Sicherheit in Kommunikationsnetzen
(Network Security)

Introduction

Dr.-Ing. Matthäus Wander

Universität Duisburg-Essen

Definitions

- Security („Sicherheit“ or „Angriffssicherheit“)
  - Protect a system from threats
- Threat („Bedrohung“): potential harm that might lead to the violation of a security goal
- Attack: a sequence of actions to realize a threat, i.e. an attempt to violate a security goal
- Subject of this lecture: network security
  - Protect a system from threats that involve the transmission of information over a computer network

Threat

- Note: threat ≠ thread
  - Threat: potential security harm
  - Thread: execution of program instructions (in the context of parallel computing or operating systems)
- Example security threats
  - Breaking into a corporate computer and copying data
  - Interception of emails in transit
  - Manipulation of payment orders sent over the network
  - Shutting down a website by temporary overloading
  - Impersonation as another person in online shopping

Out of scope of this lecture

- We won‘t cover all aspects of computer security
  - e.g. malware, phishing, code injection, privilege escalation, security bugs, forensics
- No safety („Sicherheit“ or „Betriebssicherheit“)
  - Protect the environment/human from a system
  - e.g. an electronic machine with sharp blades
- Systems may have security **and** safety threats
  - e.g. exploit a car‘s network interface (security threat) to manipulate its driving functions (safety threat)
- No data privacy („Datenschutz“)

Security Goals

- Two perspectives on security goals
- Application-specific security goals
  - How the product manager sees it
- Technical security goals
  - How the engineer sees it

Application-specific Security Goals

- Banking
  - Protect from fraudulent or accidental modification of payment orders
  - Identify and authorize customers
  - Protect customer data from disclosure
- Electronic trading
  - Assure source and authenticity of trade orders
  - Protect corporate data
  - Provide legally binding electronic signatures on trades

Application-specific Security Goals (2)

- Government
  - Ensure privacy of citizen and corporate data
  - Protect from disclosure of state secrets
  - Provide electronic signatures on government documents
- University
  - Protect from disclosure of student and staff data
  - Ensure students cannot manipulate their grades
  - Deduct cafeteria payments from the proper student account

Technical Security Goals

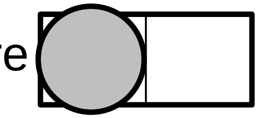
- Confidentiality („Vertraulichkeit“)
  - Reveal transmitted or stored data only to an intended audience (i.e. to authorized persons)
  - Mechanism: encryption
- Data integrity („Datenintegrität“)
  - Detect any modification/manipulation of data
- Authenticity („Authentizität“)
  - Verify that the data originates from the real source
- Integrity and authenticity usually used together

Technical Security Goals (2)

- Availability („Verfügbarkeit“)
  - Services should be available and function correctly
- Accountability („Zurechenbarkeit“)
  - Identify the person responsible for an action („who did this?“)
- Non-repudiation („Nichtabstreitbarkeit“)
  - Ensure the person cannot dispute their action („I never said that“).

Risk Assessment

- Security is not a binary option



- Security measures increase the security level
  - But also costs and effort
    („Password must be at least 10 characters long, consisting of numbers, mixed case and Egyptian hieroglyphics.“)

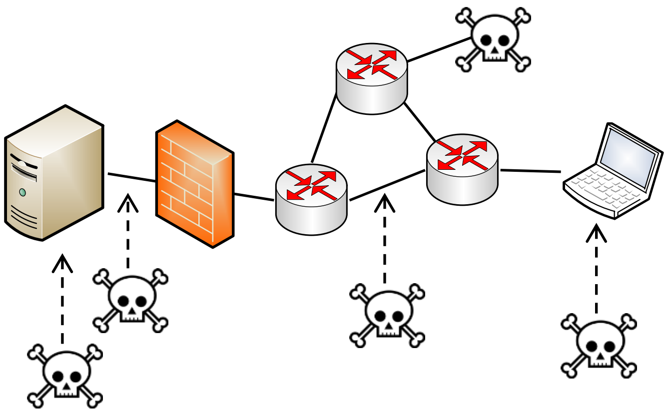
- There is no perfectly secure system
  - But we can increase the security level to reduce the residual risk („Restrisiko“)

