

Weniger Klartext reden!

Dipl.-Math. Bastian Rieck

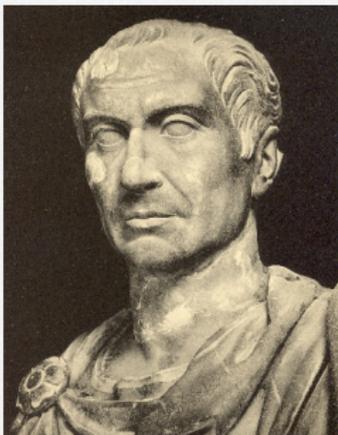
Arbeitsgruppe Computergraphik und Visualisierung
Interdisziplinäres Zentrum für Wissenschaftliches Rechnen

2. September 2013



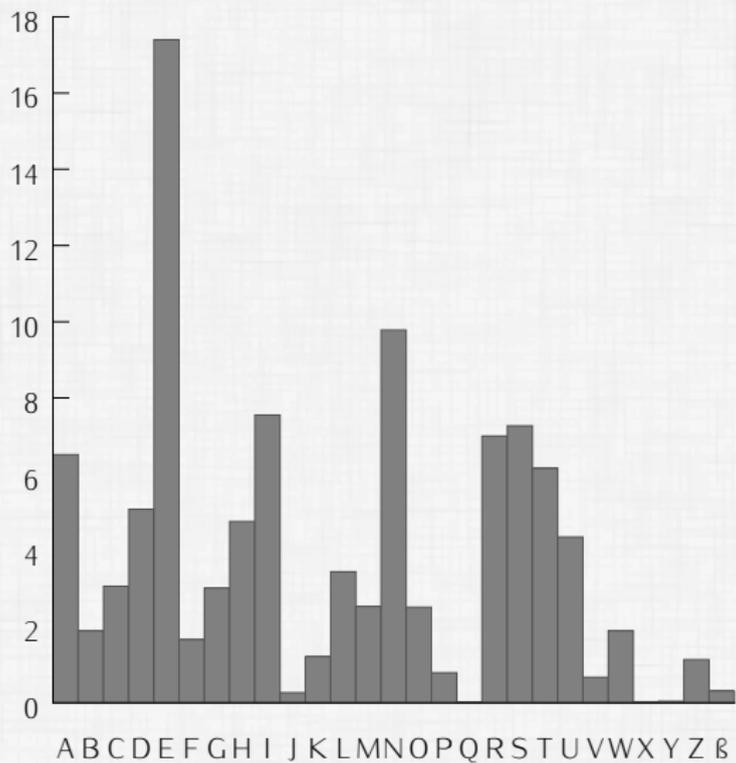
Kryptographie

1. kryptós (verborgen, geheim)
2. gráphein (schreiben)

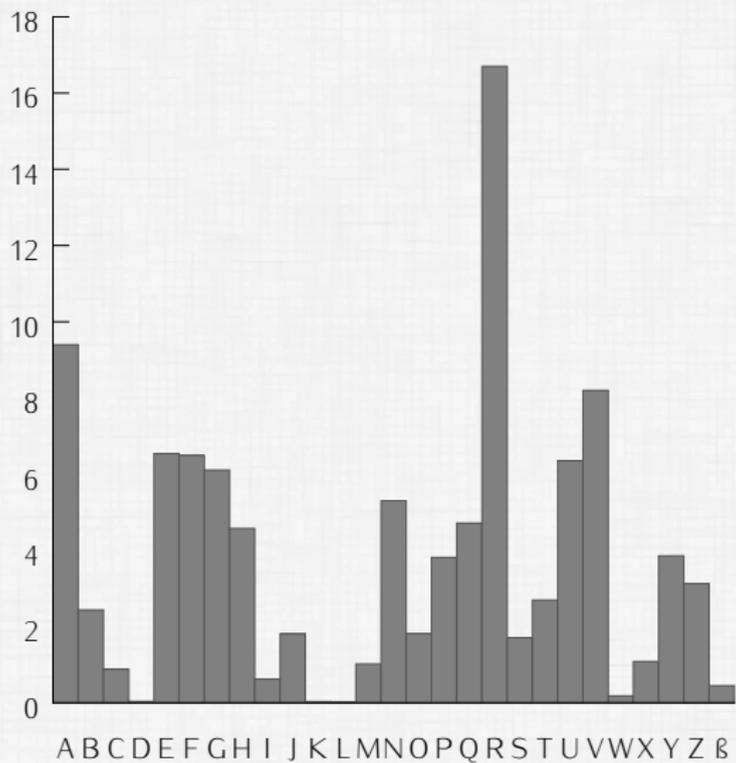


...wenn etwas Geheimes zu überbringen war, schrieb er in Zeichen, das heißt, er ordnete die Buchstaben so, dass kein Wort gelesen werden konnte: Um diese zu lesen, tauscht man den vierten Buchstaben, also D, für A aus und ebenso mit den restlichen.

