your identity.
- *Identifier:* the means by which an identity can be asserted. For digital identities this is often a cryptological token. In the real world it might be your passport.
- *Attributes:* facets of an identity. Attributes can be relatively static (like an organizational unit) or highly dynamic (IP address, device being used, if the user authenticated with MFA, location, etc.).
- *Persona:* the expression of an identity with attributes that indicates context. For example, a developer who logs into work and then connects to a cloud environment as a developer on a particular project. The identity is still the individual, and the persona is the individual in the context of that project.
- *Role:* identities can have multiple roles which indicate context. "Role" is a confusing and abused term used in many different ways. For our purposes we will think of it as similar to a persona, or as a subset of a persona. For example, a given developer on a given project may have different roles, such as "super-admin" and "dev", which are then used to make access decisions.
- *Authentication:* the process of confirming an identity. When you log in to a system you present a username (the identifier) and password (an attribute we refer to as an authentication factor). Also known as Authn.
- *Multifactor Authentication (MFA):* use of multiple factors in authentication. Common options include one-time passwords generated by a physical or virtual device/token (OTP), out-of-band validation through an OTP sent via text message, or confirmation from a mobile device, biometrics, or plug-in tokens.
- *Access control:* restricting access to a resource. Access management is the process of managing access to the resources.
- *Authorization:* allowing an identity access to something (e.g. data or a function). Also known as Authz.
- *Entitlement:* mapping an identity (including roles, personas, and attributes) to an authorization. The entitlement is what they are allowed to do, and for documentation purposes we keep these in an entitlement matrix.
- *Federated Identity Management:* the process of asserting an identity across different systems or organizations. This is the key enabler of Single Sign On and also core to managing IAM in

cloud computing.

- *Authoritative source:* the "root" source of an identity, such as the directory server that manages employee identities.
- *Identity Provider:* the source of the identity in federation. The identity provider isn't always the authoritative source, but can sometimes rely on the authoritative source, especially if it is a broker for the process.
- *Relying Party:* the system that relies on an identity assertion from an identity provider.

There are a few more terms that will be covered in their relevant sections below, including the major IAM standards. Also, although this domain may seem overly focused on public cloud, all the same principles apply in private cloud; the scope, however, will be lessened since the organization may have more control over the entire stack.

12.1.1 IAM Standards for Cloud Computing

There are quite a few identity and access management standards out there, and many of them can be used in cloud computing. Despite the wide range of options the industry is settling on a core set that are most commonly seen in various deployments and are supported by the most providers. There are also some standards that are promising but aren't yet as widely used. This list doesn't reflect any particular endorsement and doesn't include all options but is merely representative of what is most commonly supported by the widest range of providers:

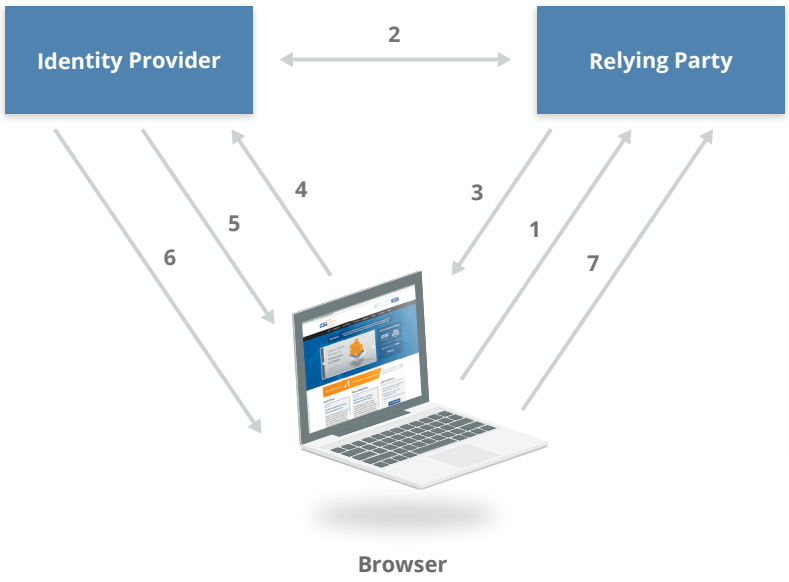
- *Security Assertion Markup Language (SAML)* 2.0 is an OASIS standard for federated identity management that supports both authentication and authorization. It uses XML to make assertions between an identity provider and a relying party. Assertions can contain authentication statements, attribute statements, and authorization decision statements. SAML is very widely supported by both enterprise tools and cloud providers but can be complex to initially configure.
- *OAuth* is an IETF standard for authorization that is very widely used for web services (including consumer services). OAuth is designed to work over HTTP and is currently on version 2.0, which is not compatible with version 1.0. To add a little confusion to the mix, OAuth 2.0 is more of a framework and less rigid than OAuth 1.0, which means implementations may not be compatible. It is most often used for delegating access control/authorizations between services.
- *OpenID* is a standard for federated authentication that is very widely supported for web services. It is based on HTTP with URLs used to identify the identity provider and the user/identity (e.g. identity.identityprovider.com). The current version is OpenID Connect 1.0 and it is very commonly seen in consumer services.

There are two other standards that aren't as commonly encountered but can be useful for cloud computing:

- *eXtensible Access Control Markup Language (XACML)* is a standard for defining attribute-based access controls/authorizations. It is a policy language for defining access controls at a Policy Decision Point and then passing them to a Policy Enforcement Point. It can be used with both SAML and OAuth since it solves a different part of the problem—i.e. deciding what an entity is

allowed to do with a set of attributes, as opposed to handling logins or delegation of authority.

- *System for Cross-domain Identity Management (SCIM)* is a standard for exchanging identity information between domains. It can be used for provisioning and deprovisioning accounts in external systems and for exchanging attribute information.



