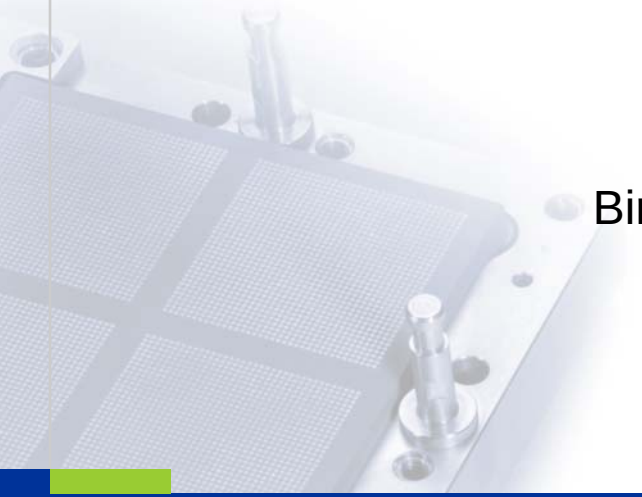
R|R|Z|N

Vertrauenswürdige Kommunikation durch digitale Signaturen



Sicherheitstage WS 05/06

Birgit Gersbeck-Schierholz, RRZN

R|R|Z|N

Schlüsselweitwurf

Zum besseren Verständnis alles auf symmetrische V. projezieren: hybride Verfahren asym. V. wird nur für den geheimen Schlüsseltausch benötigt

Einleitung und Überblick

R|R|Z|N

Motivation und Grundlagen:

- **Sicherheitsziele der digitalen Signatur**
- **Kryptografische Techniken**
- **Funktion einer CA**

Zertifikate von der UH-CA:

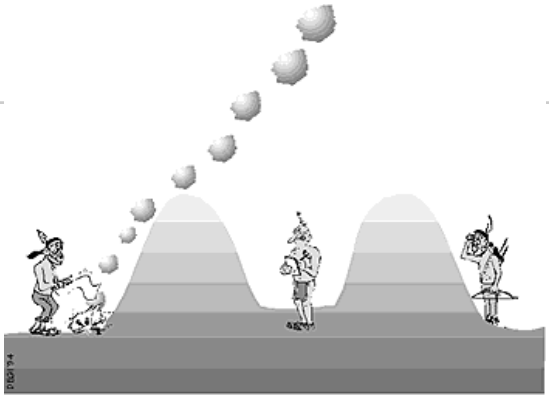
- **Konkrete Einsatzgebiete**
- ***Beantragungsverfahren***
- ***Neues Web-Frontend der UH-CA***

PKI:

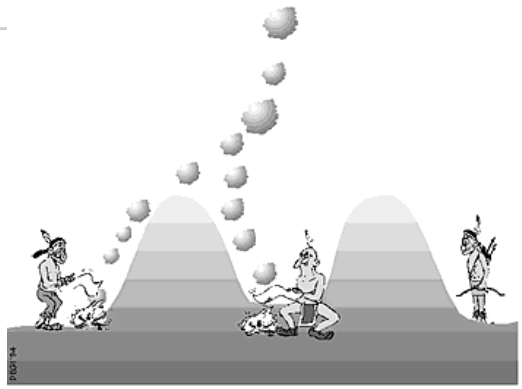
- **DFN-weite Public Key Infrastruktur**
- **UH-CA, Zertifizierungsinstanz der Universität Hannover**

Sicherheitsziele der digitalen Signatur

R|R|Z|N

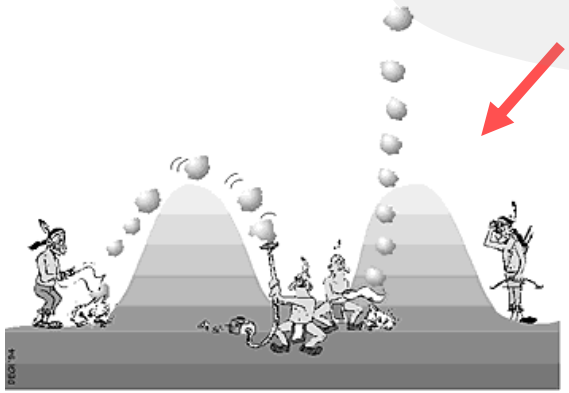


Mitlesen der Daten durch Dritte

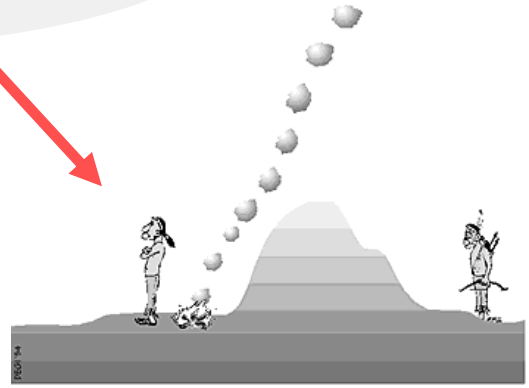


Verändern der Daten

Bekannte **Risiken** der Internet-Kommunikation



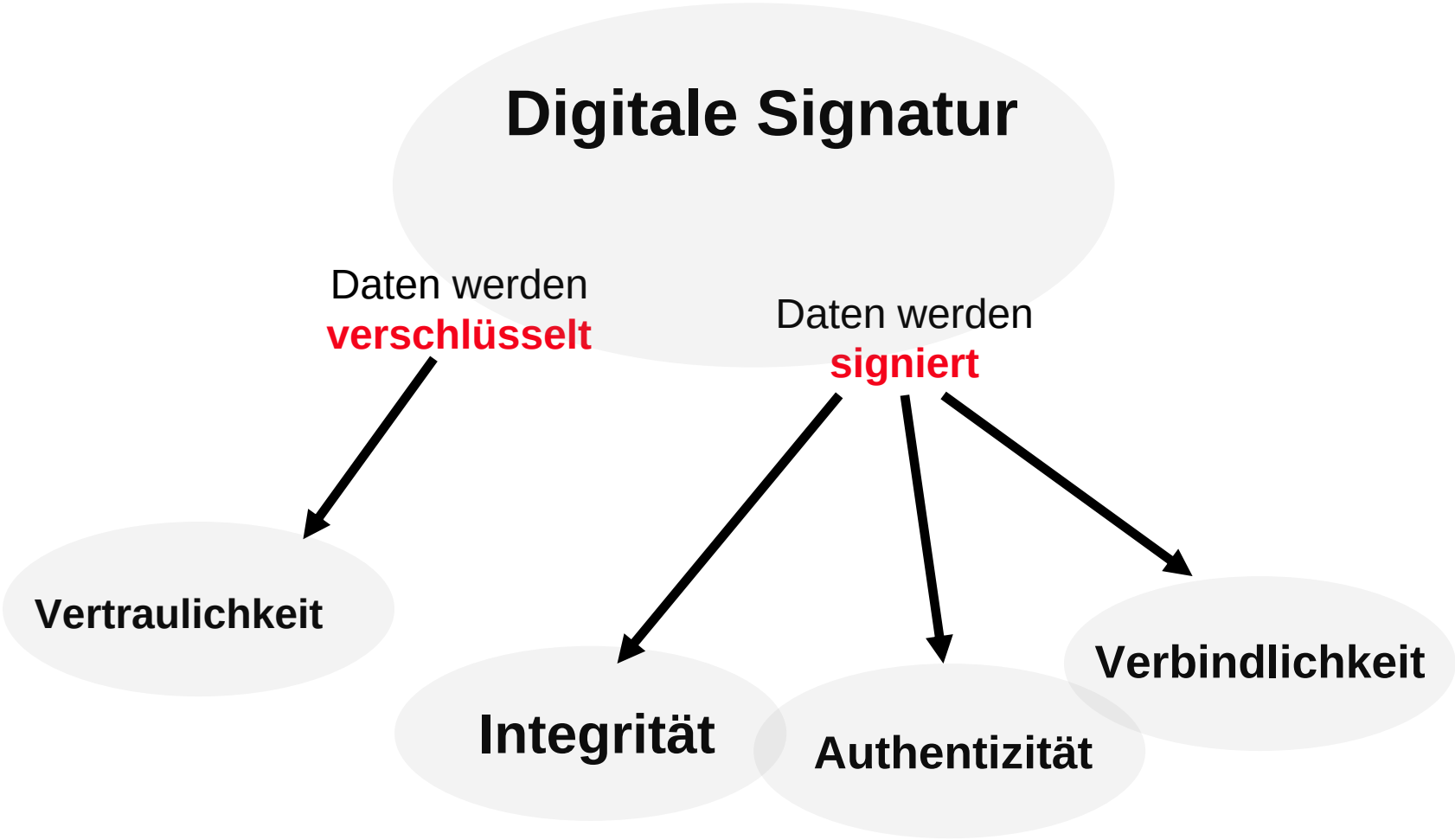
Fälschen der Identität des Absenders



Abstreiten der Daten

Quelle der Grafiken: CryptMail User's Guide, Copyright © 1994 Utimaco Belgium

Sicherheitsziele der digitalen Signatur



Sicherheitsziele der digitalen Signatur

R|R|Z|N

- **Digitale Zertifikate sind elektronische Dateien, die Personen und Server eindeutig identifizieren**
- **Digitale Zertifikate ermöglichen die authentische und vertrauliche Kommunikation über das Internet**

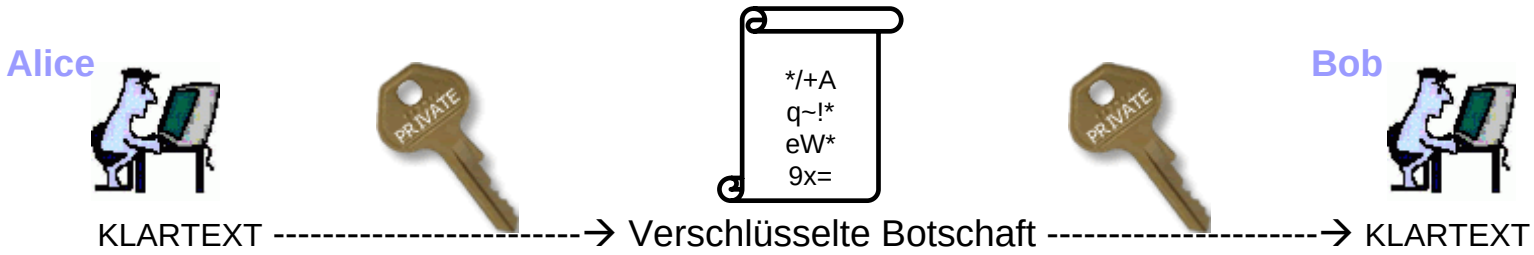
R|R|Z|N

Kryptografische Techniken

Symmetrische Verschlüsselung

- Ältestes Verfahren zur Verschlüsselung -> „Cäsar-Chiffre“
- Moderne Symmetrische Verschlüsselungsverfahren:
  - 3DES, IDEA, Blowfish, AES usw. (Schlüssellänge derzeit ca. 128 – 196 bit)

