

Konzepte von Betriebssystem- Komponenten

PGP

Verwendete Algorithmen,
PGP aus Benutzersicht,
Schlüsselzertifizierung

von Alexander Dreweke

PGP

1. Allgemeines

- symmetrische Algorithmen
- asymmetrische Algorithmen
- Hash-Funktionen

2. Ver- und Entschlüsseln

- Verschlüsseln
- ASCII-Rüstung
- Entschlüsseln
- Probleme

PGP

3. Signieren und Verifizieren

- Signieren
- Verifizieren
- Probleme

4. Zertifikate & Web-of-Trust

- X.509-Zertifikate
- PGP-Zertifikate
- Vertrauen & Gültigkeit
- Probleme (Sign & Encrypt bzw. Encrypt & Sign)

verwendete Algorithmen

symmetrische Algorithmen

PGP

- AES
- CAST
- IDEA
- TWOFISH
- 3DES

verwendete Algorithmen

	symmetrische Algorithmen	asymmetrische Algorithmen
PGP	<ul style="list-style-type: none">- AES- CAST- IDEA- TWOFISH- 3DES	<ul style="list-style-type: none">- RSA (bis 3072)- DH/DSS (bis 4096)

verwendete Algorithmen

	symmetrische Algorithmen	asymmetrische Algorithmen	Hash-Funktionen
PGP	<ul style="list-style-type: none">- AES- CAST- IDEA- TWOFISH- 3DES	<ul style="list-style-type: none">- RSA (bis 3072)- DH/DSS (bis 4096)	<ul style="list-style-type: none">- SHA1

verwendete Algorithmen

	symmetrische Algorithmen	asymmetrische Algorithmen	Hash-Funktionen
PGP	<ul style="list-style-type: none">- AES- CAST- IDEA- TWOFISH- 3DES	<ul style="list-style-type: none">- RSA (bis 3072)- DH/DSS (bis 4096)	<ul style="list-style-type: none">- SHA1
GPG	<ul style="list-style-type: none">- BLOWFISH- CAST5- TWOFISH- RIJNDAEL (= AES)- 3DES	<ul style="list-style-type: none">- DAS- ELG/ ELG-E (=DH)- RSA/ RSA-E/ RSA-S	<ul style="list-style-type: none">- MD5- RIPEMD160- SHA1