1. User sends their OpenID URL
2. IP and RP set shared secret
3. Browser redirected to get token from provider
4. Request to IP for token for site
5. Login if needed
6. Token returned to browser
7. Token handed to requesting site

How Federated Identity Management Works: Federation involves an *identity provider* making assertions to a *relying party* after building a trust relationship. At the heart are a series of cryptographic operations to build the trust relationship and exchange credentials. A practical example is a user logging in to their work network, which hosts a directory server for accounts. That user then opens a browser connection to a SaaS application. Instead of logging in, there are a series of behind-the-scenes operations, where the identity provider (the internal directory server) asserts the identity of the user, and that the user authenticated, as well as any attributes. The relying party trusts those assertions and logs the user in without the user entering any credentials. In fact, the relying party doesn't even have a username or password for that user; it relies on the identity provider to assert successful authentication. To the user they simply go to the website for the SaaS application and are logged in, assuming they successfully authenticated with the internal directory.

This isn't to imply there aren't other techniques or standards used in cloud computing for identity, authentication, and authorization. Most cloud providers, especially IaaS, have their own internal IAM systems that might not use any of these standards or that can be connected to an organization using these standards. For example, HTTP request signing is very commonly used for authenticating REST APIs and authorization decisions are managed by internal policies on the cloud provider side. The request signing might still support SSO through SAML, or the API might be completely OAuth based, or use its own token mechanism. All are commonly encountered, but most enterprise-class cloud providers offer federation support of some sort.

Identity protocols and standards do not represent a complete solution by themselves, but they are a means to an end.

The essential concepts when choosing an identity protocol are:

- No protocol is a silver bullet that solves all identity and access control problems.
- Identity protocols must be analyzed in the context of use case(s). For example, Browser-based Single Sign On, API keys, or mobile-to-cloud authentication could each lead companies to a different approach.
- The key operating assumption should be that identity is a perimeter in and of itself, just like a DMZ. So any identity protocol has to be selected and engineered from the standpoint that it can traverse risky territory and withstand malice.

12.1.2 Managing Users and Identities for Cloud Computing

The "identity" part of identity management focuses on the processes and technologies for registering, provisioning, propagating, managing, and deprovisioning identities. Managing identities and provisioning them in systems are problems that information security has been tackling for decades. It wasn't so long ago that IT administrators needed to individually provision users in every different internal system. Even today, with centralized directory servers and a range of standards, true Single Sign On for everything is relatively rare; users still manage a set of credentials, albeit a much smaller set than in the past.

> A note on scope: The descriptions in this section are generic but do skew towards user management. The same principles apply to identities for services, devices, servers, code, and other entities, but the processes and details around those can be more complex and are tightly tied to application security and architectures. This domain also only includes limited discussion of all the internal identity management issues for cloud providers, for the same reasons. It isn't that these areas are less important; in many cases they are more important, but they also bring a complexity that can't be fully covered within the constraints of this Guidance.

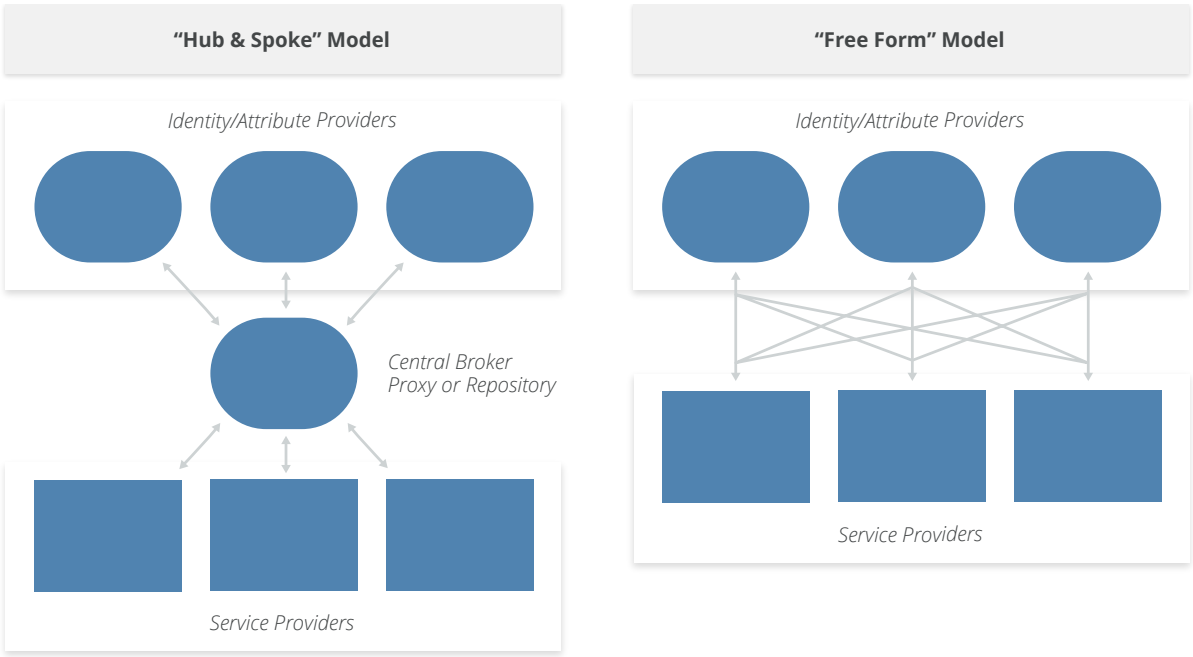
Cloud providers and cloud users need to start with the fundamental decision on how to manage identities:

- Cloud providers need to nearly always support internal identities, identifiers, and attributes for users who directly access the service, while also supporting federation so that organizations don't have to manually provision and manage every user in the provider's system and issue everyone separate credentials.
- Cloud users need to decide where they want to manage their identities and which architectural models and technologies they want to support to integrate with cloud providers.

