

| SNA Theory (Finals) - Jaish Khan

contains all topics from Week 8 till Week 12 of Systems and Network Administration Theory.

Table of Contents

- [8. Cloud Computing](#)
 - [8.1. Key Characteristics](#)
 - [8.2. Types of Cloud Computing Services](#)
 - [8.4. Cloud Deployment Models](#)
 - [8.5. Cloud Services and Providers](#)
- [9. Backup and Recovery Strategies](#)
 - [9.1. Types of Backups](#)
 - [9.2. Backup Strategies](#)
 - [9.3. Implementation Methods](#)
 - [9.4. Disaster Recovery Planning and Testing](#)
- [10. Network Troubleshooting and Performance Tuning](#)
 - [10.1. Common Network Issues](#)
 - [10.2. Diagnostic and Troubleshooting Tools and Techniques](#)
 - [10.2.1. Troubleshooting Tools](#)
 - [10.2.2. Troubleshooting Techniques](#)
 - [10.3. Network Performance Tuning](#)
 - [10.3.1. Performance Metrics](#)
 - [10.3.2. Performance Tuning](#)
- [11. Security Policies and Compliance](#)
 - [11.1. Security Standards and Frameworks](#)
 - [11.2. Implementing and Enforcing Security Policies](#)
 - [11.3. Compliance Auditing and Reporting](#)
- [12. Advanced System Administration](#)
 - [12.1 High Availability Systems](#)
 - [12.2. Load Balancing and Failover Strategies](#)

*The "easier explanation" sections are ChatGPT's wording and are just there for concepts. **Don't memorize** them.*

8. Cloud Computing

The delivery of computing services over the internet (the "cloud"). These services include storage, databases, servers, networking, software etc.

8.1. Key Characteristics

1. **On-demand Self-service** → Users can provision resources as needed without human intervention from the service provider.
2. **Broad Network Access** → Services are accessible over the network and support a variety of devices (e.g., mobile phones, laptops).
3. **Resource Pooling** → Cloud providers pool resources to serve multiple consumers using a multi-tenant model.
4. **Rapid Elasticity** → Resources can scale up or down as needed.
5. **Measured Service** → Resource usage can be monitored, controlled, and reported, offering transparency for both provider and consumer.

Easier Explanation

Think of cloud computing like renting a house. You only pay for what you use (on-demand), you can access it from anywhere (broad access), many people can rent from the same company (resource pooling), you can easily upgrade to a bigger house or downgrade to a smaller one (elasticity), and you get a bill showing exactly what you used (measured service).

8.2. Types of Cloud Computing Services

Service Type	Description	Examples
IaaS (Infrastructure as a Service)	Virtualized computing resources over the internet. Users rent infrastructure like servers, virtual machines (VMs), storage, networks, and operating systems.	<ul style="list-style-type: none">• AWS EC2• Google Compute• Azure VMs
PaaS (Platform as a Service)	Provides a platform allowing users to develop, run, and manage applications without dealing with infrastructure management.	<ul style="list-style-type: none">• App Engine• Elastic Beanstalk• Azure Apps
SaaS (Software as a Service)	Delivers software applications over the internet eliminating the need for installing and running applications on individual computers.	<ul style="list-style-type: none">• Salesforce• Microsoft 365• Google Workspace

Easier Explanation

Cloud services come in three main types:

- IaaS (is like renting an empty apartment) - you get the basics and set up everything else yourself
- PaaS (is like renting a furnished apartment) - the basics are already set up for you
- SaaS (is like staying at a hotel) - everything is managed for you

Other Service Models

Model	Description	Examples
FaaS (Function as a Service)	Allows users to execute code in response to events without the complexity of building and maintaining infrastructure.	- AWS Lambda, - Azure Functions
DaaS (Database as a Service)	Databases provided as managed services in the cloud.	- Amazon RDS, - Azure SQL, - Google Cloud BigTable

| 8.4. Cloud Deployment Models

Model	Characteristics	Benefits	Best For
Public	Third-party owned, shared resources	<ul style="list-style-type: none">• Cost-effective• Scalable• Low maintenance	General computing, Web apps
Private	Single organization, dedicated resources	<ul style="list-style-type: none">• Maximum control• Better security• Customization	Sensitive data, Compliance
Hybrid	Combined public/private	<ul style="list-style-type: none">• Flexibility• Cost optimization• Risk management	Mixed workloads

Model	Characteristics	Benefits	Best For
Community	Shared by similar organizations	<ul style="list-style-type: none"> • Cost sharing • Common compliance • Collaborative 	Healthcare, Government

Easier Explanation

Think of cloud deployments like different types of gyms:

- Public cloud is like a public gym - anyone can join, pay as you go
- Private cloud is like a personal home gym - only you can use it
- Hybrid cloud is like having both gym memberships - use what works best
- Community cloud is like a gym for a specific group (e.g., employees only)

| 8.5. Cloud Services and Providers

Major Providers → Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP), Others (IBM Cloud, Oracle Cloud, Alibaba Cloud etc...)

