Cloud Computing Fundamentals
IN401 – M1S1 – 5 Credits

Course Grading

- Partial Exam
- Final Exam
- Second Session

Course Contents

- Chapter 1: Introduction to Cloud Computing
- Chapter 2: Infrastructure as a Service (IaaS)
- Chapter 3: Platform as a Service (PaaS)
- Chapter 4: Software as a Service (SaaS)
- Chapter 5: Cloud Native Technologies & Architectures
- **Chapter 6: Cloud Security & Best Practices**
- Chapter 7: Practical Case Studies

Chapter 6:

Cloud Security & Best Practices

Content

- Cloud security basics (security groups, IAM)
- Data privacy compliance (GDPR, data residency)
- Best practices for secured deployments (Identity/access mgmt., key management)

Cloud Security Introduction

Introduction

- Cloud security is a specialized discipline within information security that addresses the protection of data, applications, and infrastructure in cloud computing environments.

- It combines traditional security principles with cloud-specific mechanisms such as virtualization, automation, and shared control planes.

- Cloud security is **not simply traditional security moved to the cloud**. Instead, it introduces new threat models, operational assumptions, and architectural patterns.

Core Security Objectives (CIA Triad)

- **Confidentiality**: Prevent unauthorized data disclosure.
- **Integrity**: Ensure data and system accuracy.
- **Availability**: Maintain continuous access to services.

Cloud Service Models and Security Responsibilities

- **IaaS Customer Responsibilities**:
- Operating system hardening and patching
- Firewall and security group configuration
- Application security
- Data protection
- **IaaS Provider Responsibilities**:
- Physical security of data centers
- Hardware lifecycle management
- Hypervisor security
- **IaaS Main Drawbacks**:
- VM vulnerabilities and misconfigurations
- Network isolation failures
- Privilege escalation within virtualized environments

Cloud Service Models and Security Responsibilities

- **PaaS Customer Responsibilities**:
- Secure application code
- Identity and access configuration
- Data governance
- **PaaS Provider Responsibilities**:
- Runtime, middleware, OS patching
- Infrastructure availability
- **PaaS Main Drawbacks** :
- API exposure risks
- Dependency vulnerabilities
- Secure application design

Cloud Service Models and Security Responsibilities

- **SaaS Customer Responsibilities**:
- User access management
- Data classification and usage
- Endpoint security
- **SaaS Provider Responsibilities**:
- Application security
- Infrastructure and platform security
- **SaaS Main Drawbacks :**
- Identity-centric security
- Insider threats
- Data leakage risks

Cloud Security Groups

Cloud Network Security

- Cloud network security is software-defined and policy-driven.
- Unlike traditional networks that rely on physical firewalls and appliances, cloud networking uses logical controls enforced by the cloud provider.

- Network security must be done as **identity-aware, segmented, and highly automated**.

Security Groups (Virtual Firewalls)

- Security Groups are **stateful, virtual firewalls** that control inbound and outbound traffic to cloud resources such as virtual machines, load balancers, and databases.

- They operate at the **instance or resource level**, not at the subnet level, and are enforced by the cloud provider’s infrastructure.

Core Characteristics of Security Groups

- **Stateful**: If inbound traffic is allowed, the corresponding outbound response is automatically permitted (and vice versa).
- **Allow-rules only**: Security groups typically do not support explicit deny rules.
- **Resource-attached**: Applied directly to instances or interfaces.
- **Evaluated as a whole**: All rules are evaluated together, not in sequence.

Inbound and Outbound Rules

- **Inbound Rules**:
- Define what traffic is allowed *into* a resource
- Common examples: HTTP (80), HTTPS (443), SSH (22)

- **Outbound Rules**:
- Define what traffic a resource can initiate
- Often overly permissive by default
- Each rule specifies:
  - Protocol (TCP, UDP, ICMP)
  - Port range
  - Source or destination (IP range, CIDR, or another security group)

Referencing Security Groups

- A powerful feature of security groups is the ability to **reference other security groups** instead of IP addresses.

- **Example Scenario**:
- Web tier security group allows inbound traffic from load balancer security group
- Application tier allows traffic only from web tier
- Database tier allows traffic only from application tier

- This enables **dynamic, identity-based network security**.