Risk Assessment (2)

- Identify risks
  - Assets („Güter“): what do we want to protect?
  - Threats: what threats do we face?
- For each risk scenario, estimate:
  - Impact/consequences („how bad is it if it happens?“)
  - Likelihood of occurrence (cost of attack)
  - Usually in qualitative categories, not an exact science (e.g. low/medium/high)
- Take appropriate security measures subject to risk assessment

Threat Model

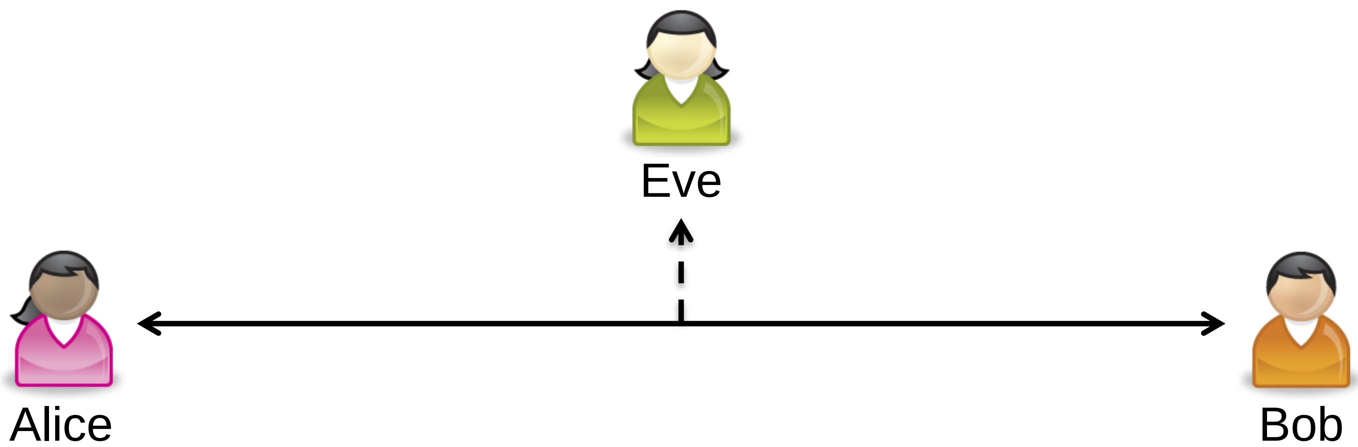
- Where is the attacker located?
  - Is it a user on our server?
  - Inside of local network?
  - Along our communication path?
  - Anywhere else in the Internet?



- What are the capabilities of the attacker?
- Two classes of network attacks
  - Passive attacks
  - Active attacks