Cloud Service Categories

- Computing → Virtual machines, serverless functions, containers.
- Storage → Block storage, object storage, file storage.
- Databases → SQL, NoSQL, in-memory databases.
- Networking → Virtual networks, load balancers, CDNs.
- AI/ML → Machine learning models, natural language processing, computer vision.
- Security → Identity management, encryption, firewall solutions.
- DevOps Tools → CI/CD tools, code repositories, testing environments.

Tools

1. Docker Desktop (containerization)
2. Docker Compose/Kubernetes (container management)
3. Ansible/Chef/Puppet (configuration management)
4. Terraform (infrastructure management)
5. CloudSim (cloud simulation)
6. Vagrant (VM provisioning)

| 9. Backup and Recovery Strategies

| 9.1. Types of Backups

1. **Full Backup** → *Complete data copy* (only data).
2. **Incremental Backup** → *Changes since last backup.*
3. **Differential Backup** → *Changes since last full backup.*
4. **Mirror Backup** → *Exact source replica* (includes every single thing).
5. **Snapshot** → *System state capture.*

Easier Explanation

Backups are like taking photos of your important documents:

- Full backup is like taking a photo of everything
- Incremental backup is like only photographing new or changed documents
- Differential backup is like photographing everything that changed since your last complete photo set
- Snapshot is like taking a quick picture of how everything looks right now

| 9.2. Backup Strategies

| **3-2-1 Rule** → Maintain 3 copies of data, on 2 different media, with 1 copy off-site.

Backup Types by Priority

- *Cold backups* (offline | for low-priority data)
- *Warm backups* (accessible | for medium-priority data)
- *Hot backups* (real-time | for high-priority data)

Backup Frequency → After how long are backups done-

Backup Retention → How long a backup is stored-

| 9.3. Implementation Methods

1. **Local Backups** → Done on *Physical* (directly attached) storage like external hard drives.
 - Pros: Fast backup and recovery, No need for internet, Complete data control.
 - Cons: Physical maintenance required, Vulnerable to local disasters, Hard to scale.
2. **Remote Backups** → Done on *Off-site* storage in some other location like data centers.

- Pros: Protection against local disasters.
 - Cons: Slower recovery times, logistical challenges.
3. **Cloud Backups** → Done on *Cloud* storage like AWS S3, GCloud Storage, Azure Blob.
- Pros: Easy to scale, Automatic and Accessible from everywhere.
 - Cons: Needs internet, Has monthly subscription cost.

| 9.4. Disaster Recovery Planning and Testing

Importance: Ensures business continuity during unexpected events. Minimizes downtime and data loss.

Steps in Disaster Recovery Planning

1. *Risk Assessment*: Identify potential risks and vulnerabilities.
2. *Define Recovery Objectives*:
 1. Recovery Point Objective (RPO) → Maximum acceptable data loss.
 2. Recovery Time Objective (RTO) → Maximum acceptable downtime.
3. *Make a Disaster Recovery Plan* (DRP): Identify critical systems and data. Specify backup and restoration procedures. Assign roles and responsibilities.

Testing DRP → Simulated discussion of the DRP (*Tabletop Exercise*), Step-by-step review of recovery procedures (*Walkthrough Test*), Actual recovery simulation (*Live Drill*).

10. Network Troubleshooting and Performance Tuning

10.1. Common Network Issues

1. Connectivity Problems → Faulty Cables
2. Bandwidth Issues → Low Bandwidth, Network Loss
3. Software Configuration Errors → Outdated Drivers, Main Terminal Failed
4. Hardware Failures → Cache Overload
5. Security Threats

Root-Cause Analysis → Use the OSI Layers to identify and compare causes with symptoms.

If there's an attack; To check the level/scope/size of attack → Count the number and Check the physical/geographical range of affected devices. The solution to these attacks is to check the equipment closest to you (proximity) to get an idea about the attack. Also, immediately disconnect affected devices with the rest of the network to avoid other devices getting affected.

Easier Explanation

Network troubleshooting is like being a detective:

- You use different tools (like Ping, Traceroute) to find clues
- Each tool tells you something different about the network's health
- Performance metrics are like vital signs (heartbeat, temperature) but for your network

10.2. Diagnostic and Troubleshooting Tools and Techniques

10.2.1. Troubleshooting Tools

There are two types of troubleshooting tools:

- **Physical Tools** → Equipment like *LAN Tester, Voltage Equipment, Splicing Tools, Clamps* etc...
- **Software Tools** → Tools like ping, traceroute, snmp, nmap, netstat, tcpdump, nslookup, dig etc.

Tool	Primary Use	Best For
Ping	Connectivity testing	Basic connectivity

Tool	Primary Use	Best For
Traceroute	Path analysis	Network mapping
SNMP	Device monitoring	Network management
Nmap	Network scanning	Security assessment

There are also "Third-party applications" like **Wireshark** are also used for deep packet inspection and traffic analysis.

10.2.2. Troubleshooting Techniques

1. *Baseline Analysis* → Compare current system or network performance to historical data (the baseline).
2. *Port and Protocol Analysis* → Check network traffic based on ports and protocols used by different applications and services.
3. *Simulation* → Simulate network conditions in a controlled environment.
4. *Event Correlation* → Analyzing logs and events from multiple sources.

