IT Sicherheit

Universitatea “Transilvania” din Brasov

3. Grundlagen der Kryptographie

Agenda: Sicherheit in der Informationstechnologie

3 Grundlagen der Kryptographie

3.1 Verschlüsselung und Vertraulichkeit

- Mono- und polyalphabetische Substitution
- On-time Pad
- Steganographie

3.2 Kryptosysteme und Vertraulichkeit

- Symmetrische und Asymmetrische Verfahren
- Hybridverfahren
- Block- und Stromchiffren

3.3 Hashwert und Datenintegrität

- Hashfunktionen
- Datenintegrität

3.4 Digitale Signatur und Authentifizierung

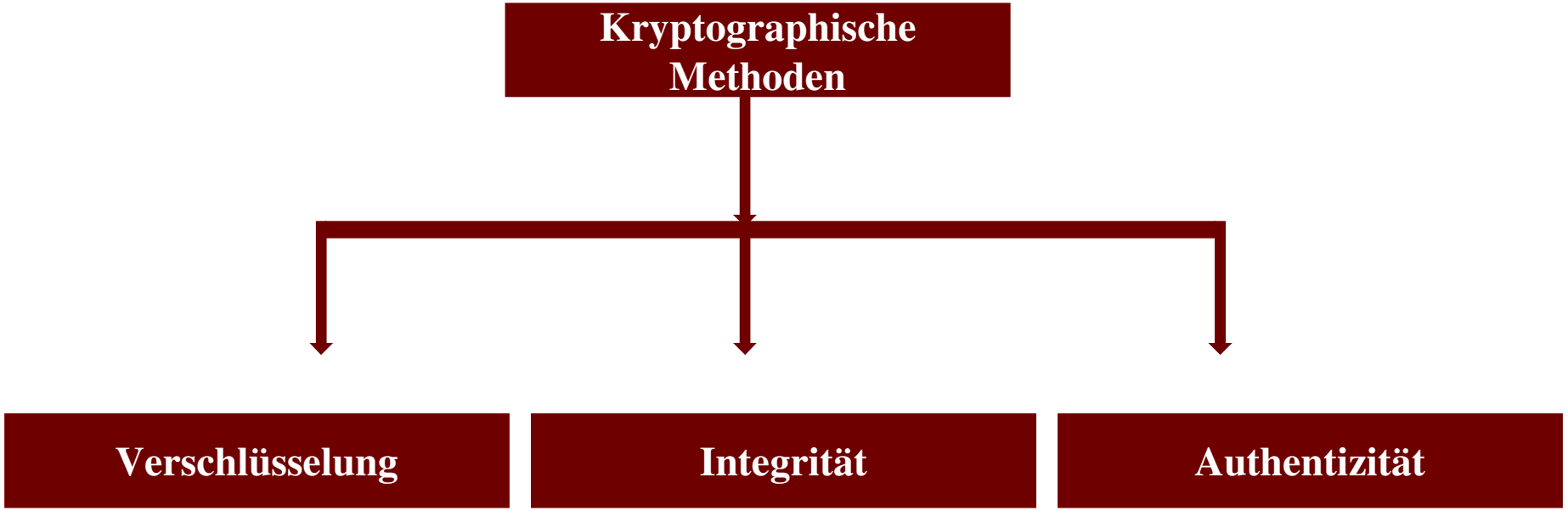
- Digitale Signatur
- Digitaler Signatur Standard

3.5 Kryptosysteme auf Elliptischen Kurven

Bedrohungsart — Sicherheitskriterium

Bedrohungsart	Sicherheitskriterium
❑Sniffing, Trojanische Pferde, Social Engineering	Vertraulichkeit
❑Attack Applets, Session Hijacking, Viren, Trojanische Pferde	Integrität
❑Mail-Spoofing, IP-, ARP-, DNS-Spoofing, Verletzung von Sicherheitsrichtlinien	Authentizität
❑Nichtauthentische Geschäftsabwicklung Ablehnung von Bestellungen	Verbindlichkeit
❑Paßwort Cracking, Session-Hijacking, ICMP-Tunneling	Zugriffskontrolle

