scademy
secure coding academy

Network security

CL-NWS | Classroom | 2 days

Audience: Network engineers and developers
Preparedness: Network engineering, general software development
Exercises: Hands-on

Since all applications today heavily rely on communication and networks, there is no application security without network security.

This course focuses on network security with a software security viewpoint, and discusses common network attacks and defenses on different OSI layers, with an emphasis on application layer issues, tackling topics like session management or denial of service.

As cryptography is a critical aspect of network security, the most important cryptographic algorithms in symmetric cryptography, hashing, asymmetric cryptography, and key agreement are also discussed. Instead of presenting an in-depth mathematical and theoretical background, these elements are discussed from a merely practical, engineering perspective, showing typical use-case examples and practical considerations related to the use of crypto, such as public key infrastructures. Security protocols in many different areas of secure communication are introduced, with an in-depth discussion on the most widely-used protocol families such as IPSEC and SSL/TLS.

Outline:

IT security and secure coding
Network security
Practical cryptography
Security protocols
Cryptographic vulnerabilities
Knowledge sources

Participants attending this course will:

Understand basic concepts of security, IT security and secure coding

Learn about network attacks and defenses at different OSI layers

Have a practical understanding of cryptography

Understand essential security protocols

Understand some recent attacks against cryptosystems

Get information about some recent related vulnerabilities

Get sources and further readings on secure coding practices

Related courses:

- CL-ANW - Network security and secure communication (Classroom, 3 days)
- CL-PCR - Practical cryptography for software engineers (Classroom, 2 days)

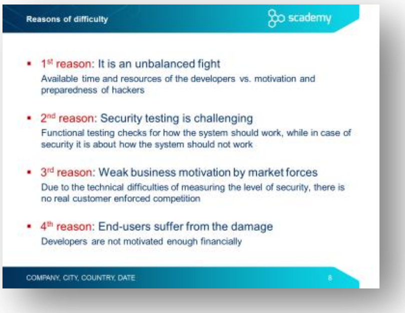
Note: Our classroom trainings come with a number of easy-to-understand exercises providing live hacking fun. By accomplishing these exercises with the lead of the trainer, participants can analyze vulnerable code snippets and commit attacks against them in order to fully understand the root causes of certain security problems. All exercises are prepared in a plug-and-play manner by using a pre-set desktop virtual machine, which provides a uniform development environment.

Detailed table of contents

Day 1

IT security and secure coding

- Nature of security
- What is risk?
- IT security vs. secure coding
- From vulnerabilities to botnets and cybercrime
    - Nature of security flaws
    - Reasons of difficulty
    - From an infected computer to targeted attacks



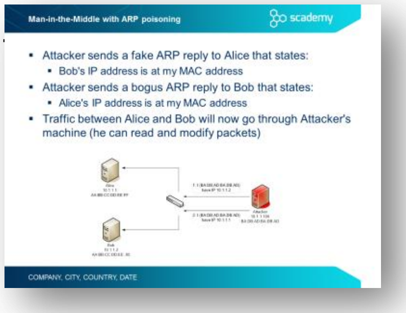
Reasons of difficulty

- 1st reason: It is an unbalanced fight
  Available time and resources of the developers vs. motivation and preparedness of hackers
- 2nd reason: Security testing is challenging
  Functional testing checks for how the system should work, while in case of security it is about how the system should not work
- 3rd reason: Weak business motivation by market forces
  Due to the technical difficulties of measuring the level of security, there is no real customer enforced competition
- 4th reason: End-users suffer from the damage
  Developers are not motivated enough financially

COMPANY, CITY, COUNTRY, DATE 8

Network security

- Overview
    - The TCP/IP stack
- Data Link layer
    - Sniffing attacks
        - What is a sniffer?
        - A revision on hubs and switches
        - MAC flooding
    - Spoofing
        - Spoofing attacks
        - Address Resolution Protocol (ARP)
        - ARP spoofing
        - Dynamic Host Configuration Protocol (DHCP)
        - DHCP starvation
    - Man-in-the-Middle
        - Man-in-the-Middle with ARP poisoning
        - Rogue DHCP server
    - Attacks against VLANs
        - VLANs, Native VLANs, DTP
        - VLAN hopping, Switch spoofing
        - Double tagging
    - Data Link layer protections
        - Segmentation
        - Detecting sniffing tools
        - VLAN security
        - Port Security



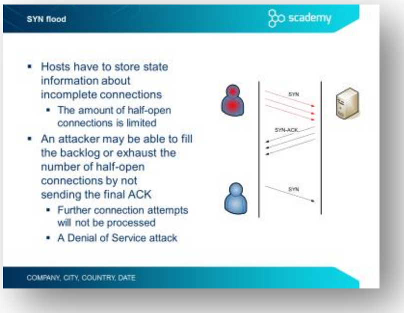
Man-in-the-Middle with ARP poisoning

- Attacker sends a fake ARP reply to Alice that states:
    - Bob's IP address is at my MAC address
- Attacker sends a bogus ARP reply to Bob that states:
    - Alice's IP address is at my MAC address
- Traffic between Alice and Bob will now go through Attacker's machine (he can read and modify packets)