As a cloud user, you can log in to a cloud provider and create all your identities in their system. This is not scalable for most organizations, which is why most turn to federation. Keep in mind there can be exceptions where it makes sense to keep all or some of the identities with the cloud provider isolated, such as backup administrator accounts to help debug problems with the federated identity connection.

When using federation, the cloud user needs to determine the authoritative source that holds the unique identities they will federate. This is often an internal directory server. The next decision is whether to directly use the authoritative source as the identity provider, use a different identity source that feeds

from the authoritative source (like a directory fed from an HR system), or to integrate an *identity broker*. There are two possible architectures:



Free-form vs. hub and spoke

- *Free-form:* internal identity providers/sources (often directory servers) connect directly to cloud providers.
- *Hub and spoke:* internal identity providers/sources communicate with a central broker or repository that then serves as the identity provider for federation to cloud providers.

Directly federating internal directory servers in the free-form model raises a few issues:

- The directory needs Internet access. This can be a problem, depending on existing topography, or it may violate security policies.
- It may require users to VPN back to the corporate network before accessing cloud services.
- Depending on the existing directory server, and especially if you have multiple directory servers in different organizational silos, federating to an external provider may be complex and technically difficult.

Identity brokers handle federating between identity providers and relying parties (which may not always be a cloud service). They can be located on the network edge or even in the cloud in order to enable web-SSO.

Identity providers don't need to be located only on-premises; many cloud providers now support cloud-based directory servers that support federation internally and with other cloud services. For example, more complex architectures can synchronize or federate a portion of an organization's identities for an internal directory through an identity broker and then to a cloud-hosted directory, which then serves as an identity provider for other federated connections.

After determining the large-scale model, there are still process and architectural decisions required for any implementation:

- How to manage identities for application code, systems, devices, and other services. You may leverage the same model and standards or decide to take a different approach within cloud deployments and applications. For example, the descriptions above skew towards users accessing services, but may not apply equally for services talking to services, systems or devices, or for application components within an IaaS deployment.
- Defining the identity provisioning process and how to integrate that into cloud deployments. There may also be multiple provisioning processes for different use cases, although the goal should be to have as unified a process as possible.
    - If the organization has an effective provisioning process in place for traditional infrastructure this should ideally be extended into cloud deployments. However, if existing internal processes are problematic then the organization should instead use the move to cloud as an opportunity to build a new, more effective process.
- Provisioning and supporting individual cloud providers and deployments. There should be a formal process for adding new providers into the IAM infrastructure. This includes the process of establishing any needed federation connections, as well as:
    - Mapping attributes (including roles) between the identity provider and the relying party.
    - Enabling required monitoring/logging, including identity-related security monitoring, such as behavioral analytics.
    - Building an entitlement matrix (discussed more in the next section).
    - Documenting any break/fix scenarios in case there is a technical failure of any of the federation (or other techniques) used for the relationship.
    - Ensuring incident response plans for potential account takeovers are in place, including takeovers of privileged accounts.
- Implementing deprovisioning or entitlement change processes for identities and the cloud provider. With federation this requires work on both sides of the connection.

Lastly, cloud providers need to determine which identity management standards they wish to support. Some providers support only federation while others support multiple IAM standards plus their own internal user/account management. Providers who serve enterprise markets will need to support federated identity, and most likely SAML.

12.1.3 Authentication and Credentials

Authentication is the process of proving or confirming an identity. In information security authentication most commonly refers to the act of a user logging in, but it also refers to essentially any time an entity proves who they are and assumes an identity. Authentication is the responsibility of the identity provider.

The biggest impact of cloud computing on authentication is a greater need for *strong authentication* using *multiple factors*. This is for two reasons:

- Broad network access means cloud services are always accessed over the network, and often

over the Internet. Loss of credentials could more easily lead to an account takeover by an attacker, since attacks aren't restricted to the local network.
- Greater use of federation for Single Sign On means one set of credentials can potentially compromise a greater number of cloud services.

Multifactor authentication (MFA) offers one of the strongest options for reducing account takeovers. It isn't a panacea, but relying on a single factor (password) for cloud services is very high risk. When using MFA with federation, the identity provider can and should pass the MFA status as an attribute to the relying party.

There are multiple options for MFA, including:

- *Hard tokens* are physical devices that generate one time passwords for human entry or need to be plugged into a reader. These are the best option when the highest level of security is required.
- *Soft tokens* work similarly to hard tokens but are software applications that run on a phone or computer. Soft tokens are also an excellent option but could be compromised if the user's device is compromised, and this risk needs to be considered in any threat model.
- *Out-of-band Passwords* are text or other messages sent to a user's phone (usually) and are then entered like any other one time password generated by a token. Although also a good option, any threat model must consider message interception, especially with SMS.
- *Biometrics* are growing as an option, thanks to biometric readers now commonly available on mobile phones. For cloud services, the biometric is a local protection that doesn't send biometric information to the cloud provider and is instead an attribute that can be sent to the provider. As such the security and ownership of the local device needs to be considered.

For customers, **FIDO** is one standard that may streamline stronger authentication for consumers while minimizing friction.

12.1.4 Entitlement and Access Management

The terms *entitlement*, *authorization*, and *access control* all overlap somewhat and are defined differently depending on the context. Although we defined them earlier in this section, here is a quick review.

An *authorization* is permission to do something—access a file or network, or perform a certain function like an API call on a particular resource.

An *access control* allows or denies the expression of that authorization, so it includes aspects like assuring that the user is authenticated before allowing access.

An *entitlement* maps identities to authorizations and any required attributes (e.g. user x is allowed access to resource y when z attributes have designated values). We commonly refer to a map of these entitlements as an entitlement matrix. Entitlements are often encoded as technical policies for distribution and enforcement.

