



Hacking Exposed

#4 TLS & Internet PKI

Pascal Knecht

Video 0: [Überblick](#)



- Dies ist eine Lehrveranstaltung.
- Die im Rahmen der Hacking-Exposed-Vorlesung vermittelten Kenntnisse sollen dazu beitragen, dass Sie Informationssicherheitsaspekte beachten und in Ihren Projekten berücksichtigen.
- Die HE-Vorlesung ist keineswegs als Anstiftung zum Hacken zu verstehen.

Inhalt heute Abend

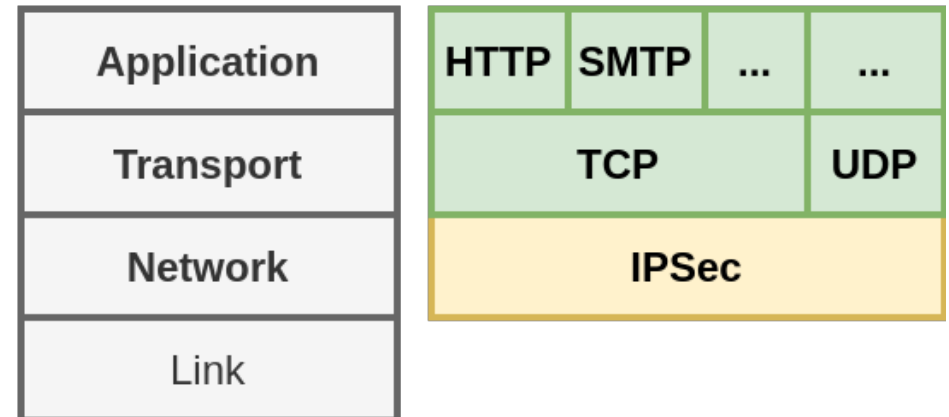
- TLS Grundlagen, Funktionsweise und Anwendungszwecke
- Internet Public Key Infrastructure

- Sie kennen die grundlegende Funktionsweise von TLS und können wichtige Komponenten eines TLS Handshakes benennen.
- Sie können TLS-Scanning beschreiben und kennen die Voraussetzungen, um dieses durchzuführen.
- Sie wissen, was eine Internet PKI ist und wie eine Vertrauenskette funktioniert und können deren Funktionsweise beschreiben.
- Sie können X.509 Zertifikate interpretieren und erstellen.

#01 Transportverschlüsselung

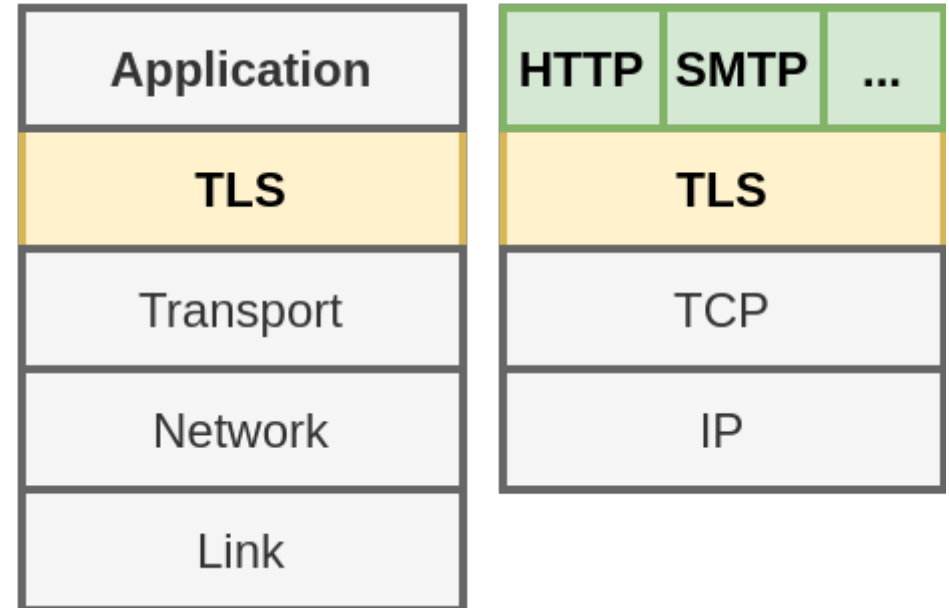
Typische Transportverschlüsselungen: IPSec

- Side-to-Side VPN
- Road-Warrior Szenario



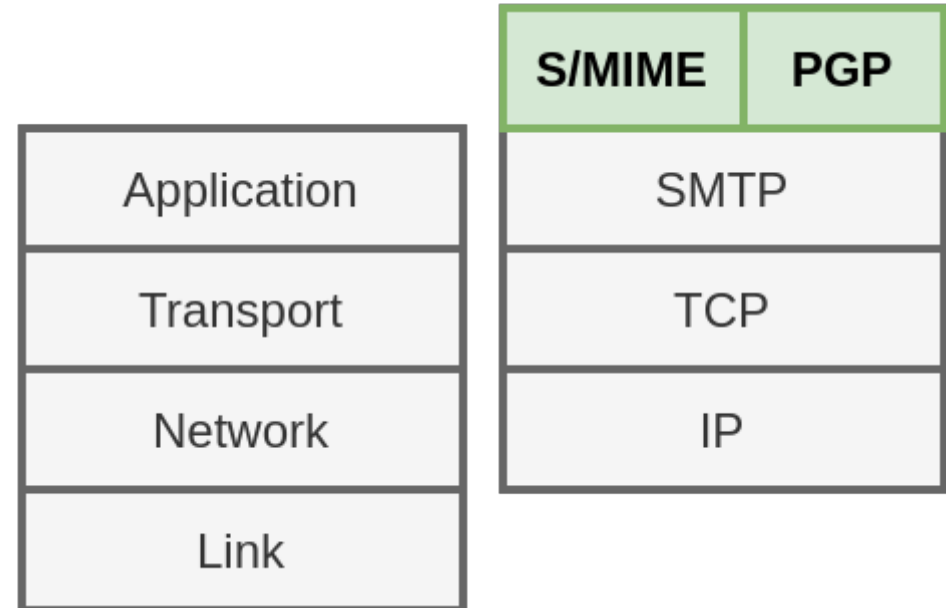
Typische Transportverschlüsselungen: TLS

- Protokollschicht auf TCP
- Application-Protokoll unabhängig / agnostisch



Inhaltsverschlüsselung

- Keine Transportverschlüsselung
- Email Verschlüsselung die sich auf den Inhalt (Payload) bezieht



#02 Transport Layer Security

Video 1: [TLS](#)

- Transportverschlüsselung bietet einen verschlüsselten Kommunikationskanal über ein nicht vertrauenswürdiges Netz.

