

# Computersystemsicherheit



TECHNISCHE  
UNIVERSITÄT  
DARMSTADT



0011011100010111 **Cryptoplexity**

Cryptography & Complexity Theory  
Technische Universität Darmstadt  
[www.cryptoplexity.de](http://www.cryptoplexity.de)

Prof. Marc Fischlin, Wintersemester 18/19

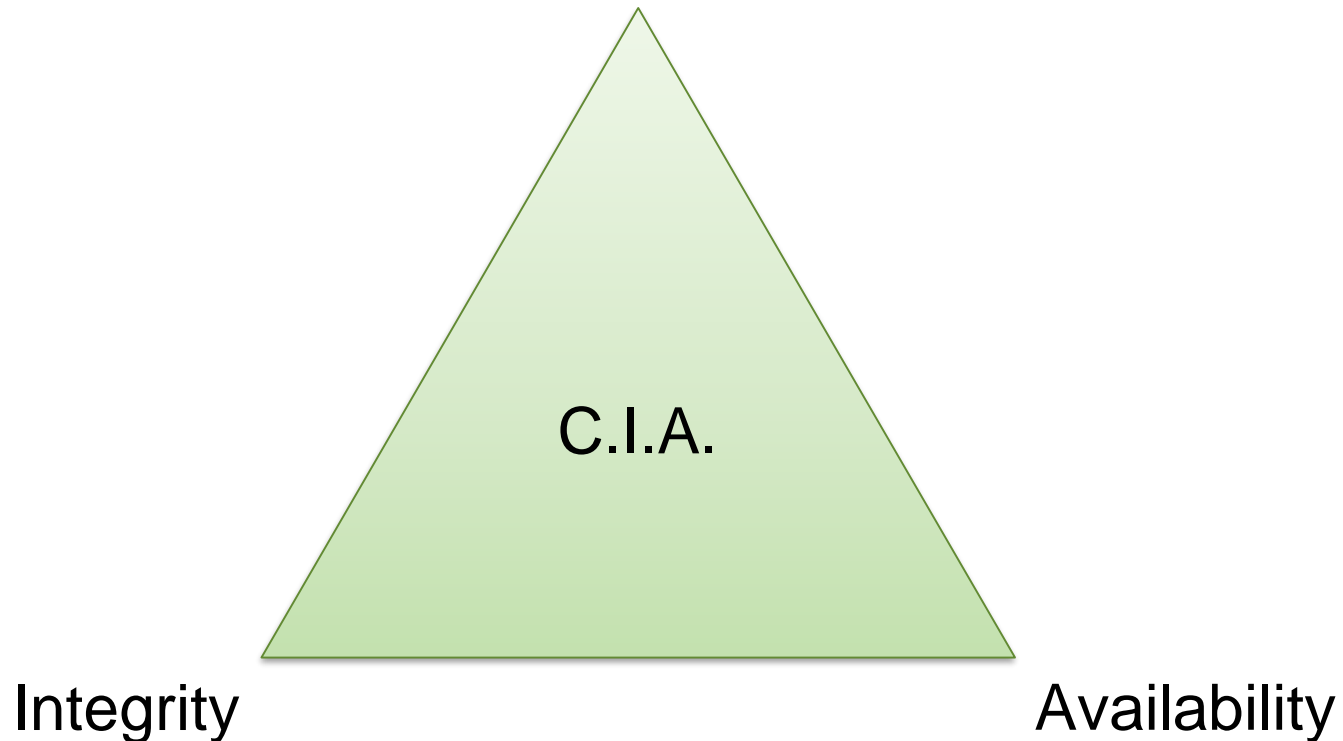
---

02

Verschlüsselung

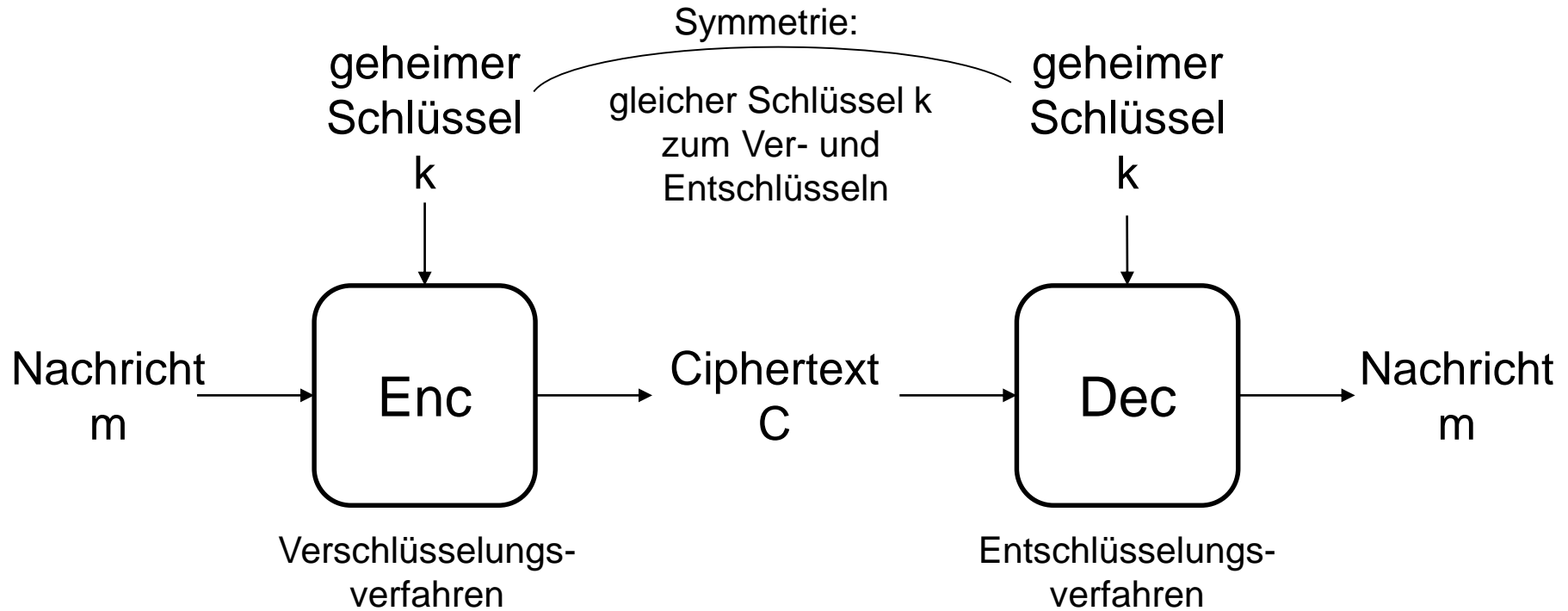
---

Verschlüsselung  $\xrightarrow{\text{sorgt für}}$  Confidentiality



Achtung:  
Verschlüsselung  
sorgt im Allgemeinen nicht für  
Integrität oder Verfügbarkeit

# Prinzip der (symmetrischen) Verschlüsselung

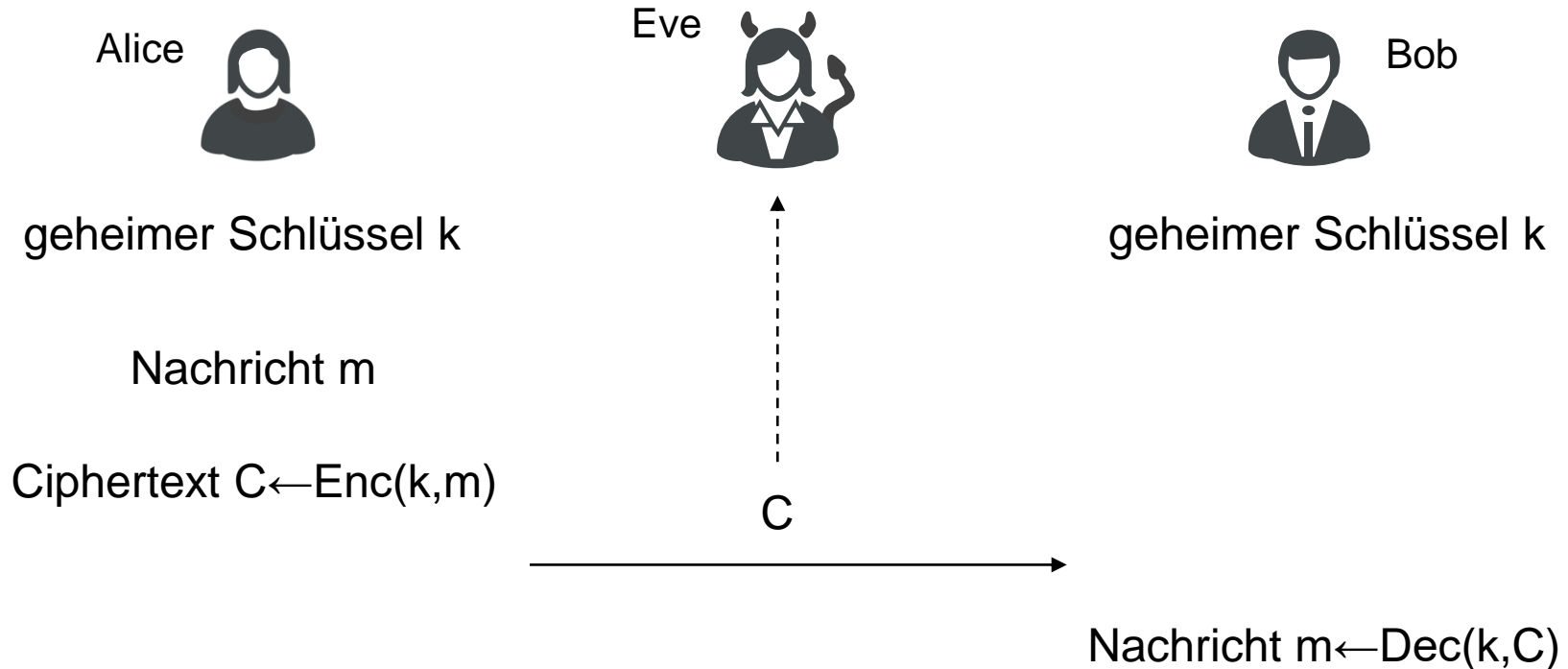


Funktionale Korrektheit (Vollständigkeit):

Für alle Nachrichten  $m$  und alle Schlüssel  $k$  gilt:  $\text{Dec}(k, \text{Enc}(k, m)) = m$

# Sicherheit?

hört Kommunikation ab



Ciphertext darf keine sinnvollen  
Informationen über Nachricht preisgeben

→ „Einführung in  
die Kryptographie“

# Kerckhoffs-Prinzip



Auguste Kerckhoffs (1835–1903)  
Quelle: Wikipedia

Die Sicherheit eines kryptographischen Systems beruht **nicht** auf der Geheimhaltung des Systems, sondern nur auf der des Schlüssels.

Angreifer kennt das Verfahren, aber nicht den konkreten Schlüssel

Kein „Security by Obscurity“

---

# Klassische symmetrische Verschlüsselungssysteme

# Skytale



Quelle: Wikipedia

ältestes bekanntes Verschlüsselungsverfahren

Spartaner, ca. 2500 vor Christus

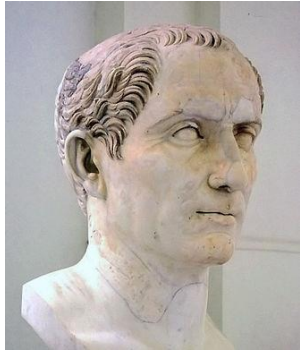
Geheimnis: Durchmesser der Skytale

D	I	E	S	I	S	T	D	E	R	K	L	A	R	T	E	X	T
↓																	...
D	T	A	I	D	R	E	E	T	S	R	E	I	K	X	S	L	T

## Transpositionsverfahren

Buchstabe wird „nur“ verschoben

# Caesars Shift-Cipher (I)



Caesar, 100BC-44BC  
Quelle: Wikipedia

D	I	E	S	I	S	T	D	E	R	K	L	A	R	T	E	X	T
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
G	L	H	V	L	V	W	G	H	U	N	O	D	U	W	H	A	W

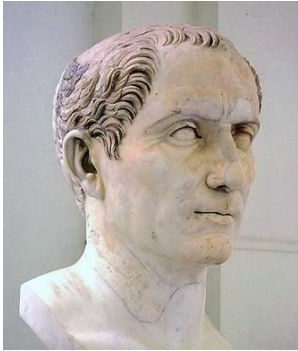
A → D  
B → E  
C → F  
...  
W → Z  
X → A  
Y → B  
Z → C

Verschlüsselungsvorschrift

geheimer Schlüssel = „3 Buchstaben weiterdrehen“  
(oder 4 Buchstaben oder 5 Buchstaben...)



# Caesars Shift-Cipher (II)



Caesar, 100BC-44BC  
Quelle: Wikipedia

G	L	H	V	L	V	W	G	H	U	N	O	D	U	W	H	A	W
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
D	I	E	S	I	S	T	D	E	R	K	L	A	R	T	E	X	T

gleicher verschlüsselter Buchstabe  
=gleicher Buchstabe im Klartext

Verschlüsselungsvorschrift

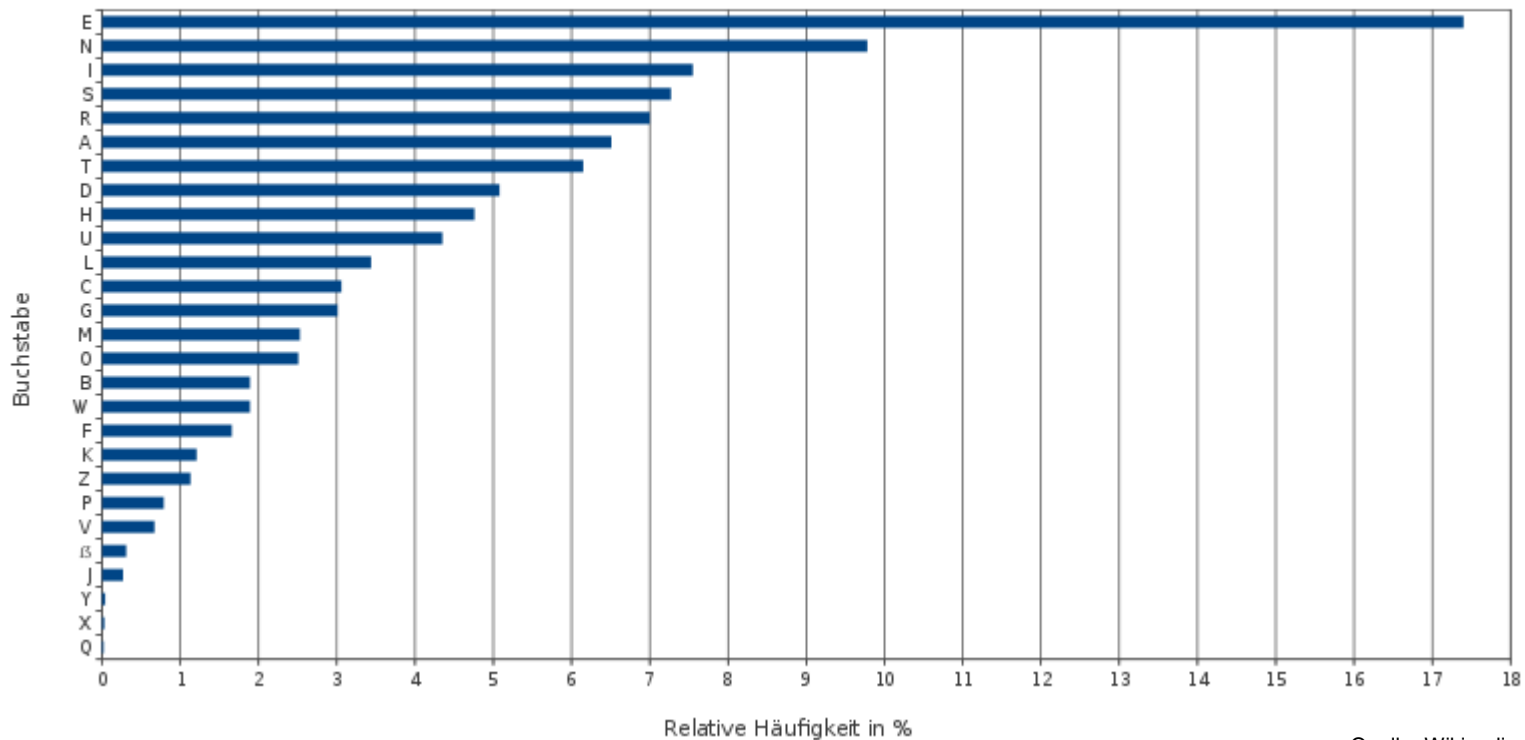
A → D  
B → E  
C → F  
...  
W → Z  
X → A  
Y → B  
Z → C

**mono-alphabetische Substitution**

geheimer Schlüssel = „3 Buchstaben weiterdrehen“  
(oder 4 Buchstaben oder 5 Buchstaben...)