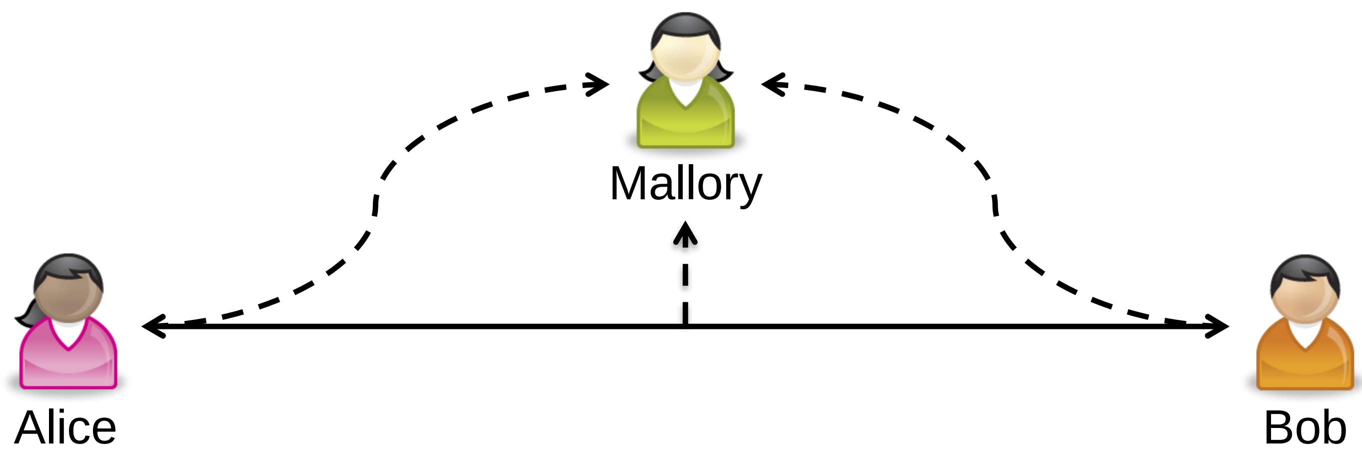
Passive Attacks

- Alice and Bob communicate over a network
- Attacker Eve is eavesdropping (listening)
  - Sees packet headers and message contents
  - e.g. in public WiFi or at Internet access provider



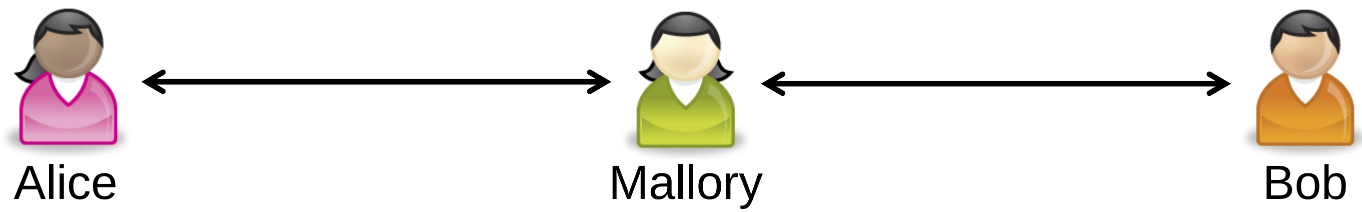
Active Attacks

- Attacker Mallory actively sends messages
  - Impersonation: claim to be another participant
  - Replay: retransmit old messages
  - Spoofing: send forged messages
  - Denial of service: crash or overload a network service



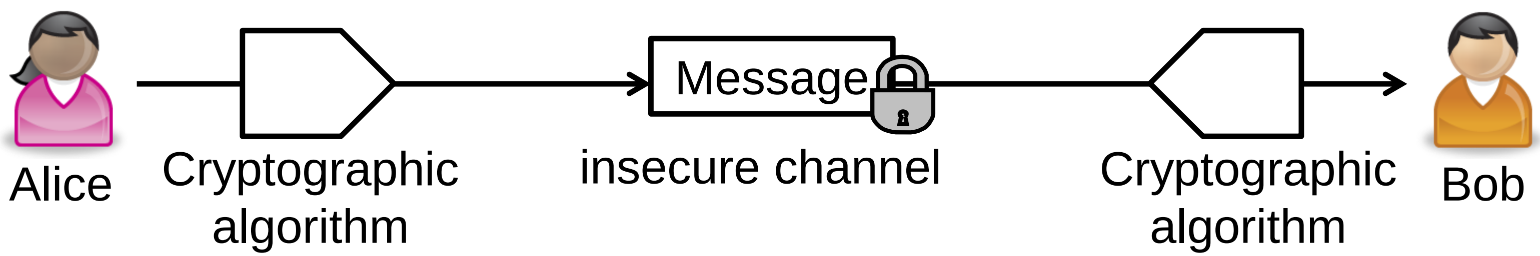
Active Attacks (2)