Verschlüsselung

A

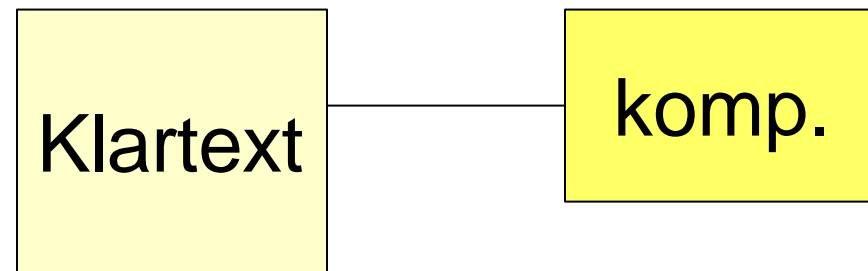
B

Klartext

Verschlüsselung

A

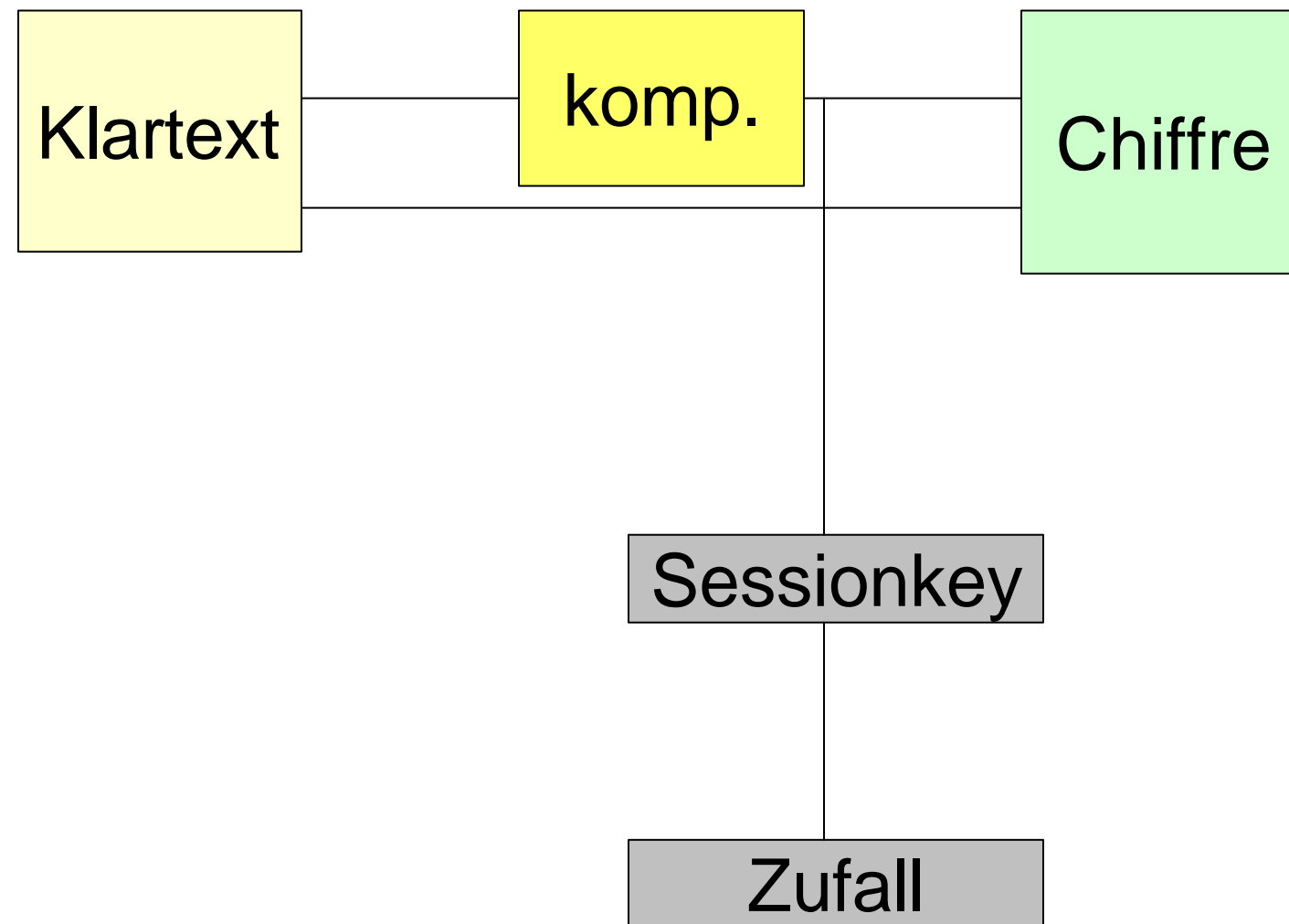
B



Verschlüsselung

A

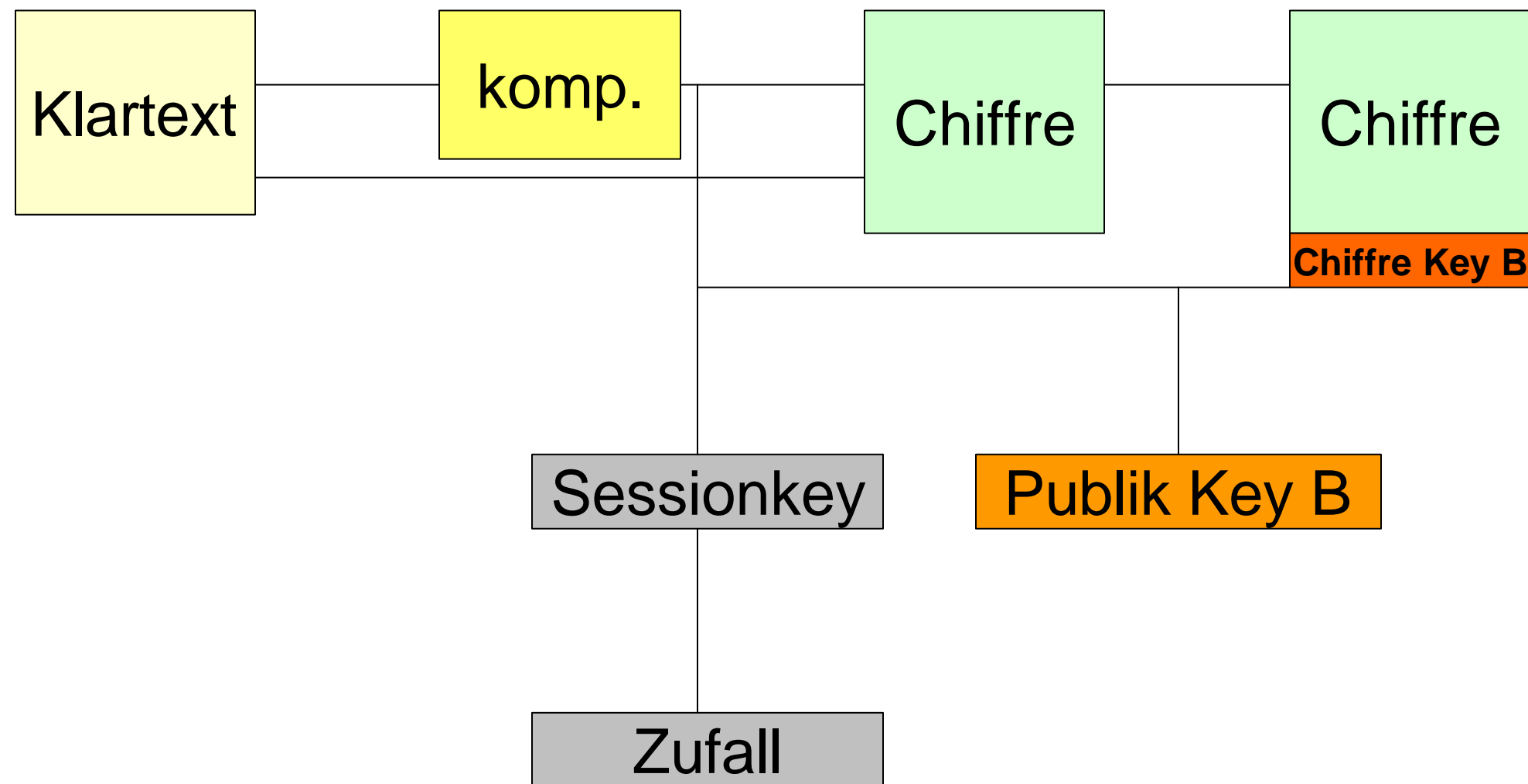
B



Verschlüsselung

A

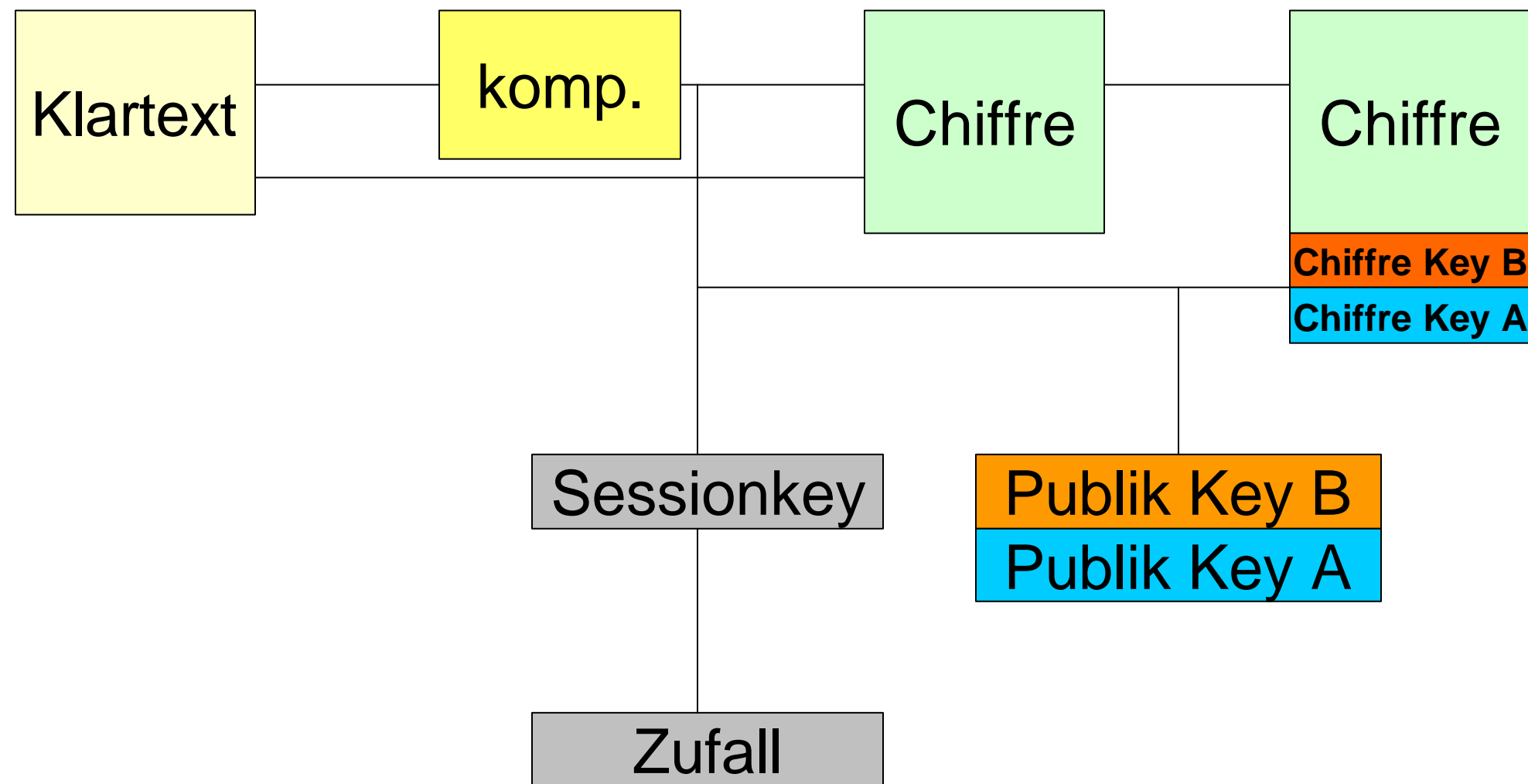
B



Verschlüsselung