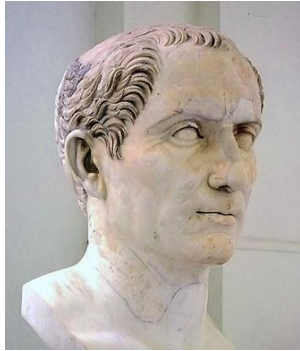
# Caesar-Chiffre: Angriff (I)

Buchstabenhäufigkeiten in deutschsprachigen Texten




Quelle: Wikipedia

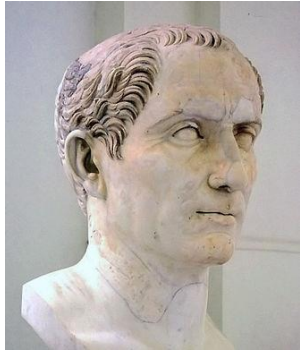
# Caesar-Chiffre: Angriff (II)



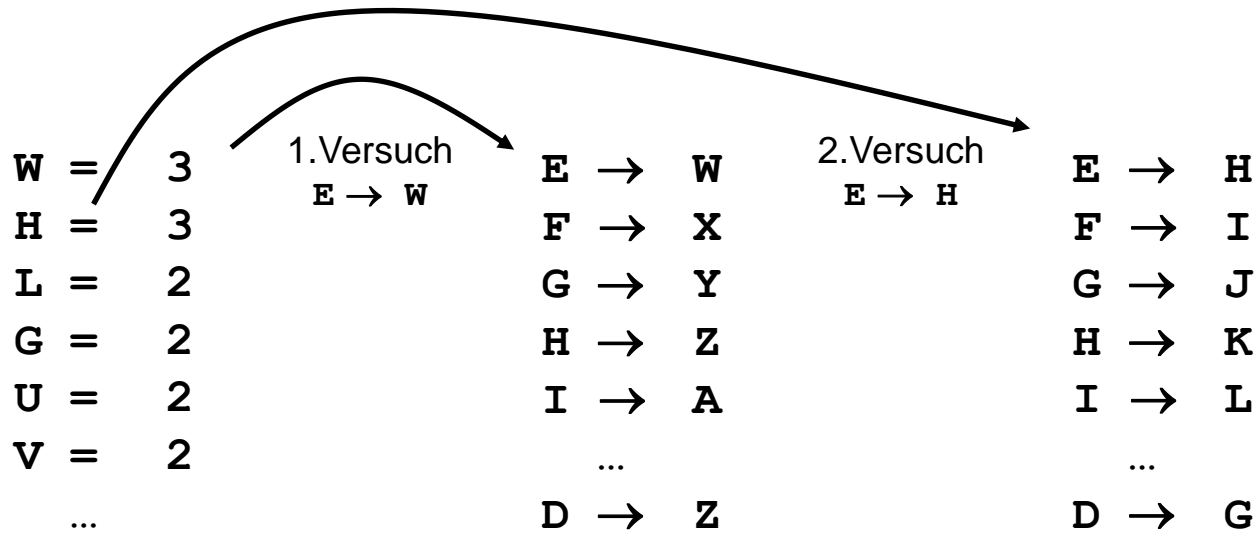
G	L	H	V	L	V	W	G	H	U	N	O	D	U	W	H	A	W
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
O	T	P	D	T	D	E	O	P	C	V	W	L	C	E	P	I	E

W = 3	 1. Versuch E → W	E → W
H = 3		F → X
L = 2		G → Y
G = 2		H → Z
U = 2		I → A
V = 2		...
...		D → V

# Caesar-Chiffre: Angriff (III)



G	L	H	V	L	V	W	G	H	U	N	O	D	U	W	H	A	W
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
D	I	E	S	I	S	T	D	E	R	K	L	A	R	T	E	X	T



# Verschlüsselung heute gestern



w.spiegel.de/wirtschaft/soziales/trotz-brexit-will-cornwall-subventionen-haben-a



Suchen



SPIEGEL ONLINE  
Regierungschef Pollard

Durchschnitt.

Schon kurz nach der Abstimmung erklärte Pollard dann in Richtung britischer Regierung, er werde "darauf bestehen, dass Cornwall Investitionen im gleichen Wert zu den EU-Programmen erhält". Die Reaktion vieler Kommentatoren: Das hättet ihr euch vorher überlegen müssen.

**Bereuen die Menschen in Cornwall ihre Entscheidung? Und welche Chancen haben sie auf weitere Hilfen? Eine Rückkehr zu Befürwortern und Gegnern des Brexit.**


**Jetzt lesen, später zahlen**

 **Diesen Artikel sofort weiterlesen für  
0,39 EUR**

Bereits gekauft?

- Sie bezahlen erst, wenn Sie eine Summe von fünf Euro erreichen
- Vorher ist keine Registrierung nötig
- Sie schließen damit kein Abo ab

So funktioniert es: LaterPay

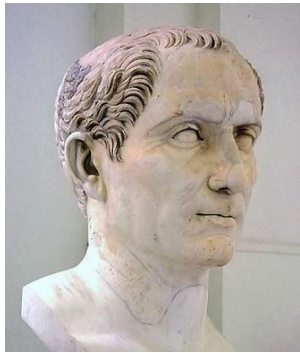
Powered by  LATERPAY

2016

# Als HTML-Code:



```
1147 <p>
1148 <b>Bereuen die Menschen in Cornwall ihre Entscheidung? Und welche Chancen haben sie auf w
1149 </p><div id="laterpay-replacement"></div><div class="laterpay-under-overlay"></div><div c
1150
1151 SPIEGEL ONLINE</div>
1153 <p>Wirtschaftsvertreterin Rothwell</p>
1154 </div>
1155 </div>
1156 <p class="obfuscated">Ibu Dpsoxbmm nju tfjofn Wpuvn ojdiu ebt Bosfdiu bvg
1157 <div class="asset-title">
1158 <a href="/fotostrecke/fotostrecke-139273.html" title="Fotostrecke zei
1159
1160 <!-- Markup Slider -->
1161 <div class="specialwidth860">
1162 <div id="happ-slider-html-139273" style="display:none !important">
1163 <div id="content-slider-139273">
1164
1165 <div class="rsContent" data-count-picturesid="1021882" data-count-par
1166 <div class="content">
1167 <div class="fitwidth">
1168 <div class="imgSliderTab"
1169 data-src="http://cdn3.spiegel.de
1170 data-credit="SPIEGEL ONLINE"
```



I	b	u	D	p	s	o	x	b	m	m	n	j	u	t	f	j
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
H	a	t	C	o	r	n	w	a	l	l	m	i	t	s	e	i

Ceasar CIPHER mit  
Verschiebung um eine Position!

# Verschlüsselung heute

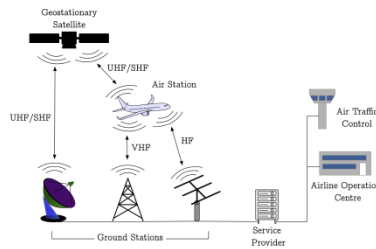


Modern Jets, Retro Ciphers:

How Monoalphabetic Substitution Ciphers are still in use,  
M.Smith, D.Moser, D.Strohmeier, V.Lenders, I.Martinovic,  
Real-World Crypto 2018

## WHAT IS ACARS?

- **Aircraft Communications Addressing and Reporting System** (ACARS) is a widely-used avionic data link on both commercial and non-commercial aircraft
- Around since late 1970's, it is now used for **vastly different purposes** to its original intention
- Since then, it has become **multi-medium and multi-purpose**
- Easily collectible with **\$10 hardware**



## ACARS:

Aircraft Communications Addressing and Reporting System:

einfaches Kommunikationssystem  
zwischen Ground Control  
und Flugzeug

z.B. Positionsdaten des Flugzeugs





Für einige wenige Flugzeugtypen mit folgender Verschlüsselung:

## ANALYSING MESSAGES

- We collected over a million VHF and SATCOM ACARS messages, and noticed that some business aircraft were sending scrambled messages

Key identifier

```
07 ?X.0)Emk.;M]..;4;Dm)m..) Y(*)]s($).M4U).U;;).MmD)..D+0
07 ?X.0)EmUkm]..D00M)4k.)]rr6) Y-\).k.<);4<k);000)..;+U
07 ?X.0)EmUmUU]..D0Mk)m;.)]E{-) 6-r).k.;);;;;);4;;)..U+.
```

```
08, suL}Zq`cLLK=LLa`aLZ`YLZP\,0ZPf0,ZLaLYZLKeeZLc}KZLLc[`
08, suL}Zq`tee}=LLaL}KZ}vvZ=yy~ZPuAfZLaYYZYevvLZY}eLZLLc[t
08, suL}Zq`KYev=LLK}aKZ}tLZbZbZLaYYZYevvZY`YvZbbbbbb
```

```
09|\L46c+Ns6,,G4418,hcN84cGeodc-r!Lc4Bh1c8B4hc8BBBc44Z5Z
09|\L46c+N,BZ,G44BBZNc614c-r|Gc-W|Pc4BhZc48hNc48BZcbbbbbb
09|\L46c+Ns8NhG44s6,,c6B4c-W|Pc-r.-c4B68c888Bc88NZc44B5,
```

Was könnte das für eine Verschlüsselungsmethode sein?

# Vigenere-Chiffre (I)



Blaise de Vigenère (1523-1596)  
Quelle: Wikipedia

	D	I	E	S	I	S	T	D	E	R	K	L	A	R	T	E	X	T
*	P	A	S	S	W	O	R	T	P	A	S	S	W	O	R	T	P	A
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
	T	J	X	L	F	H	L	X	U	S	D	E	X	G	L	Y	N	U

## Vorschrift:

(1) Wiederhole geheimen Schlüssel, bis gleiche Länge wie Klartext

**PASSWORD → PASSWORDPASSWORDPA**

(2) Drehe jeweils Klartext-Buchstaben um so viele Buchstaben weiter, wie Position des Passwort-Buchstabens im Alphabet

Beispiel: **I \* A → J**, da **A** erster Buchstabe im Alphabet

# Vigenere-Chiffre (II)



Blaise de Vigenère (1523-1596)  
Quelle: Wikipedia

	D	I	E	S	I	S	T	D	E	R	K	L	A	R	T	E	X	T
*	P	A	S	S	W	O	R	T	P	A	S	S	W	O	R	T	P	A
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
	T	J	X	L	F	H	L	X	U	S	D	E	X	G	L	Y	N	U

Buchstabe kann zu verschiedenen  
Buchstaben verschlüsselt werden

**poly-alphabetische Substitution**

Frequenzanalyse funktioniert immer noch, erfordert allerdings mehr Arbeit

---

# Zwischen klassischen und modernen symmetrische Verschlüsselungssystemen

# Shannons One-Time-Pad-Verschlüsselung



Shannon, Claude.  
"Communication Theory of Secrecy Systems".  
Bell System Technical Journal, 1949.

Vigenere-Cipher über Bits  $\{0,1\}$ ,

wobei Bitlänge(Schlüssel) = Bitlänge(Nachricht)

**One-Time-Pad-Encryption:**  
Jedes Schlüsselbit zufällig, nur einmal verwenden

# One-Time-Pad-Verschlüsselung



Claude Shannon (1916-2001)

	0	1	1	0	1	0	1	1	1	1	0	0	0	1	0	1	0	1
$\oplus$	1	0	1	1	0	0	1	0	1	1	0	1	1	0	0	0	1	0
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
	1	1	0	1	1	0	0	1	0	0	0	1	1	1	0	1	1	1

$\text{Enc}(k,m) = k \oplus m$  für bitweises Exklusiv-Oder (XOR):

$\oplus$	0	1
0	0	1
1	1	0

## Sicherheit:

Wenn i-tes Ciphertextbit = 0, dann Möglichkeiten gleich wahrscheinlich:  
i-tes Nachrichtenbit = 0 und i-tes Schlüsselbit = 0 vs.  
i-tes Nachrichtenbit = 1 und i-tes Schlüsselbit = 1

analog für Ciphertextbit = 1

# Verschlüsselung und Integrität

Kontostand = 999 €

C.I.A.

$m =$  ...00111001 00111001 00111001 00100000 10000000

$k =$  ...10001010 01010111 01100101 01001001 10100010

$m \oplus k =$  ...10110011 01101110 01011100 01101001 00100010

$m^* =$  ...00001111 00000001 00001000 00000000 00000000

$m \oplus k \oplus m^* =$  ...10111100 01101111 01010100 01101001 00100010

$k =$  ...10001010 01010111 01100101 01001001 10100010

$m \oplus m^* =$  ...00110110 00111000 00110001 00100000 10000000

Kontostand = 681 €

Verschlüsselung



Entschlüsselung

Angreifer kennt Nachricht  $m$  immer noch nicht, kann sie aber ändern!



Erklären Sie den Unterschied zwischen mono-alphabetischer und poly-alphabetischer Substitution. Was ist das One-Time-Pad-Verfahren für ein Typ?



Entwerfen Sie ein absolut sicheres Verschlüsselungssystem, das aber keine Korrektheit garantieren muss.



C.Lever möchte Shannons Verfahren noch sicherer machen. Er nimmt dazu zwei Schlüssel  $k_0$  und  $k_1$  und bildet

$$\text{Enc}((k_0, k_1), m) = k_0 \oplus k_1 \oplus m.$$

Was halten Sie davon?



