Modernizing Technology Governance

Module – 2 Workflows

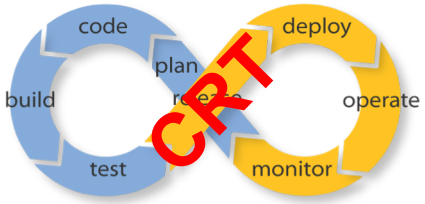
Tradition verses Cloud (Modernized) – Governance

Tradition – Governance

- Information and technology (IT) governance is a subset discipline of corporate governance, focused on information and technology (IT) and its performance and risk management.
- The interest in IT governance is due to the on-going need within organizations to focus value creation efforts on an organization's strategic objectives and to better manage the performance of those responsible for creating this value in the best interest of all stakeholders

Cloud – Governance

- Technology drives your governance alignment
- Governance is a “Shared Responsibility”
- Automation is the ***Key*** to successful governance
- Pre-Cloud decision making process are paramount (e.g. service selection, policies, frameworks architecture, data protections, etc.).
- Focus is on Continuous Risk Treatments (CRT)

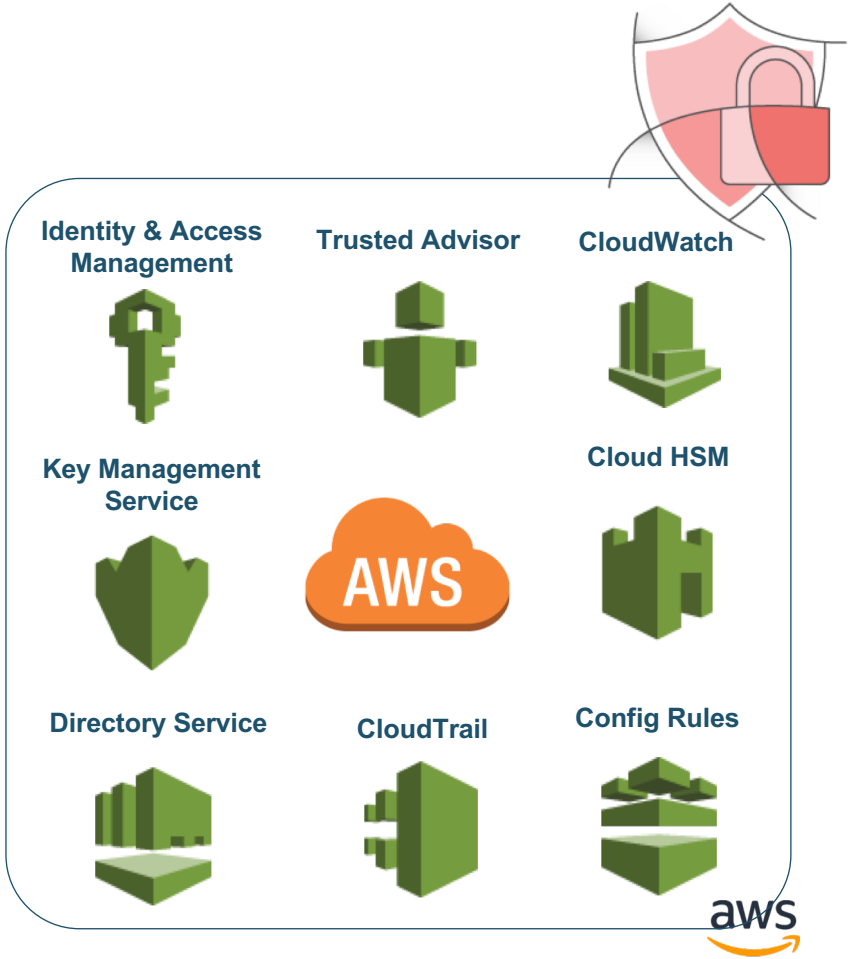


aws

Security by Design

Security by Design (SbD) is a security assurance approach that formalizes AWS account design, automates security controls, and streamlines auditing.

Instead of relying on auditing security retroactively, SbD provides security control built in throughout the AWS IT management process.



Security by Design - *Design Principles*

Developing new risk mitigation capabilities, which go beyond global security frameworks, by treating risks, eliminating manual processes, optimizing evidence and audit ratifications processes through rigid automation

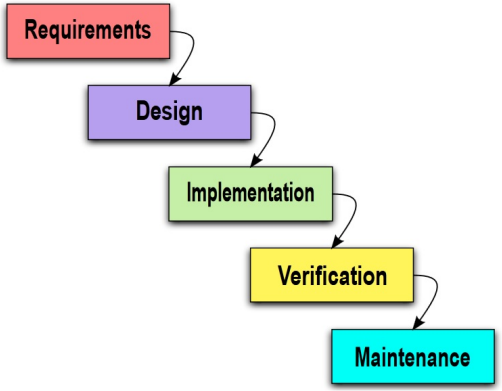
- Build security in every layer
- Design for failures
- Implement auto-healing
- Think parallel
- Plan for Breach

- Don't fear constraints
- Leverage different storage options
- Design for cost
- Treat Infrastructure as Code
    - Modular
    - Versioned
    - Constrained

aws

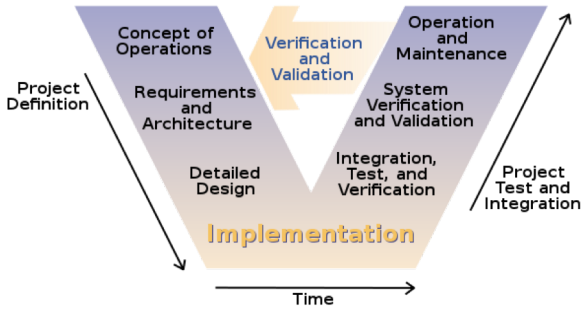
So why Security by Design…

Waterfall



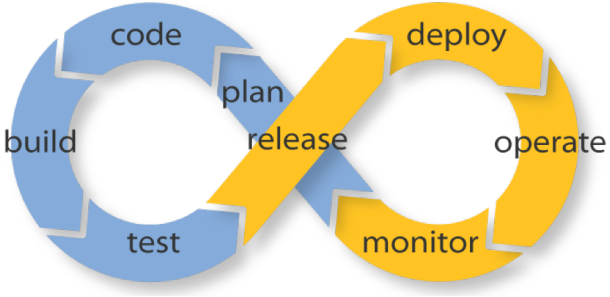
- Designed for fixed requirements
- Must finish the whole before any part is usable
- Linear steps w verification gate
- Manual processes
- Infrequent system changes

V-model



- Designed for fixed requirements
- Must finish the whole before any part is usable
- Waterfall + earlier test planning
- Manual processes + tools
- Periodic system changes

