Module-4

Security Management in the Cloud: Introduction, Security Management Standards, Security Management in the Cloud, Availability Management, SaaS Availability Management, PaaS Availability Management, IaaS Availability Management, Access Control, Security Vulnerability, Patch, and Configuration Management.	**8 Hours**

Security Management in the Cloud

- WITH THE ADOPTION OF PUBLIC CLOUD SERVICES, A LARGE PART OF YOUR NETWORK, system, applications, and data will move under third-party provider control.

- The cloud services delivery model will create islands (clouds) of virtual perimeters as well as a security model with responsibilities shared between the customer and the cloud service provider (CSP).

- This shared responsibility model will bring new security management challenges to the organization's IT operations staff.

- With that in mind, the first question a chief information security officer (CISO) must answer is **whether she has adequate transparency from cloud services to manage the governance (shared responsibilities)** and implementation of security management processes (preventive and detective controls) to assure the business that the data in the cloud is appropriately protected

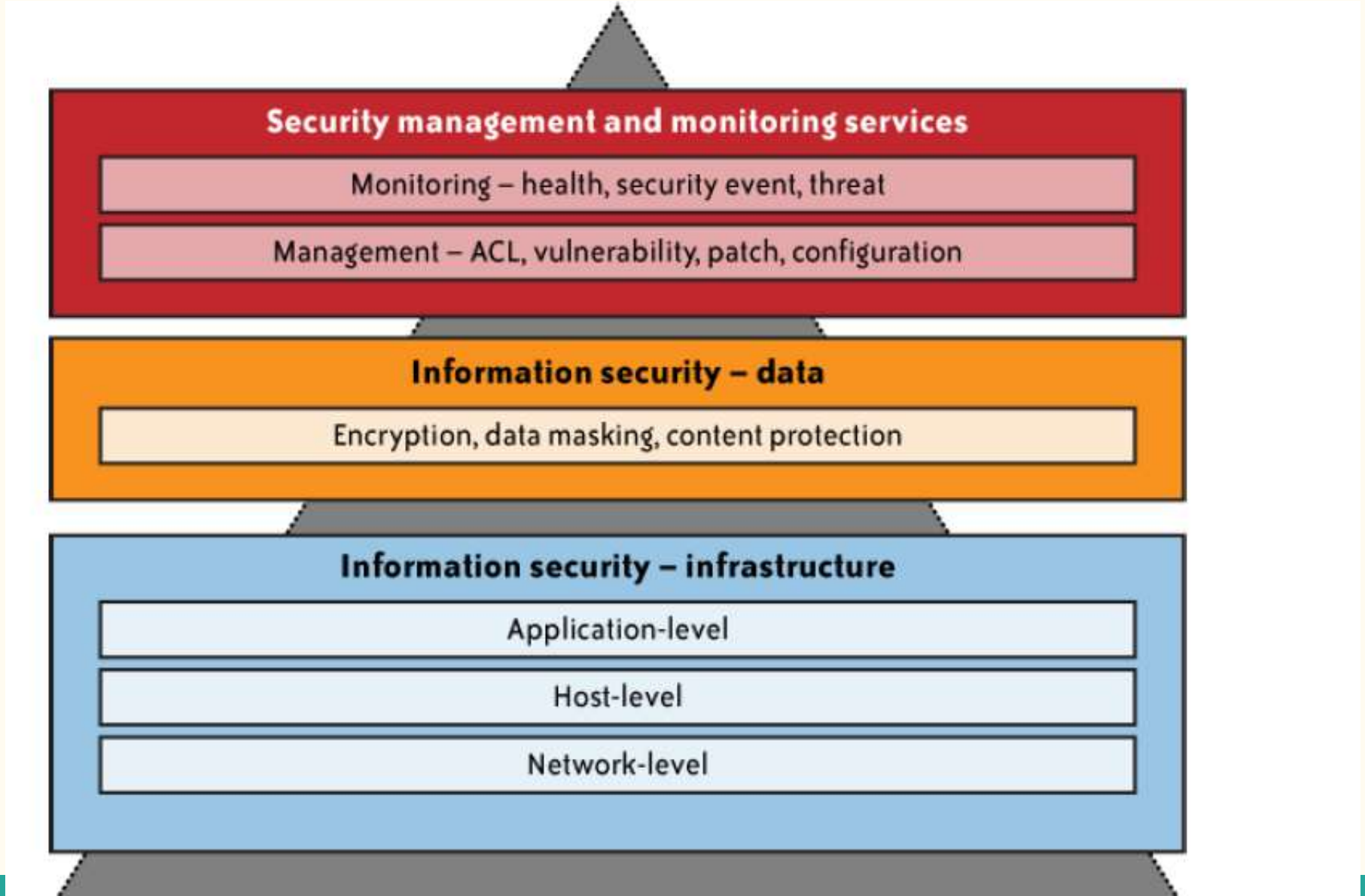
- The answer to this question has two parts: what security controls must the customer provide over and above the controls inherent in the cloud platform, and how must an enterprise's security management tools and processes adapt to manage security in the cloud.

- Both answers must be continually reevaluated based on the sensitivity of the data and the service-level changes over time.

USER -Responsibility

- As a customer of the cloud, you should start with the exercise of understanding the **trust boundary of your services in the cloud**.
- You should understand all the layers you own, touch, or interface with in the cloud service—network, host, application, database, storage, and web services including identity services .
- You also need to understand the scope of IT system management and monitoring responsibilities that fall on your shoulders, including access, change, configuration, patch, and vulnerability management.