Alice
10.1.1.1
AA:BB:CC:DD:EE:FF
1.1 (BA:DB:AD:BA:DB:AD) have IP 10.1.1.2
2.1 (BA:DB:AD:BA:DB:AD) have IP 10.1.1.1
Attacker
10.1.1.100
BA:DB:AD:BA:DB:AD
Bob
10.1.1.2

COMPANY, CITY, COUNTRY, DATE

        - DHCP snooping
        - Dynamic ARP Inspection (DAI)
        - Private VLANs
- Network layer
    - IP address spoofing
    - Maximum Transmission Unit
    - Fragmentation attack
    - ICMP attacks
        - Internet Control Message Protocol (ICMP)
        - Smurf attack
        - Ping of death
        - Route hijacking
    - Network layer protections
        - Ingress filtering, Egress filtering
        - IP Source Guard
        - Firewalls
        - Packet filtering firewalls
        - Intrusion Detection/Prevention Systems
- Transport layer
    - Transmission Control Protocol (TCP)
        - Transmission Control Protocol
        - SYN flood
        - TCP session hijacking



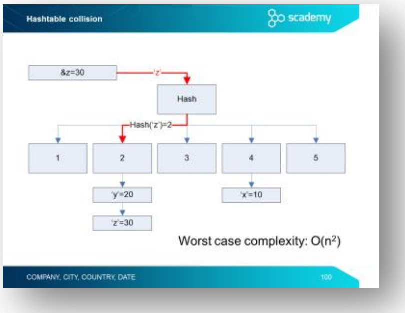
SYN flood

- Hosts have to store state information about incomplete connections
    - The amount of half-open connections is limited
- An attacker may be able to fill the backlog or exhaust the number of half-open connections by not sending the final ACK
    - Further connection attempts will not be processed
    - A Denial of Service attack

COMPANY, CITY, COUNTRY, DATE

    - User Datagram Protocol (UDP)
        - User Datagram Protocol
        - UDP flooding
    - Routing protocols
    - Fingerprinting and service detection
        - Nmap
        - Exercise – using Nmap
        - connect() scan
        - SYN scan
        - FIN scan
        - X-mas scan
    - Transport layer protection
        - SYN proxy
        - SYN cookies
        - Stateful firewalls
        - Routing protocol security

- Application layer
    - Domain Name System
    - DNS Spoofing
    - Session attacks
        - Session handling threats
        - Session handling best practices
        - Setting cookie attributes – best practices
    - Denial of services attacks
        - DoS introduction
        - Economic Denial of Sustainability (EDoS)
        - Asymmetric DoS
        - SSL/TLS renegotiation DoS
    - Hashtable collision attack
        - Using hashtables to store inputs
        - Hashtable collision



Hashtable collision

Worst case complexity: $O(n^2)$

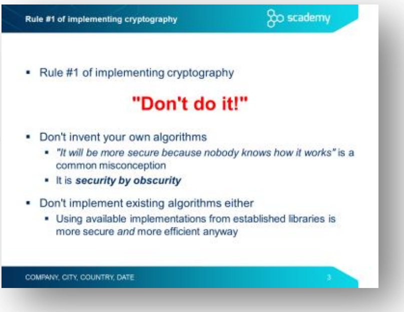
COMPANY, CITY, COUNTRY, DATE 100

-
    - Application layer protections
        - Application-level firewalls
        - Application layer security solutions
        - Secure protocols

Day 2

Practical cryptography

- Rule #1 of implementing cryptography



Rule #1 of implementing cryptography

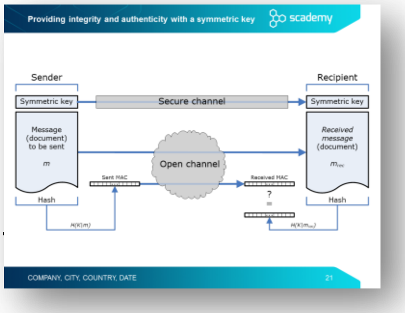
- Rule #1 of implementing cryptography

"Don't do it!"

