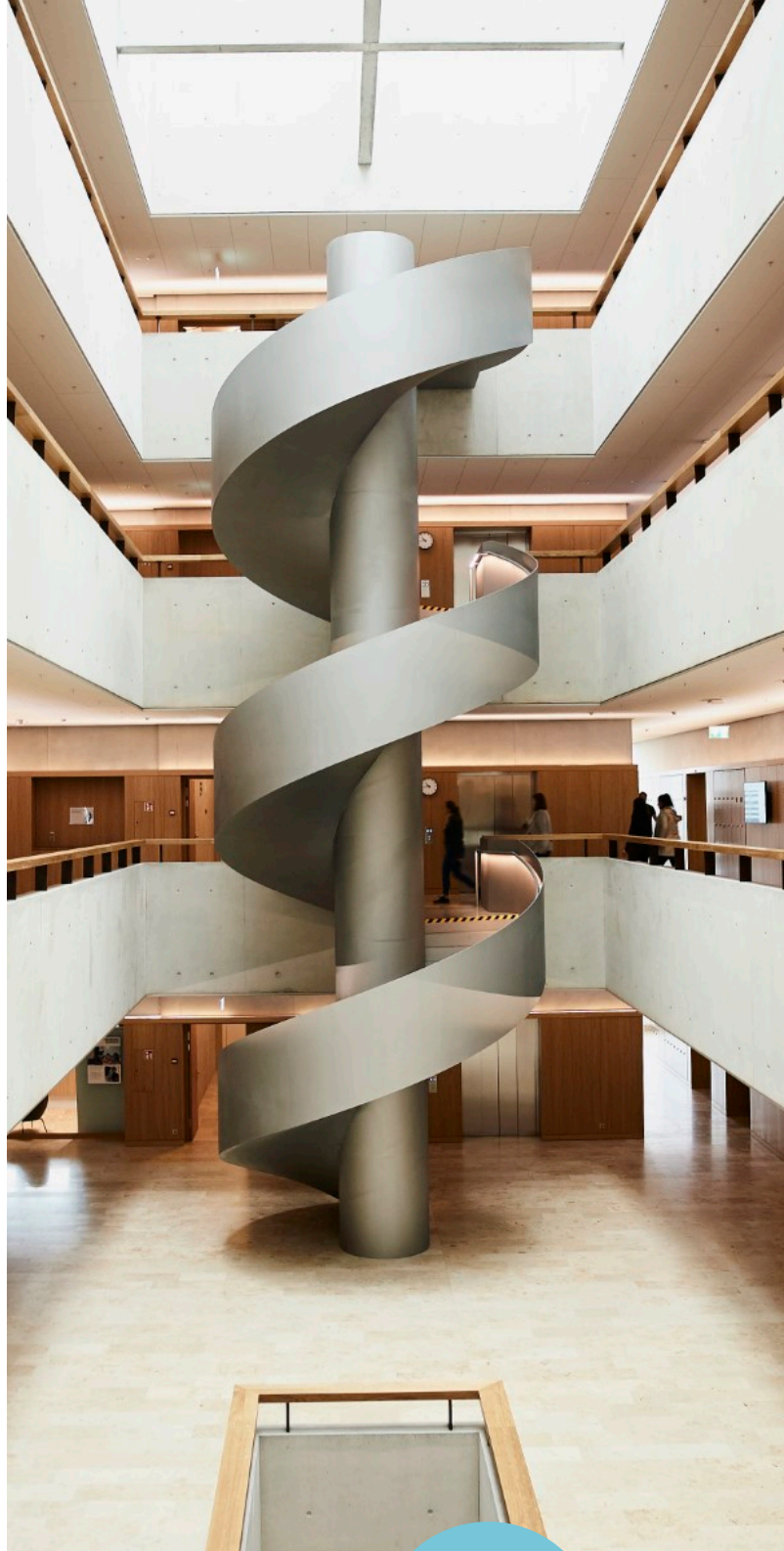


# Laborübung

## Public Key Infrastructure



Version  
1.0.17-2-  
g3dd8e56

## I. Allgemeine Informationen

Name:

Gruppe:

Bemerkungen:

### Liste der Verfasser

T. Jösler	Release 1.0
	Release 1.1
E. Sturzenegger	Release 1.2

### Copyright Informationen

Alle Rechte vorbehalten

## II. Inhaltsverzeichnis

<b>1. Vorbereitung</b>	<b>4</b>
1.1. Einleitung . . . . .	4
1.2. Hausaufgaben . . . . .	4
1.3. Benötigte Mittel . . . . .	6
<b>2. Certification Authority (CA) installieren &amp; konfigurieren</b>	<b>7</b>
<b>3. HTTPS aktivieren</b>	<b>16</b>
<b>4. Certificate Authentication</b>	<b>20</b>
4.1. Web Server konfigurieren . . . . .	20
4.2. User Certificate Autoenrollment . . . . .	21
4.3. Certificate Revocation . . . . .	23
<b>5. Code Signing</b>	<b>29</b>
5.1. Windows Scripting Standardverhalten feststellen . . . . .	29
5.2. Execution Policy anpassen . . . . .	31
5.3. Code Signing Zertifikate ausstellen . . . . .	32
5.4. Script signieren . . . . .	34
5.5. Revocation Problematik . . . . .	35
<b>6. SSL Interception</b>	<b>37</b>
6.1. mitmproxy . . . . .	37
6.2. Windows Client vorbereiten . . . . .	39
6.3. Testing . . . . .	39
6.4. Eigene Certificate Authority einbinden . . . . .	39
6.5. Testing zum Zweiten . . . . .	40
6.6. Funktionsanalyse . . . . .	40
6.7. Aufräumen . . . . .	42
6.8. Schlusswort . . . . .	42
<b>7. Anhang</b>	<b>43</b>
Anhang A – Ablauf Signieren . . . . .	43
Anhang B – Zertifikatsbasierte, gegenseitige Authentifizierung . . . . .	43

### III. Vorwort

#### Feedback

Mit Ihrer Mithilfe kann die Qualität des Versuches laufend den Bedürfnissen angepasst und verbessert werden.

Falls in diesem Versuchsablauf etwas nicht so funktioniert wie es beschrieben ist, melden Sie dies bitte direkt dem Laborpersonal oder erwähnen Sie es in Ihrem Laborbericht oder Protokoll. Behandeln Sie die zur Verfügung gestellten Geräte mit der entsprechenden Umsicht.

Bei Problemen wenden Sie sich bitte ebenfalls an das Laborpersonal.

#### Legende

In den Versuchen gibt es Passagen, die mit den folgenden Boxen markiert sind. Diese sind wie folgt zu verstehen:

##### Wichtig

Dringend beachten. Was hier steht, unbedingt merken oder ausführen.

##### Aufgabe III.1

Beantworten und dokumentieren Sie die Antworten im Laborprotokoll.

##### Hinweis

Ergänzender Hinweis / Notiz / Hilfestellung.

##### Information

Weiterführende Informationen. Dies sind Informationen, die nicht zur Ausführung der Versuche benötigt werden, aber bekannt sein sollten.

##### Story

Hierbei wird die Geschichte vermittelt, die in den Versuch einleitet oder den Zweck des Versuches vorstellt.

##### Zielsetzung

Lernziele, die nach dem Bearbeiten des Kapitels erfüllt sein sollten.

##### Erkenntnis

Wichtige Erkenntnisse, die aus dem Versuch mitgenommen werden sollten.