OSI Schicht	Protokoll	Anwendung
Anwendung (Schicht 7)	SSH	Remote-Server Administration
Transport (Schicht 4)	TLS	Verschlüsselungsplattform für Anwendungsprotokolle
Internet (Schicht 3)	IPSec	VPN Verbindung

Transport Layer Security (TLS)

- Verschlüsselungsplattform für Protokolle der Anwendungsschicht
 - HTTPS, DoH, DoT, SMTPS, IMAPS, POP3S, XMPPS, IRCS, FTPS, EAP-TLS, OpenVPN

OSI-Schicht	Protokoll
Anwendung	HTTP, DNS, SMTP, IMAP, POP3, XMPP, IRC, FTP, EAP, OpenVPN
Transport	TLS
	TCP

Ziele von TLS

- Primäre Ziel ist die Bereitstellung eines sicheren Kanals zwischen zwei kommunizierenden Peers.
- Kryptographische Stärke
 - Starke verschlüsselte Verbindung zwischen zwei Kommunikationspunkten
- Interoperabilität
 - Unabhängige Entwickler entwickeln Programme und Bibliotheken die miteinander verschlüsselt kommunizieren können
- Erweiterbarkeit
 - Unabhängig von spezifischen kryptographischen Primitiven (z.B. Cipher oder Hashing-Funktion). Parameter können verändert werden ohne neues Protokoll erstellen zu müssen
- Effizienz
 - Kostenintensive kryptographische Operationen werden minimiert, u.a. dank Sessions

TLS Versionen

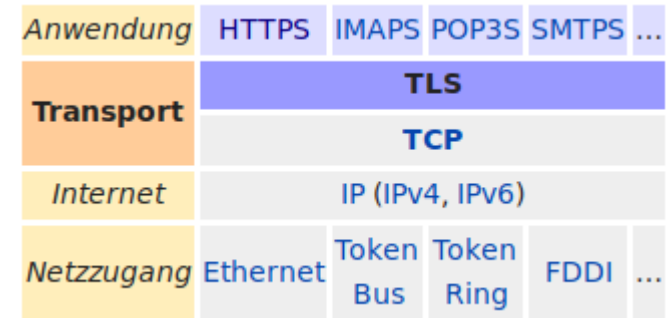
Version	Erscheinungsjahr
SSL 1.0	1994
SSL 2.0	1995
SSL 3.0	1996
TLS 1.0	1999
TLS 1.1	2006
TLS 1.2	2008
TLS 1.3	2018

Hauptunterschiede von TLS 1.2 zu TLS 1.3

- Alte symmetrische Verschlüsselungsalgorithmen entfernt
- Key Exchange und Authentication nicht mehr Teil der Cipher Suite
- Zero Round-Trip Time (0-RTT)
- Keine statische RSA und DH Cipher Suites → PK basierter Schlüsselaustausch mit Forward Secrecy
- Handshake Protokoll ab ServerHello verschlüsselt
- Einige weitere, siehe [RFC 8446](#)

TLS Übersicht

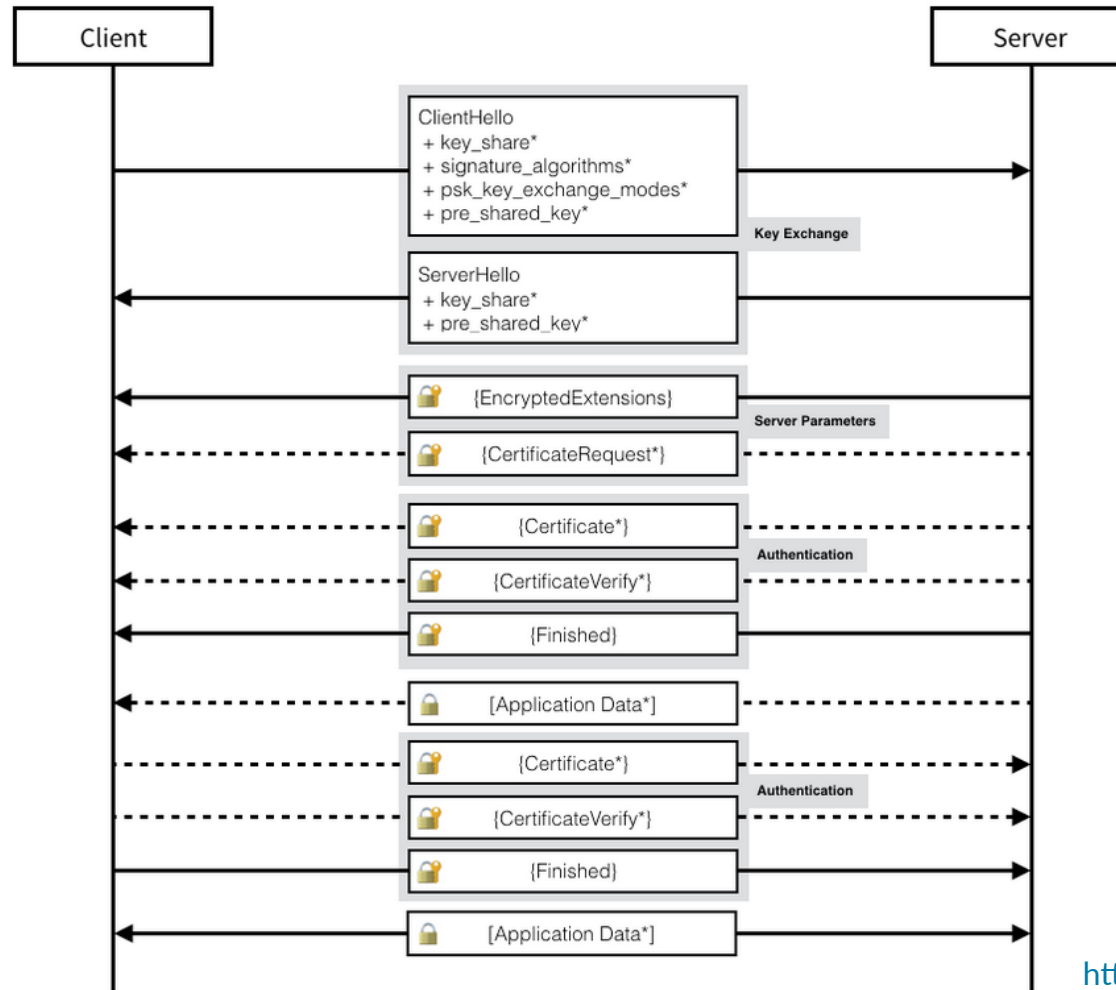
- Grundlage moderner Verschlüsselung bspw. HTTPS, IMAPS, SMTPS etc.
- TLS Versionen 1.2 und 1.3
 - SSL sowie TLS 1.0 und TLS 1.1 sind nicht zu verwenden
- Besteht aus den vier Protokollen
 - Record Protocol
 - Handshake Protocol
 - Application Data Protocol
 - Alert Protocol
- Interessante Historie: <https://www.feistyduck.com/ssl-tls-and-pki-history/>



