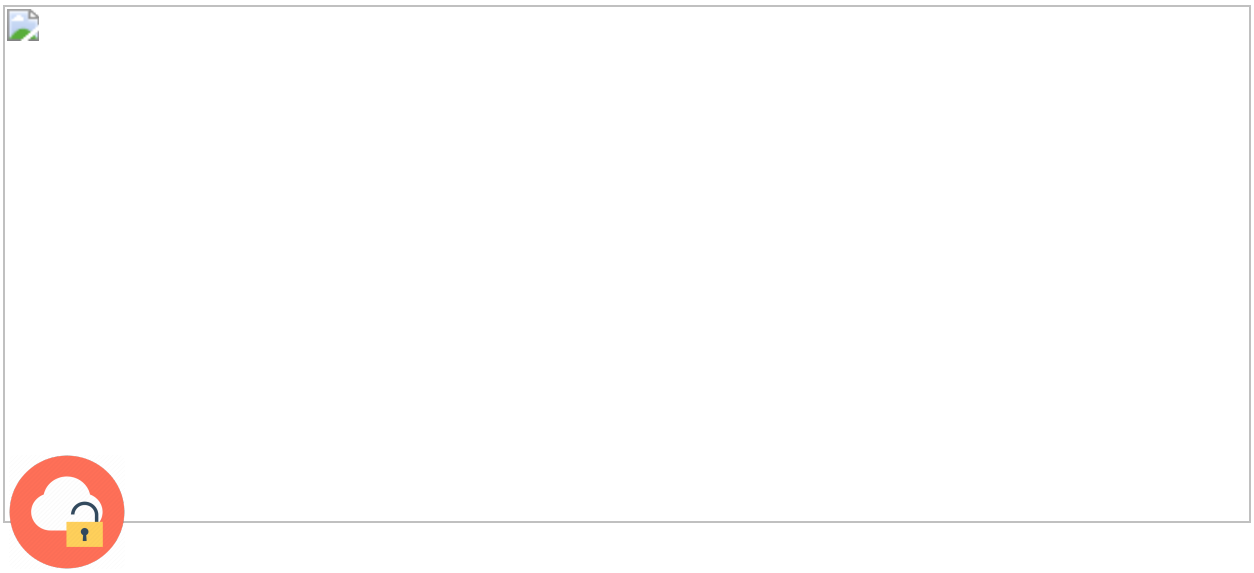


CCSK - Pass 01/08/22

- Security Guidance V4
  - Domain 1: Cloud Computing Concepts and Architectures
    - What is the cloud?

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.

▼ What is a cloud user?

> 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.

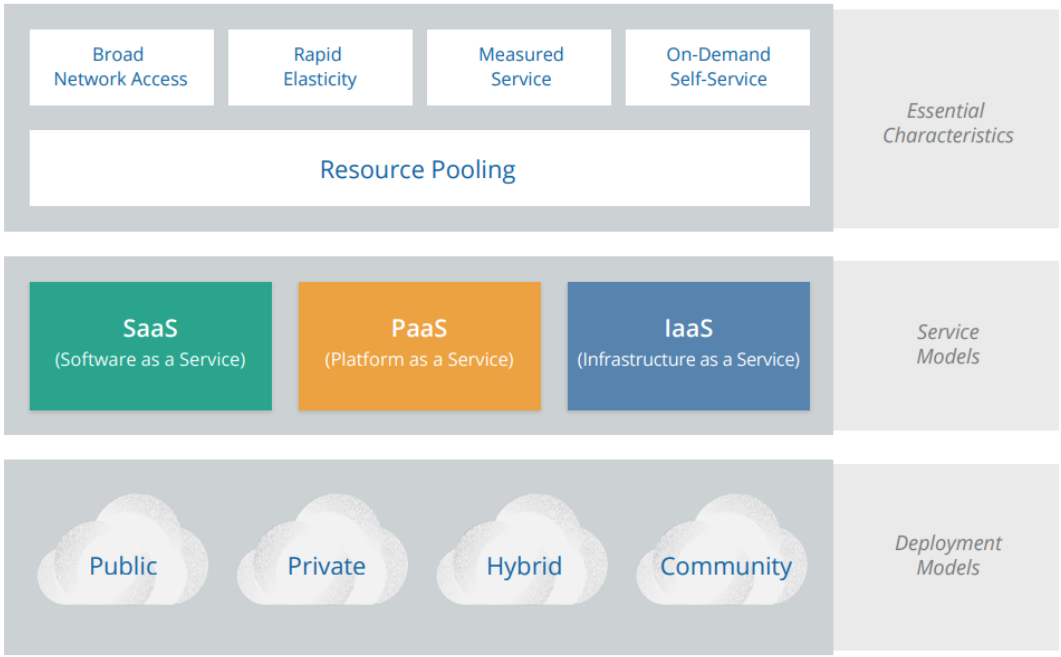
▼ Difference between virtualization and cloud?

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.

▼ Cloud definition models



	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.

▼ 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.

▼ 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.

▼ Deployment models

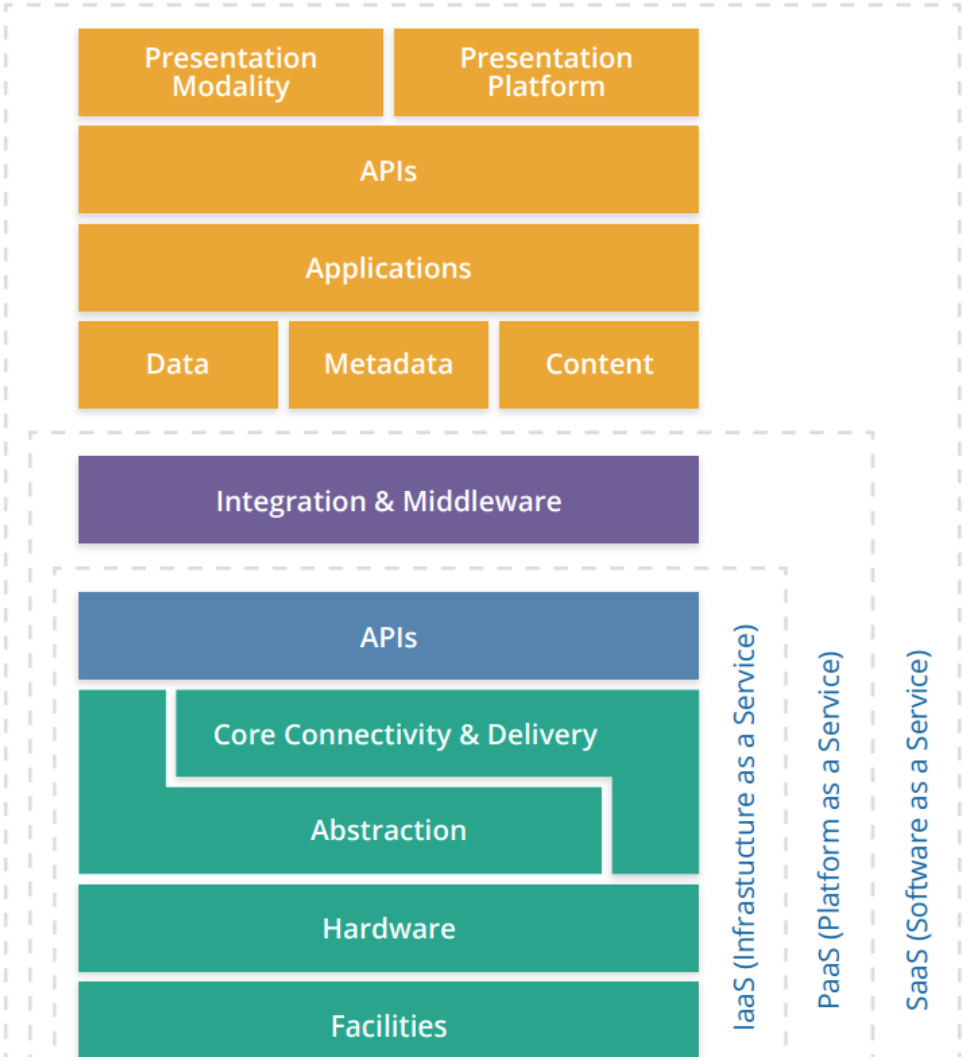
Both NIST and ISO/IEC use the same four cloud deployment models. These are how the technologies are deployed and consumed, and they apply across the entire range of service models:

- *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.

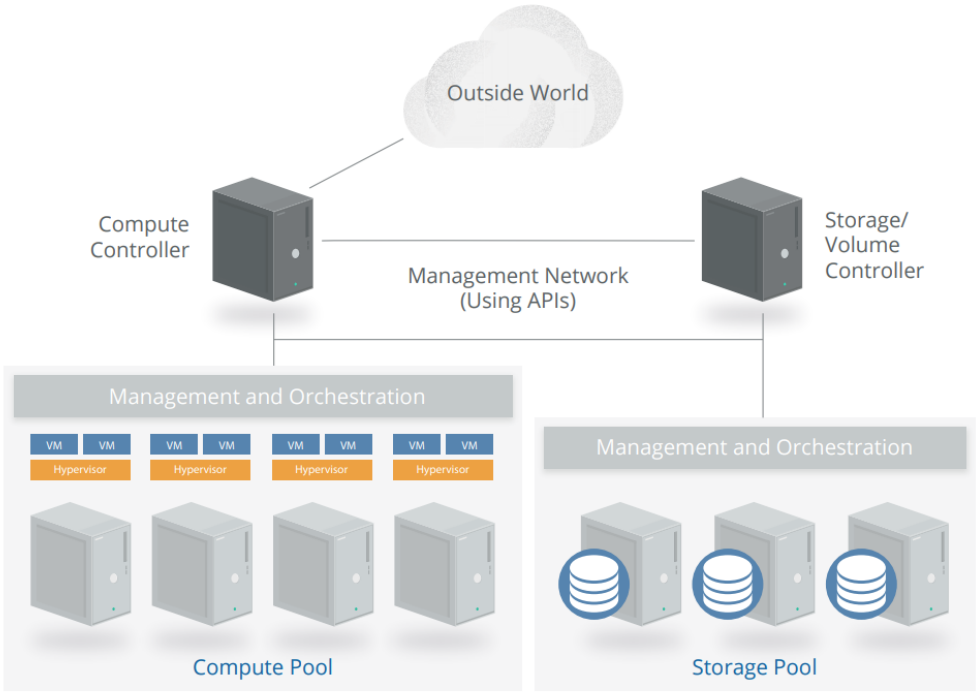
▼ Reference Architecture and models

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.