Security Groups vs Network ACLs

Feature	Security Groups	Network ACLs
Scope	Resource-level	Subnet-level
State	Stateful	Stateless
Rules	Allow only	Allow and deny
Evaluation	All rules together	Ordered rules

Common Misconfigurations and Risks

- Allowing 0.0.0.0/0 on administrative ports (SSH, RDP)
- Overly permissive outbound rules
- Reusing security groups across unrelated workloads
- Lack of documentation and naming standards

Best Practices

- Start with deny-all inbound rules
- Restrict administrative access by IP and role
- Use security group references instead of CIDRs
- Apply least privilege to outbound traffic
- Regularly audit and review rules

Identity and Access Management (IAM)

Identity and Access Management (IAM)

- IAM is the cornerstone of cloud security and effectively replaces the traditional network perimeter.
- In cloud environments, **identity becomes the new security boundary**, governing access to all resources through centrally managed policies.

Fundamental Concepts

- **Identity**: A uniquely identifiable entity that can authenticate and be authorized (human users, applications, virtual machines, containers, serverless functions).

- **Authentication (AuthN)**: The process of verifying an identity (passwords, certificates, tokens, MFA).

- **Authorization (AuthZ)**: The process of determining what an authenticated identity is allowed to do.

- **Accounting/Auditing**: Recording actions for traceability and compliance.

Types of Identities in the Cloud

- **Human Identities**:
- Administrators
- Developers
- End users

- **Non-Human (Machine) Identities**:
- Virtual machines and instances
- Containers and Kubernetes service accounts
- Serverless functions
- CI/CD pipelines and automation tools

- Modern cloud breaches often involve **compromised machine identities**, not human users.

Access Control Models

- **Role-Based Access Control (RBAC)**:
- Permissions grouped into roles
- Roles assigned to identities
- Widely used in Kubernetes and cloud IAM

- **Attribute-Based Access Control (ABAC)**:
- Decisions based on attributes (user, resource, environment)
- More flexible but complex

- **Policy-Based Access Control**:
- JSON/YAML policies evaluated by an authorization engine
- Explicit allow and implicit deny logic

IAM Threats and Attack Vectors

- Credential leakage
- Privilege escalation
- Lateral movement
- Token theft

IAM Best Practices

- Enforce MFA everywhere
- Use roles instead of static credentials
- Apply least privilege by default
- Monitor and audit IAM activity
- Automate identity lifecycle management

Data privacy compliance

Introduction

- Data privacy compliance is a critical aspect of cloud security, especially in environments that process personal, sensitive, or regulated data.

- This topic should be done at the intersection of **law, technology, and architecture**.

Data Privacy vs Data Security

- **Data Security**: Technical and organizational controls that protect data from unauthorized access, breaches, and loss.

- **Data Privacy**: Legal and ethical rules governing how personal data is collected, processed, stored, shared, and retained.

- *Security is a prerequisite for privacy, but security alone does not guarantee compliance.*

Personal Data and Data Classification

- **Types of Data**
- **Personal Data**: Any data that can identify an individual directly or indirectly (name, email, IP address).
- **Sensitive Personal Data**: Health data, biometric data, financial data.
- **Regulated Data**: Data subject to specific legal frameworks (payment data, medical records).

- **Data Classification in the Cloud**
- Common classification levels: - Public - Internal - Confidential - Restricted / Regulated

- Therefore, classification drives encryption, access control, logging, and residency decisions.

Cloud Provider and Customer Responsibilities

- Under the shared responsibility model:
- **Cloud Provider** ensures compliant infrastructure and certifications
- **Customer** ensures compliant data usage, access, and processing

- Compliance failures are almost always on the customer side.

Privacy-by-Design and Privacy-by-Default

- **Privacy-by-Design Principles**
- Proactive not reactive
- Embedded into architecture
- End-to-end security

- **Privacy-by-Default**
- Minimal data collection
- Restricted access by default

Compliance Automation and Governance

- **Governance Mechanisms**
- Policy as Code
- Automated compliance checks
- Configuration baselines

- **Continuous Compliance**
- Real-time posture monitoring
- Drift detection
- Evidence collection

- Cloud compliance is **continuous**, not audit-driven.

Privacy Risks and Common Violations

- Over-collection of personal data
- Excessive data retention
- Unrestricted internal access
- Cross-border data transfer without safeguards
- Teaching emphasis: many violations occur without any external attacker.

Data Privacy Best Practices

- Strong IAM and least privilege
- Encryption at rest and in transit
- Tokenization and anonymization
- Logging and audit trails
- Data loss prevention (DLP)
- Map each technical control to regulatory requirements.