---

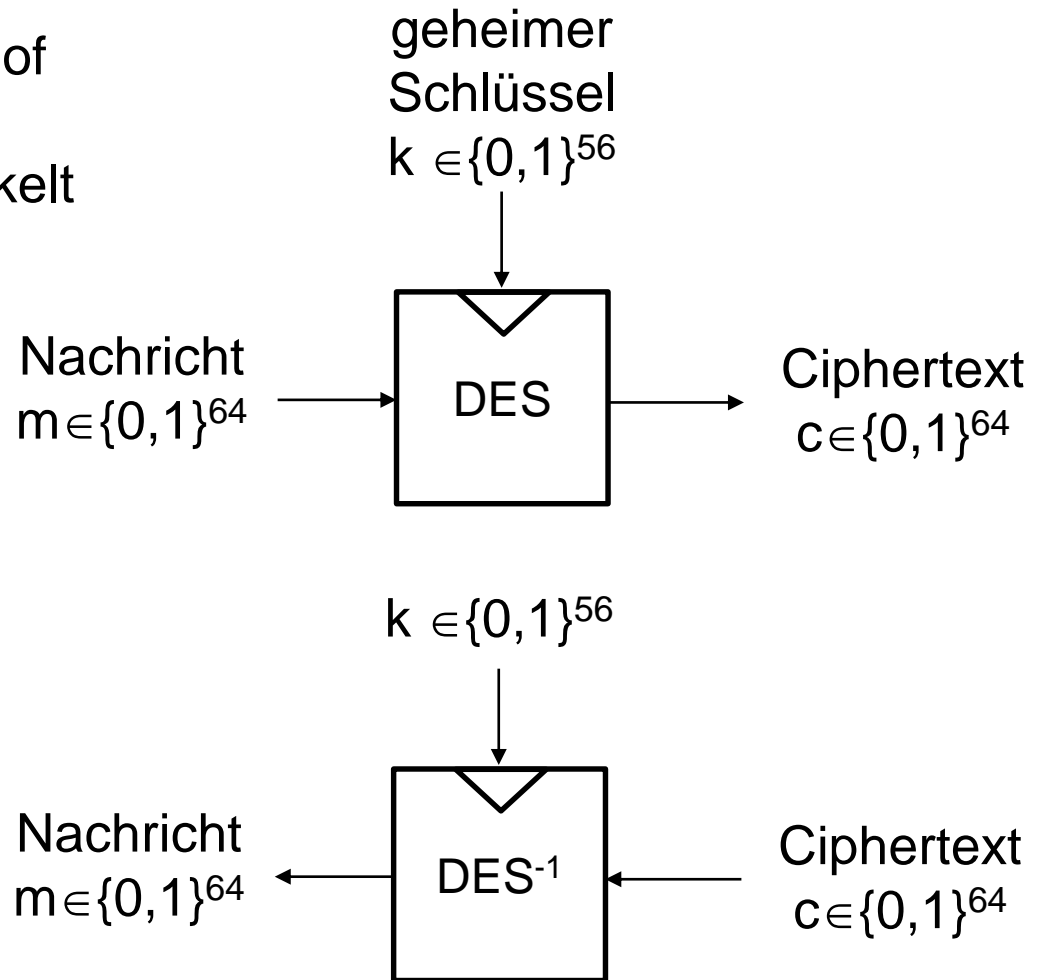
# Moderne symmetrische Verschlüsselungssysteme

# Data Encryption Standard (DES)

Im Auftrag des National Institutes of Science and Technology (NIST) von IBM Ende der 1970'er entwickelt

Blockcipher (feste Ein- und Ausgabelänge)

Inverse Funktion  $DES^{-1}$  mit  $DES^{-1}(k, DES(k, m)) = m$



## Sicherheit von DES

kann alle Schlüssel in  $2^{56}$  Operationen testen

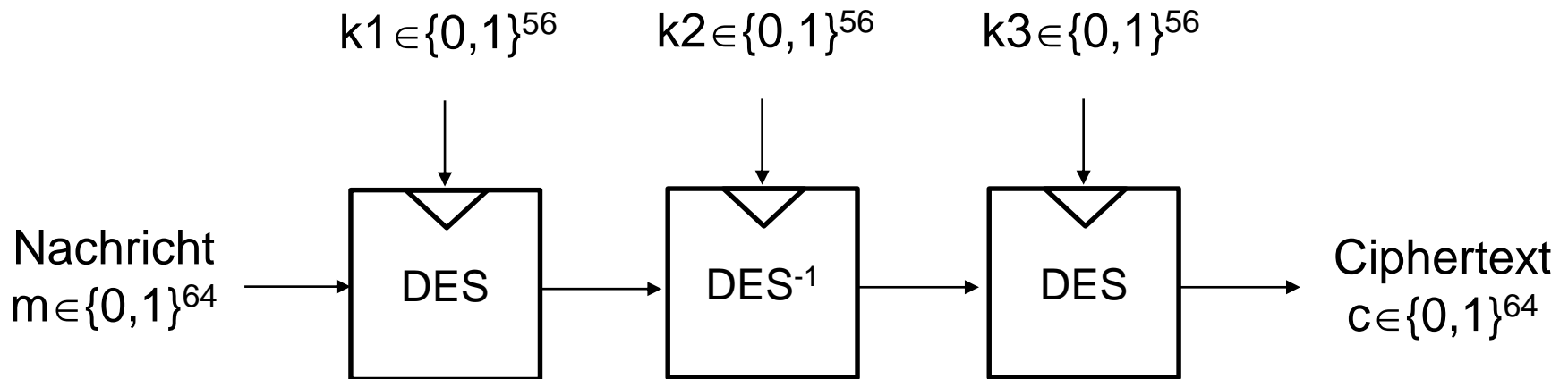
Sicherheitsniveau  
max. 56 Bits

Schlüssellänge mit 56 Bits (+8 Parity-Bits)  
nach heutigem Stand zu kurz

Sicherheitsniveau  
max.  $2 \times 56 = 112$  Bits

heute noch in der Form „Triple-DES“ verwendet:  
 $3DES(k_1|k_2|k_3, m) = DES(k_3, DES^{-1}(k_2, DES(k_1, m)))$

→ „Einführung in  
die Kryptographie“

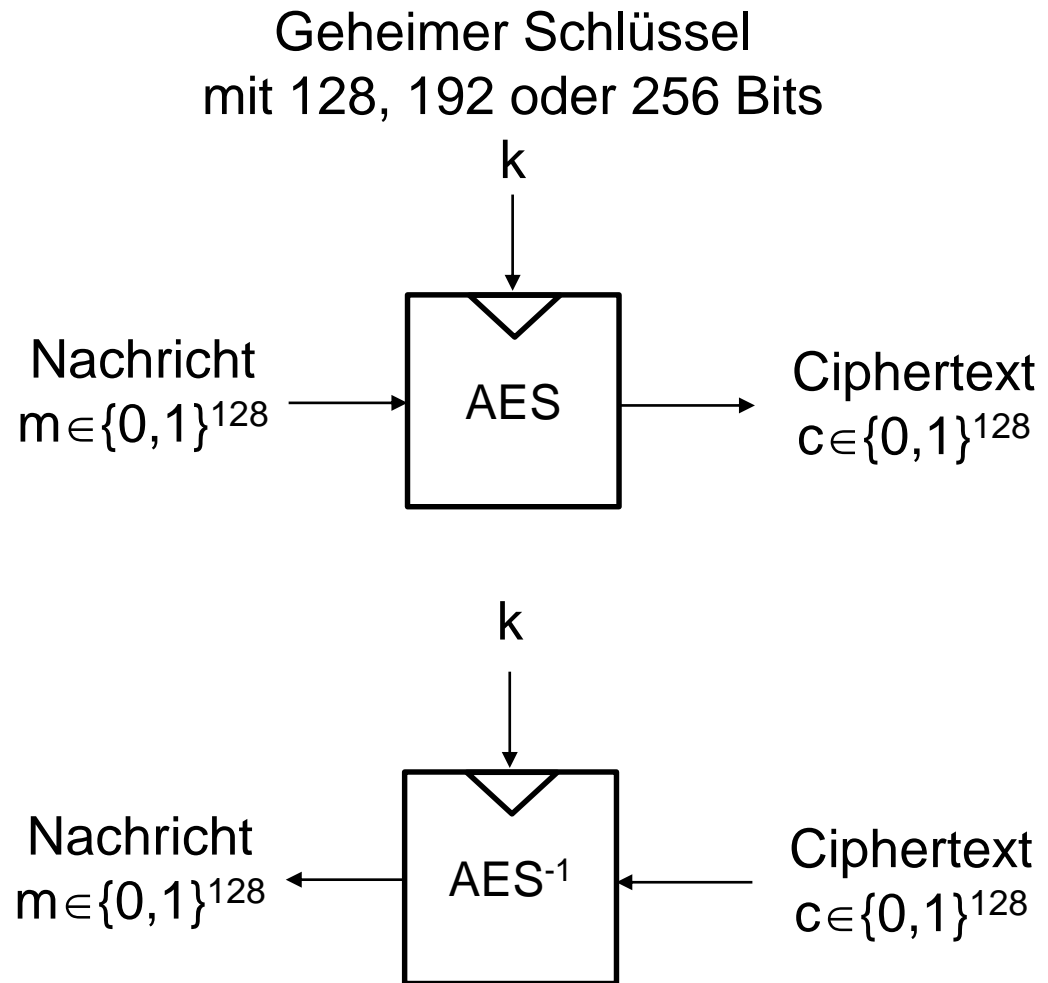


# Advanced Encryption Standard (AES)

In öffentlichem Wettbewerb von der NIST im Jahr 2000 bestimmt (Sieger Rijndael → AES)

Blockcipher (feste Ein- und Ausgabelänge, drei mögliche Schlüssellängen AES-128, AES-192, AES-256)

Inverse Funktion  $AES^{-1}$  mit  $AES^{-1}(k, AES(k, m)) = m$



# Evaluation AES

gilt aktuell als ungebrochen

AES ist der de-facto-Standard  
und sollte gegenüber 3DES  
bevorzugt werden;  
„reines“ DES sollte nicht mehr  
verwendet werden

In den USA sind AES-192 und AES-256  
für höchste Geheimhaltungsstufe zugelassen

spezielle AES-Hardware-  
Unterstützung in Intel  
und AMD-Prozessoren

→ „Embedded  
System Security“



## What is It?

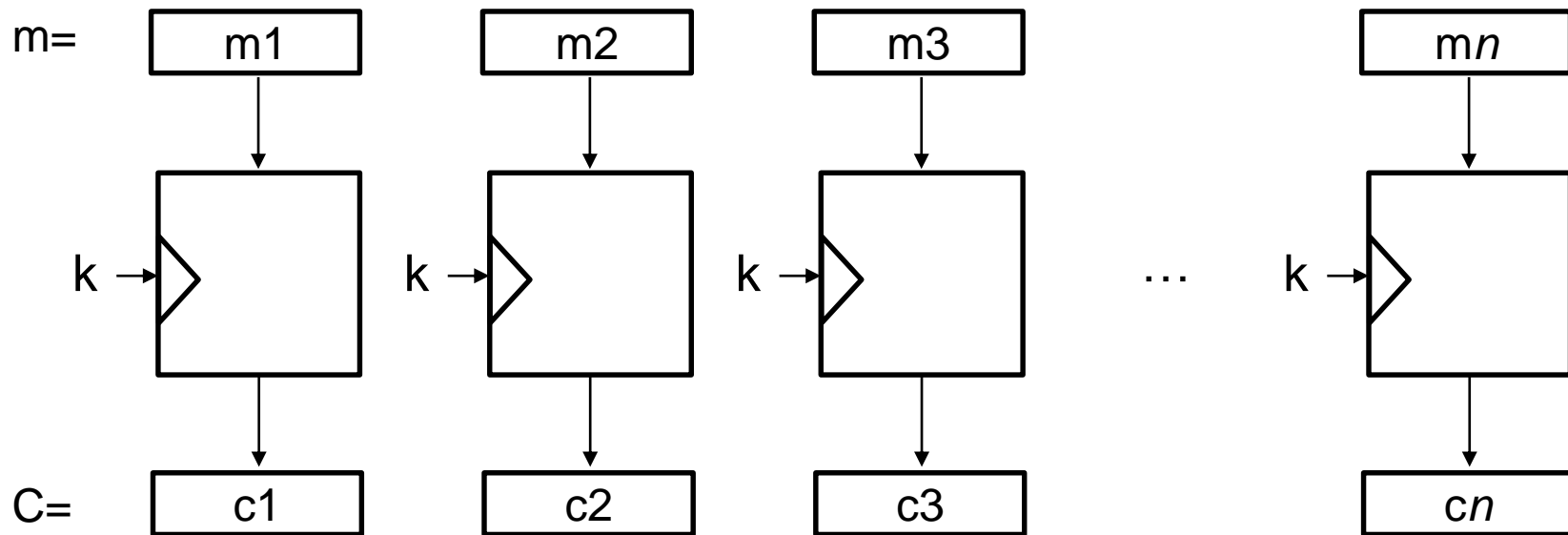
Intel® AES New Instructions (Intel® AES NI) is a new encryption instruction set that improves on the Advanced Encryption Standard (AES) algorithm and accelerates the encryption of data in the Intel® Xeon® processor family and the Intel® Core™ processor family.

Comprised of seven new instructions, Intel® AES-NI gives your IT environment faster, more affordable data protection and greater security; making pervasive encryption feasible in areas where previously it was not.

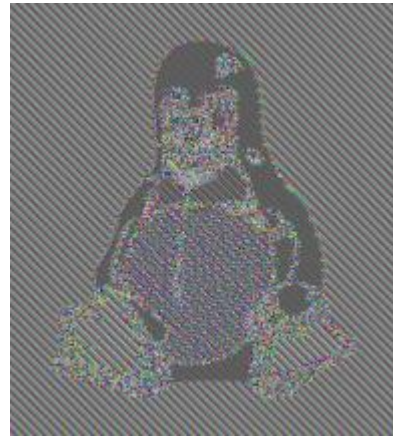
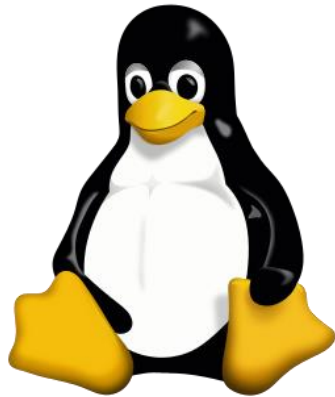
# Verschlüsseln langer Nachrichten?