10.3. Network Performance Tuning

10.3.1. Performance Metrics

Metric	Definition	Impact Factors	Optimization Methods
Latency	Data travel time	<ul style="list-style-type: none"> • Network distance • Processing delays 	<ul style="list-style-type: none"> • CDN usage • Edge computing
Bandwidth	Data capacity	<ul style="list-style-type: none"> • Network infrastructure • Congestion 	<ul style="list-style-type: none"> • Traffic shaping • Load balancing
Jitter	Delay variation	<ul style="list-style-type: none"> • Network congestion • Route changes 	<ul style="list-style-type: none"> • QoS policies • Buffer management
Packet Loss	Failed transmission	<ul style="list-style-type: none"> • Network errors • Congestion 	<ul style="list-style-type: none"> • Error correction • Path optimization

Easier Explanation

Network performance is like traffic on a road:

- Latency is like the time it takes to drive from A to B
- Bandwidth is like how many lanes the road has
- Jitter is like inconsistent traffic speeds

- Packet loss is like cars not reaching their destination

10.3.2. Performance Tuning

1. System-Level Tuning

- *Optimize Hardware* → Upgrade hardware components into better ones.
- *Update Firmware/Software* → Keep system firmware and software up to date.
- *Load Balancing* → Distribute network traffic across multiple servers to prevent any single server from becoming overloaded.
- *Quality of Service* → Prioritize specific types of network traffic (e.g., voice, video, or business-critical applications) over others.

2. Network-Level Tuning

- *Traffic Shaping* → Controls the flow of network traffic to manage bandwidth usage and prevent congestion.
 - *Segmentation* → Dividing a network into smaller, isolated segments.
 - *Caching* → Storing frequently accessed data in a cache (a temporary storage location) closer to users.
 - *DNS Optimization* → Configuring efficient DNS settings to reduce the time it takes to resolve domain names to IP addresses.
-

| 11. Security Policies and Compliance

| 11.1. Security Standards and Frameworks

- **ISO/IEC 27001:** A global standard for managing information security risks via an ISMS.
- **ISO/IEC 27002:** A guide to best practices and controls for information security.**
- **NIST CSF:** A flexible framework to identify, protect, detect, respond to, and recover from cybersecurity risks.
- **NIST SP 800-53:** A detailed catalog of security and privacy controls for managing federal information systems.
- **GDPR:** A European regulation enforcing data privacy and protection rights for individuals, with heavy penalties for non-compliance.

| 11.2. Implementing and Enforcing Security Policies

When developing Security Policies: Define scope, involve stakeholders, and customize policies.

- *Enforcement Mechanisms* → Access controls (RBAC, MFA), Monitoring tools (IDS, IPS, SIEM), and Employee training.
- *Incident Response Plans* → Establish procedures for security incidents.

| 11.3. Compliance Auditing and Reporting

Compliance Auditing is the process of verifying whether an organization adheres to regulatory, industry, or internal standards. **Reporting** ensures findings are communicated effectively to stakeholders.

- *Prepare for Audits* → Maintain documentation, conduct internal reviews, and ensure audit readiness.
- *Conduct Compliance Audits* → External and internal audits, automated compliance tools, and penetration testing.
- *Report Findings* → Create audit reports, develop action plans, and communicate with stakeholders.

| 12. Advanced System Administration

System Administration is differently done in Linux vs Windows:

1. Linux Administration

- Command-line tools → *grep*, *sed*, *awk*, and *find*.
- System monitoring → *top*, *htop*, and *vmstat*.
- Security enhancements → *SELinux*, *firewalls*, and *SSH* practices.
- Configuration management → *Ansible* or *Puppet*.

2. Windows Administration

- Scripting/Automation → *Powershell*.
- Managing "Active Directory" and "Group Policies".
- Performance monitoring → *Event Viewer* and *Resource Monitor*.
- Security → *BitLocker*, *Windows Defender*, and *Firewall* settings.

| 12.1 High Availability Systems

Ensuring minimal downtime and redundancy and Monitoring and failover mechanisms.

1. **Linux High-Availability Solutions** → Clustering tools (Pacemaker, Corosync), Shared storage (NFS, iSCSI), Load balancing (HAProxy and Keepalived).
2. **Windows High-Availability Solutions** → WSFC (Failover clustering), SQL Server "Always On" for database redundancy, NIC teaming and network load balancing.

| 12.2. Load Balancing and Failover Strategies

Load Balancing → Distributes network traffic across multiple servers for reliability and performance.

- *Load Balancing Tools* → Software-based (HAProxy, Nginx), Hardware-based (F5 BIG-IP, Citrix ADC), Cloud-based (AWS Elastic Load Balancer, Azure Load Balancer).
- *Failover Strategies* → **Active-Passive** (One server is active while another waits in standby), **Active-Active** (All servers handle traffic simultaneously), **DNS-Level** and **Database Failover** (Reroute traffic or database connections when primary services fail).