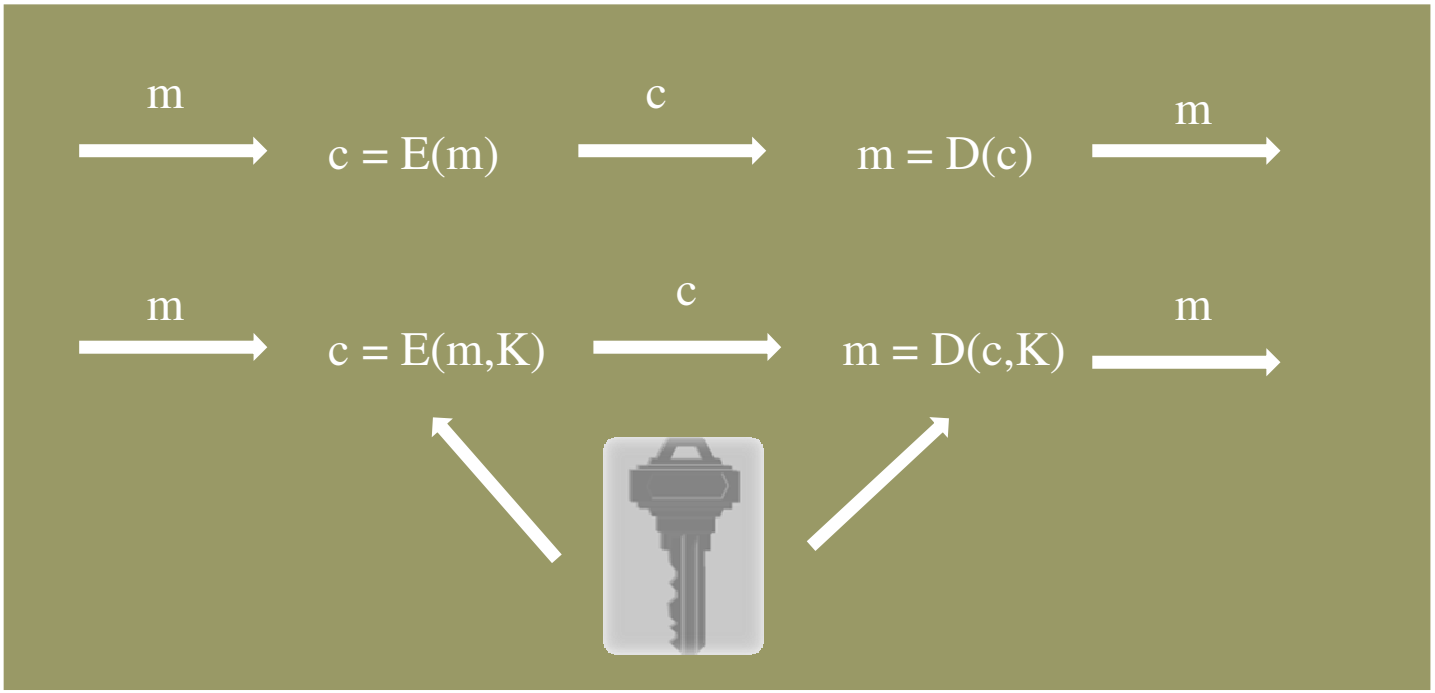
Grundlagen der Kryptographie



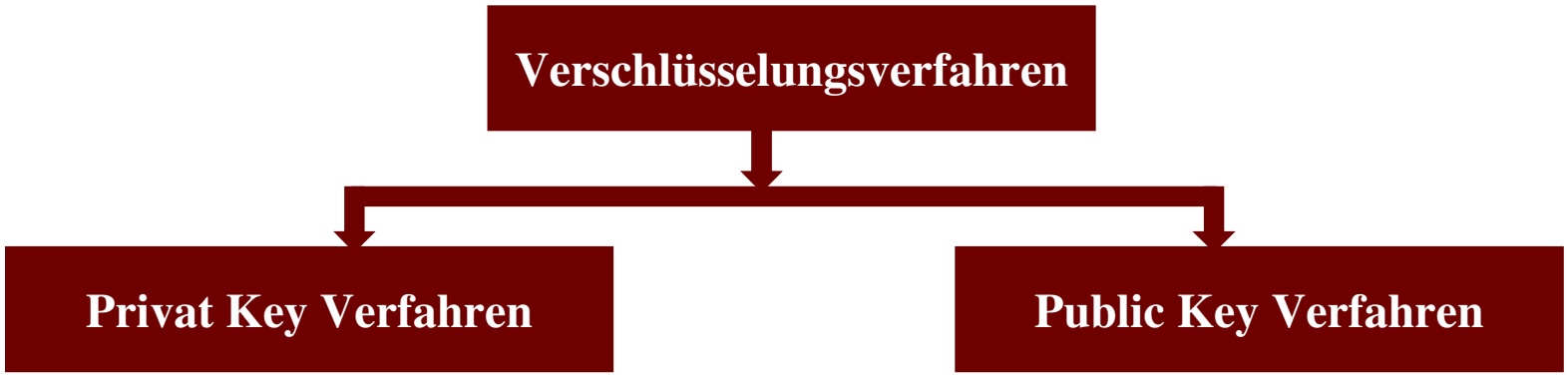
Kryptographische Methoden bilden die theoretische Grundlage für Sicherheitsanwendungen in Rechnernetzen !

