b
UNIVERSITÄT
BERN

UNIVERSITÄT
BERN

Network Security

I. Introduction

Prof. Dr. Torsten Braun, Institut für Informatik

Bern, 21.02.2022 – 28.02.2022

Network Security: Introduction

Table of Contents

1. Concepts

1. Information and Network Security

Information Security

- Preservation of confidentiality, integrity, and availability of information
- Other properties like authenticity, accountability, non-repudation, reliability can be involved.

Network Security

- Protection of networks and their services from unauthorized modification, destruction, disclosure
- Provision that network performs functions currently and there are no harmful side effects

UNIVERSITÄT BERN

1. Concepts

2. Standardization Organizations

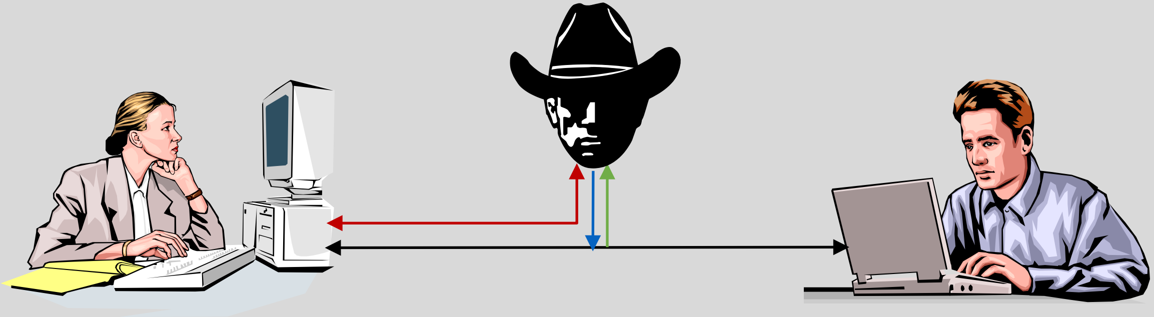
- National Institute of Standards and Technology
  - US federal agency
- Internet Society
  - Professional membership society
- International Telecommunication Union – Telecommunication
  - United Nations
- International Organization for Standardization (ISO)
  - Federation of national standardization organizations

UNIVERSITÄT BERN

1. Concepts

3. Key Security Objectives

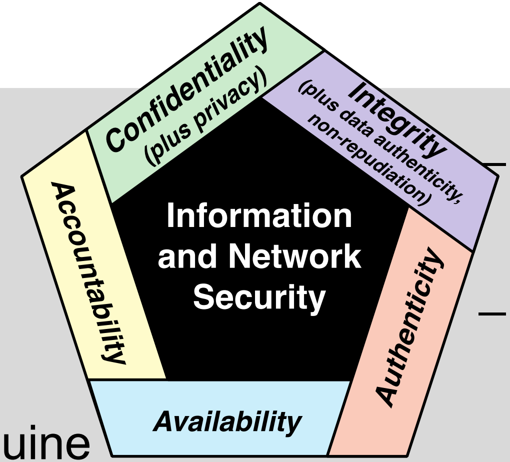
- Confidentiality
- Authenticity
- Integrity



UNIVERSITÄT BERN

1. Concepts

4. Essential Information/Network Security Objectives



- Confidentiality
  - Data confidentiality
  - Privacy
- Authenticity
  - Property of being genuine and being able to be verified
- Integrity
  - Data integrity
  - System integrity
- Availability
  - Timely and reliable access
- Accountability
  - Requirement for actions of an entity to be traced, including nonrepudiation, deterrence, fault isolation, intrusion detection etc.