This is only one definition of these terms and you may see them used differently in other documents. We also use the term access management as the "A" portion of IAM and it refers to the entire process of defining, propagating, and enforcing authorizations.

Sample Entitlement Matrix

Entitlement	Super-Admin	Service-1 Admin	Service-2 Admin	Dev	Security-Audit	Security-Admin
Service 1 List	X	X		X	X	X
Service 2 List	X		X	X	X	X
Service 1 Modify Network	X	X		X		X
Service 2 Modify Security Rule	X	X				X
Read Audit Logs	X				X	X

Here's a real-world cloud example. The cloud provider has an API for launching new virtual machines. That API has a corresponding authorization to allow launching new machines, with additional authorization options for what virtual network a user can launch the VM within. The cloud administrator creates an entitlement that says that users in the developer group can launch virtual machines in only their project network and only if they authenticated with MFA. The group and the use of MFA are attributes of the user's identity. That entitlement is written as a policy that is loaded into the cloud provider's system for enforcement.

Cloud impacts entitlements, authorizations, and access management in multiple ways:

- Cloud providers and platforms, like any other technology, will have their own set of potential authorizations specific to them. Unless the provider supports XACML (rare today) the cloud user will usually need to configure entitlements within the cloud platform directly.
- The cloud provider is responsible for enforcing authorizations and access controls.
- The cloud user is responsible for defining entitlements and properly configuring them within the cloud platform.
- Cloud platforms tend to have greater support for the *Attribute-Based Access Control* (ABAC) model for IAM, which offers greater flexibility and security than the *Role-Based Access Control* (RBAC) model. RBAC is the traditional model for enforcing authorizations and relies on what is often a single attribute (a defined role). ABAC allows more granular and context aware decisions by incorporating multiple attributes, such as role, location, authentication method, and more.
    - ABAC is the preferred model for cloud-based access management.
- When using federation, the cloud user is responsible for mapping attributes, including roles and groups, to the cloud provider and ensuring that these are properly communicated during authentication.

- Cloud providers are responsible for supporting granular attributes and authorizations to enable ABAC and effective security for cloud users.

12.1.5 Privileged User Management

In terms of controlling risk, few things are more essential than privileged user management. The requirements mentioned above for strong authentication should be a strong consideration for any privileged user. In addition, account and session recoding should be implemented to drive up accountability and visibility for privileged users.

In some cases, it will be beneficial for a privileged user to sign in through a separate tightly controlled system using higher levels of assurances for credential control, digital certificates, physically and logically separate access points, and/or jump hosts.

12.2 Recommendations

- Organizations should develop a comprehensive and formalized plan and processes for managing identities and authorizations with cloud services.
- When connecting to external cloud providers, use federation, if possible, to extend existing identity management. Try to minimize silos of identities in cloud providers that are not tied to internal identities.
- Consider the use of identity brokers where appropriate.
- Cloud users are responsible for maintaining the identity provider and defining identities and attributes.
    - These should be based on an authoritative source.
    - Distributed organizations should consider using cloud-hosted directory servers when on-premises options either aren't available or do not meet requirements.
- Cloud users should prefer MFA for all external cloud accounts and send MFA status as an attribute when using federated authentication.
- Privileged identities should always use MFA.
- Develop an entitlement matrix for each cloud provider and project, with an emphasis on access to the metastructure and/or management plane.
- Translate entitlement matrices into technical policies when supported by the cloud provider or platform.
- Prefer ABAC over RBAC for cloud computing.
- Cloud providers should offer both internal identities and federation using open standards.
- There are no magic protocols: Pick your use cases and constraints first and find the right protocol second.

DOMAIN 13
Security as a Service

13.0 Introduction

While most of this Guidance focuses on securing cloud platforms and deployments, this domain shifts direction to cover security services delivered *from* the cloud. These services, which are typically SaaS or PaaS, aren't necessarily used exclusively to protect cloud deployments; they are just as likely to help defend traditional on-premises infrastructure.

Security as a Service (SecaaS) providers offer security capabilities as a cloud service. This includes dedicated SecaaS providers, as well as packaged security features from general cloud-computing providers. Security as a Service encompasses a very wide range of possible technologies, but they must meet the following criteria:

- SecaaS includes security products or services that are delivered as a cloud service.
- To be considered SecaaS, the services must still meet the essential NIST characteristics for cloud computing, as defined in Domain 1.

This section highlights some of the more common categories in the market, but SecaaS is constantly evolving and the descriptions and following list should not be considered canonical. There are examples and services not covered in this document, and more enter the market on a constant basis.

13.1 Overview

13.1.1 Potential Benefits and Concerns of SecaaS

Before delving into the details of the different significant SecaaS categories it is important to understand how SecaaS is different from both on-premises and self-managed security. To do so, consider the potential benefits and consequences.

13.1.1.1 Potential Benefits

- *Cloud-computing benefits.* The normal potential benefits of cloud computing—such as reduced capital expenses, agility, redundancy, high availability, and resiliency—all apply to SecaaS. As with any other cloud provider the magnitude of these benefits depend on the pricing, execution, and capabilities of the security provider.
- *Staffing and expertise.* Many organizations struggle to employ, train, and retain security professionals across relevant domains of expertise. This can be exacerbated due to limitations of local markets, high costs for specialists, and balancing day-to-day needs with the high rate of attacker innovation. As such, SecaaS providers bring the benefit of extensive domain knowledge and research that may be unattainable for many organizations that are not solely focused on security or the specific security domain.
- *Intelligence-sharing.* SecaaS providers protect multiple clients simultaneously and have the opportunity to share data intelligence and data across them. For example, finding a malware sample in one client allows the provider to immediately add it to their defensive platform, thus protecting all other customers. Practically speaking this isn't a magic wand, as the effectiveness will vary across categories, but since intelligence-sharing is built into the service, the potential upside is there.
- *Deployment flexibility.* SecaaS may be better positioned to support evolving workplaces and cloud migrations, since it is itself a cloud-native model delivered using broad network access and elasticity. Services can typically handle more flexible deployment models, such as supporting distributed locations without the complexity of multi-site hardware installations.
- *Insulation of clients.* In some cases, SecaaS can intercept attacks before they hit the organization directly. For example, spam filtering and cloud-based Web Application Firewalls are positioned *between* the attackers and the organization. They can absorb certain attacks before they ever reach the customer's assets.
- *Scaling and cost.* The cloud model provides the consumer with a "Pay as You Grow" model, which also helps organizations focus on their core business and lets them leave security concerns to the experts.

