1.7.3 TCP/IP REFERENCE Model

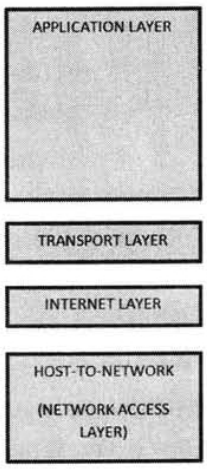
TCP/IP means Transmission Control Protocol and Internet Protocol. It is the network model used in the current Internet architecture as well. Protocols are set of rules which govern every possible communication over a network. These protocols describe the movement of data between the source and destination or the internet. These protocols offer simple naming and addressing schemes.

TCP/IP that is Transmission Control Protocol and Internet Protocol was developed by Department of Defence's Project Research Agency (ARPA, later DARPA) as a part of a research project of network interconnection to connect remote machines.

The features that stood out during the research, which led to making the TCP/IP reference model were:

- Support for a flexible architecture. Adding more machines to a network was easy.
- The network was robust, and connections remained intact untill the source and destination machines were functioning.

The overall idea was to allow one application on one computer to talk to(send data packets) another application running on different computer.



Description of different TCP/IP protocols

1.7.3.1 Layer 1: Host-to-network Layer

1. Lowest layer of the all.
2. Protocol is used to connect to the host, so that the packets can be sent over it.
3. Varies from host to host and network to network.

1.7.3.2 Layer 2: Internet layer

1. Selection of a packet switching network which is based on a connectionless internetwork layer is called a internet layer.
2. It is the layer which holds the whole architecture together.
3. It helps the packet to travel independently to the destination.
4. Order in which packets are received is different from the way they are sent.
5. IP (Internet Protocol) is used in this layer.

1.7.3.3 Layer 3: Transport Layer

1. It decides if data transmission should be on parallel path or single path.
2. Functions such as multiplexing, segmenting or splitting on the data is done by transport layer.
3. The applications can read and write to the transport layer.
4. Transport layer adds header information to the data.
5. Transport layer breaks the message (data) into small units so that they are handled more efficiently by the network layer.
6. Transport layer also arrange the packets to be sent, in sequence.

1.7.3.4 Layer 4: Application Layer

The TCP/IP specifications described a lot of applications that were at the top of the protocol stack. Some of them were TELNET, FTP, SMTP, DNS etc.

1. TELNET is a two-way communication protocol which allows connecting to a remote machine and run applications on it.
2. FTP(File Transfer Protocol) is a protocol, that allows File transfer amongst computer users connected over a network. It is reliable, simple and efficient.
3. SMTP(Simple Mail Transport Protocol) is a protocol, which is used to transport electronic mail between a source and destination, directed via a route.

4. DNS(Domain Name Server) resolves an IP address into a textual address for Hosts connected over a network.

1.7.3.5 Merits of TCP/IP model

1. It operated independently.
2. It is scalable.
3. Client/server architecture.
4. Supports a number of routing protocols.
5. Can be used to establish a connection between two computers.

1.7.3.6 Demerits of TCP/IP

1. In this, the transport layer does not guarantee delivery of packets.
2. The model cannot be used in any other application.
3. Replacing protocol is not easy.
4. It has not clearly separated its services, interfaces and protocols.

Comparison of OSI Reference Model and TCP/IP Reference Model

Following are some major differences between OSI Reference Model and TCP/IP Reference Model

OSI(Open System Interconnection)	**TCP/IP(Transmission Control Protocol / Internet Protocol)**
1. OSI is a generic, protocol independent standard, acting as a communication gateway between the network and end user.	1. TCP/IP model is based on standard protocols around which the Internet has developed. It is a communication protocol, which allows connection of hosts over a network.
2. In OSI model the transport layer guarantees the delivery of packets.	2. In TCP/IP model the transport layer does not guarantees delivery of packets. Still the TCP/IP