Iaas:

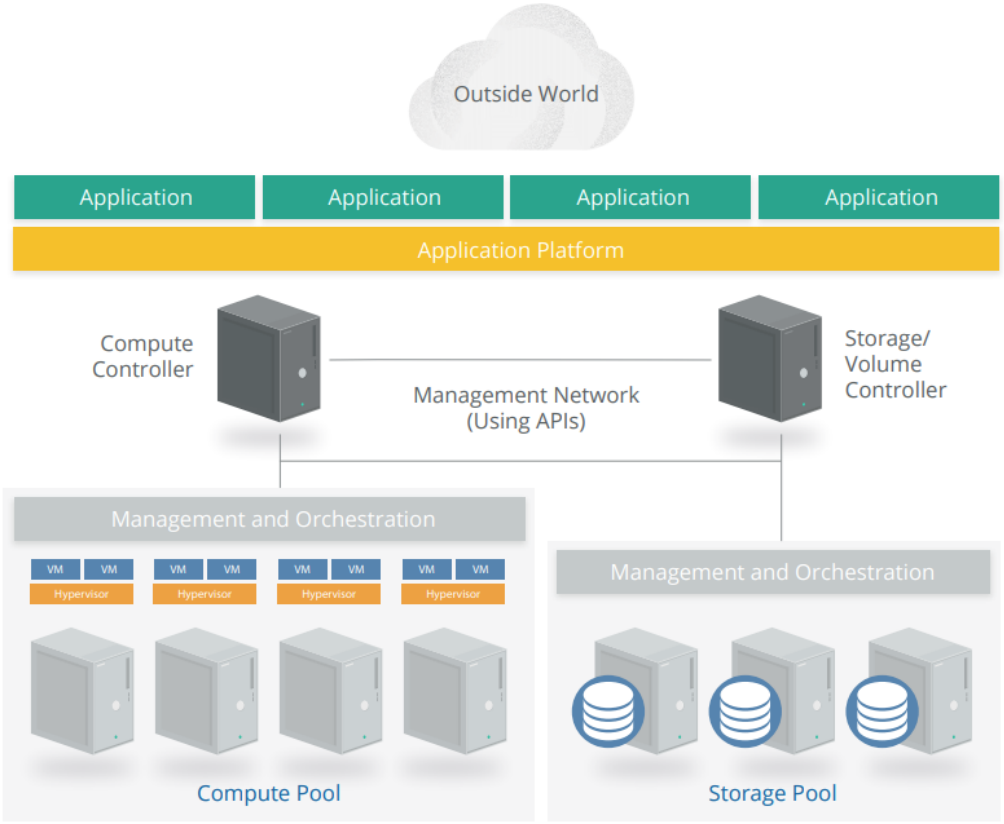
Here is a simplified architectural example of a compute IaaS platform:



> *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.*

PaaS:

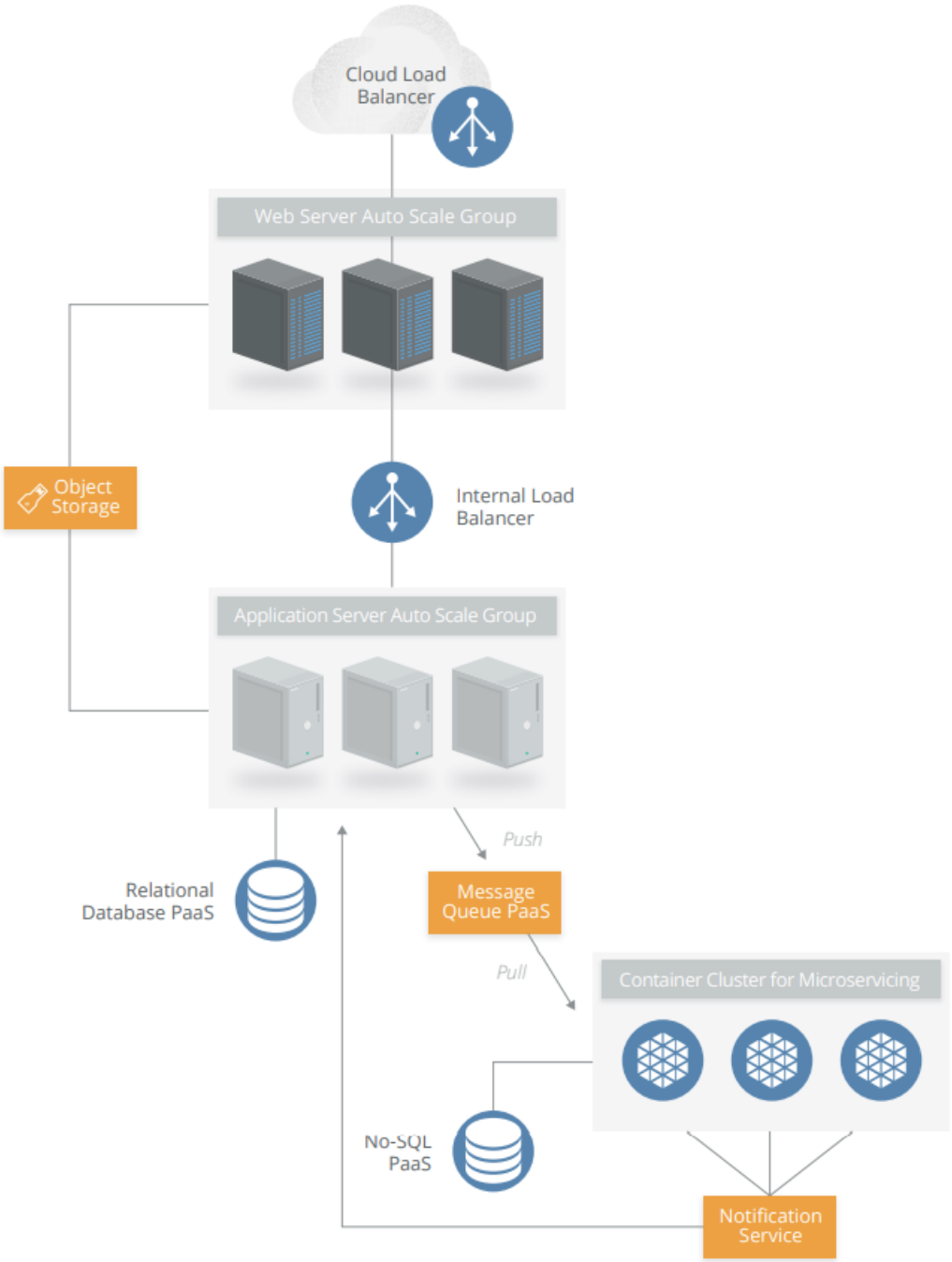
This simplified architecture diagram shows an application platform (PaaS) running on top of our IaaS architecture:



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).

SaaS:

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

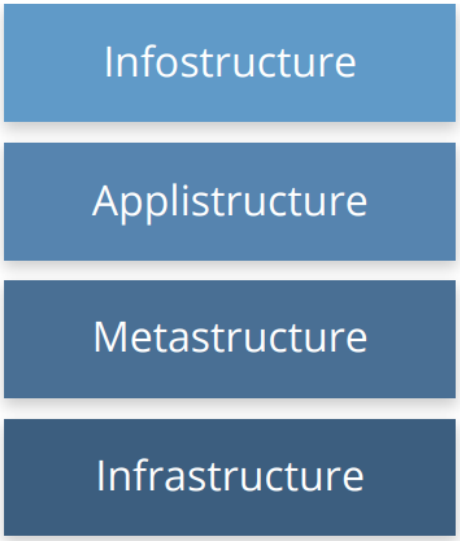


▼ Logical model

- *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.



▼ 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**.

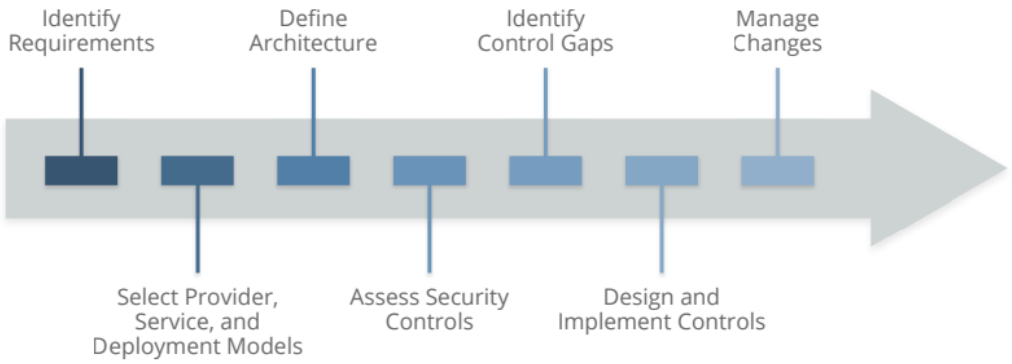
▼ Cloud security process model

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.



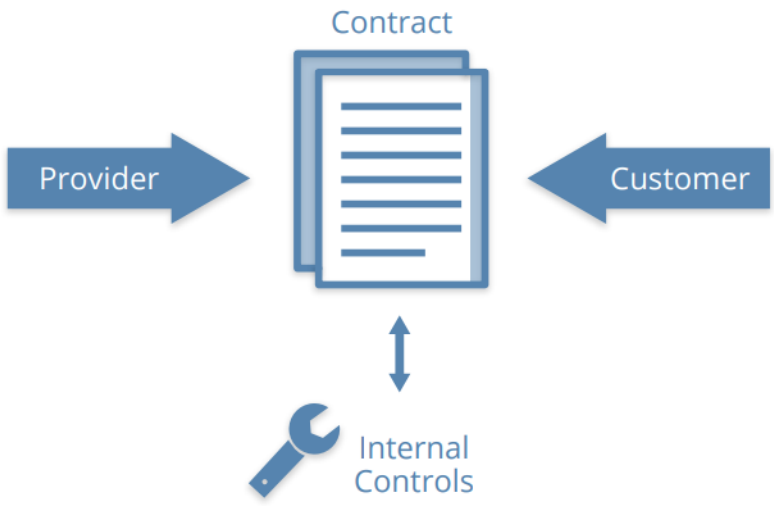
▼ Recommendations

- 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.

▼ Domain 2: Governance and Enterprise Risk Mgmt

▼ Cloud Governance: Contract

- *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.

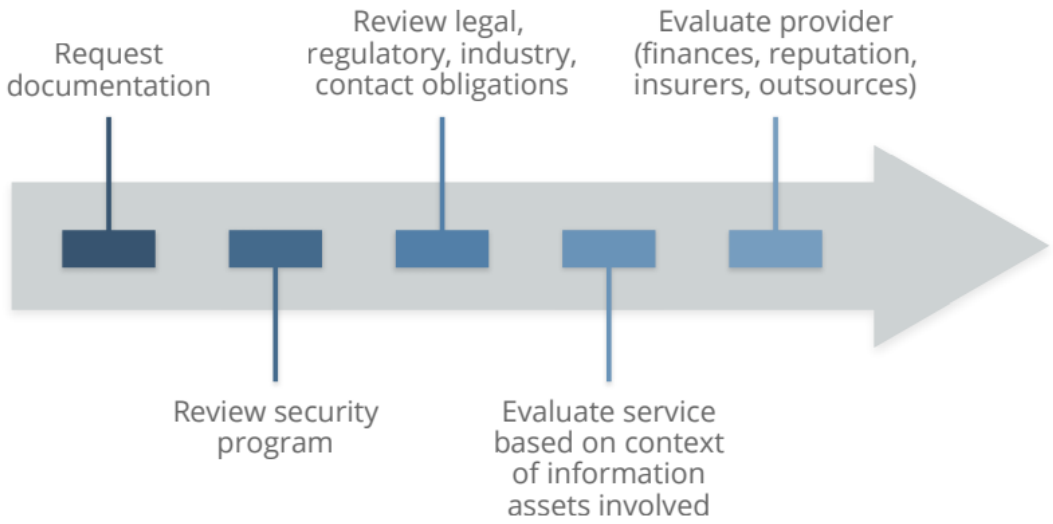


> Contracts define the relationship between providers and customers and are the primary tool for customers to extend governance to their suppliers.

▼ Risk mgmt: supplier assessments

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.

▼ 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

▼ Legal 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.*

▼ Recommendations

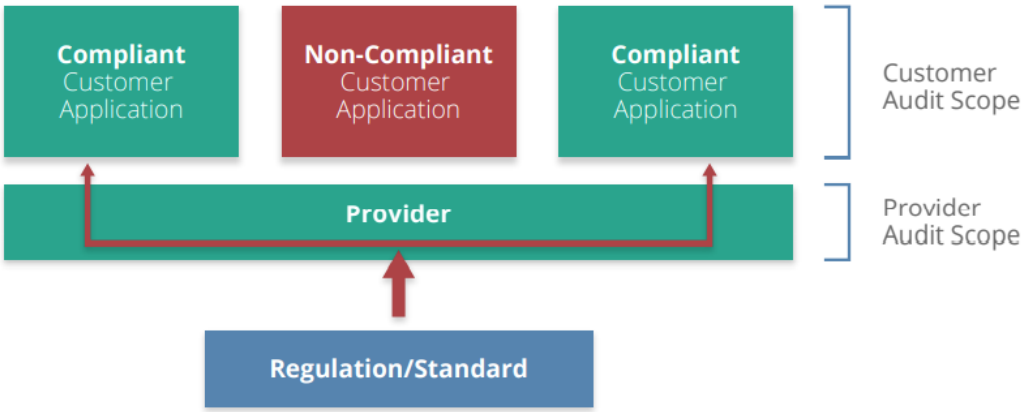
- Cloud customers should understand the relevant legal and regulatory frameworks, as well as contractual requirements and restrictions that apply to the handling of their data or data in their custody, and the conduct of their operations, before moving systems and data to the cloud.
- Cloud providers 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

▼ How the cloud changes compliance

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 Audits

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.

