Layer Four Attacks

Ethical Hacking

Alessandro Brighente
Eleonora Losiouk

October, 2022

Master Degree on Cybersecurity

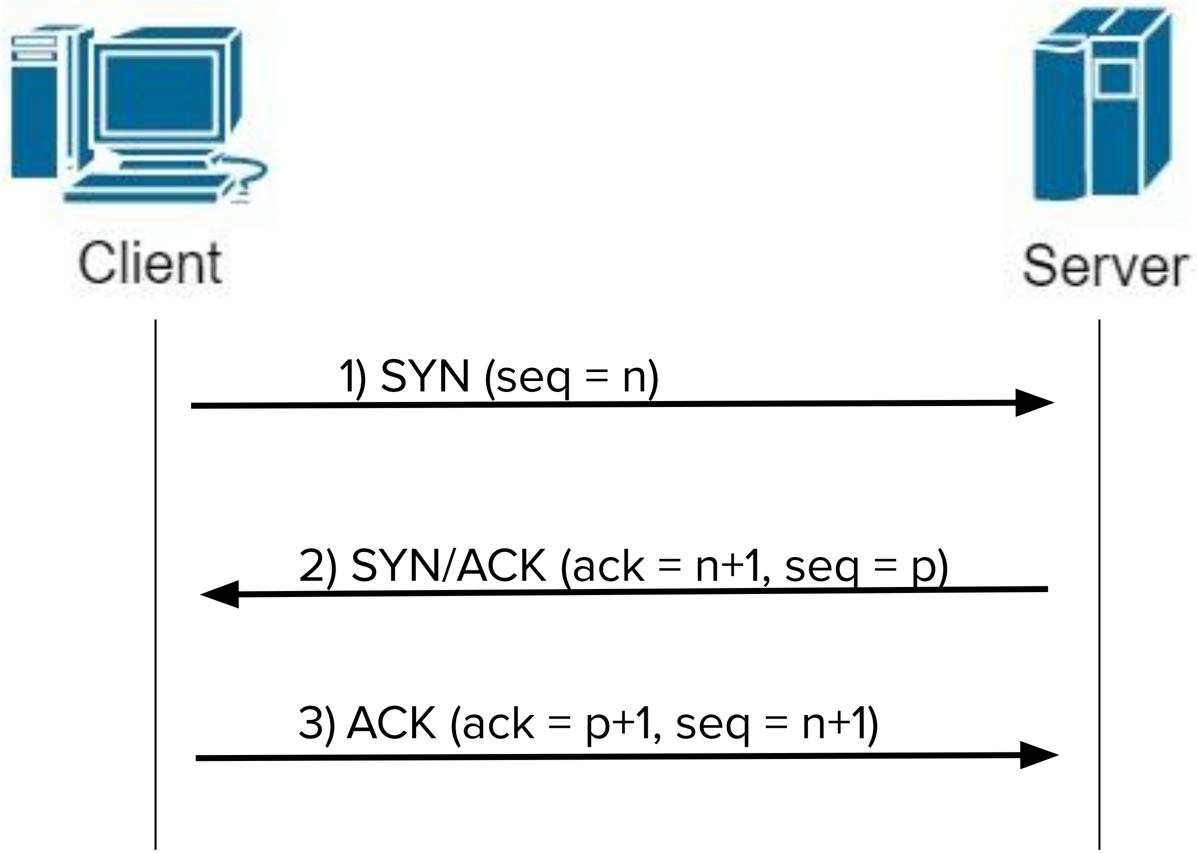
UNIVERSITÀ
DEGLI STUDI
DI PADOVA

SPRITZ
SECURITY & PRIVACY
RESEARCH GROUP

Layer 4 Protocols

- Transmission Control Protocol (TCP)
  - Connection based
  - Includes error recovery (acknowledgment)
- User Datagram Protocol (UDP)
  - Connectionless
  - Smaller overhead
  - No error recovery
- Both use ports to identify communicating parties
  - 16 bits UINT (0-65535)
  - Well known ports for services