Security management and monitoring scope



Cloud Security Management and ITIL Framework

When organizations adopt cloud services, some operational responsibilities are shifted to the provider.

However, the **extent of responsibility varies** depending on factors such as:

- Service delivery model (SPI): SaaS, PaaS, or IaaS
- Provider’s SLA (Service-Level Agreement): what is formally guaranteed
- Provider’s capabilities: ability to integrate with your existing security processes and tools

Role of Security Management Frameworks

- Mature IT organizations often adopt **industry-standard frameworks** like:
    - **ISO/IEC 27000:** Provides best practices for information security management.
    - **ITIL (Information Technology Infrastructure Library):** Offers structured guidance for IT service management.
- These frameworks:
- Help plan and implement **governance programs** that protect information assets.
- Provide **detailed practices, checklists, and procedures** that can be adapted to any IT organization.

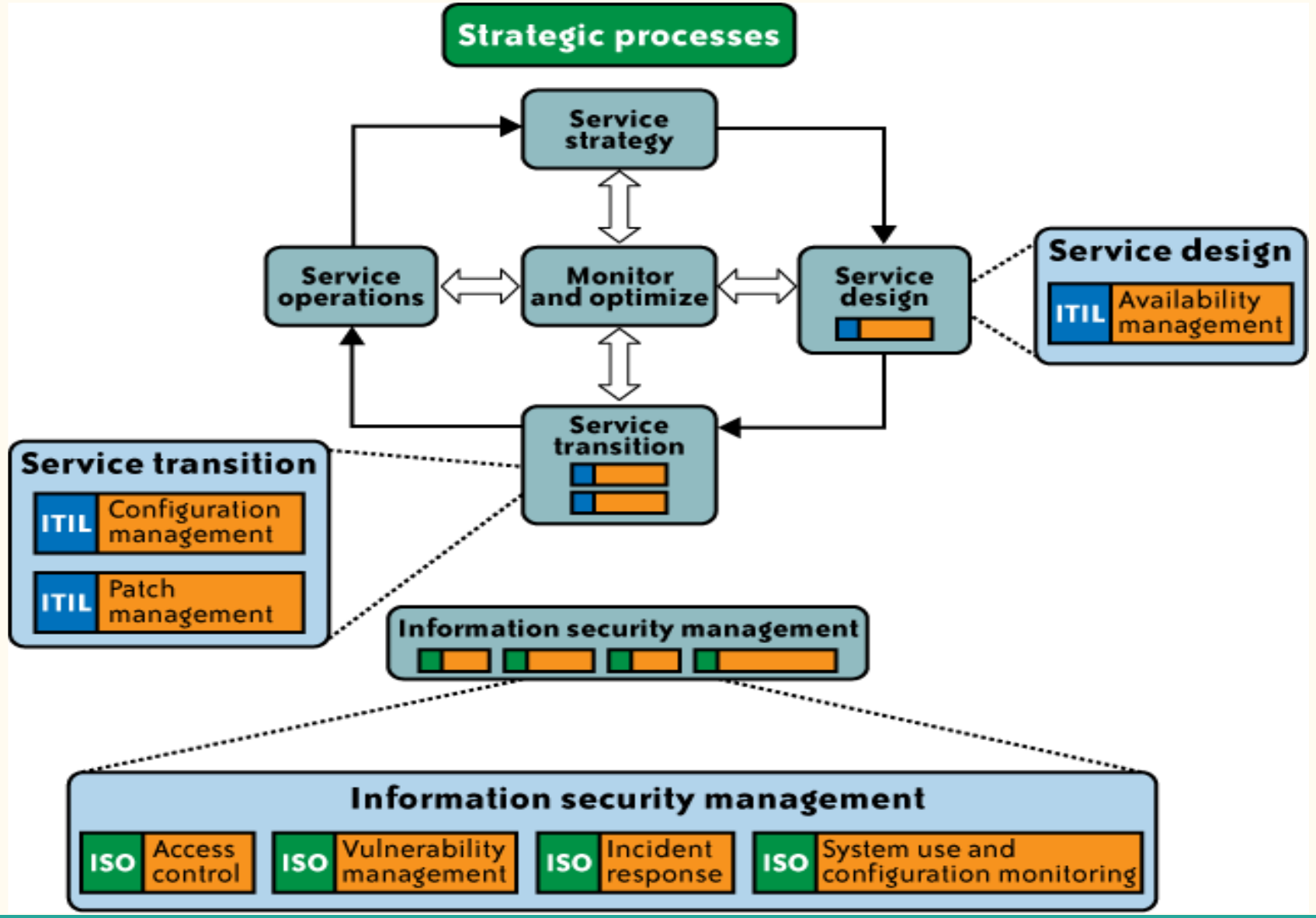
ITIL and Continuous Service Improvement

- A central principle of ITIL is recognizing that:
  - **Organizations, processes, and IT systems constantly change.**
  - Cloud computing, being **dynamic by nature, demands continuous alignment of IT services with business needs.**
- **Continuous Service Improvement (CSI)** in ITIL means:
  - Regularly identifying and implementing improvements in IT services.
  - Ensuring IT services (e.g., cloud-hosted CRM or sales automation tools) continue to support evolving business processes effectively.
- For cloud environments, this means **security management processes must be regularly revised** to stay effective and up to date.