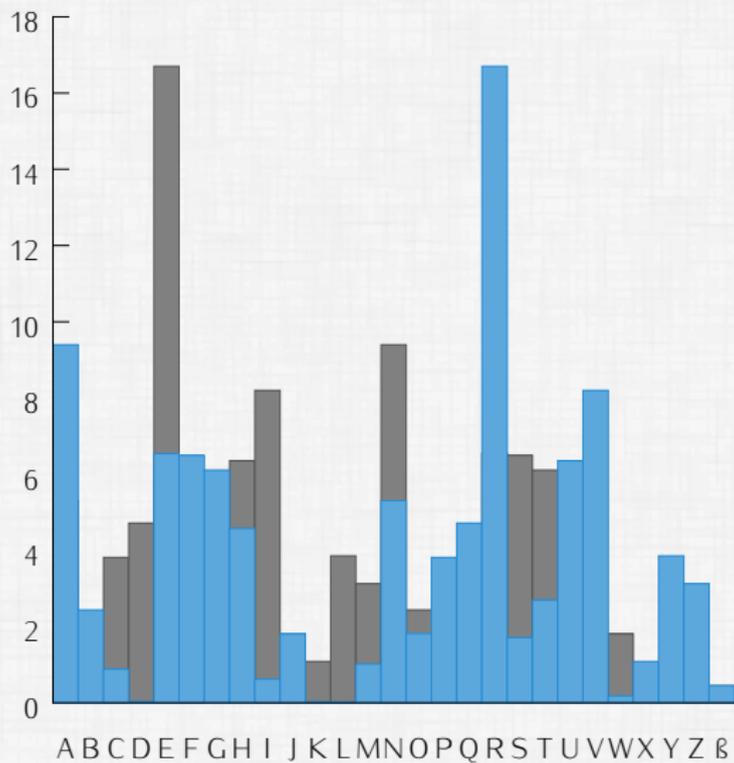
Ist das sicher?



Häufigkeiten deutscher Buchstaben

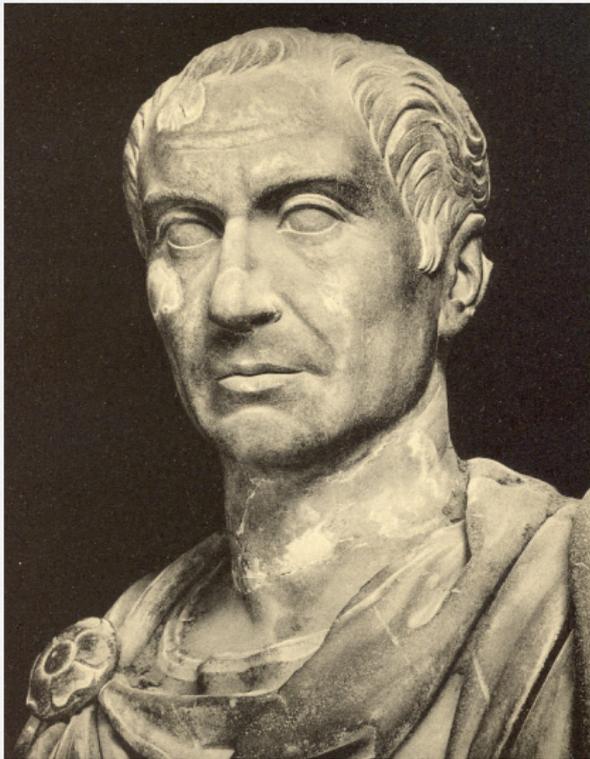


Häufigkeiten in einer verschlüsselten Fassung von "Faust"



Vergleich beider Häufigkeitstabellen

$$E = R$$



Können wir es wenigstens schön
beschreiben?

Exkurs: Modulo-Rechnung

Die Modulo-Operation, kurz **mod**, beschreibt den *Rest* bei ganzzahliger Division.