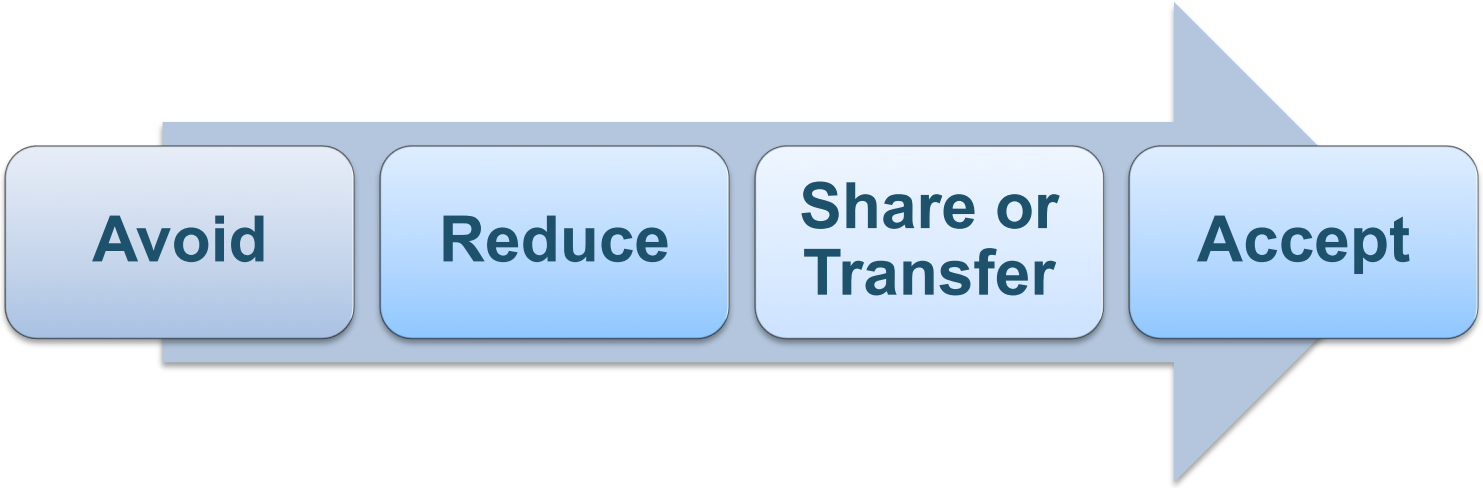
DevOps



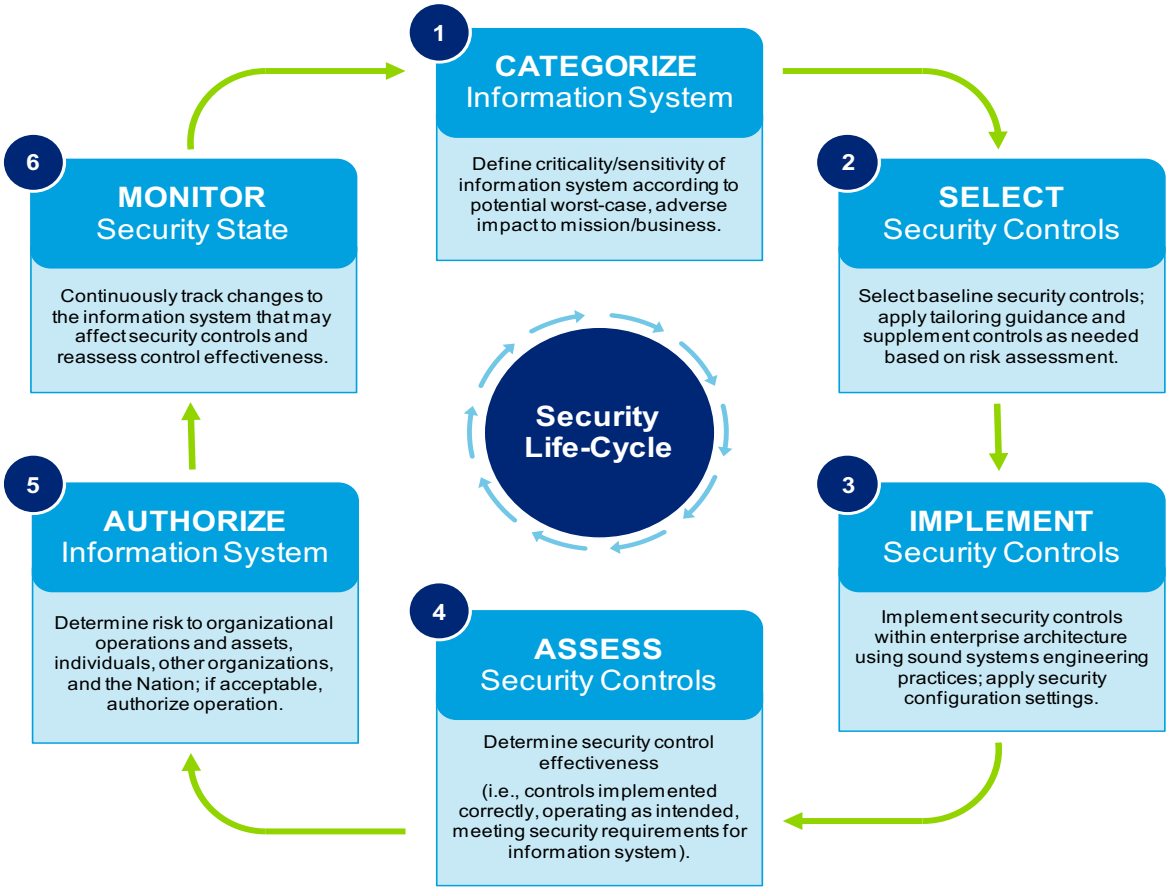
- Designed for changing requirements
- Deliver smaller parts with immediate functionality
- Iterative steps w testing in each
- Continuous deployment

aws

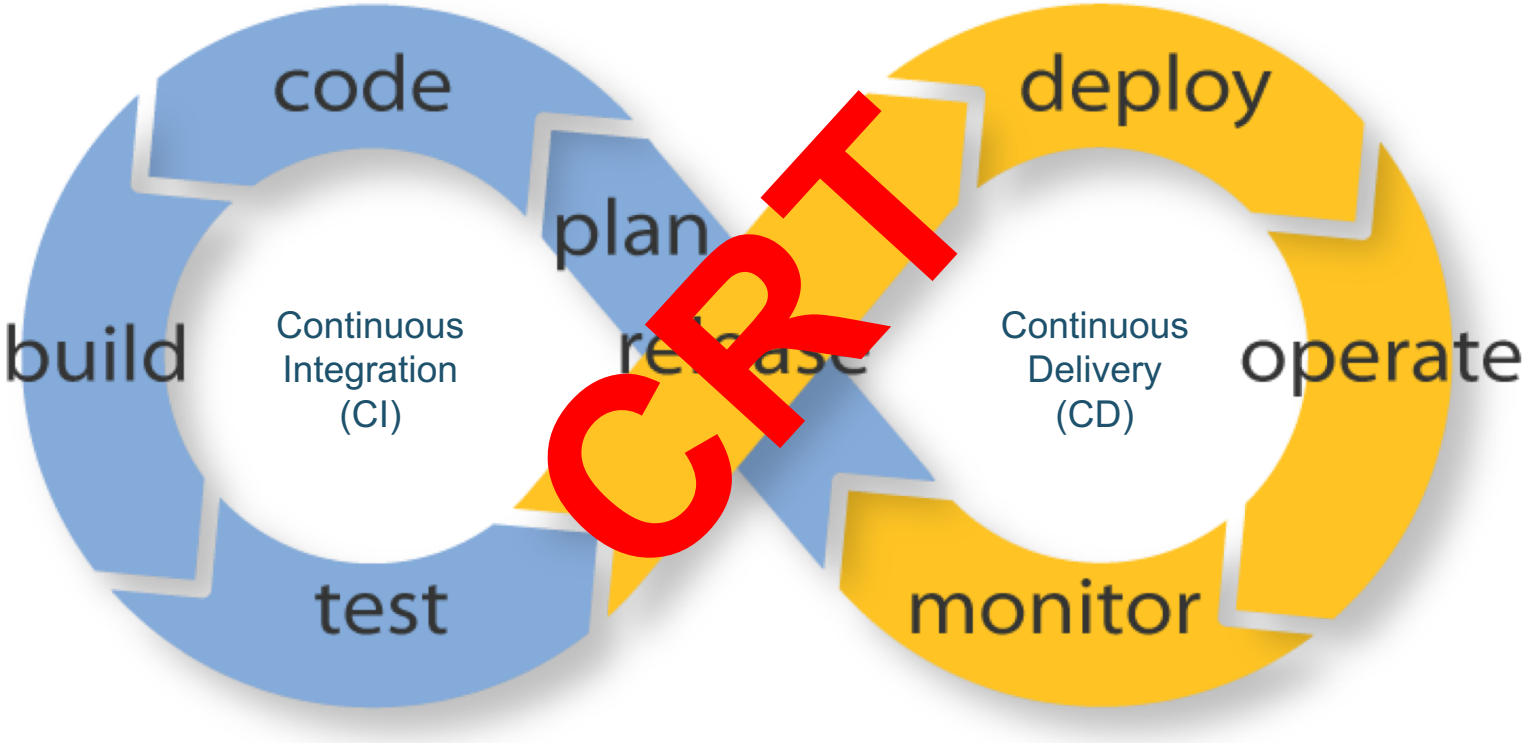
Traditional Risk Treatments



Traditional Security Lifecycle



DevOps


code
plan
build
Continuous
Integration
(CI)
test
deploy
Continuous
Delivery
(CD)
operate
monitor
CRT