Mit Hilfe von speziellen Modi für den Blockcipher

Electronic Code Book (ECB) Mode:



## ECB ist (in der Regel) keine gute Idee



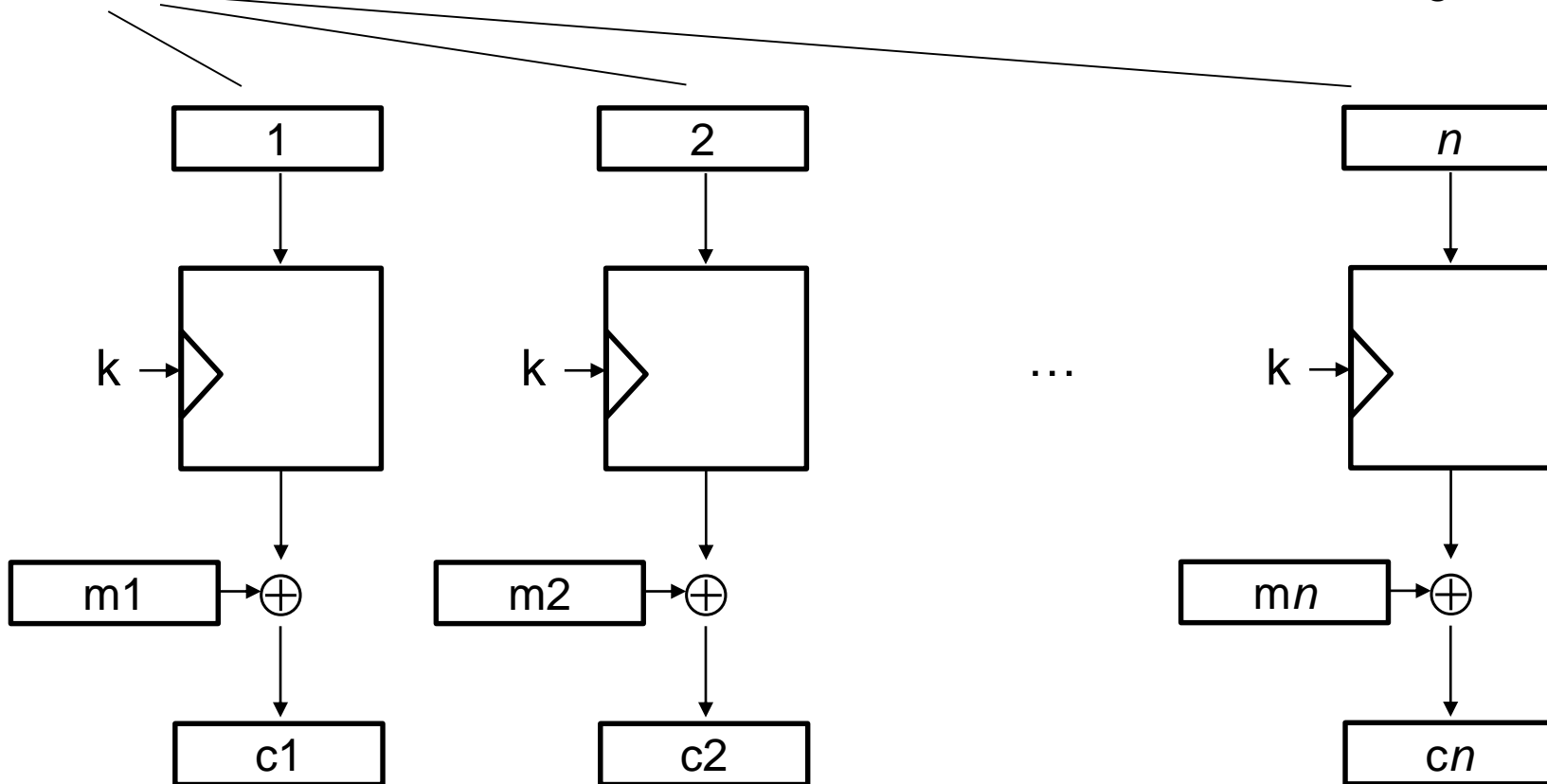
Quelle: Wikipedia

ECB-Verschlüsselung behält viel Struktur bei:  
gleiche Nachrichtenblöcke werden in gleiche Ciphertextblöcke verschlüsselt

## „ECB mit Zähler“ → Counter Mode

in vielen Standards so verwendet  
als Galois/Counter Mode (GCM)

Zählerwert muss verschieden sein, auch über mehrere Verschlüsselungen hinweg!

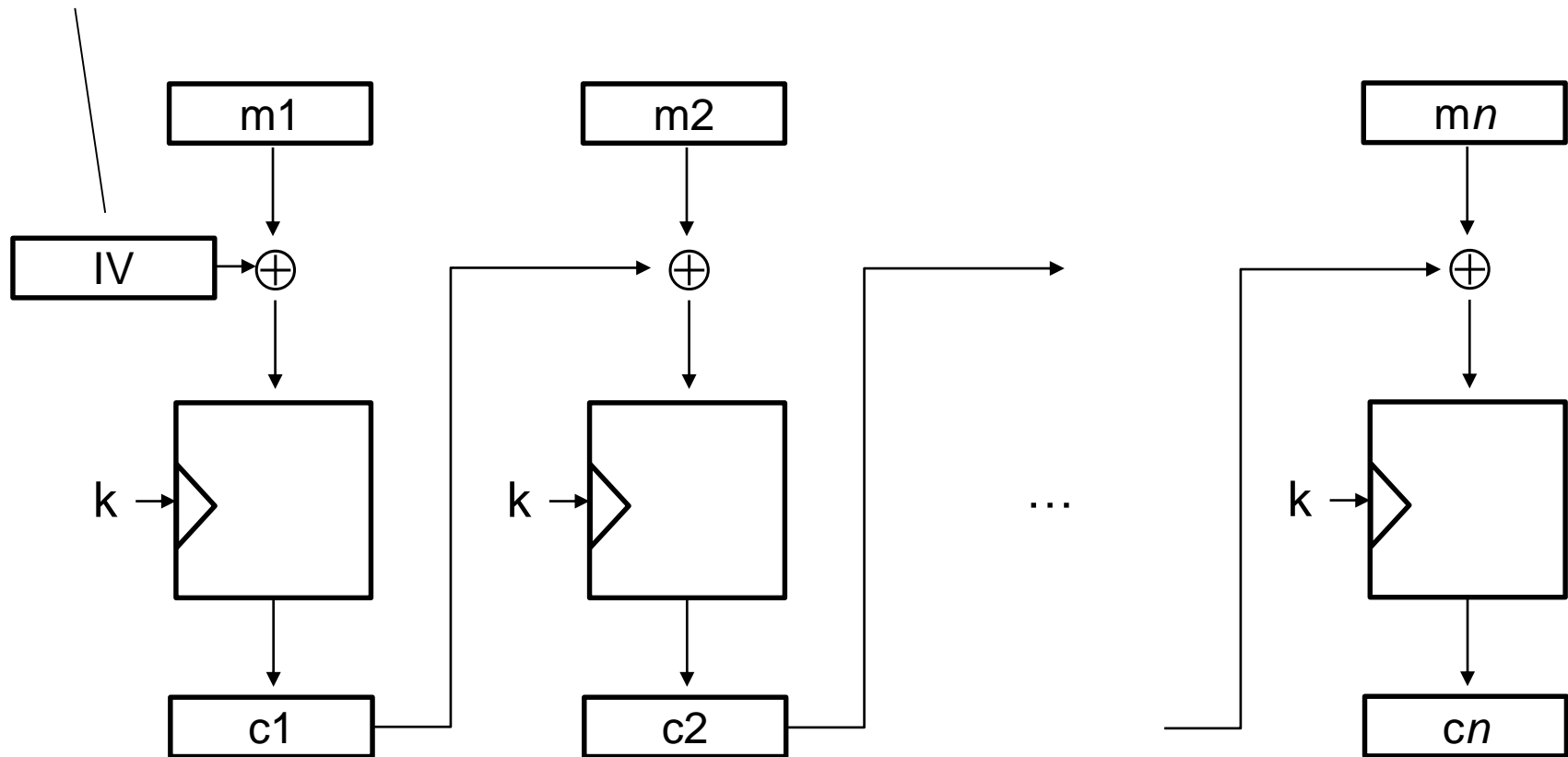




# Cipher Block Chaining (CBC)

auch in vielen  
Standards zu finden

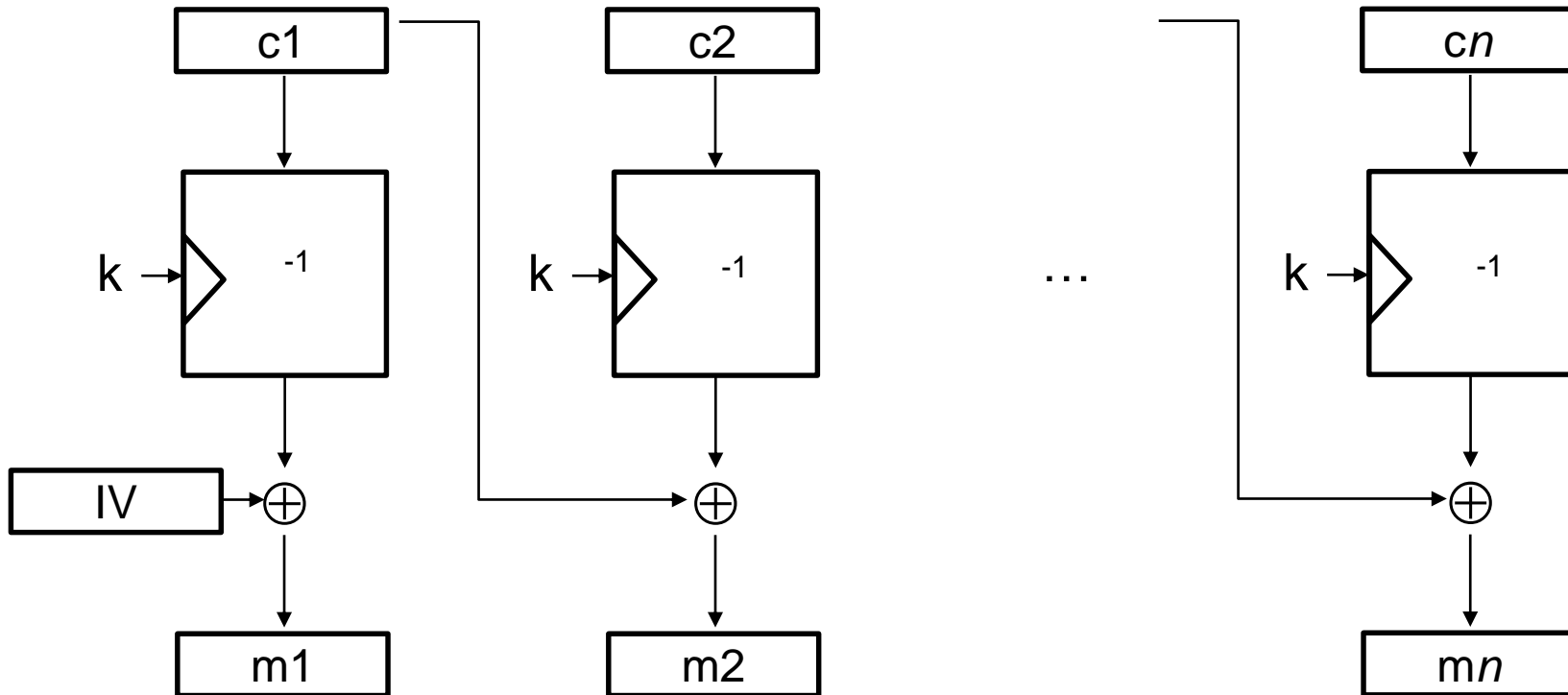
zufälliger Initialisierungsvektor (IV) wird Teil des Ciphertexts  $C=(IV,c1|...|cn)$



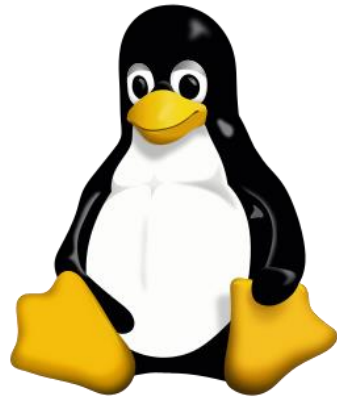
## CBC-Entschlüsselung

Verschlüsselung:  $c_j = E(k, m_j \oplus c_{j-1})$  mit  $c_0=IV$

Entschlüsselung:  $m_j = D(k, c_j) \oplus c_{j-1}$  mit  $c_0=IV$



# „Gute“ Verschlüsselung des Pinguins



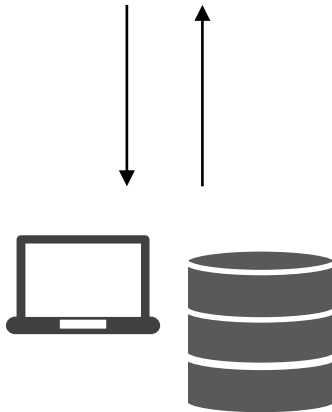
Quelle: Wikipedia

# Symmetrische Verschlüsselung von Dateien

Schlüssel  $k$



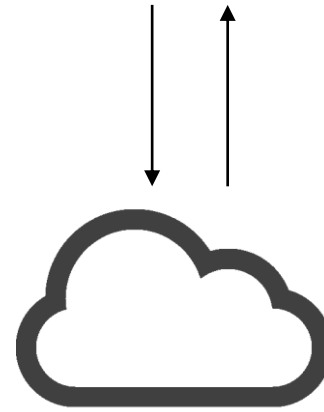
$\text{Enc}(k, \text{Datei})$      $\text{Dec}(k, \text{Datei})$



Schlüssel  $k$



$\text{Enc}(k, \text{Datei})$      $\text{Dec}(k, \text{Datei})$



Problem:

verhindert Deduplication,

dass Cloud für alle User  
nur eine Kopie einer  
Datei speichert, um  
Platz zu sparen



# Angriffe auf CBC-Padding

## Security Flaws Induced by CBC Padding Applications to SSL, IPSEC, WTLS...

Serge Vaudenay

Swiss Federal Institute of Technology (EPFL)  
Serge.Vaudenay@epfl.ch

**Abstract.** In many standards, e.g. SSL/TLS, IPSEC, WTLS, messages are first pre-formatted, then encrypted in CBC mode with a block cipher. Decryption needs to check if the format is valid. Validity of the format is

Eurocrypt 2002

Angriff auf bestimmte  
Anwendung von CBC,  
nicht auf CBC an sich

viele praktische Angriffe auf weit verbreitete Protokolle:

**BEAST** <sup>2011</sup>

**Browser Exploit  
Against SSL/TLS**

**Lucky Thirteen** <sup>2013</sup>

**Poodle** <sup>2014</sup>

...

**Padding Oracle On Down-  
graded Legacy Encryption**



## CBC-Padding

Was machen, wenn letzter Nachrichtenblock kürzer?