1. Concepts

5. Terminology

OSI Terms

- Security attack
  - Actions compromising security of information
- Security mechanism
  - Process to detect, prevent, or recover from attacks
- Security service
  - Processing or communication service to enhance security using security mechanisms

Literature

- Threat
  - Circumstance or event with potential to impact organizational operations
- Attack
  - Malicious activity to collect, disrupt, deny, degrade, or destroy information or system resources

1. Concepts

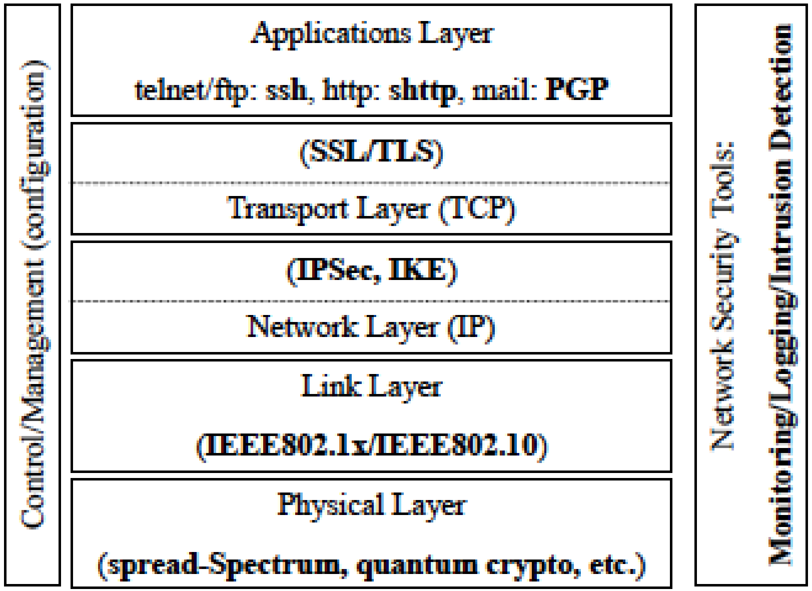
6. Security Design Principles

- Economy of mechanism, complexity
- Fail-safe defaults
- Complete mediation
- Open design
- Separation of privilege
- Least privilege
- Least common mechanism
- Psychological acceptability
- Isolation
- Encapsulation
- Modularity
- Layering
- Least astonishment

1. Concepts

7. Securing Networks

- Where to put the security in a protocol stack?
- Practical considerations:
  - End to end security
  - No modification to operating system



1. Concepts

8. Device Security

Concern

– Intruders gain access to network devices or end systems.

Systems

– Firewall
  – Hardware / software system limiting access between network and devices attached to network
– Intrusion detection
  – Analysis of network traffic to find malicious access attempts
– Intrusion prevention
  – Stopping of malicious activities after detection

2. Security Attacks

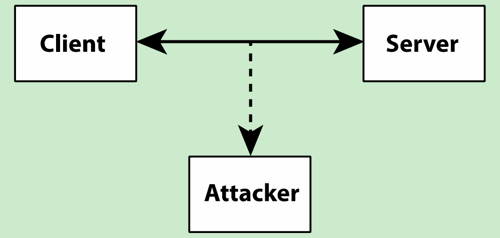
1. Attacks and Concepts

– Interception (confidentiality)
– Interruption (availability)
– Modification (integrity)
– Fabrication (authenticity)

2. Security Attacks

2.1 Kent‘s Classification: Passive Attacks

- Packet eavesdropping,
  e.g. packet sniffing: detection of data (e.g., passwords, credit card numbers) in routers or unprotected transmission media
- Traffic analysis:
  detection of end points and traffic type, e.g., addresses, packet lengths

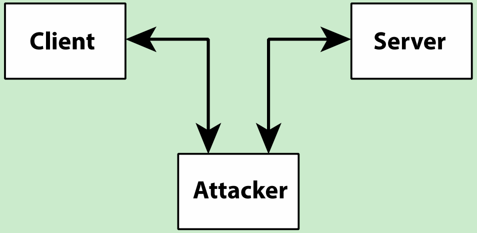


(b) Passive attack (eavesdrop, traffic analysis)

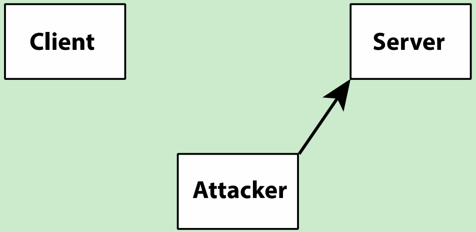
2. Security Attacks

2.2 Kent‘s Classification: Active Attacks

- Imitation of wrong identities (masquerading), e.g. IP Spoofing: use of a foreign IP address
- Modification of messages
- Replay attacks, i.e. repeated data transmission
- Denial-of-Service attacks
  - Blocking of network or server functions
  - Repetition of TCP SYN packets: Server allocates resources for TCP connection.