## 1. Vorbereitung

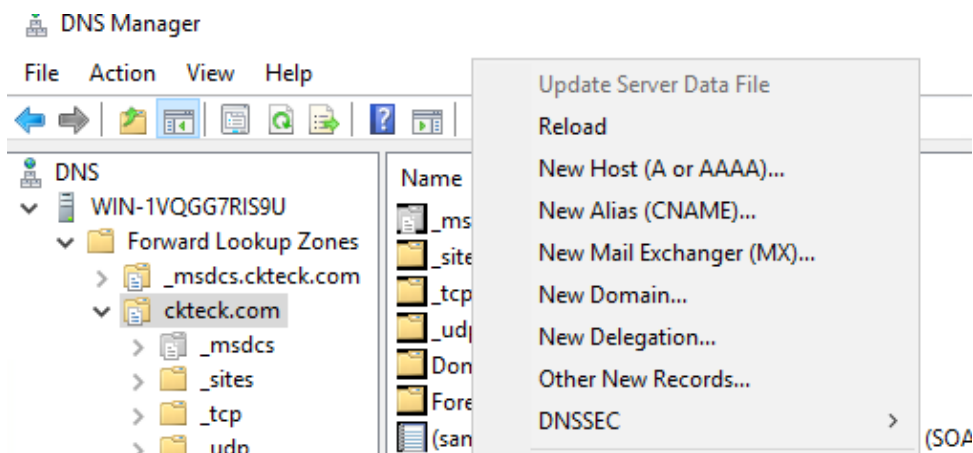
In diesem Kapitel werden vorbereitende Aktionen durchgeführt, damit die Übung funktioniert.

Loggen Sie sich auf Ihrem Domain Controller ein. Nutzen Sie den Account labadmin.

Öffnen Sie den DNS Manager. Sie finden diesen in den Windows Administrative Tools.

Erstellen Sie einen A-Record in der Forward Lookup Zone "**g01.ckteck.com**". Dabei entspricht der Teil **g01** Ihrer Gruppennummer. Navigieren Sie dazu durch den Baum auf der linken Seite des Fensters und erstellen Sie einen neuen Host Eintrag mit Hilfe des Kontextmenüs der rechten Seite des Fensters.

Wählen Sie als Name "www". Der Rest Ihrer Domäne wird automatisch ergänzt. Bringen Sie die private IPv4 Adresse Ihres **Linux Clients** in Erfahrung, z.B. mit `ping -4 islab-lc-XX`. Tragen Sie diese ein und erstellen Sie den A-Record. Nun kann Ihr Linux Client / Web Server auch mittels DNS-Namen angesprochen werden. Dies werden Sie im Kapitel "HTTPS aktivieren" brauchen.



### 1.1. Einleitung

Diese Laborübung soll den Studierenden den praktischen Umgang mit einer Public Key Infrastructure näherbringen. Dabei werden die Studierenden das praktisch Umgesetzte immer wieder mit den gelernten theoretischen Grundlagen verifizieren.

### 1.2. Hausaufgaben

Dieses Kapitel beschreibt Vorbereitungsmaßnahmen, die vor Beginn der Übung durchzuführen sind.

#### 1.2.1. Theorie

Lesen Sie Kapitel 1 des Buches Jörg Schwenk, Sicherheit und Kryptographie im Internet: Theorie und Praxis; 4., überarbeitete und erweiterte Auflage.

Aus dem Hochschul-Netzwerk können Sie das Buch gratis als PDF herunterladen: [https://link.springer.com/chapter/10.1007/978-3-658-06544-7\\_1](https://link.springer.com/chapter/10.1007/978-3-658-06544-7_1)

### 1.2.2. Fragen zur Theorie

Beantworten Sie die folgenden Fragen und notieren Sie Ihre Antworten. Sie finden die nötigen Informationen, um die Fragen zu beantworten im oben erwähnten Buch, im Anhang oder auch im grossen bösen Internet.

#### Aufgabe 1.1

Wie unterscheiden sich symmetrische und asymmetrische Verschlüsselung? Nennen Sie für beide mindestens je zwei Eigenschaften.

#### Aufgabe 1.2

Was für Authentifizierungsmechanismen existieren? Erklären Sie die jeweilige Funktionsweise jeweils mit eins bis zwei Sätzen.

#### Aufgabe 1.3

Wie ist ein handelsübliches X.509 Zertifikat aufgebaut? Was für Felder kommen vor? Beschreiben Sie!

#### Aufgabe 1.4

Wie funktioniert das Signieren einer Nachricht? Erklären Sie welche Komponenten beim Sender sowie beim Empfänger welche Rolle spielen. (siehe Anhang A – Ablauf Signieren)

### Aufgabe 1.5

Was gibt es für Möglichkeiten, ein Zertifikat für ungültig zu erklären?