A

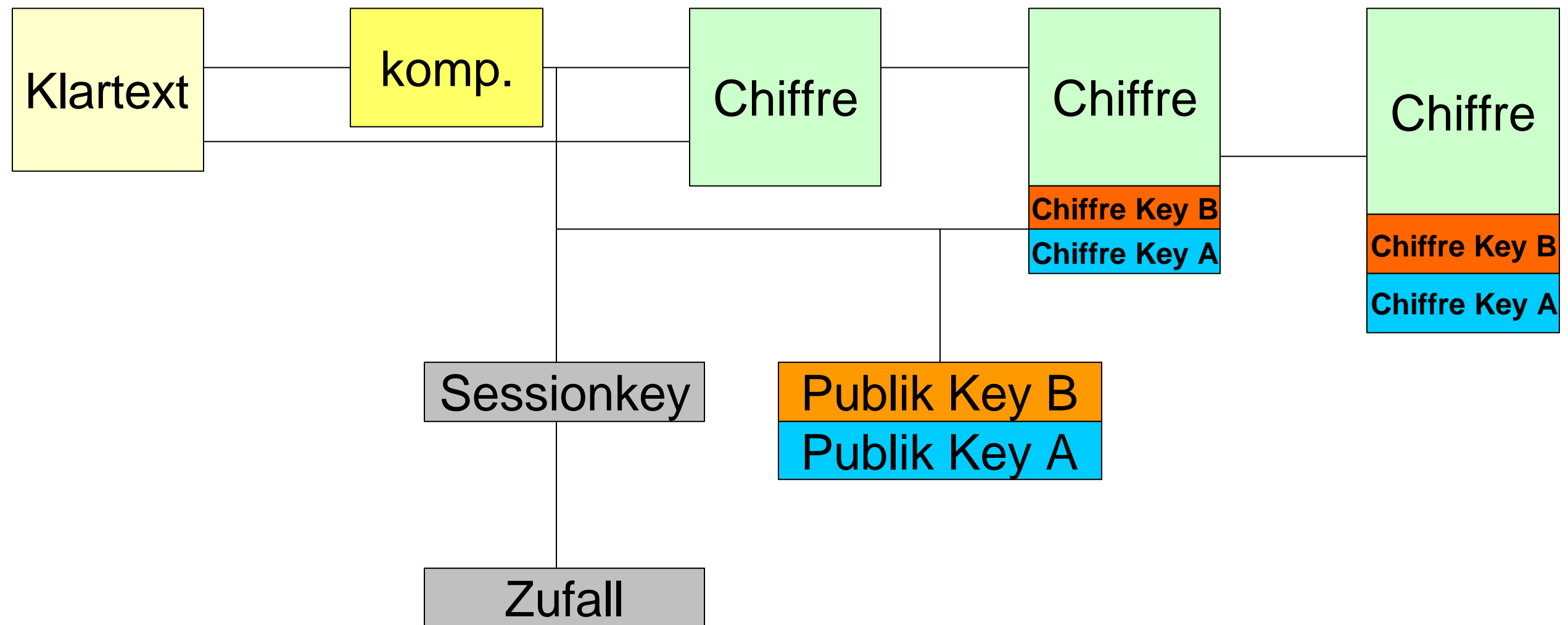
B



Verschlüsselung

A

B



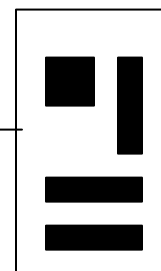
ohne ASCII - Rüstung

A ----- B

Server

Nachricht

```
.....01  
01011101  
10011010  
10110110  
01.....
```



```
.....01  
01011101  
00011010  
00110110  
01.....
```