TLS Handshake Protokoll

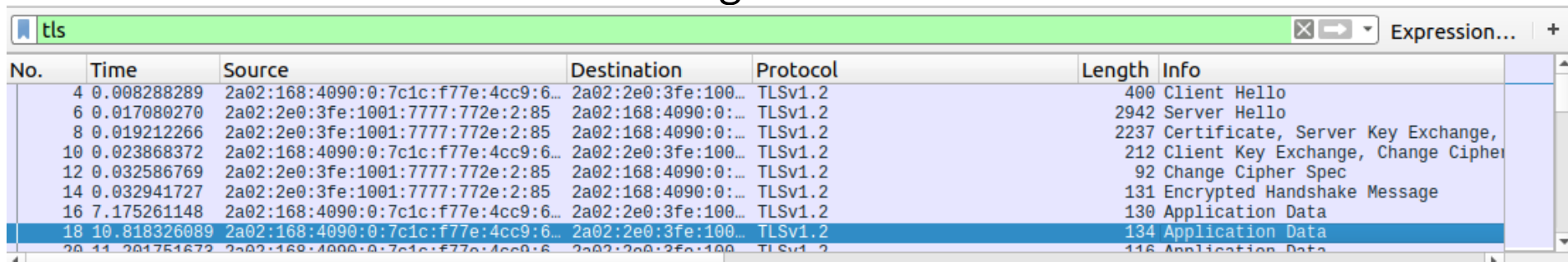
- Aufbau einer verschlüsselten Verbindung vom Client zum Server
- Handelt Parameter zwischen Client und Server aus:
 - TLS Version, kryptographische Parameter, Erweiterungen und Features etc.
- Ist der interessanteste Teil!

TLS 1.3 Handshake



TLS Handshake Ablauf in Praxis

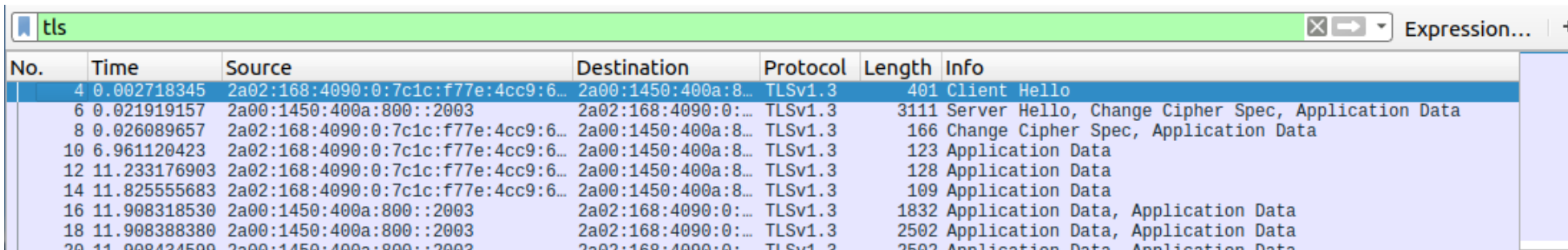
- Handshake einer TLS Verbindung in Version 1.2



A screenshot of a Wireshark packet capture showing a TLS 1.2 handshake. The filter is 'tls'. The packet list shows the following sequence:

No.	Time	Source	Destination	Protocol	Length	Info
4	0.008288289	2a02:168:4090:0:7c1c:f77e:4cc9:6...	2a02:2e0:3fe:100...	TLSv1.2	400	Client Hello
6	0.017080270	2a02:2e0:3fe:1001:7777:772e:2:85	2a02:168:4090:0:...	TLSv1.2	2942	Server Hello
8	0.019212266	2a02:2e0:3fe:1001:7777:772e:2:85	2a02:168:4090:0:...	TLSv1.2	2237	Certificate, Server Key Exchange,
10	0.023868372	2a02:168:4090:0:7c1c:f77e:4cc9:6...	2a02:2e0:3fe:100...	TLSv1.2	212	Client Key Exchange, Change Cipher
12	0.032586769	2a02:2e0:3fe:1001:7777:772e:2:85	2a02:168:4090:0:...	TLSv1.2	92	Change Cipher Spec
14	0.032941727	2a02:2e0:3fe:1001:7777:772e:2:85	2a02:168:4090:0:...	TLSv1.2	131	Encrypted Handshake Message
16	7.175261148	2a02:168:4090:0:7c1c:f77e:4cc9:6...	2a02:2e0:3fe:100...	TLSv1.2	130	Application Data
18	10.818326089	2a02:168:4090:0:7c1c:f77e:4cc9:6...	2a02:2e0:3fe:100...	TLSv1.2	134	Application Data
20	11.201751672	2a02:168:4090:0:7c1c:f77e:4cc9:6...	2a02:2e0:3fe:100...	TLSv1.2	116	Application Data

- Handshake einer TLS Verbindung in Version 1.3



A screenshot of a Wireshark packet capture showing a TLS 1.3 handshake. The filter is 'tls'. The packet list shows the following sequence:

No.	Time	Source	Destination	Protocol	Length	Info
4	0.002718345	2a02:168:4090:0:7c1c:f77e:4cc9:6...	2a00:1450:400a:8...	TLSv1.3	401	Client Hello
6	0.021919157	2a00:1450:400a:800::2003	2a02:168:4090:0:...	TLSv1.3	3111	Server Hello, Change Cipher Spec, Application Data
8	0.026089657	2a02:168:4090:0:7c1c:f77e:4cc9:6...	2a00:1450:400a:8...	TLSv1.3	166	Change Cipher Spec, Application Data
10	6.961120423	2a02:168:4090:0:7c1c:f77e:4cc9:6...	2a00:1450:400a:8...	TLSv1.3	123	Application Data
12	11.233176903	2a02:168:4090:0:7c1c:f77e:4cc9:6...	2a00:1450:400a:8...	TLSv1.3	128	Application Data
14	11.825555683	2a02:168:4090:0:7c1c:f77e:4cc9:6...	2a00:1450:400a:8...	TLSv1.3	109	Application Data
16	11.908318530	2a00:1450:400a:800::2003	2a02:168:4090:0:...	TLSv1.3	1832	Application Data, Application Data
18	11.908388380	2a00:1450:400a:800::2003	2a02:168:4090:0:...	TLSv1.3	2502	Application Data, Application Data
20	11.908424500	2a00:1450:400a:800::2003	2a02:168:4090:0:...	TLSv1.3	2502	Application Data, Application Data

TLS 1.3 Client Hello

Secure Sockets Layer

- ▼ TLSv1.3 Record Layer: Handshake Protocol: Client Hello
 - Content Type: Handshake (22)
 - Version: TLS 1.0 (0x0301)
 - Length: 311
- ▼ Handshake Protocol: Client Hello
 - Handshake Type: Client Hello (1)
 - Length: 307
 - Version: TLS 1.2 (0x0303)
 - Random: 574b7b10b9fcb49ed52b71d5579484b75e54d15b056bb4fd...
 - Session ID Length: 32
 - Session ID: 6d1ab1ef41ac3e13f53fad035e19b4623ff8d7a15537e317...
 - Cipher Suites Length: 62
 - ▶ Cipher Suites (31 suites)
 - Compression Methods Length: 1
 - ▶ Compression Methods (1 method)
 - Extensions Length: 172
 - ▶ Extension: server_name (len=19)
 - ▶ Extension: ec_point_formats (len=4)
 - ▶ Extension: supported_groups (len=12)
 - ▶ Extension: SessionTicket TLS (len=0)
 - ▶ Extension: encrypt_then_mac (len=0)
 - ▶ Extension: extended_master_secret (len=0)
 - ▶ Extension: signature_algorithms (len=48)
 - ▶ Extension: supported_versions (len=9)
 - ▶ Extension: psk_key_exchange_modes (len=2)
 - ▶ Extension: key_share (len=38)

