**Name - Manan Solanki**

**Module - 9 WAN Technologies**

**1.** **Explain How Automation Impacts Network Management**

Automation makes network management **faster, easier, and more reliable** by reducing manual work and improving efficiency. Here’s how it helps:

**1. Saves Time & Effort**

Routine tasks like setting up, monitoring, and fixing networks happen automatically.

No need for manual configurations every time.

**2. Improves Security**

Detects and fixes security issues instantly.

Keeps networks safe by applying updates automatically.

**3. Reduces Errors**

Humans can make mistakes, but automation ensures things are done correctly.

Prevents network problems caused by misconfigurations.

**4. Handles Large Networks Easily**

Can manage thousands of devices without extra effort.

Perfect for big companies and cloud networks.

**5. Fixes Problems Faster**

Finds and solves issues before they cause serious trouble.

Reduces downtime and keeps everything running smoothly.

**6. Saves Money**

Less manual work means lower costs.

Uses resources efficiently, reducing waste.

**7. Keeps Performance High**

Balances traffic and prevents network slowdowns.

Ensures smooth and fast connections.

**8. Makes Compliance Easy**

Generates reports for audits without extra work.

**2**.**Compare Traditional network with Controller based networking**

**Ans. Comparison: Traditional Network vs. Controller-Based Networking**

| | **Feature** | **Traditional Network** | **Controller-Based Networking (SDN)** | | --- | --- | --- | | **Architecture** | Distributed, device-based management | Centralized, software-defined control | | **Configuration** | Manual, per-device configuration | Automated, centralized management | | **Scalability** | Limited, complex for large networks | Highly scalable, easy to expand | | **Flexibility** | Less flexible, changes require manual updates | More flexible, quick changes via controller | | **Automation** | Minimal, requires human intervention | High, uses automation and AI | | **Troubleshooting** | Manual, slow issue resolution | Automated, quick problem detection and resolution | | **Security** | Device-level security, harder to manage | Centralized security policies, better threat detection | | **Cost** | High operational costs due to manual work | Cost-efficient due to automation and optimization | | **Performance** | Slower, depends on manual optimization | Faster, dynamic traffic control for efficiency | | **Management** | Requires individual configuration on each device | Centralized control, easier management | | **Adaptability** | Difficult to integrate with modern technologies | Easily integrates with cloud, IoT, and AI | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**3.** **Explain Virtualization**

Virtualization is a technology that allows you to create multiple virtual environments on a single physical system. It enables better resource utilization, cost savings, and flexibility by running multiple operating systems or applications on the same hardware.

**Types of Virtualization :**

1. **Server Virtualization**
   * Divides a single physical server into multiple virtual servers.
   * Each virtual server operates independently with its own OS and applications.
   * Example: VMware, Microsoft Hyper-V.
2. **Network Virtualization**
   * Creates multiple virtual networks on shared physical infrastructure.
   * Improves security and network performance.
3. **Storage Virtualization**
   * Combines multiple physical storage devices into a single virtual storage unit.
   * Makes storage management easier and more efficient.
   * Example: SAN (Storage Area Network), NAS (Network Attached Storage).
4. **Desktop Virtualization**
   * Allows users to access a virtual desktop remotely from any device.
   * Increases security and reduces hardware costs.
   * Example: Virtual Desktop Infrastructure (VDI), Citrix.
5. **Application Virtualization**
   * Runs applications in isolated environments without installing them on a physical device.
   * Helps in software compatibility and security.
   * Example: Microsoft App-V, VMware ThinApp.

**4** .**Describe Characteristics of REST-based API.**

### **1. Stateless**

* Each request from the client contains all the information needed by the server to process it.
* The server does **not store session** or client context between requests.

### **2. Client-Server Architecture**

* The client and server are **separate**, promoting loose coupling.
* The client handles the UI and user interaction, while the server manages data and processing.