Modernizing Technology Governance (MTG)

1 DECIDE WHAT TO DO (STRATEGY)	 1.1 IDENTIFY STAKEHOLDERS	 1.2 IDENTIFY YOUR WORKLOADS		
2 ANALYZE + DOCUMENT	 2.1 RATIONALIZE SECURITY REQUIREMENTS	 2.2 DEFINE DATA PROTECTIONS + CONTROLS	 2.3 DOCUMENT SECURITY ARCHITECTURE	
3 AUTOMATE, DEPLOY + MONITOR	 3.1 BUUILD/DEPLOY SECURITY ARCHITECTURE	 3.2 AUTOMATE SECURITY OPERATIONS	 3.3 CONTINUOUS MONITOR	 3.4 TESTING + GAME DAYS
4 CERTIFY	 4.1 AUDIT + CERTIFICATION			

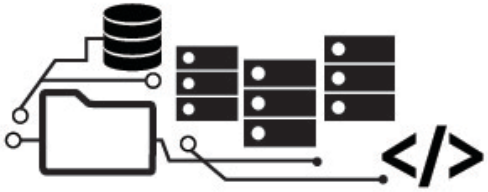
Step 1 - Decide what to do (Strategy)

1

DECIDE WHAT TO DO

(STRATEGY)

1.1
IDENTIFY STAKEHOLDERS



1.2
IDENTIFY YOUR WORKLOADS

1.1 Identify Business Units

Division

Sales, Business Development and Marketing

- No regulatory requirements
- Interconnections between HQ & Other divisions

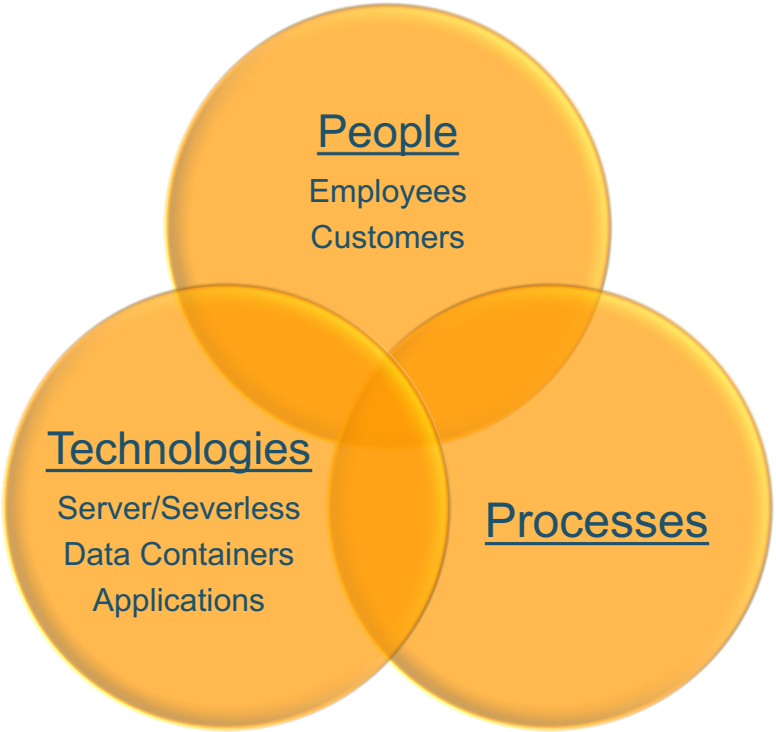
Division

Division

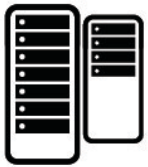
NIST 800-171
Compliance Criteria

aws

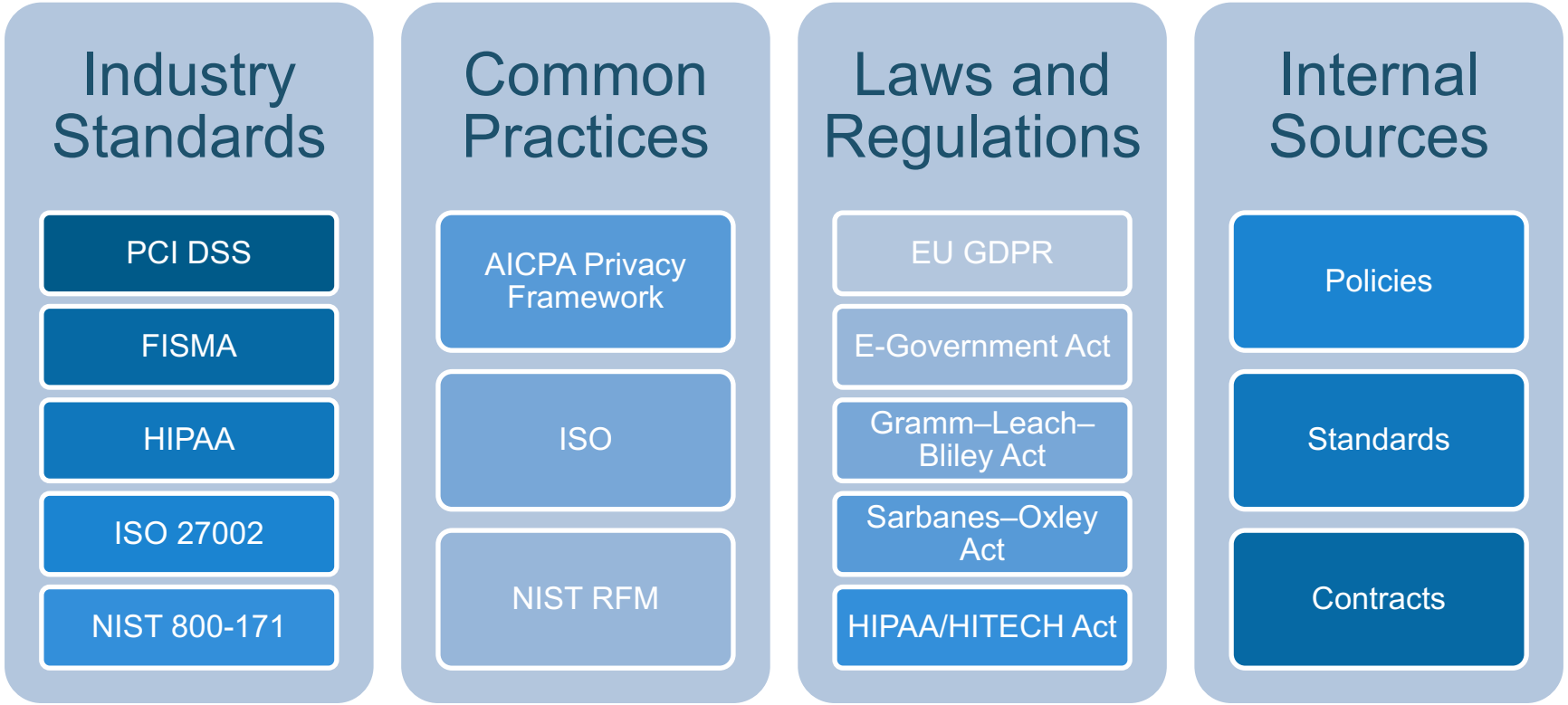
1.2 Identify your workloads



Phase 2 - Analyze, Define and Document

2 ANALYZE + DOCUMENT	 **2.1** RATIONALIZE SECURITY REQUIREMENTS	 **2.2** DEFINE DATA PRODUCTION + CONTROLS	 **2.3** DOCUMENT SECURITY ARCHITECTURE

Rationalizing the Security Requirements



aws

Shared Responsibility Control Types

Control Type	Description
Inherited Controls	Controls which a customer fully inherits from AWS (e.g. Data Center Controls).
Hybrid Controls	Controls for which AWS provides partial implementation of the control requirement, but require the customer to also take responsibility to fully implement the control requirement. (e.g. Access Controls and Resiliency).
Shared controls	Controls which apply to both the infrastructure layer and customer layers, but in completely separate contexts or perspectives. In a shared control, AWS provides the requirements for the infrastructure, and the customer must provide their own control implementation within their use of AWS services.
Customer Specific	Controls which are solely the responsibility of the customer, based on the application they are deploying within AWS services.