### Zielsetzung

Der vorliegende Laborversuch deckt primär die Themen rund um das Themengebiet PKI ab. Dabei wird jedoch auf das Themengebiet "verschlüsselte E-Mails" verzichtet. Dieses Themengebiet ist bereits Lerngegenstand anderer Module.

Während in einem ersten Teil eine Certification Authority installiert und konfiguriert, sowie eine Webseite HTTPS-fähig gemacht wird, steht im zweiten Teil der Übung die Authentifizierung von Usern und die Verifikation von Dateien mittels Zertifikats im Mittelpunkt. Im letzten Teil der Übung wird aufgezeigt, wie einfach SSL Interception betrieben werden kann und was dies für Auswirkungen auf das vermeintlich sichere Surfen über HTTPS hat.

## 1.3. Benötigte Mittel

Sie benötigen die ISLAB Laborumgebung. Diese befindet sich auf SWITCHengines und Sie können sich Remote per Remote Desktop damit verbinden.

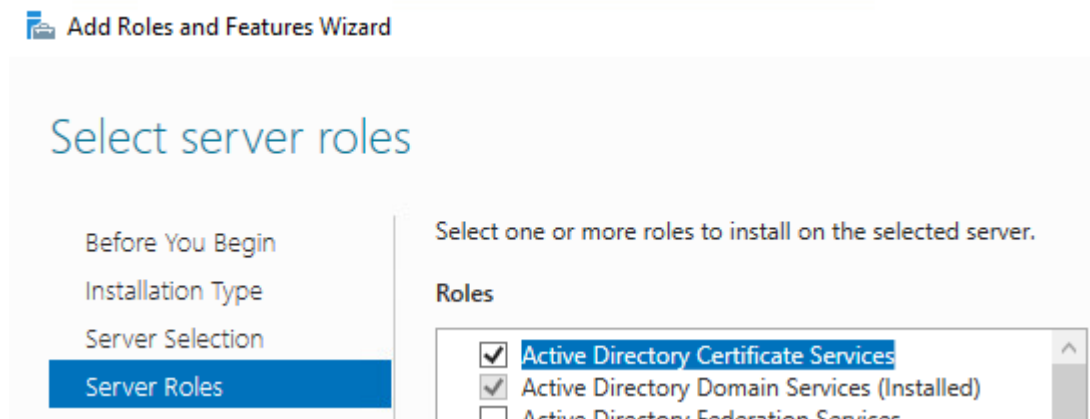
- Windows Server
- Windows Client
- Linux Client

### Wichtig

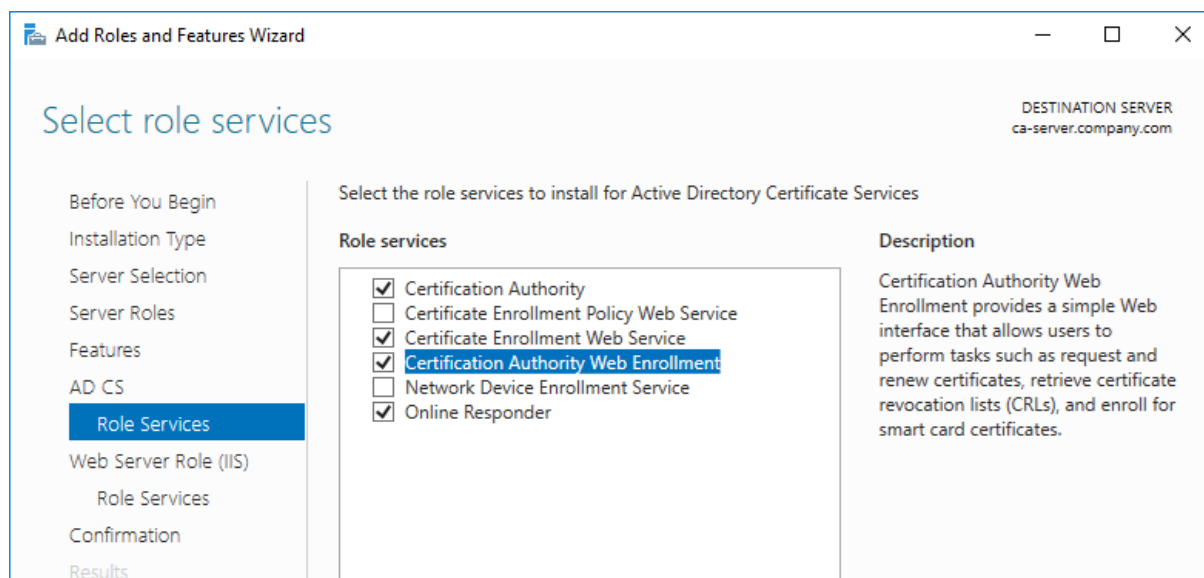
Diese Durchführung wird auf neuer Infrastruktur durchgeführt, somit kann es zu Abweichungen kommen. Vor und während dem Bearbeiten der Laborübungen bitte Informationen des Laborpersonals beachten und den Discord-Kanal im Auge behalten. Das Laborpersonal wird mitteilen, sollte trotz der Tests etwas nicht wie in diesem Dokument beschrieben funktionieren.