TLS 1.3 Server Hello

```
Secure Sockets Layer
└─ TLSv1.3 Record Layer: Handshake Protocol: Server Hello
   Content Type: Handshake (22)
   Version: TLS 1.2 (0x0303)
   Length: 122
   └─ Handshake Protocol: Server Hello
      Handshake Type: Server Hello (2)
      Length: 118
      Version: TLS 1.2 (0x0303)
      Random: 9ffc61d49e4ebcd0b400fcfd15cf5698420b4d948b188d0a...
      Session ID Length: 32
      Session ID: 6d1ab1ef41ac3e13f53fad035e19b4623ff8d7a15537e317...
      Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
      Compression Method: null (0)
      Extensions Length: 46
      ┌─ Extension: key_share (len=36)
      └─ Extension: supported_versions (len=2)
   └─ TLSv1.3 Record Layer: Change Cipher Spec Protocol: Change Cipher Spec
      Content Type: Change Cipher Spec (20)
      Version: TLS 1.2 (0x0303)
      Length: 1
      Change Cipher Spec Message
```

Cipher Suite in TLS 1.2 und frühere

Authentication Algorithm Strength Mode

TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256

Key exchange Cipher MAC or PRF

The diagram illustrates the structure of the TLS cipher suite **TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256**. It is composed of four main parts, each with a functional label above or below it:

- ECDHE**: Labeled as **Key exchange** below.
- RSA**: Labeled as **Authentication** above.
- AES_128_GCM**: Labeled as **Cipher** below. This part is further detailed with labels above: **Algorithm** (for AES), **Strength** (for 128), and **Mode** (for GCM).
- SHA256**: Labeled as **MAC or PRF** below.

Cipher Suite ab TLS 1.3

Algorithm Strength Mode

TLS_AES_128_GCM_SHA256

Cipher HKDF Hash

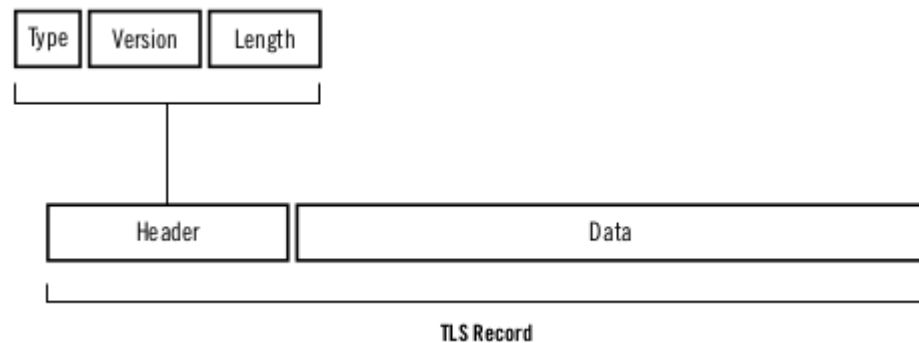
The diagram illustrates the naming convention for TLS cipher suites. It shows the example suite 'TLS_AES_128_GCM_SHA256' with brackets identifying its components: 'Algorithm' (AES), 'Strength' (128), and 'Mode' (GCM) are grouped under the 'Cipher' label; 'SHA256' is grouped under the 'HKDF Hash' label. The 'TLS_' prefix is also shown.

Cipher Suite in TLS 1.3

- Nur noch Cipher und Hash-Algorithmus
 - TLS_AES_128_GCM_SHA256
 - TLS_AES_256_GCM_SHA384
 - TLS_CHACHA20_POLY1305_SHA256
 - TLS_AES_128_CCM_SHA256
 - TLS_AES_128_CCM_8_SHA256
- Key Exchange und Authentisierung ist in Extensions ausgelagert

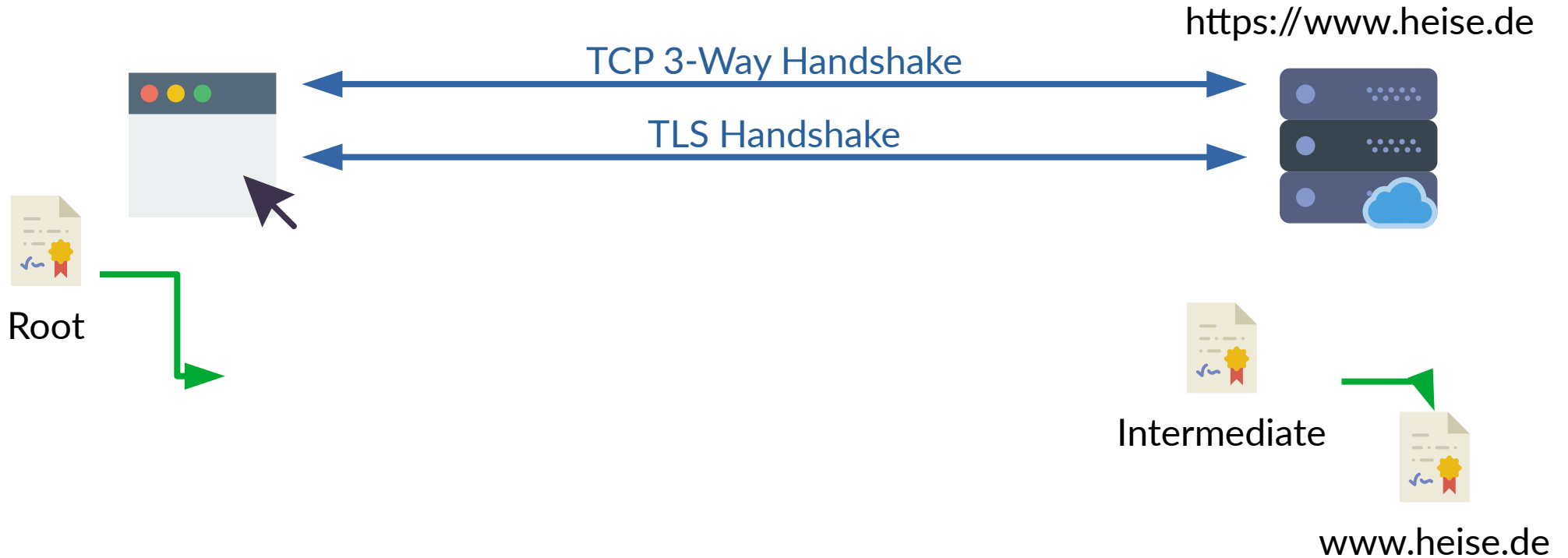
Record-, Application Data- & Alert Protocol

- **Record Protocol** ist der Rahmen der anderen drei Protokolle
 - «Containerschiff»