▼ 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

▼ Data security lifecycle and control mapping

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.

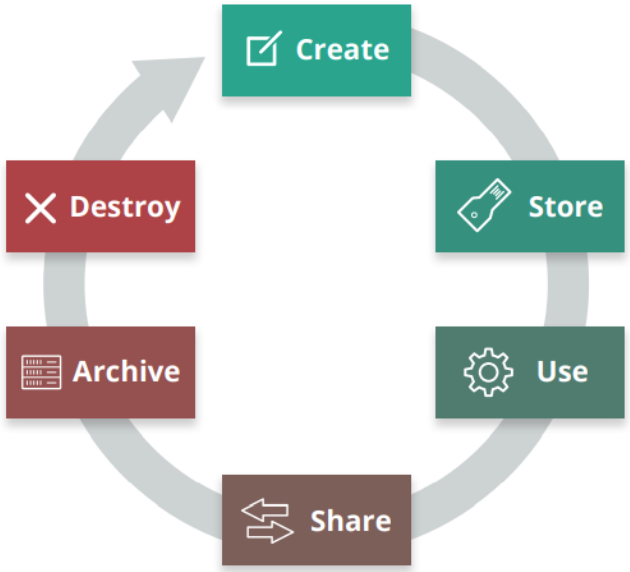
Store. Storing is the act committing the digital data to some sort of storage repository and typically occurs nearly simultaneously with creation.

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

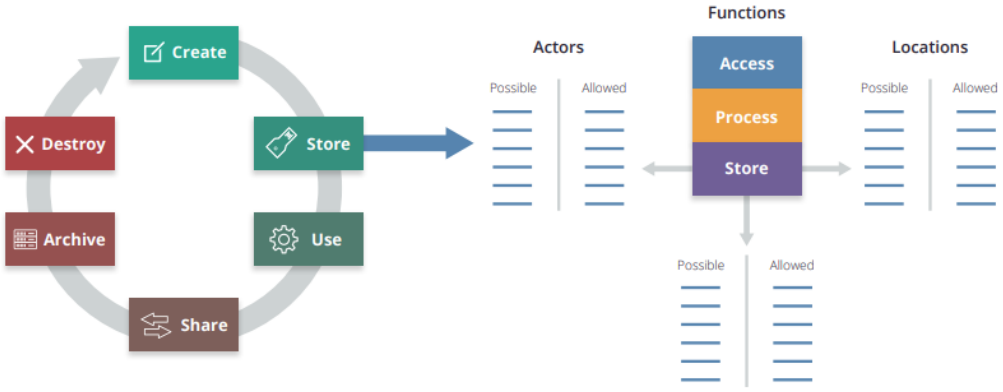
Share. Information is made accessible to others, such as between users, to customers, and to partners.

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



Mapping the lifecycle to functions and controls.

▼ 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

▼ 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

▼ Immutable workloads

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*.

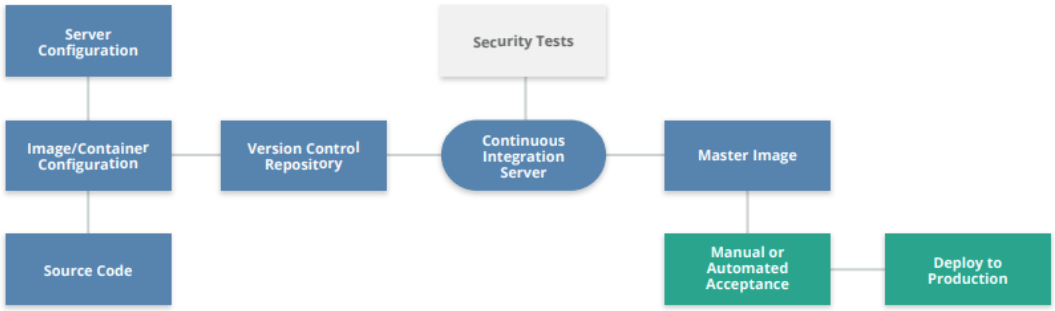
Immutable workloads enable significant security benefits:

- 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.

▼ 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

▼ 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

▼ IR 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

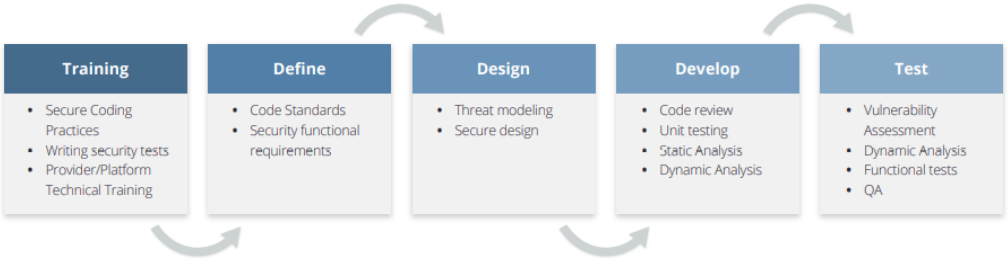
▼ Recommendations

- 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

▼ Secure Design and Development Process

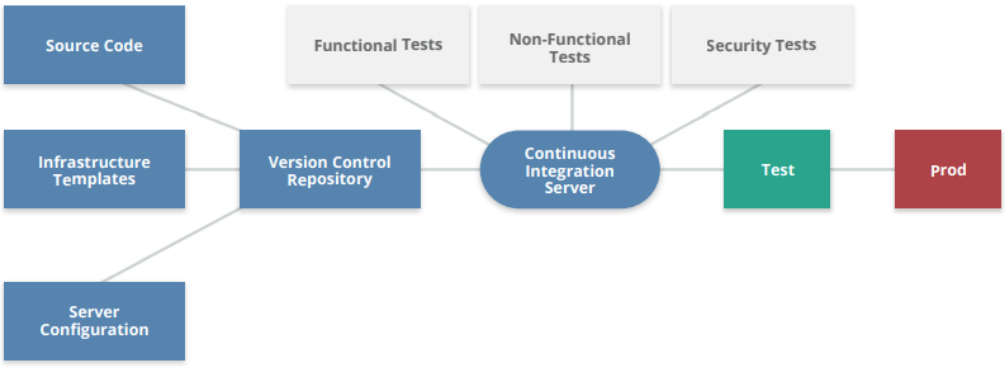
There are five main phases in secure application design and development, all of which are affected by cloud computing:



▼ 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.

▼ 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.

▼ 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

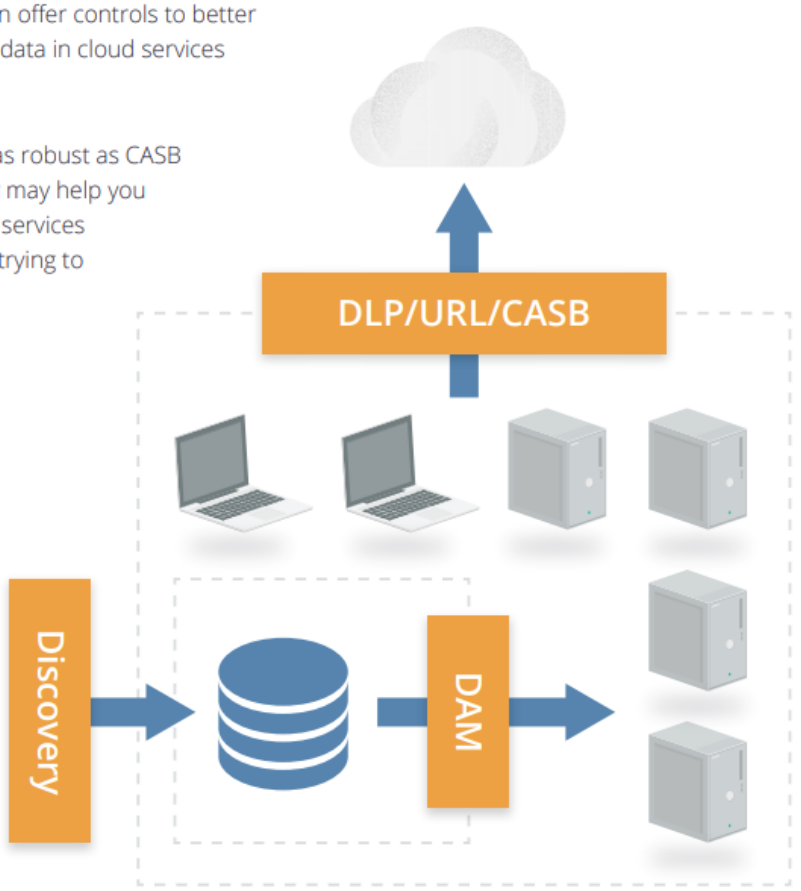
▼ Managing data migrations to the cloud

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.



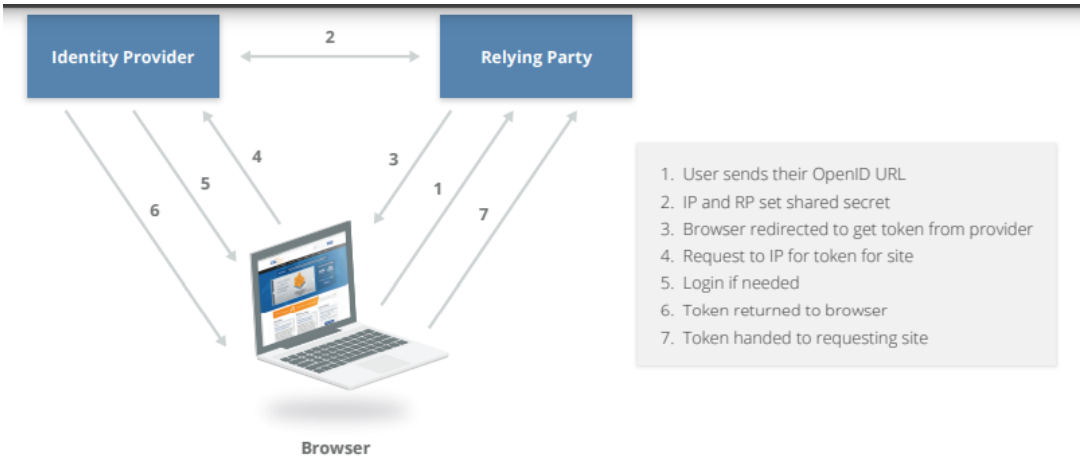
Managing data migrations to the cloud.

▼ 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

▼ Federated identity mgmt



> 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.

▼ 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

▼ Types

- IAM
- CASB
- Web Security (Web security gateways)
- Email Security
- Security Assessment
- WAF
- IDS/IPS
- SIEM
- Encryption and Key mgmt
- BC/DR
- DDoS protection

▼ 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

▼ 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.