### **3. Uniform Interface**

* A consistent way to access resources using:  
  + **HTTP methods**: GET, POST, PUT, DELETE
  + **URIs** to identify resources
  + **Media types** like JSON or XML

### **4. Resource-Based**

* Data and functionality are considered **resources**, each identified by a unique **URI**.  
  + Example: GET /users/123 fetches data for user with ID 123.

### **5. Use of Standard HTTP Methods**

* **GET** – Read data
* **POST** – Create new data
* **PUT** – Update existing data
* **DELETE** – Remove data

### **6. Representation-Oriented**

* Resources are sent in a **representation format** (usually JSON or XML).
* Clients interact with representations, not the actual server-side data or logic.

### **7. Stateless Communication**

* Each request is **independent**, improving scalability.
* Easier to cache, load balance, and handle failures.

### **8. Cacheable**

* Responses can be explicitly marked as **cacheable** or **non-cacheable** to improve performance.

### **9. Layered System**

* REST APIs can be structured in **layers**, like authentication, load balancers, etc., which are **transparent** to the client.

**5. Explanation methods of Automation.**

### **Common Methods of Automation:**

### **1. Scripting**

* Writing custom scripts in languages like Python, Bash, PowerShell to automate tasks.
* Example: A Python script to back up a folder daily.

### **2. Task Scheduler / Cron Jobs**

* Time-based automation tools:  
  + Task Scheduler (Windows)
  + Cron Jobs (Linux/Unix)
* Automatically run scripts or commands at specified intervals.

### **3. Configuration Management Tools**

* Tools like Ansible, Puppet, Chef, SaltStack are used to configure systems automatically.
* They define system configurations as code ("Infrastructure as Code").

### **4. CI/CD Pipelines (DevOps)**

* Tools like Jenkins, GitLab CI, GitHub Actions, CircleCI automate:  
  + Code build → Test → Deploy → Monitor

### **5. Workflow Automation Tools**

* Tools like Zapier, Microsoft Power Automate, IFTTT connect apps/services and automate multi-step workflows.
* No coding required.

**6.** **Explain SDN**

Software-Defined Networking (SDN)

Software-Defined Networking (SDN) is a modern networking approach that separates the control plane (decision-making) from the data plane (packet forwarding) to create a more flexible, efficient, and programmable network.

Use Cases of SDN :

· Data Centers – Cloud providers use SDN to manage large-scale network infrastructure dynamically.

· Enterprise Networks – Businesses deploy SDN to improve network security and optimize traffic.

· 5G and IoT Networks – SDN enables dynamic network slicing for different applications.

· Cybersecurity – Centralized monitoring helps detect threats and enforce security policies.

**7. Explain DNA centres.**

It is the **control center** for Cisco’s **intent-based networking**, which means you define *what you want*, and the system figures out *how to do it* automatically.

### **1. Network Automation**

* Automates device provisioning, configuration, and updates
* Supports “zero-touch provisioning” of devices
* Automates policy application across the network

### **2. Policy-Based Management**

* Uses **software-defined access (SD-Access)** to define user/device policies
* Applies policies consistently across the network (campus, branches, etc.)

### **3. Assurance & Analytics**

* Provides real-time monitoring and **AI-powered analytics**
* Detects issues like slow apps, device failures, or connectivity problems
* Uses **machine learning** to recommend fixes

### **4. Integration with APIs**

* Offers **REST APIs** for integration with third-party systems (e.g., ServiceNow, custom scripts)
* Enables DevOps and NetDevOps workflows

### **5. Security Integration**

* Integrates with **Cisco ISE (Identity Services Engine)** for secure access control
* Tracks users/devices and enforces security policies automatically

**8. Explain SD-Access and SD-WAN.**

SD- Access - **SD-Access** is Cisco's solution for automating and securing **campus and branch LAN (Local Area Network)** infrastructure. It simplifies how users and devices access the network by using **policy-based automation**.

SD-WAN - **SD-WAN** is Cisco's (and others’) solution for managing **WAN (Wide Area Network)** traffic across multiple locations, like **branch offices, data centers, and cloud**. It replaces traditional MPLS with cheaper and more flexible broadband or LTE links, **securely** and **intelligently**.