Inherited Security and Compliance

Control #	Control Name	Control #	Control Name	Control #	Control Name
A.11.1.1	Physical security perimeter	A.11.2.1	Equipment siting and protection	A.11.2.7	Equipment siting and protection
A.11.1.2	Physical entry controls	A.11.2.2	Supporting utilities	A.11.2.8	Supporting utilities
A.11.1.3	Securing offices, rooms and facilities	A.11.2.3	Cabling security	A.11.2.9	Cabling security
A.11.1.4	Protecting against external and environmental threats	A.11.2.4	Equipment maintenance	A.11.2.7	Equipment maintenance
A.11.1.5	Working in secure areas	A.11.2.5	Removal of assets	A.17.2.1	Availability of information processing facilities
A.11.1.6	Delivery and loading areas	A.11.2.6	Security of equipment and assets off-premises	A.13.1.2	Communications security

AWS Federal Inherited Security Controls

Control ID	Requirement Mapping						Customer Responsibility
	FedRAMP Moderate	FedRAMP High	FedRAMP (Mod) + DoD SRG IL4	FedRAMP (Mod) + DoD SRG IL5	FedRAMP (High) + DoD SRG IL5	DOD SRG VOP	
MA-5	M	H	IL4	IL5	IL5		N/A
MA-5 (1)	M	H	IL4	IL5	IL5		N/A
MA-6	M	H	IL4	IL5	IL5		N/A
MP-1	M	H	IL4	IL5	IL5		N/A
MP-2	M	H	IL4	IL5	IL5		N/A
MP-3	M	H	IL4	IL5	IL5		N/A
MP-4	M	H	IL4	IL5	IL5		N/A
MP-5	M	H	IL4	IL5	IL5		N/A
MP-5 (4)	M	H	IL4	IL5	IL5		N/A
MP-6	M	H	IL4	IL5	IL5		N/A
MP-6 (1)		H					N/A
MP-6 (2)	M	H	IL4	IL5	IL5		N/A
MP-6 (3)		H					N/A
MP-7	M	H	IL4	IL5	IL5		N/A
MP-7 (1)	M	H	IL4	IL5	IL5		N/A
PE-1	M	H	IL4	IL5	IL5		N/A
PE-2	M	H	IL4	IL5	IL5		N/A
PE-2 (3)						x	N/A
PE-3	M	H	IL4	IL5	IL5		N/A
PE-3 (1)		H	IL4	IL5		x	N/A
PE-3 (4)						x	N/A
PE-4	M	H	IL4	IL5	IL5		N/A
PE-5	M	H	IL4	IL5	IL5		N/A
PE-6	M	H	IL4	IL5	IL5		N/A
PE-6 (1)	M	H	IL4	IL5	IL5		N/A
PE-6 (2)						x	N/A
PE-6 (3)						x	N/A
PE-6 (4)		H				x	N/A
PE-8	M	H	IL4	IL5	IL5		N/A
PE-8 (1)		H					N/A
PE-9	M	H	IL4	IL5	IL5		N/A
PE-10	M	H	IL4	IL5	IL5		N/A
PE-11	M	H	IL4	IL5	IL5		N/A
PE-11 (1)		H					N/A
PE-12	M	H	IL4	IL5	IL5		N/A
PE-13	M	H	IL4	IL5	IL5		N/A
PE-13 (1)		H					N/A
PE-13 (2)	M	H	IL4	IL5	IL5		N/A
PE-13 (3)	M	H	IL4	IL5	IL5		N/A
PE-14	M	H	IL4	IL5	IL5		N/A
PE-14 (2)	M	H	IL4	IL5	IL5		N/A
PE-15	M	H	IL4	IL5	IL5		N/A
PE-15 (1)		H					N/A
PE-16	M	H	IL4	IL5	IL5		N/A
PE-17	M	H	IL4	IL5	IL5		N/A
PE-18		H					N/A

Reference AWS Artifact: AWS_Customer_Responsibility_Matrix(CRM)_v4.5

Rationalizing Controls

PCI-DSS 3.2
10.1: Implement audit trails to link all access to system components to each individual user. It is critical to have a process or system that links user access to system components accessed…
10.2: Implement automated audit trails for all system components to reconstruct the following events…

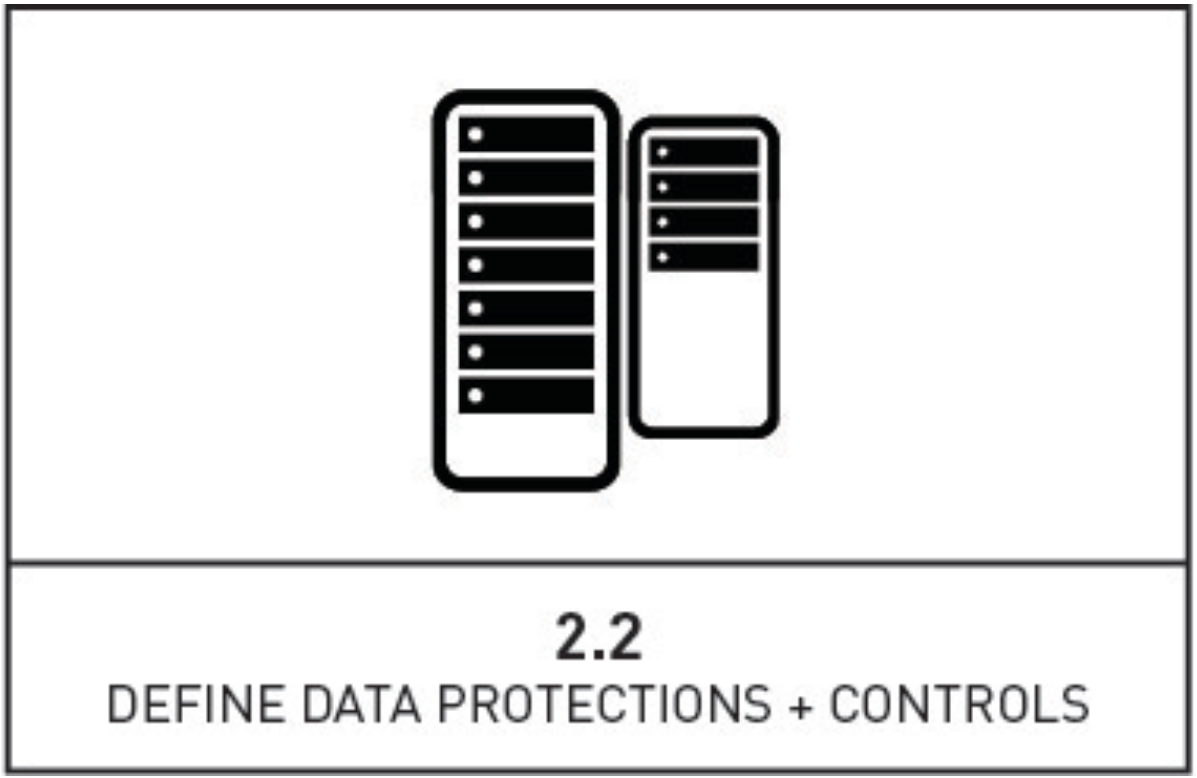
NIST 800-53 revision 4
AU-2: Audit Events - The organization: a. Determines that the information system is capable of auditing the following events…
AU-3: Content of Audit Records - The information system generates audit records containing information that establishes what type of event occurred…

HIPAA Security Rule
164.308(a)(1)(ii)(D) - Implement procedures to regularly review records of information system activity, such as audit logs, access reports, and security incident tracking reports…
164.308(a)(5)(ii)(c) - Procedures for monitoring log-in attempts and reporting discrepancies…

Fused Control Approach

Fused Audit Trail (Control Example) - Implement auditing for the following events (e.g. people, processes and actions) within the organizational use of cloud computing. Monitor for both positive and negative actions of users, system, services and applications. Secure, retain and automated audit trails as well as create communication paths to other security systems for analysis, reporting and investigations.