Goals of the ITIL Security Management Framework

- **Realization of Security Requirements**
  - Defined in SLAs, contracts, legislation, or internal/external policies.
- **Realization of a Basic Level of Security**
  - Ensures **continuity and resilience** of the organization.
  - Simplifies **service-level management** for security.

The ITIL life cycle in an enterprise



1. Strategic Processes (Top Level)

- **Service Strategy ---**Defines the overall goals and direction of IT services.
- **Service Design----** Translates strategy into detailed service specifications (availability, performance, security).
- **Service Transition**:----Focuses on introducing new/changed services into operations (configuration, patching).
- **Service Operations------** Day-to-day running of services (monitoring, incident management).
- **Monitor and Optimize** ---Continuous feedback loop to improve efficiency and effectiveness

2. Service Design (Right Side)

- Includes **Availability Management** (ITIL): ensures services meet agreed availability levels.

3. Service Transition (Left Side)

- Covers how new or modified services are transitioned into live use.
- Linked ITIL practices:
  - **Configuration Management**: Tracks and manages service assets and configurations.
  - **Patch Management**: Ensures software updates and fixes are applied securely

4. Information Security Management (Bottom Layer)

- This underpins all ITIL processes and connects to ISO standards for security:
- **Access Control (ISO)-----** Ensures only authorized users have access.
- **Vulnerability Management (ISO):-----**Identifies and mitigates weaknesses.
- **Incident Response (ISO):---------** Processes for detecting, responding to, and recovering from security incidents.
- **System Use and Configuration Monitoring (ISO)-------**Ensures secure system usage and configuration consistency.

Security Management Standards

- **ITIL**
- **ISO 27001/27002**

ITIL (Information Technology Infrastructure Library)

1. What is ITIL?

- **ITIL (Information Technology Infrastructure Library)** = A set of **best practices** and **guidelines** for IT service management (ITSM).
- Provides an **integrated, process-based approach** for delivering and managing IT services.
- Applicable to almost any IT environment, including **cloud computing**.

2. ITIL & Information Security

- ITIL ensures **security is addressed at all levels**:
  - **Strategic** (long-term planning and goals).
  - **Tactical** (policies, processes, governance).
  - **Operational** (day-to-day security actions).
- Treats **information security as an iterative process**:
  - Controlled → Planned → Implemented → Evaluated → Maintained.

3. ITIL Security Framework Components

- **Policies** → Define overall security objectives.
- **Processes** → What needs to be done to achieve those objectives.
- **Procedures** → Who performs tasks, when, and how.
- **Work Instructions** → Detailed step-by-step instructions for specific actions.

4. Standards and Alignment

- ITIL's **security management process** is based on **ISO/IEC 17799:2005** (best practices for information security, now evolved into ISO/IEC 27002).
- Strong relationships with other ITIL processes:
  - **Service-Level Management** → Defines and enforces security-related SLAs.
  - **Incident Management** → Responds to and mitigates security incidents.
  - **Change Management** → Ensures security is maintained during system changes.
- ITIL is also related to **ISO/IEC 20000**:
  - The **first international standard** for IT Service Management.
  - Based on ITIL, but unlike ITIL, **ISO/IEC 20000 allows certification**.

ISO/IEC 27001 – Information Security Management System (ISMS)

1. Definition

- **ISO/IEC 27001** is an **international standard** that specifies the requirements for **establishing, implementing, maintaining, and continuall**y improving an **Information Security Management System (ISMS)**.
- It provides a **risk-based framework** to protect the **confidentiality, integrity, and availability** (CIA triad) of information.

2. Purpose

- To **help organizations** **manage information security systematically** rather than ad hoc.
- To ensure that **security controls are aligned with business needs and risks.**
- To **provide assurance to stakeholders** (customers, regulators, partners) that the organization manages information security responsibly.

3.Key Features

- **Risk Management Approach** → identify, assess, and treat risks to information assets.
- **Mandatory Requirements** → organizations must document policies, assign roles, conduct risk assessments, monitor performance, and implement continual improvement.
- **Certifiable Standard** → organizations can undergo an **independent audit** and achieve **ISO/IEC 27001 certification**.

4. Scope of ISO 27001

- Covers **all types of information assets**: digital, physical, intellectual property, employee knowledge.
- Applies to **organizations of all sizes and industries** (corporate, government, startups, etc.).
- Covers **people, processes, and technology** in information security.

Security Management in the Cloud

- After analyzing **the management process disciplines across the ITIL and ISO frameworks**, (the researchers) identified the following **relevant processes** as the **recommended security management** focus areas for securing services in the cloud:

1. **Availability management (ITIL)**
2. **Access control (ISO/IEC 27002, ITIL)**
3. **Vulnerability management (ISO/IEC 27002)**
4. **Patch management (ITIL)**
5. **Configuration management (ITIL)**
6. **Incident response (ISO/IEC 27002)**
7. **System use and access monitoring (ISO/IEC 27002)**

Availability management (ITIL)

- Availability management (ITIL)
  Factors Impacting Availability
    - **SaaS Availability Management**
        - **Customer Responsibility**
    - **PaaS Availability Management**
        - **Customer Responsibility**
        - **PaaS Health Monitoring**
    - **IaaS Availability Management**

Availability management (ITIL)

- Cloud services are **not immune to outages**; plan for failure.
- **Availability** is a shared responsibility between **customer and CSP.**
- **Criticality of the** application determines business impact.
- Short outages can **still cause large financial or reputational loss.**
- **Productivity loss** for internal teams and customers.
- Direct revenue loss (e-commerce, transactions, billings).
- **Regulatory and SLA** penalties for downtime.