## 2. Certification Authority (CA) installieren & konfigurieren

Zuerst muss die Rolle "Active Directory Certificate Services" installiert werden. Über den Server Manager kann der "Add Roles and Features" Wizard gestartet werden. Wählen Sie die Rolle "Active Directory Certificate Services".



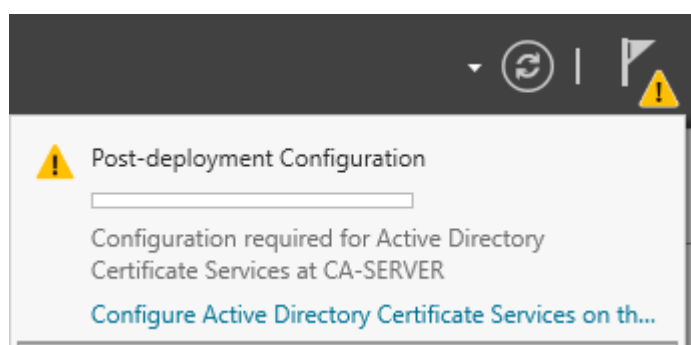
Von den zur Verfügung stehenden AD CS Role Services werden die folgenden benötigt:



Ansonsten sind keine Änderungen an den Standardeinstellungen notwendig. Die Installation kann einige Zeit dauern. Gönnen Sie sich einen circa 5-minütigen Kaffee.

Starten Sie Ihren Server neu, falls ein Neustart verlangt wird. Es kann einige Minuten dauern, bis alle Änderungen vorgenommen und Ihr Server neugestartet ist. Haben Sie Geduld.

Nach der erfolgreichen Installation müssen die Services noch via Server Manager konfiguriert werden. Der Dialog dafür befindet sich im Server Manager und sieht folgendermassen aus.





Begonnen wird mit dem Role Service "Certification Authority". Die anderen Role Services werden in dieser Übung nach und nach konfiguriert.

### Select Role Services to configure

- ☒ Certification Authority
- ☐ Certification Authority Web Enrollment
- ☐ Online Responder
- ☐ Network Device Enrollment Service
- ☐ Certificate Enrollment Web Service
- ☐ Certificate Enrollment Policy Web Service

Es soll eine Enterprise CA installiert werden. Diese sind Mitglieder einer Domäne und können Domänenmitglieder mit diversen Zertifikaten ausrüsten. Da Ihr Windows Server zeitgleich als Domain Controller fungiert, ist dieser bereits Teil der Domäne. Ansonsten müsste Ihr Server zuerst in Ihre Domäne aufgenommen werden.