Verschlüsselung

Prinzip der Verschlüsselung



Verschlüsselung



Geheimer Schlüssel
einfache Rechenoperationen
Verschlüsselung großer Datenmengen

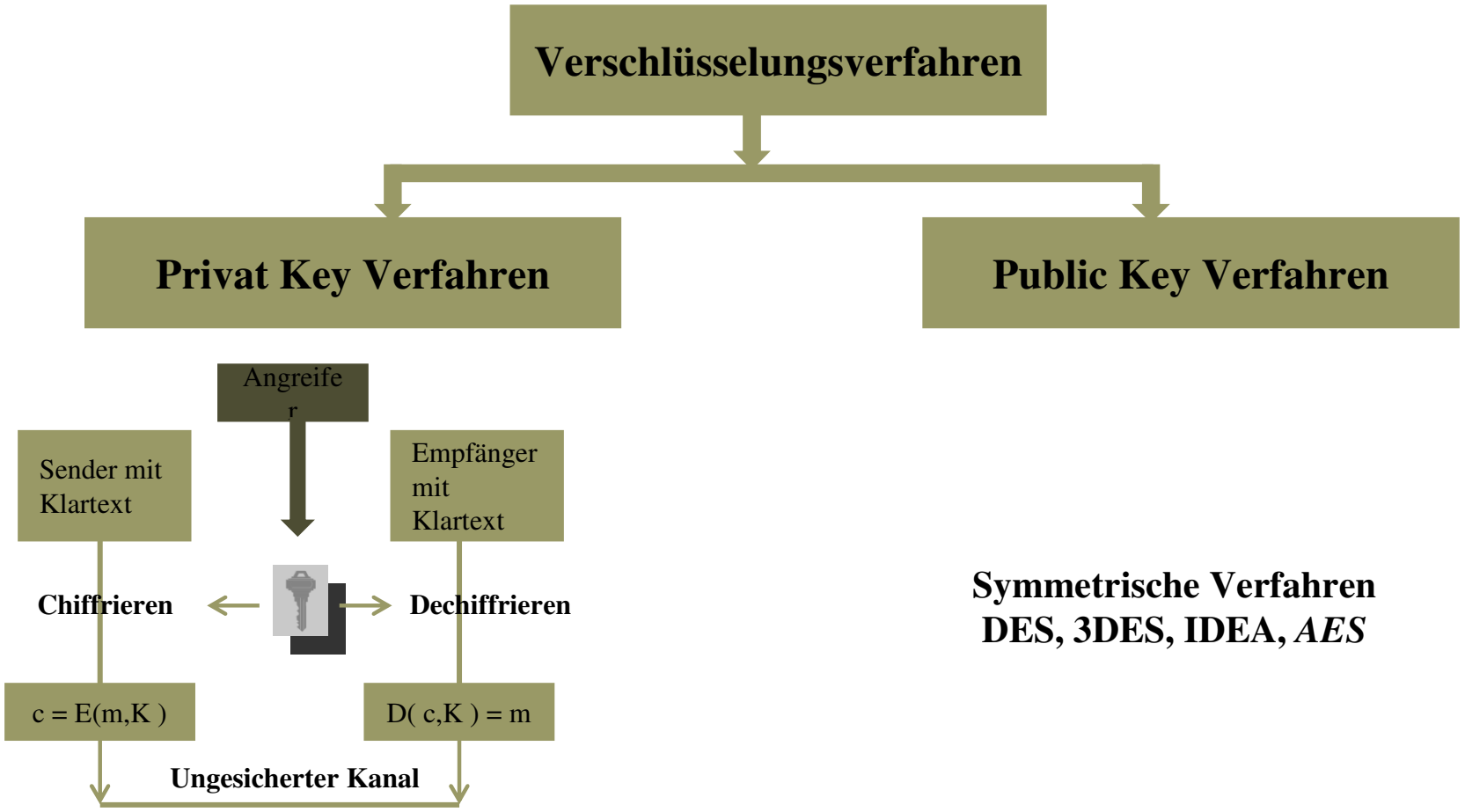
Geheimer und öffentlicher Schlüssel
aufwendige Rechenoperationen
Verschlüsselung geringer Datenmengen