- **Outage Characteristics (expanded)**
  - **Duration**: minutes → hours → rare multi-day events.
  - **Scope: single region, single service, multi-region, or global**.
  - **Type**: complete outage, partial degradation, or intermittent failures.
- **Common Root Causes**
  - Network/routing/DNS misconfigurations or BGP issues.
  - Vendor/service software bugs and platform upgrades gone wrong.
  - Database cluster failures or storage corruption.
  - Human/configuration error during deployment (wrong rollback).
  - Capacity overloads and cascading failures under load.

Factors Impacting Availability

- The cloud service resiliency and availability depend on a few factors, **including the CSP's data center architecture** (load balancers, networks, systems**), application architecture, hosting location redundancy, diversity of Internet service providers (ISPs), and data storage architecture.**
- Following is a list of the major factors:
  1. **SaaS and PaaS application architecture and redundancy.**
  2. Cloud service **data center architecture, and network and systems architecture**, including geographically diverse and fault-tolerance architecture.
  3. Reliability and redundancy **of Internet connectivity** used by **the customer and the CSP.**
  4. Customer's ability to respond **quickly and fall back on internal applications** and other processes, including manual procedures.

1. Customer's visibility of the fault. In some downtime events, if the impact affects a small subset of users, it may be difficult to get a full picture of the impact and can make it harder to troubleshoot the situation.[**when only a small group of users is affected, typical global monitors and dashboards can look "green," so you may not see the problem right away. That lack of obvious signals makes it harder to detect, scope and troubleshoot the root cause.]**
2. **Reliability of hardware and software components** used in delivering the cloud service.
3. Efficacy of the **security and network infrastructure** to withstand a distributed **denial of service (DDoS) attack** on the cloud service.
4. Efficacy of **security controls and processes that reduce human error and protect infrastructure** from malicious **internal and external threats**, e.g., privileged users abusing privileges.

SaaS Availability Management

•SaaS providers **own business continuity, application and infrastructure management.**

•This shifts **operational responsibilities** from IT to the CSP.

•IT must map **internal service-level needs to vendor SLAs and terms**.

•Governance challenge: align business criticality (e.g., Marketing app) with what the SaaS vendor actually promise

•Some vendors provide formal SLAs; others rely on terms & conditions

•**Uptime / availability** goal (e.g., 99.5% / 99.9%).Scheduled vs unscheduled maintenance (what counts as downtim

•**Service credit / remediation** rules and how to claim them.

•**Exclusions** (customer-caused issues , third-party failures).

SaaS Health Monitoring

- **The following options are available to customers to stay informed on the health of their service:**

1. Service **health dashboard** published by the CSP. Usually SaaS providers, such as Salesforce.com, publish the current state of the service, current outages that may impact customers, and upcoming scheduled maintenance services on their website (e.g., *http:// trust.salesforce.com/trust/status/*).

- The **Cloud Computing Incidents Database (CCID).** (This database is generally community supported, and may **not reflect all CSPs and all incidents** that have occurred.)

- **Customer mailing list** that notifies customers of occurring and recently occurred outages.
- Internal or third-party-based service monitoring tools that periodically check SaaS provider health and alert customers when service becomes unavailable (e.g., Nagios monitoring tool).
- RSS feed hosted at the SaaS service provider.

- **What is an RSS feed from a SaaS provider?**
- An **RSS feed** (Really Simple Syndication) is a machine-readable XML feed the provider publishes to broadcast short updates — typically status updates, outage notices, and scheduled-maintenance announcements. Subscribing to that RSS feed means you'll automatically receive those updates (**in a feed reader, monitoring tool, or custom scrip**t) as soon as the provider posts them.

What an RSS item looks like (example)

```
<item>
  <title>Scheduled Maintenance: Database cluster</title>
  <link>https://trust.example.com/incidents/123</link>
  <pubDate>Tue, 30 Sep 2025 22:00:00 GMT</pubDate>
  <description>We will perform maintenance on DB cluster starting 22:00
UTC.</description>
  <guid>https://trust.example.com/incidents/123</guid>
</item>
```

PaaS Availability Management

- **Availability management of the PaaS application can be complicated?**
- **Why ?**

- In a typical PaaS service, customers (developers) build and deploy PaaS applications on top of the CSP-supplied PaaS platform.
- The PaaS platform is typically built on a CSP owned and managed network, servers, operating systems, storage infrastructure, and application components (web services).
- Given that the customer PaaS applications are assembled with CSP-supplied application components and, in some cases, third-party web services components (mash-up applications), availability management of the PaaS application can be complicated—

- **Shared-responsibility model** — the CSP owns the PaaS platform and underlying infra; the customer owns the application code and any third-party services the app uses.
- **CSP-managed infrastructure** — PaaS runs on provider-controlled network, servers, OS, storage and platform components (so platform availability is the provider's responsibility).
- **Mash-ups increase risk** — apps that combine CSP-supplied components and external web services (e.g., Google Maps, Facebook APIs) have more points of failure.
- **Dependency mapping is essential** — identify and document every external service (CSP components + third parties) your app depends on and the impact if each fails.

