

# Section 1.1.2 Computer Networks Fundamentals

## P2P and Client-Server Architecture

### Cracking OSCP: Your Roadmap to Ethical Hacking Success

[YouTube: HackProKP – Kailash Parshad](#)

[LinkedIn: Kailash Parshad](#)

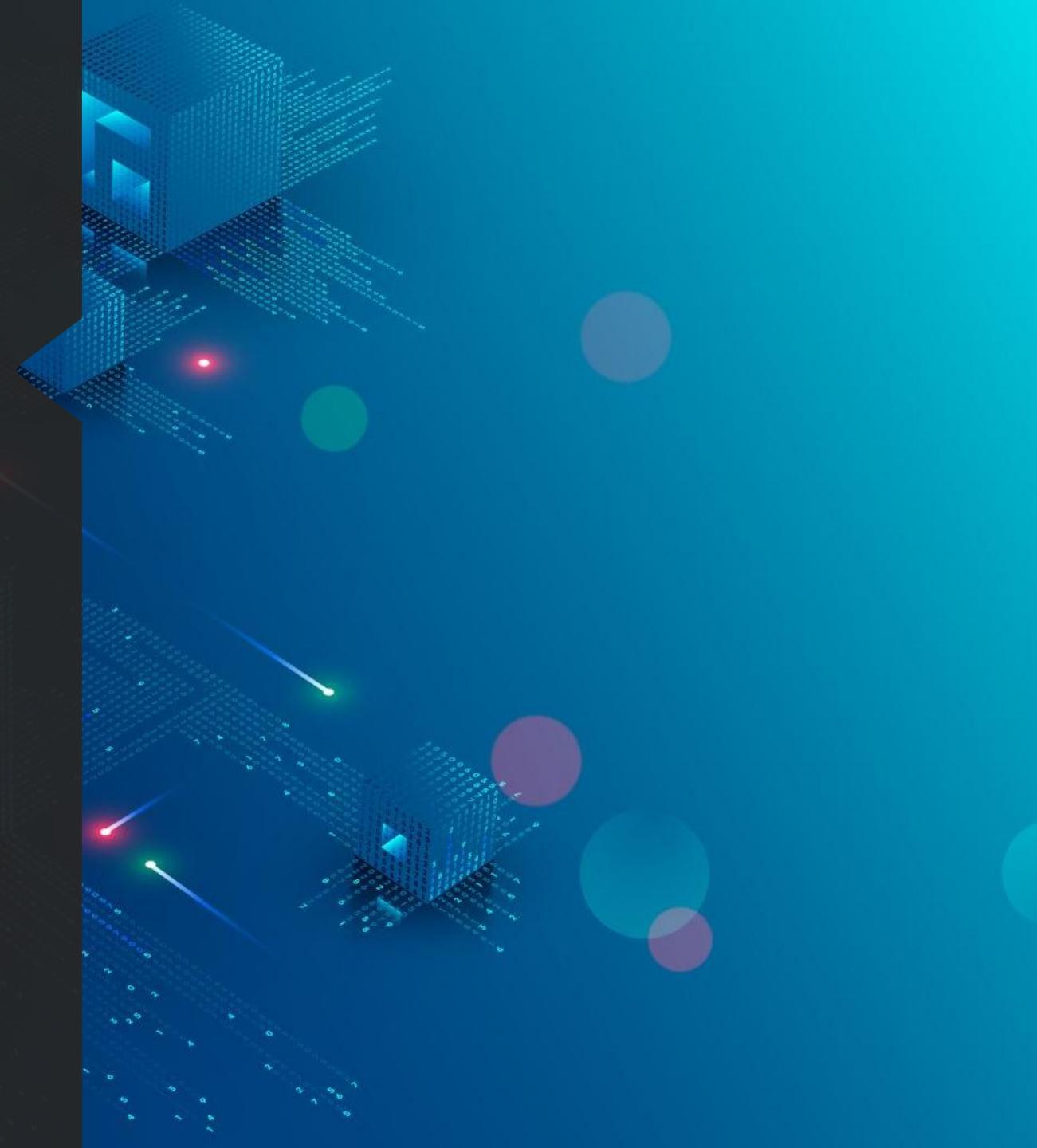
[Github: https://github.com/at0m-b0mb/Cracking-OSCP-Your-Roadmap-to-Ethical-Hacking-Success](https://github.com/at0m-b0mb/Cracking-OSCP-Your-Roadmap-to-Ethical-Hacking-Success)

