Cloud Networking

1. Resource Monitoring Techniques

Resource monitoring helps track the usage and performance of system resources like CPU, memory, disk, and network. It's essential for identifying bottlenecks and ensuring system health.

Common Techniques:

- **Command-line tools:**
  - **Linux:**
    - top, htop – Monitor CPU and memory usage.
    - vmstat – Virtual memory stats.
    - iostat – Disk I/O stats.
    - netstat, iftop, nload – Monitor network traffic.
  - **Windows:**
    - **Task Manager** – View running processes and performance.
    - **Resource Monitor** – Detailed info on CPU, memory, disk, network.
    - perfmon – Performance Monitor for custom tracking.
- **Cloud-native monitoring:**
  - **AWS CloudWatch** – Monitor resources on AWS.
  - **Azure Monitor** – Tracks usage and performance metrics.
  - **Google Cloud Operations (formerly Stackdriver)** – Logging and monitoring on GCP.
- **Third-party tools:**
  - **Nagios**, **Zabbix**, **Prometheus**, **Grafana** – Used for advanced monitoring and visualization.
  - **Datadog**, **New Relic**, **SolarWinds** – Cloud and infrastructure monitoring solutions.

---

2. How to Access Compute (Windows and Linux) from Internet? Tools & Security

To manage remote compute instances over the internet, you need remote access tools and proper security measures.

For Windows:

- **Tool:** Remote Desktop Protocol (RDP)

    - Access via: mstsc command or Remote Desktop App
- **Port:** TCP **3389**
- **Security:**
    - Use strong passwords and MFA
    - Limit IP addresses using firewall rules
    - Use VPN or Bastion Host for added protection
    - Disable RDP when not in use

For Linux:

- **Tool:** Secure Shell (SSH)
    - Command: ssh user@IP_address
- **Port:** TCP **22**
- **Security:**
    - Use SSH key-based authentication (disable password login)
    - Change default SSH port (optional)
    - Use firewall (e.g., UFW, iptables) to restrict access
    - Enable fail2ban or similar tools to block brute-force attacks
    - Use a **jump host** or **bastion host** for indirect access

---

3. Encryption Technologies and Methods

Encryption is used to protect data confidentiality during storage (at rest) or transmission (in transit).

Types of Encryption:

- **Symmetric Encryption (Same key for encryption & decryption):**
    - Fast but less secure for large systems.
    - Algorithms: **AES**, **DES**, **RC4**
- **Asymmetric Encryption (Public & private keys):**
    - Used in secure communications and SSL/TLS.
    - Algorithms: **RSA**, **ECC**
- **Hashing (One-way encryption for integrity):**
    - Algorithms: **SHA-256**, **MD5**, **SHA-1** (deprecated)

Encryption in Cloud:

- **At Rest:** Data is encrypted while stored (e.g., AWS S3 default encryption).
- **In Transit:** Data is encrypted during transmission using SSL/TLS.
- **Key Management:** Use of KMS (Key Management Services) to manage encryption keys securely.

---

4. Describe Network Security in Cloud, Compute Security, and Storage Security

Network Security in Cloud:

- Use **Virtual Private Cloud (VPC)** or equivalent.
- Control traffic using **security groups**, **firewalls**, and **NACLs (Network ACLs)**.
- Use **VPN** or **private endpoints** to avoid public internet exposure.
- Apply **DDoS protection** (e.g., AWS Shield, Azure DDoS Protection).

Compute Security:

- Harden OS (disable unused ports/services).
- Keep instances patched and updated.
- Use secure access (SSH keys, RDP with MFA).
- Monitor and log all access.
- Install antivirus or endpoint protection.

Storage Security:

- Encrypt data at rest and in transit.
- Use **access control policies** (e.g., IAM roles/policies in AWS).
- Enable versioning and backups.
- Use **Object Lock** (e.g., in AWS S3) to prevent deletion.
- Audit and monitor access logs.