Example:

- The customer is responsible for managing the **availability of the customer-developed application and third-party services,** and the PaaS CSP is responsible for the PaaS platform and any other services supplied by the CSP.
- For example, Force.com is responsible for the management of the AppExchange platform, and customers are responsible for managing the applications developed and deployed on that platform.

PaaS Quotas & Service Levels

- Customers are encouraged to **read and understand the PaaS platform service levels** (if available), including quota triggers that may limit resource availability for their application (usually outlined in the SLA, or in the terms and conditions of the PaaS service).
- In cases where the PaaS platform enforces quotas on compute resources (CPU, memory, network I/O), upon reaching the thresholds the application may not be able to respond within the normal latency expectations and could eventually become unavailable.
- For example, **the Google App Engine has a quota system** whereby each App Engine resource is measured against one of two kinds of quotas: a **billable quota or a fixed quota.**

- What are Quotas?
  - Quotas limit resources (CPU, memory, network I/O, API calls) to protect platform integrity.They can be billable (you control & pay to raise) or fixed (provider-enforced hard limits).
- **Billable Quotas (user-controlled)**
  - Set by the **app admin to cap spend**; you can increase them by enabling **billing and allocating budget.**
  - Every app usually gets free-tier amounts; charges apply only above free thresholds.
  - Use billable quotas to **prevent unexpected costs** while allowing scaling when needed.

Fixed Quotas (provider-controlled)

- Enforced by the platform to ensure **fairness and system stability.**
- Cannot be exceeded; hitting a fixed quota can cause throttling, errors, or app unavailability.
- Examples: per-minute API calls, thread limits, per-request CPU time.

Customer Responsibility

- Considering all of the variable parameters in availability management**, the PaaS application customer should carefully analyze the dependencies of the application on the third-party web services** (components) and outline a holistic management strategy to manage and monitor all the dependencies.
- The following considerations are for PaaS customers:

1. ***PaaS platform service levels***
    - Customers should carefully review the terms and conditions of the CSP's SLAs and understand the availability constraints.

2. *Third-party web services provider service levels*

- When your PaaS application depends on a third-party service, it is critical to understand the SLA of that service.

PaaS Health Monitoring

- **The following options are available to customers to monitor the health of their service:**

1. **Service health dashboard** published by the CSP (e.g., http://status.zoho.com)
2. **CCID** (this database is generally community-supported, and may not reflect all CSPs and all incidents that have occurred)
3. CSP customer **mailing list** that notifies customers of occurring and recently occurred
4. outages
5. **RSS feed** for RSS readers with availability and outage information
6. **Internal or third-party-based service monitoring tools** that periodically check your PaaS application, as well as third-party web services that monitor your application (e.g., Nagios monitoring tool)

IaaS Availability Management

- Availability considerations for the IaaS delivery model should include both a computing and storage (persistent and ephemeral) infrastructure in the cloud.
- IaaS providers may also offer other services such as account management, a message queue service, an identity and authentication service, a database service, a billing service, and monitoring services.
- Hence, availability management should take into consideration all the services that you depend on for your IT and business needs.

Managing IaaS virtual infrastructure in the cloud depends on five factors:

1. Availability of a CSP network, host, storage, and support application infrastructure. This factor depends on the following:

1. CSP data center architecture, including a geographically diverse and fault-tolerance architecture.
2. Reliability, diversity, and redundancy of Internet connectivity used by the customer and the CSP.
3. Reliability and redundancy architecture of the hardware and software components used for delivering compute and storage services.
4. Availability management process and procedures, including business continuity processes established by the CSP.
5. Web console or API service availability.
6. SLA.

2. Availability of your virtual servers and the attached storage (persistent and ephemeral) for compute services (e.g., Amazon Web Services' S3† and Amazon Elastic Block Store).

3. Availability of virtual storage that your users and virtual server depend on for storage service.

4. Availability of your network connectivity to the Internet or virtual network connectivity to IaaS services.

5. Availability of network services, including a DNS, routing services, and authentication services required to connect to the IaaS service.

IaaS Health Monitoring

- **The following options are available to customers to monitor the health of their service:**

1. **Service health dashboard** published by the CSP (e.g., http://status.zoho.com)
2. **CCID** (this database is generally community-supported, and may not reflect all CSPs and all incidents that have occurred)
3. CSP customer **mailing list** that notifies customers of occurring and recently occurred
4. outages
5. **RSS feed** for RSS readers with availability and outage information
6. **Internal or third-party-based service monitoring tools** that periodically check your PaaS application, as well as third-party web services that monitor your application (e.g., Nagios monitoring tool)

Access Control

- Access control management is a broad function that encompasses access requirements for your users and system administrators (privileged users) who access network, system, and application resources. The access control management functions should address the following:
- Who should have access to what resource? (Assignment of entitlements to users)
- Why should the user have access to the resource? (Assignment of entitlements based on the user's job functions and responsibilities)
- How should you access the resource? (What authentication method and strength are required prior to granting access to the resource)
- Who has access to what resource? (Auditing and reporting to verify entitlement assignments)

Access Control in the Cloud:

1. **Shift from Network-Based to User-Based Access Control**

   Traditional (on-premise) model:

   Access control was mostly **network-based**, meaning it depended on **hostnames, IP addresses, or physical network boundaries** (e.g., only allow access from devices within the company LAN).