[Complete Youtube Playlist:](#)

<https://www.youtube.com/watch?v=MvkNbn8i2so&list=PLyrv3TPh3ejYNZipa0OIUvkdjHeUTRJ3J&index=1&t=0s>

# APPLICATION ARCHITECTURE

- The application architecture is designed by the application developer and dictates how the application is structured over the various end systems.
- Both P2P and client-server architectures can utilize protocols at the Application layer of the TCP/IP model.



An aerial photograph of a modern city at night, featuring a dense network of glowing yellow and green lines that form a complex web across the urban landscape. This visual metaphor represents the interconnectedness and distributed nature of Peer-to-Peer (P2P) architecture.

# Peer-to-Peer (P2P) Architecture

# Peer-to-Peer (P2P) Architecture

- In a P2P architecture, computers (or nodes) communicate with each other on an equal footing. There is no central server controlling the communication; instead, each node can act as both a client and a server. P2P networks are decentralized and are often used for file-sharing applications, such as BitTorrent.

# Key components:

- **Peer:** A peer is an individual computer or device that participates in the P2P network. Each peer has equal status and can act both as a client and a server. Peers initiate communication, share resources (such as files or processing power), and contribute to the overall functionality of the P2P network.

# Key components:

- **Tracker:** A tracker is a centralized server that helps coordinate communication among peers in a P2P network. In certain P2P systems, like BitTorrent, a tracker maintains information about which peers have which parts of a shared resource (e.g., a file). Peers communicate with the tracker to discover other peers in the network.

# Key components:

- **Seeder:** A seeder is a peer that has a complete copy of the shared resource (e.g., a file) and is actively sharing it with other peers. Seeders contribute to the distribution of the resource by providing complete copies to other peers. They help improve the availability of the resource in the network.

# Key components:

- **Leecher:** A leecher is a peer that is in the process of downloading a resource but may not be actively sharing the complete resource with others. Leechers primarily download content from other peers. Once they have the complete resource, they can become seeders and contribute to the distribution.

# Key components:

- **Swarm:** A swarm is a group of peers that are actively sharing and downloading parts of a specific resource. The swarm collectively works to distribute and share the resource efficiently. Peers in the swarm communicate with each other to exchange data.

# Key components:

- **Distributed Hash Table (DHT):** In some P2P systems, a Distributed Hash Table is used to decentralize the tracking of resources and peers, eliminating the need for a central tracker. DHT allows peers to locate and discover other peers in the network without relying on a central server. It provides a distributed and scalable way to manage information.

# Key components:

- **Overlay Network:** An overlay network is a virtual network that is built on top of the underlying physical network, connecting peers in the P2P network. The overlay network facilitates communication and resource sharing among peers. It allows for direct connections between peers, bypassing the need for a central server in some cases.

# Key characteristics:

- **Decentralization:** No single node has complete control, and each node can initiate communication.
- **Resource Sharing:** Nodes can share resources directly with each other, such as files or processing power.
- **Scalability:** P2P networks can be easily scaled by adding more nodes to the network.