Auffüllen (padding):

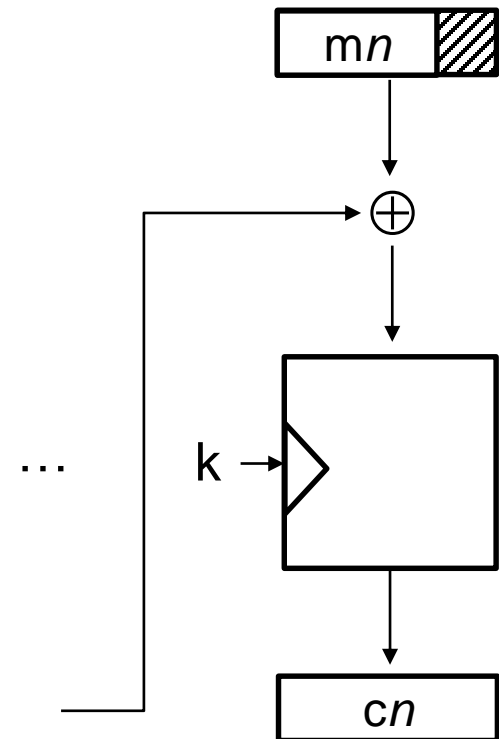
Wenn 1 Byte fehlt, hänge 1 Byte **0x01** an

Wenn 2 Bytes fehlen, hänge 2 Bytes **0x02** an

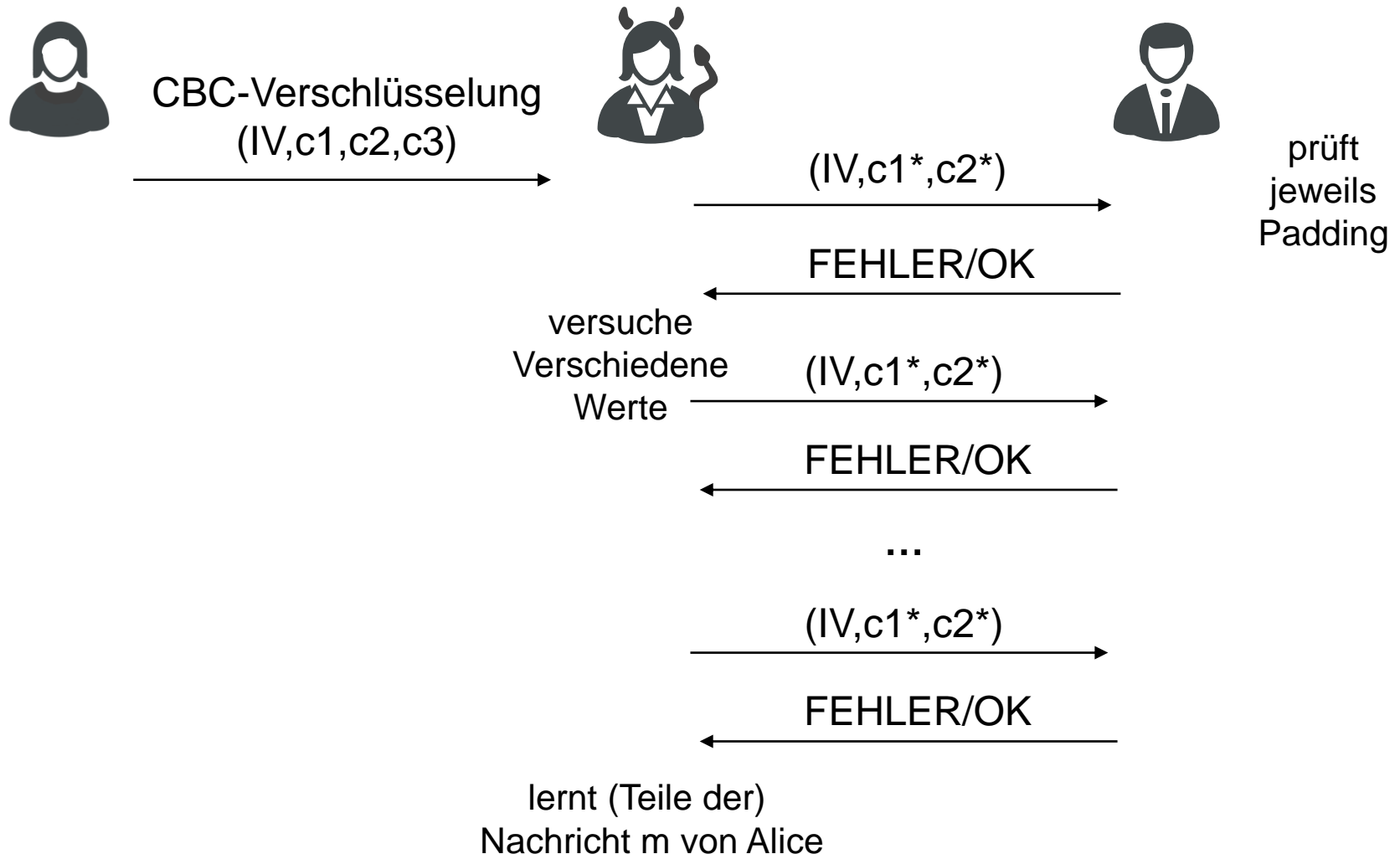
Wenn 3 Bytes fehlen, hänge 3 Bytes **0x03** an

Wenn 4 Bytes fehlen, hänge 4 Bytes **0x04** an

...



# Angriffsszenario





## Angriff

$(IV, c1, c2, c3) \rightarrow (IV, c1^*, c2^*)$

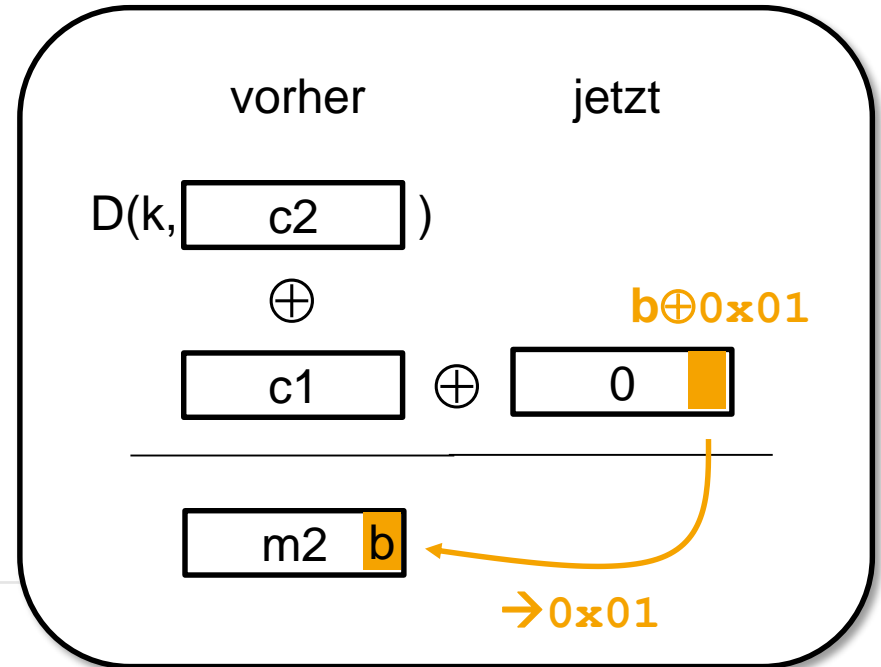
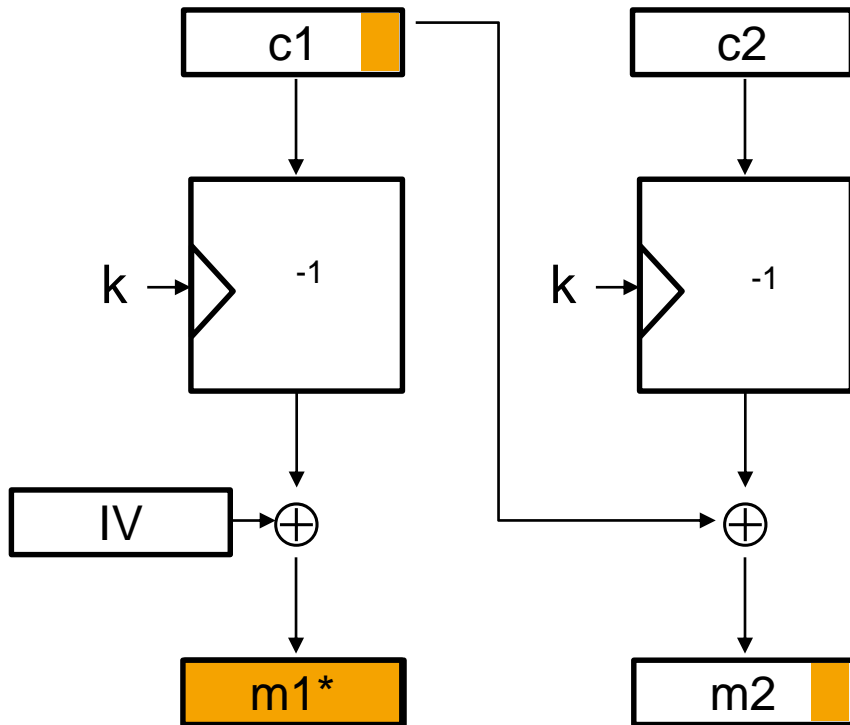
Rate letztes Byte b von m2

Setze  $c1^* = c1 \oplus b \oplus 0x01$

Setze  $c2^* = c2$

links mit  
0en auffüllen

Wenn b richtig geraten,...



...dann hier  
korrektes Padding  $0x01$





## Angriff iterieren

Nach maximal  $2^8=256$  Versuchen Byte  $b$  richtig bestimmt.

(Wir ignorieren hier die kleine Fehlerwahrscheinlichkeit, dass auch ein falsches Bit ein korrektes Padding ergeben kann.)

Wiederhole Ansatz:

Rate vorletztes Byte  $b'$  von  $m_2$

Setze  $c1^* = c1 \oplus b' | b \oplus 0x02 | 0x02$

Setze  $c2^* = c2$

usw. bis  $m_2$  vollständig bestimmt

Für AES mit 16 Bytes beispielsweise nach maximal  $16 * 256 = 4096$  Versuchen bestimmt



Nennen Sie drei technische Unterschiede zwischen DES und AES.

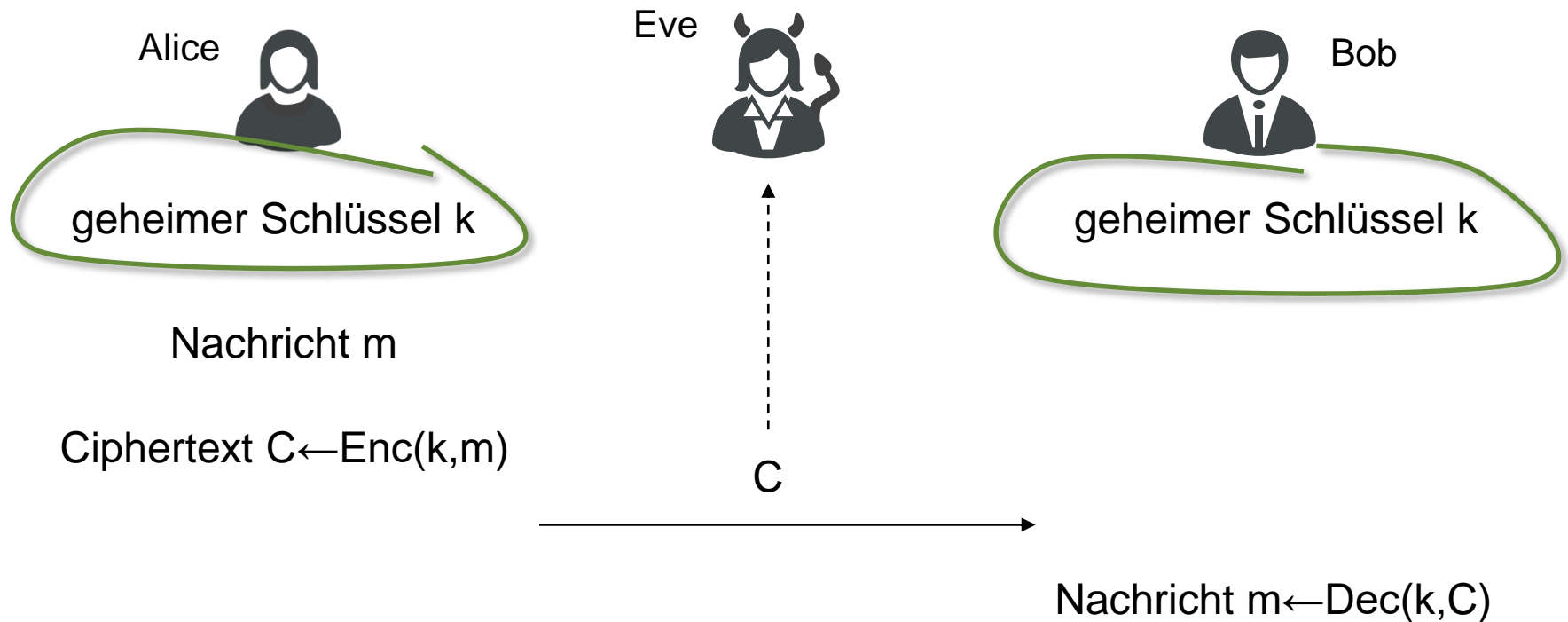


Bestimmen Sie die Umkehrfunktion zu Triple-DES:  
 $3DES(k_1|k_2|k_3, m) = DES(k_3, DES^{-1}(k_2, DES(k_1, m)))$



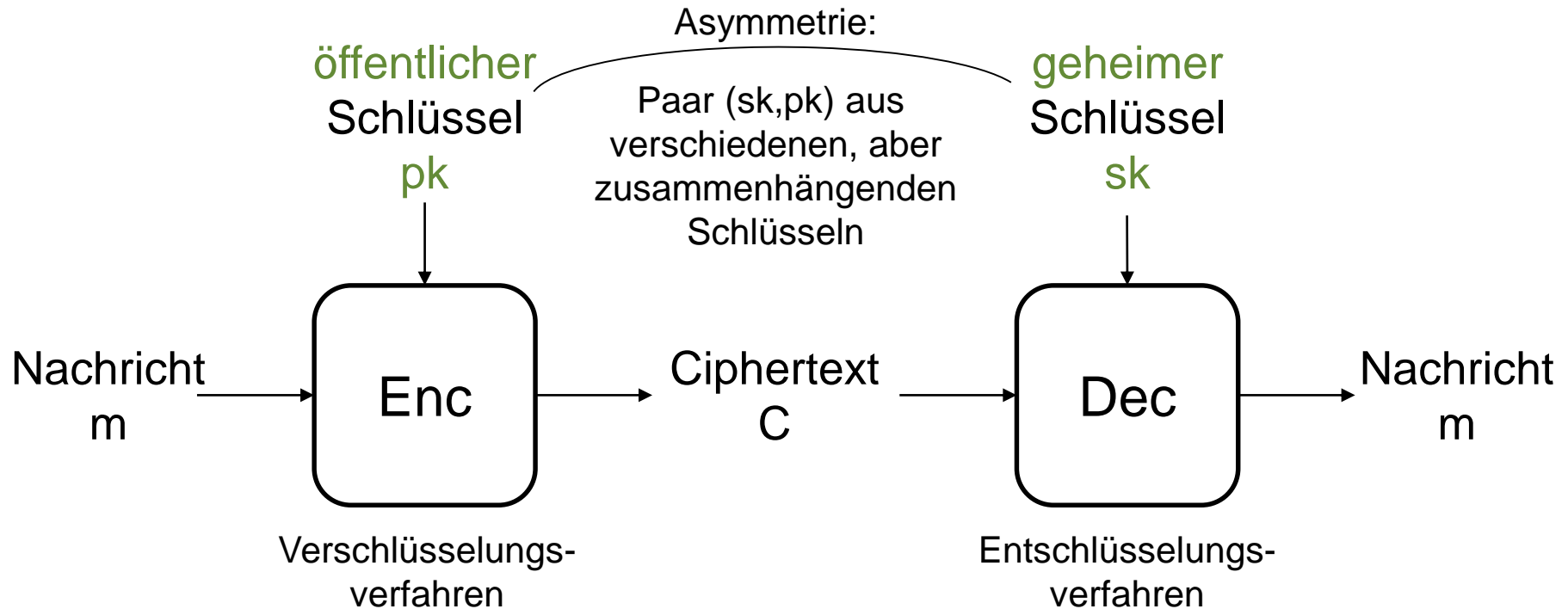
Warum müssen Sie bei AES-CBC-Verschlüsselung mit Padding auch noch 16-mal 0x10 anhängen, selbst wenn der letzte Nachrichtenblock schon auf Blocklänge ist?