13.1.1.2 Potential Concerns

- *Lack of visibility.* Since services operate at a remove from the customer, they often provide less visibility or data compared to running one's own operation. The SecaaS provider may not reveal details of how it implements its own security and manages its own environment. Depending on the service and the provider, that may result in a difference in data sources and the level of detail available for things like monitoring and incidents. Some information that the customer

may be accustomed to having may look different, have gaps, or not be available at all. The actual evidence and artifacts of compliance, as well as other investigative data, may not meet the customer's goals. All of this can and should be determined before entering into any agreement.

- *Regulation differences.* Given global regulatory requirements, SecaaS providers may be unable to assure compliance in all jurisdictions that an organization operates in.
- *Handling of regulated data.* Customers will also need assurance that any regulated data potentially vacuumed up as part of routine security scanning or a security incident is handled in accordance with any compliance requirements; this also needs to comply with aforementioned international jurisdictional differences. For example, employee monitoring in Europe is more restrictive than it is in the United States, and even basic security monitoring practices could violate workers' rights in that region. Likewise, if a SecaaS provider relocates its operations, due to data center migration or load balancing, it may violate regulations that have geographical restrictions in data residence.
- *Data leakage.* As with any cloud computing service or product, there is always the concern of data from one cloud user leaking to another. This risk isn't unique to SecaaS, but the highly sensitive nature of security data (and other regulated data potentially exposed in security scanning or incidents) does mean that SecaaS providers should be held to the highest standards of multitenant isolation and segregation. Security-related data is also likely to be involved in litigation, law enforcement investigations, and other discovery situations. Customers want to ensure their data will not be exposed when these situations involve another client on the service.
- *Changing providers.* Although simply switching SecaaS providers may on the surface seem easier than swapping out on-premises hardware and software, organizations may be concerned about lock-in due to potentially losing access to data, including historical data needed for compliance or investigative support.
- *Migration to SecaaS.* For organizations that have existing security operations and on-premises legacy security control solutions, the migration to SecaaS and the boundary and interface between any in-house IT department and SecaaS providers must be well planned, exercised, and maintained.

13.1.2 Major Categories of Security as a Service Offerings

There are a large number of products and services that fall under the heading of Security as a Service. While the following is not a canonical list, it describes many of the more common categories seen as of this writing.

13.1.2.1 Identity, Entitlement, and Access Management Services

Identity-as-a-service is a generic term that covers one or many of the services that may comprise an identity ecosystem, such as Policy Enforcement Points (PEP-as-a-service), Policy Decision Points (PDP-as-a-service), Policy Access Points (PAP-as-a-service), services that provide Entities with Identity, services that provide attributes (e.g. Multi-Factor Authentication), and services that provide reputation.

One of the better-known categories heavily used in cloud security is Federated Identity Brokers. These services help intermediate IAM between an organization's existing identity providers (internal

or cloud-hosted directories) and the many different cloud services used by the organization. They can provide web-based Single Sign On (SSO), helping ease some of the complexity of connecting to a wide range of external services that use different federation configurations.

There are two other categories commonly seen in cloud deployments. Strong authentication services use apps and infrastructure to simplify the integration of various strong authentication options, including mobile device apps and tokens for MFA. The other category hosts directory servers in the cloud to serve as an organization's identity provider.

13.1.2.2 Cloud Access and Security Brokers (CASB, also known as Cloud Security Gateways)

These products intercept communications that are directed towards a cloud service or directly connect to the service via API in order to monitor activity, enforce policy, and detect and/or prevent security issues. They are most commonly used to manage an organization's sanctioned and unsanctioned SaaS services. While there are on-premises CASB options, it is also often offered as a cloud-hosted service.

CASBs can also connect to on-premises tools to help an organization detect, assess, and potentially block cloud usage and unapproved services. Many of these tools include risk-rating capabilities to help customers understand and categorize hundreds or thousands of cloud services. The ratings are based on a combination of the provider's assessments, which can be weighted and combined with the organization's priorities.

Most providers also offer basic Data Loss Prevention for the covered cloud services, inherently or through partnership and integration with other services.

Depending on the organization discussing "CASB," the term is also sometimes used to include Federated Identity Brokers. This can be confusing: Although the combination of the "security gateway" and "identity broker" capabilities is possible and does exist, the market is still dominated by independent services for those two capabilities.

13.1.2.3 Web Security (Web Security Gateways)

Web Security involves real-time protection, offered either on-premises through software and/or appliance installation, or via the Cloud by proxying or redirecting web traffic to the cloud provider (or a hybrid of both). This provides an added layer of protection on top of other protection, such as anti-malware software to prevent malware from entering the enterprise via activities such as web browsing. In addition, it can also enforce policy rules around types of web access and the time frames when they are allowed. Application authorization management can provide an extra level of granular and contextual security enforcement for web applications.

13.1.2.4 Email Security

Email Security should provide control over inbound and outbound email, protecting the organization from risks like phishing and malicious attachments, as well as enforcing corporate polices like acceptable use and spam prevention, and providing business continuity options.