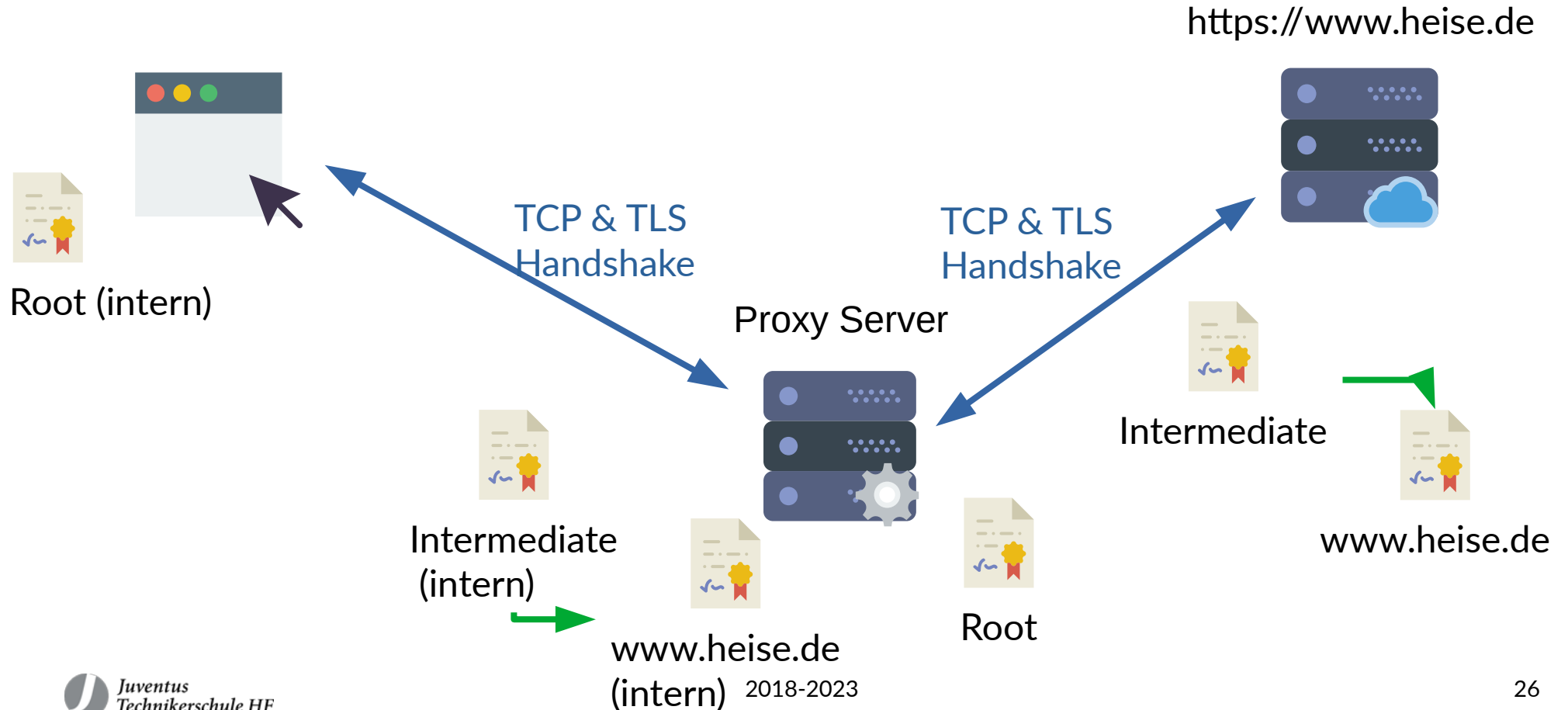


- **Application Data Protocol** beinhaltet verschlüsselte Daten
- **Alert Protocol** meldet Fehler
 - **fatal** → führt zum sofortigen Verbindungsabbruch bspw. **Decryption failed**
 - **warning** → kann Gegenstelle über Ereignis informieren bspw. **Unsupported extension**

TLS und X.509 Zertifikate



TLS-Scanning

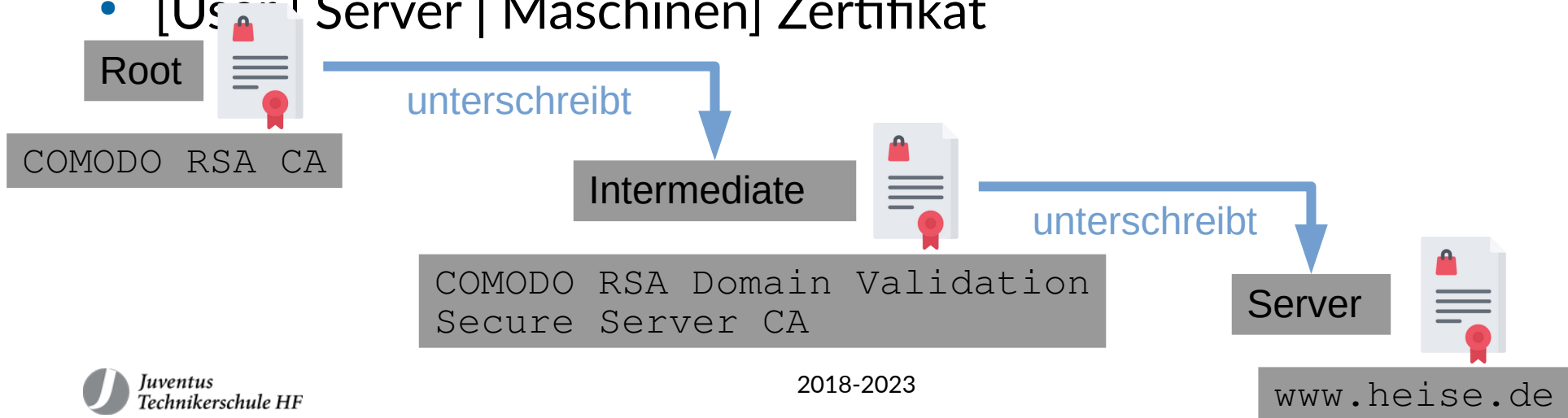


#03 Internet PKI

Video 2: [Internet PKI](#)

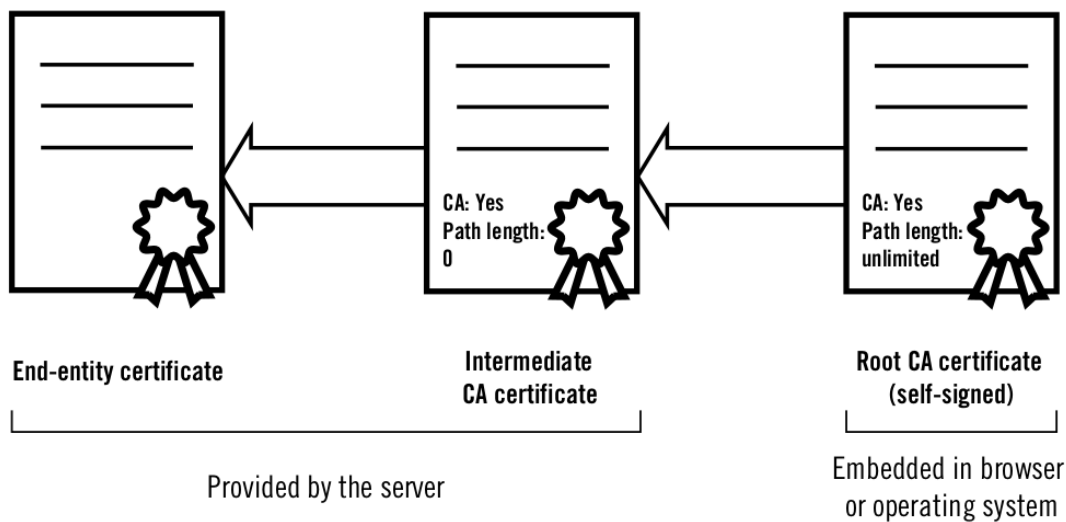
Chain of Trust

- Root Zertifikat ausgestellt von einer Certification Authority (CA)
- Intermediate Zertifikat 1 ausgestellt ebenfalls von der CA
- Intermediate Zertifikat n ausgestellt ebenfalls von der CA
- [Usual Server | Maschinen] Zertifikat