# Public-Key (oder: asymmetrische) Verschlüsselung



Wie haben sich Alice und Bob auf den gemeinsamen Schlüssel geeinigt?

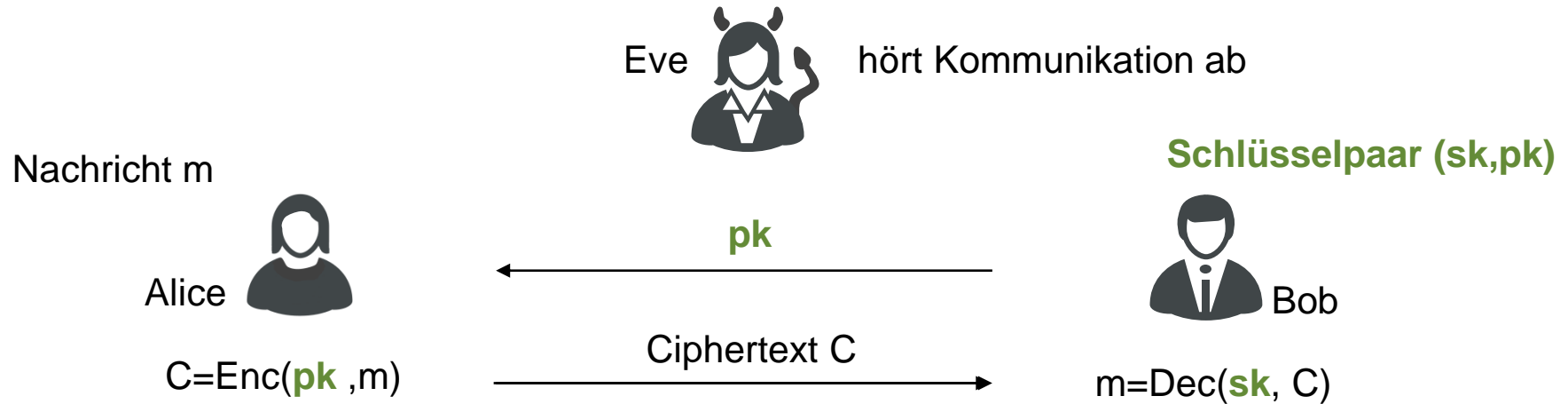
# Prinzip der Public-Key-Verschlüsselung



Funktionale Korrektheit (Vollständigkeit):

Für alle Nachrichten  $m$  und alle Paare  $(sk, pk)$  gilt:  $Dec(sk, Enc(pk, m)) = m$

# Public-Key-Verschlüsselung



Unterschied zur vorigen Private-Key-Verschlüsselung:  
vertrauliche Kommunikation „mit Fremden“ möglich

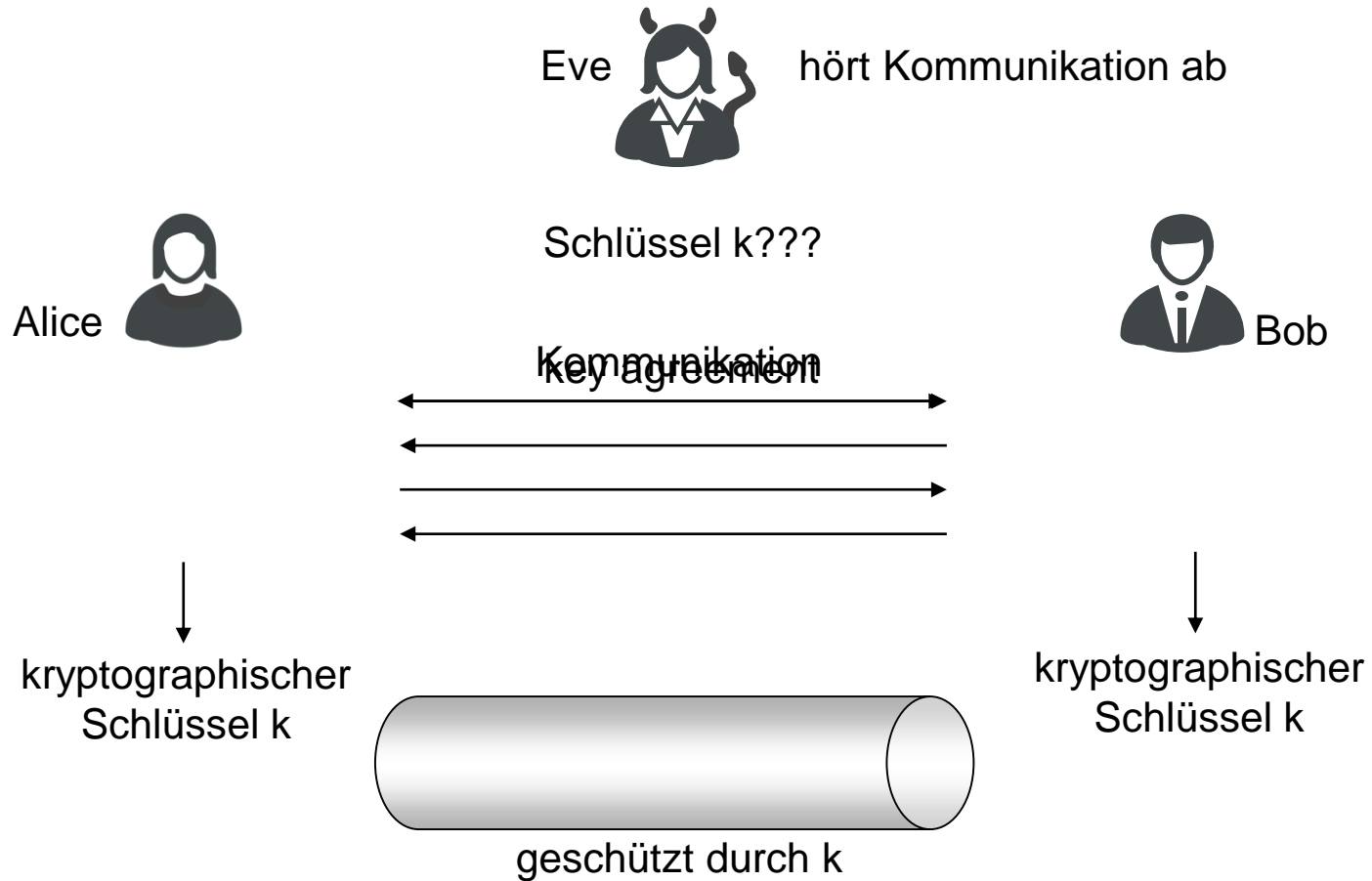
Synonyme:

symmetrisch = private-key

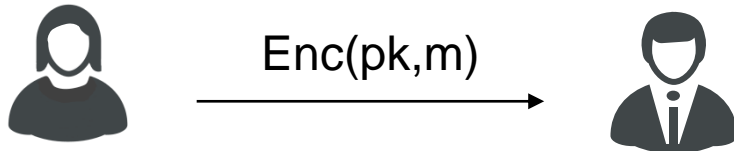
asymmetrisch = public-key

# Schlüsselaustausch

W.Diffie, M.Hellman: New Directions in Cryptography, IEEE Transactions on Information Theory, 1976

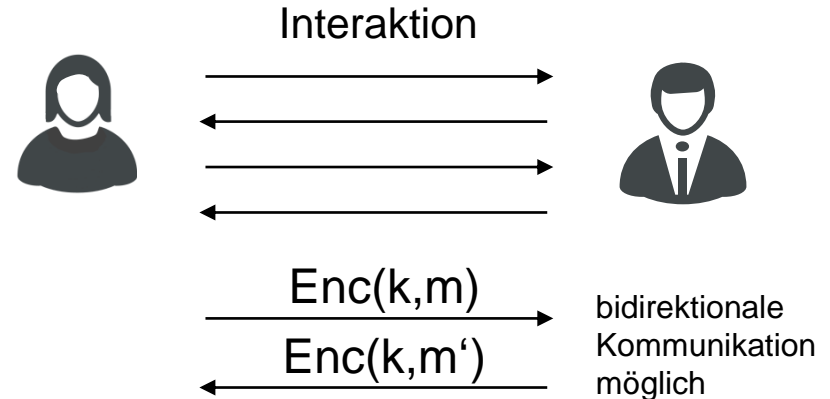


# Public-Key-Verschlüsselung vs. Schlüsselaustausch



Key Transport ergibt Schlüsselaustausch:  
Erst  $\text{Enc}(\text{PK}, k)$ , dann  $\text{Enc}(k, m)$ ,  $\text{Enc}(k, m')$

geeignet, wenn keine Interaktion:



andere Vorteile wie  
Authentisierung der Partner,  
Forward-Security,...

geeignet, wenn sowieso Interaktion:





# RSA-Verschlüsselung

Rivest, Shamir, Adleman  
"A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". Communications of the ACM, 1978

Rivest

Shamir

Adleman



from Len Adleman's homepage

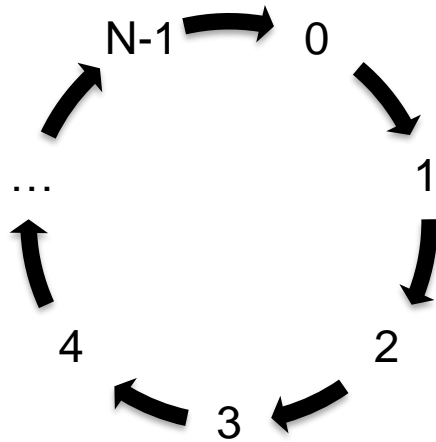
$$\text{Enc}((e, N), m) = m^e \bmod N, \quad \text{Dec}((d, N), C) = C^d \bmod N$$

# Modulare Arithmetik...

$$\text{Enc}(pk, m) = m^e \bmod N,$$

$$\text{Dec}(sk, C) = C^d \bmod N$$

...ist „Rechnen im Kreis“



Wenn man 1 zu N-1 addiert,  
landet man wieder bei 0



A	→	D
B	→	E
C	→	F
...		
W	→	Z
X	→	A
Y	→	B
Z	→	C

entspricht  
 $+3 \bmod 26$

z.B.  $x=23$  auf  
 $23+3=0 \bmod 26$ ,  
entspricht **A**

# Modulare Arithmetik für RSA

$$\text{Enc}(pk, m) = m^e \bmod N, \quad \text{Dec}(sk, C) = C^d \bmod N$$

hier wird multipliziert, nämlich  $m^e = \underbrace{m \cdot m \cdots m}_{e \text{ mal}} \bmod N$

Multiplikation  $a \cdot b \bmod N$  für  $a, b \in \{0, 1, \dots, N-1\}$

1. Berechne  $a \cdot b$  über ganzen Zahlen
2. Ziehe so oft  $N$  ab, bis Ergebnis zwischen 0 und  $N-1$

Beispiel:  $3 \cdot 5 = 1 \bmod 7$

# $\text{mod } N \neq \text{mod } N$

Zwei verschiedene Sichtweisen auf „mod N“:

als Struktur:



„Rechne im  
Restklassenring  $\mathbb{Z}_N$ “

Beispiel:  
 $3 \cdot 6 = 3 \text{ mod } 5$

sogar Schreibweise:  $3 \cdot 6 = 18 = 3 \text{ mod } 5$

als Operator:  $f(x)$

„bilde x auf die kleinste Zahl y  
zwischen 0 und N-1  
mit  $x = y \text{ mod } N$  ab“

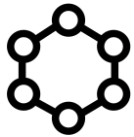
Beispiel:  
 $17 \text{ mod } 5 = 2$

alternative Schreibweise:  $17 \% 5 = 2$

Beispiel:  $18 \stackrel{?}{=} 3 \text{ mod } 5$

–in der Struktur richtig, als Operator falsch

# mod N in der Vorlesung



Berechnungen üblicherweise in der Struktur:  
 $C = m^e = m \cdot \dots \cdot m \bmod N$

$f(x)$

Wenn Ausgabe eines Algorithmus'  
(z.B. Ciphertext aus Enc-Verfahren),  
dann Operator mod N anwenden und  
Zahl zwischen 0 und N-1 ausgeben

Beachte:

$$\begin{array}{ccc} a = b \bmod N & \Leftrightarrow & (a \bmod N) = (b \bmod N) \\ \text{(Struktur)} & & \text{(Operatoren)} \end{array}$$

# Funktionale Korrektheit von RSA

$$\text{Enc}(pk, m) = m^e \bmod N, \quad \text{Dec}(sk, C) = C^d \bmod N$$

Wähle  $e$  und  $d$  als ganze Zahlen passend zu  $N$ , so dass  $(m^e)^d = m \bmod N$   
(„ $d$  ist multiplikatives Inverses zu  $e \bmod \varphi(N)$ “)

Damit kann man Nachrichten zwischen 0 und  $N-1$  verschlüsseln, wenn man die Bits der Nachricht auf kanonische Weise als Zahl interpretiert.

(Man schränkt aber aus Sicherheitsgründen die Nachrichten auf die zu  $N$  teilerfremden Elemente aus  $\mathbb{Z}_N^*$  ein, also sind nur Nachrichten  $m$  mit  $\text{ggT}(m, N) = 1$  zulässig.)

# Sicherheit von RSA?

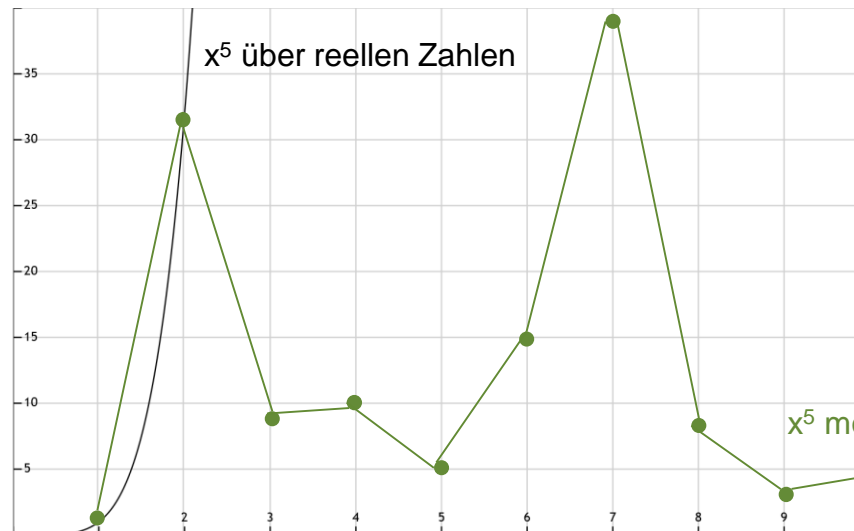
$$\text{Enc}(pk, m) = m^e \bmod N, \quad \text{Dec}(sk, C) = C^d \bmod N$$

Warum kann man als Angreifer nicht einfach die e-te Wurzel aus C ziehen?

Die (spezielle) modulare Arithmetik macht das schwierig:



(teilweise) erstellt  
mit fooplplot.com



$$x \mapsto x^5$$

$$y \mapsto y^{1/5}$$

# Sicherheit von RSA?

$$\text{Enc}(pk, m) = m^e \bmod N, \quad \text{Dec}(sk, C) = C^d \bmod N$$

Wurzelziehen über reellen Zahlen ist einfach...