Nachricht

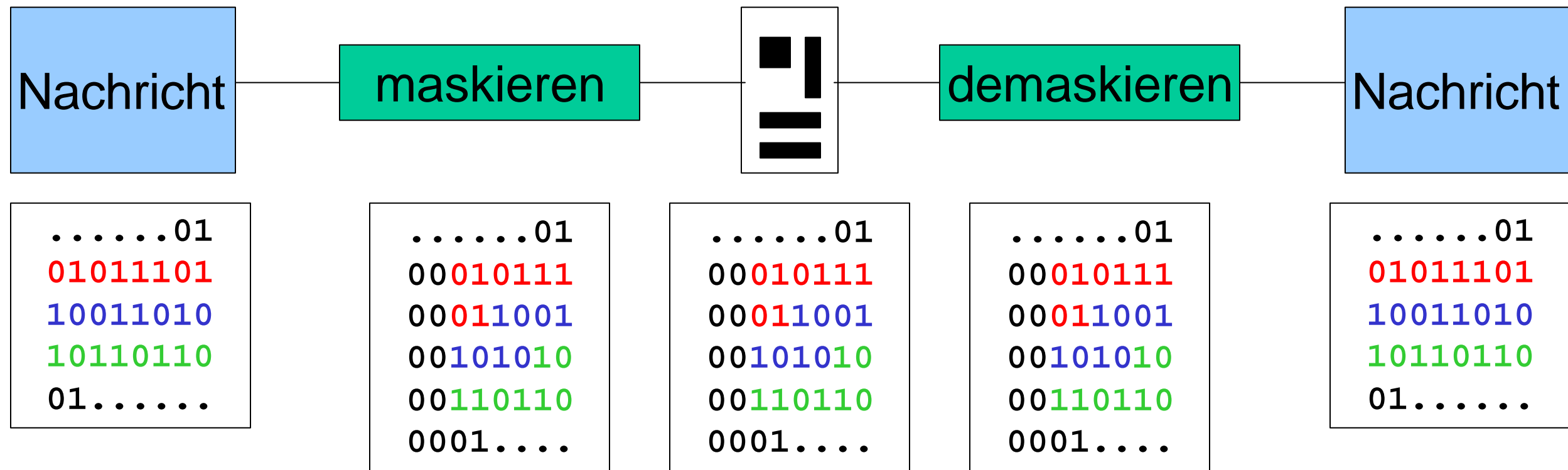
```
.....01  
01011101  
00011010  
00110110  
01.....
```