Define Data Protections

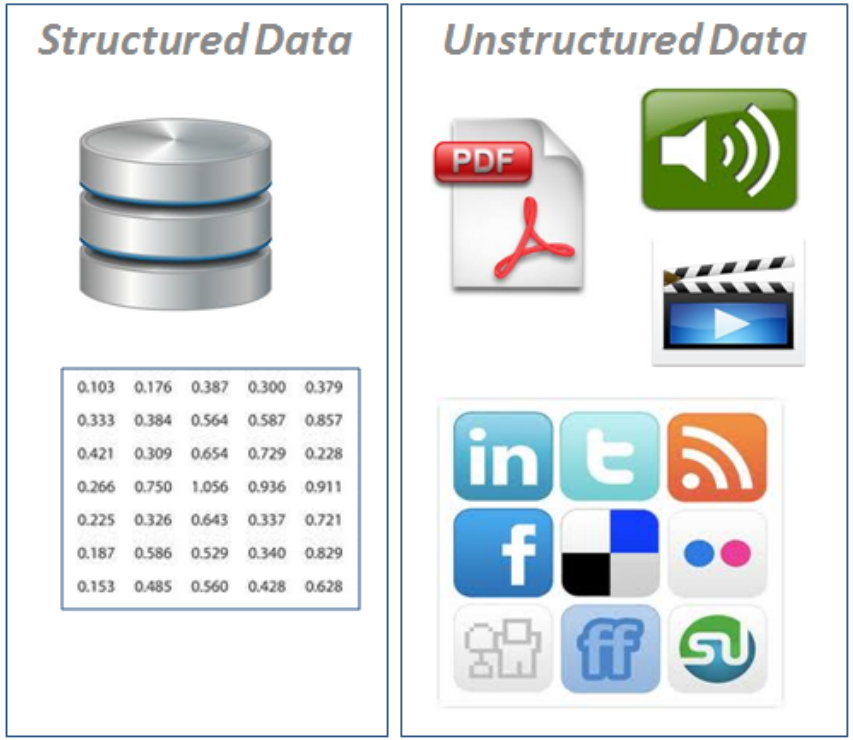


Problem Statement…

Issue #1 – The majority of organization do not have a mature “Data Classification” policy, process or user education schemes for internal use of data.

Issue #2 – Most organizations do not have a clean single source of “Truth” for what is their authoritative source for data. (Structured or Unstructured).

Issue #3 – Most organizations do not have an “Data Lifecycle” policy, procedure and/or operational processes for how data should be derived, protected, used, secured, transferred, achieved and destroyed when no longer relevant.



Data Protection Requirements

There are a number of regulatory, standards and frameworks which can impact data cloud computing.

- US - Health Insurance Portability and Accountability Act (HIPPA)
- US - Health Information Technology for Economic and Clinical Health Act (HITECH Act)
- US - Consumer Data Security and Notification Act (Amendment to Gramm-Leach-Bliley Act)
- EU - Directive 95/46/EC of the European Parliament and of the Council
- EU - Directive 2002/58 on Privacy and Electronic Communications (e.g.-Privacy Directive)
- General Data Protection Regulation (GDPR) (Regulation (EU) 2016/679)
- Canada - The Personal Information Protection and Electronic Documents Act (PIPEDA)
- UK - Data Protection Act 1998 (DPA)
- Australian - The Federal Privacy Act 1988
- Japan - The Act on the Protection of Personal Information (“APPI”)
- Singapore - Personal Data Protection Act 2012
- Philippines - Data Privacy Act of 2012
- South Korea - Personal Information Protection Act (“PIPA”)
- Hong Kong - The Personal Data (Privacy) Ordinance (Cap. 486) (“Ordinance”)

Data Protection Considerations

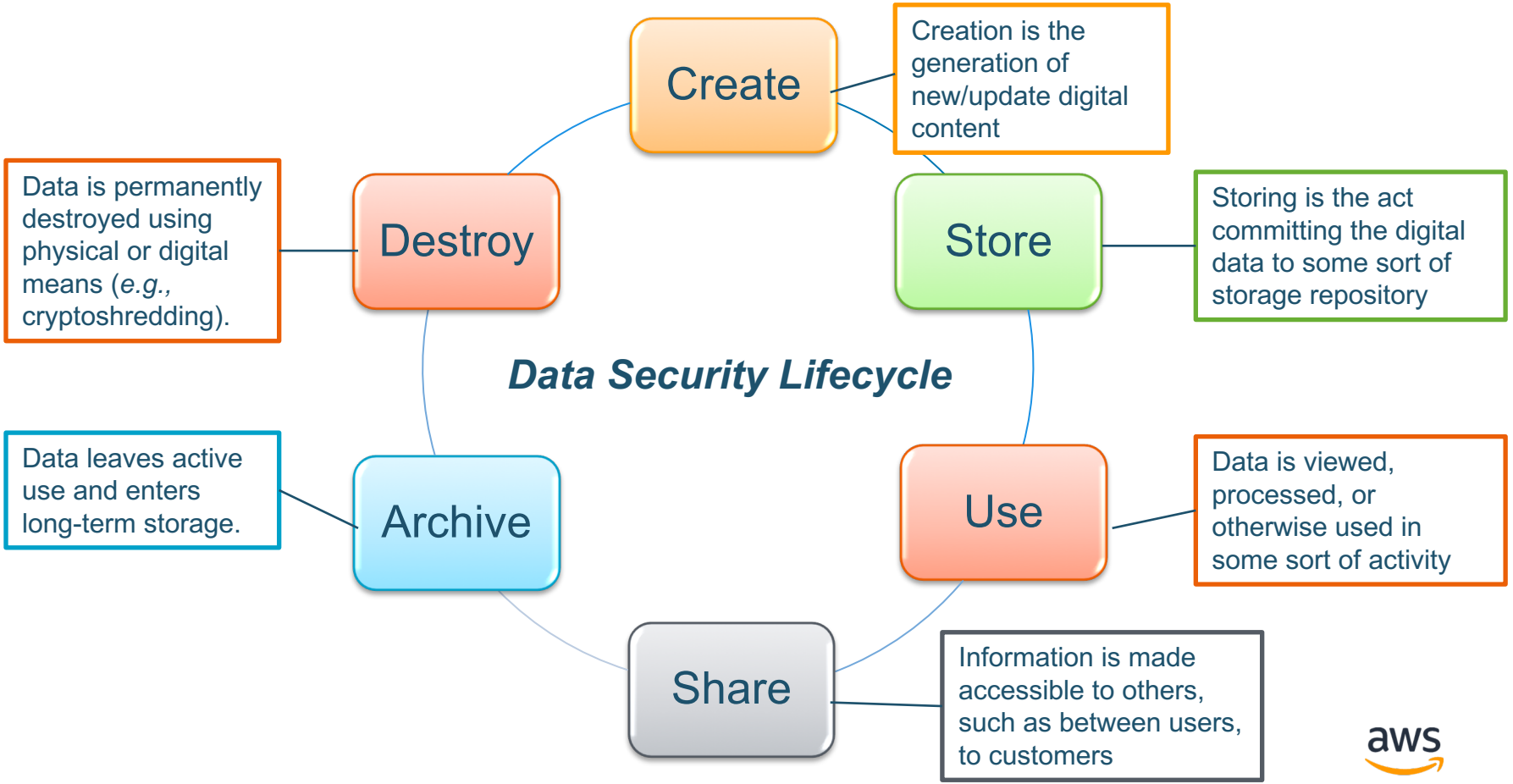
- **Access Controls**: Ensure organization can manage access to their content, services and resources independently of the cloud service provider.
    - Additionally, ensure cloud provider supports advanced set of access controls (MFA), encryption, and logging features.