(c) Active attack (masquerade, data modification)



(d) Active attack (masquerade, data modification, denial of service)

2. Security Attacks

3. Surfaces

Examples

- Open ports in servers
- Services available inside a firewall
- Code processing incoming data
- Interfaces, SQL, web forms
- Employees

Categories

- Network
- Software
- Humans

3. Security Services and Mechanisms

1. Network Security Services: X.800, RFC 2828

- Peer-entity and data-origin authentication
  - assures the recipient of a message the authenticity of the claimed source or the entity connected.
- Access control
  - limits the access to authorized users.
- Data confidentiality
  - protects against unauthorized release of message content.
- Data integrity
  - guarantees that a message is received as sent.
- Non-repudiation
  - protects against sender/receiver denying sending/receiving a message.
- Availability
  - guarantees that the system services are always available when needed.
- Security audit
  - keeps track of transactions for later use (diagnostic, alarms…).
- Key management
  - allows to negotiate, setup and maintain keys between communicating entities.

UNIVERSITÄT BERN

3. Security Services and Mechanisms

2. Security Mechanisms

- Cryptographic algorithms (reversible, non-reversible)
- Data integrity
- Digital signatures
- Authentication exchange
- Traffic padding
- Routing control
- Notarization
- Access control

3. Security Services and Mechanisms

3. Cryptographic Algorithms

- Keyless Algorithms
  - Cryptographic hash functions
  - Cryptographic random number generation
- Single-Key Algorithms
  - Symmetric Encryption (e.g., AES)
  - Message Authentication Codes (e.g., HMAC)
- Two-Key Algorithms
  - Asymmetric Encryption (e.g., RSA)
  - Digital Signature (e.g., RSA)
  - Key Exchange
  - User Authentication

UNIVERSITÄT BERN

3. Security Services and Mechanisms

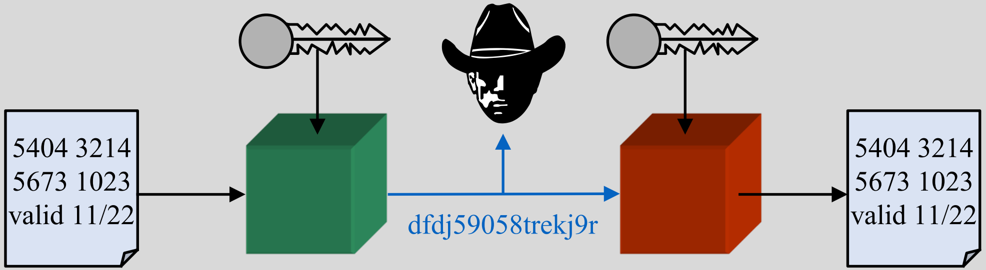
4. Relationship of Security Services and Mechanisms

Service \ Mechanism	Encipherment	Digital signature	Access control	Data integrity	Authentication exchange	Traffic padding	Routing control	Notarization
Peer entity authentication	Y	Y			Y			
Data origin authentication	Y	Y						
Access control			Y					
Confidentiality	Y						Y	
Traffic flow confidentiality	Y					Y	Y	
Data integrity	Y	Y		Y				
Nonrepudiation		Y		Y				Y
Availability				Y	Y			

4. Encryption

1. Operation

- Communication over an insecure channel
  - Encryption by sender
  - Decryption by receiver
- Attacker must not be able to understand the communication.



4. Encryption

2. Algorithm Types

Block Ciphers

- Input: block of n bits
- Output: block of n bits
- Example: AES

Block ciphers can be used to build stream ciphers.