mit ASCII - Rüstung

A

B

Server



Entschlüsselung

A

B

Chiffre

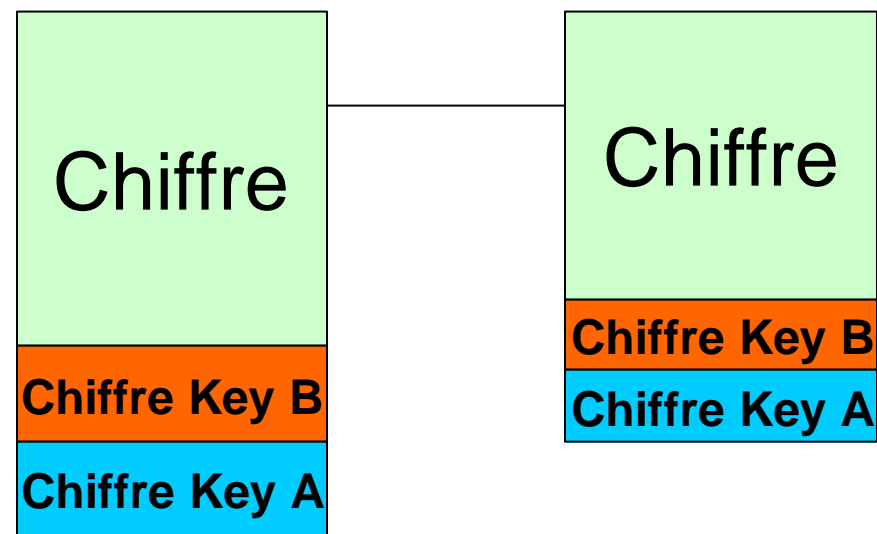
Chiffre Key B

Chiffre Key A

Entschlüsselung

A

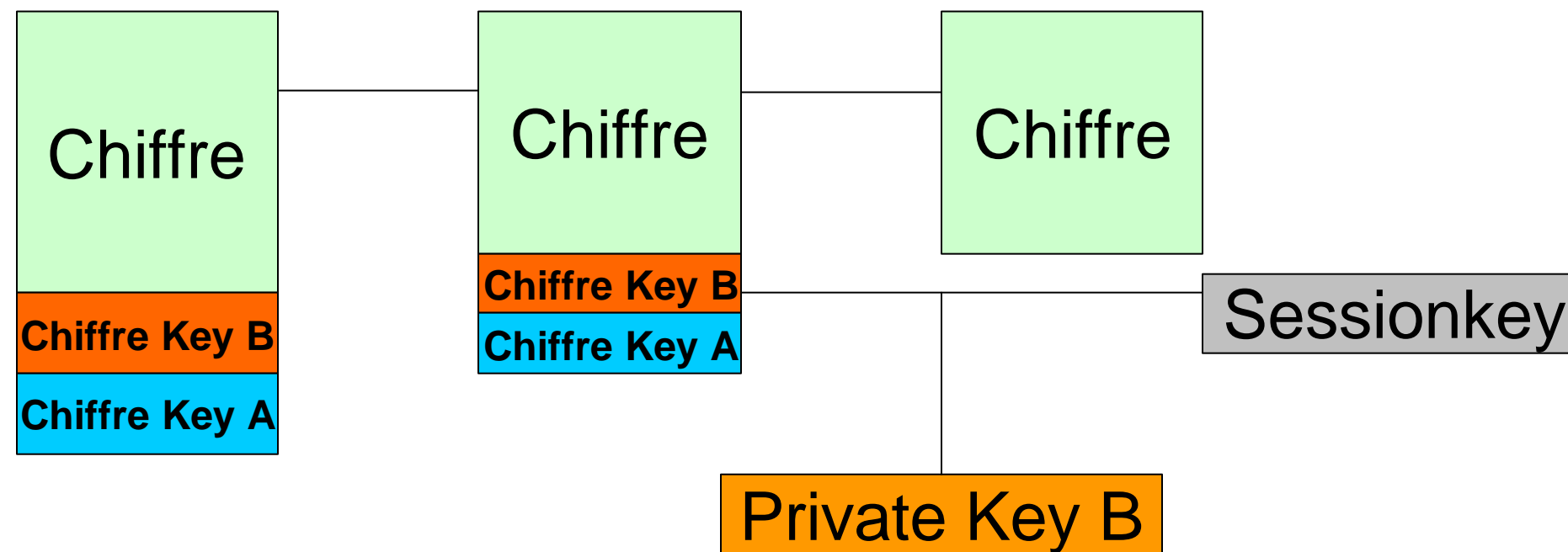
B



Entschlüsselung

A

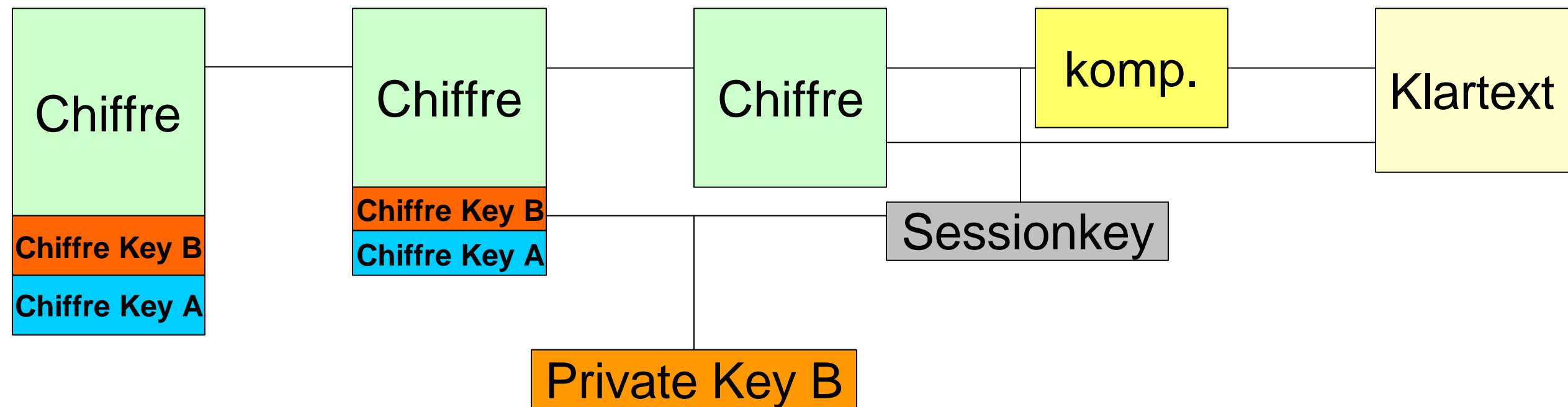
B



Entschlüsselung

A

B



Problem

B weiß nur, dass die Nachricht von jemandem stammt, der B's **Public-Key** hat, dieser soll aber gerade für **jedermann zugänglich** sein

