

Uni IT Security Notes

Felix Pojtinger

June 4, 2021

Contents

Uni IT Security Notes	1
Basics	1
Security Mindset	1
Security Objectives	2
Attacks, Threats and Vulnerabilities	2
Threat Identification	2
Security Frameworks	3
Network Specific Threat Examples	3
STRIDE: Attacks on a Multi-User System	3
Security policies	3
Malware	3
Networking	3
TCP Overview	3
TCP Connection Establishment	4
IP Security Issues	4
TCP Security Issues	4
Port Scanning	5
TCP Protection Mechanisms	5
Session Hijacking	6
RST Attacks (In-Connection DoS)	6
Blind IP Spoofing	6

Uni IT Security Notes

Basics

Security Mindset

- Focus on weaknesses, not on features
- Don't rely on the "good case"
- Anticipate what an attacker could do to a system
- Weight security against user experience and privacy

Security Objectives

- **Confidentiality/conf**
 - Nobody but the legitimate receiver can read a message
 - Third party cannot gain access to communication patterns
- **Integrity/int**: The contents of communication can't be changed
- **Authenticity/authN**
 - **Entity Authentication**: Communication partners can prove their respective identity to one another
 - **Message Authentication**: It can be verified that a message is authentic (unaltered and sent by the correct entity)
- **Authorization/authZ**
 - Service or information is only available to those who have correct access rights
 - Depends on authentication being set up
- **Non-Repudiation/nRep**: A sender cannot deny having sent a message or used a service
- **Availability/avail**: Service is available with sufficient performance
- **Access Control/ac**: Access to services and information is controlled
- **Privacy/priv**
 - Restricted access to identity-related data
 - Anonymity
 - Pseudonymity

Attacks, Threats and Vulnerabilities

- **Attacker**: A person who has the skill and motivation to carry out an attack: The steps needed to carry out an attack
- **Vulnerability**: Some characteristics of the target that can result in a security breach
- **Threat**: Combination of an attacker, an attack vector and a vulnerability
- **Attack**: A threat that has been realized and has caused a security breach

Threat Identification

- Define **system boundaries**: What is part of your system, what is not?
- Define **security objectives**: What is important for your system to be secure?
- **List all threats** you can think of: Brainstorming and discussion with experts
- Use **conventions**:
 - Similar threat models
 - Requirement specifications
 - How to break or circumvent the specifications
 - Note security assumptions of the system
 - Be careful with perimeter security: What if perimeter has been breached?

- Note *possible*, but not yet exploitable vulnerabilities

Security Frameworks

Network Specific Threat Examples

- Remote Attacks
- Eavesdropping: Sniffing of information
- Altering information
- Spoofing
- DoS
- Session hijacking
- Viruses attacking clients
- Spam
- Phishing
- Data trails/privacy leaks

STRIDE: Attacks on a Multi-User System

- Spoofing of Identity
- Tampering with Information
- Repudiation
- Information Disclosure
- DoS
- Escalation of Privileges

Security policies

- Classification of system states into “allowed” and “forbidden” states
- Secure system: Is only in allowed states
- Breached system: Is in forbidden state

Malware

- Performs unwanted functions
- Often runs without user’s consent
- Telemetry (often hidden in proprietary software behind EULAs)
- Backdoors

Networking

TCP Overview

- Characteristics
 - Reliable
 - Connection-Oriented
 - Full-Duplex
 - Layer atop IP

- Connection management: Setup, Release and Abort
- Ordered delivery (package sequence control)
- Repetition of lost packets
- End-to-End ACKs
- Checksum in header
- Identified by a 5-tuple
 - Source IP
 - Destination IP
 - Transport Protocol
 - Source Port
 - Destination Port

TCP Connection Establishment

- Virtual connection between two systems
- 3-Way-Handshake with connection states

An example connection from the client to the server:

<Client>		<Server>
[Closed]		[Closed]
	SEQ=x CTL=SYN =>	
[SYN Sent]	<= SEQ=y CTL=SYN+ACK ACK=x+1	
		[SYN Received]
	SEQ=x+1 CTL=ACK ACK=y+1 =>	
[Established]		[Established]

IP Security Issues

- IP header doesn't have confidentiality or integrity protection
 - Faking the sender address is easy to do
 - Traffic can be analyzed by sniffing packet headers
- IP payload doesn't have confidentiality or integrity protection
 - Eavesdropping is possible by sniffing packets
- Loose coupling with lower layers:
 - Easy to divert traffic
 - Availability can be easily attacked
 - Confidentiality and integrity can't be guaranteed
- Unprotected error signaling via ICMP: Fake error messages can affect availability
- DNS is insecure; i.e. DNS spoofing

TCP Security Issues

- TCP header doesn't have confidentiality or integrity protection
- Session hijacking

- When sniffing session details, attacker can impersonate a peer in a TCP connection
 - Attackers can guess session details and attack remotely using spoofed IP addresses
- RST attack: Attackers can reset/abort attacks by injecting packets with the RST flag
- Port scanning
 - Find out open ports
 - Determine software running on port
- SYN flooding
 - Overload system resources by initializing many connections and not pursuing them

Port Scanning

- Objective: **Collect information**
 - Installed services
 - Software versions
 - OS
 - Firewall
- Enumeration based on port
 - Well-known ports (i.e. SSH → 22)
 - Invalid connection requests: Different way of error handling can be used to fingerprint the OS
- Possible scanning methods
 - TCP connect scan
 - Half-open scan
 - SYN-ACK scan
 - ACK scan

TCP Protection Mechanisms

- SYN flood protection
 - Limit rate of SYN packets
 - SYN cookies (RFC 4987)
 - * Limit resources
 - * Half-open connections are not stored in the connection table but instead as a hash in the ISN
 - * Only if the 3rd ACK handshake packet matches the sequence number, the connection is added to the connection table
 - * Server does not need to maintain any state information on half-open connections: Resources can't be exhausted
- Connections are only accepted if the sequence numbers are within a certain range of acceptable values (attackers would have to sniff sequence numbers or guess them)

Session Hijacking

- Attacker takes over existing connection between two peers
- Requirement: Attacker has to sniff or guess sequence numbers of the connection correctly

RST Attacks (In-Connection DoS)

Inject packet with RST flag into ongoing connection: Connection has to be aborted immediately

Blind IP Spoofing

Firewall is configured to only allow one source IP address and destination IP address ($A \rightarrow B$).

To circumvent this restriction:

1. Attacker starts DoS attack on A to prevent A from sending RST packets to B
2. Attacker sends TCP connection setup packet with A's source IP address to B
3. B sends SYN+ACK packet to A, but can't respond due to DoS
4. Attacker sends TCP connection ACK packet to B with ACK matching the initial sequence number chosen by B (which has to be guessed, as B sent the SYN+ACK packet to A, not the attacker)

Only works if B uses a predictable algorithm for its ISN and packet filters aren't in place.