# Advantages:

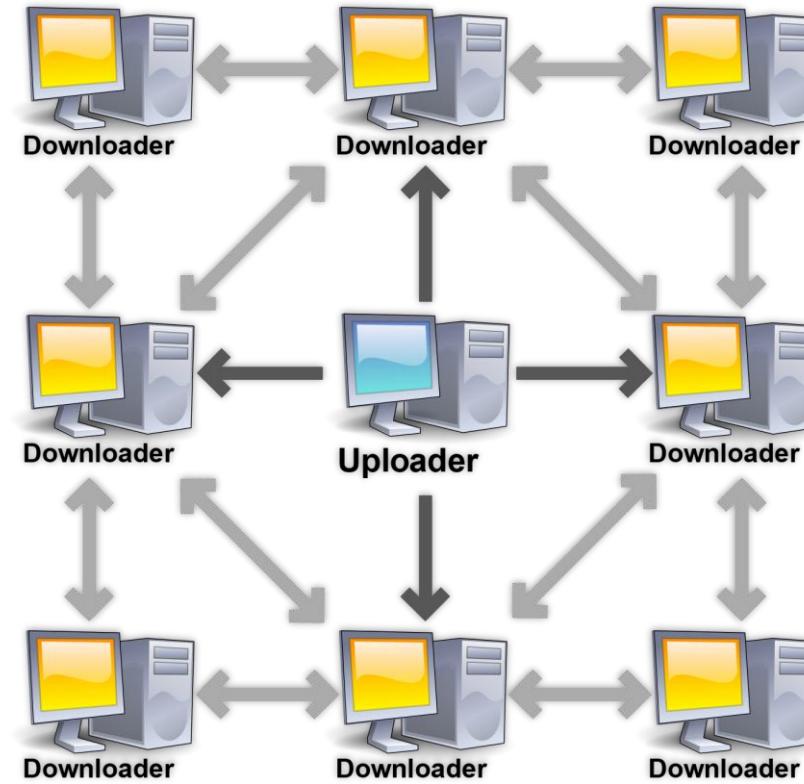
- **Redundancy:** No single point of failure; if one node goes down, others can continue to communicate.
- **Scalability:** Easily scalable with the addition of more nodes.
- **Resource Utilization:** Nodes can directly share resources without relying on a central server.

# Disadvantages:

- **Security:** Can be challenging to manage security in a fully decentralized environment.
- **Coordination:** Coordination and organization become complex as the network grows.
- **Reliability:** Relies on the availability and reliability of individual nodes.

# BitTorrent

The middle computer acts as a "seed" to provide a file to the other computers which act as peers.



<https://en.wikipedia.org/wiki/BitTorrent>

# How it works (BitTorrent)

1. **File Distribution:** A user who wants to share a file (Seeder) divides it into smaller pieces.
2. **Peer Connection:** Other users (Peers) connect to the Seeder and each other, forming a swarm.
3. **Piece Exchange:** Peers exchange pieces of the file with each other simultaneously.
4. **Upload and Download:** While downloading parts of the file, a peer can also upload the parts it has to others.

# Advantages (BitTorrent)

- **Redundancy:** If one peer goes offline, others can still share the file.
- **Scalability:** More peers can join the network to share and download files.

# Disadvantages (BitTorrent)

- **Security:** Ensuring the integrity and security of shared files can be challenging.
- **Coordination:** Managing the decentralized network and ensuring efficient file distribution.

# CLIENT-SERVER Architecture

# Client-Server Architecture:

- In a client-server architecture, communication is structured around clients and servers. Clients request services or resources, and servers provide these services. This model is more centralized compared to P2P, with a clear distinction between the roles of clients and servers.
- The HTTP (HyperText Transfer Protocol) is a fundamental protocol used for the World Wide Web.
- It employs a client-server architecture, where web browsers (clients) request web pages from web servers.

# Key components:

- **Client:** Initiates requests for services or resources.
- **Server:** Listens for client requests and provides the requested services or resources.

# Key characteristics:

- **Centralized Control:** Servers control and manage resources, providing services to clients.
- **Specialization:** Servers are specialized for particular tasks, and clients use these specialized services.
- **Reliability:** Centralized control can lead to better management and reliability.