TCP Three-Way Handshake

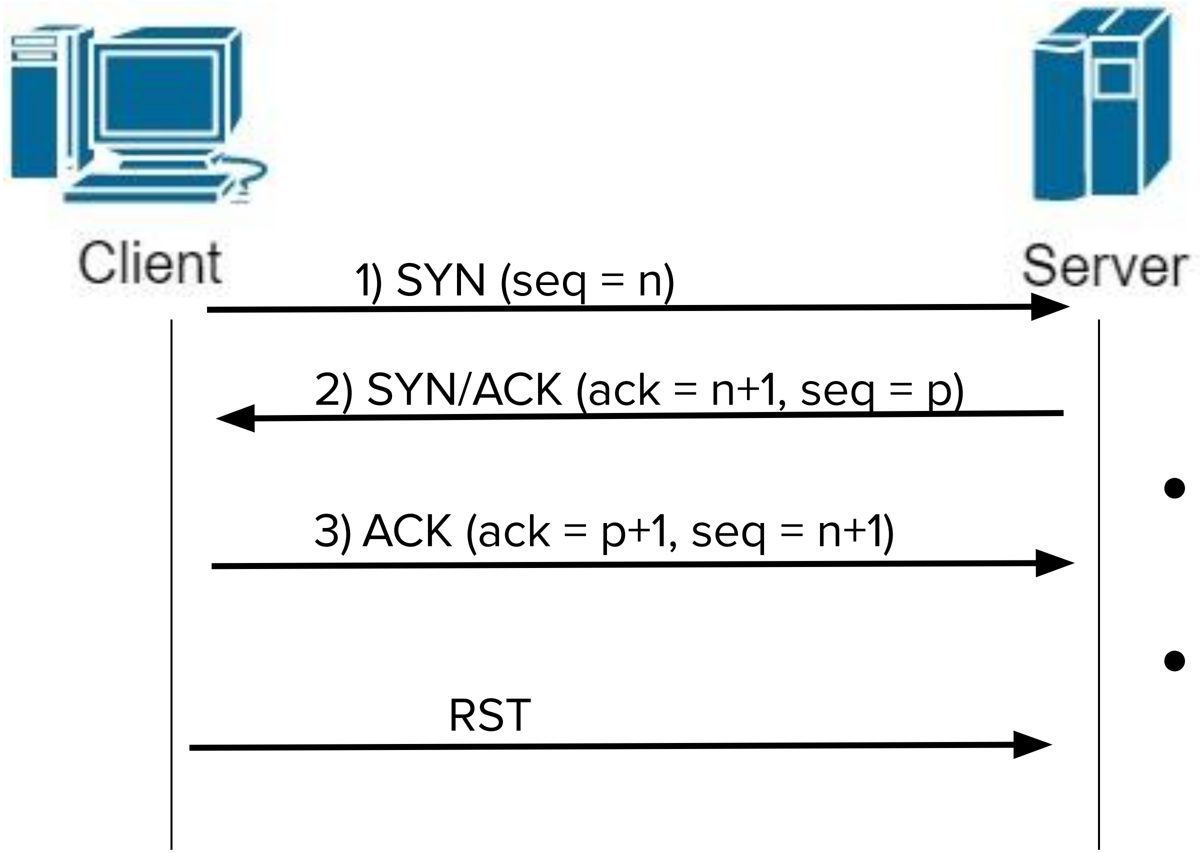


1) SYN (seq = n)

2) SYN/ACK (ack = n+1, seq = p)

3) ACK (ack = p+1, seq = n+1)

TCP Reset


Client
Server
1) SYN (seq = n)
2) SYN/ACK (ack = n+1, seq = p)
3) ACK (ack = p+1, seq = n+1)
RST