- **Contractually**: Ensure the cloud providers does not access or use customer content for any purpose other than as legally required and for maintaining the cloud services.

- **Storage Locations**: Organization should be able to choose the geographic location in which their content will be stored. Cloud Providers should not move or replicate customer content outside of the customer's chosen locations.

- **Security**: Customers should choose how their customer content is secured. Through the use of various encryption of content in transit or at rest.
    - Additionally, organization should ensure they have option to manage their own encryption keys.

Define Data Protection + Controls



(Security Controls + Data Protections)

- Define your architecture capacity in advance
- Align your Test/Dev systems to Production
- Automate your Architecture
- Enable Continuous integration and Deploy
- Create a Tagging Strategy
- Enable a Data Protection architecture
- Test and game days

Defining a Tagging Strategy for Data Protection

Tagging allows organization to assign metadata to their cloud resources in the form of *tags*. Each tag is a simple label consisting of an organizational-defined key and an optional value that can make it easier to manage, search for, and filter resources.

Although there are no inherent types of tags, they enable organizations to categorize resources by purpose, owner, environment, or other criteria. As an example AWS has an outline of commonly used tagging categories and strategies to help AWS customers implement a consistent and effective tagging strategy.

General Best Practices for Tagging

- Always use a standardized, case-sensitive format for tags, and implement it consistently across all resource types.

- Consider tag dimensions that support the ability to manage resource access control, cost tracking, automation, and organization.

- Implement automated tools to help manage resource tags.
- The Resource Groups Tagging API enables programmatic control of tags, making it easier to automatically manage, search, and filter tags and resources. It also simplifies backups of tag data across all supported services with a single API call per AWS Region.

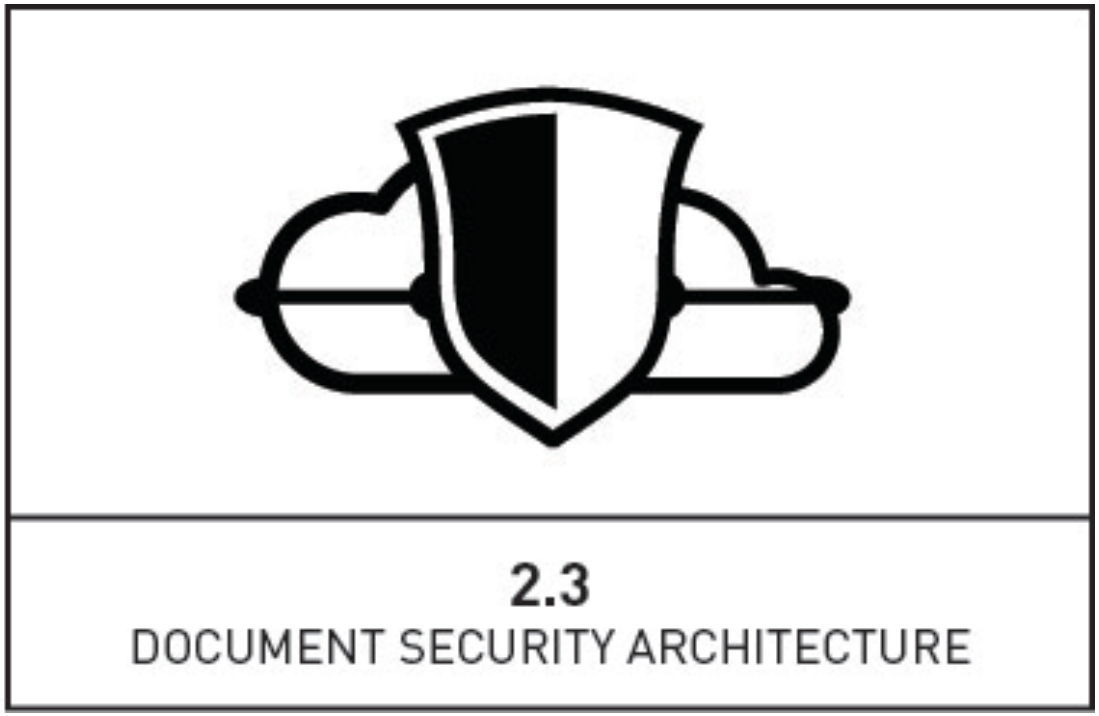
Tagging Categories

Companies that are most effective in their use of tags typically create business-relevant tag groupings to organize their resources along technical, business, and security dimensions.

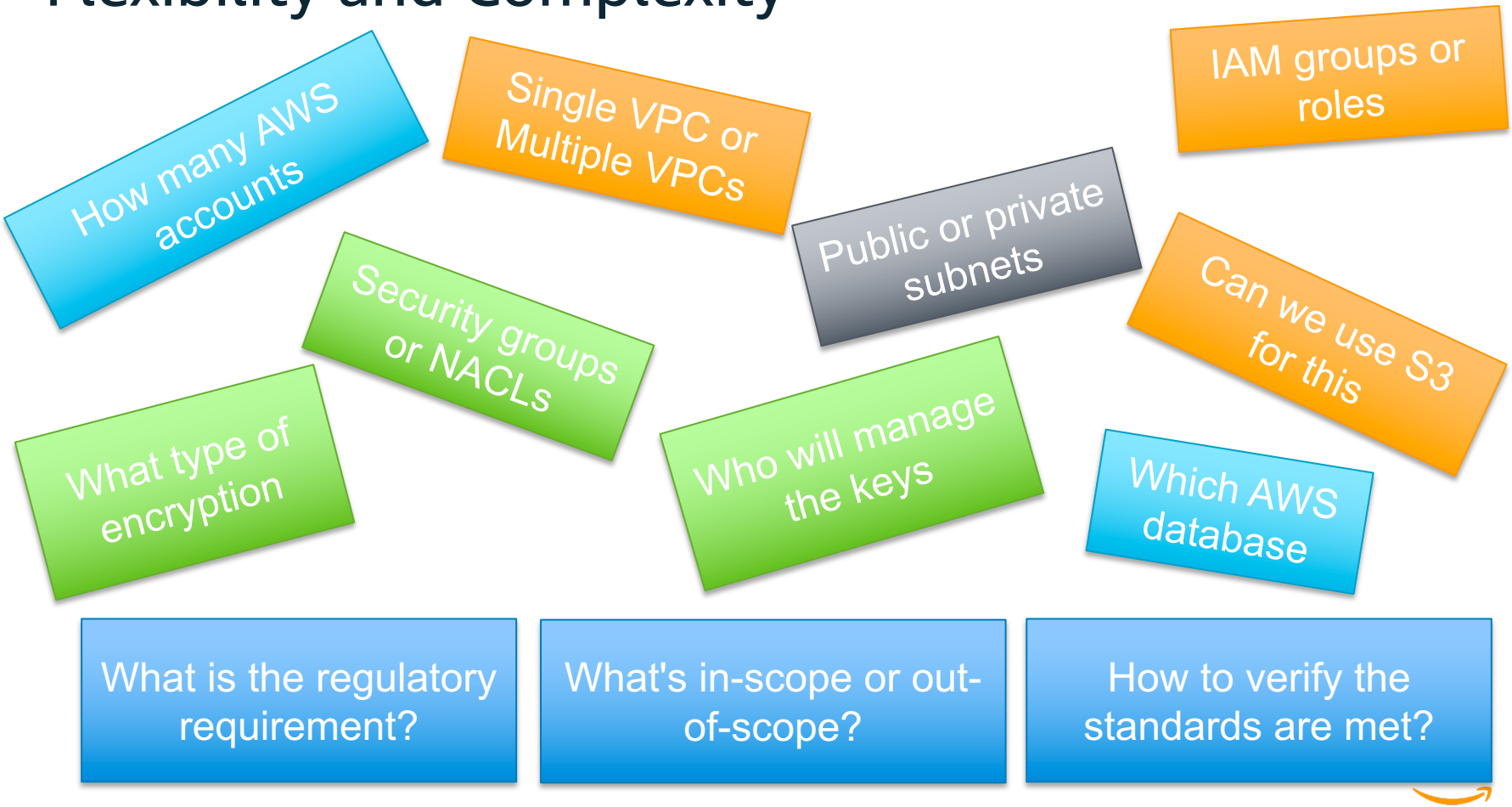
Companies that use automated processes to manage their infrastructure also include additional, automation-specific tags to aid in their automation efforts.