	model is more reliable.
3. Follows vertical approach.	3. Follows horizontal approach.
4. OSI model has a separate Presentation layer and Session layer.	4. TCP/IP does not have a separate Presentation layer or Session layer.
5. OSI is a reference model around which the networks are built. Generally it is used as a guidance tool.	5. TCP/IP model is, in a way implementation of the OSI model.
6. Network layer of OSI model provides both connection oriented and connectionless service.	6. The Network layer in TCP/IP model provides connectionless service.
7. OSI model has a problem of fitting the protocols into the model.	7. TCP/IP model does not fit any protocol
8. Protocols are hidden in OSI model and are easily replaced as the technology changes.	8. In TCP/IP replacing protocol is not easy.
9. OSI model defines services, interfaces and protocols very clearly and makes clear distinction between them. It is protocol independent.	9. In TCP/IP, services, interfaces and protocols are not clearly separated. It is also protocol dependent.
10. It has 7 layers	10. It has 4 layers

Introduction to Network Ports

What is a network port?

Think of a port as a virtual gateway that a specific service, process, or application on your computer uses for network communication. Each port is assigned a unique number, allowing different types of traffic to be directed to the appropriate software. For example, your email might use one port, while your web browsing uses another. When combined with an IP address, a port number creates a complete socket address, enabling precise routing of data to and from your computer across the network

The role of ports in network communication

Ports help computers sort the network traffic they receive, ensuring that different types of traffic are directed to the correct applications. Ports allow different services or applications on the same device to communicate with each other and with external systems.

Overview of port ranges (Well-known, Registered, Dynamic/Private)

Network ports are categorized into three main ranges, each serving distinct purposes in network communication. Most ports are permanently assigned as Well-Known Ports or Registered Ports. By the Internet Assigned Numbers Authority (IANA). The IANA is responsible for the global coordination of port number assignments and maintains the official registry of port number assignments.

Port Ranges Overview

Well-Known/System Ports (0-1023): The Well-Known or System Ports, ranging from 0 to 1023, are reserved for common, widely used services. These are only used by system processes, operating systems and default applications. These common network ports include HTTP (80), HTTPS (443), SMTP (25), and SSH (22).

Registered Ports (1024-49151): Thees ports are used by applications or services that are less common but still require specific ports to function properly.Important port numbers in this range include Remote Desktop Protocol (3389), Xbox LIVE and Games for Windows (3074) and IBM Lotus Notes/Domino (1352).

Dynamic/Private Ports (49152-65535): These ports are used for temporary or short-lived connections and are not assigned to specific services. They're often employed as source ports for outgoing connections and can be used by any process.

Understanding TCP and UDP Ports

TCP UDP are two different protocols that use ports to manage network communications. The main difference between them lies in how they handle data transmission. Think of the options for sending a letter. TCP is like sending a registered letter that requires confirmation of receipt and to ensure ordered delivery. UDP, on the other hand, is akin to dropping a letter in a mailbox. While it may be cheaper and faster, it offers no guarantee of delivery or order. TCP prioritizes reliability, while UDP favors speed and efficiency.

These protocols are part of the transport layer in the OSI model, which directs how data is transmitted between devices.

TCP, or Transmission Control Protocol, is connection oriented. This means it establishes a connection before sending data and ensures that all packets arrive in the correct order and without errors. This makes TCP reliable but can slow down the communication process. It is commonly used for applications where accuracy is crucial, such as web browsing, email, and file transfers.

UDP, or User Datagram Protocol, is connectionless and does not establish a connection before sending data. It sends packets without checking if they arrive

correctly or in order, which makes UDP faster and more efficient. This speed is beneficial for real-time applications like online gaming, video streaming, and voice calls where timely delivery is more important than perfect accuracy.

When to use TCP vs. UDP ports

TCP (Transmission Control Protocol) ports are best used when:

- Reliability is crucial and you need to ensure all data arrives intact and in order.
- Error-checking and correction are important.
- The application requires confirmation of data delivery.
  UDP (User Datagram Protocol) ports are preferable when:

- Speed is more important than perfect reliability.
- Small data losses are acceptable.
- Real-time applications are involved, such as live streaming, online gaming, or VoIP.