Stream Ciphers

- Input: stream of symbols
- Output: stream of symbols
- Example: GSM

4. Encryption

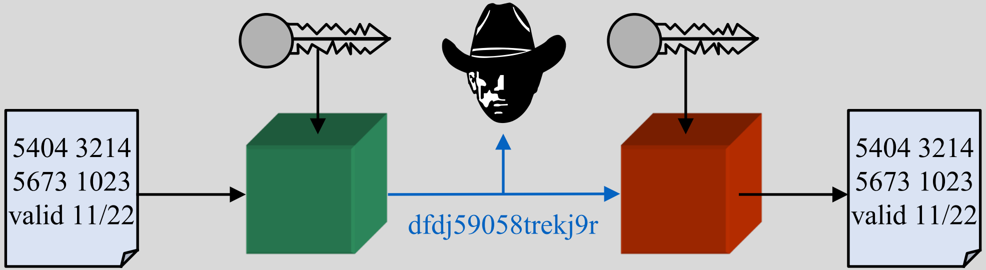
3. Models

Symmetric Encryption

- Encryption Key = Decryption Key
- Decrpytion key can be derived from encryption key.
- Example: AES

Asymmetric Encryption

- Encryption key ≠ Decryption key
- Decryption key can not be derived from encryption key.
- Example: RSA



UNIVERSITÄT BERN

4. Encryption

4. Symmetric vs Asymmetric Algorithms

- Symmetric algorithms are much faster, e.g. in the order of a 3 magnitudes, i.e. 1000 times faster
- Symmetric algorithms require a shared secret, which is impractical, if the communicating entities do not have another secure channel.
- Both types of algorithms are combined to provide practical and efficient secure communication, e.g.,
  - establish a secret session key using asymmetric crypto and
  - use symmetric crypto for encrypting the traffic

4. Encryption

5. Kerchoff's Principle

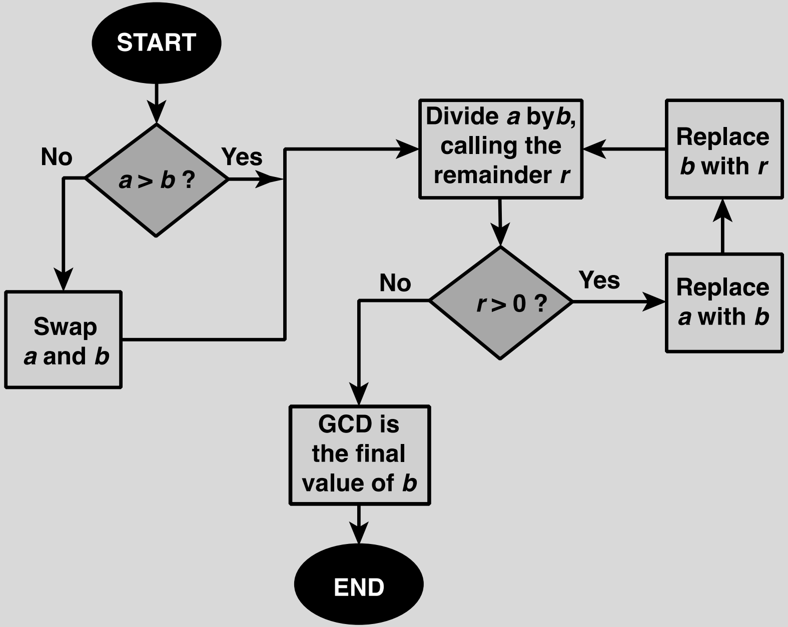
A cipher should be secure even if the intruder knows all the details of the encryption process except for the secret key.

"No security by obscurity"

5. Number Theory

1. Finding Prime Numbers: Euclid Algorithm

- To find greatest common divisor of two integers
- Example: gcd (595, 408) = 17
  - 595 / 408 = 1 remainder 187
  - 408 / 187 = 2 remainder 34
  - 187 / 34 = 5 remainder 17
  - 34 / 17 = 2 remainder 0



5. Number Theory

2.1 Fermat Theorem

If p is prime and
a (> 0) is not divisible by p:

$$a^{p-1} = 1 \ (mod\ p)$$

Examples:

- $a = 5, p = 3$:
  $5^{(3-1)} = 5^2 = 25 = 8 \cdot 3 + 1$
- $a = 7, p = 3$:
  $7^{(3-1)} = 7^2 = 49 = 16 \cdot 3 + 1$

Alternate form:

If p is prime and a > 0

$$a^p = a \ (mod\ p)$$

UNIVERSITÄT BERN

5. Number Theory

2.2 Proof of Fermat's Theorem

- Set of positive integers < p:
  P = {1, 2, …, p-1} and multiply by (a modulo p)
  → X = {a mod p, 2a mod p, 3a mod p, …, (p-1) a mod p}
- No element of X is 0 because p does not divide a *and* none of two integers of X are equal, i.e. X = set of positive integers < p: X = {1, 2, …, p-1} in some order
- Proof:
  - Otherwise: ∃ j, k ($1 \leqq j<k \leqq p-1$): $j \cdot a = (k \cdot a) \pmod p$
  - Since a is relatively prime to p, we can eliminate a: j = k (mod p), which is impossible, since j, k < p

Multiplying sets X and P and taking the result (mod p) yields:

- $a \cdot 2a \cdot 3a \cdot \ldots \cdot (p-1)\, a = (1 \cdot 2 \ldots \cdot (p-1)) \pmod p$
- $a^{p-1} (p-1)! = (p-1)! \pmod p$
  ((p-1)! is relatively prime to p)
- $a^{p-1} = 1 \bmod p$

5. Number Theory

3. Euler’s Totient Function ø(n)

ø(n): the number of positive integers less than *n* and relatively prime to *n.*

Two integers are relatively prime,
if their only common integer factor is 1.

n	$\phi(n)$
1	1
2	1
3	2
4	2
5	4
6	2
7	6
8	4
9	6
10	4

n	$\phi(n)$
11	10
12	4
13	12
14	6
15	8
16	8
17	16
18	6
19	18
20	8

n	$\phi(n)$
21	12
22	10
23	22
24	8
25	20
26	12
27	18
28	12
29	28
30	8

UNIVERSITÄT BERN

5. Number Theory

4. Euler's Theorem

For every *a* and *n* that are relatively prime:

$$a^{ø(n)} = 1(mod\ n)$$

An alternative form is:

$$a^{ø(n)+1} = a(mod\ n)$$

b UNIVERSITÄT BERN

5. Number Theory

5.1 Miller-Rabin Algorithm

used to test a large number n for primality,
probability for failed test < ¼, $(¼)^{10} < 10^{-6}$

```
Find integers k, q, with k > 0, q odd, so that (n – 1)=2^k q ;
Select a random integer a, 1 < a < n – 1 ;
if a^q mod n = 1 then return (“inconclusive") ;
for j = 0 to k – 1 do
        if (a^(2^j·q) mod n = n – 1) then
                return (“inconclusive") ;
return (“composite") ;
```

5. Number Theory

5.2 Miller-Rabin Algorithm

Example: n = 29

$n - 1 = 28 = 2^2 (7) = 2^k q$

a = 10:

- $10^7 \bmod 29 = 17$
- $(10^7)^2 \bmod 29 = 28$
- Test returns inconclusive

Example: n = 221 = 13 · 17

$n - 1 = 220 = 2^2 (55) = 2^k q$

a = 21:

- $21^{55} \bmod 221 = 200$
- $(21^{55})^2 \bmod 221 = 220$
- Test returns inconclusive !!

b UNIVERSITÄT BERN

5. Number Theory

6. Deterministic Primality Algorithm

– Prior to 2002 there was no known method of efficiently proving the primality of very large numbers.
– All algorithms in use produced a probabilistic result.

– In 2002 Agrawal, Kayal, and Saxena (AKS) developed an algorithm that efficiently determines whether a given large number is prime.
– It does not appear to be as efficient as the Miller-Rabin algorithm.

5. Number Theory

7. Discrete Logarithm

- $a^m = 1 \ (mod\ n)$
- If a and n are relatively prime, there is at least one m = $ø(n)$ satisfying the equation above
- $\log_b a$ is an integer k (***discrete logarithm:*** *the smallest one*) such that $b^k = a$.

Consider y = g^x mod p

- y is straight forward to calculate, i.e. at worst by x multiplications
- However, given y, g, p, x is difficult to calculate, in particular for large primes. This is used for Diffie-Hellman key exchange.

Thanks

for Your Attention

UNIVERSITÄT BERN

Prof. Dr. Torsten Braun, Institut für Informatik

Bern, 21.02.2022 – 28.02.2022