- Beide Kommunikationspartner benutzen den gleichen Schlüssel zur Ver- und Entschlüsselung



Prinzip: Es gibt nur einen Schüssel (Geheimnis)

Vorteil: Sicheres Verfahren mit guter Performance aufgrund rel. geringer Schlüssellängen

Nachteil: der **Schlüsselaustausch** ist nur über ein persönliches Treffen oder einen Kurier zu realisieren

- Mühseliger Transport von Schlüsseln über weite Entfernungen
- Sichere Verbindung zum Schlüsseltausch notwendig

R|R|Z|N

Asymmetrische Verschlüsselung

- Gibt es seit 1976 (Diffie/Hellmann)
- Das Revolutionäre war damals **die Idee die Schlüssel zum Ver- und Entschlüsseln zu trennen**

- Jeder Kommunikationspartner verfügt über ein **Schlüsselpaar**, dieses besteht aus 2 unterschiedlichen Schlüsseln:

  1. ***Geheimer Schlüssel*** *(private key)* zum Verschlüsseln (nur für mich)

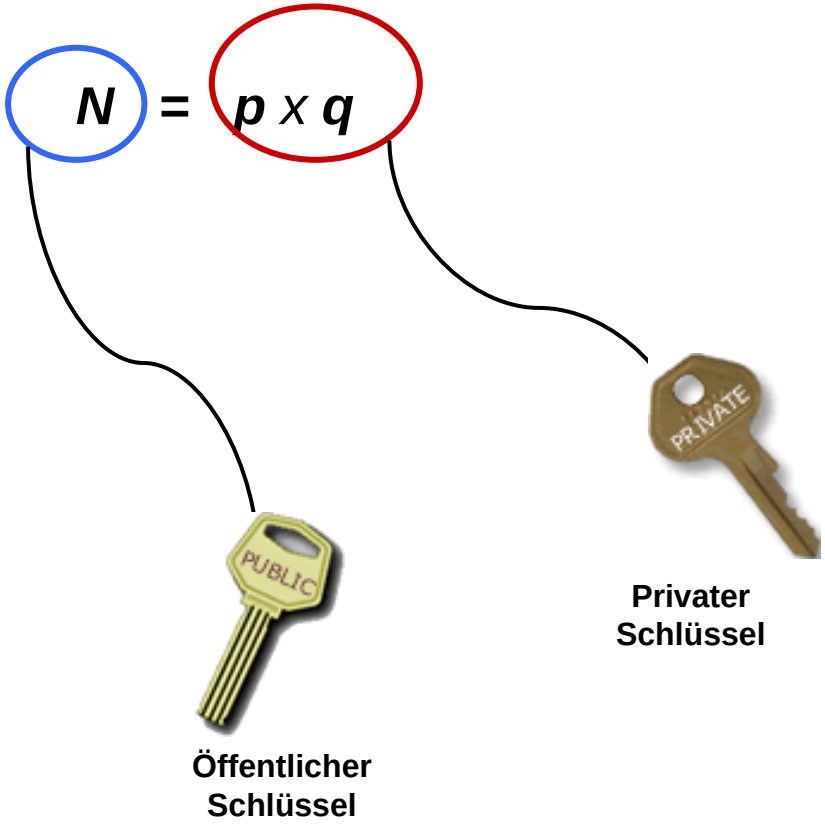
  2. ***Öffentlicher Schlüssel*** *(public key)* zur Rückgewinnung des Textes (für alle anderen)

- Was mit dem öffentlichen Schlüssel verschlüsselt wurde, kann nur mit dem geheimen Schlüssel lesbar gemacht werden und umgekehrt!

- **Worauf basiert die Asymmetrische Verschlüsselung?**
  - Einwegfunktionen (Primzahlen, diskrete Logarithmen, elliptische Kurven)

- **Aktuelle Verfahren:**
  - 1976 Diffie-Hellmann (Schlüsseltausch)
    - Diskrete Logarithmen
  - 1977 RSA – Rivest, Shamir, Adleman
    - Primzahlen
  - El´Gamal
    - Diskrete Logarithmen
  - 1985 Elliptische Kurven (ECC)
  - 1994 DAS – Digital Signature Algorithm
    - Modifikation des El´Gamal-Verfahrens

- RSA
  - Basiert im Wesentlichen auf der Schwierigkeit der Primfaktorzerlegung



***p und q sind sehr große* Primzahlen**

- Exkurs: **Wie sicher ist die asymmetrische Verschlüsselung?**
  - Bei Auswahl hinreichend großer Primzahlen können **p** und **q** nicht ermittelt werden

Beispiel: **RSA-640**

N= 310 7418240490 0437213507 5003588856 7930037346 0228427275 4572016194
8823206440 5180815045 5634682967 1723286782 4379162728 3803341547
1073108501 9195485290 0733772482 2783525742 3864540146 9173660247
7652346609

wurde am 2. November 2005 faktorisiert und zwar in:

p= 1634733 6458092538 4844313388 3865090859 8417836700 3309231218
1110852389 3331001045 0815121211 8167511579
×
q= 1900871 2816648221 1312685157 3935413975 4718967899 6851549366
6638539088 0271038021 0449895719 1261465571

→20.000 US-Dollar

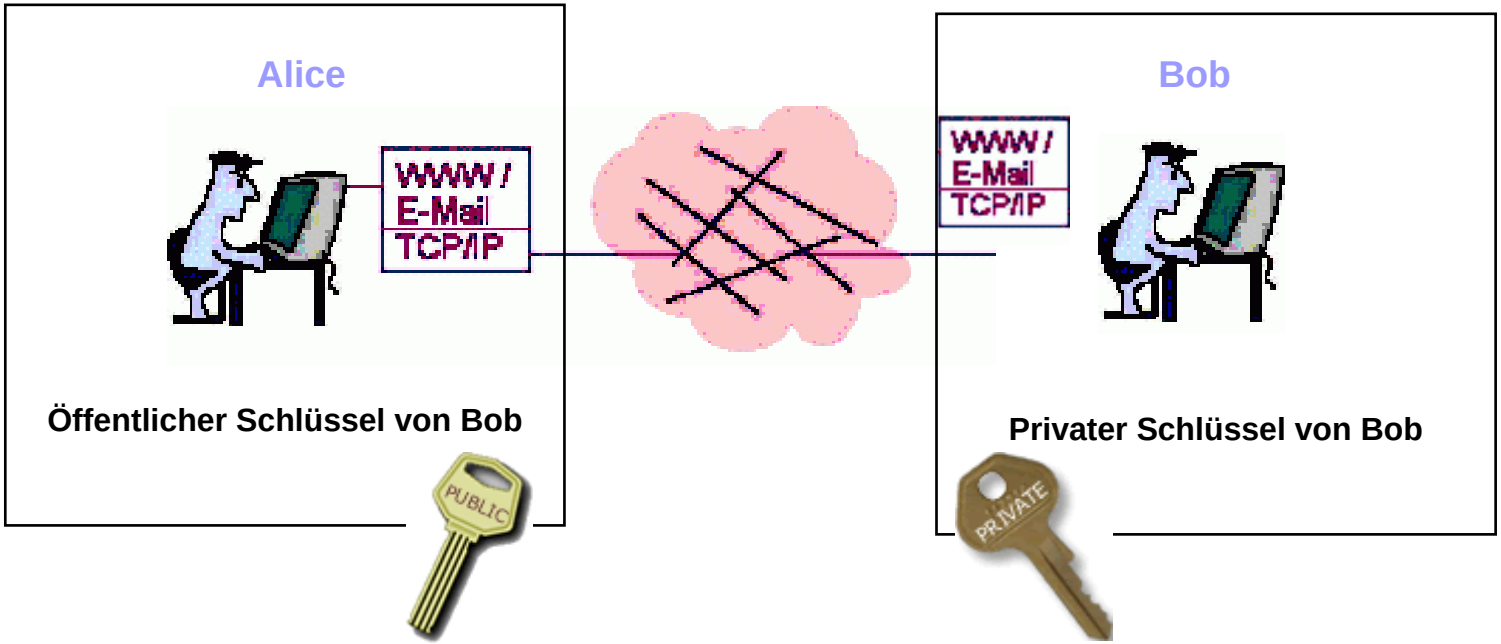
http://www.rsasecurity.com/rsalabs/node.asp?id=2093

- Die aktuell für RSA-Verschlüsselungsverfahren empfohlenen Schlüssellängen von **1024** und **2048** Bit können mit praktikablen Mitteln nicht entschlüsselt werden

- Asymmetrische Verschlüsselung: **Daten verschlüsseln**

  gewährleistet

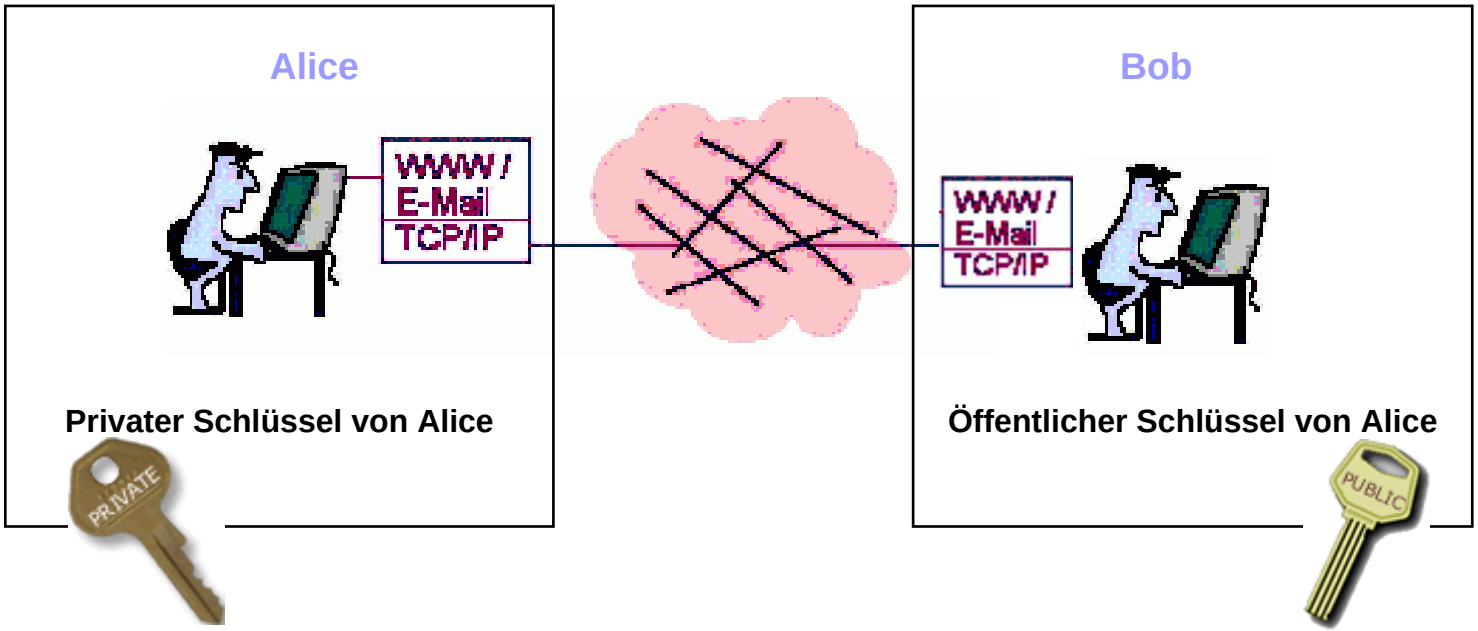
  - Vertraulichkeit : Daten können von Dritten nicht gelesen werden