(a) Public Key Verfahren für Austausch eines geheimen Schlüssels

Hybride Verfahren

(b)Datenverschlüsselung mit Privat Key Verfahren

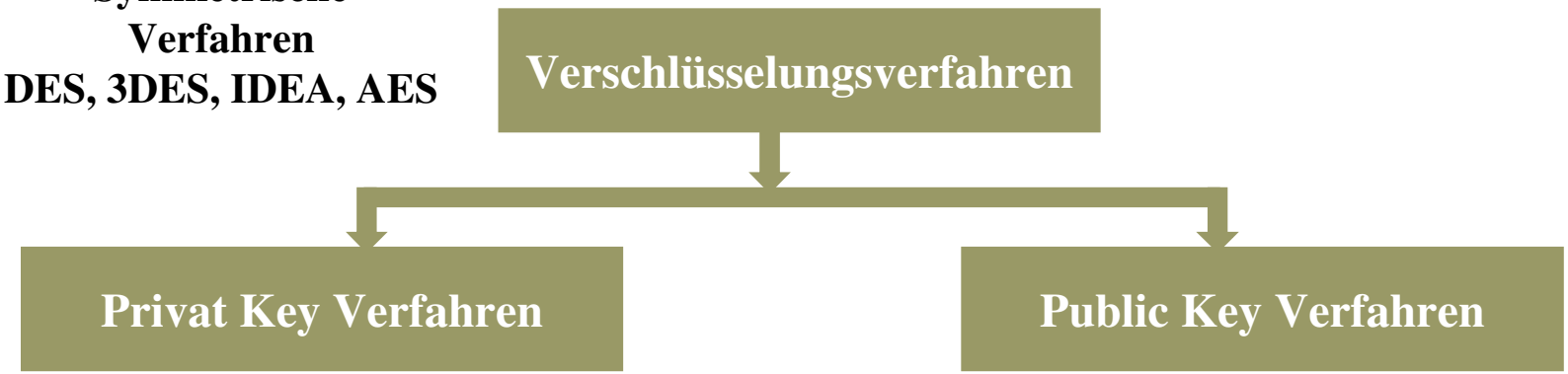
Verschlüsselung



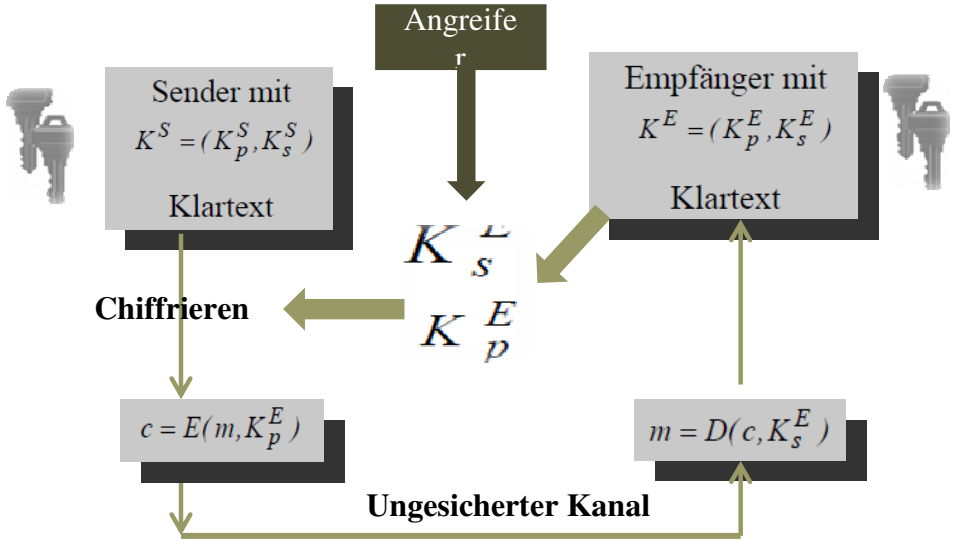
Symmetrische Verfahren
DES, 3DES, IDEA, *AES*

Verschlüsselung

Symmetrische Verfahren
DES, 3DES, IDEA, AES



Asymmetrische Verfahren
RSA, DH, ElGamal,
EC-DH, EC-ElGamal



Verschlüsselung

Beispiel RSA Verfahren

- **Entwicklung**

  Ron Rivest, Adi Shamir, Leonard Adleman, 1978, basiert auf Faktorisierungsproblem

- **Initialisierung**

  p,q prim n = p* q φ (n) = (p – 1)* (q – 1) d ∈Zφ (n) = {0,1,...,φ (n) – 1},d ≠ 0

  ggT(d ,φ (n)) = 1 e* d =1 mod φ (n)

  $K^E = (K_p^E, K_s^E)$ $K_p^E \equiv (n,e)$ $K_s^E \equiv (d)$

- **Verschlüsselung**

  $c = E(m,e) = m^e \bmod n$

- **Entschlüsselung**

  $m = D(c,d) = c^d \bmod n$ denn : $c^d \bmod n = (m^e)^d \bmod n = m^{e \cdot d} \bmod n = m$

- **Entschlüsselung**

  Ist die Faktorisierung von n in p und q bekannt, so kann (e,d) bestimmt werden !
  Daher: $n > 10^{160}$ p und q unterschiedliche Länge
  weitere Bedingungen an p und q: (p-1)/2, (q-1)/2 Primzahlen,
  Schlüssellänge mindestens 1024 Bit

Verschlüsselung

Rechenbeispiel RSA Verfahren

❑ **(1)**

p = 47 q = 59 n = 2773 φ (n) = 46 * 58 = 2668

d = 157 0 < e < 2668 e* 157 = 1 mod 2668 e = 17

Schlüssel $K_p^E = (2773,17)$ $K_s^E = (157)$

Verschlüsselung $c = E(m,17) = m^{17} \bmod 2773$

Entschlüsselung $m = D(c,157) = c^{157} \bmod 2773$

❑ **(2)**

p = 3 q = 17 n = 51 φ (n) = 2* 16 = 32

d = 13 0 < e < 32 e * 13 = 1 mod 32 e = 5 Sei m = 19

Schlüssel $K_p^E = (51,5)$ $K_s^E = (13)$

Verschlüsselung $c = E(19,5) = 19^5 \bmod 51 = (19^2 19^2 19) \bmod 51 = (4 \cdot 4 \cdot 19) \bmod 51$