- $14 \bmod 12 = 2$
- $16 \bmod 12 = 4$
- $21 \bmod 12 = 9$

Mathematische Beschreibung der Caesar-Chiffre

Sei k der Schlüssel, d.h. die Anzahl der Stellen, um die verschoben wird. Wir setzen $A = 0, B = 1, \dots, Z = 25$.

$$E_k(x) = (x + k) \bmod 26$$

$$D_k(x) = (x - k) \bmod 26$$

Einige Maschinen



Problem: Die Schlüsselverteilung!

The image shows an open book with two pages of a key schedule. The pages are titled "Teil A". Each page contains a table with columns for "Kerngruppe", "Sprache", and "Schlüssel". The tables list various key groups and their corresponding letters in a specific order.

No.	Kerngruppe	Sprache	Schlüssel	No.	Kerngruppe	Sprache	Schlüssel													
1	DD	ARCK	51	Q	B	FOID	101	R	A	J	FOER	111	M	V	X	ARE				
2	LV	BRML	52	L	T	ORHO	102	H	O	V	HOVF	112	H	O	V	HOVD				
3	FP	BOIF	53	P	E	NOGL	103	S	F	H	HOFT	113	H	O	V	HOFT				
4	FO	DBER	54	A	U	COHF	104	O	N	L	EOOH	114	Q	H	H	HOFT				
5	RF	ERHO	55	W	R	AFPE	105	W	H	H	HOFT	115	V	L	H	HOFT				
6	U	FCOR	56	W	W	LEEL	106	K	H	I	DOOR	116	V	L	H	HOFT				
7	FO	ODHR	57	A	A	CAHE	107	E	H	I	DOOR	117	K	S	T	PFJ				
8	U	FCOR	58	D	R	CAHT	108	K	H	I	DOOR	118	W	V	L	ARF				
9	F	PHZD	59	P	O	V	FOOO	109	D	W	H	HOFT	119	A	X	R	HOPO			
10	M	HRIT	60	H	R	ITTO	110	H	R	I	DOOR	120	H	R	I	DOOR				
11	K	OZ	TFXK	61	O	K	Z	SNOP	111	K	W	H	HOFT	121	H	R	I	DOOR		
12	N	VK	ECUZ	62	J	V	NOGL	112	H	K	W	HOFT	122	H	R	I	DOOR			
13	H	EK	TIER	63	N	D	AHIE	113	Q	P	Z	LFJF	123	H	R	I	DOOR			
14	V	J	IFPK	64	Q	W	MOGL	114	Q	J	A	WHIE	124	H	R	I	DOOR			
15	H	H	IFNK	65	H	H	IFNK	115	Q	J	A	WHIE	125	H	R	I	DOOR			
16	B	L	TOFR	66	T	S	U	FOOX	116	T	Y	Q	NFXK	126	H	R	I	DOOR		
17	H	O	J	TYHA	67	K	E	T	ERWF	117	W	H	HOFT	127	H	R	I	DOOR		
18	K	D	F	HOON	68	E	S	T	PBL	118	W	D	H	FOOD	128	H	R	I	DOOR	
19	L	O	T	PFTT	69	A	V	X	EMGR	119	H	U	K	AGKT	129	H	R	I	DOOR	
20	O	M	A	TFP	70	K	V	T	XGOS	120	V	F	H	HOFT	130	H	R	I	DOOR	
21	P	Q	S	AAML	71	G	M	V	TCDE	121	H	R	X	FRS	131	H	R	I	DOOR	
22	T	O	P	PCFR	72	J	N	X	DKL	122	O	R	S	HOFT	132	H	R	I	DOOR	
23	E	W	H	OCM	73	L	W	E	SGX	123	J	K	E	TSEN	133	H	R	I	DOOR	
24	C	P	T	EHFX	74	N	X	Q	IOVF	124	L	Q	N	FRFK	134	H	R	I	DOOR	
25	K	V	F	UGCE	75	H	H	I	LEKO	125	N	D	S	HOFT	135	H	R	I	DOOR	
26	G	E	N	AKSE	76	H	V	O	LEVF	126	H	P	T	HOFT	136	H	R	I	DOOR	
27	M	O	R	ROMS	77	T	J	D	DDLE	127	H	R	F	HOFT	137	H	R	I	DOOR	
28	N	L	E	OKEH	78	U	D	E	NOHD	128	V	C	E	GRS	138	H	R	I	DOOR	
29	B	E	R	ALLIE	79	F	T	ONHD	129	V	W	X	WERN	139	H	R	I	DOOR		
30	X	X	W	NOVS	80	V	O	C	SKHI	130	A	B	T	OTFJ	140	H	R	I	DOOR	
31	X	X	W	NOZL	81	H	E	R	EWOK	131	H	K	R	FFLE	141	H	R	I	DOOR	
32	O	A	E	XFO	82	A	B	E	XDFV	132	H	K	R	FFLE	142	H	R	I	DOOR	
33	H	O	T	YFJA	83	H	O	T	YFJA	133	G	L	A	TEN	143	H	R	I	DOOR	
34	H	Y	H	TYHT	84	H	T	C	TIME	134	H	L	E	VOGZ	144	H	R	I	DOOR	
35	L	A	N	LPMK	85	K	P	R	OKFU	135	H	C	O	NAFE	145	H	R	I	DOOR	
36	O	C	S	CFBY	86	M	Y	M	OHFF	136	L	E	C	FRNO	146	H	R	I	DOOR	
37	T	C	Z	ANHE	87	H	O	K	YNOG	137	H	G	V	FFJF	147	H	R	I	DOOR	
38	U	V	S	WLO	88	O	Q	O	LEVT	138	H	Q	W	KYFF	148	H	R	I	DOOR	
39	Y	T	E	TEPK	89	H	K	G	UITY	139	H	H	B	IRDO	149	H	R	I	DOOR	
40	A	L	N	WLEI	90	H	X	V	SIEM	140	U	T	O	FTGO	150	H	R	I	DOOR	
41	C	H	M	ZERF	91	X	H	C	OPGZ	141	W	O	K	ADFE	151	H	R	I	DOOR	
42	S	H	N	TEWH	92	H	T	Y	GOZ	142	H	M	U	SMFP	152	H	R	I	DOOR	
43	H	O	K	ROGT	93	H	A	F	H	HOFF	143	H	O	G	YVBE	153	H	R	I	DOOR
44	M	D	E	STOG	94	X	E	D	SDLE	144	A	G	J	EUFC	154	H	R	I	DOOR	