- Man-in-the-middle attack
  - Drop messages
  - Modify messages before forwarding them
  - Delay or re-order messages
  - Intercept and replace messages
  - Mallory may impersonate herself as Alice or Bob



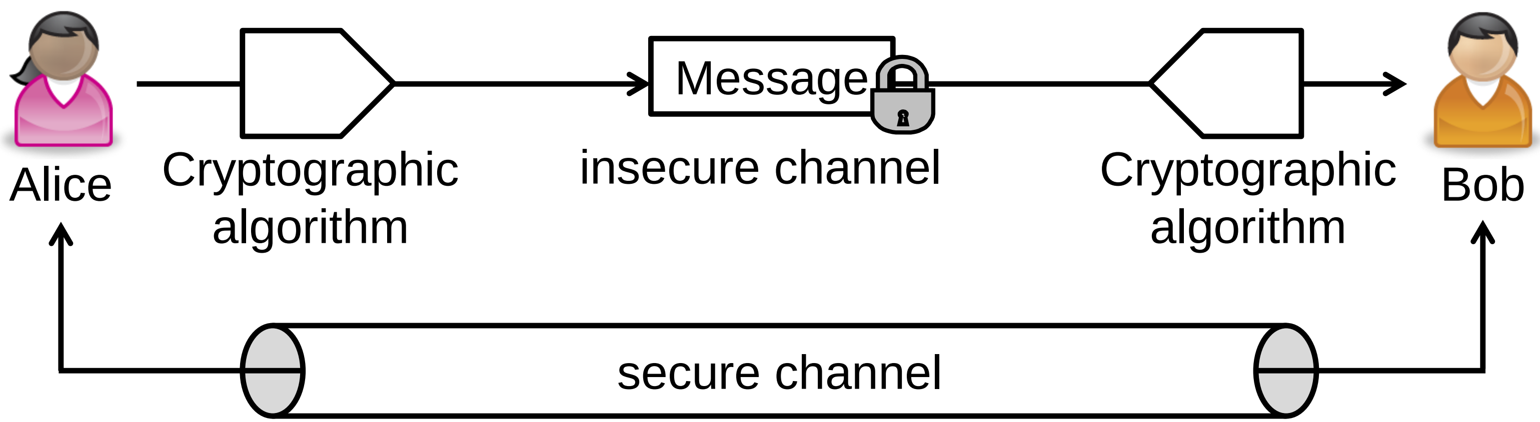
Model for Network Security

- Alice and Bob use a cryptographic algorithm to transform data when sending/receiving
- Messages are sent over an insecure channel