Chain of Trust im Browser

- Am Beispiel von www.heise.de



Certificate Viewer: "www.heise.de"

General Details

Certificate Hierarchy

- COMODO RSA Certification Authority
 - COMODO RSA Domain Validation Secure Server CA
- www.heise.de

Certificate Fields

- Validity
 - Not Before
 - Not After
- Subject
- Subject Public Key Info
 - Subject Public Key Algorithm
 - Subject's Public Key
- Extensions
 - Certificate Authority Key Identifier

Field Value

CN = www.heise.de
OU = Domain Control Validated

Certification Authorities (CA)

- Müssen strenge Policy befolgen
- Haben Root Zertifikate in Trusted Store von Browsern und OS
- Können Zertifikate für alle(!) Domains erstellen

Rank	Issuer	Usage	Market share
1	IdenTrust	20.4%	39.7%
2	Comodo	17.9%	34.9%
3	DigiCert	6.3%	12.3%
4	GoDaddy	3.7%	7.2%
5	GlobalSign	1.8%	3.5%
7	Certum	0.4%	0.7%
8	Actalis	0.2%	0.3%
9	Entrust	0.2%	0.3%
9	Secom	0.1%	0.3%
10	Let's Encrypt	0.1%	0.2%
11	Trustwave	0.1%	0.1%
12	WISeKey Group	< 0.1%	0.1%
13	StartCom	< 0.1%	0.1%
14	Network Solutions	< 0.1%	0.1%

- Eine Initiative von Mozilla und der Electronic Frontier Foundation (EFF)
- Start 2014, bis jetzt bereits über **200 Millionen aktive Zertifikate** ausgestellt
- Sind gratis und jeweils nur 90 Tage gültig
- Verlängerung einfach möglich

Date	Certificates issued
March 8, 2016	1 million ^[49]
April 21, 2016	2 million ^[50]
June 3, 2016	4 million ^[51]
June 22, 2016	5 million ^[52]
September 9, 2016	10 million ^[53]
November 27, 2016	20 million ^[54]
December 12, 2016	24 million ^[55]
June 28, 2017	100 million ^[56]
August 6, 2018	115 million ^[57]
September 14, 2018	380 million ^[58]
October 24, 2019	837 million ^[59]
February 27, 2020	1 billion ^[60]

Mai 2021 158 Millionen aktive Zertifikate

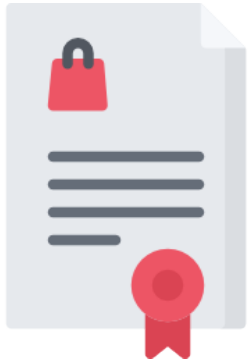
2018-2023

Certificate Trust Store

- Root Zertifikaten muss vertraut werden
 - Kommen in OS, Browsern und weiterer SW «vorinstalliert» mit
- Apple
 - IOS und OS X Plattform
- Chrome
 - Vertraut dem OS Root Store und führt zusätzliche Sicherheitsmechanismen ein
- Microsoft
 - MS Plattform und Produkte
- Mozilla

Der Standard X.509

- Wichtige Felder eines v3 X.509 Zertifikates: www.heise.de



Version: 3

Serial Number: 16:96:80:B7:7D:03:78:36:...

Signature Algorithm: SHA256

Issuer: COMODO RSA Domain Validation Secure Server CA

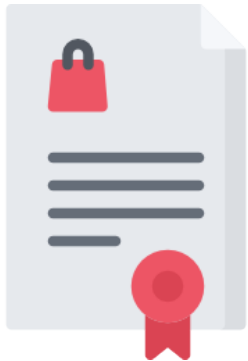
Validity: 8.1.2018 bis 8.4.2020

Subject: www.heise.de

Public Key: RSA 2048 Bit

X.509 Zertifikat v3 Extensions

- Wichtige Extension Felder eines v3 Zertifikates



Subject Alternative Name: `www.heise.de` und `heise.de`

Key Usage: `Signing & Key Encipherment`

Extended Key Usage: `wie Key Usage`

Signature Algorithm: `SHA256`

CRL Distribution Points: `http://crl.comodoca.com/COMODORSADo`

Zertifikatspaar erstellen

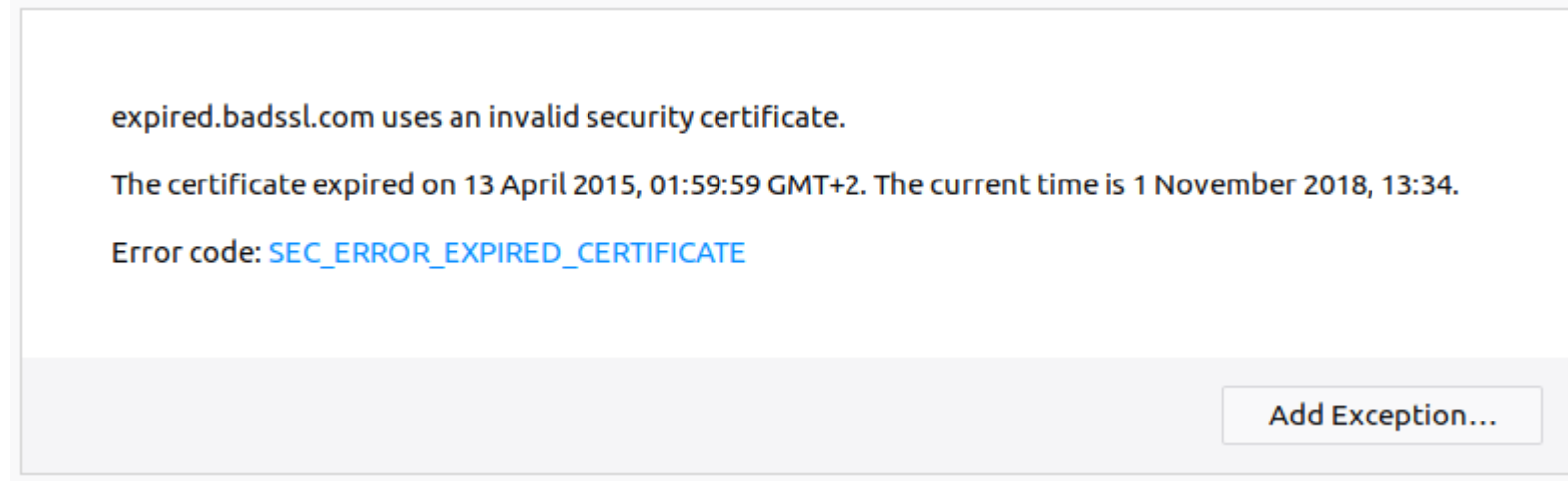
- Ein X.509 Zertifikat besteht immer aus einem Zertifikat und einem Private Key
 - Zertifikat = Meta Informationen + Public Key
- OpenSSL erlaubt die einfache Erzeugung von Zertifikaten:

```
$ openssl req -x509 -newkey rsa:2048 -keyout key.pem -out  
cert.pem -days 30 -nodes
```