- Zu wenige Möglichkeiten zur Verschlüsselung
- Verfahren sind *symmetrisch*

“Der Feind kennt das System”



Ron Rivest



Adi Shamir



Leonard Adleman



Exkurs: Primzahlen

Eine Primzahl ist eine natürliche Zahl größer **1**, die nur durch sich selbst und **1** teilbar ist.

Exkurs: Primzahlen

Eine Primzahl ist eine natürliche Zahl größer 1, die nur durch sich selbst und 1 teilbar ist.

Beispiele: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449, 457, 461, 463, 467, 479, 487, 491, 499, 503, 509, 521, 523, 541, 547, 557, 563, 569, 571, 577, 587, 593, 599, 601, 607, 613, 617, 619, 631, 641, 643, 647, 653, 659, 661, 673, 677, 683, 691, 701, 709, 719, 727, 733, 739, 743, 751, 757, 761, 769, 773, 787, 797, 809, 811, 821, 823, 827, 829, 839, 853 . . .

Exkurs: Primzahlen

Eine Primzahl ist eine natürliche Zahl größer 1, die nur durch sich selbst und 1 teilbar ist.

Beispiele: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449, 457, 461, 463, 467, 479, 487, 491, 499, 503, 509, 521, 523, 541, 547, 557, 563, 569, 571, 577, 587, 593, 599, 601, 607, 613, 617, 619, 631, 641, 643, 647, 653, 659, 661, 673, 677, 683, 691, 701, 709, 719, 727, 733, 739, 743, 751, 757, 761, 769, 773, 787, 797, 809, 811, 821, 823, 827, 829, 839, 853 . . .

Jede natürliche Zahl lässt sich als (im Wesentlichen eindeutiges) Produkt von Primzahlen darstellen.