In addition, the solution may support policy-based encryption of emails as well as integrating with various email server solutions. Many email security solutions also offer features like digital signatures that enable identification and non-repudiation. This category includes the full range of services, from those as simple as anti-spam features all the way to fully-integrated email security gateways with advanced malware and phishing protection.

13.1.2.5 Security Assessment

Security assessments are third-party or customer-driven audits of cloud services or assessments of on-premises systems via cloud-provided solutions. Traditional security assessments for infrastructure, applications, and compliance audits are well defined and supported by multiple standards such as NIST, ISO, and CIS. A relatively mature toolset exists, and a number of tools have been implemented using the SecaaS delivery model. Using that model, subscribers get the typical benefits of cloud computing: variant elasticity, negligible setup time, low administration overhead, and pay-per-use with low initial investments.

There are three main categories of security assessments:

- Traditional security/vulnerability assessments of assets that are deployed in the cloud (e.g. virtual machines/instances for patches and vulnerabilities) or on-premises.
- Application security assessments, including SAST, DAST, and management of RASP.
- Cloud platform assessment tools that connect directly with the cloud service over API to assess not merely the assets deployed in the cloud, but the cloud configuration as well.

13.1.2.6 Web Application Firewalls

In a cloud-based WAF, customers redirect traffic (using DNS) to a service that analyzes and filters traffic before passing it through to the destination web application. Many cloud WAFs also include anti-DDoS capabilities.

13.1.2.7 Intrusion Detection/Prevention (IDS/IPS)

Intrusion Detection/Prevention systems monitor behavior patterns using rule-based, heuristic, or behavioral models to detect anomalies in activity which might present risks to the enterprise. With IDS/IPS as a service, the information feeds into a service-provider's managed platform, as opposed to the customer being responsible for analyzing events themselves. Cloud IDS/IPS can use existing hardware for on-premises security, virtual appliances for in-cloud (see Domain 7 for the limitations), or host-based agents.

13.1.2.8 Security Information & Event Management (SIEM)

Security Information and Event Management systems aggregate (via push or pull mechanisms) log and event data from virtual and real networks, applications, and systems. This information is then correlated and analyzed to provide real-time reporting on and alerting of information or events that may require intervention or other types of responses. Cloud SIEMs collect this data in a cloud service, as opposed to a customer-managed, on-premises system.

13.1.2.9 Encryption and Key Management

These services encrypt data and/or manage encryption keys. They may be offered by cloud services to support customer-managed encryption and data security. They may be limited to only protecting assets within that specific cloud provider, or they may be accessible across multiple providers (and even on-premises, via API) for broader encryption management. The category also includes encryption proxies for SaaS, which intercept SaaS traffic to encrypt discrete data.

However, encrypting data *outside* a SaaS platform may affect the ability of the platform to utilize the data in question.

13.1.2.10 Business Continuity and Disaster Recovery

Providers of cloud BC/DR services back up data from individual systems, data centers, or cloud services to a cloud platform instead of relying on local storage or shipping tapes. They may use a local gateway to speed up data transfers and local recoveries, with the cloud service serving as the final repository for worst-case scenarios or archival purposes.

13.1.2.11 Security Management

These services roll up traditional security management capabilities, such as EPP (endpoint) protection, agent management, network security, mobile device management, and so on into a single cloud service. This reduces or eliminates the need for local management servers and may be particularly well suited for distributed organizations.

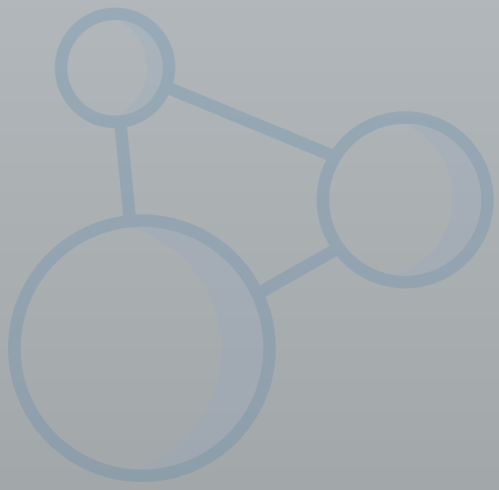
13.1.2.12 Distributed Denial of Service Protection

By nature, most DDoS protections are cloud-based. They operate by rerouting traffic through the DDoS service in order to absorb attacks before they can affect the customer's own infrastructure.

13.2 Recommendations

- Before engaging a SecaaS provider, be sure to understand any security-specific requirements for data-handling (and availability), investigative, and compliance support.
- Pay particular attention to handling of regulated data, like PII.
- Understand your data retention needs and select a provider that can support data feeds that don't create a lock-in situation.
- Ensure that the SecaaS service is compatible with your current and future plans, such as its supported cloud (and on-premises) platforms, the workstation and mobile operating systems it accommodates, and so on.

DOMAIN 14
Related Technologies



14.0 Introduction

Throughout this Guidance we have focused on providing background information and best practices for directly securing cloud computing. As such a foundational technology, there are also a variety of related technologies that bring their own particular security concerns.

While covering all potential uses of cloud is well beyond the scope of this document, CSA feels it is important to include background and recommendations for key technologies that are interrelated with cloud. Some, such as containers and Software-Defined Networks, are so tightly intertwined that we cover them in other respective domains of the Guidance. This Domain provides more depth on additional technologies that don't fit cleanly into existing domains.

Breaking these out into their own section provides more flexibility to update coverage, adding and removing technologies as their usage shifts and new capabilities emerge.

14.1 Overview

Related technologies fall into two broad categories:

- Technologies that rely nearly exclusively on cloud computing to operate.
- Technologies that don't necessarily rely on cloud, but are commonly seen in cloud deployments.

That isn't to say these technologies *can't* work without cloud, just that they are often seen overlapping or relying on cloud deployments and are so commonly seen that they have implications for the majority of cloud security professionals.