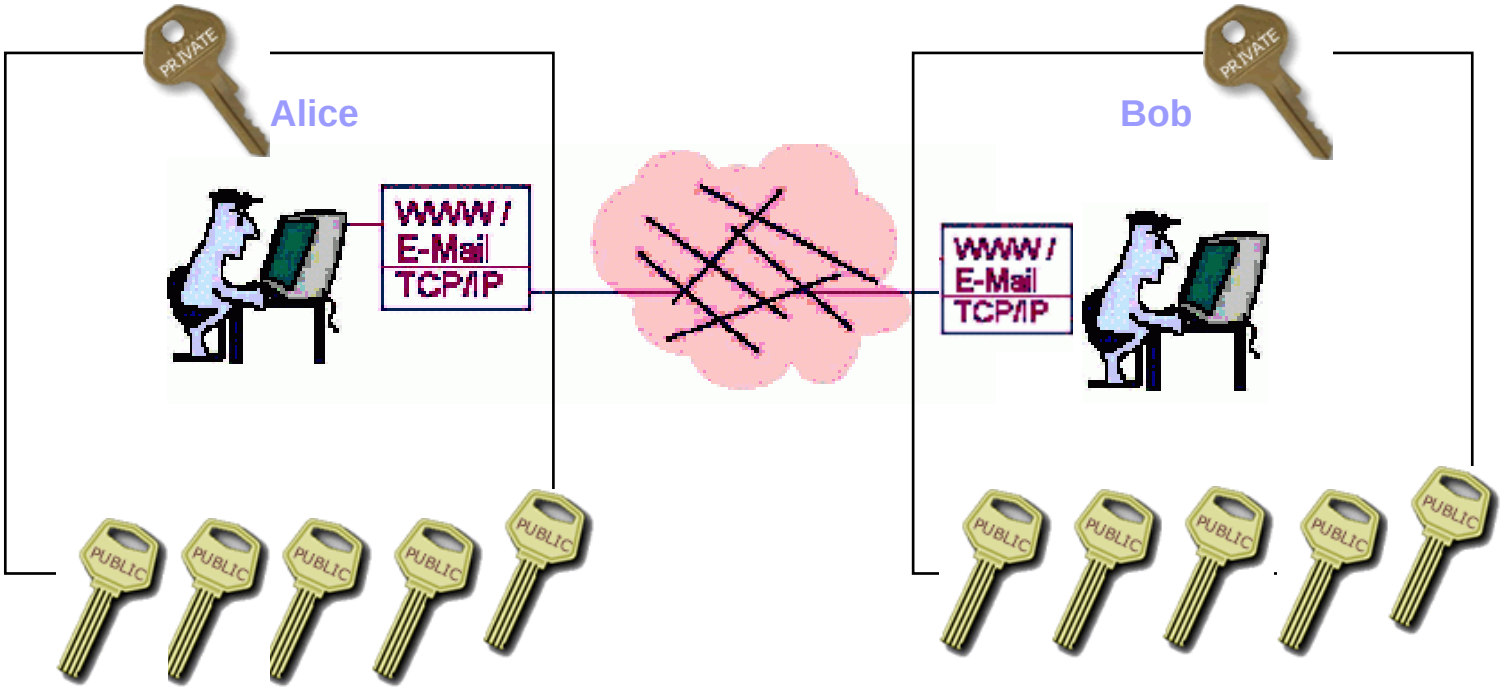
- Asymmetrische Verschlüsselung - **Daten signieren**

  gewährleistet

  - Authentizität : Identität des Absenders ist eindeutig
  - Integrität : Daten wurden nicht verändert
  - Verbindlichkeit : kann nicht mehr abgestritten werden

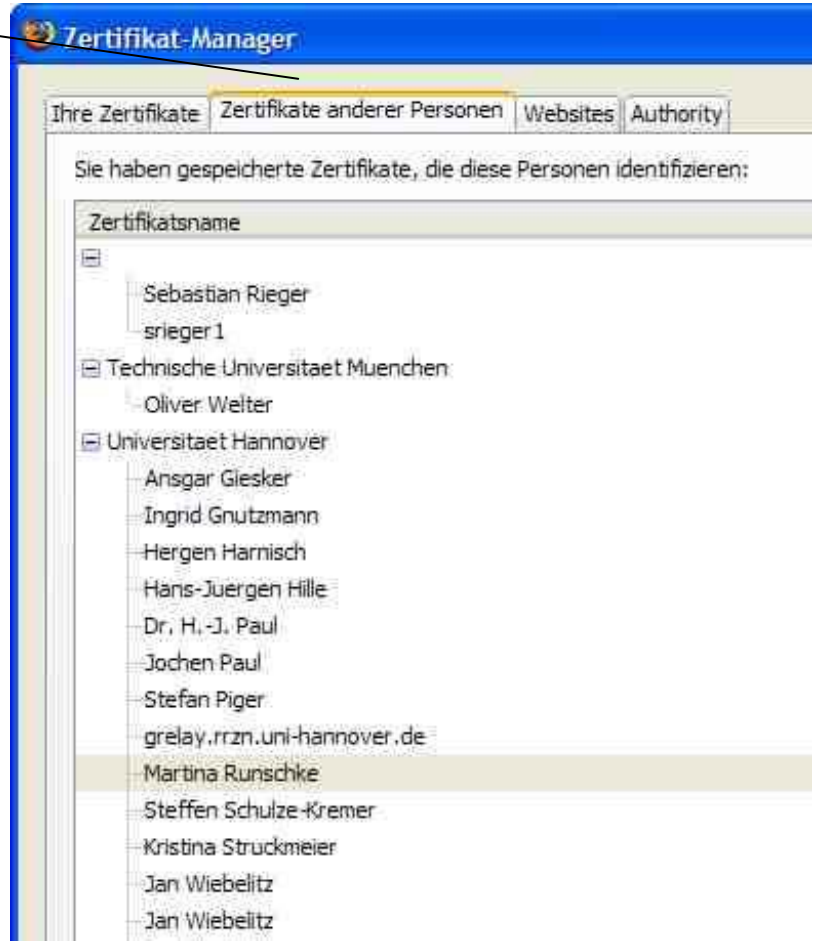


- **Nutzer verfügt in der Regel über**
  - mehrere (viele) Öffentliche Schlüssel (jeweils einen pro Kommunikationspartner)
  - einen privaten Schlüssel



- **Schlüsselmanagment im Browser und Mailklienten**

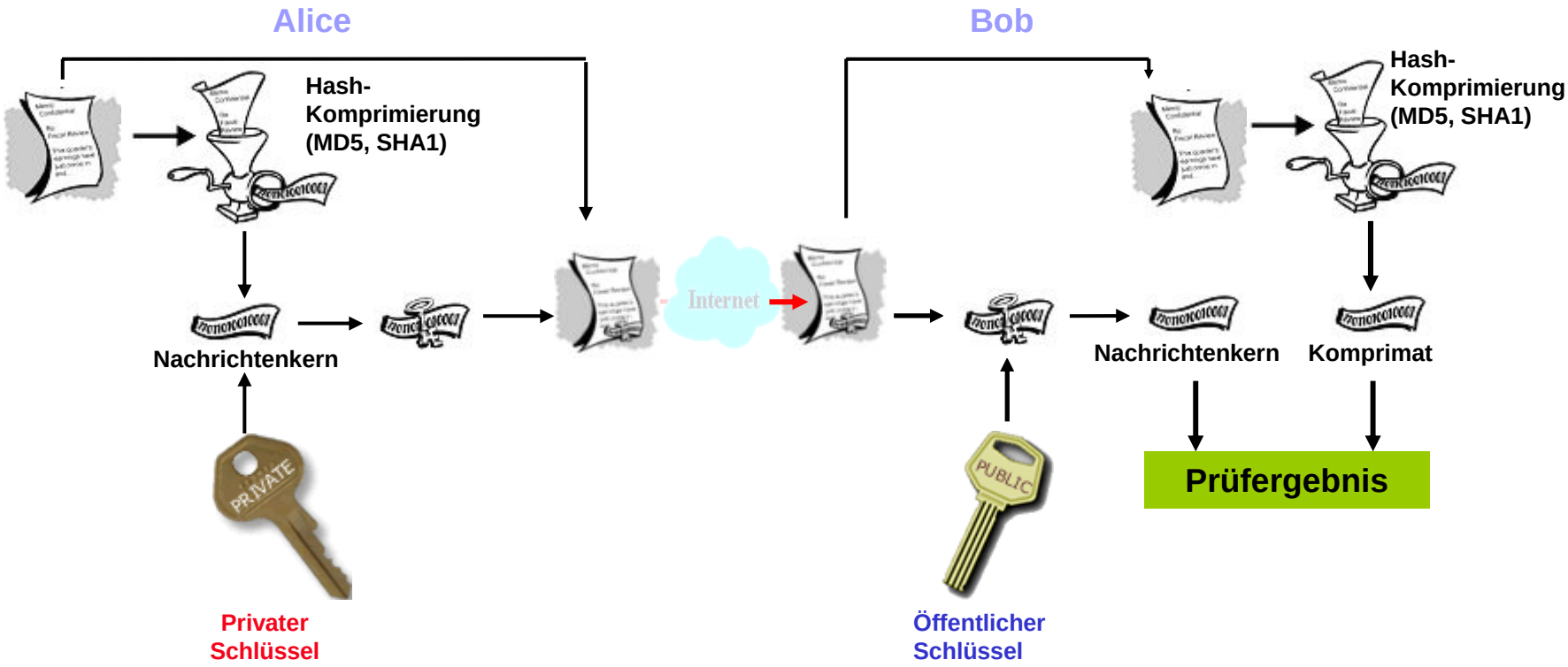
Öffentliche Schlüssel der Kommunikationspartner