Die Grundlage von RSA

Die Grundlage von RSA

Diese Richtung ist einfach:

$$2161 \cdot 2179 = 4708819$$

$$7883 \cdot 9199 = 72515717$$

$$48673 \cdot 26711 = 1300104503$$

Die Grundlage von RSA

Diese Richtung ist einfach:

$$2161 \cdot 2179 = 4708819$$

$$7883 \cdot 9199 = 72515717$$

$$48673 \cdot 26711 = 1300104503$$

Diese Richtung nicht so sehr:

$$50621 = 223 \cdot 227$$

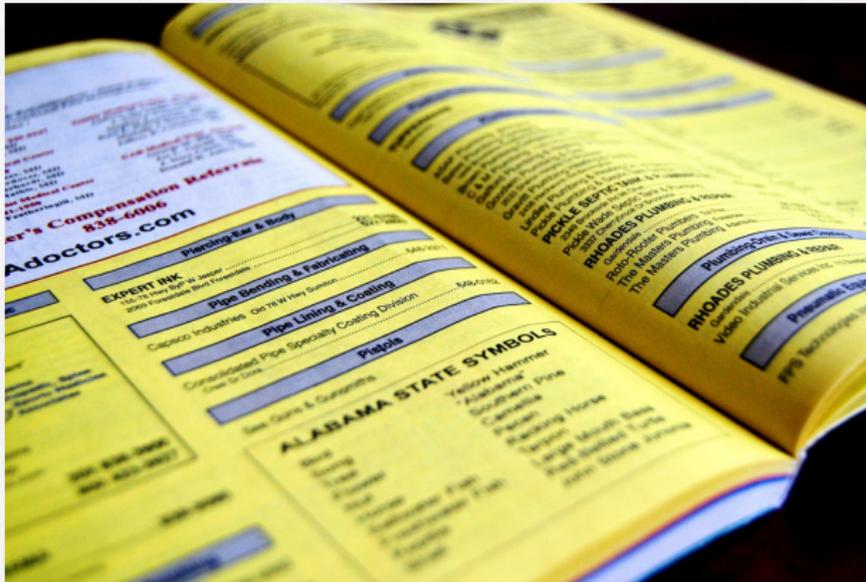
$$11840417 = 3433 \cdot 3449$$

$$2985959 = 1723 \cdot 1733$$

Multiplikation ist leichter als
Faktorisierung

Mathematisch formuliert

Die Multiplikation von zwei Primzahlen p und q ist eine *Einwegfunktion*: Den Funktionswert zu berechnen ist einfach (Multiplikation), aber die zu einem Funktionswert gehörenden Primzahlen zu finden, ist schwer (Faktorisierung).