The current list includes:

- Big Data

- Internet of Things (IoT)
- Mobile devices
- Serverless computing

Each of these technologies is currently covered by additional Cloud Security Alliance research working groups in multiple ongoing projects and publications:

- **Big Data Working Group**
- **Internet of Things Working Group**
- **Mobile Working Group**

14.1.1 Big Data

Big data includes a collection of technologies for working with extremely large datasets that traditional data-processing tools are unable to manage. It's not any single technology but rather refers commonly to distributed collection, storage, and data-processing frameworks.

Gartner defines it as such: **"Big data is high volume, high velocity, and/or high variety information assets that require new forms of processing to enable enhanced decision making, insight discovery and process optimization."**

The "3 Vs" are commonly accepted as the core definition of big data, although there are many other interpretations.

- *High volume:* a large size of data, in terms of number of records or attributes.
- *High velocity:* fast generation and processing of data, i.e., real-time or stream data.
- *High variety:* structured, semi-structured, or unstructured data.

Cloud computing, due to its elasticity and massive storage capabilities, is very often where big data projects are deployed. Big data is not exclusive to cloud by any means, but big data technologies are very commonly integrated into cloud-computing applications and offered by cloud providers as IaaS or PaaS.

There are three common components of big data, regardless of the specific toolset used:

- *Distributed data collection:* Mechanisms to ingest large volumes of data, often of a streaming nature. This could be as "lightweight" as web-click streaming analytics and as complex as highly distributed scientific imaging or sensor data. Not all big data relies on distributed or streaming data collection, but it is a core big data technology.
- *Distributed storage:* The ability to store the large data sets in distributed file systems (such as Google File System, Hadoop Distributed File System, etc.) or databases (often NoSQL), which is often required due to the limitations of non-distributed storage technologies.
- *Distributed processing:* Tools capable of distributing processing jobs (such as map reduce, spark, etc.) for the effective analysis of data sets so massive and rapidly changing that single-origin processing can't effectively handle them.

14.1.1.1 Security and Privacy Considerations

Due to a combination of the highly distributed nature of big data applications (with data collection, storage, and processing all distributed among diverse nodes) and the sheer volume and potential sensitivity of the information, security and privacy are typically high priorities but are challenged by a patchwork of different tools and platforms.

14.1.1.2 Data Collection

Data collection mechanisms will likely use intermediary storage that needs to be appropriately secured. This storage is used as part of the transfer of data from collection to storage. Even if primary storage is well-secured it's important to also check intermediary storage, which might be as simple as some swap space on a processing node. For example, if collection is run in containers or virtual machines, ensure the underlying storage is appropriately secured. Distributed analysis/processing nodes will also likely use some form of intermediate storage that will need additional security. This could be, for example, the volume storage for instances running processing jobs.

14.1.1.3 Key Management

Key management for storage may be complicated depending on the exact mechanisms used due to the distributed nature of nodes. There are techniques to properly encrypt most big data storage layers today, and these align with our guidance in *Domain 11- Data Security and Encryption*. The complicating factor is that key management needs to handle distributing keys to multiple storage and analysis nodes.

14.1.1.4 Security Capabilities

Not all big data technologies have robust security capabilities. In some cases cloud provider security capabilities can help compensate for the big data technology limitations. Both should be included in any security architecture and the details will be specific to the combination of technologies selected.

14.1.1.5 Identity and Access Management

Identity and Access Management will likely occur at both cloud and big data tool levels, which can complicate entitlement matrices.

14.1.1.6 PaaS

Many cloud providers are expanding big data support with *machine learning* and other platform as a service options that rely on access to enterprise data. These should not be used without a full understanding of potential data exposure, compliance, and privacy implications. For example, if the machine learning runs as PaaS inside the provider's infrastructure, where provider employees could technically access it, does that create a compliance exposure?

This doesn't mean you shouldn't use the services, it just means you need to understand the implications and make appropriate risk decisions. Machine learning and other analysis services aren't necessarily insecure and don't necessarily violate privacy and compliance commitments.

14.1.2 Internet of Things (IoT)

The Internet of Things is a blanket term for non-traditional computing devices used in the physical world that utilize Internet connectivity. It includes everything from Internet-enabled operational technology (used by utilities like power and water) to fitness trackers, connected lightbulbs, medical devices, and beyond. These technologies are increasingly deployed in enterprise environments for applications such as:

- Digital tracking of the supply chain.
- Digital tracking of physical logistics.
- Marketing, retail, and customer relationship management.
- Connected healthcare and lifestyle applications for employees, or delivered to consumers.

A very large percentage of these devices connect back to cloud computing infrastructure for their back-end processing and data storage. Key cloud security issues related to IoT include:

- Secure data collection and sanitization.
- Device registration, authentication, and authorization. One common issue encountered today is use of stored credentials to make direct API calls to the back-end cloud provider. There are known cases of attackers decompiling applications or device software and then using those credentials for malicious purposes.
- API security for connections from devices back to the cloud infrastructure. Aside from the stored credentials issue just mentioned, the APIs themselves could be decoded and used for attacks on the cloud infrastructure.
- Encrypted communications. Many current devices use weak, outdated, or non-existent encryption, which places data and the devices at risk.
- Ability to patch and update devices so they don't become a point of compromise. Currently, it is common for devices to be shipped as-is and never receive security updates for operating systems or applications. This has already caused multiple significant and highly publicized security incidents, such as massive botnet attacks based on compromised IoT devices.

14.1.3 Mobile

Mobile computing is neither new nor exclusive to cloud, but a very large percentage of mobile applications connect to cloud computing for their back-end processing. Cloud can be an ideal platform to support mobile since cloud providers are geographically distributed and designed for the kinds of highly dynamic workloads commonly experienced with mobile applications. This section won't discuss overall mobile security, just the portions that affect cloud security.