Examples of common protocols utilizing TCP and UDP

TCP is generally used for applications requiring reliable, ordered data transmission. The most common TCP ports include:

- HTTP/HTTPS (ports 80 and 443) for web browsing
- FTP (port 21): File Transfer Protocol for transferring files
- SMTP (port 25): Simple Mail Transfer Protocol for outgoing email
- SSH (port22): Secure Shell for secure remote administration of devices
  UDP is preferred for applications prioritizing speed and low latency over perfect reliability. Common UDP ports include:

- DNS (port 53): Domain Name System used for domain name resolution
- DHCP (port 67): Dynamic Host Configuration Protocol for issuing IP addresses
- NTP (port 123): Network Time Protocol

- VoIP (port 56): Voice over Internet Protocol used for phone conversations

Top 15 Most Commonly Used Ports

Below is a list of the 15 most common ports and protocols in numerical order showing which protocol they use.

1. 21 (TCP, UDP) – FTP (File Transfer Protocol)
2. 22 (TCP, UDP) – SSH (Secure Shell)
3. 23 (TCP) – Telnet
4. 25 (TCP) – SMTP (Simple Mail Transfer Protocol)
5. 53 (TCP, UDP) – DNS (Domain Name System)
6. 80 (TCP) – HTTP (Hypertext Transfer Protocol)
7. 110 (TCP) – POP3 (Post Office Protocol version 3)
8. 143 (TCP, UDP) – IMAP (Internet Message Access Protocol)
9. 443 (TCP) – HTTPS (HTTP Secure)
10. 445 (TCP) – SMB (Server Message Block)
11. 993 (TCP, UDP) – IMAPS (IMAP over SSL)
12. 995 (TCP, UDP) – POP3S (POP3 over SSL)
13. 3306 (TCP) – MySQL database system
14. 3389 (TCP) – RDP (Remote Desktop Protocol)
15. 8080 (TCP) – Alternative HTTP port, often used for web proxies

Common Ports and Protocols in Daily Use

How common ports support everyday services (e.g., web browsing, email, file transfers)

Think of how you contact the people you communicate the most on your cell phone. You connect to them by the contact's name in your phone, not thinking

about the phone number that is assigned to them. All the common things you do on your computer such as web browsing or sending an email use ports behind the scenes as well. These ports act like invisible channels, directing different types of internet traffic to the right applications. Just as you don't need to remember phone numbers, you don't have to think about common port numbers when using the internet. The system automatically uses the appropriate ports for each service, ensuring smooth communication between your computer and various online services.

Use cases for HTTP, HTTPS, FTP, and SMTP

We all spend most of our time on the internet when on our computing devices. You open a web browser and begin to surf the internet using HTTP on port 80. You begin to shop for an item and use the HTTPS protocol to secure the purchase transaction using port 443. You then send an email to a friend using SMTP on port 25. You then transfer a file to a local server on your corporate network using FTP on port 21. Each of these actions were performed on a dedicated channel or port.

Importance of Port 443 (HTTPS) in securing web traffic

Both HTTP and HTTPS are used for web traffic. Unlike HTTP that sends data in plain text, HTTPS ensures that sensitive information like passwords, credit card details, and personal data remains confidential. This encryption ensures privacy and data integrity. HTTPS provides authentication through digital certificates. When a user connects to a website via Port 443, the server presents its SSL/TLS certificate. This certificate, issued by a trusted Certificate Authority, verifies the website's identity, helping users confirm they're connecting to the legitimate site and not a malicious impersonator.

Essential Ports for Network Administrators

Key ports for network diagnostics and security

Network managers have a vast IT estate that they must monitor and manage to keep their networks optimized and secure. Some of the common ports they use every day include:

- Port 3389 (RDP): Used for remotely accessing Windows desktops and servers. It enables administrators to control and manage Windows machines remotely as if they were sitting in front of the device.
- Port 22 (SSH): Provides secure remote access to network devices like switches and routers. It is the standard port for encrypted command-line access to network equipment and Linux systems.
- Port 123 (NTP): Ensures time synchronization across all network devices, servers, and workstations. This port is crucial for maintaining accurate and consistent timestamps throughout the network.
- Ports 389 and 636: Port 389 is used for standard LDAP (Lightweight Directory Access Protocol) connections to access directory services like Windows Active Directory. Port 636 is used for LDAPS (LDAP over SSL/TLS), providing secure, encrypted access to those same directory services.
- Port 161/162 (SNMP): These ports are used for Simple Network Management Protocol. SNMP facilitates network device monitoring and management by allowing administrators to query devices for information (via port 161), make configuration changes, and receive alerts about network events or issues (via port 162). SNMP is used for maintaining, troubleshooting, and optimizing network performance across a wide range of devices including routers, switches, servers, and other networked equipment.