$(4 \cdot 76) \bmod 51 = (4 \cdot 25) \bmod 51 = 49 \rightarrow c = 49.$

Entschlüsselung $m = D(49,13) = 49^{13} \bmod 51 = (-2)^{13} \bmod 51 = (1024 \cdot (-8)) \bmod 51$

$= (4 \cdot (-8)) \bmod 51 = (-32) \bmod 51 = 19 \bmod 51 = 19$

Integrität

Prinzip der kryptographischen Hashfunktion

Nachricht m = (10011010...1000110)

mit beliebiger Länge

Hashwert h = (01...101)

mit fester Länge (128, 160 Bit)

Hashfunktion m → h = H (m)

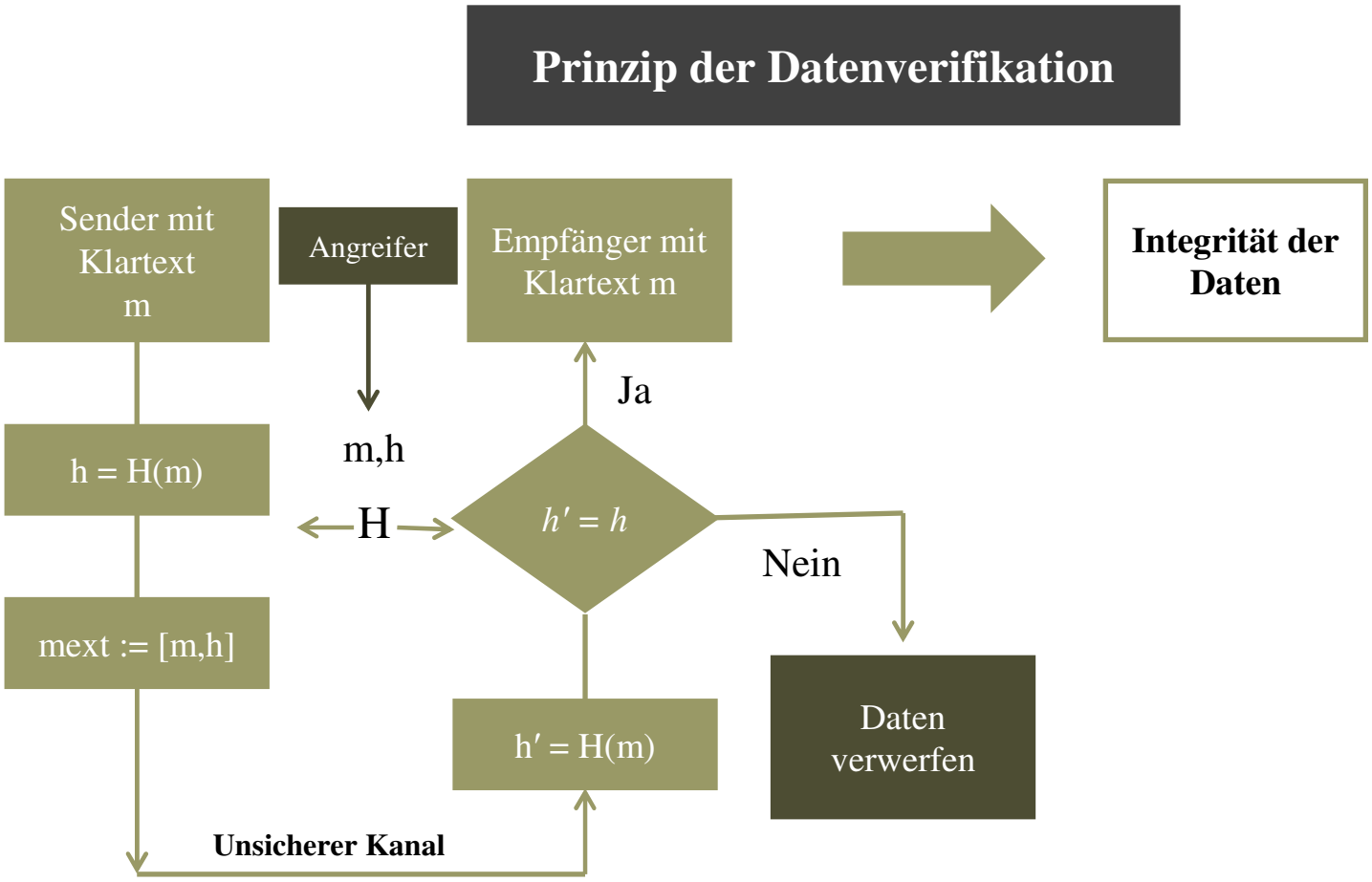
- Einweg-Funktion

  H(m) ⇒ m unmöglich

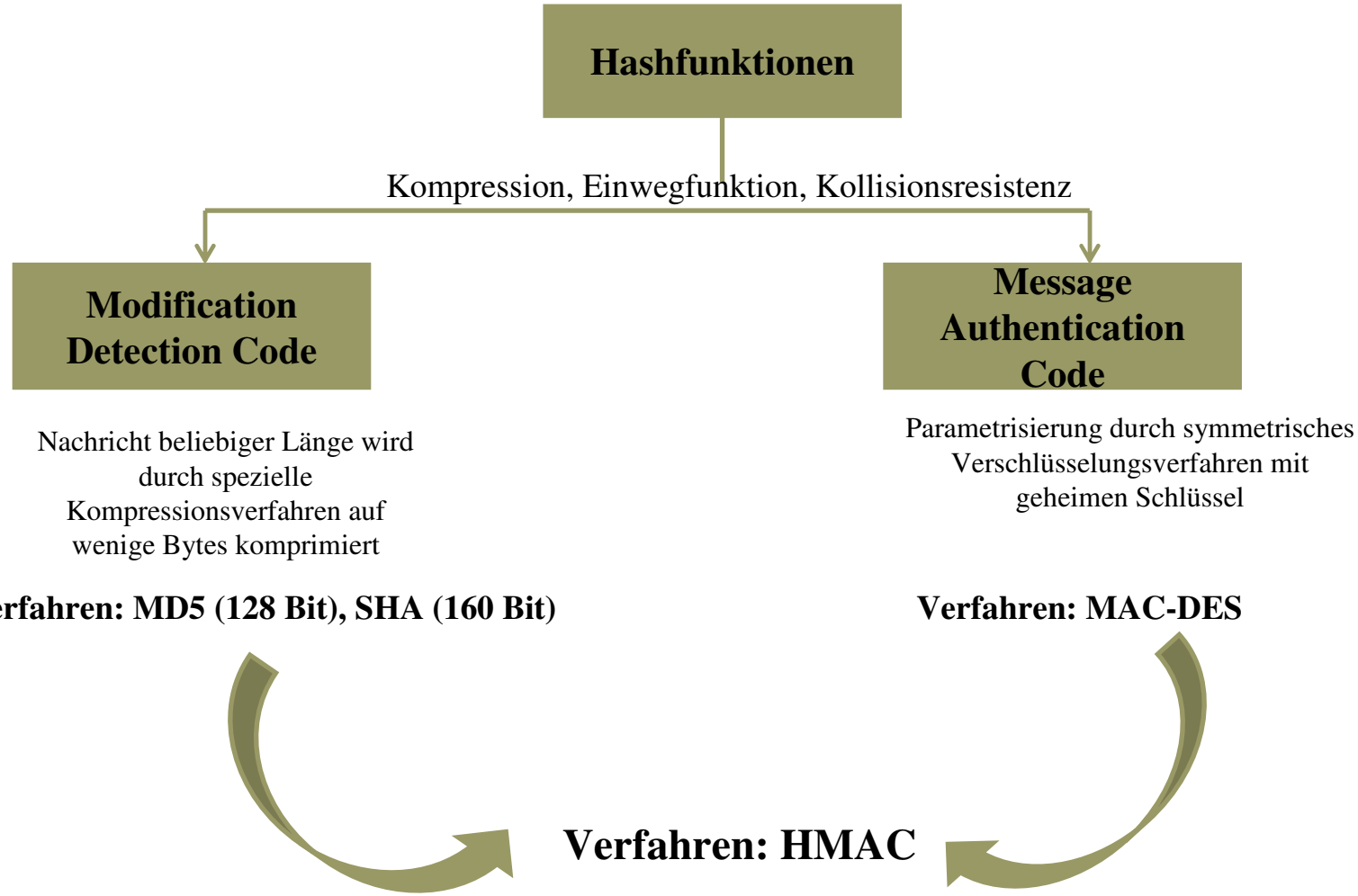
- Kollisionsresistent

  $m \neq m' \Rightarrow H(m) \neq H(m')$

Integrität


Prinzip der Datenverifikation
Sender mit Klartext m
Angreifer
Empfänger mit Klartext m
Integrität der Daten
Ja
m,h
h = H(m)
H
h′ = h
Nein
mext := [m,h]
h′ = H(m)
Daten verwerfen
Unsicherer Kanal

Integrität


Hashfunktionen
Kompression, Einwegfunktion, Kollisionsresistenz
Modification Detection Code
Message Authentication Code
Nachricht beliebiger Länge wird durch spezielle Kompressionsverfahren auf wenige Bytes komprimiert
Parametrisierung durch symmetrisches Verschlüsselungsverfahren mit geheimen Schlüssel
Verfahren: MD5 (128 Bit), SHA (160 Bit)
Verfahren: MAC-DES
Verfahren: HMAC

Integrität

Keyed Hashfunktion
HMAC

❑ Der Algorithmus HMAC wurde im Jahre 1996 entwickelt und ist der verbreitetste MDCbasierende MAC-Algorithmus. Von einer Nachricht m beliebiger Länge und einem geheimen Schlüssel K werden ein Hashwert h fester Länge wie folgt gebildet:

❑ **Hashverfahren** H(m) (meist MD5 oder SHA-1) **Geheimer Schlüssel K**

Nachricht m in Blöcke von 64 Byte zerlegen (ggf. Padding)

Geheimer Schlüssel K wird bis zur Blocklänge mit Nullen aufgefüllt (Padding): K+.