R|R|Z|N

- **Prüfsummen/Hashverfahren**
  - Verschlüsselung ohne Schlüssel
  - Hash-Wert:
    - Ein „digitaler Fingerabdruck“ eines Objektes
    - Verkürzung des Objektes auf einen „Identifikator“
  - Aktuelle Verfahren
    - HAVAL (basierend auf MD5)
    - HMAC
    - MD2, MD4, **MD5**
    - RIPEMD-160
    - **SHA-1** (basierend auf MD4)
    - SHA-256 (basierend auf MD4)
    - SHA-512 (basierend auf MD4)
    - SNEFRU

- Schematische Darstellung des Signiervorganges (Hybrides Verfahren)



Schlüssel und Zertifikate

Schlüssel und Zertifikate

R|R|Z|N

- **Kann man den Angaben im öffentlichen Schlüssel vertrauen?**
  - Wie kann ich erkennen, wem ein öffentlicher Schlüssel wirklich gehört oder ob ich schon eine Fälschung besitze?
  - Dies ist nur dann kein Problem, wenn man den Schlüssel persönlich vom Eigentümer bekommen hat (Vorteil der Schlüsselverteilung gegenüber der symmetrischen Verschlüsselung wäre damit hinfällig).
  - Ein vertrauenswürdiger Dritter kann mittels einer Urkunde/Zertifikat den Schlüssel beglaubigen

- **Zertifikate**
  - Beglaubigung von Schlüsseln
  - stellen die eindeutige, zweifelsfreie Zuordnung zwischen einem Schlüsselpaar und einer Person oder einem Rechner her
  - Dabei wird die **reale Identität** an die **digitale Identität** gebunden

- **Zertifizierungsinstanz, CA (Certification Authority)**
  - Vertrauenswürdiger Dritte
  - Nach Prüfung von Identität und öffentlichem Schlüssel, stellt die CA das Zertifikat aus

Schlüssel und Zertifikate

- ein **Zertifikat** nach dem **Standard X.509v3** besteht im Wesentlichen aus 4 Teilen:

Öffentlicher Schlüssel		00:ab:77:e0:53:4a:4a:6b:42:8b:e0:4b:91:14:6f: df:e7:28:4f:58:e5:43:b5:01:71:fa:24:2f:6c:4e: … 39:04:62:2f:fd:20:4a:a3:d0:00:78:c8:e7:44:7a
Angaben über den Schlüsselinhaber (Common Name nach X.509v3)		C=DE, O=Universitaet Hannover, OU=RRZN, CN=Birgit Gersbeck-Schierholz/serialNumber=3
Attribute wie Seriennummer und Gültigkeitsdauer		Serial Number: 3 (0x3) Signature Algorithm: sha1WithRSAEncryption Validity Not Before: May 26 15:42:55 2004 GMT Not After : May 26 15:42:55 2005 GMT
Beglaubigung (digitale Signatur) der CA, dass die Angaben stimmen		-----BEGIN CERTIFICATE----- …MBwGA1UEChMVVW5pdmVyc2l0YWV0I m5vdmVyMRAwDgYDVQQLFAdSUlpOX0NB… sDUNW/3L63Epiozxuah/9jzoLI+/Q32Dg== -----END CERTIFICATE-----

Schlüssel und Zertifikate

R|R|Z|N

- **X.509 Hierarchie**
  - Aufbau einer Zertifikatkette
  - Dadurch, dass die CA wiederum von einer übergeordneten Zertifizierungsstelle beglaubigt wird, entsteht eine Hierarchie des Vertrauens
  - Die Aussteller-Signatur in einem Zertifikat ist jeweils mit dem privaten Schlüssel des Ausstellers verschlüsselt worden
  - Mit dem im Zertifikat enthaltenen öffentlichen Schlüssel kann das in der Hierarchie darüber liegende Zertifikat verifiziert werden

Schlüssel und Zertifikate

R|R|Z|N

- **Zertifikat : Angaben im Browser**

Zertifikatnehmer

Zertifizierungsstelle

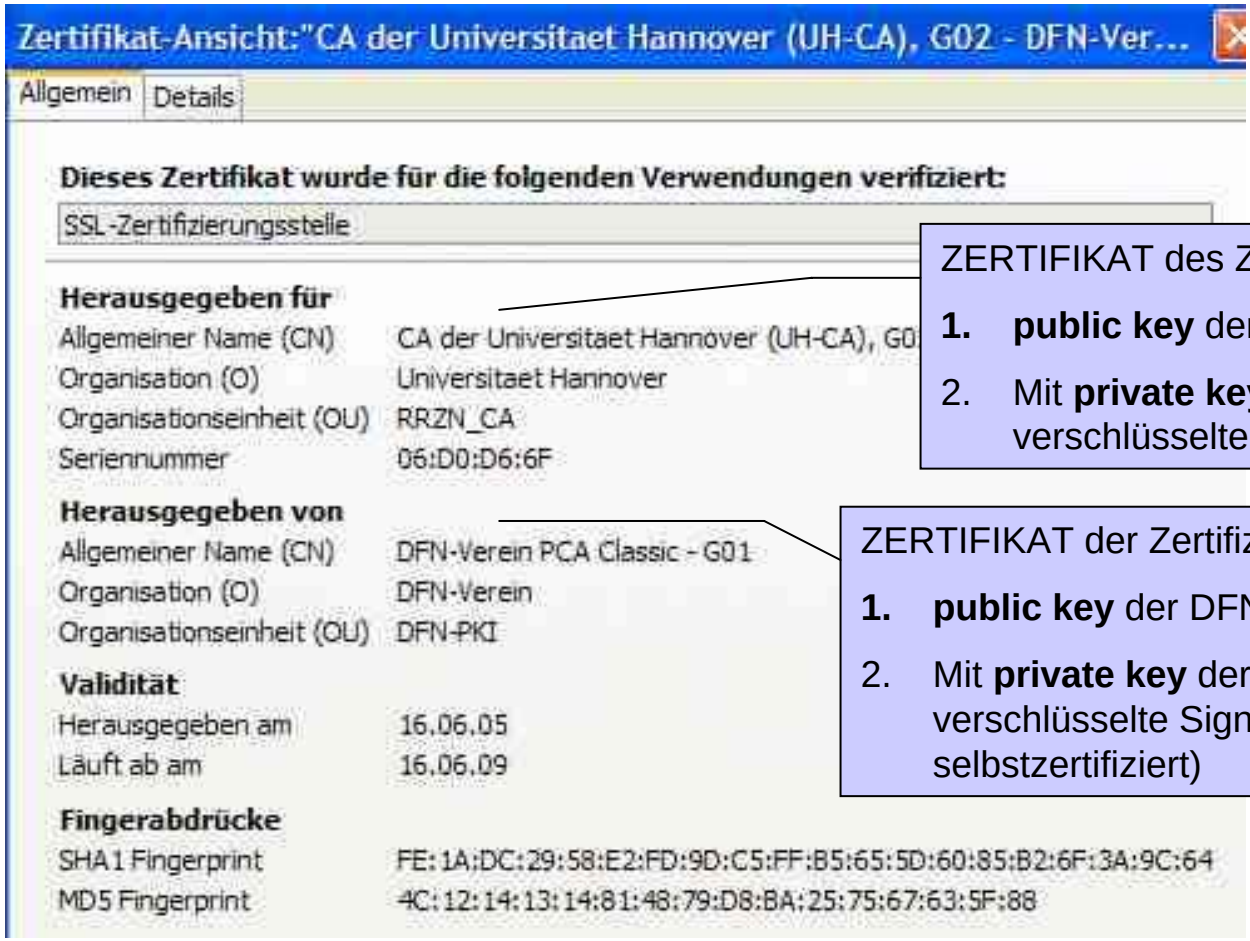
Gültigkeitszeitraum

Fingerabdruck, Hashfunktionen MD5 und SHA1

Schlüssel und Zertifikate

RRZN

- **Zertifikat : Angaben im Browser**



ZERTIFIKAT des Zertifkatnehmers UH-CA

1. **public key** der UH-CA
2. Mit **private key** der DFN-PCA verschlüsselte Signatur

ZERTIFIKAT der Zertifizierungsstelle DFN-PCA

1. **public key** der DFN-PCA
2. Mit **private key** der darüberliegenden Instanz verschlüsselte Signatur (da Root-CA, selbstzertifiziert)

Digitale Zertifikate - praktische Anwendung

Digitale Zertifikate - praktische Anwendung

R|R|Z|N

- **S/MIME Zertifikate**
  - Persönliches Zertifikat zum Unterschreiben (signieren) und Verschlüsseln von E-Mail
- **SSL Server-Zertifikate**
  - Ermöglicht gesicherten Datenaustausch nach einer Server-Authentifizierung
  - Gewährleistet damit Vertraulichkeit, Integrität, einseitige Authentifizierung
- **SSL Client-Zertifikate**
  - Ermöglicht gesicherten Datenaustausch nach Authentifizierung beider Kommunikationspartner
  - Gewährleistet damit Vertraulichkeit, Integrität, beidseitige Authentifizierung und ersetzt so Credentials wie z. B. Username-Passwort-Paare -> „Single Sign On“
- **Code Signing**
  - Gewährleistet Authentizität und Integrität von Programm-Code (z.B. Office-Macros)

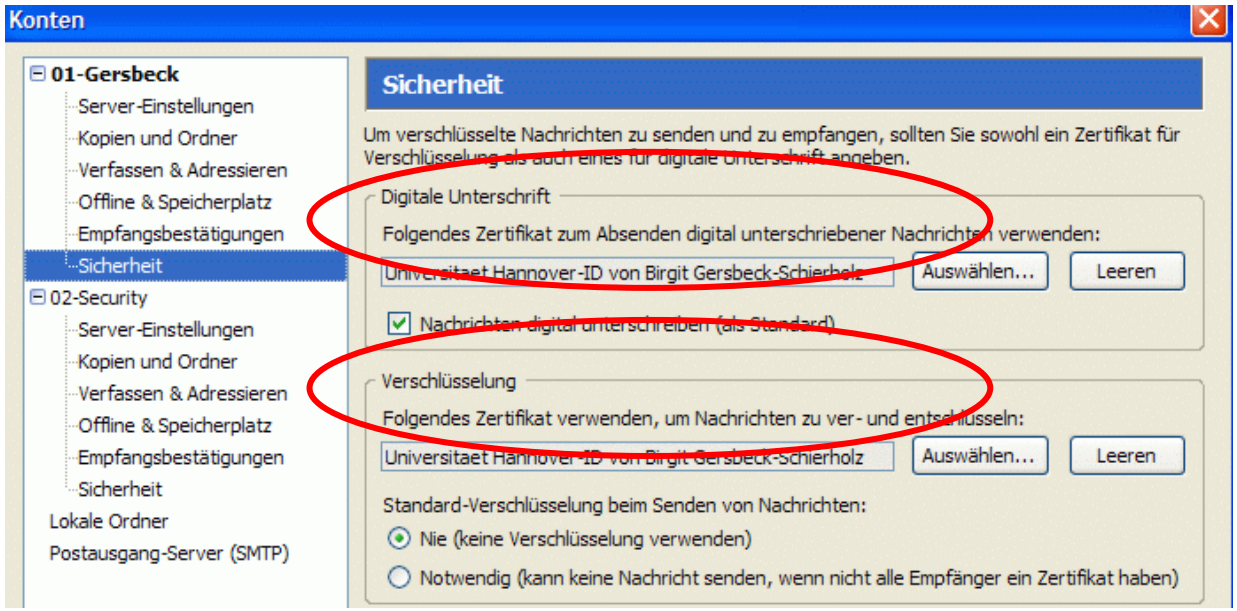
Digitale Zertifikate - praktische Anwendung