Signieren

A

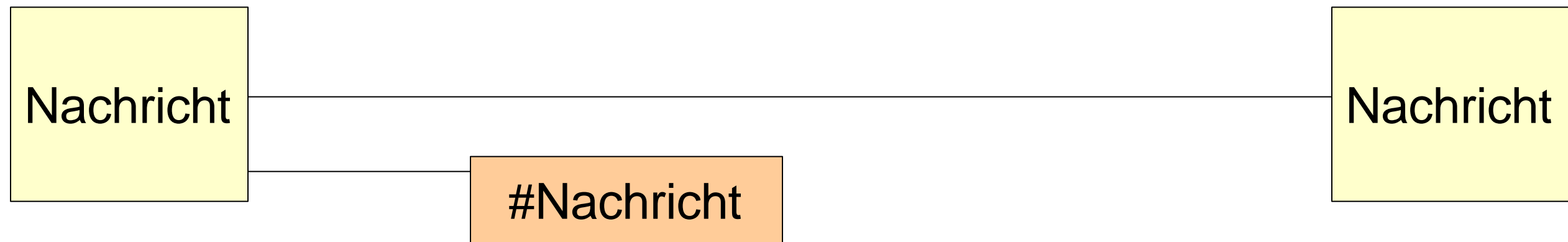
B

Nachricht

Signieren

A

B

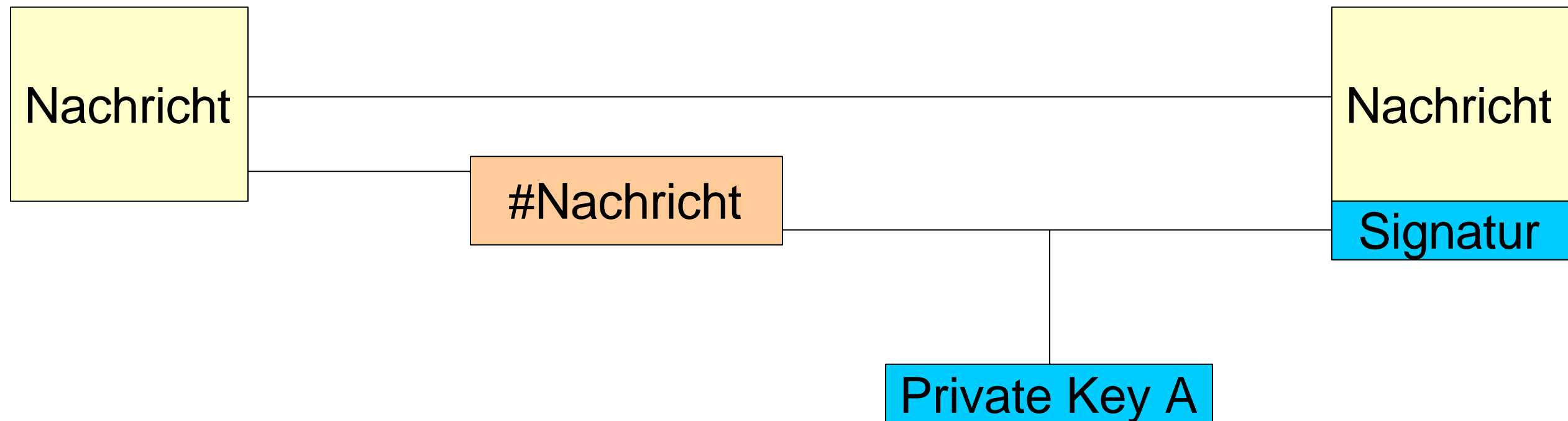


#: Hash-Wert

Signieren

A

B



#: Hash-Wert

Verifizieren

A

B

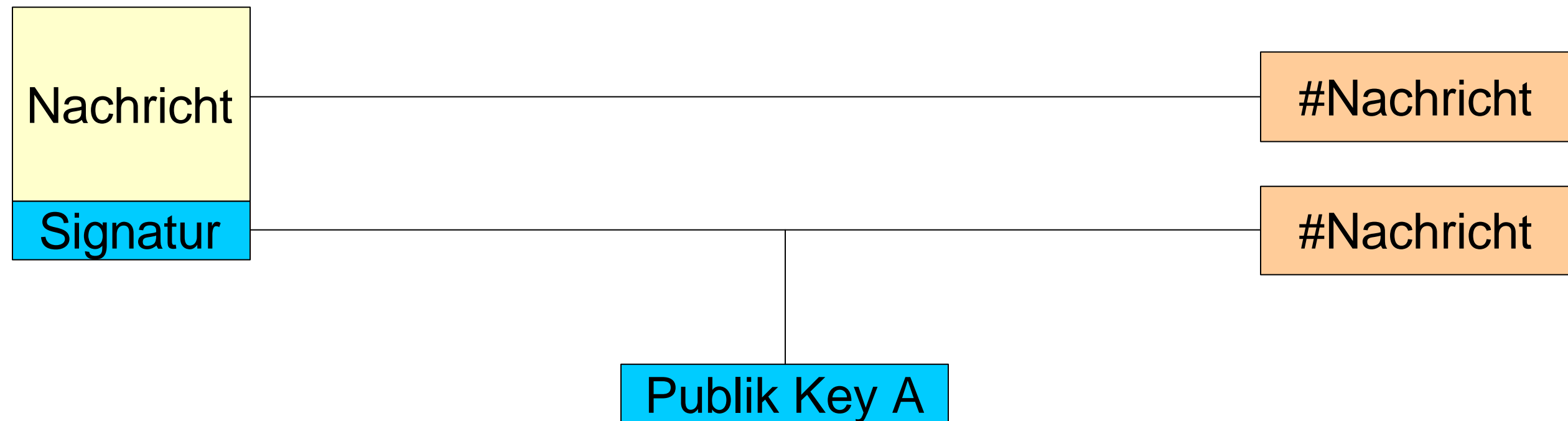
Nachricht

Signatur

Verifizieren

A

B



#: Hash-Wert

Problem

B weiß nur, dass die Nachricht von jemandem stammt, der den passenden **Private-Key** besitzt (damit ist nicht sichergestellt, dass die Person wirklich die ist für die sie sich ausgibt)

X.509-Zertifikate

- X.509-Versionnummer
- öffentliche Schlüssel des Zertifikatsinhabers
- Seriennummer des Zertifikats
- eindeutige Kennung des Zertifikatsinhabers
- Gültigkeitsdauer des Zertifikats
- eindeutiger Name des Zertifikatsausstellers
- digitale Unterschrift des Ausstellers
- Kennung des Unterschriftenalgorithmus

Nachteil:

- Ausstellung eines X.509-Zertifikats muss bei einer Zertifizierungsinstanz angefordert werden (zumeist kostenpflichtig)
- Nur eine einzige digitale Unterschrift zur Bestätigung der Schlüsselgültigkeit.

PGP-Zertifikate

- PGP-Versionnummer
- öffentlicher Schlüssel des Zertifikatsinhabers
- Daten des Zertifikatsinhabers
- digitale Unterschrift des Zertifikatseigentümers
- Gültigkeitsdauer des Zertifikats
- bevorzugte symmetrische Verschlüsselungsalgorithmus für die Schlüssel

Vertrauen & Gültigkeit

Schlüssel	Vertrauen	Gültigkeit	signiert von
own			-
A			own
B			own
C	?		A
D	?		B
E	?		A & B

Vertrauen & Gültigkeit

Schlüssel	Vertrauen	Gültigkeit	signiert von
own			-
A			own
B			own
C	?		A
D	?		B
E	?		A & B

Vertrauen & Gültigkeit

Schlüssel	Vertrauen	Gültigkeit	signiert von
own			-
A			own
B			own
C	?		A
D	?		B
E	?		A & B