Technical Tags	Tags for Automation	Business Tags	Security Tags
Name – Used to identify individual resources **Application ID** – Used to identify disparate resources that are related to a specific application **Application Role** – Used to describe the function of a particular resource (e.g. web server, message broker, database) **Cluster** – Used to identify resource farms that share a common configuration and perform a specific function for an application **Environment** – Used to distinguish between development, test, and production infrastructure **Version** – Used to help distinguish between different versions of resources or applications	**Date/Time** – Used to identify the date or time a resource should be started, stopped, deleted, or rotated **Opt in/Opt out** – Used to indicate whether a resource should be automatically included in an automated activity such as starting, stopping, or resizing instances **Security** – Used to determine requirements such as encryption or enabling of VPC Flow Logs, and also to identify route tables or security groups that deserve extra scrutiny	**Owner** – Used to identify who is responsible for the resource **Cost Center/Business Unit** – Used to identify the cost center or business unit associated with a resource; typically for cost allocation and tracking **Customer** – Used to identify a specific client that a particular group of resources serves **Project** – Used to identify the project(s) the resource supports	**Confidentiality** – An identifier for the specific data-confidentiality level a resource supports **Compliance** – An identifier for workloads designed to adhere to specific compliance requirements

Security Architecture



Flexibility and Complexity

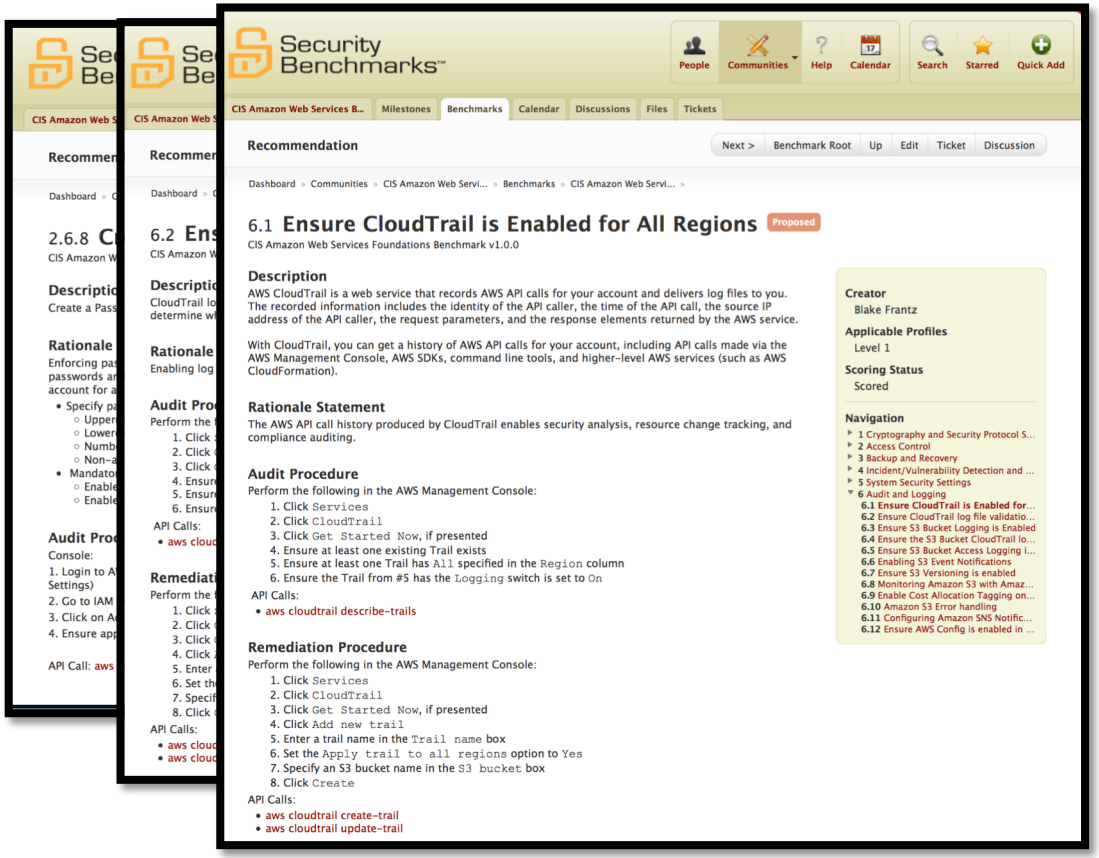


AWS Security Architecture Recipes

AWS has partnered with CIS Benchmarks to create consensus-based, best-practice security configuration guides which will align to multiple security frameworks globally.

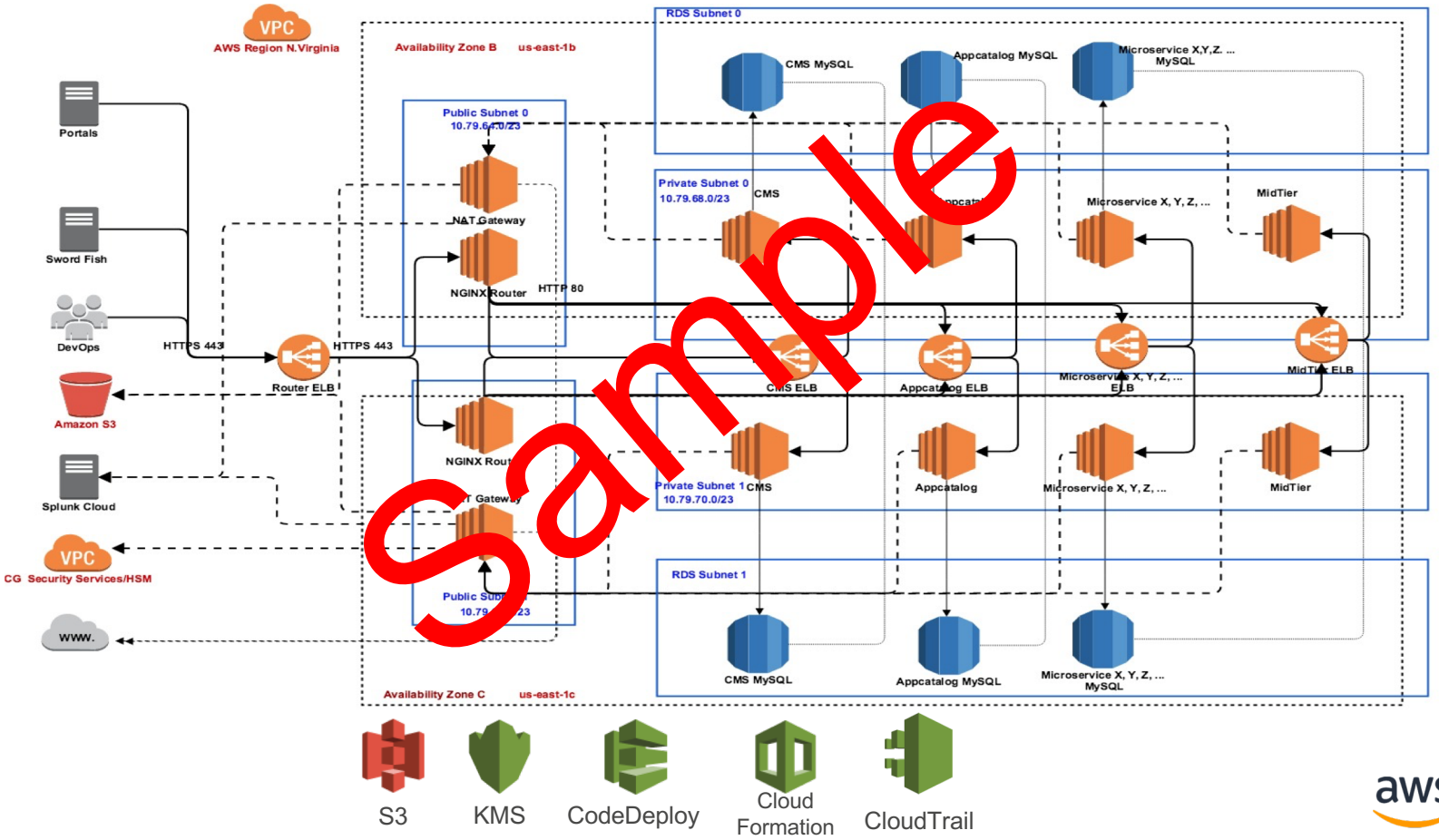
The Benchmarks are:

- Recommended technical control rules/values for hardening operating systems, middle ware and software applications, and network devices;
- Distributed free of charge by CIS in .PDF format
- Used by thousands of enterprises as the basis for security configuration policies and the de facto standard for IT configuration best practices.

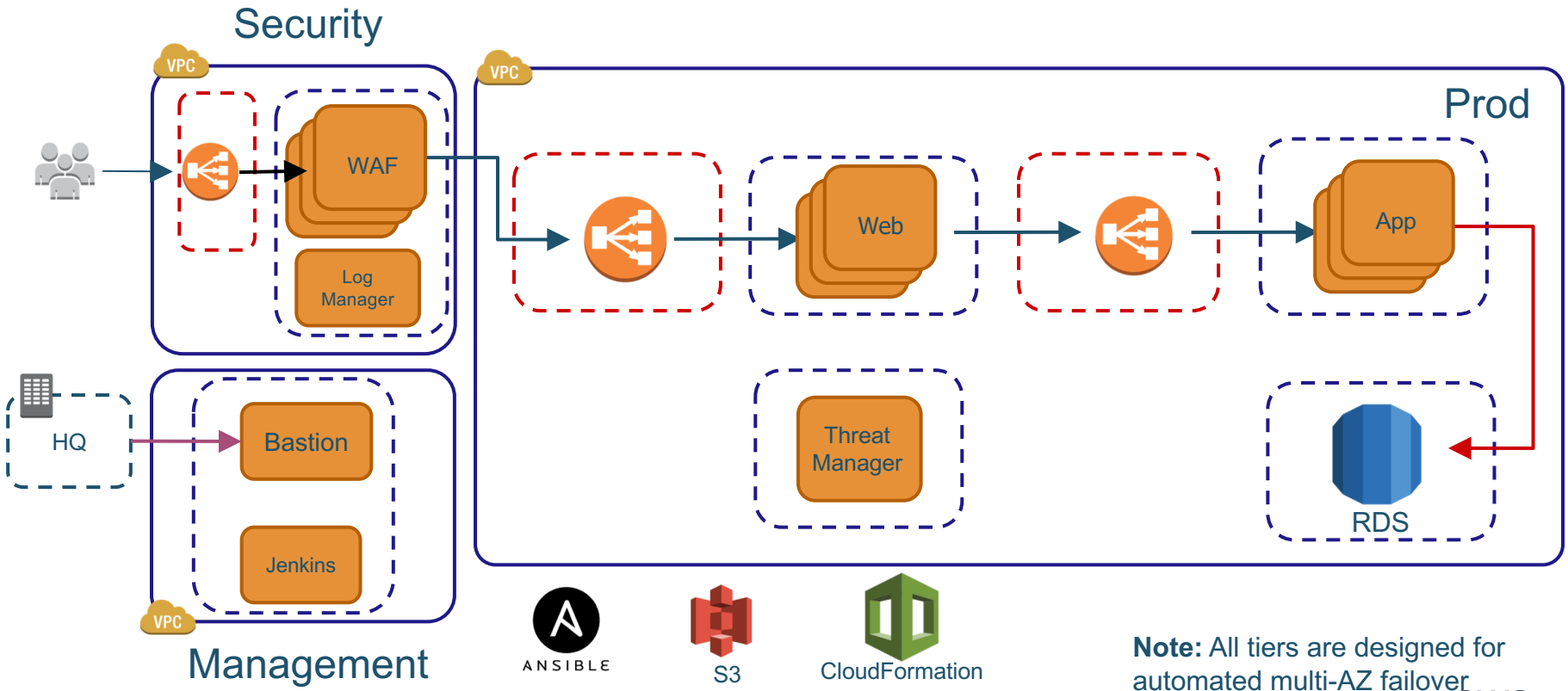


https://www.cisecurity.org/

Document your Security Architecture



Automated multi-AZ failover



Note: All tiers are designed for automated multi-AZ failover

aws

Governance as Code – *Alignments*

Modernizing Technology Governance *in the Cloud*

Risk Management

Security & Compliance lifecycle

Human Governance

Policy, Procedure and System Security Plans

Incident Response

(Identify, Protect, Detect, Respond & Recover)

Configuration Management

(Packaging, Configuration and Continuous Delivery)

aws

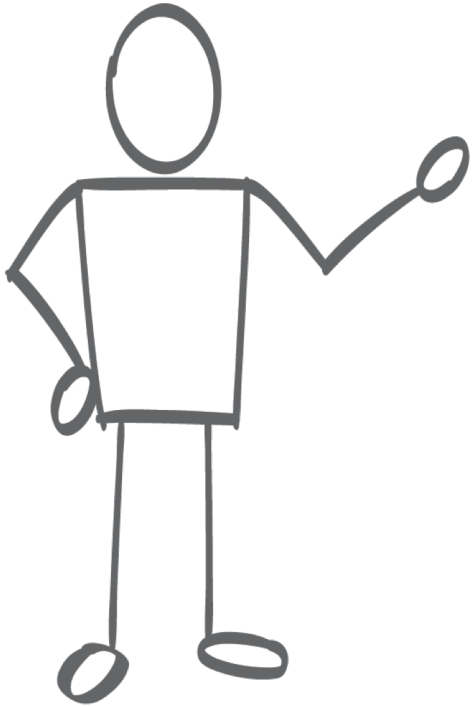
Governance as Code

Is the process of managing and provisioning machine-readable definition files, templates, scripts and recipes for regulatory workload configurations. GaC interacts with Continuous Configuration Automation (CCA) tools (e.g., Chef, Puppet, Ansible etc.) and can be thought of as an extension of traditional Infrastructure of Code frameworks. The goal of GaC is to version control solutions as scripts or declarative definitions which meet regulatory requirements and adherence with audit frameworks.

GaC continued…

There are generally three approaches to GaC:

- **Declarative (functional) –** Aspirational (e.g. **desired state**) target configuration against regulatory requirements.
- **Imperative (procedural) –** Defines code management (**desired conclusion**) and assertion to the regularity adherence.
- **Intelligent (environment aware) –** Specifies the configuration (**correct desired state**) based on relationships, dependencies and interaction in a regulated production environment.



Questions