- **Cloud model:**
  - These traditional controls are **less effective** because cloud resources are **dynamic, distributed, and accessible via the Internet**, not a single secure network.
  - Instead, control is based on **user identity** — who the user is, not where they are connecting from.
  - This approach is called **identity-based access control**, often implemented using technologies like **IAM (Identity and Access Management)**, **OAuth**, or **SAML**.
  - It ensures that **each user's identity and permissions** directly determine what cloud resources they can access.
- **Example:**

  In AWS or Azure, rather than granting access to a specific IP, you assign permissions to a user or role (e.g., "developer" can access S3 buckets A and B).

2. Network Access Control in the Cloud

- Although user-based control is dominant, **network-level control still exists** — mainly through **cloud firewalls**.
- These **firewall policies** control **ingress (incoming)** and **egress (outgoing)** network traffic to and from virtual machines or containers in the cloud.
- Access is configured using **TCP/IP parameters**, such as:
    - **Source IP/Port:** Where the traffic originates.
    - **Destination IP/Port:** Which system and service it's targeting.
- **Example:**

  You can configure rules to only allow SSH access (port 22) from your office IP to a specific cloud server.
- Such rules help to:
    - **Group instances logically** (e.g., web-tier vs. database-tier).
    - **Enforce host-level restrictions**, ensuring only authorized traffic reaches sensitive systems.

3. Importance of User Access Management

- Critical for ensuring **fine-grained access control**, **user accountability**, and **data protection**.
- Key mechanisms include:
  - **Strong authentication**
  - **Single Sign-On (SSO)**
  - **Privilege management**
  - **Logging and monitoring**

4. ISO/IEC 27002 Access Control Objectives

- ISO/IEC 27002 defines **six access control objectives** that apply to end users, privileged users, networks, applications, and information.
- **Main objective:**

  “Ensure authorized user access and prevent unauthorized access to information systems.”

5. Lifecycle-Based Access Control

- Access management should cover **all stages**:
  - User registration → Allocation of rights → De-registration
- Must also **control privileged access rights**, as these can override system controls.

6. Six ISO/IEC 27002 Control Statements

- Control access to information
- Manage user access rights
- Encourage good access practices
- Control access to network services
- Control access to operating systems
- Control access to applications and systems

Access Control: SaaS

1. CSP Responsibility

- In the **SaaS model**, the **Cloud Servic5e Provider (CSP)** is responsible for managing **everything** — including the **network, servers, storage, and applications**.
- Customers (end users) do **not control or manage** the infrastructure or platform components.
- The CSP ensures **secure access**, uptime, and proper handling of data.

2. Focus on User Access Controls

- SaaS applications are typically **accessed via web browsers** (e.g., Gmail, Salesforce, Microsoft 365).
- Therefore, **traditional network-based access control** (based on IPs or firewalls) is **less relevant**.
- The **main security focus** shifts to **user access management**, which includes:
  - **Authentication:** Verifying user identity (passwords, MFA, etc.)
  - **Federation:** Using a single identity across multiple systems (e.g., via SAML, OAuth).
  - **Privilege Management:** Assigning role-based permissions (e.g., admin, editor, viewer).
  - **Deprovisioning:** Revoking access when users leave or roles change.

3. Combined Network and User Controls

- Some **SaaS providers** combine both **network-based** and **user-based** access control.
- For instance, **Salesforce.com** can restrict access using:
  - **Network/IP policies** (only certain IP ranges can log in), and
  - **User-based rules** (permissions tied to identity, role, or policy).
- This **hybrid approach** provides stronger layered security.

4. Inconsistent IAM Support

- **Identity and Access Management (IAM)** features — such as **authentication**, **Single Sign-On (SSO)**, and **multi-factor authentication (MFA)** — **vary across providers**.
- Some SaaS platforms offer **robust IAM**, while others have **limited or non-standardized controls**.
- Hence, enterprises must evaluate IAM capabilities before adopting a SaaS solution.

5. Adoption of Standards by Leading Providers

- Major cloud providers like **Google, Microsoft, and Salesforce** have adopted **SAML (Security Assertion Markup Language)** to enable **Single Sign-On (SSO)** and **identity federation**.
- **SAML** allows users to access multiple cloud applications using one set of credentials — improving both **security** and **user convenience**.
- However, even though these standards are supported, **capabilities differ among providers** — not all offer the same level of flexibility, integration, or customization.
- For example, one provider may support SSO with third-party IdPs easily, while another may have limited configuration options.

6. Customer and Enterprise Role

- Enterprises play a critical role in **managing identity and access** in the SaaS environment.
- They should **leverage their existing Identity Provider (IdP)** systems (e.g., Active Directory, Okta) to integrate with SaaS applications for **centralized authentication and policy control**.
- Organizations are encouraged to **demand IAM standards** such as:
  - **SAML (Security Assertion Markup Language)** – for SSO
  - **SPML (Service Provisioning Markup Language)** – for automated provisioning/deprovisioning of users
  - **Open APIs** – for seamless integration between enterprise IAM and cloud applications
- This ensures **consistent, secure, and manageable user access** across multiple SaaS platforms.

Access Control: PaaS

1. CSP Responsibility

- The **Cloud Service Provider (CSP)** manages **access control** for the **underlying infrastructure** — including **networks, servers, and the PaaS platform itself**.
- This ensures that the platform runs securely and reliably, without customer interference in low-level system management.

2. Customer Responsibility

- The **customer** (application developer or organization) is responsible for **managing access control to the applications** they deploy on the PaaS platform.
- This includes setting up **end-user access management** for their own users (e.g., customers, employees).