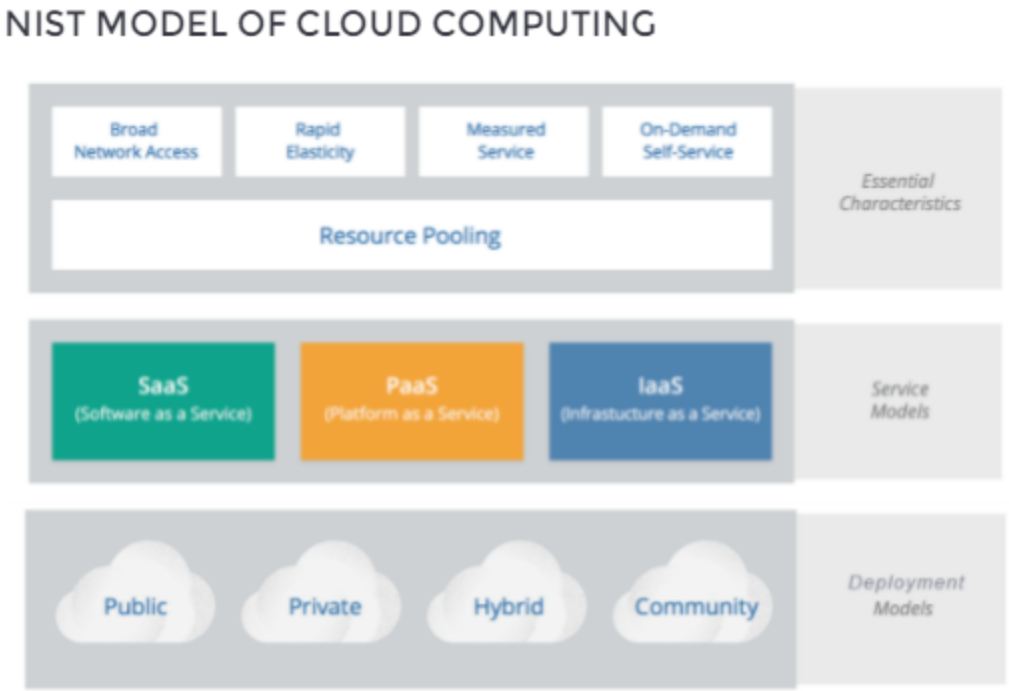
▼ CSSK Course

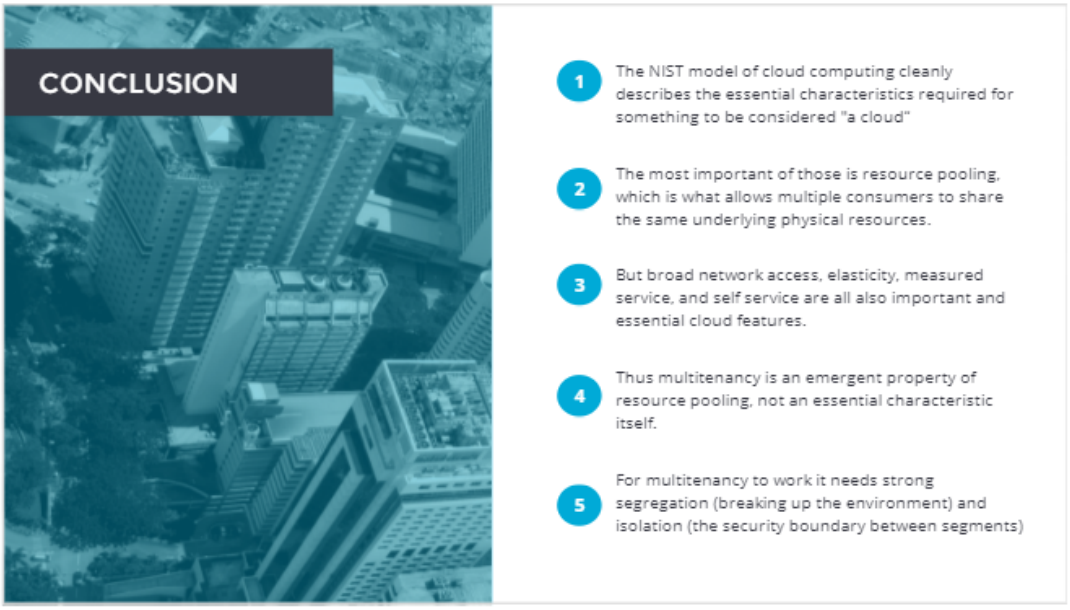
▼ Module 1: Cloud Architecture

▼ Intro & Cloud Architecture

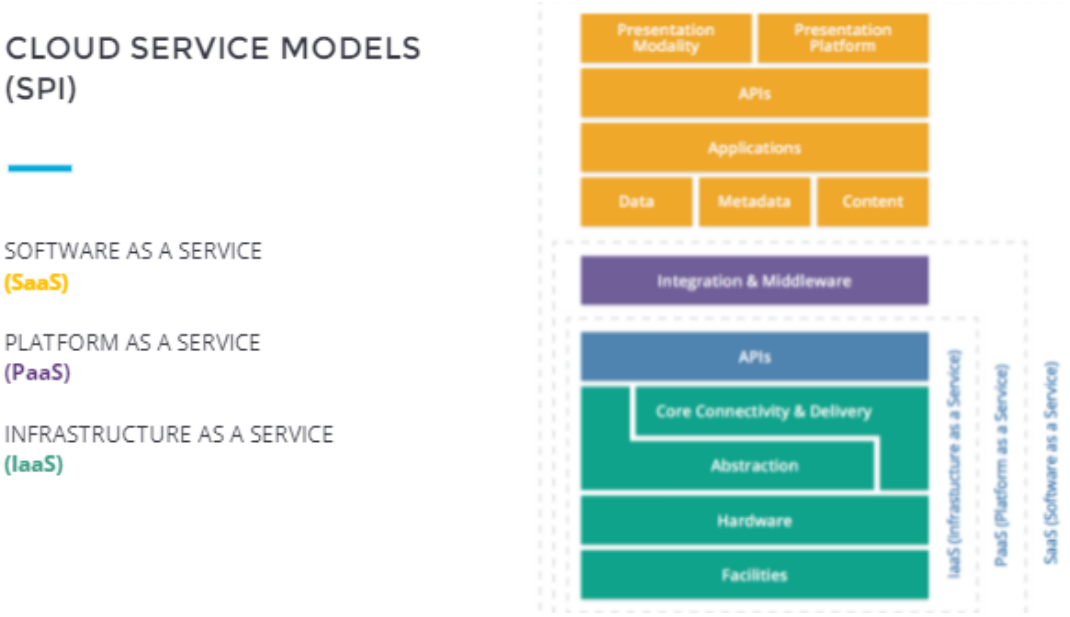
1. Cloud computing is a new operational model that combines the benefits of abstraction (virtualization) and automation (orchestration) for new ways of delivering and consuming technology.
2. Abstraction separates resources from their underlying physical infrastructure. It allows us to create resource pools out of those underlying assets.
3. Automation (orchestration) allows us to rapidly provision and deprovision those resources from the resource pool.
4. This is different than traditional virtualization which includes the abstraction piece, but doesn't necessarily use that to build resource pools, and lacks the advanced orchestration of cloud.
5. Cloud can potentially provide a wide range of benefits, but the key ones are economic, agility, and resiliency.

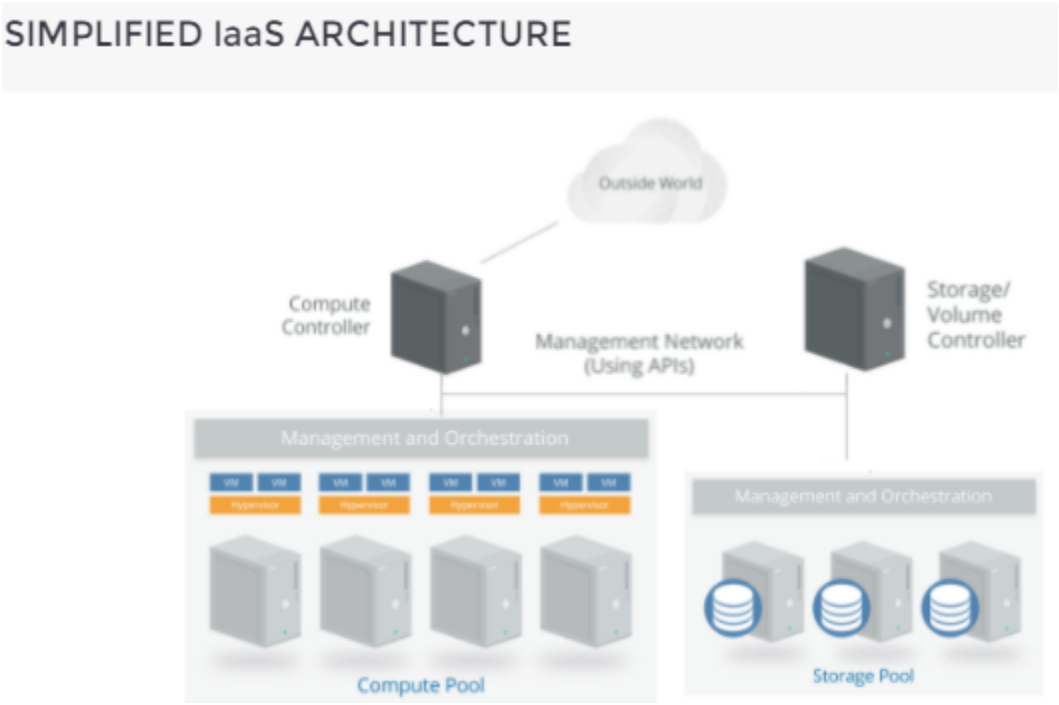
▼ Cloud Essential Characteristics





▼ Cloud Service Models



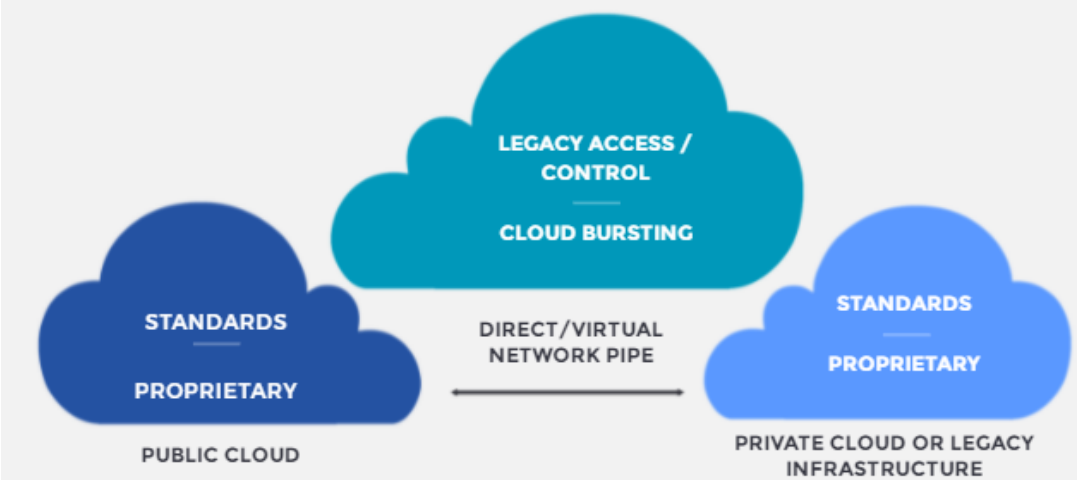


▼ Cloud Deployment Models

CLOUD DEPLOYMENT MODELS & RESPONSIBILITIES

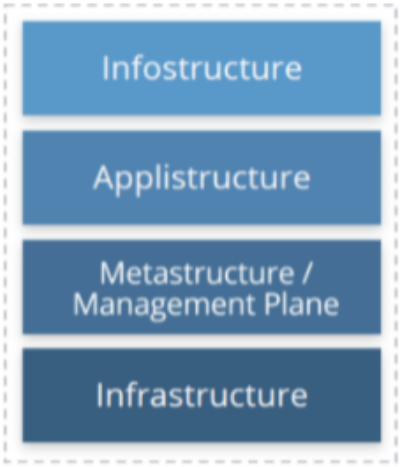
	Infrastructure Managed By[1]	Infrastructure Owned By[2]	Infrastructure Located[3]	Accessible and Consumed By[4]
Public	Third Party Provider	Third Party Provider	Off-Premise	Untrusted
Private/ Community	Organization / Third Party Provider	Organization / Third Party Provider	On-Premise / Off-Premise	Trusted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

HYBRID CLOUD



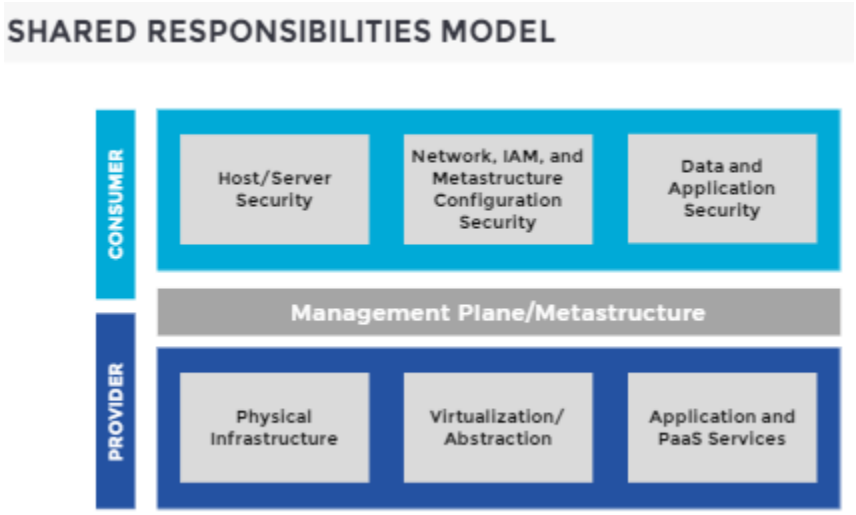
LOGICAL MODEL

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.