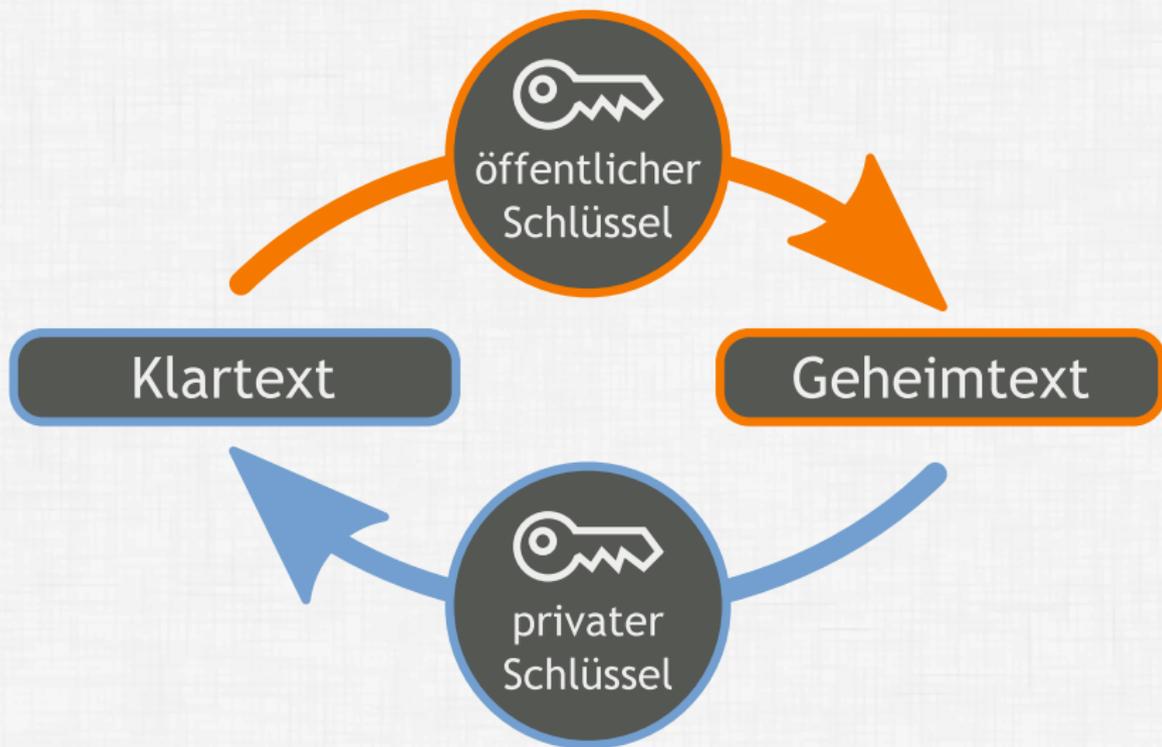


Auch eine Art Einwegfunktion

RSA

- Öffentlicher Schlüssel (an alle verteilen)
- Privater Schlüssel (geheim halten)
- Nachrichten werden mit dem öffentlichen Schlüssel *verschlüsselt* und mit dem privaten Schlüssel *entschlüsselt*.





Algorithmus: Schlüsselerzeugung

- Wähle zwei unterschiedliche Primzahlen p und q

Algorithmus: Schlüsselerzeugung

- Wähle zwei unterschiedliche Primzahlen p und q
- Berechne $n = p \cdot q$ und $\phi = (p - 1) \cdot (q - 1)$

Algorithmus: Schlüsselerzeugung

- Wähle zwei unterschiedliche Primzahlen p und q
- Berechne $n = p \cdot q$ und $\phi = (p - 1) \cdot (q - 1)$
- Wähle $1 < e < \phi$ teilerfremd zu ϕ

Algorithmus: Schlüsselerzeugung

- Wähle zwei unterschiedliche Primzahlen p und q
- Berechne $n = p \cdot q$ und $\phi = (p - 1) \cdot (q - 1)$
- Wähle $1 < e < \phi$ teilerfremd zu ϕ
- Wähle d , sodass $ed \bmod \phi = 1$ (das geht nur, wenn e teilerfremd zu ϕ)

Algorithmus: Schlüsselerzeugung

- Wähle zwei unterschiedliche Primzahlen p und q
- Berechne $n = p \cdot q$ und $\phi = (p - 1) \cdot (q - 1)$
- Wähle $1 < e < \phi$ teilerfremd zu ϕ
- Wähle d , sodass $ed \bmod \phi = 1$ (das geht nur, wenn e teilerfremd zu ϕ)
- (e, n) ist dann der öffentliche Schlüssel

Algorithmus: Schlüsselerzeugung

- Wähle zwei unterschiedliche Primzahlen p und q
- Berechne $n = p \cdot q$ und $\phi = (p - 1) \cdot (q - 1)$
- Wähle $1 < e < \phi$ teilerfremd zu ϕ
- Wähle d , sodass $ed \bmod \phi = 1$ (das geht nur, wenn e teilerfremd zu ϕ)
- (e, n) ist dann der öffentliche Schlüssel
- (d, n) ist dann der private Schlüssel

Algorithmus: Schlüsselerzeugung

- Wähle zwei unterschiedliche Primzahlen p und q
- Berechne $n = p \cdot q$ und $\phi = (p - 1) \cdot (q - 1)$
- Wähle $1 < e < \phi$ teilerfremd zu ϕ
- Wähle d , sodass $ed \bmod \phi = 1$ (das geht nur, wenn e teilerfremd zu ϕ)
- (e, n) ist dann der öffentliche Schlüssel
- (d, n) ist dann der private Schlüssel
- Länge von n ist typischerweise ≈ 300 Dezimalstellen

Algorithmus: Verschlüsselung

Sei $0 \leq m < n$ die zu verschlüsselnde Nachricht. m muss teilerfremd zu n sein. Berechne dann:

$$c = m^e \bmod n$$

Algorithmus: Entschlüsselung

Sei c die empfangene Nachricht. Berechne dann:

$$c^d \bmod n = m$$



Angela



Barack



Wladimir

Beispiel

Barack erzeugt seine Schlüssel

- Er wählt $p = 7$, $q = 11$



Beispiel

Barack erzeugt seine Schlüssel

- Er wählt $p = 7, q = 11$
- Barack berechnet $n = p \cdot q = 77$



Beispiel

Barack erzeugt seine Schlüssel

- Er wählt $p = 7, q = 11$
- Barack berechnet $n = p \cdot q = 77$
- Und $\phi = 6 \cdot 10 = 60$