The primary security issues for mobile computing (in the cloud context) are very similar to IoT, except a mobile phone or tablet is also a general purpose computer:

- Device registration, authentication, and authorization are common sources of issues. Especially (again), the use of stored credentials, and even more so when the mobile device

connects directly to the cloud provider's infrastructure/APIs. Attackers have been known to decompile mobile applications to reveal stored credentials which are then used to directly manipulate or attack the cloud infrastructure. Data stored on the device should also be protected with the assumption that the user of the device may be a hostile attacker.
- Application APIs are also a potential source of compromise. Attackers are known to sniff API connections, in some cases using local proxies that they redirect their own devices towards, and then decompile the (likely now unencrypted) API calls and explore them for security weaknesses. Certificate pinning/validation inside the device application may help reduce this risk.

For additional recommendations on the security of mobile and cloud computing see the latest research from the CSA **Mobile Working Group**.

14.1.4 Serverless Computing

Serverless computing is the extensive use of certain PaaS capabilities to such a degree that all or some of an application stack runs in a cloud provider's environment without any customer-managed operating systems, or even containers.

"Serverless computing" is a bit of a misnomer since there is always a server running the workload someplace, but those servers and their configuration and security are completely hidden from the cloud user. The consumer only manages settings for the service, and not any of the underlying hardware and software stacks.

Serverless includes services such as:

- Object storage
- Cloud load balancers
- Cloud databases
- Machine learning
- Message queues
- Notification services
- Code execution environments (These are generally restricted containers where a consumer runs uploaded application code.)
- API gateways
- Web servers

Serverless capabilities may be deeply integrated by the cloud provider and tied together with event-driven systems and integrated IAM and messaging to support construction of complex applications without any customer management of servers, containers, or other infrastructure.

From a security standpoint, key issues include:

- Serverless places a much higher security burden on the cloud provider. Choosing your provider and understanding security SLAs and capabilities is absolutely critical.
- Using serverless, the cloud user will not have access to commonly-used monitoring and

logging levels, such as server or network logs. Applications will need to integrate more logging, and cloud providers should provide necessary logging to meet core security and compliance requirements.

- Although the provider's services may be certified or attested for various compliance requirements, not necessarily every service will match every potential regulation. Providers need to keep compliance mappings up to date, and customers need to ensure they only use services within their compliance scope.
- There will be high levels of access to the cloud provider's management plane since that is the only way to integrate and use the serverless capabilities.
- Serverless can dramatically reduce attack surface and pathways and integrating serverless components may be an excellent way to break links in an attack chain, even if the entire application stack is not serverless.
- Any vulnerability assessment or other security testing must comply with the provider's terms of service. Cloud users may no longer have the ability to directly test applications, or must test with a reduced scope, since the provider's infrastructure is now hosting everything and can't distinguish between legitimate tests and attacks.
- Incident response may also be complicated and will definitely require changes in process and tooling to manage a serverless-based incident.

14.2 Recommendations

- Big data
  - Leverage cloud provider capabilities wherever possible, even if they overlap with big data tool security capabilities. This ensures you have proper protection within the cloud metastructure and the specific application stack.
  - Use encryption for primary, intermediary, and backup storage for both data collection and data storage planes.
  - Include both the big data tool and cloud platform Identity and Access Management in the project entitlement matrix.
  - Fully understand the potential benefits and risks of using a cloud machine-learning or analytics service. Pay particular attention to privacy and compliance implications.
    - Cloud providers should ensure customer data is not exposed to employees or other administrators using technical and process controls.
    - Cloud providers should clearly publish which compliance standards their analytics and machine-learning services are compliant with (for their customers).
    - Cloud users should consider use of data masking or obfuscation when considering a service that doesn't meet security, privacy, or compliance requirements.
  - Follow additional big data security best practices, including those provided by the tool vendor (or Open Source project) and the **Cloud Security Alliance**.
- Internet of Things
  - Ensure devices can be patched and upgraded.
  - Do not store static credentials on devices that could lead to compromise of the cloud application or infrastructure.
  - Follow best practices for secure device registration and authentication to the cloud-side

application, typically using a federated identity standard.
    - Encrypt communications.
    - Use a secure data collection pipeline and sanitize data to prevent exploitation of the cloud application or infrastructure through attacks on the data-collection pipeline.
    - Assume all API requests are hostile.
    - Follow the additional, more-detailed guidance issued by the CSA **Internet of Things Working Group**.
- Mobile
    - Follow your cloud provider's guidance on properly authenticating and authorizing mobile devices when designing an application that connects directly to the cloud infrastructure.
    - Use industry standards, typically federated identity, for connecting mobile device applications to cloud-hosted applications.
    - Never transfer unencrypted keys or credentials over the Internet.
    - Test all APIs under the assumption that a hostile attacker will have authenticated, unencrypted access.
        - Consider certificate pinning and validation inside mobile applications.
        - Validate all API data and sanitize for security.
        - Implement server/cloud-side security monitoring for hostile API activity.
    - Ensure all data stored on device is secured and encrypted.
        - Sensitive data that could allow compromise of the application stack should not be stored locally on-device where a hostile user can potentially access it.
    - Follow the more detailed recommendations and research issued by the **CSA Mobile Working Group**.
- Serverless Computing
    - Cloud providers must clearly state which PaaS services have been assessed against which compliance requirements or standards.
    - Cloud users must only use serverless services that match their compliance and governance obligations.
    - Consider injecting serverless components into application stacks using architectures that reduce or eliminate attack surface and/or network attack paths.
    - Understand the impacts of serverless on security assessments and monitoring.
        - Cloud users will need to rely more on application-code scanning and logging and less on server and network logs.
    - Cloud users must update incident response processes for serverless deployments.
    - Although the cloud provider is responsible for security below the serverless platform level, the cloud user is still responsible for properly configuring and using the products.