Vertrauen & Gültigkeit

Schlüssel	Vertrauen	Gültigkeit	signiert von
own			-
A			own
B			own
C	?		A
D	?		B
E	?		A & B

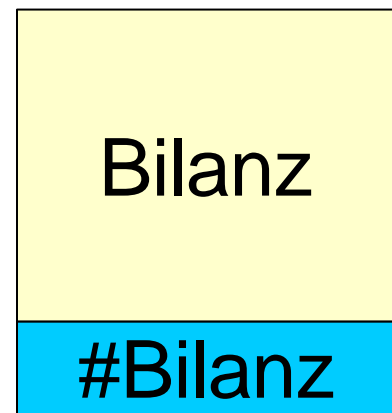
Sign & Encrypt Schwäche

A ————— B

Bilanz

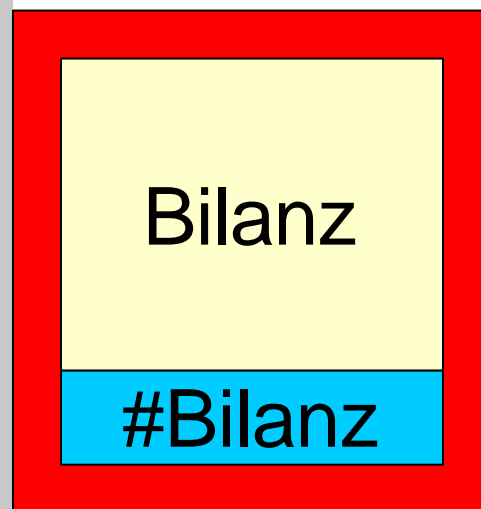
Sign & Encrypt Schwäche

A ————— B



Sign & Encrypt Schwäche

A ————— B



Sign & Encrypt Schwäche

A ————— B



Sign & Encrypt Schwäche

A

B

C

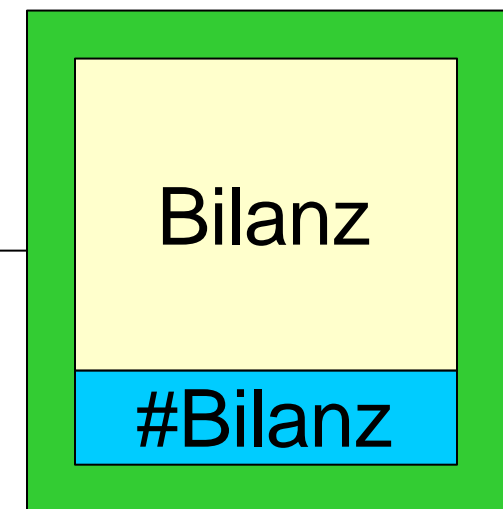
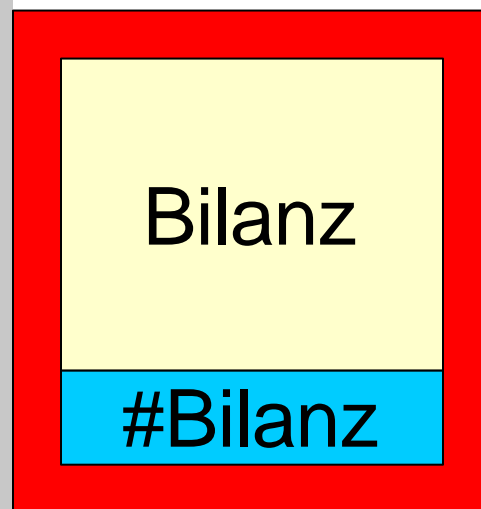


Sign & Encrypt Schwäche

A

B

C



Encrypt & Sign Schwäche

A

C

Patent

Encrypt & Sign Schwäche

A

C

Patent

Encrypt & Sign Schwäche

A

C

Patent

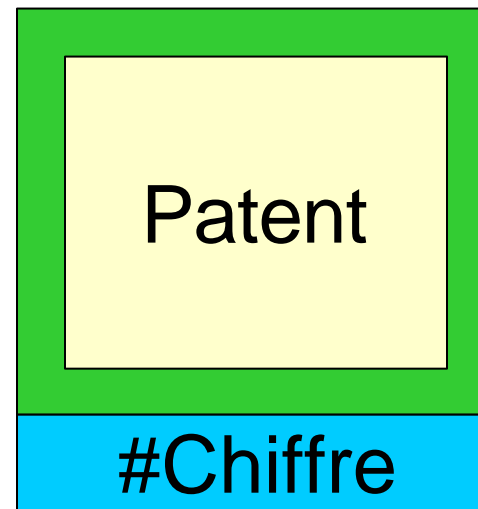
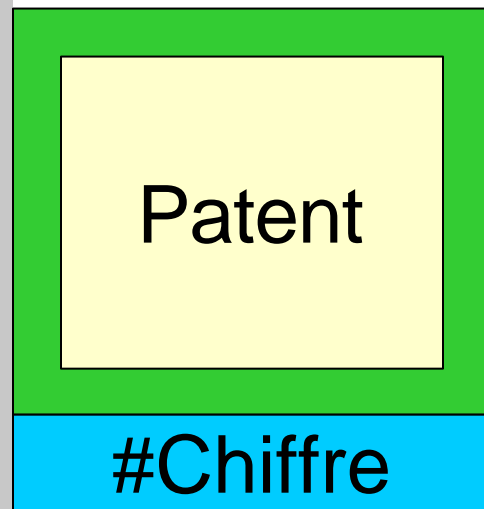
#Chiffre