R|R|Z|N

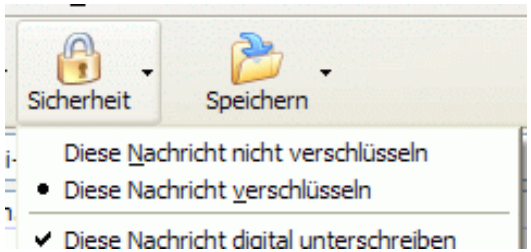
Signieren und Verschlüsseln von E-Mail

(Beispiel: Thunderbird)

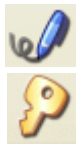
Implementieren:



Verfahren wählen:



Darstellung beim Empfänger:

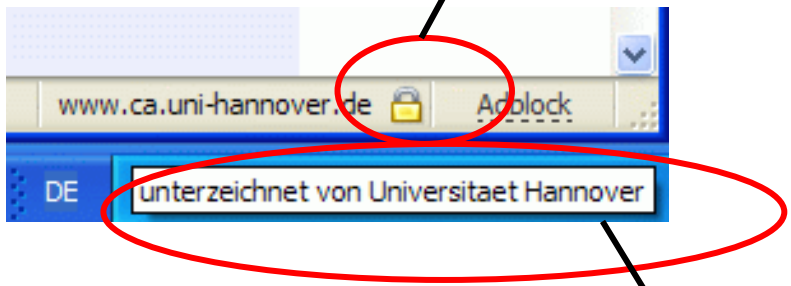


Digitale Zertifikate - praktische Anwendung

R|R|Z|N

- **Gesicherte Kommunikation mit Servern**

SSL-verschlüsselte Server-Verbindung
(https)



Zertifizierungsstelle (CA)

Digitale Zertifikate - praktische Anwendung

R|R|Z|N

- **Warum HTTPS?**

- **Weil HTTP…**

  - … nicht die Vertraulichkeit der Daten gewährleistet
  - … keine Möglichkeit bietet die Integrität der Daten zu prüfen
  - … nur äußerst unzureichende Methoden zur Authentisierung liefert
  - … Digest Authentisierung zwar ein verschlüsseltes Passwort überträgt, aber die weitere Kommunikation wieder unverschlüsselt abläuft

Digitale Zertifikate - praktische Anwendung

R|R|Z|N

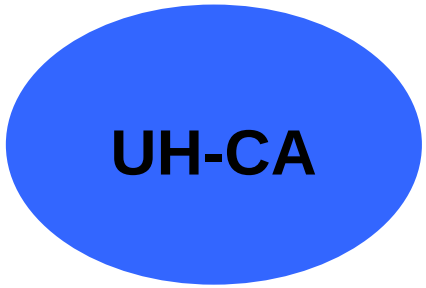
- **HTTPS Server**
  - Verfügen über ein X.509 Zertifikat
  - Authentizität wird vom Klienten überprüft
  - Kommunikation erfolgt verschlüsselt, z.B. Austausch von Formulardaten, Login, ..
  - Können vom Klienten eine Authentisierung verlangen
  - Gewährleisten Schutz vor Manipulation

Certification Authority: UH-CA

Certification Authority: UH-CA

R|R|Z|N

- **Zertifizierungsstelle der Universität Hannover seit Mai 2004**
- **Benutzerschnittstelle*:***
  www.rrzn.uni-hannover.de/zertifizierung.html



- **Zertifiziert öffentliche Schlüssel für Nutzer und Server**
- **Zertifikate für Mitglieder der Universität Hannover**
- **Integriert in die PKI (Public Key Infrastructure) des Deutschen Forschungsnetzes**

DFN CERT

Certification Authority: UH-CA

R|R|Z|N

- **Zertifizierungsrichtlinien zum Betrieb der UH-CA**

- **Basis für das Vertrauen der Zertifikatnehmer und der Zertifikat-Nutzer („relying party“)**

http://www.rrzn.uni-hannover.de/uhca-policy.html

Policy

- **Beschreibt das Zertifikat-Format nach dam Standard X.509**

- **Definiert die Sicherheitsanforderungen an die CA**
  - ***Persönliche Identifizierung der Teilnehmer***
  - ***Offline CA …***
  - **Veröffentlichungsintervalle von Rückruflisten (CRLs)**

Certification Authority: UH-CA

R|R|Z|N

- **Bearbeitung von Zertifizierungsanträgen:**
  - Online-Antrag
  - Schriftliche Teilnehmererklärung
  - Persönliche Identifizierung
- **Sperrung von Zertifikaten, z.B. bei Kompromittierung oder Schlüsselverlust**

Aufgaben

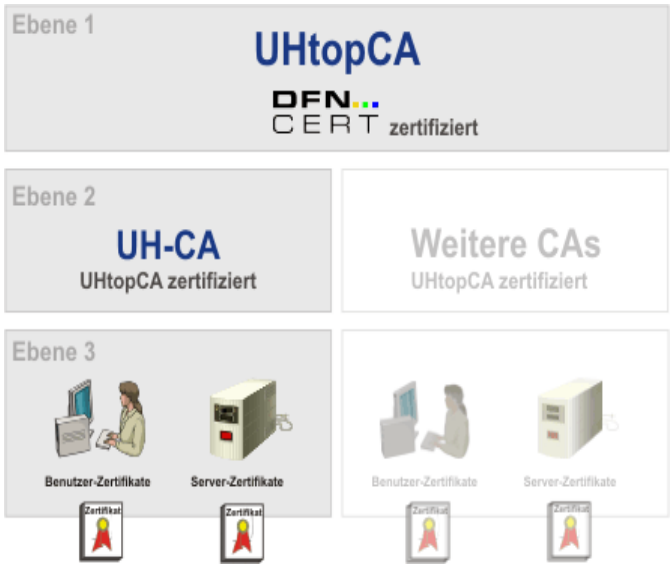
- **Ausstellen von Zertifikaten**
- **Regelmäßige Veröffentlichung von Sperrlisten**
- **Verwalten und Verteilen von Zertifikaten**
  - Verzeichnisdienst
  - Bereitstellung über die Webschnittstelle
  - Benachrichtigung der Zertifikatnehmer (Ausstellung, Sperrung, Gültigkeit)
  - Backup

Certification Authority: UH-CA

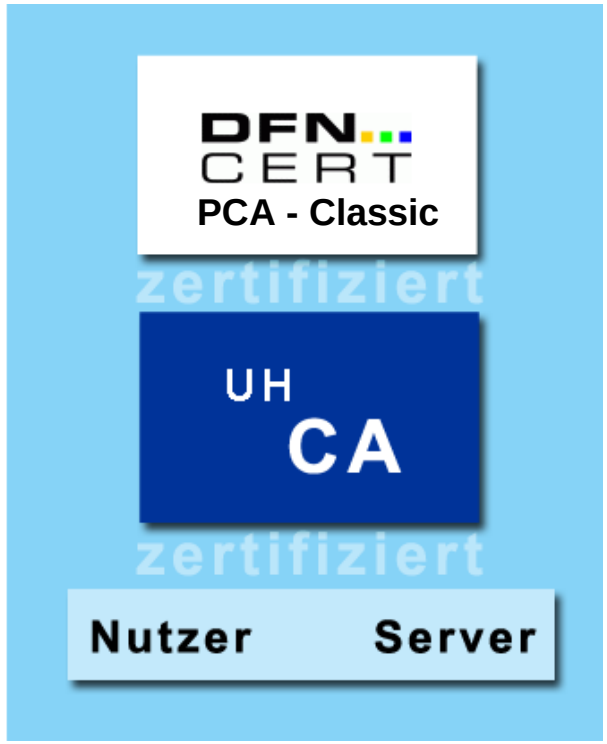
- DFN-PKI - alt und neu

Alte Policy, Laufzeit bis 25.05.2006:

DFN-PCA Wurzelzertifikat (selbstzertifiziert)