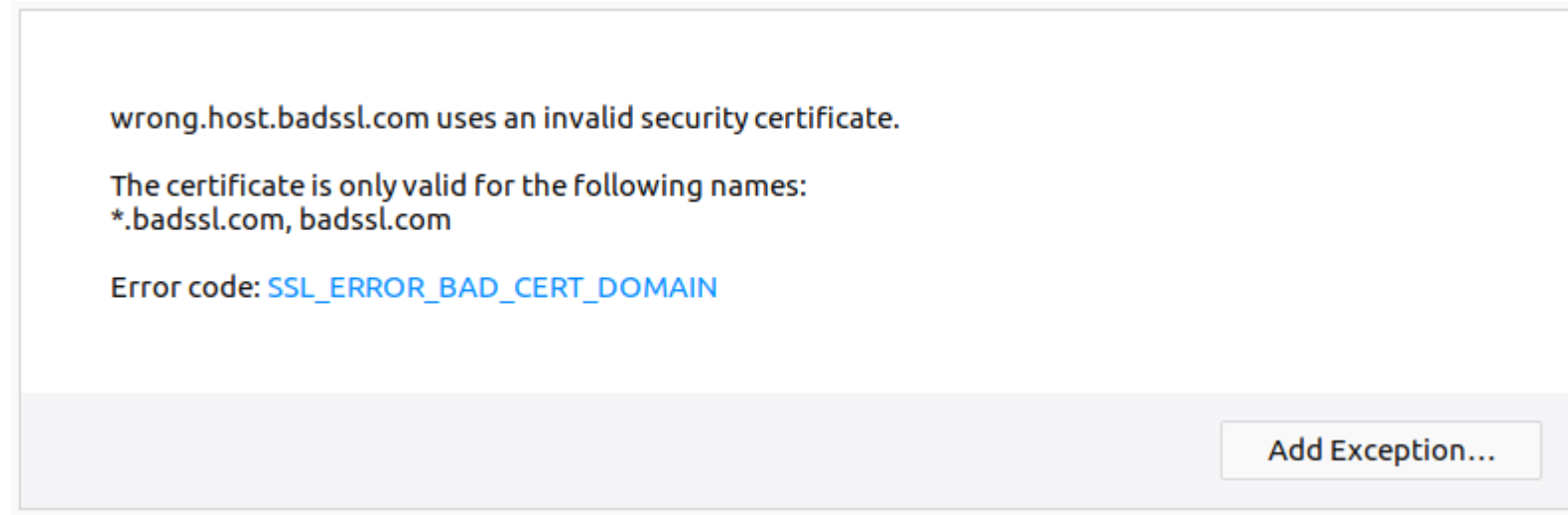
Zertifikat Lebenszyklus

- Wenn Sie ein offizielles Webserver Zertifikat benötigen:
 - Erstellen eines Certificate Signing Request (CSR)
 - Private Key bleibt immer bei Ihnen
 - `openssl req -nodes -new -newkey rsa:2048 -sha256 -out csr.pem`
 - CSR an CA senden → zwecks Unterschrift
 - CA validiert die Anfrage
 - Erfolgreich: Sendet Zertifikat zurück
 - Zertifikat mit Private Key einsatzbereit
 - Bis Ablauf oder Zurückziehung

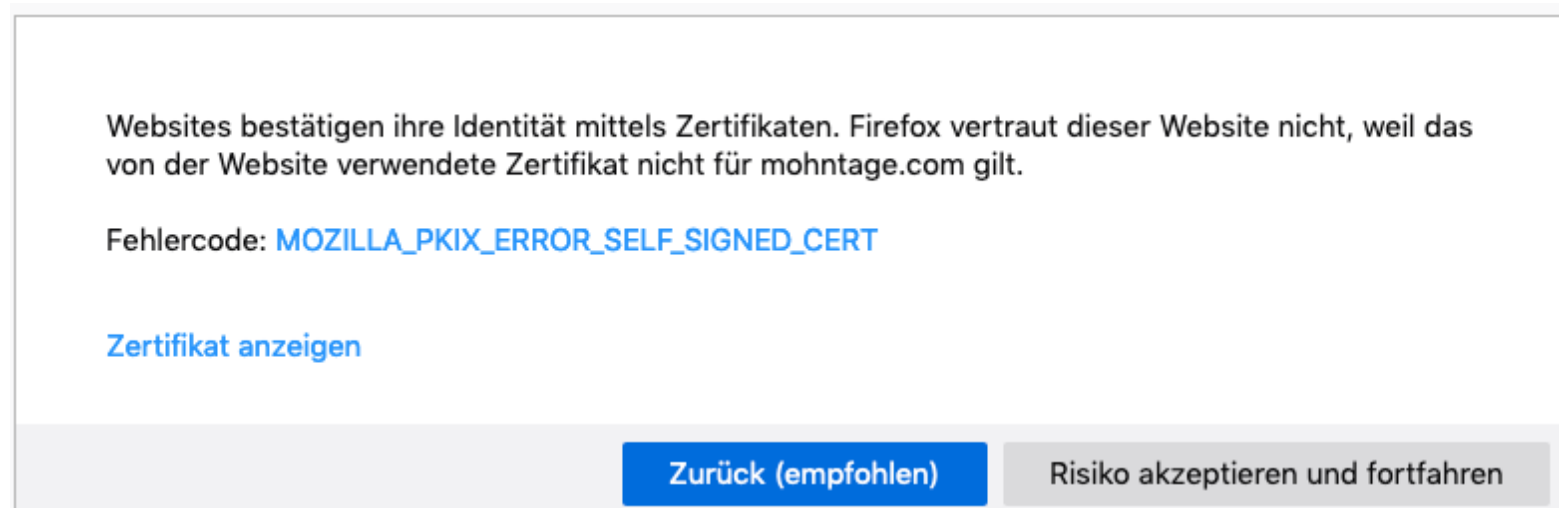
- Was passiert, wenn ein Zertifikat abgelaufen ist und nicht erneuert wurde?



- Was passiert, wenn ein Zertifikats Common Name nicht mit dem Servernamen übereinstimmt?



- Was passiert, wenn ein Browser das Vertrauen eines Zertifikates nicht herleiten kann?