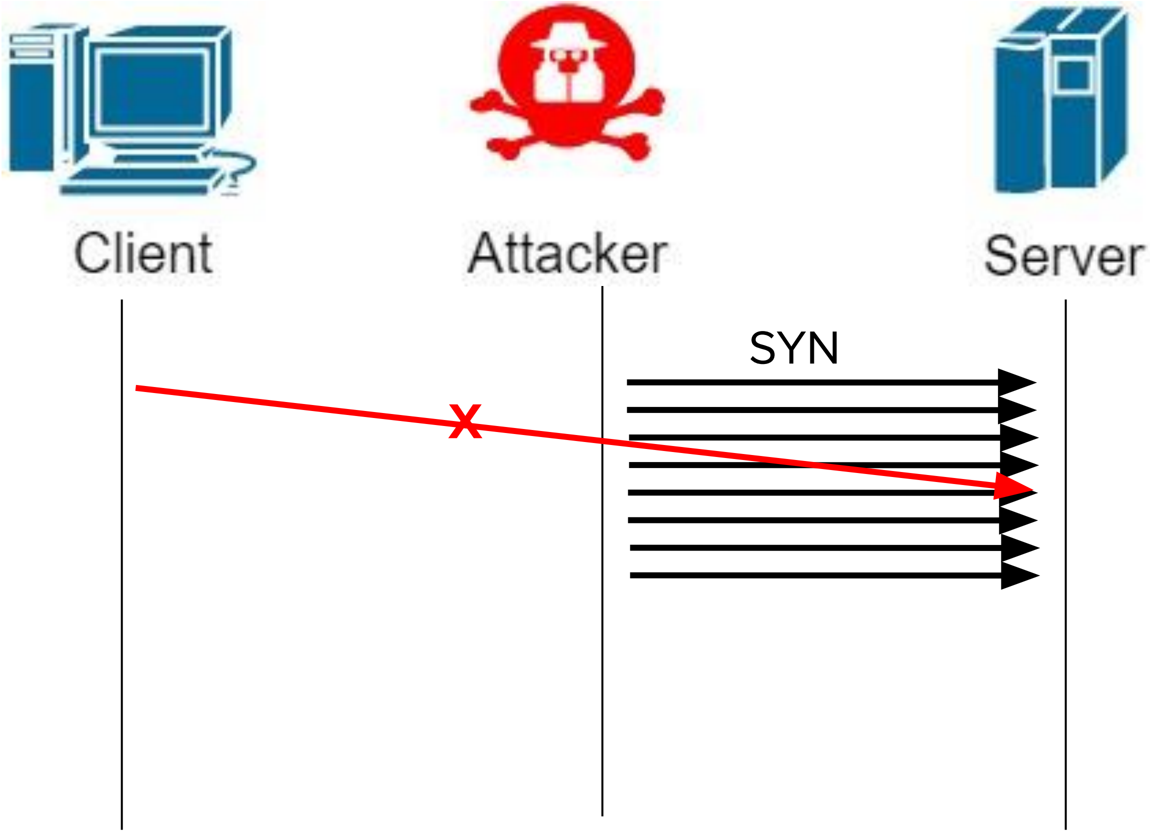
- With RST packet the client asks to terminate the existing TCP session
- Used for error reporting

Attacks to TCP

- Exploit native TCP special packets for malicious purposes
- The main attacks to TCP include:
  - SYN Flooding
  - RST attack
  - TCP Session Hijacking