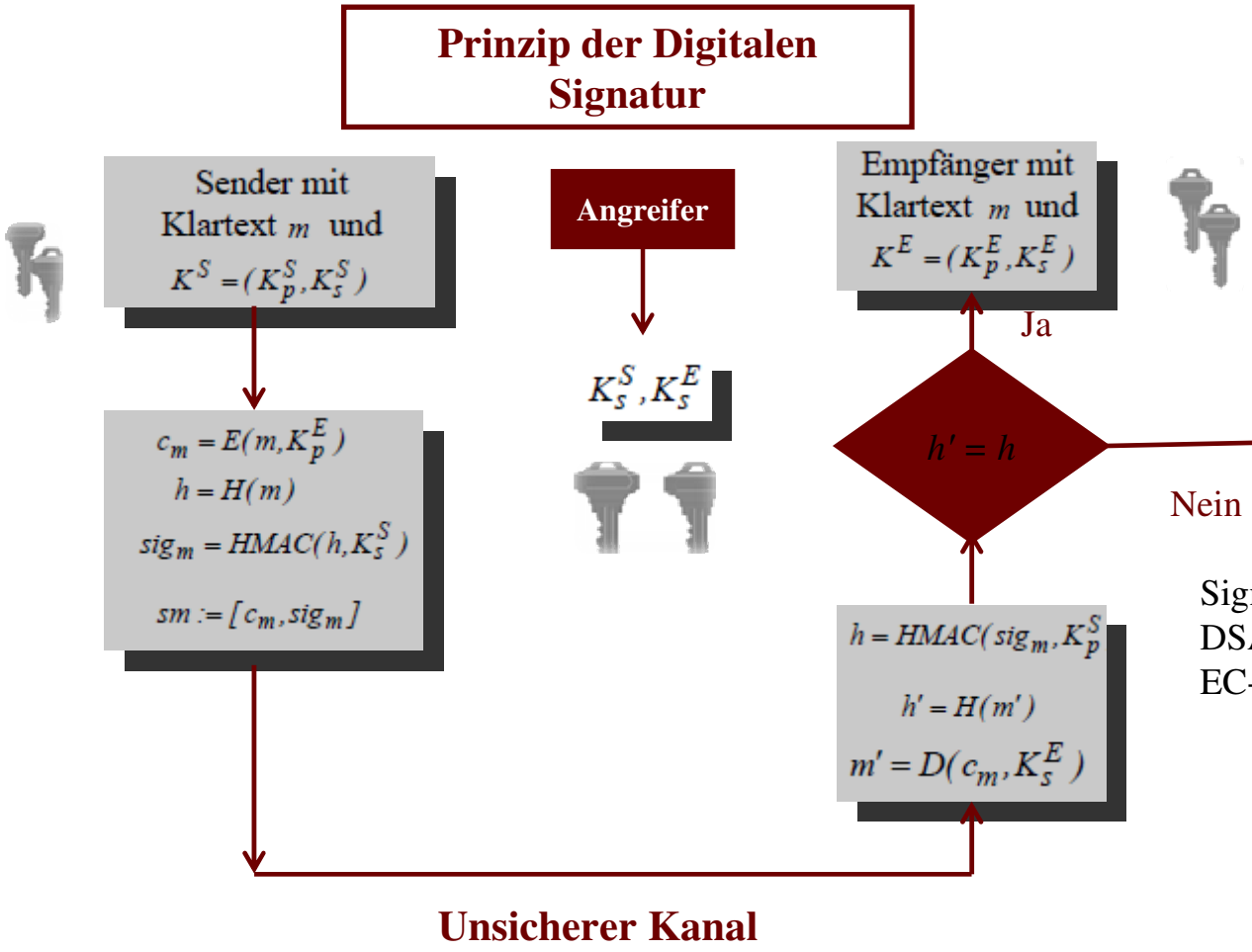
❑ **Hilfsschlüssel**

$$S_i := K^+ \otimes ipad \qquad ipad = (00110110)_{64mal} = (36)_{64mal}$$
$$S_o := K^+ \otimes opad \qquad opad = (01011100)_{64mal} = (5C)_{64mal}$$