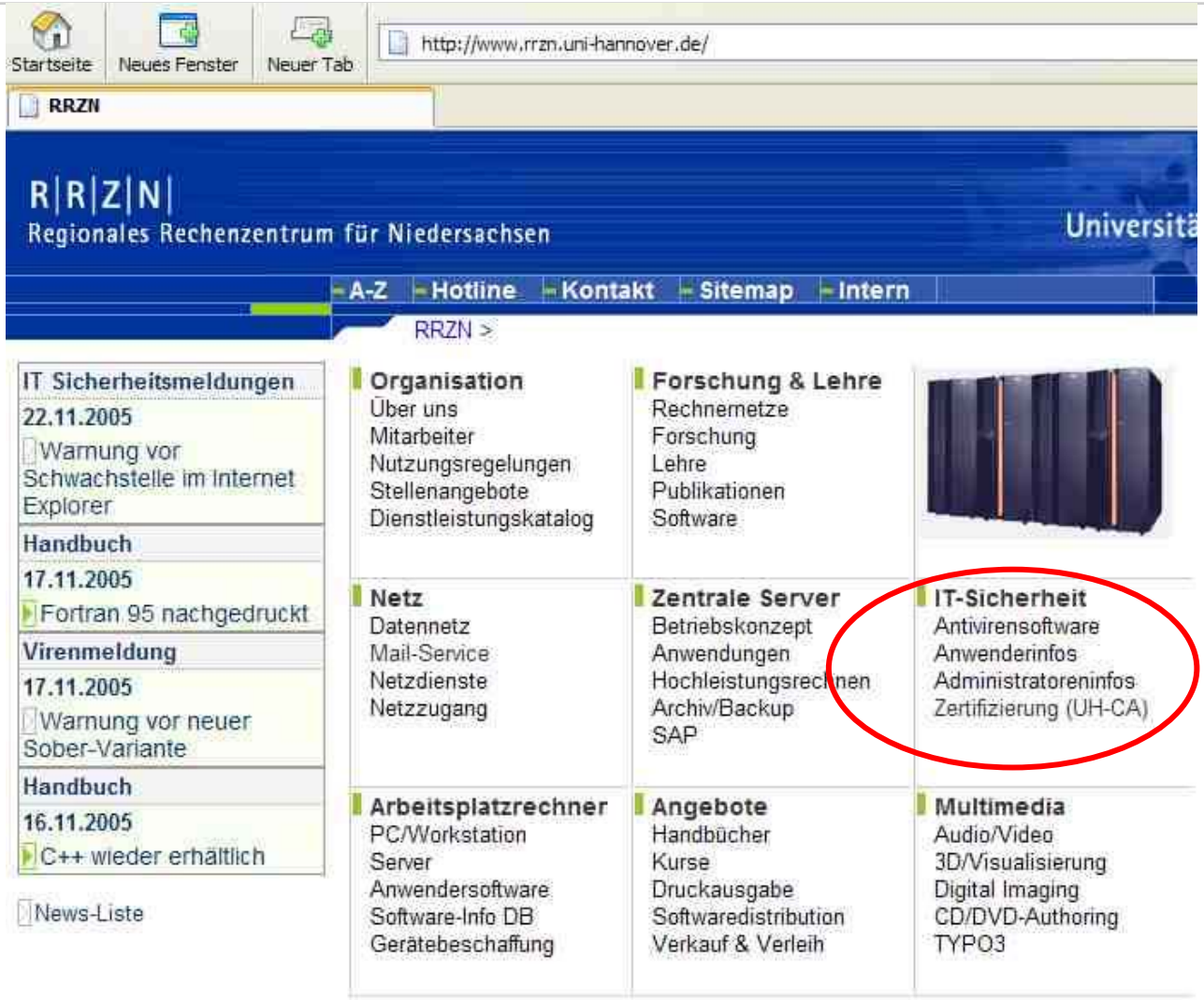


Neue Policy, Laufzeit ab 16. Juni 2005:



Benutzerschnittstelle der UH-CA

Benutzerschnittstelle der UH-CA



Benutzerschnittstelle der UH-CA (neu)

Benutzerschnittstelle der UH-CA (alt)

Benutzerschnittstelle der UH-CA (neu)

Benutzerschnittstelle der UH-CA (neu)

R|R|Z|N

- **Antrag auf ein persönliches E-Mail-Zertifikat**

R|R|Z|N
Regionales Rechenzentrum für Niedersachsen
Universität Hannover

A-Z | Hotline | Kontakt | Sitemap | Intern

RRZN > IT-Sicherheit > Zertifizierung (UH-CA) >

- Organisation
- Forschung und Lehre
- Netze
- Zentrale Server
- IT-Sicherheit
  - Antivirensoftware
  - Anwenderinfos
  - Administratoreninfos
  - Abo Security Mails
  - IT-Sicherheitsbeauftr. UH
  - RRZN-Netzschutz
  - Zertifizierung (UH-CA)
    - FAQ
- Arbeitsplatzrechner
- Angebote
- Multimedia
- News
- Suche

(Kurz-)Anleitungen

- Nutzer-Zertifikate (Signieren und Verschlüsseln von E-Mail)
- Server-Zertifikate

Kurzanleitung für Nutzer-Zertifikate (Signieren und Verschlüsseln von E-Mail)

1. Laden Sie sich unbedingt als erstes die CA Zertifikate herunter und installieren Sie sie in Ihrem Browser.
   - Das Wurzelzertifikat der DFN-PCA
   - Das UH-CA Zertifikat
2. Lesen Sie sich die Policy der UH-CA durch.
3. Füllen Sie den Antrag auf Zertifizierung am Bildschirm aus.
4. Unterschreiben Sie den ausgedruckten Antrag und legen ihn persönlich, (weisen Sie sich mit Ihrem Personalausweis aus) bei der Registrierungsstelle (RA) der UH-CA im RRZN vor.
5. Nach Erhalt der Antwort E-Mail importieren Sie sich das Zertifikat in den Browser.
6. Sichern Sie sich das Zertifikat auf Diskette bzw. einen anderen externen Datenträger.

Eine ausführliche Erklärung zum Beantragen eines Zertifikates finden Sie unter dem Menüpunkt Anleitungen.

Benutzerschnittstelle der UH-CA (neu)

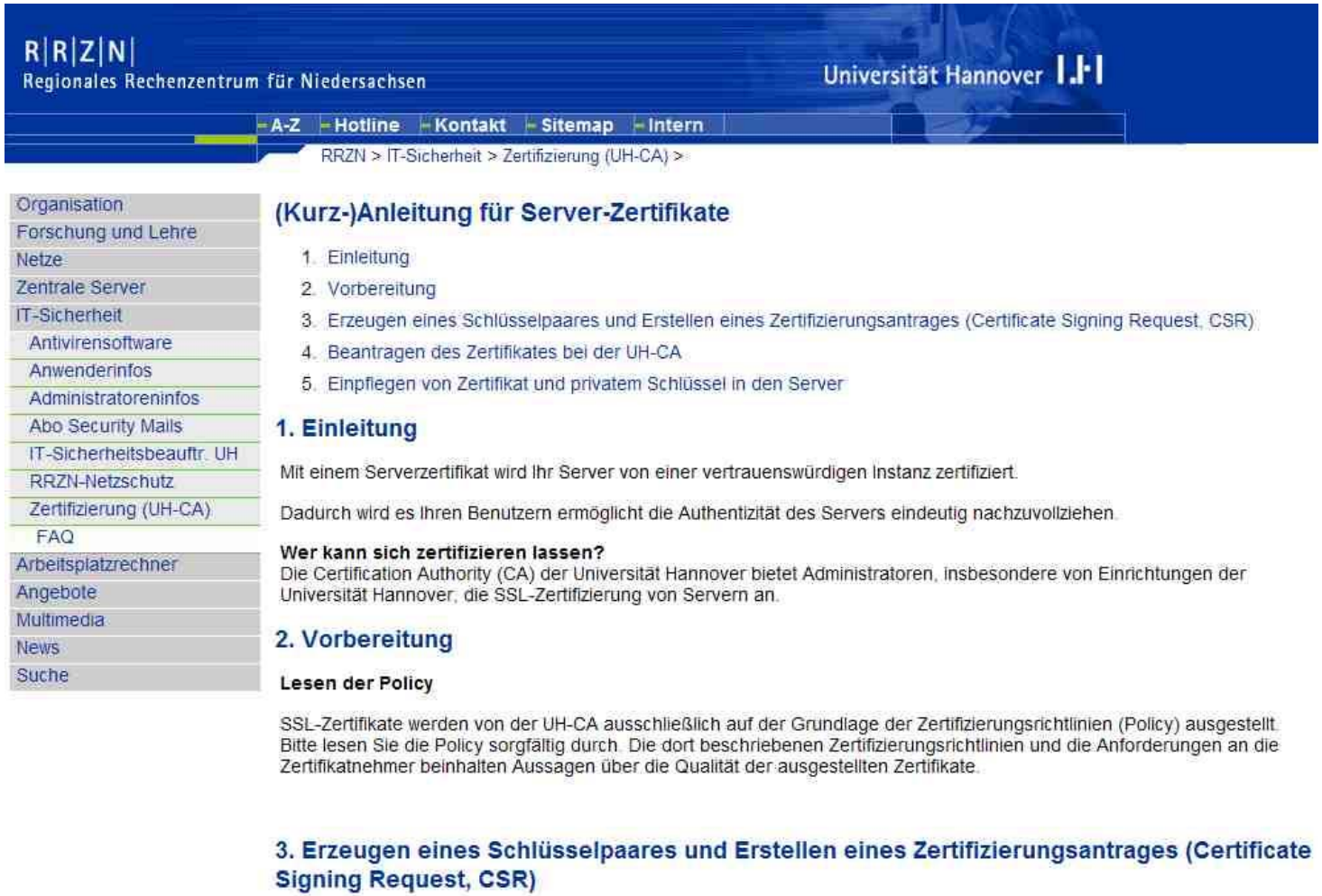
R|R|Z|N

- **Antrag auf ein persönliches E-Mail-Zertifikat**

Benutzerschnittstelle der UH-CA (neu)

R|R|Z|N

- **Antrag auf ein Server-Zertifikat**



R|R|Z|N
Regionales Rechenzentrum für Niedersachsen
Universität Hannover

A-Z | Hotline | Kontakt | Sitemap | Intern

RRZN > IT-Sicherheit > Zertifizierung (UH-CA) >

- Organisation
- Forschung und Lehre
- Netze
- Zentrale Server
- IT-Sicherheit
  - Antivirensoftware
  - Anwenderinfos
  - Administratoreninfos
  - Abo Security Mails
  - IT-Sicherheitsbeauftr. UH
  - RRZN-Netzschutz
  - Zertifizierung (UH-CA)
    - FAQ
- Arbeitsplatzrechner
- Angebote
- Multimedia
- News
- Suche

(Kurz-)Anleitung für Server-Zertifikate

1. Einleitung
2. Vorbereitung
3. Erzeugen eines Schlüsselpaares und Erstellen eines Zertifizierungsantrages (Certificate Signing Request, CSR)
4. Beantragen des Zertifikates bei der UH-CA
5. Einpflegen von Zertifikat und privatem Schlüssel in den Server

1. Einleitung

Mit einem Serverzertifikat wird Ihr Server von einer vertrauenswürdigen Instanz zertifiziert.

Dadurch wird es Ihren Benutzern ermöglicht die Authentizität des Servers eindeutig nachzuvollziehen.

Wer kann sich zertifizieren lassen?
Die Certification Authority (CA) der Universität Hannover bietet Administratoren, insbesondere von Einrichtungen der Universität Hannover, die SSL-Zertifizierung von Servern an.