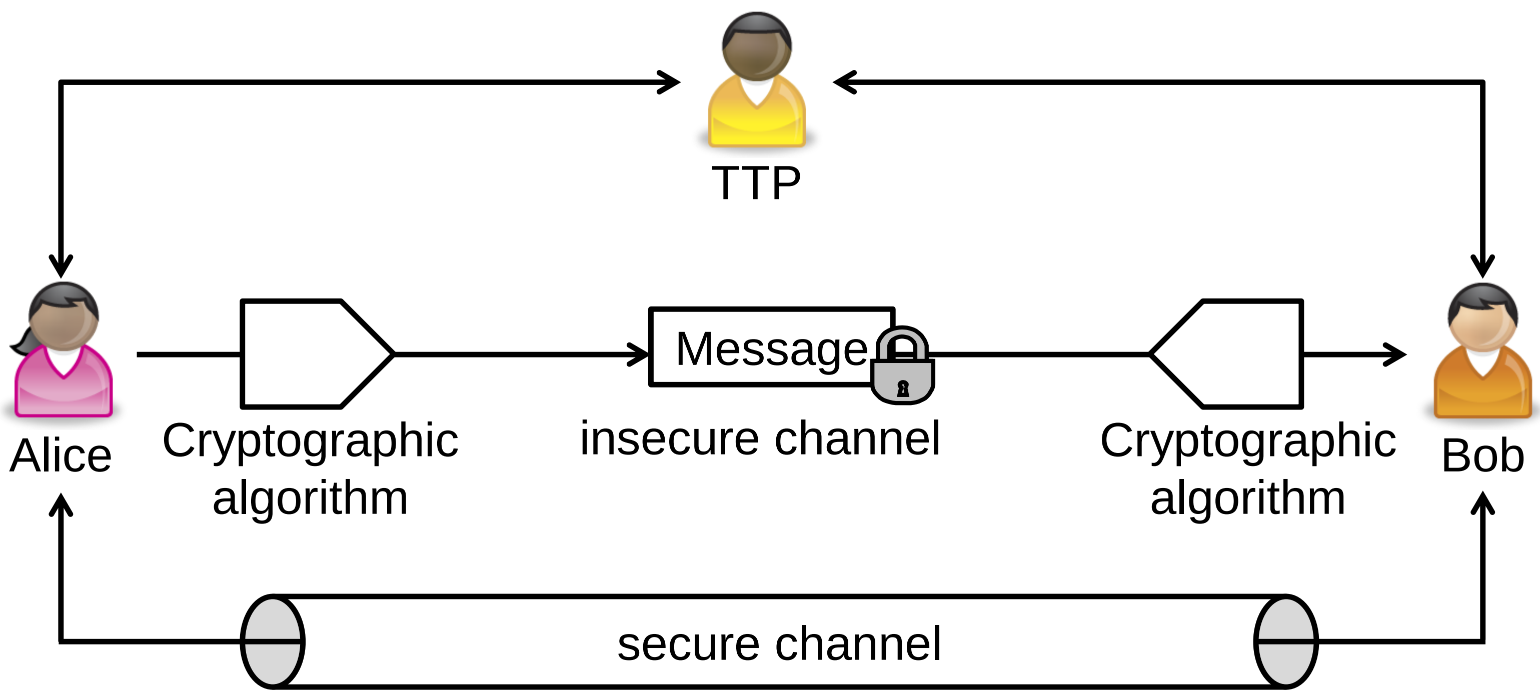
Model for Network Security (2)

- Alice and Bob have a secure channel to exchange cryptographic keys
  - But this channel is slow or expensive (e.g. in-person meeting, courier)