1. Deployment models describe how the cloud (regardless of service model) is offered to consumers. The easiest way to think about it is "who gets to use the cloud?"
2. Public clouds are open to anyone who signs up for the service, which means different cloud consumers do not know or trust each other and the cloud provider is responsible for keeping them isolated.
3. Private and community clouds are reserved only for trusted users; those from the same organization or a group of trusted organizations. Someone else can still own and operate the cloud, but only the trusted users are allowed.
4. Hybrid cloud connects on premise resources to a public cloud deployment.
5. The logical model is a different way of describe how we distribute our resources and application components and is useful in showing how data, infrastructure, application, or management components are organized across environments.

▼ Shared Responsibilities



1. Cloud describes the use of pools of compute, network, information, and storage resources.
2. Characteristics of cloud services include: broad network access, rapid elasticity, measured service, on-demand self service, and resource pooling
3. Cloud services tend to be delivered as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), or Software as a Service (SaaS), though the distinctions are blurring.
4. Cloud services can be deployed as public, private, hybrid or community clouds depending on the security and sharing requirements of the application.

▼ Module 2: Infra Security for Cloud Computing

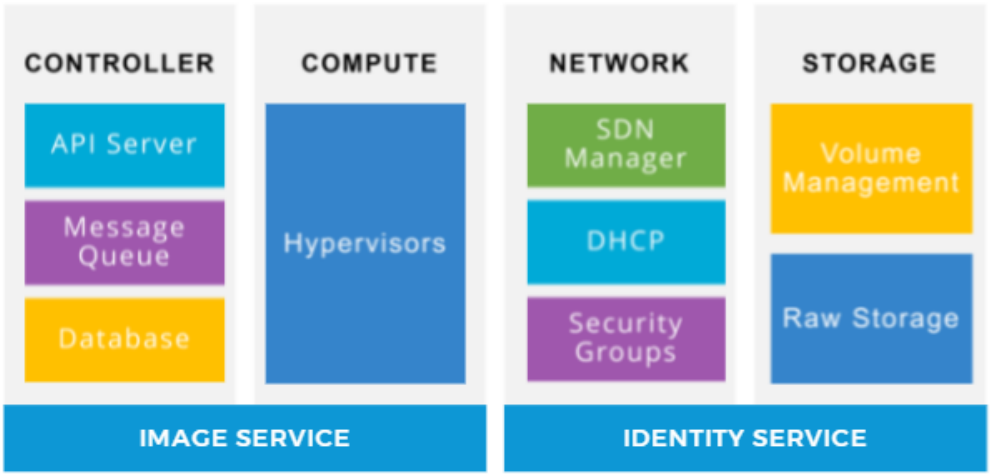
▼ Intro to Infrastructure Security for Cloud Computing

MACRO LAYERS

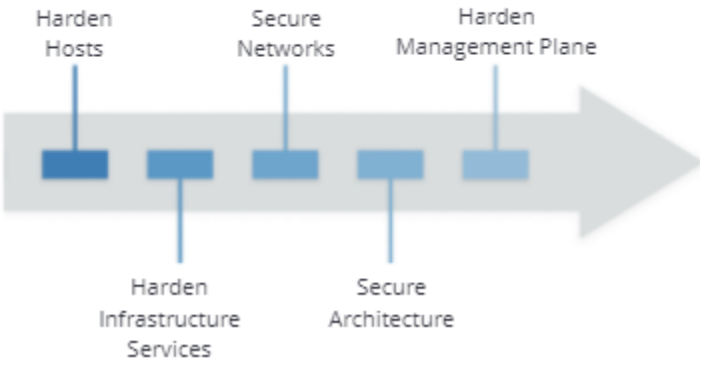
CLOUD INFRASTRUCTURE SECURITY OVERVIEW

- Reduced capital expenditures
- Better agility and resiliency
- Better economic benefits
- Better security benefits

SIMPLIFIED INFRASTRUCTURE COMPONENTS



SECURING CLOUD INFRASTRUCTURE



1. Infrastructure security includes all of the underlying physical resources and the software, like operating systems, that runs on them.
2. With private cloud you are responsible for securing all of the hardware and software that makes up the cloud platform. With public cloud you are only responsible for what you deploy in the cloud.
3. Cloud platforms, especially private cloud, are often built using common components including operating systems, message queues, and databases. All of these need to be properly secured.
4. Security cloud infrastructure starts with proper design, then hardening of the base systems, the various services, and eventually the management plane.

▼ Software Defined Network

UNDERLYING IaaS NETWORKS

Management	• Management plane to nodes
Storage	• Storage nodes (volumes) to compute nodes (instances)
Service	• Internet to compute nodes • Instances to instance

BUILDING UNDERLYING NETWORKS FOR CLOUD

USE SEPARATE PHYSICAL NETWORKS.

- Don't rely on VLANs.
- SDN may be a good option, but depends on the version and the hardware you use... *physical separation is still preferred, as much for performance as security.*

ISOLATE THE CLOUD NETWORKS FROM THE LAN.

- There should be only 2 outside connections:
  - The network manager to route Internet traffic.
  - The management and web and API server.

VIRTUAL NETWORKS

VIRTUAL NETWORKS AND SECURITY

- Virtual networks are subject to the same security concerns of a physical network.
- Virtual networks always run on a physical network.

VIRTUAL NETWORKS MAY PROVIDE A SIMPLER STACK TO BUILD THE PRIVATE CLOUD.

- Greater control is afforded through SDN.

VIRTUAL NETWORKS MAY include inherent security capabilities.

VLAN

- Leverage existing technology available in essentially all networks
- Designed for network segregation, not isolation, in single-tenant environments
- Not effective as a security barrier
- Have performance and address space limitations at cloud scale

SDN

- Software Defined Networks decouple the network control plane from the underlying hardware
- Abstracts virtual networking from traditional LAN limitations
- Extremely flexible (e.g. overlapping IP address ranges on same physical hardware)
- Multiple implementations, both standard and proprietary
- Can create effective security barriers

SOFTWARE-DEFINED NETWORKING

- Provides a decoupled control plane that is (potentially) easier to secure.
- OpenFlow is an example of a SDN.
- Remote access is controlled by the Administrator.
- Different flavors support different capabilities, but generally they can couple tightly with the cloud platform and possibly security tools.
- Nearly all implementations are API-enabled.
- While they may look like a regular network to the cloud consumer, they function VERY differently.
  - Rely heavily on *packet encapsulation*.

SDN SECURITY BENEFITS

EASIER ISOLATION

TOPOLOGY NOT LIMITED TO PHYSICAL STRUCTURE

- E.g., you can put multiple overlapping virtual networks, even with the same address ranges, on the same physical network

SDN FIREWALLS / SECURITY GROUPS

- Default Deny
- Orchestrated
- Granularity of host firewall with manageability of a network appliance

SECURITY POLICIES AND CONTROLS BASED ON TAGS AND OTHER CONTEXT

SDN FIREWALLS / SECURITY GROUPS

POLICY-BASED

- Not necessarily tied to IP addresses
- Can include context/tagging and other intelligence

NO ADDITIONAL HARDWARE OR SOFTWARE TO DEPLOY

TYPICALLY DEFAULT-DENY

- Even assets in the same security group can't communicate

APPLY ON A PER-ASSET LEVEL (INSTANCE OR PaaS OBJECT)

- But managed outside that asset. For example, if a virtual machine is compromised that can't be used to disable the firewall

INTEGRATED INTO CORE SDN LOGIC

- Traffic/packets simply dropped if they don't match the policy's rules
- Tightly coupled with the cloud orchestration so fully capable of keeping up with high velocity changes

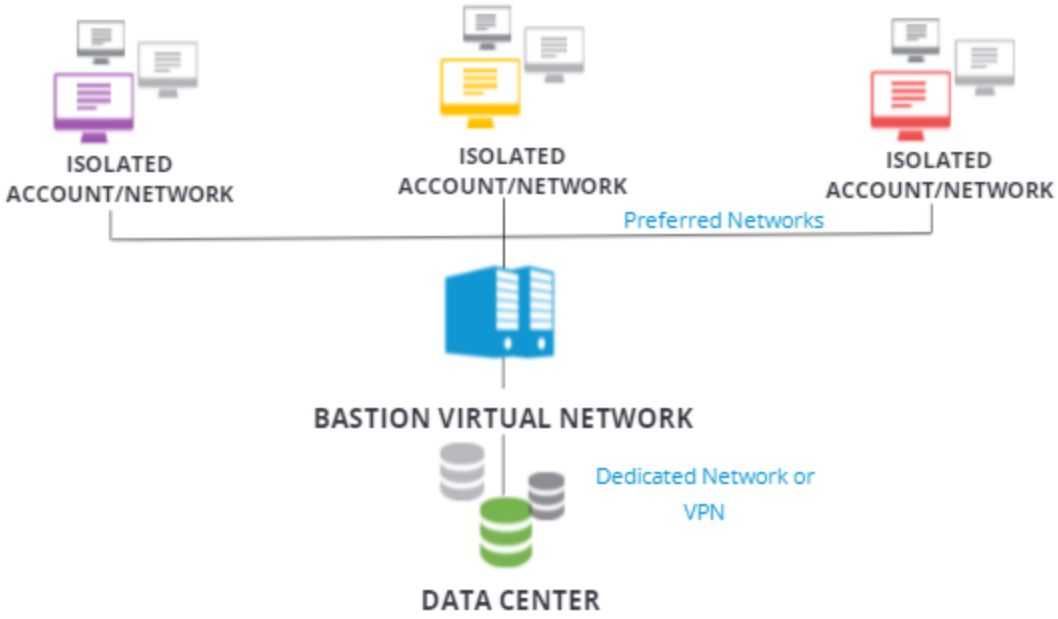
1. Cloud platforms typically rely on 3 physical networks (at a minimum). One for management, one for storage, and one for traffic between resources.
2. The two most common virtual networking technologies used in cloud are VLANs and Software Defined Networks. For security, SDNs are preferred since they provide better isolation and security.
3. SDNs decouple the control plane from the underlying physical network and provide tremendous flexibility. They are capable, for example, of deploying the same IP address range across isolated networks on the same physical hardware.
4. Security Groups is the common name for the firewalling built into SDNs. They can provide the manageability of a network firewall with the granularity of a host firewall.