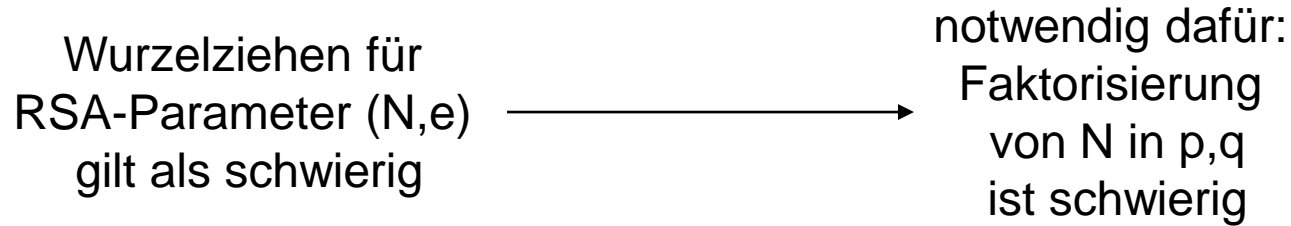
Wurzelziehen für modulare Arithmetik  
ist für Primzahlpotenzen  $p^k$  in der Regel einfach...



RSA verwendet daher  $N=pq$  für (verschiedene) zufällige Primzahlen  $p, q$ .



# Wahl der RSA-Parameter



Wenn man Faktorisieren kann, dann kann man RSA brechen (Umkehrung offen)

BSI-Empfehlungen für Parameter (Mai 2018):

[BSI TR-02102-1](#)

$N$  sollte mindestens 2000 Bits haben,  
weitere Anforderungen an  $p, q, e$  und  $d$

# Hybrid-Verschlüsselung

Zwei Nachteile von Public-Key-Verfahren:

- erlauben zunächst nur Verschlüsselung kurzer Nachrichten  
(z.B. < 2000 Bits für RSA)
- Symmetrische Verfahren wie AES sind wesentlich schneller  
als asymmetrische Verfahren wie RSA

Hybrid-Verschlüsselung: kombiniere Vorteile der beiden Typen:

Effizienz  
symmetrischer  
Verfahren      +      Kommunikation  
mit Fremden      =      Hybrid-Verschlüsselung

# Hybrid-Verschlüsselung ausführen

so wird in der  
Praxis verschlüsselt

Verschlüsselung:

1. Wähle kurzen Schlüssel  $k$  für symmetrisches Verfahren
2. Verschlüssele  $k$  mit asymmetrischem Verfahren
3. Verschlüssele lange Nachricht mit symmetrischem Verfahren unter  $k$

$$C = ( \text{PubKeyEnc}(\text{pk}, \boxed{k} ), \quad \text{SymEnc}(k, \boxed{m} ) )$$

z.B. 128-Bit-AES-Schlüssel  
in 2048-Bit-RSA-System

z.B. CBC-Verschlüsselung mit AES( $k, \cdot$ )

Entschlüsselung:

1. entschlüssele kurzen Schlüssel  $k$  für symmetrisches Verfahren
2. entschlüssele lange Nachricht mit symmetrischem Verfahren unter  $k$

In der Praxis verwendet man RSA nicht in der hier vorgestellten einfachen Form zum Verschlüsseln!

Man muss die Nachricht noch vorverarbeiten,  
um Sicherheit zu gewährleisten  
(z.B. wie in RSA-OAEP – Optimal Asymmetric Encryption Padding)

z.B. kann sonst für „kleine“  $m$  gelten:  
 $m^e = m$  über ganzen Zahlen (ohne mod  $N$ -Reduktion)

z.B. gilt sonst auch für  $C1 = (m1)^e \bmod N$  und  $C2 = (m2)^e \bmod N$ , dass  
 $C = C1 \cdot C2 = (m1)^e \cdot (m2)^e = (m1 \cdot m2)^e \bmod N$   
eine gültige Verschlüsselung von  $m1 \cdot m2$  ist

# Schlüsselaustausch nach Diffie und Hellman

Beruh auf dem Diskreten-Logarithmus-Problem:

schwierig aus  $g^x \bmod p$  den zufälligen Wert  $x$  zu bestimmen,  
wobei  $g$  geeignetes Element mod Primzahl  $p$

Gilt wegen modularer Arithmetik auch als schwierig

BSI-Empfehlung (Mai 2018):

$p$  mindestens 2000 Bits,  
über Elliptischen Kurven mindestens 250 Bits

# Diffie-Hellman Key Agreement



Alice

public:  $g$  and  $p$



Bob

choose  $x$

compute  $X = g^x \bmod p$

choose  $y$

compute  $Y = g^y \bmod p$

$Y$

$X$

Key

$$K = Y^x = (g^y)^x = g^{xy} \bmod p$$

Key

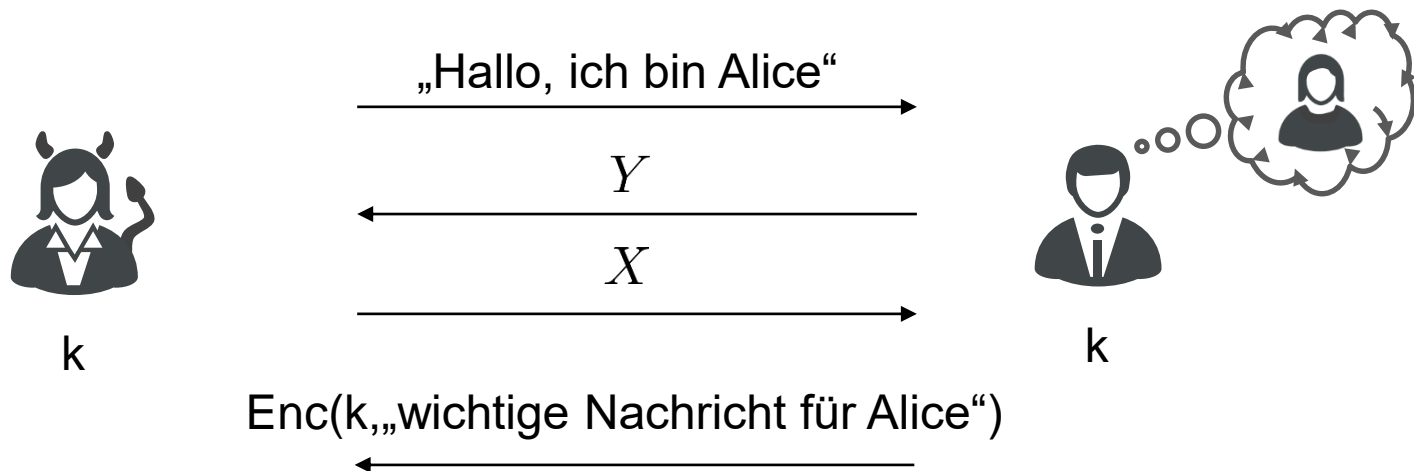
$$K = X^y = (g^x)^y = g^{xy} \bmod p$$



Eve cannot compute  $K$  from  $X, Y$

In der Praxis verwendet man DH nicht in der hier vorgestellten einfachen Form zum Schlüsselaustausch!

Man muss z.B. noch das Problem der Authentisierung lösen.

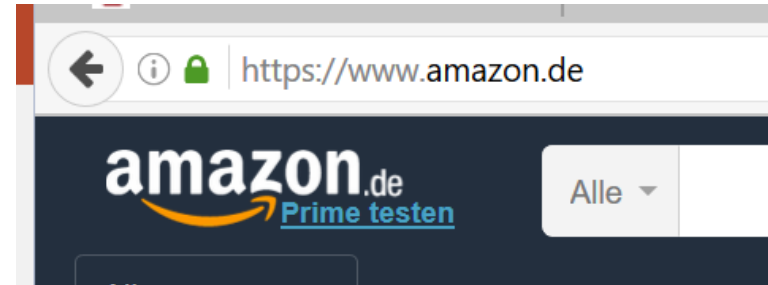


# Diffie-Hellman-Schlüsselaustausch ist überall

allerdings eingebettet in kompliziertere kryptographische Protokolle



Quelle: BMI



some.email@gmail.com

Server-Einstellungen

Kopien & Ordner

Verfassen & Adressieren

Junk-Filter

Synchronisation & Speicherplatz

OpenPGP-Sicherheit

Empfangsbestätigungen (MDN)

S/MIME-Sicherheit

Lokale Ordner

Server-Einstellungen

Servertyp: IMAP

Server: imap.gmail.com Port: 993 Standard: 993

Benutzername: some.email@gmail.com

Sicherheit und Authentifizierung

Verbindungssicherheit: SSL/TLS

Authentifizierungsmethode: OAuth2

Server-Einstellungen

Man kann aus DH-Key-Exchange auch das  
ElGamal-Public-Key-Verschlüsselungssystem bauen





Beschreiben Sie kurz das RSA-Verschlüsselungssystem.



Kann man aus jedem Public-Key-Verschlüsselungssystem ein Schlüsselaustauschverfahren konstruieren? Und umgekehrt?



Überlegen Sie sich, dass der Angreifer beim DH-Schlüsselaustausch den Wert  $g^{x+y}$  berechnen kann.

---

# Verschlüsselung und Pseudozufallsgeneratoren

---

Wie erzeugt man die zufälligen Schlüssel und Werte (wie den IV bei CBC)?

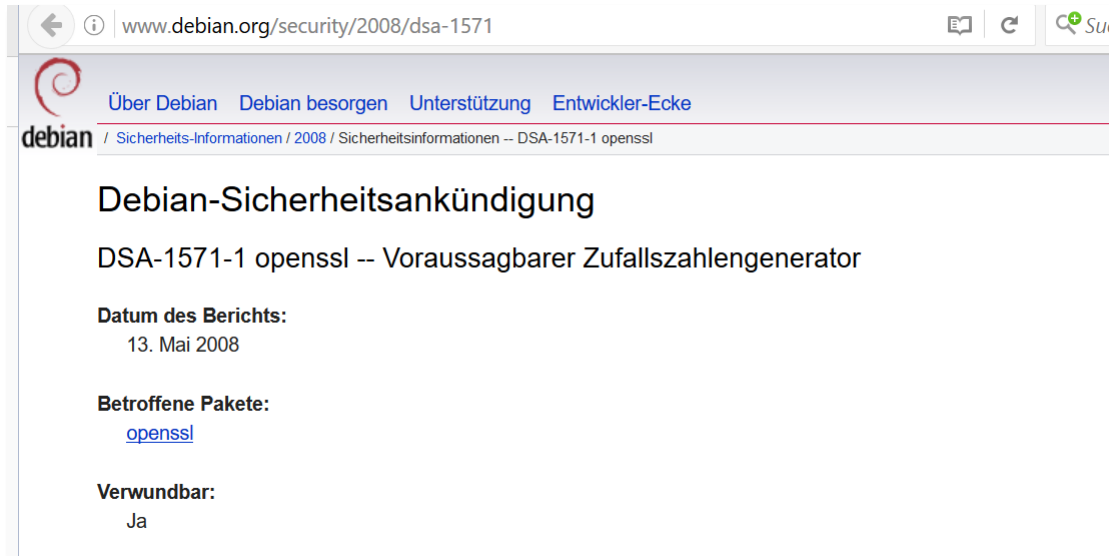
„Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin.”

*John von Neumann*

Wenn man's richtig macht, helfen sogenannte Pseudozufallsgeneratoren.

Aber nur, wenn man's richtig macht.

# Schwache RSA-Schlüssel

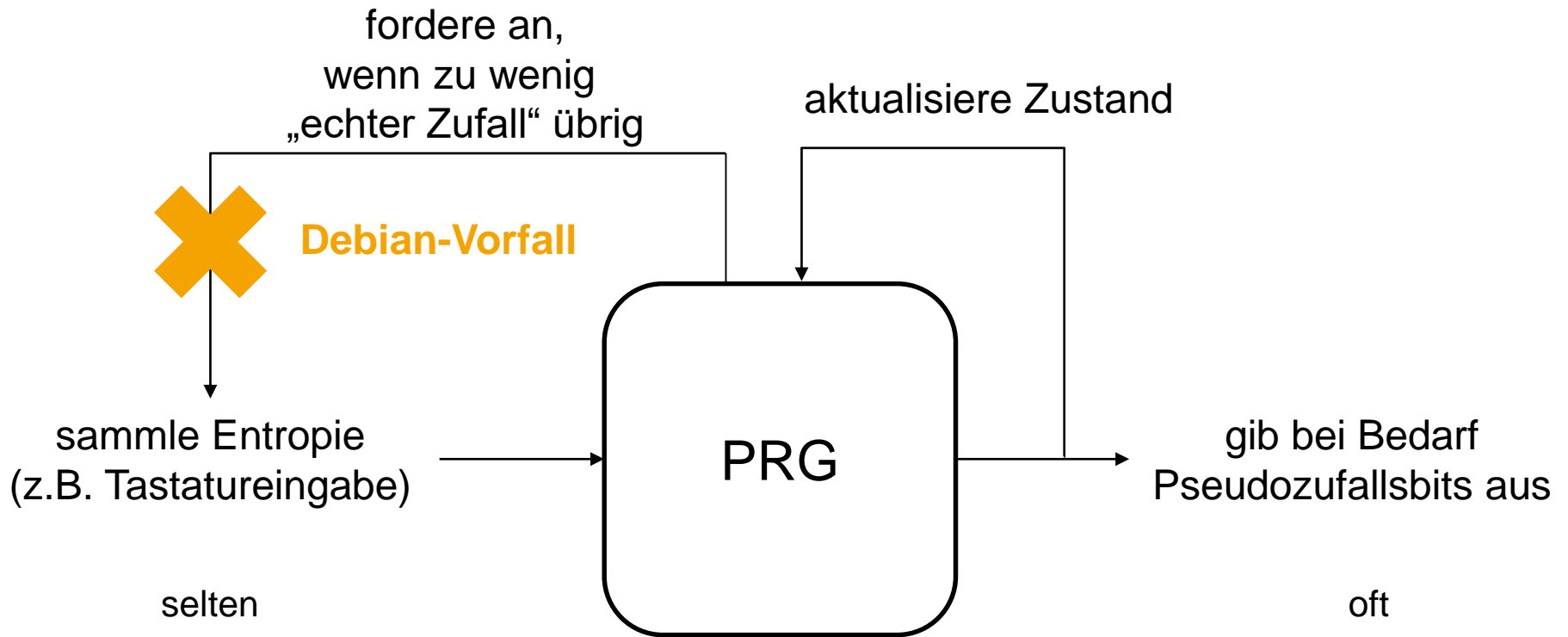


Angriff auf bestimmte  
Anwendung von RSA,  
nicht auf RSA an sich

Schlüssel für (beide Arten von) Verschlüsselungssystemen  
müssen gut gewählt werden