3. Application-Level Access Control

- Customers handle access mechanisms like **user provisioning** (creating accounts, assigning roles) and **authentication** (verifying identity).
- Examples: setting up login systems, managing API keys, or integrating with identity management systems.

4. Variation in Access Control Support

- **PaaS providers differ** in how much built-in user access control they offer.
- Some platforms provide only **basic or limited access features**, requiring customers to implement their own security solutions.

Access Control: IaaS

•In **IaaS**, the **customer** is fully responsible for **managing access control** to all their cloud resources.

•Customers must design and manage access for **virtual servers, networks, storage, and hosted applications**.

•Access control in IaaS is divided into **two main categories**:

1.CSP Infrastructure Access Control – Managed by the **Cloud Service Provider** for its **host systems, networks, and management tools**.

2.Customer Virtual Infrastructure Access Control – Managed by the **customer** for their **VMs, virtual networks, storage, and hosted applications**.

CSP Infrastructure Access Control

- **CSP** manages access control for the **administrative network** used for internal management tasks.
- Covers access to **administrative processes** such as:
  - Backups
  - Host (hypervisor) and network maintenance
  - Router and firewall policy management
  - System monitoring and management
- **Strong authentication** and **role-based access control (RBAC)** must protect administrative functions.
- **Operational procedures** should ensure proper **provisioning and revocation** of admin privileges.
- **Periodic audits** and **administrative user certifications** must be conducted to ensure **least privilege** and **separation of duties**.

Customer virtual infrastructure access control

- Understanding Resources & Controls:
  - Customers must understand all virtual components (network, host, firewall, load balancer, management console, etc.) and their access protection mechanisms.
  - CSPs often provide **full root or administrative access** to virtual servers.
- Customer Responsibility:
  - Customers must implement **access restrictions and secure virtual resources** (servers, networks, storage, etc.).

1. **Network Access Control**
2. **Virtual Server Access Control**
3. **Cloud Management Station**
4. **Web-Based Console**

1. Network Access Control

- Customers manage **network-level access** to their virtual servers **using firewall rules and security groups.**
- Default settings **usually deny all incoming traffic, so users must explicitly configure rules to allow specific connections** (e.g., SSH access only from trusted IPs).
- Example: Allowing access from IP 192.168.0.1 to the virtual server 10.0.0.1 on port 22 for SSH.

2. Virtual Server Access Control

- Protect virtual servers using **OS-level authentication mechanisms** (Linux, Solaris, Windows).
- Use **SSH-based logins** with **strong authentication** to prevent attacks (IP spoofing, MITM, DNS spoofing, etc.).
- Recommended methods:
  - **RSA host authentication** or pure RSA
  - **One-time passwords (S/Key)**
  - **Kerberos authentication**
- **Store RSA keys securely** (e.g., encrypted media with passphrase protection).

3. Cloud Management Station

- Cloud resources are managed remotely using a **client system** that communicates through **CSP-proprietary APIs** such as REST, SOAP, or HTTP with XML/JSON.
- A **client management toolkit** (provided by the CSP) is installed on the management station to interact with the cloud's management services.
- The management station holds **sensitive data**, including host keys, user credentials, and firewall policies.
- It acts as the **command and control center** for managing cloud infrastructure.
- Therefore, access must be protected using **strong authentication** and **strict access control procedures**.

- **A software suite that provides a unified console to manage, monitor, and optimize cloud resources across multiple cloud environments.**

4. Web-Based Console

- Provides an **alternative interface** to manage cloud infrastructure.
- Offers control over **virtual servers, networks, and security configurations**.
- Must be protected as it gives **high-level administrative access**.
- Access should be **restricted to HTTPS** connections and authorized users only.

Security Vulnerability, Patch, and Configuration Management

1. INTRODUCTION

- Cloud systems face major risks from **malware** or attackers exploiting vulnerabilities in **infrastructure, networks, and applications.**
- This risk is **higher in PaaS and IaaS models**, where customers have more control — and therefore, more **security responsibilities**.
- Because cloud environments are **multi-tenant**, the **security of one tenant affects others**.
- **Customers must understand their responsibilities** and **demand transparency from CSPs** to plan complementary security measures.

2. Shared Responsibilities

- **CSP Responsibilities:**
  - Manage **vulnerability, patch, and configuration (VPC)** of the infrastructure components they operate **(networks, hosts, applications, storage).**
  - Include **third-party services** they depend on.
  - Maintain internal **VPC programs** aligned with **ISO/IEC 27002 controls** for technical assurance.
- **Customer Responsibilities:**
  - Manage vulnerabilities and configurations for **systems and applications they deploy or control**.
  - Understand their **VPC boundaries and ensure end-to-end security (**especially for customer-managed systems).
  - Extend their **internal security management** to cover cloud-based components.

Security Vulnerability Management

Definition & Objective

- Vulnerability management is a **proactive process** to detect and mitigate weaknesses in **hosts, networks, and applications** before attackers can exploit them.
- Based on **ISO/IEC 27002**, the **objective** is:
- **“To reduce risks resulting from exploitation of published technical vulnerabilities.”**
- Requires a **systematic, repeatable, and measurable process**, including:
  - Regular **vulnerability scans**.
  - **Risk assessment** for each finding.
  - **Remediation** through patching or configuration updates.
- **Shared Role**
- **Both CSPs and customers** share responsibility depending on the **cloud model (SPI)**:
  - **SaaS: CSP handles most vulnerability management.**
  - **PaaS: Shared — CSP manages platform, customer manages app-level vulnerabilities.**
  - **IaaS: Customer handles VM, OS, and app vulnerabilities.**

Security Patch Management

Purpose

- Patch management protects systems from attacks exploiting **known vulnerabilities**.
- It follows a **change management process**, often triggered by vulnerability scan results.
- Regular patching reduces **insider and outsider threats**.