▼ Cloud Network Security

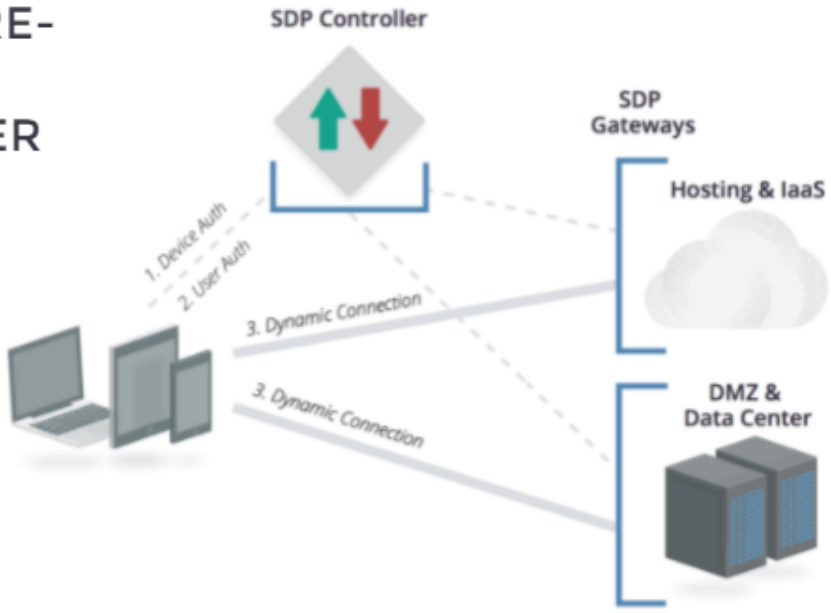
3RD-PARTY SECURITY TOOLS ADVANTAGES & DISADVANTAGES

PHYSICAL APPLIANCE	VIRTUAL APPLIANCE	HOST SOFTWARE
• Unusable in public cloud • Lack visibility into virtual networks • If used, require highly inefficient routing and topologies • Typically incapable of keeping up with cloud rates of change	• Become bottlenecks • May materially increase cloud costs due to resource requirements • Must support autoscaling, elastic pricing, and other cloud native orchestration • Must support high rates of change (e.G., IP address velocity)	• Typically better suited to cloud topologies and rates of change • Must be consistently embedded in images/virtual machines • Will add to local resource requirements • Agents should be designed for cloud, lightweight, and cloud-aware • Must communicate and save data externally

BASTION NETWORKS / ACCOUNTS FOR HYBRID



SOFTWARE-DEFINED PERIMETER



PROVIDER & CONSUMER RESPONSIBILITIES

CONSUMER

- Proper virtual network design
- Implementing virtual security controls (e.g., security groups)
- Securing their portion of the management plane/metastructure (e.g., proper IAM)

PROVIDER

- Security of the virtualization technology
- Exposing security controls (e.g., security groups)
- Disabling attack surface (e.g., packet sniffing)
- Securing the virtual management infrastructure

1. Prefer SDN when available.
2. Use SDN capabilities for multiple virtual networks and multiple cloud accounts/segments to increase network isolation.
3. Separate accounts and virtual networks dramatically limit blast radius compared to traditional data centers.
4. Implement default deny with cloud firewalls.
5. Apply cloud firewalls on a per-workload basis as opposed to a per-network basis.
6. Always restrict traffic between workloads in the same virtual subnet using a cloud firewall (security group) policy whenever possible.
7. Minimize dependency on virtual appliances that restrict elasticity or cause performance bottlenecks

▼ Securing Compute Workloads

IMPACT ON TRADITIONAL WORKLOAD SECURITY CONTROLS

CCSK

CONTROLS

- May not be able to run agents (E.g., AV)
- "Traditional" agents may not work properly in cloud or will impede performance
- Agents must be cloud aware
  - E.g., not rely on static IP addresses and capable of communicating across virtual network boundaries
- Agents should be lightweight and support autoscaling and autoregistration
- Agents should not increase attack surface
  - E.g., require ports to be open for management

MONITORING

- Network addresses are not sufficient to identify a workload in the cloud
- Logs should be offloaded quickly due to more-ephemeral nature of cloud workloads
- Logging architectures should be redesigned to account for cloud topology and variable costs of different storage tiers
  - Cascading log collection is generally preferred. Collect locally in object storage and use filtering tools to migrate security-sensitive logs to central collection and management

ASSESSMENT

- Providers often limit vulnerability assessment
- Default deny networks may further limit network assessment effectiveness
- Host assessment (agents) is often preferable
- Assess images rather than instances when using immutable

IMMUTABLE WORKLOADS ENABLE SECURITY

STANDARD / LONG RUNNING

- Managed just like traditional servers

AUTOMATED CONFIGURATION MANAGEMENT

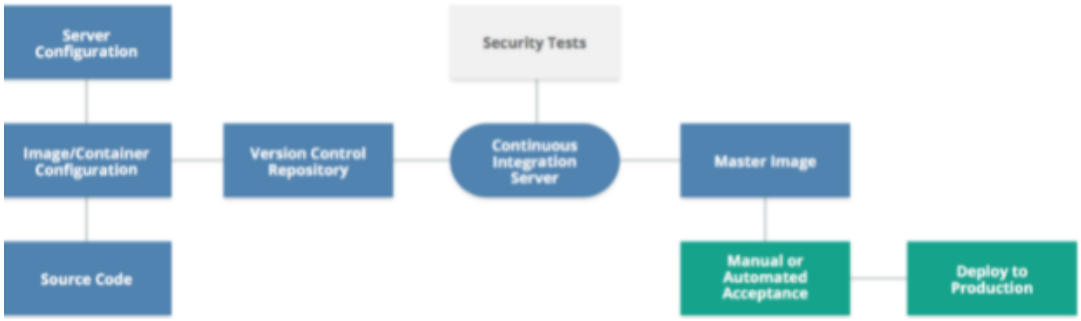
- The virtual machine is automatically configured using a template/policy based tool (e.g. Chef / Puppet / Ansible / Salt)
- It changes, but manual changes disabled since the automation would overwrite

IMMUTABLE

- Based on images and automatically deployed (e.g. by an auto scale group)
- Login disabled since changes won't propagate to other instances
- You replace with new versions instead of patching/updating old versions
- Very easy to harden for security (e.g. disable SSH)

RELATIVE SECURITY

CREATING IMMUTABLE IMAGES WITH DEPLOYMENT PIPELINES



PROVIDER & CONSUMER RESPONSIBILITIES

CONSUMER

- Security settings
- Monitoring and logging
- Image asset management
- Use dedicated hosting if needed and available
- All in-workload security controls (e.g., patching virtual machines)

PROVIDER

- Workload isolation
- Underlying infrastructure security
- Securing the virtualization technology
- Providing consumers adequate security controls
- Protecting volatile memory

COMPUTE SECURITY RECOMMENDATIONS

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.

1. Disable remote access.
2. Integrate security testing into image creation.
3. Alarm with file integrity monitoring.
4. Patch by updating images, not patching running instances.
5. Apply cloud firewalls on a per-workload basis as opposed to a per-network basis.
6. Choose security agents that are cloud-aware and minimize performance impact, if needed.
7. Maintain security controls for long-running workloads, but use tools that are cloud aware.
8. Store logs external to workloads.
9. Understand and comply with cloud provider limitations on vulnerability assessments and penetration testing.

▼ Management Plane Security

THE MANAGEMENT PLANE

KEY FUNCTIONS

- Provisioning resources
- Starting/stopping/terminating
- Configuring resources

SECURITY CONSIDERATIONS

- Authentication
- Access Control
- Logging/Monitoring

THE MANAGEMENT PLANE IS THE LITERAL KEY TO YOUR PRIVATE CLOUD. PROTECT IT WISELY.

ACCESS METHODS

WEB

API
(Usually REST)

MANAGEMENT PLANE ACCESS & CREDENTIALS

DIFFERENT PROVIDERS / PLATFORMS USE DIFFERENT AUTHENTICATION OPTIONS

- We cover some of these in more depth in the Identity Management section

WEB CONSOLE LOGINS ARE TYPICALLY LIKE LOGGING INTO ANY OTHER WEB SERVICE

- Username and password
- Maybe MFA

APIs USE MULTIPLE TECHNIQUES THAT MAY OR MAY NOT HAVE CEDENTIALS DIFFERENT FROM THE WEB CONSOLE

- HTTP request signing (crypto using keys)
- Tokens
- Oauth/SAML

ALL CONNECTIONS SHOULD ALWAYS USE TLS

MANAGEMENT PLANE SECURITY

- Secure root account
- Manage Non-Root Users
- Enable Monitoring / Auditing

ROOT ACCOUNT SECURITY

ENABLE HARDWARE MULTI-FACTOR AUTHENTICATION (MFA)

- Store in a locked, central location

USE ISOLATED CREDENTIALS (A DESIGNATED EMAIL OR USER ACCOUNT NOT USED FOR ANYTHING ELSE)

- Use a name with a random seed if possible to reduce phishing

IF AVAILABLE, USE ACCOUNT SECURITY QUESTIONS

- Record and store securely

NEVER USE ACCOUNT EXCEPT FOR EMERGENCIES

CLOUD IAM MANAGEMENT

- Role-Based Access Control (RBAC)
- Variable granularity across providers/platforms
- Variable granularity within product lines
- Look for ability to integrate w/SSO or directory services
- Investigate third-party tools

MONITORING / AUDITING

CLOUD SIDE

- Logs all API and internal activity
- Best option when available
- Pull logs to secure, central location

PORTAL / PROXY

- Route users through a portal, they don't have direct credentials
- Misses internal activity or compromised creds
- May be the only option
- CASB tools often used for SaaS (we discuss later) Host / Network Logs

1. The management plane is how you manage your cloud deployments. It's the biggest difference from traditional infrastructure security, and the most critical piece to protect.
2. Nearly all clouds support both web console and API access to the management plane. When running your own cloud it's critical to make sure these are effectively locked down.
3. Management planes support different kinds of credentials, all of which must be managed securely.
4. Always start by securing the root or master account since losing control of that means losing complete control over your cloud deployment,
5. Enforce least privilege when setting up your other privileged users and administrators.
6. Always use multifactor authentication for all cloud accounts, especially privileged users.

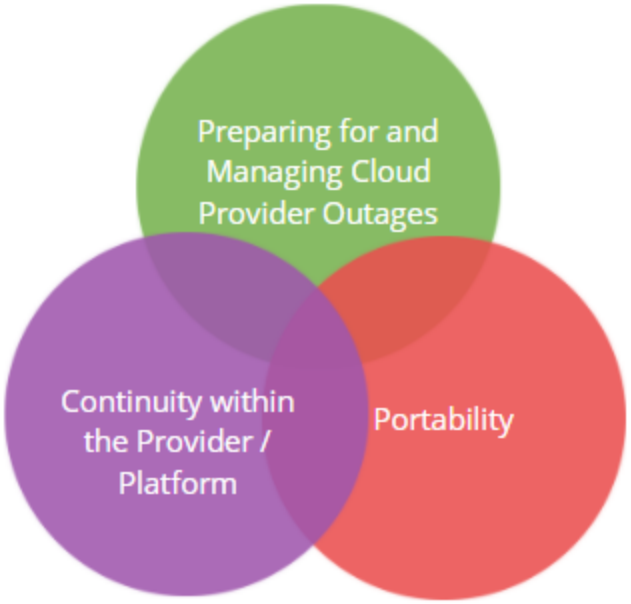
▼ BCDR

RULE #1

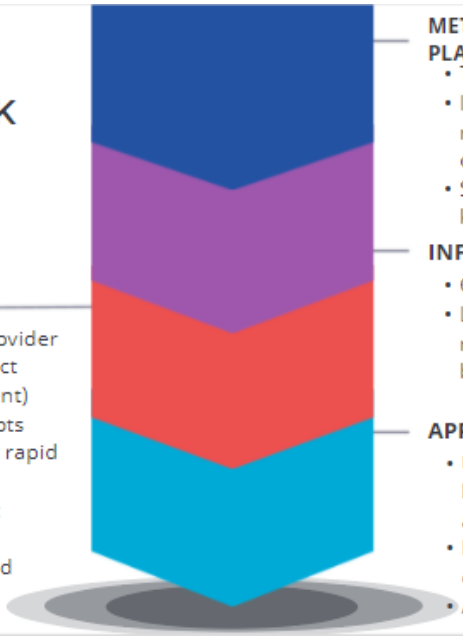
ARCHITECT FOR FAILURE!