Common open ports and their potential security risks

Some ports are more secure than others, but no port is completely safe from compromise. Here are some of the potential security risks for popular ports you use every day.

- Ports 80 and 443 (HTTP/HTTPS): Vulnerable to web attacks like cross-site scripting (XSS), SQL injections, and DDoS attacks.
- Authentication-required ports (FTP, SSH, RDP): Susceptible to brute force attacks targeting passwords.
- Port 25 (SMTP): Vulnerable to email-based attacks including phishing, spoofing, and spam.
- Unencrypted ports (21-FTP, 23-Telnet): At risk of traffic interception.
- Port 445 (SMB): Known for the EternalBlue vulnerability, exploited in the WannaCry ransomware attack.
- Port 3389 (RDP): Vulnerable to exploits like BlueKeep.
- Ports 1433/1434 (SQL Server): Targeted for SQL injection attacks and unauthorized data access.

Port Security Considerations

Impact of open ports on network security

At the very least, your network should be protected by a perimeter firewall. By default, a firewall closes ports for all incoming traffic and opens all ports for outgoing traffic. Every time a port is opened on the firewall it creates a potential entry point for attackers, thus increasing the attack surface of your network. Open ports can also reveal valuable information about the network infrastructure and services, aiding attackers in reconnaissance effort. Outgoing common open ports can be exploited for data exfiltration or to send malicious emails. Open ports aren’t just an issue for perimeter security, however. Open ports on your local servers can enable lateral movement for attackers and facilitate the spread of malware.

Strategies for securing ports

Below is a list of security initiatives you should take to secure port traffic in your network.

- Conduct frequent port scans using tools like Nmap or vulnerability scanners to identify open ports, associated services, and potential vulnerabilities. Maintain an up-to-date inventory of all open ports and their purposes.
- Conduct Regular Vulnerability Assessments and Penetration Testing focused on open ports to identify potential weaknesses in port security
- Enable local firewalls on all your servers and only open the ports required for essential services and applications on those machines.
- Implement access control lists (ACLs) to restrict access to specific ports from authorized sources only. This should be done on routers and servers.
- Always use a secure protocol when possible, such as HTTPS instead of HTTP and SFTP instead of FTP.
- Implement Strong Authentication using multi-factor authentication for accessing critical services. Implement SH keys instead of passwords where possible and regularly rotate credentials while enforcing strong password policies.
- Monitor and Log Port Activities: Set up comprehensive logging for all port access attempts (both successful and unsuccessful) and regularly review these logs to identify suspicious activities or unauthorized access attempts.

Comprehensive Port List and Descriptions

- Port 20/21 (FTP): File Transfer Protocol for data transfer and control
- Port 22 (SSH): Secure Shell for encrypted remote access
- Port 23 (Telnet): Unencrypted text communications (insecure)
- Port 25 (SMTP): Simple Mail Transfer Protocol for email transmission
- Port 53 (DNS): Domain Name System for name resolution
- Port 80 (HTTP): Hypertext Transfer Protocol for web browsing
- Port 88: Kerberos authentication
- Port 110 (POP3): Post Office Protocol version 3 for email retrieval
- Port 123 (NTP): Network Time Protocol for time synchronization
- Port 139 NetBIOS Session Service over TCP/IP
- Port 143 (IMAP): Internet Message Access Protocol for email retrieval
- Port 161/162 (SNMP): Simple Network Management Protocol for network device monitoring
- Port 389 (LDAP): Lightweight Directory Access Protocol for directory services
- Port 443 (HTTPS): HTTP Secure for encrypted web communication