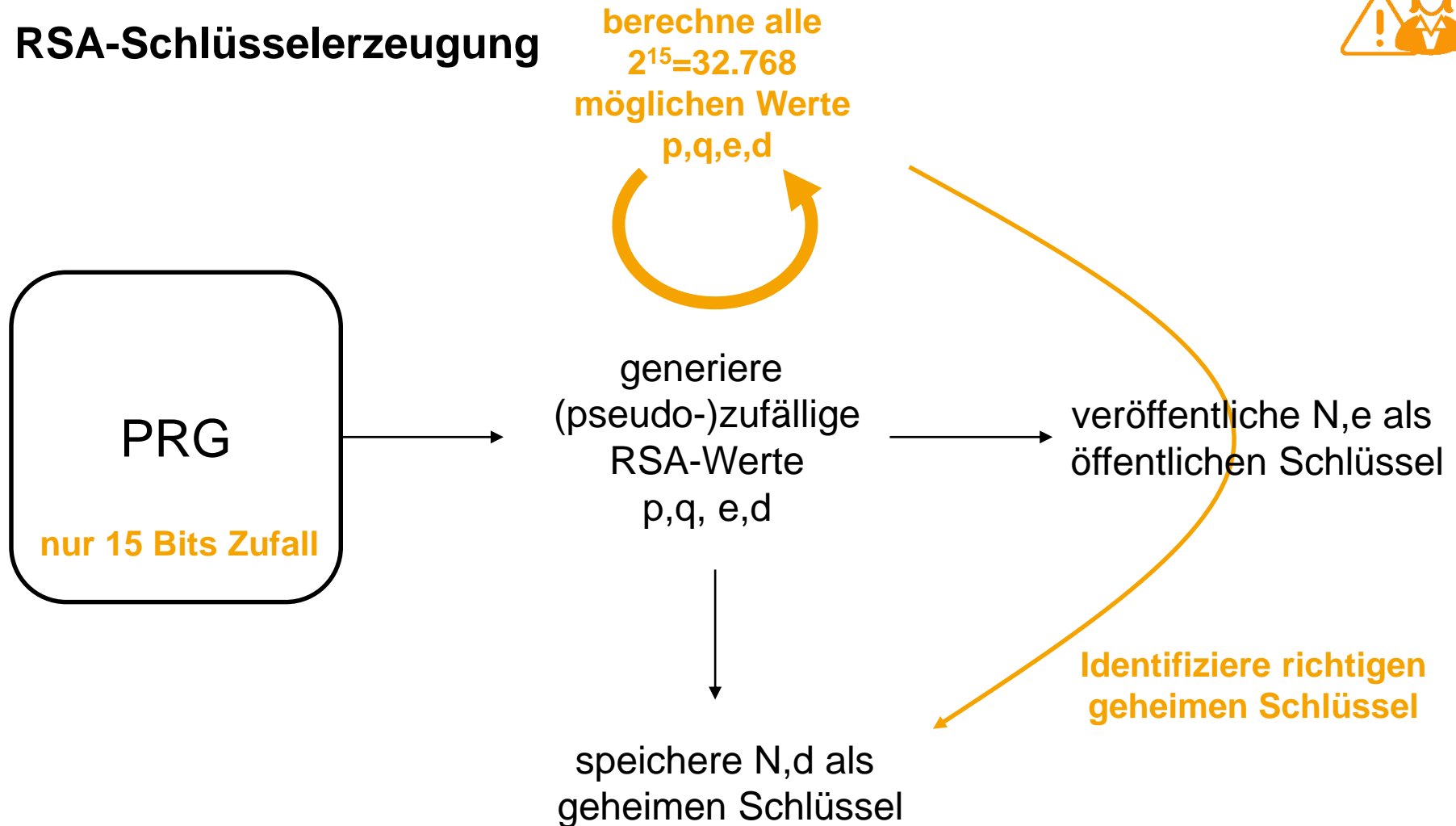


# Pseudozufallsgeneratoren (PRGs)



Beispiele: `/dev/random` und `/dev/urandom`

# RSA-Schlüsselerzeugung





## Weitere schwache Schlüssel

A shorter version of this paper will appear in *Proc. 21st USENIX Security Symposium*, Aug. 2012. Rev. 2; July 11, 2012. For the newest revision of this paper, partial source code, and our online key-check service, visit <https://factorable.net>.

### Mining Your Ps and Qs: Detection of Widespread Weak Keys in Network Devices

Nadia Heninger<sup>†\*</sup> Zakir Durumeric<sup>‡\*</sup> Eric Wustrow<sup>‡</sup> J. Alex Halderman<sup>‡</sup>

<sup>†</sup>*University of California, San Diego*  
nadiah@cs.ucsd.edu

<sup>‡</sup>*The University of Michigan*  
{zakir, ewust, jhalderm}@umich.edu

#### Abstract

RSA and DSA can fail catastrophically when used with malfunctioning random number generators, but the extent to which these problems arise in practice has never been comprehensively studied at Internet scale. We perform

expect that today's widely used operating systems and server software generate random numbers securely. In this paper, we test that proposition empirically by examining the public keys in use on the Internet.

The first component of our study is the most comprehensive Internet-wide survey to date of two of the most

## Ansatz:

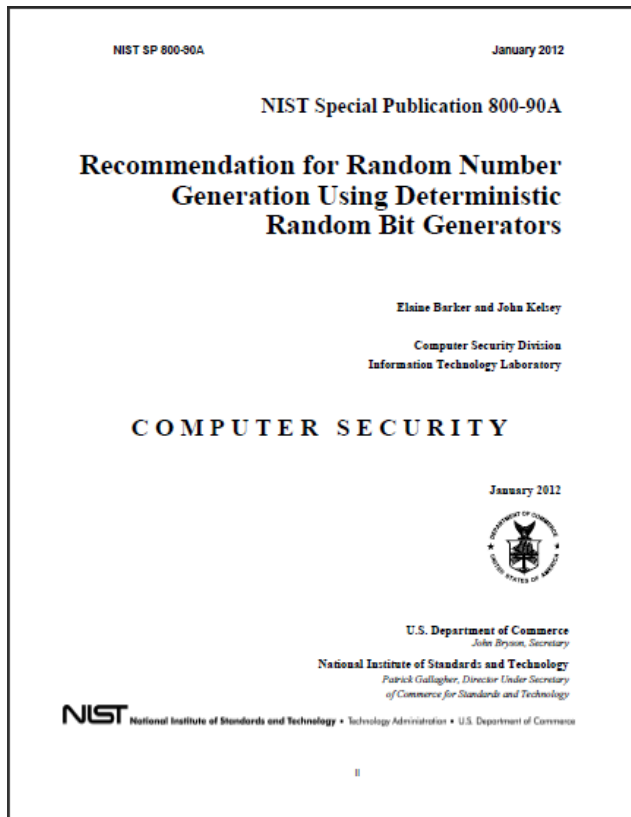
Wenn sich  $N=pq$  und  $N'=pq'$  einen Primfaktor  $p$  teilen,

dann liefert  $\text{ggT}(N, N')=p$

und damit dann auch  $q, q'$  und  $d, d'$

konnten von ca. 10 Mio öffentlichen RSA-Schlüsseln im Netz für ca. 100.000 den geheimen Schlüssel berechnen (ca. 1%)

# Der Fall Dual\_EC\_DRBG



National Institute of Standards and Technology  
(NIST)

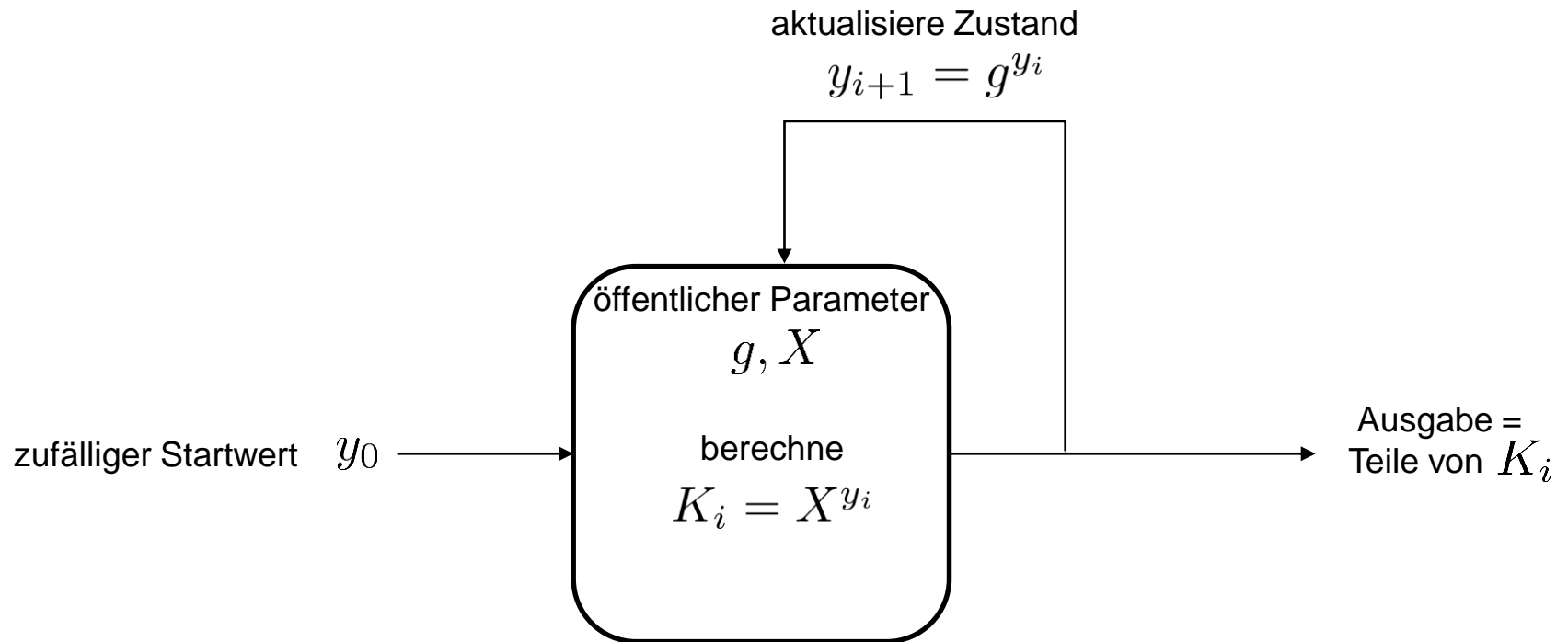
Sicherheitsstandards  
„Special Publications (800 series)“

teilweise explizit übernommen in Standards der  
International Organization for Standards  
(ISO)

SP800-90A in ISO 18031

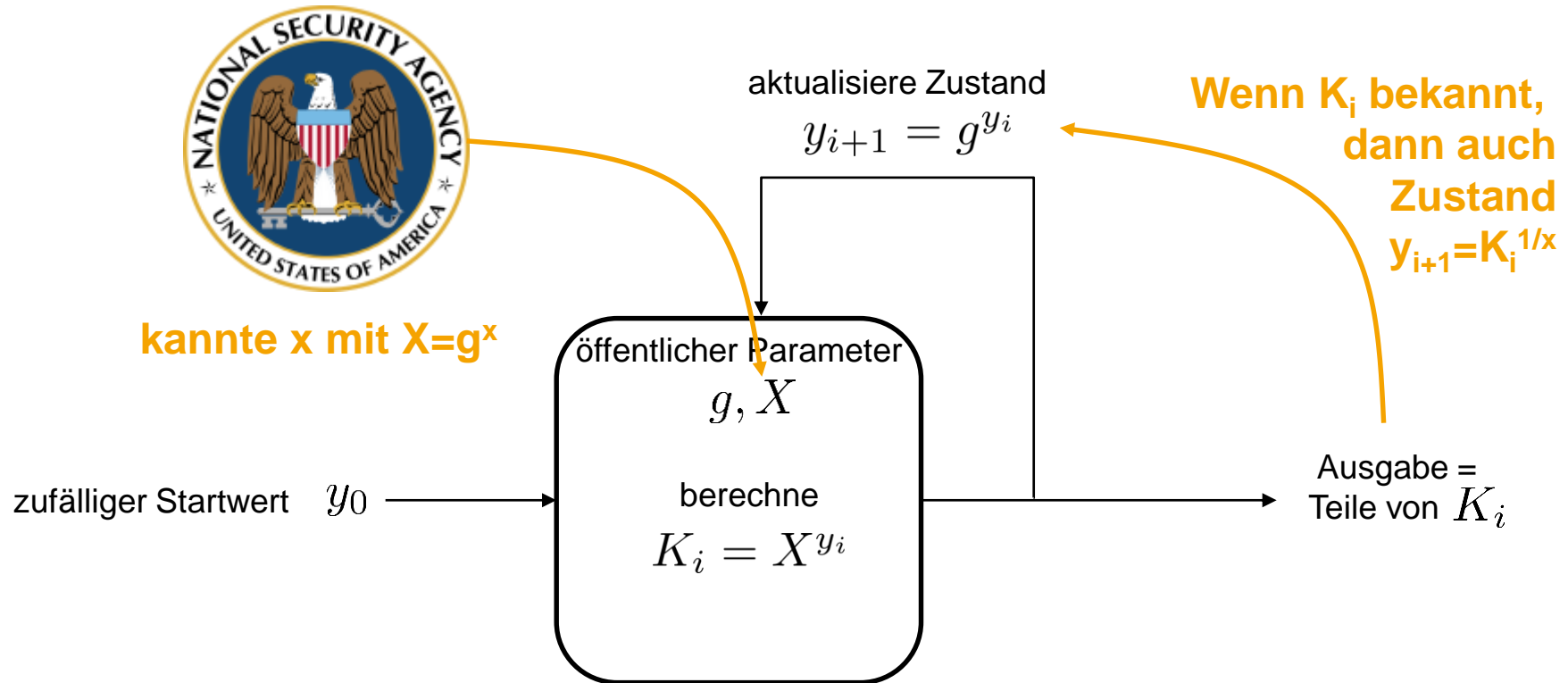


# Dual\_EC\_DRBG



Design-Idee prinzipiell ok:  
Diffie-Hellman-Schlüssel  $K$  aus  $X$  und  $g^y$  sollte gut sein

# Angriff auf Dual\_EC\_DRBG



Beim DH-Austausch kennen  
aber Zwei das Geheimnis!

Design-Idee prinzipiell ok:  
Diffie-Hellman-Schlüssel  $K$  aus  $X$  und  $g^y$  sollte gut sein

# Geschichte

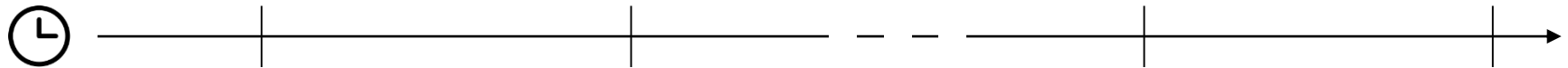


Forscher beschreiben  
Möglichkeit, dass DUAL\_EC\_DRBG  
Hintertüren haben könnte  
2007

neuer SP800-90A  
ohne DUAL\_EC\_DRBG  
2015

SP800-90A veröffentlicht  
2006

NIST rät von Verwendung  
von DUAL\_EC\_DRBG ab  
2013



2006  
Forscher entdecken  
erste Schwächen  
im DUAL\_EC\_DRBG

2013  
NY Times Artikel bestätigt  
absichtliche Hintertür für NSA



---

# Was Sie gelernt haben sollten

---

Symmetrische Verschlüsselung

Kandidaten in der Praxis und Verarbeitungsmodi

Asymmetrische Verschlüsselung

Kandidaten in der Praxis

Unterschied symmetrische und asymmetrische Verschlüsselung

Hybrid-Verschlüsselung

Schlüsselaustauschverfahren

Kandidaten in der Praxis

Unterschied asymmetrische Verschlüsselung und Schlüsselaustausch  
(Pseudozufallsgeneratoren)