Design for your platform(s) and don't expect existing architectures to lift and shift without compromise

KEY ASPECTS OF BC/DR



COVER THE ENTIRE STACK



METASTRUCTURE / MANAGEMENT PLANE

- The cloud configuration
- IAM, monitoring, and other in-cloud management controls and compliance artifacts
- Software Defined Infrastructure is a key tool

INFRASTRUCTURE

- Core configuration
- Leverage platform/provider resiliency capabilities rather than building from scratch inside VMs

APPLISTRUCTURE

- Understand PaaS limitations and lock-in, including the historical availability of services
- Downtime is nearly always an option – have realistic standards
- Adopt Chaos Engineering

INFOSTRUCTURE

- Leverage "resilient" provider storage (e.g. most object storage is highly resilient)
- Keep backups/snapshots within the provider for rapid restore
- Always use lowest cost storage and transfer mechanisms within and between providers

BC/DR IN THE CLOUD

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.

BC/DR IN THE CLOUD

PREPARE FOR GRACEFUL FAILURE IN CASE OF A CLOUD PROVIDER OUTAGE.

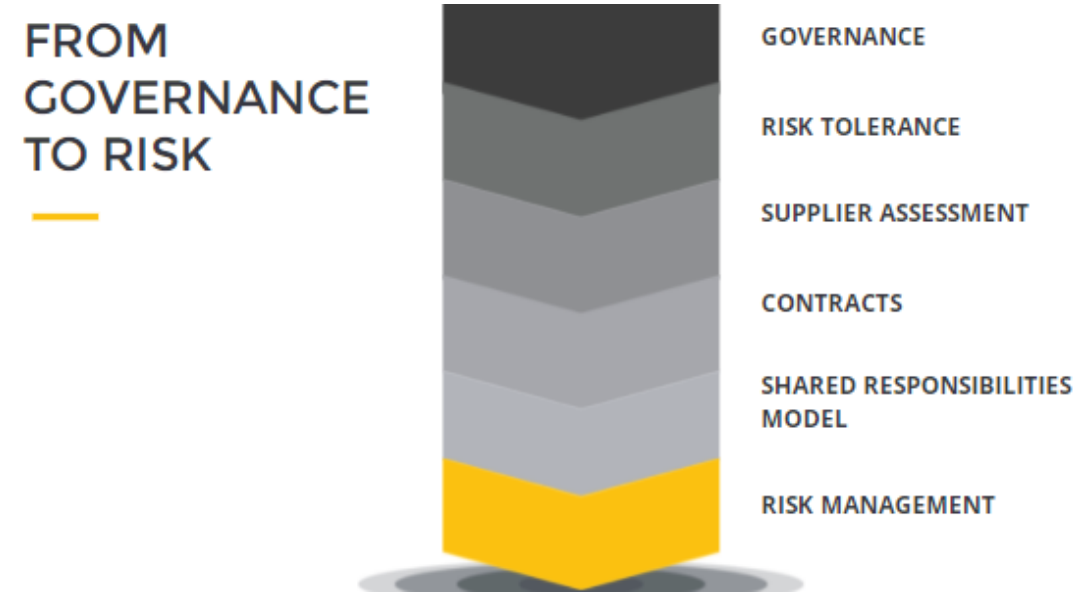
This can include plans for interoperability and portability with other cloud providers or a different region with your current provider.

- 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.

1. The first rule of cloud is to architect for failure. Since any individual virtual resource may be less resilient, cloud provides and platforms have built in tools to improve systemic resilience. But fail to use these and you are more likely to experience an outage.
2. The two major areas to focus on are resiliency within your cloud provider, then resiliency if your provider goes down. Portability can play a role here but don't get so hung up on it that you become paralized and can't use all of the capabilities of your platform or provider.
3. Your BC/DR should cover the entire stack of the logical model- from the metastructure / management plane and infrastructure to your data and application architecture.

▼ Module 3: Managing Cloud Security & Risk

▼ Governance



▼ Managing Cloud Security Risk

RISK MANAGEMENT

ENTERPRISE RISK MANAGEMENT

- Rooted in providing value to stakeholders.
- How to measure, manage, and mitigate uncertainty.

INFORMATION RISK MANAGEMENT

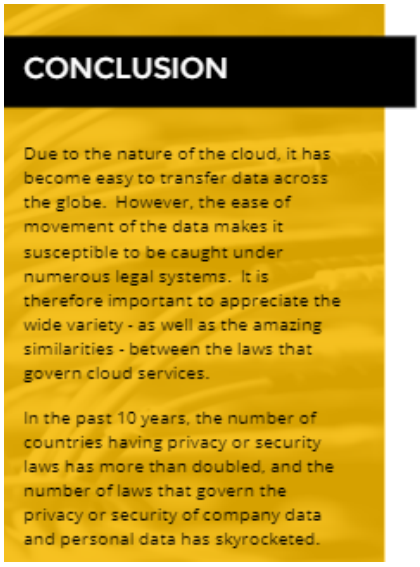
- Aligning risk management to the tolerance of the data owner.
- Primary means of decision support for IT/security on the CIA of information assets.

1. Enterprise risk management includes all-risk management for the entire organization.
2. Information risk management focuses on the risk to information, and must still align with the risk tolerance of the data owner.
3. The effort in a risk assessment should align with the value of the data. Just because something is moving to the cloud doesn't mean you now need to treat it as being higher-value.
4. In terms of risk, like security, IaaS is most closely aligned to traditional infrastructure, while with SaaS there is a greater reliance on the cloud provider.
5. Risks in private cloud may be similar to that of public cloud if the private cloud is hosted and/or managed by a third party.

▼ Compliance

1. Compliance is a tool to ensure organizations are meeting corporate obligations.
2. Audits are how we validate compliance, and they can be performed internally or externally using third parties.
3. Cloud changes compliance because it now becomes a shared responsibility between the cloud consumer and the provider.
4. Compliance inheritance is the principle that if a cloud provider's service is compliant with a regulation/standard, then cloud consumers can build compliant services/applications using that service. But it does not guarantee compliance since the cloud consumer can still build a non-compliant application on top of a compliant service.

▼ Legal Considerations for Cloud



GLOBAL TRENDS:

Protection of privacy and allowing individuals to have some control over the collection and use of their personal data.

There is a concern for the security of personal data and company data. A significant number of laws require the adoption of formal security policies

Countries and states are recognizing that security breach occurs, for a variety of reasons - state actors, hackers, disgruntled employees, negligence or inadvertent error. These breaches should be notified to be affected parties. Numerous new laws require prompt disclosures to individuals and government agencies.

There is a concern that data laws many not be equivalent from state to state and countries are establishing barriers to prevent the transfer of data to those that do not offer "adequate protection".

Finally like for any other relationship, things are better recorded in writing. Contracts are important. Cloud contract can be tricky because too easy to sign when they are just posted on a website for the customer to click on "I agree". Make sure you read them carefully to understand the terms.

▼ Audit

1. Different types of audits and assessments have different focuses, and even when the same name is used can have different focus and scope across cloud providers.
2. Cloud providers often limit the kinds of assessments their customers can use since some of these, like vulnerability assessments, can't be distinguished from real attacks without being constrained.
3. Ensure you know the scope, results, and timing (dates) of previous audits. Not all audits on a provider's website are necessarily up to date or cover the service under consideration.
4. Cloud consumers are responsible for maintaining their own artifacts of compliance for their own audits, such as log files.

▼ CSA Tools

1. The Cloud Controls Matrix is a list of cloud security controls mapped by domain and aligned to various regulatory frameworks.
2. The CCM is an excellent tool for evaluating your cloud security controls and is useful to both cloud providers and consumers.
3. The Consensus Assessment Initiative Questionnaire is a standard set of security questions for cloud providers. It allows cloud consumers to directly compare providers, and allows providers to reduce the need to respond to non-standard RFPs.
4. The Cloud Security Alliance Guidance (which this training is based on) tells you how to implement your controls, while the CCM tells you which controls to implement.
5. The Star Registry and StarWatch tool serve as central repositories for cloud provider security documentation, including the CAIQ.

▼ Module 4: Data Security for Cloud

▼ Cloud Data Storage

1. Data stored in the cloud may use a range of different abstraction and virtualization technologies. To the user these may look like traditional storage, but behind the scenes the mechanisms will be quite different. However, all data is still eventually stored on physical media.
2. Volume storage is virtual hard drives, while object storage is like a database for files that is managed via APIs. Databases in the cloud may be multitenant and don't necessarily work like on-premise database systems. Cloud applications may store files using a wide range of techniques that the cloud consumer has no insight into.
3. Most cloud storage uses data dispersion for resilience, which breaks our traditional ties to knowing the physical location of drives.
4. Tools like CASB, DLP, and even URL filtering can help us visualize or manage data migrating to the cloud.
5. Data can be encrypted before migrating to the cloud, protected in transit with TLS or other network encryption, and may be consolidated and encrypted through an on-premise proxy, especially for cloud-backed backups.

▼ Securing Data in the Cloud

1. Access controls are the most fundamental security control, even in cloud computing.
2. There is massive variability of available access controls between cloud providers, and cloud storage may offer new categories of controls, such as sharing, beyond those in more-traditional storage.
3. An entitlement matrix is the documentation of authorizations. It defines who should be allowed to not only access data, but what they should be allowed to do with it.

▼ Encryption for IaaS

1. There are multiple layers where you can encrypt, each with benefits and complications. Encrypting higher in the application stack is often best for discreet data, while lower-level encryption, like volume, is better for bulk data.
2. Encryption systems are composed of the data, the encryption engine, and the key management, Where you place these determines the architecture and affects the security of the system.
3. Whenever possible, you want to separate the encryption key from the data and the encryption engine.
4. For object storage encryption, you can encrypt the data on the client site, the server side (using multiple techniques), or even through storage proxies (which we frequently see used for site backups).

▼ Encryption for PaaS & SaaS

1. Platform as a Service encryption will depend almost completely on the kind of platform and options supported by your provider. For workloads though, you can nearly always program your own encryption at the application layer.
2. When encrypting in your application, you can handle the encryption in your own code or send it off to an external encryption server or service.
3. For Software as a Service you only have two options – rely on your provider's supported encryption, or use a third-party encryption proxy that sits as a man in the middle.
4. SaaS encryption proxies may introduce new security concerns due to requiring you to break any network encryption to the cloud provider. They may also break application functionality. However, there are still valid use cases, albeit limited.

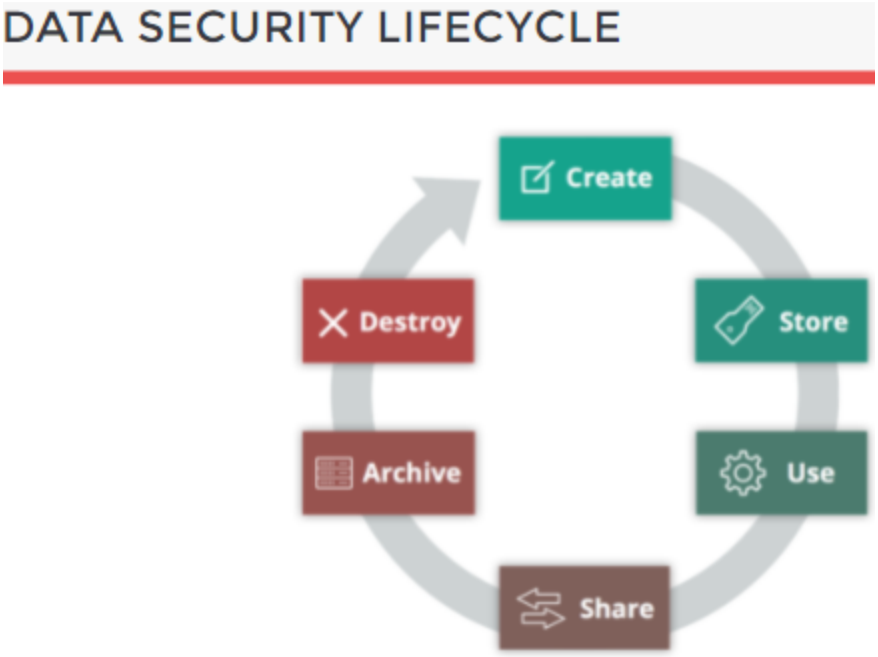
▼ Encryption Key Management

1. Proper key management is essential to effective encryption, and we have more options at our disposal in cloud computing.
2. HSMs and physical appliances may be offered by your cloud provider, or you can look at deploying software or virtual appliances in the cloud, connecting to existing hardware over a hybrid connection, or even leverage new options like a key management service from your cloud provider or a third party.
3. Providers offer a range of key management options, from the provider completely managing the keys, to allowing you to manage your own keys in their environment or even provide keys as needed.
4. Bring Your Own Key will work differently on different providers and services, with varying levels of relative security.
5. The final choice will come down to your risk and threat models. Once you know the risk you are trying to prevent, you can evaluate the technical options in your provider and platform of choice. Remember, not all data needs the same level of security so you don't always need to default to the most secure option.