# Advantages:

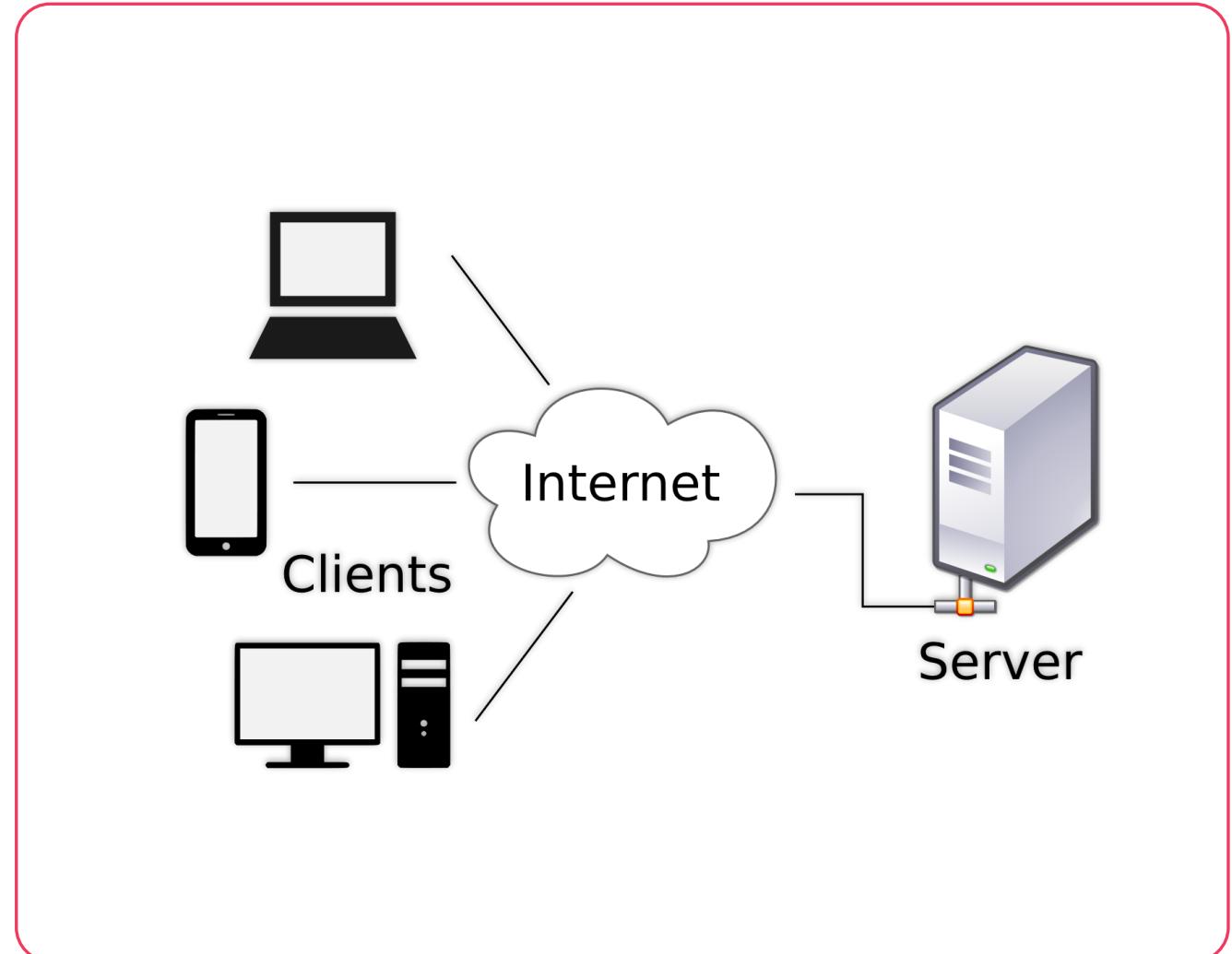
- **Centralized Control:** Easier to manage and secure compared to P2P networks.
- **Resource Management:** Efficient resource management with specialized servers.
- **Security:** Centralized control can provide better security measures.

# Disadvantages:

- **Scalability:** Scaling may require more powerful servers to handle increased client requests.
- **Single Point of Failure:** If the server goes down, clients cannot access the services.
- **Cost:** Setting up and maintaining servers can be costly.

# Web Browsing

Diagram of clients communicating  
with a server via the Internet



# How it works (Web Browsing)

- 1. Client Request:** A user opens a web browser and enters a URL (Uniform Resource Locator).
- 2. Server Response:** The browser sends an HTTP request to the web server specified in the URL.
- 3. Data Transfer:** The server processes the request and sends back the requested web page (HTML, images, etc.).
- 4. Rendering:** The browser receives the data and renders it for the user to view.

# Thank You!



Please Like and Subscribe to never miss a video

And to help me with the Algorithm

And It's completely Free!!!

Love you guys