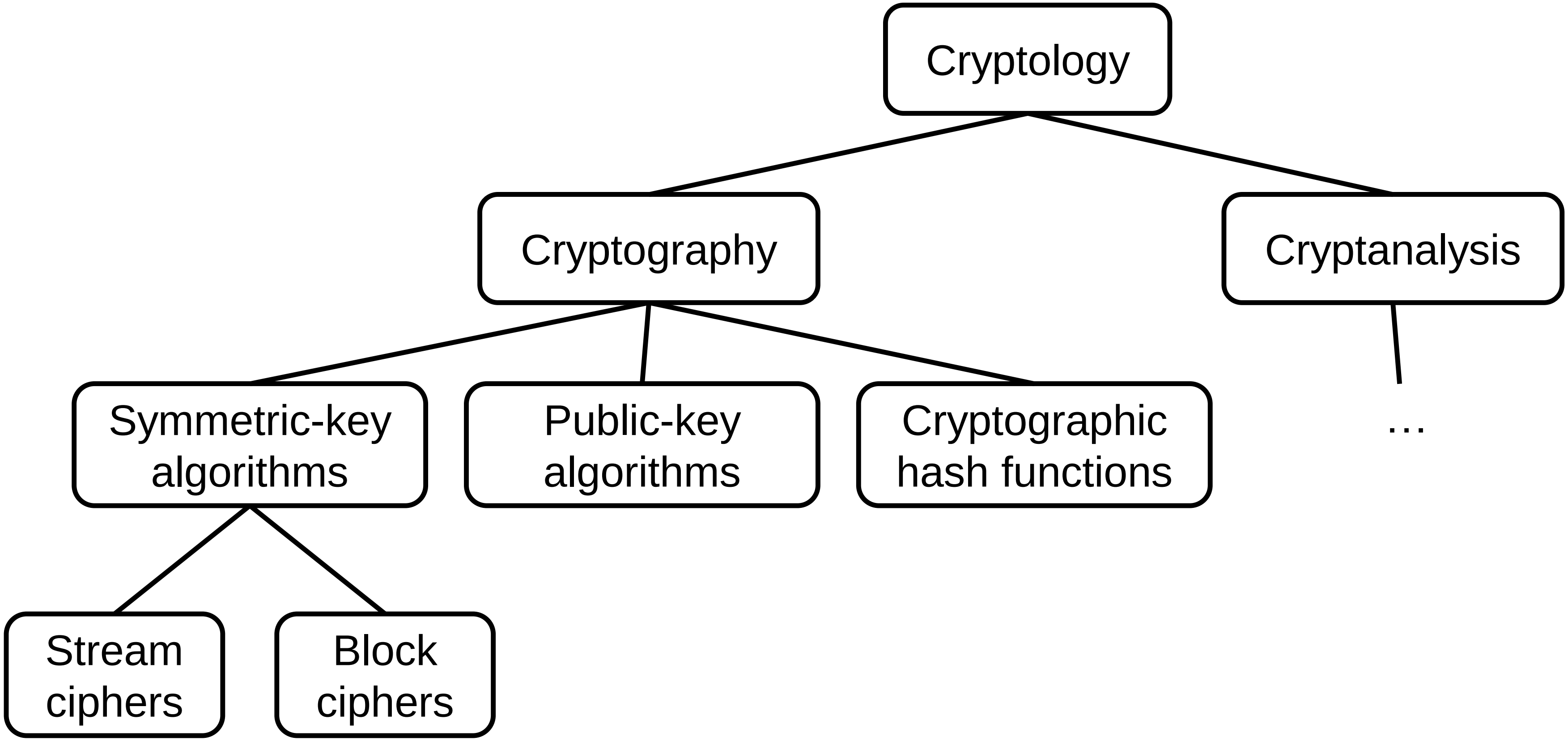


Model for Network Security (3)

- There may be a trusted third party (TTP)