2. Vorbereitung

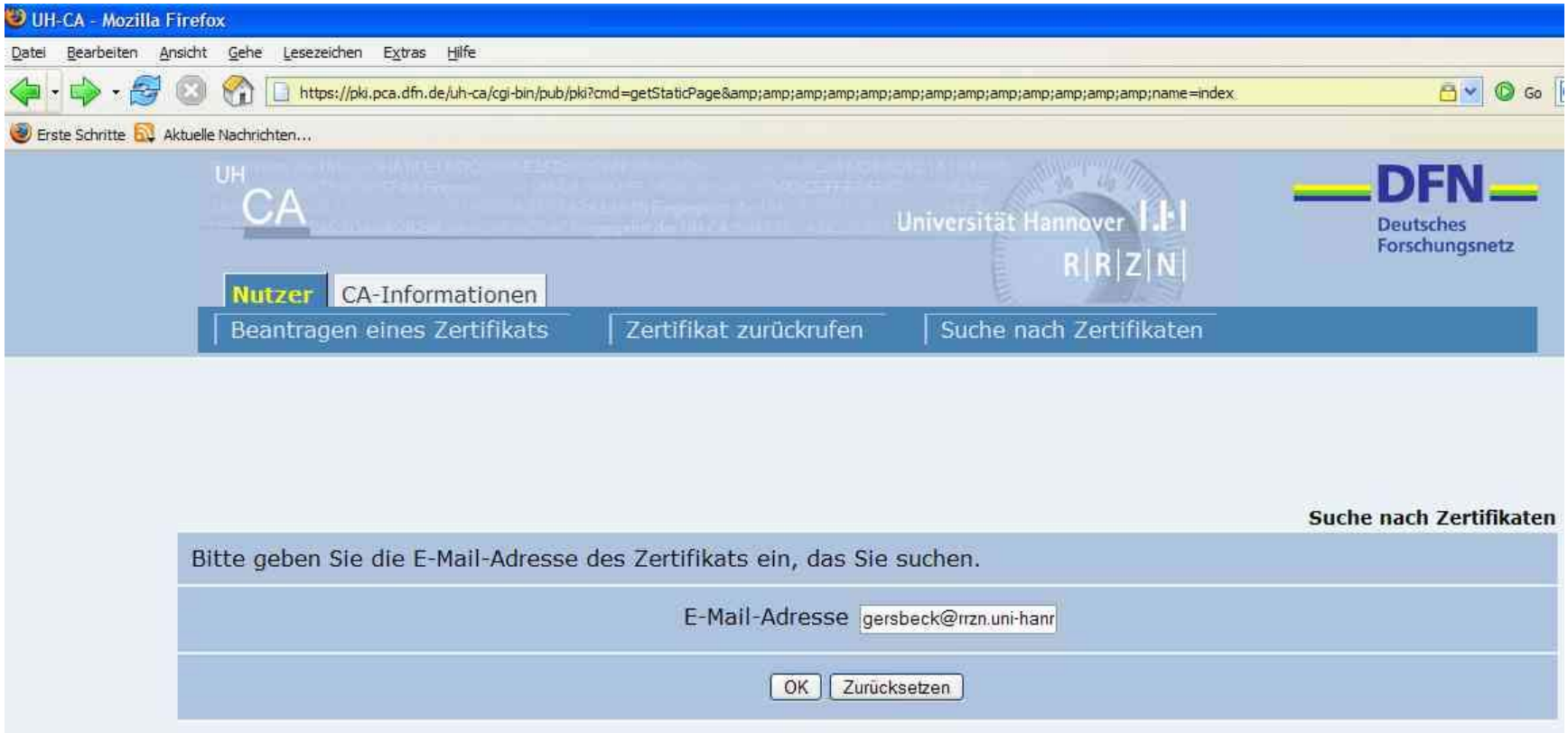
Lesen der Policy

SSL-Zertifikate werden von der UH-CA ausschließlich auf der Grundlage der Zertifizierungsrichtlinien (Policy) ausgestellt. Bitte lesen Sie die Policy sorgfältig durch. Die dort beschriebenen Zertifizierungsrichtlinien und die Anforderungen an die Zertifikatnehmer beinhalten Aussagen über die Qualität der ausgestellten Zertifikate.

3. Erzeugen eines Schlüsselpaares und Erstellen eines Zertifizierungsantrages (Certificate Signing Request, CSR)

Benutzerschnittstelle der UH-CA (neu)

- **Zertifikate suchen, Zertifikat zurückrufen, Sperrlisten**



„Klippen“

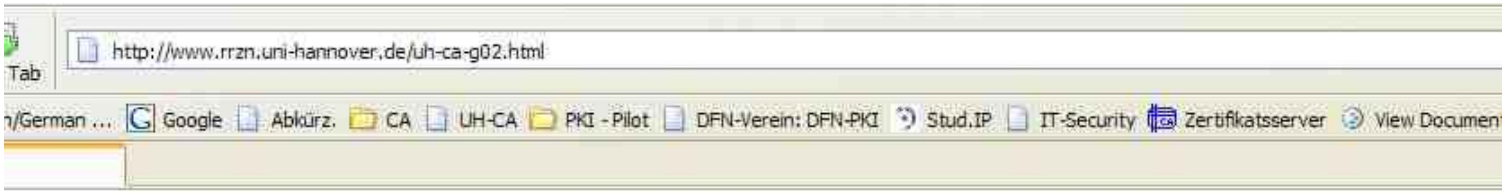
„Klippen“

- Beispiel 1:
  - **Ich erhalte eine signierte E-Mail -> mein Mail-Client stuft die Signatur als ungültig ein und meldet eine wenig Vertrauen erweckende Warnung**
- Beispiel 2:
  - **Ich wähle eine Verbindung zu einem Webserver, der sich mit einem Zertifikat ausweist –> mein Browser schlägt Alarm und warnt mich vor möglichem Betrug**
- Ursache in beiden Fällen:
  - **die Zertifikate der Zertifizierungs-Hierarchie DFN-PCA, UH-CA sind noch nicht in Standard-Browsern und Mail-Klienten enthalten.**
- Etwas lästig aber unabdingbar: die CA-Zertifikate müssen nachinstalliert werden.

„Klippen“

R|R|Z|N

- **Zertifikate der Zertifizierungsinstanzen:**



2. CA-Zertifikate zum Download

Die Vertrauenswürdigkeit der Kommunikation innerhalb der PKI wird nur dann uneingeschränkt von Ihrem System anerkannt, wenn Sie folgende Root- bzw. CA-Zertifikate der DFN-PKI in Ihren Browser geladen haben. Aktivieren Sie die Links durch jeweils einen Mausklick und bestätigen Sie die CAs als vertrauenswürdig:

Wurzelzertifikat der PKI des Deutschen Forschungsnetzes (DFN)
SHA1 Fingerprint = 12:63:41:60:D0:8C:FE:6A:87:6D:F7:86:D3:AD:C2:F7:74:FF:21:9F
MD5 Fingerprint = EF:08:E6:9F:6A:C7:25:2C:58:8C:55:FD:45:13:31:0A

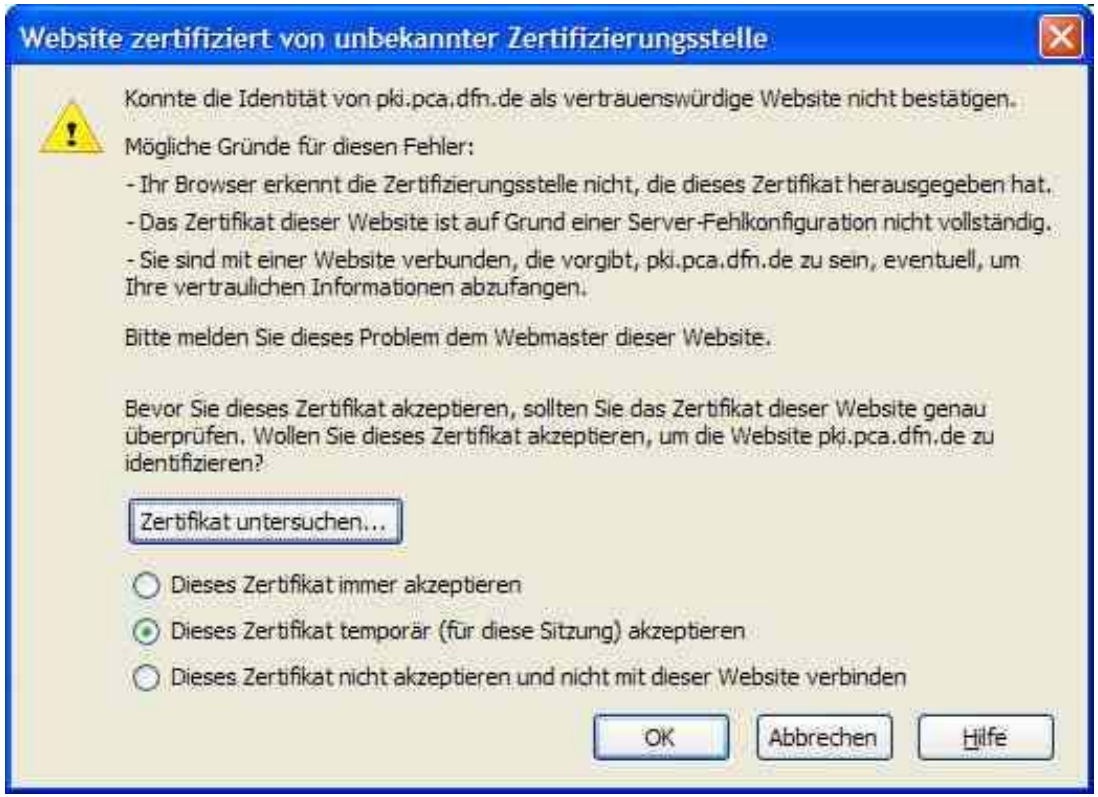
Zertifikat der Zertifizierungsstelle der Universitaet Hannover(UH-CA)
SHA1 Fingerprint = FE:1A:DC:29:58:E2:FD:9D:C5:FF:B5:65:5D:60:85:B2:6F:3A:9C:64
MD5 Fingerprint = 4C:12:14:13:14:81:48:79:D8:BA:25:75:67:63:5F:88

Die Fingerprints dienen zum Abgleich des geladenen Zertifikates mit den Angaben des Ausstellers. Je nachdem, wem Sie bei der Überprüfung dieser Fingerprints vertrauen wollen, verlassen Sie sich auf die Angaben oben, schauen auf der Homepage des Ausstellers nach oder, wenn Sie den ganzen Internet-Verkehr für unsicher halten, rufen Sie beim Aussteller an und lassen sich den Fingerprint durchgeben.