Encrypt & Sign Schwäche

A

B

C

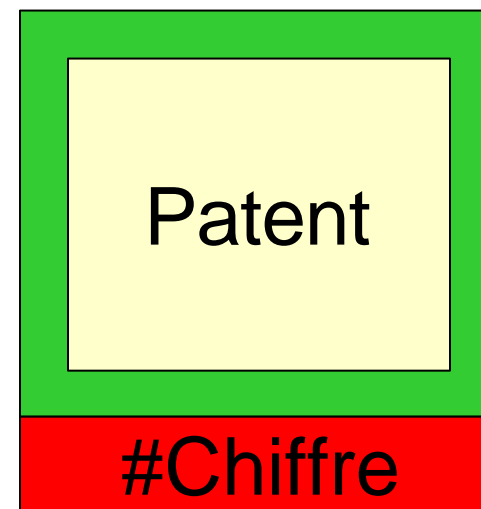
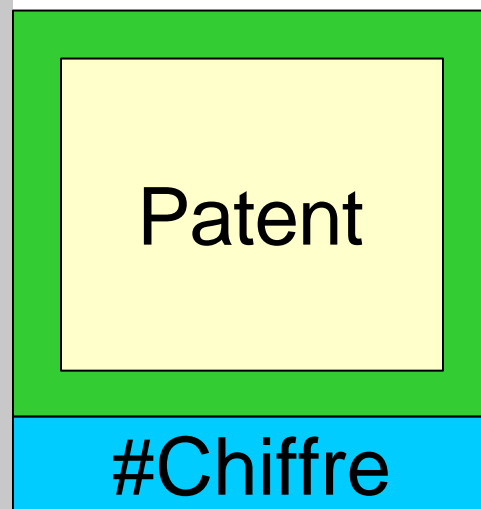


Encrypt & Sign Schwäche

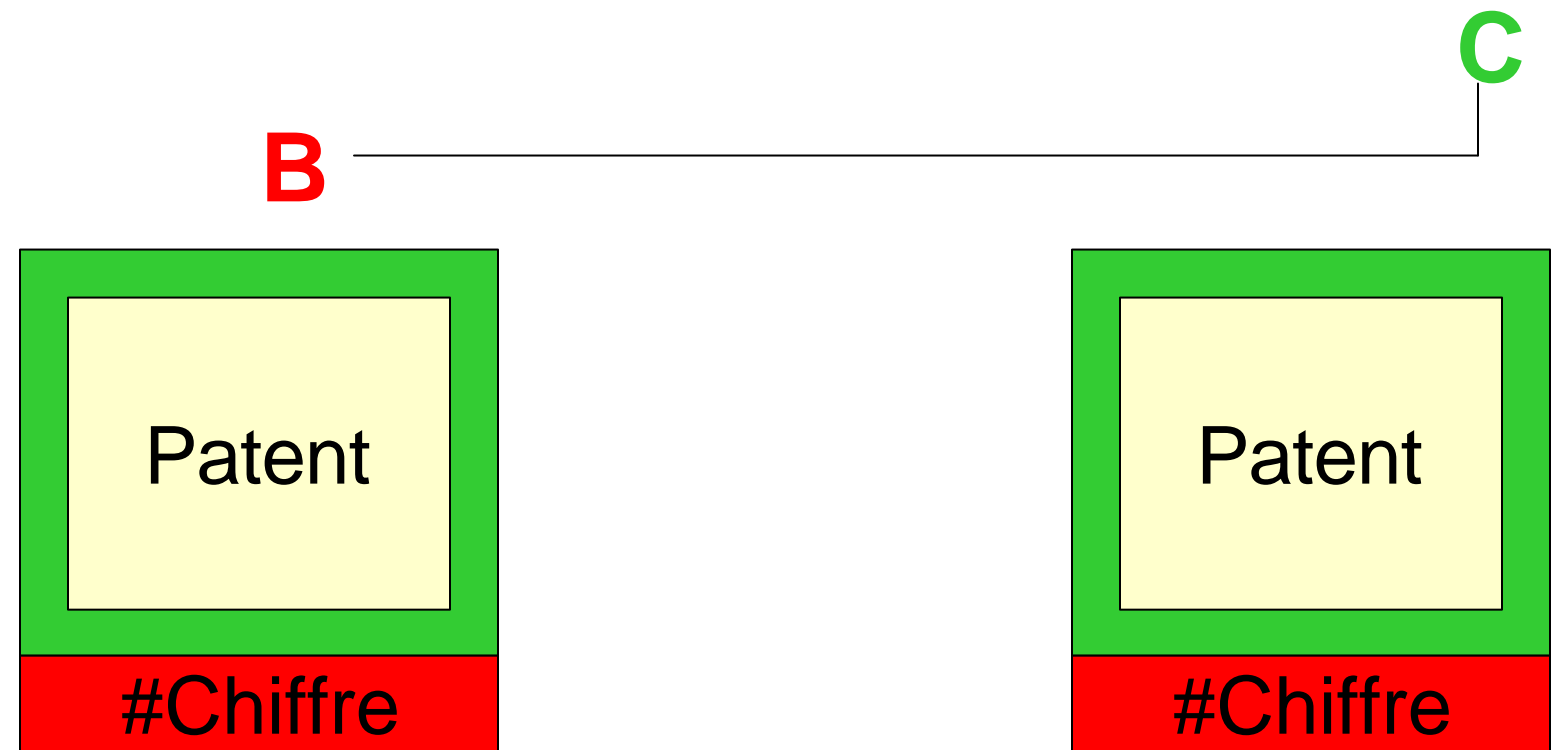
A

B

C



Encrypt & Sign Schwäche



Zusammenfassung

PGP bietet die **Vorraussetzungen** für eine **sichere Kommunikation**

Aber nur bei Zusammenspiel aller Funktionen:

- ? sichere Passphrase
- ? Sicherheit des privaten Schlüssels
- ? ausreichend starke Verschlüsselung
- ? Signatur mit verifizierter Unterschrift
- ? Wissen um die Schwächen asymmetrischer Verfahren (E&S- bzw. S&E-Schwäche)
- ? Wissen, dass es keine absolut sichere Kommunikation geben kann

Konzepte von Betriebssystem- Komponenten

PGP

Verwendete Algorithmen,
PGP aus Benutzersicht,
Schlüsselzertifizierung

von Alexander Dreweke