Cryptography: Overview

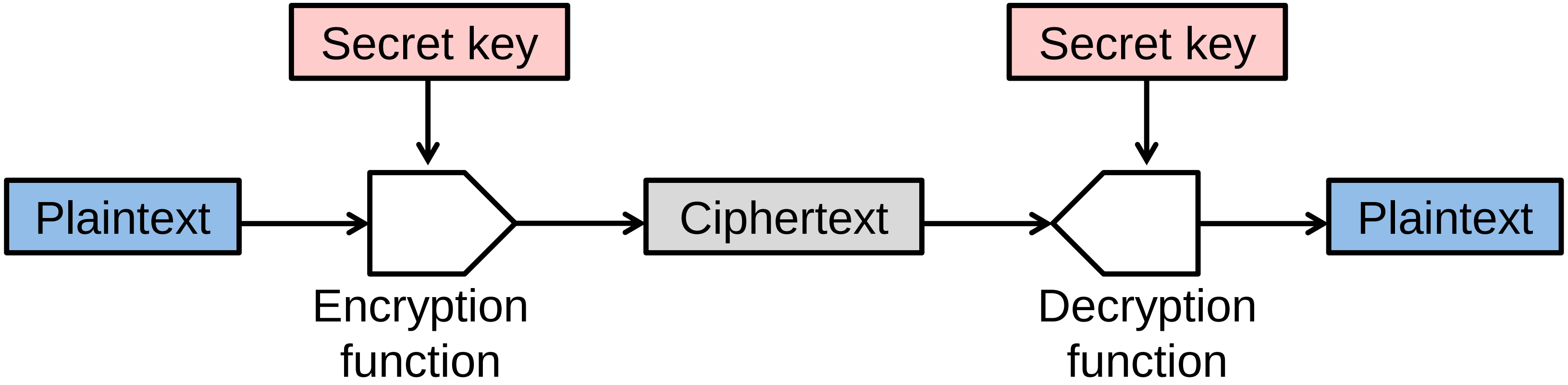


Cryptography: Definitions

- Cryptography („Kryptographie“) is one of the building blocks to achieve a security goal
  - Derived from Greek κρυπτὸς (kryptós: „hidden“) and γράφειν (gráphein: „writing“)
  - Study of techniques to conceal a message (encryption) so that it cannot be read by unauthorized entities
- Cryptographic algorithm or primitive
  - Transforms input data (message, key) to output data
- Cryptographic protocol
  - Sequence of steps to perform a security function

Encryption and Decryption

- Cipher: algorithm for encryption and decryption
- Plaintext: original message
- Ciphertext: encrypted, unreadable message
- Secret key: information used to encrypt the plaintext and required to decrypt the ciphertext



Formalization of Encryption

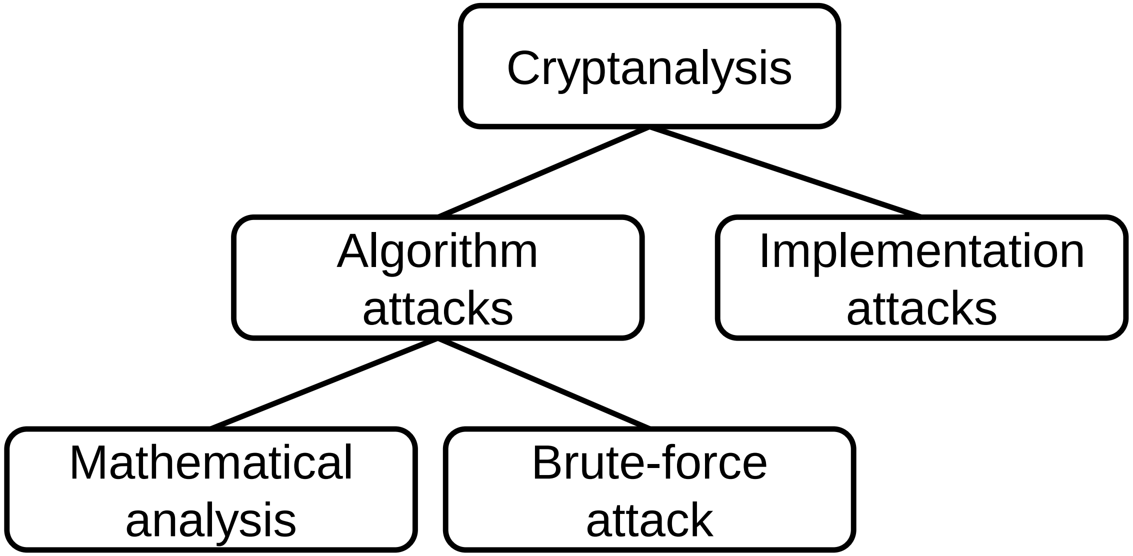
- Secret key k: sequence of characters or bits
  - Bits from {0, 1}; characters from an alphabet $\mathcal{A}$
- Key space $\mathcal{K}$: all possible keys $k \in \mathcal{K}$
  - Number of possible keys with n bits: $|\mathcal{K}| = 2^n$
  - $|\mathcal{K}| = |\mathcal{A}|^n$ e.g. with $\mathcal{A} := \{A, \ldots, Z\}$ ⇨ $|\mathcal{K}| = 26^n$
- Plaintext m: sequence of characters or bits
  - Space of possible plaintext messages: $\mathcal{M}$
- Ciphertext c: sequence of character or bits
  - Space of possible ciphertexts: $\mathcal{C}$