- Don't invent your own algorithms
    - *"It will be more secure because nobody knows how it works"* is a common misconception
    - It is ***security by obscurity***
- Don't implement existing algorithms either
    - Using available implementations from established libraries is more secure *and* more efficient anyway

COMPANY, CITY, COUNTRY, DATE 3

- Cryptosystems
    - Elements of a cryptosystem
- Symmetric-key cryptography
    - Providing confidentiality with symmetric cryptography
    - Symmetric encryption algorithms
    - Modes of operation
- Other cryptographic algorithms
    - Hash or message digest
    - Hash algorithms
    - SHAttered
    - Message Authentication Code (MAC)
    - Providing integrity and authenticity with a symmetric key



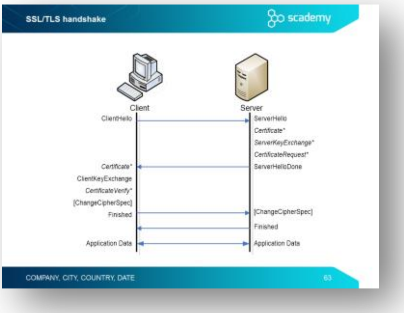
Providing integrity and authenticity with a symmetric key

COMPANY, CITY, COUNTRY, DATE 21

        - Random numbers and cryptography
        - Cryptographically-strong PRNGs
        - Hardware-based TRNGs
- Asymmetric (public-key) cryptography
    - Providing confidentiality with public-key encryption
    - Rule of thumb – possession of private key
    - The RSA algorithm
        - Introduction to RSA algorithm
        - Encrypting with RSA
        - Combining symmetric and asymmetric algorithms
        - Digital signing with RSA
- Public Key Infrastructure (PKI)
    - Man-in-the-Middle (MitM) attack
    - Digital certificates against MitM attack
    - Certificate Authorities in Public Key Infrastructure
    - X.509 digital certificate

Security protocols

- Secure network protocols
- Specific vs. general solutions
- IPSEC protocol family
    - IPSEC standards
    - Security Association (SA)
    - Message formats
    - AH packet structure
    - ESP packet structure
    - Protected channels
    - More complex set-ups
    - Traffic control
    - SA structure
    - Key management
- SSL/TLS protocols
    - Security services
    - SSL/TLS handshake ....

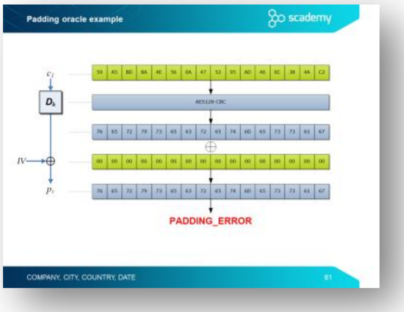


SSL/TLS handshake

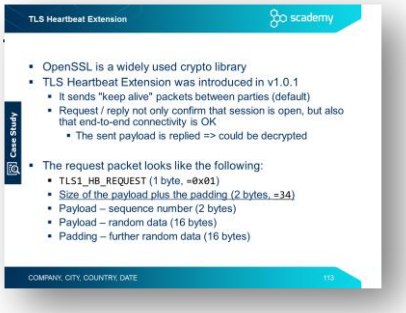
Cryptographic vulnerabilities

- Protocol-level vulnerabilities
    - BEAST

    - FREAK
    - FREAK – attack against SSL/TLS
    - Logjam attack
- Padding oracle attacks
    - Adaptive chosen-ciphertext attacks
    - Padding oracle attack
    - CBC decryption
    - Padding oracle example
    - Lucky Thirteen
    - POODLE


Padding oracle example
scademy
c_1
D_k
IV
p_1
59 A5 BD 8A 4E 56 8A 47 52 95 AD 46 0C 38 4A C2
AES128 CBC
76 65 72 79 73 65 63 72 65 74 6D 65 73 73 61 67
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
76 65 72 79 73 65 63 72 65 74 6D 65 73 73 61 67
PADDING_ERROR
COMPANY, CITY, COUNTRY, DATE
81

- RSA timing attack
    - Implementation of encoding/decoding in RSA
    - Fast exponentiation
    - Differences in execution times
    - RSA timing attack
    - Measurements
    - RSA timing attack – principles
    - Correlation of total and partial execution times
    - RSA timing attack – in practice
    - The RSA timing attack algorithm
    - Practical exploitation using the RSA timing attack
    - Attacking SSL/TLS servers
    - Protection against timing attacks
        - Hiding: RSA timing attack countermeasures
        - Masking: using blind signature
        - Real RSA implementations
- Implementation problems
    - Case study – Heartbleed
        - TLS Heartbeat Extension
        - Heartbleed – information leakage in OpenSSL
        - Heartbleed – fix in v1.0.1g


TLS Heartbeat Extension

- OpenSSL is a widely used crypto library
- TLS Heartbeat Extension was introduced in v1.0.1
    - It sends "keep alive" packets between parties (default)
    - Request / reply not only confirm that session is open, but also that end-to-end connectivity is OK
        - The sent payload is replied => could be decrypted
- The request packet looks like the following:
    - `TLS1_HB_REQUEST` (1 byte, =`0x01`)
    - Size of the payload plus the padding (2 bytes, =34)
    - Payload – sequence number (2 bytes)
    - Payload – random data (16 bytes)
    - Padding – further random data (16 bytes)

Knowledge sources

- Secure coding sources – a starter kit
- Vulnerability databases
- Recommended books – cloud security