Cloud Model	Who Patches?	Scope of Responsibility
SaaS	CSP	CSP patches infrastructure, OS, and applications delivering the service.
PaaS	Shared	CSP patches platform; customer patches their deployed applications.
IaaS	Customer	Customer patches entire software stack (OS, middleware, apps, DB).

Security Patch Management

- **Purpose —** Security patch management protects systems, network devices, and applications from exploitation of known vulnerabilities.
- **Process —** It follows a change management framework and works in coordination with the vulnerability management program.
- **Risk Reduction —** Regular patching helps mitigate both insider and outsider threats.
- **Provider Role —** In SaaS, the service provider handles all patching of firmware and software used to deliver the service.
- **Customer Responsibility —** In PaaS and IaaS, customers are responsible for patching their applications (PaaS) and the entire software stack (IaaS).

Steps:

- **Identify** – Detect systems and applications that need updates.
- **Evaluate** – Assess the importance or severity of each patch (e.g., critical security patch).
- **Test** – Check patch compatibility in a controlled environment before deployment.
- **Deploy** – Install the patch across systems and applications.
- **Verify** – Ensure patches are properly installed and systems function correctly.
- **Document** – Record patching activities for compliance and future reference.

Security Configuration Management

- It protects **hosts, network devices, and applications** from attacks exploiting configuration weaknesses.
- It is closely linked to vulnerability management and forms part of overall IT configuration management.
- Involves securing and monitoring system, network, and application configurations, including OS, firewall, and storage settings.
- In SaaS and PaaS, the service provider manages configuration security of the underlying platform.
- In IaaS, customers must secure and manage configurations of their operating systems, databases, and applications.

SaaS VPC Management

- SaaS VPC management covers **vulnerability management, security patching, and system configuration** for both CSP-managed and customer-connected infrastructures.
- Since SaaS applications are accessed over the Internet via browsers, securing user endpoints is critical.
- The VPC management program must include endpoint security requirements tailored to the organization’s environment.
- Companies typically maintain standard OS images for employee computers to ensure consistent security configurations.
- Standard OS images usually include antivirus, anti-malware, firewall, and centralized automatic patch management.

PaaS VPC Management

- Focuses on managing vulnerabilities, patching, and configurations in both CSP-managed and customer infrastructures.
- Includes endpoint devices like PCs, virtual desktops, and mobiles that access the PaaS service.
- The VPC management program should include endpoint protection measures.

PaaS Provider Responsibilities

- The PaaS provider (CSP) manages VPC for infrastructure operated by them and by third-party services they depend on.
- Responsibilities are similar to those of a SaaS provider.

PaaS Customer Responsibilities

- Customers manage **VPC for their own applications deployed on the PaaS platform.**
- Must handle vulnerabilities or configuration weaknesses in their applications.
- Treat PaaS applications like **standard apps in a private data center**.
- Vulnerabilities can occur due to design flaws, coding errors, or poor authentication and privilege settings.
- Third-party web services may also introduce vulnerabilities outside customer control.

Customer Actions for Security

- Work with the PaaS vendor or third-party providers to fix vulnerabilities in applications.
- Understand vendor policies, SLAs, and vulnerability disclosure methods.
- Follow **Software Development Life Cycle (SDLC)** best practices to reduce risks.

Standard Security Practices

- **White-box testing —** Analyze source code for issues like buffer overflows using tools (e.g., Ounce Labs, Fortify).
- **Black-box testing —** Test without source code to find issues like SQL injection, XSS, or cookie poisoning.
- **Penetration testing —** Simulate real attack scenarios with vendor approval to detect vulnerabilities.
- **Vulnerability alerts —** Stay updated on new risks through PaaS provider alerts, RSS feeds, or web portals.

IaaS VPC Management

- IaaS VPC management focuses on **the CSP-managed infrastructure**, as well as the **customer infrastructure interfacing with the IaaS service.**
- IaaS VPC management diverges from SaaS and PaaS in that the infrastructure delineation, network boundary between customers, and CSP infrastructure are blurred.
- For each layer of infrastructure (network, host, storage), **the customer and CSP have responsibilities in managing VPC in the respective layers from their perspective (i.**e., the CSP is responsible for the common CSP infrastructure available to all customers, and the customer is responsible for the virtual infrastructure available to the customer for the duration of use).
- Hence, **a VPC management program should address both the common and shared infrastructures.**

IaaS provider responsibilities

- Scope — The IaaS CSP manages **VPC for the infrastructure it owns and operates.**
- Third-Party Reliance — The CSP is also responsible for VPC management of third-party systems and services it depends on.
- Infrastructure Coverage — Includes systems, networks, hosts (hypervisors), storage, and applications under CSP control.
- Management Tools — The web console or **management station** used by customers to **control virtual infrastructure** is part of the VPC scope.
- Employee Devices — CSP must secure personal computers and devices used by its employees and contractors.