Hashwert

$$h := HMAC(m,K) := H(S_o \,|\, H^+(S_i \,|\, m)) \equiv H(K^+ \otimes opad, H^+(K^+ \otimes ipad, m))$$

Authentizität



Authentizität des (unbekannten) Senders und Integrität der Daten

Signaturverfahren
DSA, RSA, RIPEMD, ElGamal, EC-DSA, EC-ElGamal

Authentizität

Grundstruktur: Digitale Signaturverfahren

- **Parameter** — **Setup Algebraische Gruppe, u.a. Ordnung, Generator**
  Schlüsselpaar der Kommunikationsteilnehmer
  Hash-Verfahren
- **Signaturerzeugung** — **Sender der Nachricht erzeugt**
  (a) den Hashwert und
  (b) dann die Signaturparameter
- **Signaturprüfung** — **Empfänger der Nachricht berechnet**
  (a) Hilfswerte und
  (b) dann den Vergleichswert für Signaturparameter
  (c) Vergleichstest

Authentizität

Beispiel Digital Signature Algorithm (DSA)

- **Entwicklung:** von NSA entwickelt, basiert auf diskreten Logarithmusproblem, seit 1994 Standard durch National Institute of Standards der USA
- **Setup:**

$$G = Z_p,\ p\ prime \quad q\ prim\ mit\ q \mid (p-1) \quad und\ g \in G\ mit\ ord g = q$$

$$Schlüsselpaar\ K^{user} := (k_p^u, k_s^u)\ mit\ k_s^u := l \in Z_q^*,\ l \neq 1\ und\ k_p^u := g^l \in Z_p^*$$

$$Hashalgorithmus\ SHA-1 \qquad K_p \equiv (p,q,g,k_p^u)\ \ und \qquad K_s \equiv (k_s^u)$$

- **Erzeugung:**

$$(1)\ \ random\ \ k \in Z_q^*, k \neq 1,\ und\ \ Wert\ \ r := g^k \bmod p$$

$$(2)\ \ berechne\ \ r' := r \bmod q\ \ und\ \ s := (k^{-1}(k_s^{Send} \cdot r' + H(m)) \bmod q$$

$$\Rightarrow\ Signatur\ \ von\ \ m\ \ ist\ \ sig := (r', s)$$

- **Verifikation:**

$$(1)\ \ 1 \leq r', s \leq q-1$$

$$(2)\ \ (a)\ berechne\ Hilfswerte\ \ u := s^{-1} \bmod q,\ \ v_1 := (u \cdot H(m)) \bmod q,\ \ v_2 := (u \cdot r') \bmod q$$

$$(b)\ berechne\ Vergleichswert\ \ w := ((g^{v_1}(k_p^{Send})^{v_2}) \bmod p) \bmod q$$

$$\Rightarrow\ akzeptiere,\ \ falls\ \ r' = w$$

Trust Center

Öffentliche Schlüssel müssen authentisch sein !!!

Hierarchical Trust
PKIX-Standard

Trust Center, Certification Service Provider
Dritte unabhängige vertrauenswürdige Instanz (Certification Authority)

Zertifizierungsdienst

Registrierungsdienst

Verzeichnisdienst

Zeitstempeldienst

- **Schlüssel erzeugen (auch geheime Signierschlüssel, aber ohne Key Recovery)**
- **Schlüssel mit Zusatzinformationen beglaubigen (CA-Signatur)**
- **Schlüssel abspeichern und verteilen (geschützte Datenbank, sichere Verteilungswege)**
- **Schlüssel sperren bzw. neu ausstellen (Zertifikate-Management)**

Authentizität

Formen der elektronischen Signatur nach Signaturgesetz

Beispiel

1. (Einfache) Elektronische Signatur (ES)
Elektronische Daten, die mit anderen elektronischen Daten verknüpft sind und zu ihrer Authentifizierung dienen

❑ Gescannte Unterschrift, PIN, TAN, Kennziffer und elektronische Empfangsbestätigung (ohne Regularien)

2. Fortgeschrittene Elektronische Signatur (FES)
Einfache Elektronische Signatur zuzüglich
❑ Signaturschlüssel, der dem Inhaber zweifelsfrei zuordenbar ist
❑ Identifizierung des Schlüsselinhabers
❑mit Mitteln erzeugt, die der Schlüsselinhaber kontrollieren kann
❑ Erkennung nachträglicher Veränderungen der Daten

❑ Digitale Signatur, PGP asymmetrische Krypto-Verfahren geschl. Benutzergruppen (ohne Smart Card)

3. Qualifizierte Elektronische Signatur (QES)
Fortgeschrittene Elektronische Signatur zuzüglich
❑ Echtheit des qualifizierten Zertifikats zum Zeitpunkt der Erzeugung
❑ Erstellung auf einer sicheren Signaturerstellungseinheit

❑Digitale Signatur mit Signaturgesetzkonformer PKI und Smart Card

Zusammenfassung

- S
- E
- C
- U
- R
- I
- T
- Y