Formalization of Encryption (2)

- Encryption function e : $\mathcal{K} \times \mathcal{M} \rightarrow \mathcal{C}$
- Decryption function d : $\mathcal{K} \times \mathcal{C} \rightarrow \mathcal{M}$
- Short notation: we write $e_k(m) = c$
  instead of $e(k, m) = c$
- d_k is the inverse function of e_k for all $k \in \mathcal{K}$
- Thus: $d_k(e_k(m)) = m$ for all $k \in \mathcal{K}$, $m \in \mathcal{M}$
- We transfer $e_k(m) = c$ over an insecure channel
  - Without knowledge of k, Eve cannot recover $d_k(c)=m$

Cryptanalysis

- Cryptanalysis („Kryptoanalyse“)
  - Study of techniques to break cryptographic algorithms
  - e.g. recover plaintext from ciphertext without key



Cryptanalysis (2)

- Implementation attacks
  - Side-channel attacks leak information about the implementation‘s execution state (and thus the key)
  - via e.g. CPU power consumption, ultrasonic noise, execution timings, cache timings
  - Usually requires physical access to machine
- Brute-force attack: exhaustive search
  - Attempt decryption with every possible key
  - Attacker is guaranteed to find correct key eventually
  - Practical remedy: use very large key space

Cryptanalysis (3)

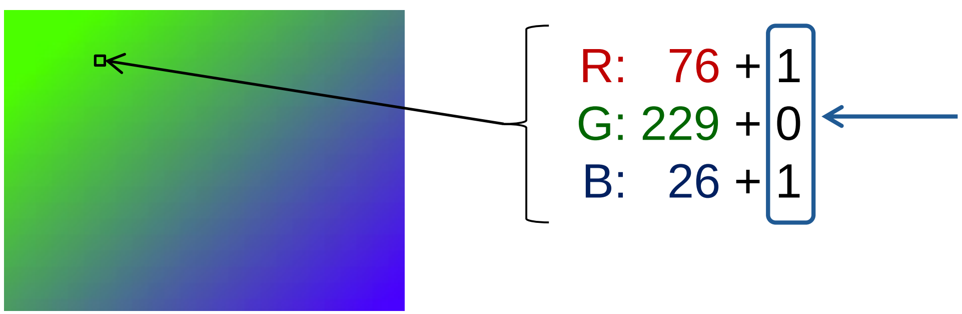
- Mathematical analysis
  - Exploit properties of cryptographic algorithms, solve mathematical problems, find shortcuts
- There is no proof of security for most ciphers
  - How do we know they are secure enough?
  - We presume security if there is no feasible attack known yet
  - Once secure, now broken: DES, MD5, RC4, SHA-1

Kerckhoffs‘ Principle

- Idea: improve security by keeping the system design and its algorithms secret
  - This is called security by obscurity
  - Experience shows: systems get reverse-engineered
  - Systems relying on obscurity will be broken
- Kerckhoffs‘ principle:
  - **System should be secure even if the attacker knows all details except for the key** —Auguste Kerckhoffs
- Shannon‘s maxim:
  - **The enemy knows the system** —Claude Shannon

What Cryptography is not: Steganography

- Steganography („Steganographie“)
  - Study of techniques to hide a message in an unsuspicious carrier medium (image, audio, video, …)
  - Use cases: hidden messaging, digital watermarking
- Example: embed extra bits in image bitmap



  - Three extra bits
  - Minor alteration, not visible with bare eye
- Can be combined with cryptography: first encrypt, then embed

Conclusions

- Security analysis always refers to a threat or security goal
- Confidence in cryptographic algorithms relies on amount of cryptanalysis by experts
  - Be skeptical against new or uncommon ciphers
- Encryption algorithms rely on secrecy of key
- Size of key space is a major security factor
- Weakest link of a security chain breaks
  - Algorithm may be secure, but implementation not