Diverse Zertifikatsprobleme unter badssl.com
2018-2023

- Domain Validation (DV)
 - Erfolgt bei Feststellung des Besitzes der Domain via DNS, HTTP, ACME oder Email
- Extended Validation (EV)
 - Erfolgt durch zusätzliche Prüfung der anfragenden Entität: Hauptsächlich vertraglich / nicht technisch
 - Banken, Versicherungen, Unternehmen welche besonderes Augenmerk auf Security

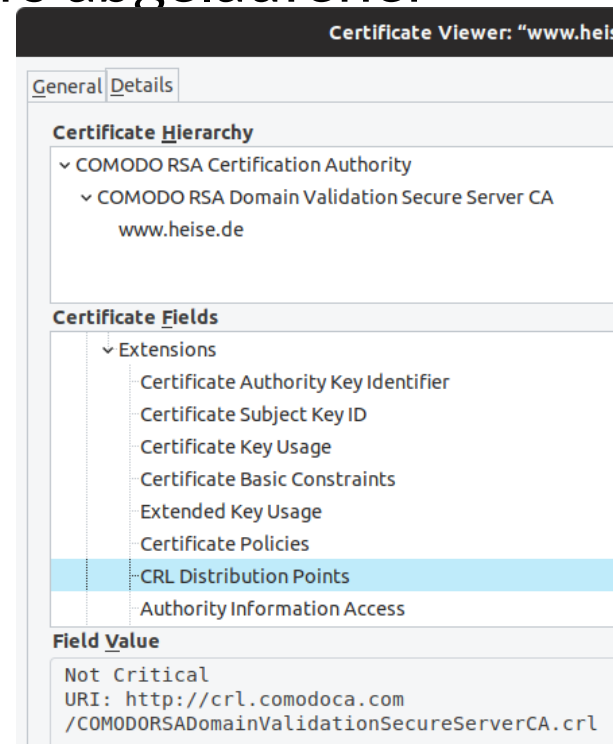
  PostFinance AG (CH) | <https://www.postfinance.ch/en/private.html>

Zurückziehung / Revocation

- Zertifikate können und sollen für ungültig erklärt werden wenn:
 - Private Key gestohlen/veröffentlicht wurde
 - Das Zertifikat nicht mehr benötigt wird
- Voraussetzung ist ein Revocation Certificate
 - Wird auf Basis des Private Keys erstellt
- Empfehlung: Erstellen Sie immer gleich ein Revocation Certificate, wenn Sie ein neues Zertifikat erzeugen
- Zwei Möglichkeiten
 - Certificate Revocation List (CRL)
 - Online Certificate Status Protocol (OCSP)

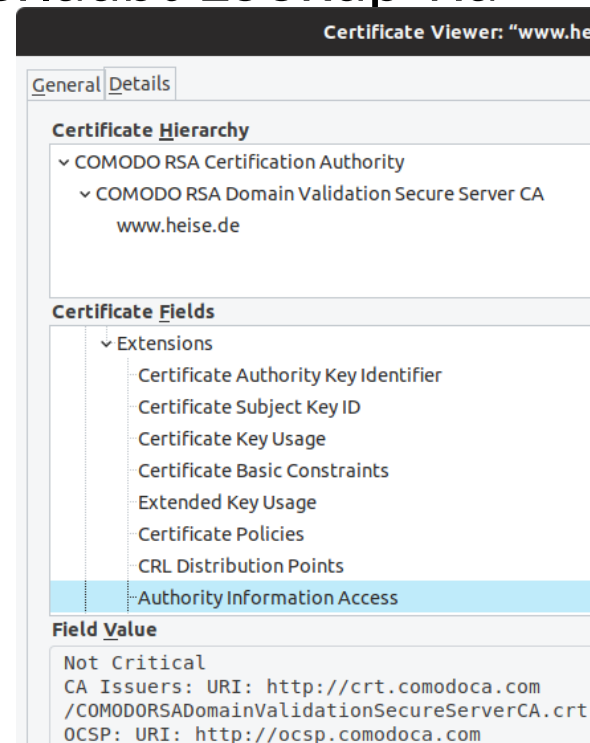
Certificate Revocation List (CRL)

- Von der CA gepflegte Liste von Serial Numbers abgelaufener Zertifikate
- URL im Zertifikat verankert
- Gross und langsam
 - Beispiel COMODO (heise.de) 3.4 MB

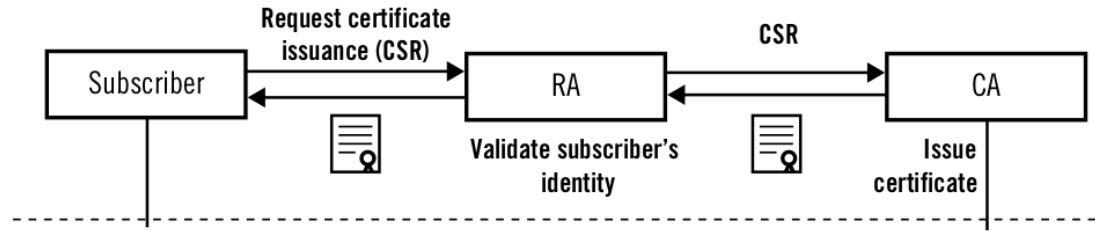


Online Certificate Status Protocol (OCSP)

- Von CA betriebene OCSP Responder Service erlaubt Lookup via API
- URL im Zertifikat verankert
- Löst das Problem mit CRL und bringt neue Probleme mit sich:
 - Performance und Privatsphäre
 - Lösung: OCSP Stapling



Internet PKI Zusammenspiel



Angriffe auf Internet PKI

- CAs sind einem grossen Risiko ausgesetzt
- Bereits angegriffen in der Vergangenheit:
 - VeriSign, Thawte, StartCom (2008, 2011), CertStar, RapidSSL, Comodo, DigiNotar, DigiCert, TURKTRUST, ...

Angriff auf PKI am Beispiel DigiNotar

Neuer SSL-Gau: Falsches Google-Zertifikat ist Folge eines Hacks
unentdeckt

CA-Hack: Noch mehr falsche Zertifikate
Ver
CA-Hack: Auch Anonymisierungs-Projekt TOR im Visier der Angreifer

Nach dem Angriff auf DigiNotar haben die
Über 500 Zertifikate: Ausmaß des CA-Hacks schlimmer als erwartet

Die Angreifer, die bei der niederländischen Zertifikatsbehörde DigiNotar
Niederländische Regierung übernimmt Kontrolle über DigiNotar

DigiNotar-Hack: Kritische Infrastruktur war unzureichend geschützt
niederländischen Staates kompromittiert. Um

DigiNotar hatte laut einem ersten Zwischenbericht der
DigiNotar-Hack: Auch Apple reagiert auf Zertifikatsklau

Aufsichtsbehörde untersagt DigiNotar das Ausstellen qualifizierter
Zertifikate
auch Apple auf die Kompromittierung des

Nachdem ein Hacker die Kontrolle über die
übernommen hatte, darf diese nun keine
DigiNotar wird liquidiert

Zertifikate muss DigiNotar für ungültig erklären
Der Eigentümer Vasco hat in den Niederlanden einen Insolvenzantrag für den Zertifikatsherausgeber
gestellt. Seit Bekanntwerden des CA-Hacks sind gerade einmal drei Wochen vergangen,

16.09.2011

21.09.2011

21.10.2011

21.10.2011

Übungen & Labor

Übungen: HE4

Labor: github.com/ryru/HackingExposed

Videoempfehlungen für's Selbststudium

- DNS mit DoT und DoH (34 Min)