▼ Other Data Security Options

1. Integrating PaaS and other new cloud architectural options into applications and data storage may allow cloud consumers to shift more security burden onto cloud providers and reduce the stack's attack surface.
2. Good activity monitoring and alerting are important to cloud data security, and providers may also support a variety of additional security controls.
3. Data Loss Prevention tends to be more useful for SaaS and may be integrated into CASB tools.
4. Traditional DRM/ERM isn't necessarily useful for cloud, but some SaaS/PaaS services may have "DRM-like" capabilities such as sharing or view controls that provide similar protections.
5. Data masking is critical for test data generation and to ensure production data is not exposed in development environments.

▼ Data Security Lifecycle



▼ Module 5: Securing Cloud Applications, Users, and Related Technologies

▼ SSDLC

1. The secure software development lifecycle (SSDLC) is a structured process for ensuring security needs are met throughout application development processes.
2. There are multiple frameworks, such as those from Microsoft and OWASP.
3. Cloud will impact each phase of the lifecycle, from training all the way into operations. Changes in visibility and more reliance on the shared responsibilities model are constant threads.
4. When modeling application threats for cloud deployments, some risk will be greater and some less. Threat modeling is a great way to evaluate these differences.

▼ Testing and Assessment

1. Secure software development involves a range of security testing. All of these are impacted by cloud computing.
2. With static analysis you should place a greater emphasis on looking for stored cloud credentials, as well as the proper configuration of API usage.
3. Dynamic analysis and vulnerability assessment may require permission from your cloud provider.
4. New vulnerability analysis options, such as scanning in a deployment pipeline or using host-based agents, are often better used for cloud.
5. Assessing the configuration of the cloud environment should now be within scope for an application security assessment.

▼ DevOps

1. There are many definitions of DevOps, but a key defining characteristic is the use of continuous integration and/or continuous delivery (CI/CD).
2. Continuous integration pipelines support consistency and integrated security testing.
3. DevOps also supports immutable deployments, Where instead of updating things or making manual changes we replaced them from a known good definition.
4. This supports security through consistency and allowing us to even remove the need to log into production assets.
5. There is a lot more to DevOps, but integrated security testing, consistency due to use of CI/CD, and immutable are some of the key security benefits.

▼ Secure Operations and Architecture

1. Secure operations is all about keeping your application secure once it is deployed into production.
2. When using cloud, the management plane is now a concern and the cloud configuration is now within scope for change management.
3. WAF will need to be adjusted to account for the different deployment options, like autoscaling, used in cloud.
4. New cloud architectural options, such as serverless and micro services may offer security benefits, and are increasingly common.
5. Serverless puts more responsibility onto the cloud provider, leveraging the shared responsibilities model to reduce the customer's attack surface and scope of security operations.

▼ IAM Definitions

INTRO TO CSA IDENTITY TERMS

ENTITY	• Discrete types that will have *Identity*; these are to Users, Devices, Code, Organizations and Agents
IDENTITY	• The unique expression of an entity within a given namespace.
IDENTIFIER	• The means by which an *Identity* can asserted, usually using crypto tokens for digital identities
ATTRIBUTES	• Facets of an *identity* (e.g., org. unit or IP address)
PERSONA	• Expression of an *identity* with attributes that indicates context. E.g., a developer logged into a given project

INTRO TO CSA IDENTITY TERMS (Cont'd)

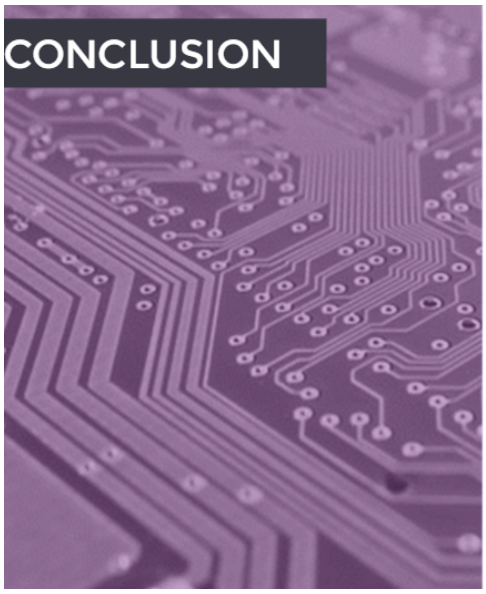
ROLE	• Has multiple meanings. Typically used to indicate a persona or subset. E.g., "developer" vs. "admin"
AUTHENTICATION	• The process of confirming an identity. *Authn*
MULTIFACTOR AUTHENTICATION	• Use of multiple factors in authentication (e.g., username + password + *token*)
ACCESS CONTROL	• Restricting access to a resource, Access management is the corresponding process
AUTHORITATIVE SOURCE	• The "root" source for an identity, such as a directory server

1. It is important to understand the foundational terminology for identity and access management (IAM)
2. At the heart is an entity, which is a person, device, or other "thing" that will be given access.
3. An identity is the expression of that entity within a namespace, such as an email address or username for a given system.
4. Entities prove their identity by providing identifiers during authentication.
5. After being authenticated, users may be granted access to objects or actions. This is called an authorization, and an entitlement is a specific approval.
6. Federated identity is critical for cloud computing because it allows us to manage identities across different systems.

▼ IAM Standards

IAM STANDARDS FOR CLOUD

MOST COMMON			LESS COMMON	
SAML	OAUTH	OPENID	XACML	SCIM



1. IAM for cloud computing relies on federated identity due to the requirement to manage authentication and authorization between the cloud consumer and the cloud provider.
2. The most widely supported and used federation standard for connecting enterprises with their cloud providers SAML.
3. Oauth and OpenID are web-centric federation standards often used by both consumers and organizations.
4. Federation involves multi-step cryptographically supported processes to connect an Identity Provider (the source for the identity) and the Relying Party (most often the cloud provider, where authorizations occur).
5. Typically the identity provider handles authentication, then the relying party handles authorization (enforcing what someone can actually do).

▼ IAM in Practice

1. Due to the complexity of managing multiple directory servers and cloud providers, a hub and spoke model using a federated identity broker is often preferred.
2. Because of the broad network access supported by cloud providers, multi factor authentication is critical to help reduce the chances of account takeovers.
3. Many cloud providers are now supporting attribute based access controls, which support greater granularity in entitlements than traditional role-based access controls.
4. Building an entitlement matrix can help document authorizations for your cloud providers different services, and help with assessments and audits.

▼ Module 6: Cloud Security Ops

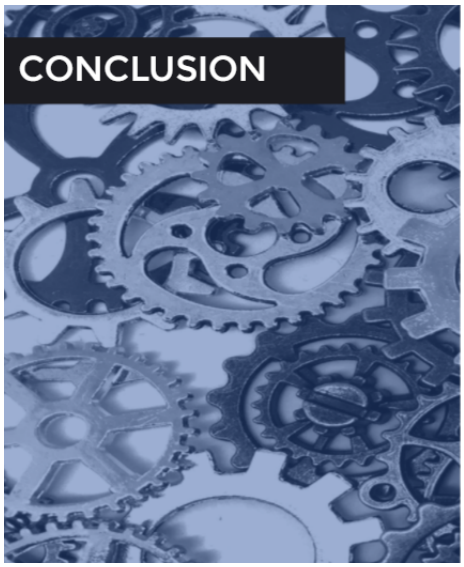
▼ Selecting a Cloud Provider

1. The critical security capabilities for cloud providers are a list of features required in a cloud platform to fully enable customers to build a comprehensive cloud security program.
2. When evaluating cloud providers, cloud consumers should also look at all available documentation, and pay particular attention to internal security controls that ensure a strong baseline level of security over time.
3. Individual security features are not as indicative as strong programmatic controls.
4. Cloud providers should also offer a wide array of reviewable third-party audits and assessments to validate their security program and control.
5. The Cloud Security Alliance provides the CAIQ, CCM, STAR, and STARWatch to help both Cloudproviders and consumers in communicating security posture.

▼ IR

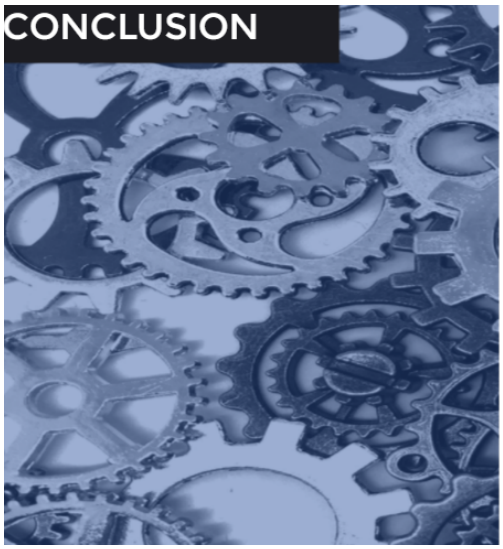
1. The fundamental nature of cloud changes the likelihood and nature of incidents. It is important to adjust your incident response process to account for these.
2. Cloud consumers and cloud providers will have different priorities in an incident. These may conflict when a provider needs to contain a consumer.
3. Focus on preparation, especially communications with cloud providers, adjusting IR plans, and building tool or "jump" kits to more-rapidly respond.
4. When available, infrastructure as code can allow isolation of a compromised environment while rebuilding a functional environment in parallel to reduce downtime. But don't forget, this will carry over any active vulnerabilities and configuration errors in the templates.

▼ SECaaS Fundamentals



1. Security as a Service is defined as a security product or service, with cloud-based management. These services can secure systems and data in the cloud, in traditional on-premises networks, or hybridized environments.
2. Security as a Service includes security products delivered as a cloud service, that also meet the NIST essential characteristics.
3. They offer the same benefits as the rest of cloud computing, and may also offer benefits such as deeper expertise among their staff, as well as intelligence sharing across all the customers they protect.
4. Drawbacks can include regulatory differences, reduced visibility, and the potential for your data leaking through the provider.
5. The cloud consumer can never outsource their security accountability.

▼ SECaaS Categories & Recommendations



1. Security as a Service offerings include a wide range of categories that span most, if not all, major security domains.
2. The key is that the service meets the NIST essential characteristics, and this also includes hybrid offerings (e.g. the management is in the cloud with some on-premise components.
3. Common categories include everything from security assessment, to cloud-based defensive tools like WAF , email, and web filtering, to SIEM and logging.
4. When selecting a provider ensure you understand your data handling and compliance requirements and evaluate services that are compatible with your existing technology requirements, such as architectures and operating systems.

▼ Related Technologies

1. Related technologies are key technologies often seen with, and used by, cloud deployments. They include Big Data, the Internet of Things, mobile computing, and serverless computing.
2. Big Data platforms tend to have low inherent security, so using the cloud for isolation is important. It's also critical to understand where and how data is stored and, often, to protect it with distributed encryption.
3. The Internet of Things often uses cloud computing for back end processing, application logic, and data storage. Security concerns tend to focus on device and user authentication and authorization, secure communications, and data storage.
4. Mobile issues are often very similar to those of IoT when it comes to cloud as the cloud becomes the back-end for many mobile apps.
5. Serverless is a cloud-native technology and used in most modern deployments to some degree. IAM and logging tend to be a security focus since they are so different compared to on-premise or even virtualized workloads.

▼ CCSK Exam Prep