- Port 445 (SMB): Server Message Block for file sharing on Windows networks
- Port 465 (SMTPS): SMTP over SSL for secure email transmission
- Port 500 (IKE): Internet Key Exchange for VPN key negotiation
- Port 636 (LDAPS): LDAP over SSL for secure directory services
- Port 993 (IMAPS): IMAP over SSL for secure email retrieval
- Port 1433/1434 (MSSQL): Microsoft SQL Server database communication
- Port 3306 (MySQL): MySQL database communication
- Port 3389 (RDP): Remote Desktop Protocol for remote access to Windows systems
- Port 5060/5061 (SIP): Session Initiation Protocol for VoIP communication
- Port 8080 (HTTP Alternate): Commonly used for web proxies and caching

Registered Ports: Applications in Use

While every network utilizes many of the well-known system ports, the registered ports used will vary according to the applications deployed by your organization. More examples of commonly used registered ports include:

- Port 1433/1434: Microsoft SQL Server
- Port 1521 – Oracle Database
- Port 1812 – RADIUS Authentication
- Port 2049 – Network File System (NFS)
- Port 3306 – MySQL
- Port 3389 – Remote Desktop Protocol (RDP)
- Port 5631-5632: pcAnywhere
- Port 6665-6669: Internet Relay Chat (IRC)

Dynamic/Private Ports

How dynamic ports are assigned

Dynamic ports fall within a range of 49152-65535 and are typically used on the client side of a connection. In some cases, the dynamic port range can be configured to meet specific network requirements. These ports can either be specifically assigned by an operating system or can be randomized within the

dynamic range. Dynamic ports are assigned temporarily and are released back to the pool when the connection is closed.

A classic example of dynamic ports in use is Network Address Translation (NAT). Workstations in a large organization are routed out the firewall to the internet. As outgoing packets pass through the firewall or NAT device, it changes the source IP address from the private IP to a public IP. Of course, there are not enough public IP addresses for each device to have a unique address. This is where dynamic ports come into play. The source is assigned a dynamic port, with each outgoing request receiving a different port number. All ports are stored in a NAT table to consistently translate packets from the same internal host and port to the same external IP and port. This is how return traffic is matched with the source devices.

Common applications using dynamic ports, e.g.,

Common applications that utilize dynamic ports include the following:

- Web browsers: When you open multiple tabs or windows, each connection may use a different dynamic port.
- Email clients: Email applications often use dynamic ports for outgoing connections.
- Instant messaging applications: Apps like Skype and WhatsApp use dynamic ports for peer-to-peer connections.
- File transfer applications: Secure FTP clients, torrent clients, and cloud storage sync tools often use dynamic ports for data transfer.
- Online gaming: Many multiplayer games use dynamic ports for player-to-player connections or game server communications.
- Remote desktop applications: Tools like TeamViewer or VNC viewers use dynamic ports for outgoing connections.
- VoIP applications: Voice over IP software often uses dynamic ports for call setup and audio transmission.
- Streaming media players: When streaming content, media players may use dynamic ports for data reception.
- Database clients: Applications connecting to remote databases may use dynamic ports for outgoing connections.

- VMware: A port is created and assigned to a VM when the VM is powered on connected to the network.

Advanced Port Usage

Port forwarding

By default, a perimeter firewall blocks all incoming traffic from the Internet. If your organization hosts web facing applications, websites, email servers or data transfer sites, you will need to utilize port forwarding. Port forwarding allows specific incoming traffic to reach internal devices by mapping external ports on the router's public IP address to a specific internal IP address and port. When incoming traffic arrives on the specified external port, the router forwards it to the designated internal device.

Port scanning

Port scanning allows administrators and security professionals to gain a comprehensive understanding of their network infrastructure by identifying open ports, active services, and potential vulnerabilities. Regular port scanning can help ensure compliance with various security standards and regulations. Beyond security, port scanning aids in network troubleshooting, performance optimization, and change management by providing a clear picture of the network's current state. Some port scanning tools include Nmap, Netcat, and Angry IP Scanner. One of the most basic TCP Connect Scanning techniques involves attempting to complete a full TCP three-way handshake with the target system. Another simple way is to simply send UDP packets to detect open UDP ports.