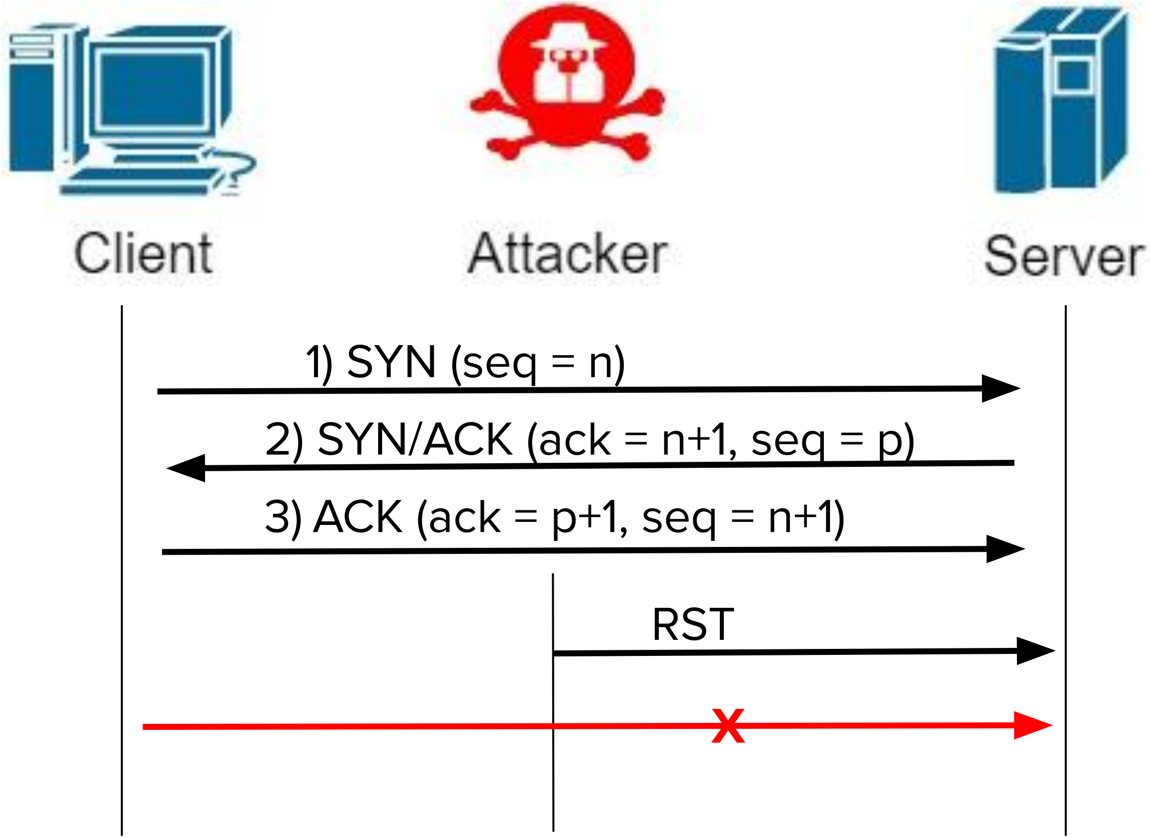
SYN Flood

- Upon receiving a SYN packet, the server waits for the SYN/ACK
- Fill server’s input queue with SYN packets
- Spoof the sender’s address to prevent detection



RST Attack

- The attacker knows the port and address of the victim (sniffing)
- The attacker sends an RST spoofing the victim's address
- When the victim sends a packet to the server, the server discards it

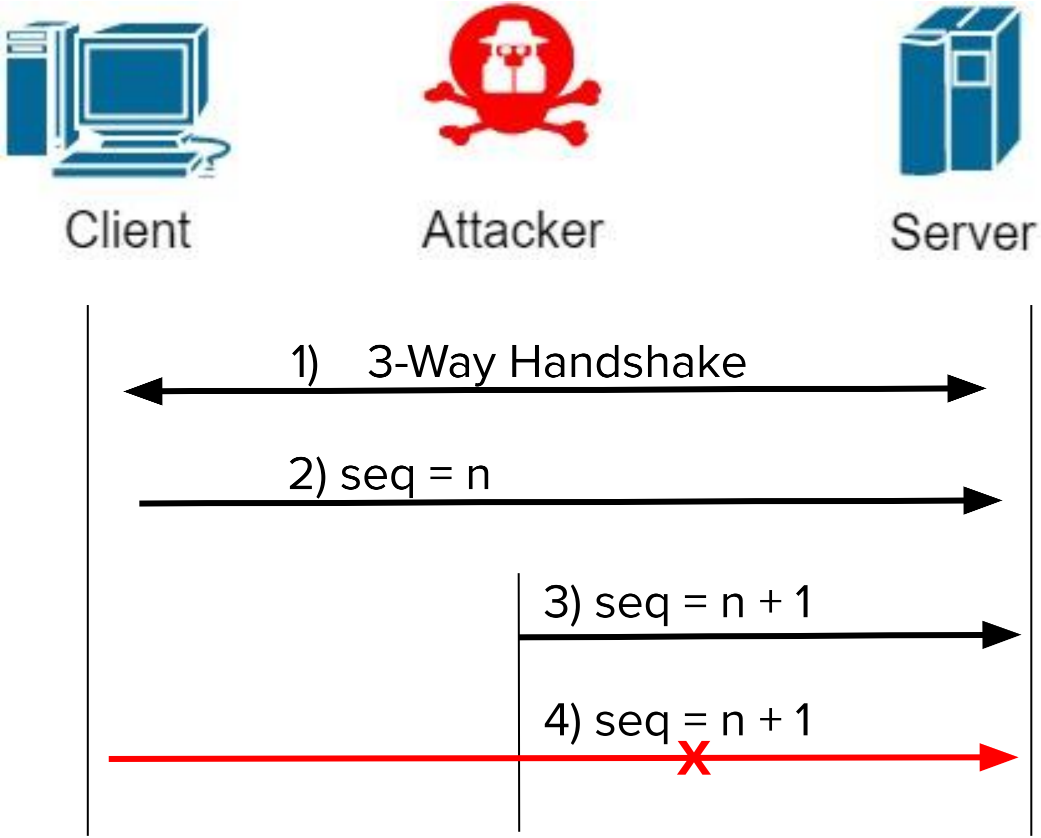

Client
Attacker
Server
1) SYN (seq = n)
2) SYN/ACK (ack = n+1, seq = p)
3) ACK (ack = p+1, seq = n+1)
RST
X

TCP Session Hijacking

- The attacker takes control of an existing TCP connection
- Exploits the sequence numbers update in subsequent packets
- The attacker must be able to sniff the connection to infer addresses and sequence numbers

TCP Session Hijacking

- The attacker uses the spoofed information to send a packet with the server's expected seq
- The client falls back on seq updates, and its packets get discarded
- The attacker controls the session



Countermeasures to TCP attacks

- Use a random initial sequence number to prevent its guessing by counting the number of exchanged packets
- Use secure alternatives (e.g., SSL/TLS)
- Use encryption to prevent packet sniffing
- Use anomaly/intrusion detection systems to identify spoofing attacks