Beispiel

Barack erzeugt seine Schlüssel

- Er wählt $p = 7, q = 11$
- Barack berechnet $n = p \cdot q = 77$
- Und $\phi = 6 \cdot 10 = 60$
- Er wählt $e = 23$



Beispiel

Barack erzeugt seine Schlüssel

- Er wählt $p = 7, q = 11$
- Barack berechnet $n = p \cdot q = 77$
- Und $\phi = 6 \cdot 10 = 60$
- Er wählt $e = 23$
- Und findet $d = 47$



Beispiel

Barack erzeugt seine Schlüssel

- Er wählt $p = 7, q = 11$
- Barack berechnet $n = p \cdot q = 77$
- Und $\phi = 6 \cdot 10 = 60$
- Er wählt $e = 23$
- Und findet $d = 47$
- Er schickt $(23, 77)$ an Angela



Beispiel

Barack erzeugt seine Schlüssel

- Er wählt $p = 7$, $q = 11$
- Barack berechnet $n = p \cdot q = 77$
- Und $\phi = 6 \cdot 10 = 60$
- Er wählt $e = 23$
- Und findet $d = 47$
- Er schickt $(23, 77)$ an Angela
- Er schließt $(47, 77)$ in den Safe ein



Beispiel

Angela verschickt ihre Nachricht

- Angela will $m = 48$ (Anzahl Gäste für Putins Party) senden



Beispiel

Angela verschickt ihre Nachricht

- Angela will $m = 48$ (Anzahl Gäste für Putins Party) senden
- Baracks öffentlicher Schlüssel ist $(23, 77)$



Beispiel

Angela verschickt ihre Nachricht

- Angela will $m = 48$ (Anzahl Gäste für Putins Party) senden
- Baracks öffentlicher Schlüssel ist $(23, 77)$
- Sie berechnet $48^{23} \bmod 77 = 20$



Beispiel

Barack entschlüsselt Angelas Nachricht

- Barack empfängt $c = 20$



Beispiel

Barack entschlüsselt Angelas Nachricht

- Barack empfängt $c = 20$
- Sein privater Schlüssel ist $(47, 77)$



Beispiel

Barack entschlüsselt Angelas Nachricht

- Barack empfängt $c = 20$
- Sein privater Schlüssel ist $(47, 77)$
- Er berechnet $20^{47} \bmod 77 = 48$



Warum funktioniert das überhaupt?

Exkurs: Satz von Euler-Fermat

Für zwei Primzahlen p und q und jede natürliche Zahl m , die teilerfremd zu pq ist, gilt:

$$m^{(p-1)(q-1)} \bmod pq = 1$$

(Der Fundamentalsatz von RSA)

$$\begin{aligned}c^d \bmod n &= m^{ed} \bmod n \text{ (nach Definition von } c\text{)} \\&= m^{1+k\phi} \bmod n \text{ (denn } ed \bmod \phi = 1\text{)} \\&= m^{1+k(p-1)(q-1)} \bmod n \text{ (nach Definition von } \phi\text{)} \\&= m \cdot m^{k \cdot (p-1)(q-1)} \bmod n \text{ (ausmultipliziert)} \\&= m \cdot m^{(p-1)(q-1)^k} \bmod n \text{ (Potenzierung geändert)} \\&= m \text{ (nach Euler-Fermat)}\end{aligned}$$

Warum ist das sicher?

- Wladimir kennt (e, n)



Warum ist das sicher?

- Wladimir kennt (e, n)
- Er braucht aber d zur Entschlüsselung, damit er $c^d \bmod n = m$ berechnen kann!



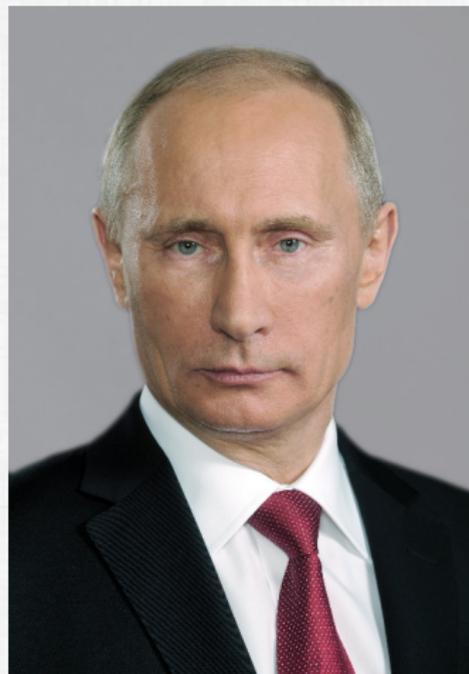
Warum ist das sicher?