Data Residency

Data Residency

- **Data residency** refers to the **geographical location where data is physically stored and processed**.

- In cloud computing, this means the specific **country or region** in which a cloud service provider (CSP) stores customer data in its data centers.

- Data residency is distinct from:
- **Data sovereignty**: Legal authority of a country over data.
- **Data localization**: Mandatory requirement that data must remain within a country.

Legal and Regulatory Compliance

- Many laws require certain categories of data to remain within national borders:
  - **GDPR (EU)** – personal data transfer restrictions
  - **HIPAA (USA)** – healthcare data protections
  - **PCI DSS** – payment card data
  - **Banking and financial regulations** – customer and transaction data

- Failure to comply may result in:
  - Regulatory fines
  - Loss of certifications
  - Service suspension
  - Legal liability

Data Privacy and User Trust

- Data residency ensures:
- Compliance with local privacy laws
- Reduced exposure to foreign surveillance laws
- Increased trust for citizens and enterprises

- Example: EU customers often require assurance that their personal data never leaves the EU.

Jurisdiction and Government Access

- When data is stored in a country:
- It is subject to **local laws**
- Governments may request access through lawful processes

- Example: **US CLOUD Act** allows U.S. authorities to request access to data from U.S.-based providers, even if stored abroad.

Data Residency in Cloud Architectures

- **Cloud Regions and Availability Zones**
- Cloud providers divide infrastructure into:
  - **Regions** (e.g., EU-West, US-East)
  - **Availability Zones** (isolated data centers within a region)

- Key principle: Data remains within the selected region unless explicitly moved.

Data Residency in Cloud Architectures

- **Multi-Region and Global Services**
- Some cloud services are **global by default**:
  - Identity services
  - DNS
  - Content Delivery Networks (CDNs)
  - Monitoring and logging services

- Risk: Metadata or logs may cross borders unintentionally.

Risks and Challenges

- **Operational Complexity**
- Limited region choices
- Higher latency for global users
- Increased cost for regional redundancy

- **Misconfiguration Risks**
- Cross-region logging
- Global identity services storing metadata
- Third-party SaaS integrations violating residency rules

Data Residency Best Practices

- Select compliant cloud regions early
- Understand regulatory obligations
- Restrict cross-border data flows
- Use encryption and key residency
- Monitor and audit continuously
- Educate development teams
- Review CSP compliance reports
- Track data flows
- Maintain data location documentation

- **Tools**:
  - Cloud security posture management (CSPM)
  - Data loss prevention (DLP)
  - Cloud audit logs

Key Management

Introduction

- **Key Management** refers to the **secure generation, storage, distribution, rotation, use, and destruction of cryptographic keys** used to protect data at rest, in transit, and in use.

- In cloud environments, key management is typically implemented through **Key Management Services (KMS)** or **Hardware Security Modules (HSMs)**.

Types of Cryptographic Keys

- **1 Symmetric Keys**
- Same key for encryption and decryption
- High performance
- Used for data encryption (AES)

- **2 Asymmetric Keys**
- Public/private key pairs
- Used for identity, TLS, digital signatures
- Slower but more flexible

- **3 Data Encryption Keys (DEK) vs Key Encryption Keys (KEK)**

Key Management Lifecycle

1. Key Generation

- Keys generated by CSP or customer
- Should use strong entropy sources
- FIPS 140-2/3 compliant algorithms

2. Key Storage

- Stored in:
  - Cloud KMS
  - Dedicated HSMs
- Keys never exposed in plaintext

Key Management Lifecycle

3. Key Usage

- Keys used via APIs
- Applications never directly access raw key material
- Strict access control enforced

4. Key Rotation

- Periodic replacement of keys
- Limits impact of key compromise
- Can be automatic or manual
- Rotate DEKs frequently
- Rotate KEKs on a defined schedule

Key Management Lifecycle

5. Key Revocation and Deletion

- Immediate disablement in case of compromise
- Secure destruction
- Irreversible once deleted

Common Risks and Misconfigurations

- Over-privileged access to keys
- Keys shared across environments
- No rotation policy
- Storing keys in application code
- Using global KMS for regulated data

Key Management Best Practices

- Use envelope encryption
- Prefer customer-managed keys for sensitive data
- Store keys in the same region as data
- Enforce least privilege and MFA
- Rotate and audit keys regularly
- Use HSM-backed keys for high-risk workloads