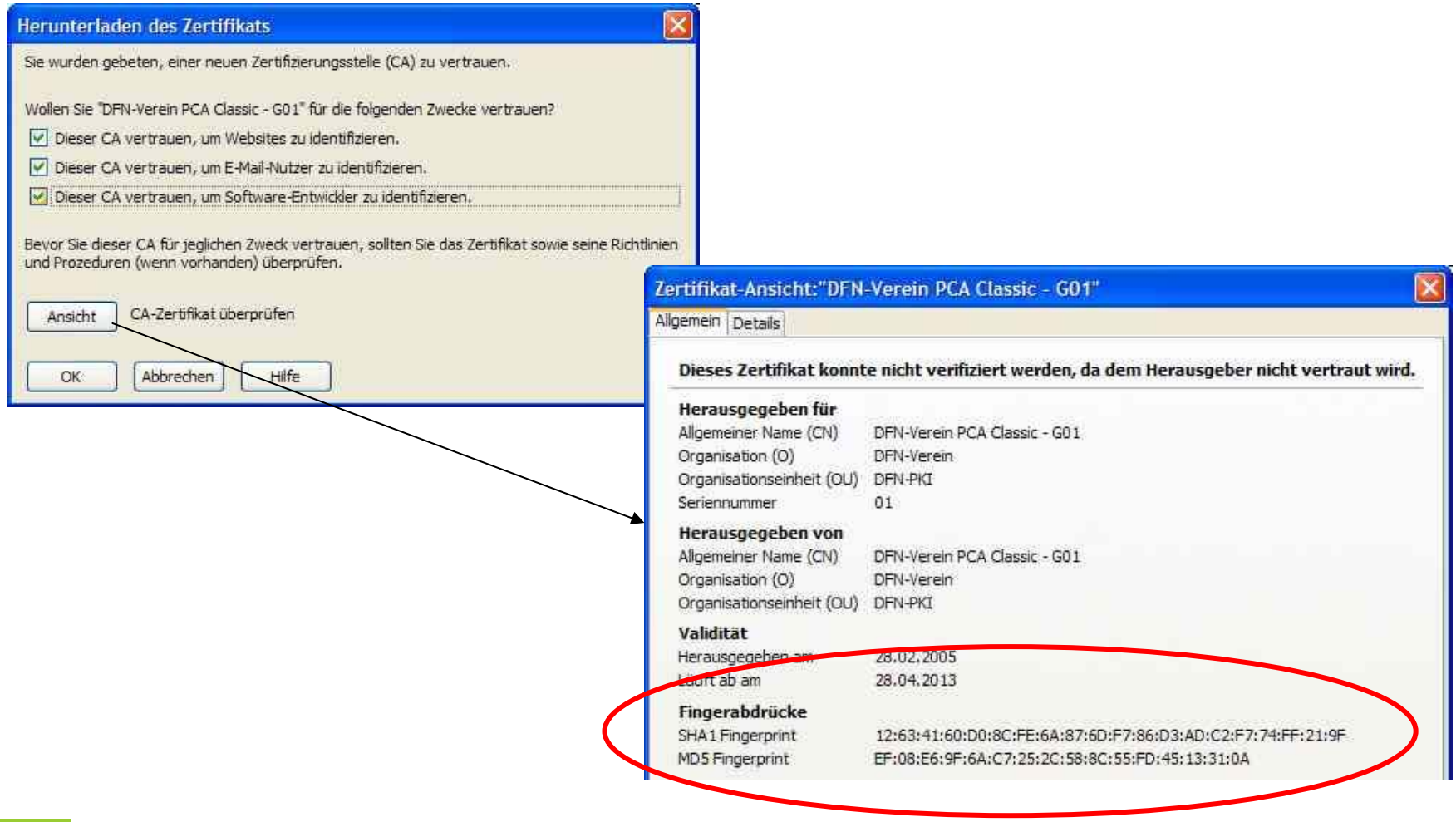
Vollständige Liste aller gängigen Zertifikatformate:

- DFN-Wurzelzertifikat
- UH-CA-Zertifikat

- **Zertifikate der Zertifizierungsinstanzen:**



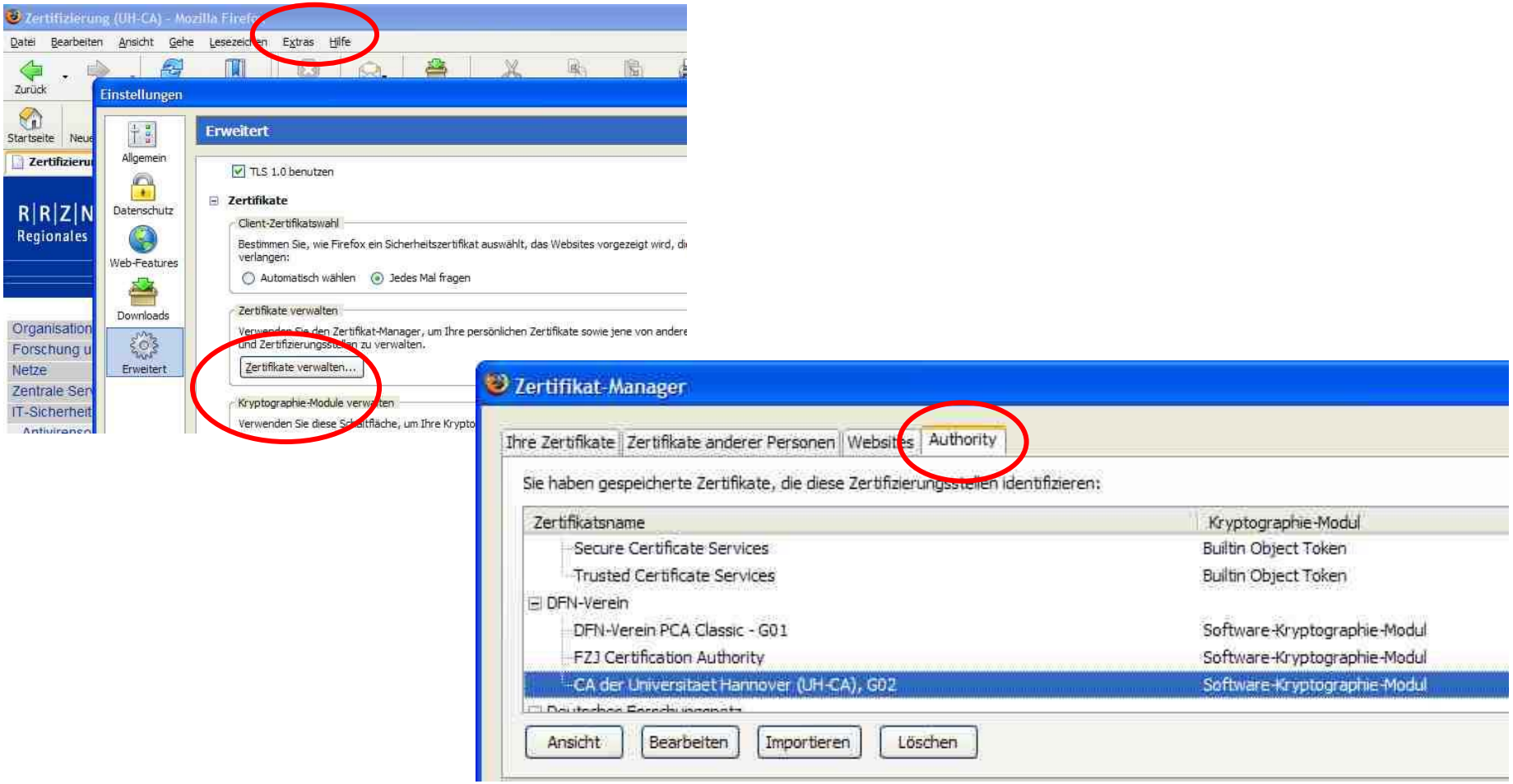
- **Zertifikate der Zertifizierungsinstanzen:**



„Klippen“

R|R|Z|N

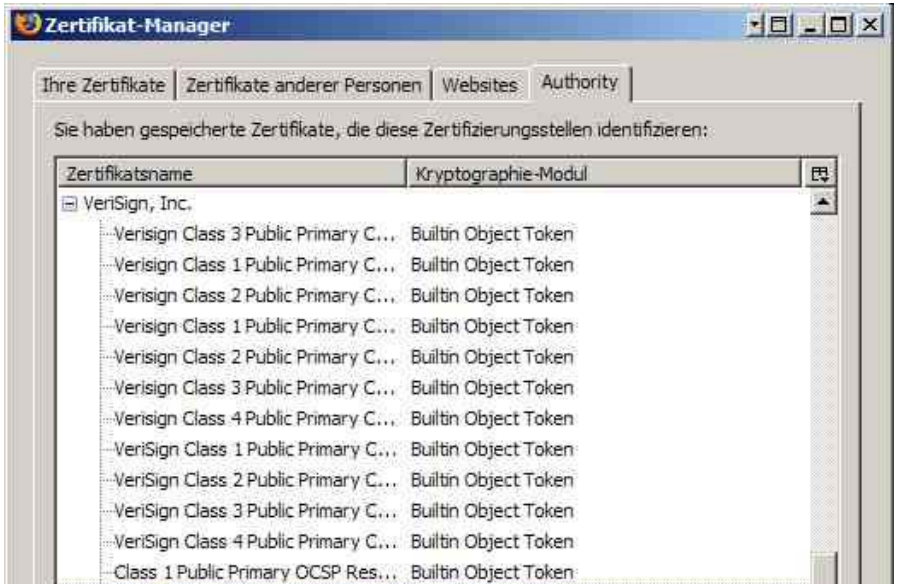
- **Zertifikate der Zertifizierungsinstanzen:**



„Klippen“

R|R|Z|N

- **Zertifikate der Zertifizierungsinstanzen:**



Wie sicher ist der geheime Schlüssel (private key)?

R|R|Z|N

- **Der private Schlüssel muss geschützt werden! Bei Kompromittierung ist das Zertifikat hinfällig!**
- **Mit einer Kopie des Schlüssels kann der Angreifer eine falsche Identität vortäuschen und vertrauliche Daten entschlüsseln. Missbrauch wird nicht zwingend erkannt!**
- **Anwendungen, die Zertifikate verwenden, speichern i.d.R. private Schlüssel (Software – PSE (Private Security Environment))**
  - Beispiel **Firefox, Thunderbird**: privater Schlüssel wird, durch Master-Passwort geschützt, in gesichertem Bereich gehalten (Keystore)
  - Beispiel **Windows**: Schlüssel wird in Registry-Schlüsselcontainer gehalten. Schutz durch Master-Passwort nur bei Wahl der hohen Sicherheitsstufe!
- **Für Anwendungen mit hohen Sicherheitsanforderungen reicht die Software-PSE möglicherweise nicht aus!**
  - Einsatz von USB-Token, Chipkarten

Literatur

- S. Singh, Geheime Botschaften, Hanser, 2000
- C. Adams, S. Lloyd, *Understanding PKI, 2nd Edition*, Addison Wesley, 2003
- A. Nash, W.Duane, C. Joseph, D. Brink, *PKI, e-security implementieren, Deutsche Ausgabe*, RSA, 2002
- A.J. Menezes, P.C. van Oorschot, S. A. Vanstone, *Handbook of Applied Cryptography, last updated October 4*, 2004, *http://www.cacr.math.uwaterloo.ca/hac/*