- Wladimir kennt (e, n)
- Er braucht aber d zur Entschlüsselung, damit er $c^d \bmod n = m$ berechnen kann!
- e und d stehen durch $ed \bmod \phi = 1$ in Verbindung



Warum ist das sicher?

- Wladimir kennt (e, n)
- Er braucht aber d zur Entschlüsselung, damit er $c^d \bmod n = m$ berechnen kann!
- e und d stehen durch $ed \bmod \phi = 1$ in Verbindung
- Wladimir braucht also $\phi = (p - 1) \cdot (q - 1)$



Warum ist das sicher?

- Wladimir kennt (e, n)
- Er braucht aber d zur Entschlüsselung, damit er $c^d \bmod n = m$ berechnen kann!
- e und d stehen durch $ed \bmod \phi = 1$ in Verbindung
- Wladimir braucht also $\phi = (p - 1) \cdot (q - 1)$
- Dazu braucht er aber p und q



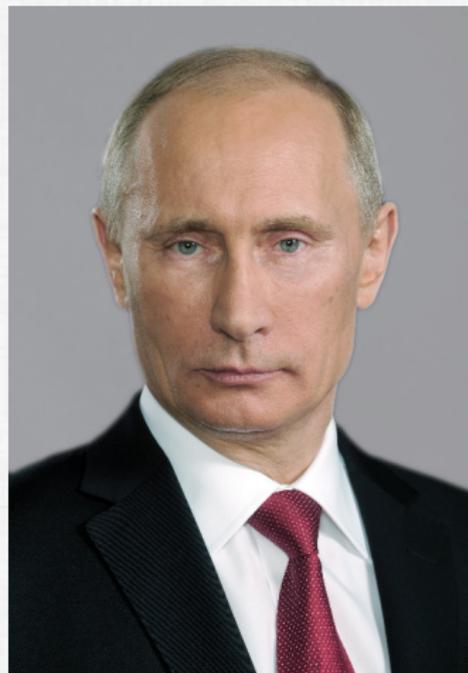
Warum ist das sicher?

- Wladimir kennt (e, n)
- Er braucht aber d zur Entschlüsselung, damit er $c^d \bmod n = m$ berechnen kann!
- e und d stehen durch $ed \bmod \phi = 1$ in Verbindung
- Wladimir braucht also $\phi = (p - 1) \cdot (q - 1)$
- Dazu braucht er aber p und q
- Er muss also $n = p \cdot q$ faktorisieren!



Warum ist das sicher?

- Wladimir kennt (e, n)
- Er braucht aber d zur Entschlüsselung, damit er $c^d \bmod n = m$ berechnen kann!
- e und d stehen durch $ed \bmod \phi = 1$ in Verbindung
- Wladimir braucht also $\phi = (p - 1) \cdot (q - 1)$
- Dazu braucht er aber p und q
- Er muss also $n = p \cdot q$ faktorisieren!
- Für große Zahlen braucht das lange (länger als das Universum alt ist?)



Warum ist das sicher?

- Wladimir kennt (e, n)
- Er braucht aber d zur Entschlüsselung, damit er $c^d \bmod n = m$ berechnen kann!
- e und d stehen durch $ed \bmod \phi = 1$ in Verbindung
- Wladimir braucht also $\phi = (p - 1) \cdot (q - 1)$
- Dazu braucht er aber p und q
- Er muss also $n = p \cdot q$ faktorisieren!
- Für große Zahlen braucht das lange (länger als das Universum alt?)
- Barack kann einfach zusätzliches Bit benutzen, was den Aufwand verdoppelt!



Wie geht es weiter?

Literatur:

- Simon Singh: *Codes – Die Kunst der Verschlüsselung*
- Bruce Schneier: *Angewandte Kryptographie*
- Neal Stephenson: *Cryptonomicon*

Software:

- *HTTPS Everywhere*
- *Off-the-record messaging*
- *The GNU Privacy Guard*
- *RedPhone*