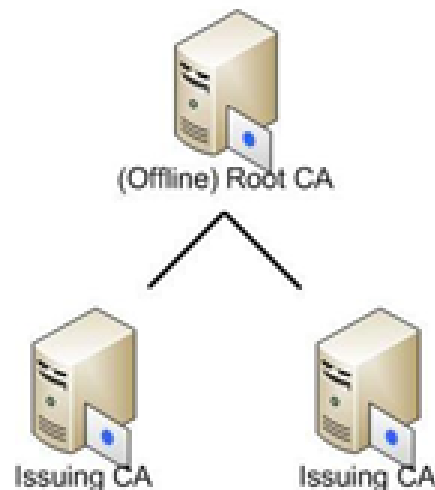
### Specify the setup type of the CA

Enterprise certification authorities (CAs) can use Active Directory Domain Services (AD DS) to simplify the management of certificates. Standalone CAs do not use AD DS to issue or manage certificates.

- ☒ Enterprise CA  
Enterprise CAs must be domain members and are typically online to issue certificates or certificate policies.
- ☐ Standalone CA  
Standalone CAs can be members or a workgroup or domain. Standalone CAs do not require AD DS and can be used without a network connection (offline).

Grundsätzlich gibt es verschiedene Architekturen, wie eine PKI Infrastruktur betrieben werden kann. Dies reicht von einer Single Tier bis Three Tier Hierarchie. Mehr Informationen zum Design von PKI Infrastrukturen finden sich auf folgender Webseite: <https://techcommunity.microsoft.com/t5/ask-the-directory-services-team/designing-and-implementing-a-pki-part-i-design-and-planning/ba-p/396953>

Der Einfachheit halber verwenden wir in diesem Versuch eine Two Tier Hierarchie. In diesem Kapitel werden wir eine Issuing oder auch Subordinate CA installieren. Diese bezieht ihr Zertifikat von einer Root CA. Diese wurde bereits vom Laborteam installiert. Normalerweise wird die Root CA nur für das Ausstellen der Zertifikate der Issuing CAs benötigt. Zur übrigen Zeit wird die Root CA aus Sicherheitsgründen ausgeschaltet.



**Abbildung 1:** Two Tier PKI Hierarchie (Quelle: <https://techcommunity.microsoft.com/t5/ask-the-directory-services-team/designing-and-implementing-a-pki-part-i-design-and-planning/ba-p/396953> )

### Aufgabe 2.1

Warum sollte die Root CA die meiste Zeit ausgeschaltet, respektive offline sein? Was gibt es hier für Sicherheitsbedenken? Begründen Sie!

Wählen Sie nun den Radio Button "Subordinate CA" aus.

#### Specify the type of the CA

When you install Active Directory Certificate Services (AD CS), you are creating or extending a public key infrastructure (PKI) hierarchy. A root CA is at the top of the PKI hierarchy and issues its own self-signed certificate. A subordinate CA receives a certificate from the CA above it in the PKI hierarchy.

☐ Root CA

Root CAs are the first and may be the only CAs configured in a PKI hierarchy.

☒ Subordinate CA

Subordinate CAs require an established PKI hierarchy and are authorized to issue certificates by the CA above them in the hierarchy.

Erstellen Sie einen neuen Private Key. Wählen Sie eine Schlüssellänge von **4096** Bits. Als Algorithmus belassen Sie SHA256.

## Private Key

- Credentials
- Role Services
- Setup Type
- CA Type
- Private Key**
- Cryptography
- CA Name
- Certificate Request
- Certificate Database
- Confirmation

DESTINATION SERVER  
islab-ws-dev-01.g01dev.ckteck.com

### Specify the type of the private key

To generate and issue certificates to clients, a certification authority (CA) must have a private key.

☒ **Create a new private key**  
Use this option if you do not have a private key or want to create a new private key.

☐ **Use existing private key**  
Use this option to ensure continuity with previously issued certificates when reinstalling a CA.

☐ **Select a certificate and use its associated private key**  
Select this option if you have an existing certificate on this computer or if you want to import a certificate and use its associated private key.

Wählen Sie einen sprechenden Namen für Ihre neue CA.

## CA Name

- Credentials
- Role Services
- Setup Type
- CA Type
- Private Key
- CA Name**
- Certificate Request
- Certificate Database
- Confirmation
- Progress

DESTINATION SERVER  
WIN-1VQGG7RIS9U.ckteck.com

### Specify the name of the CA

Type a common name to identify this certification authority (CA). This name is added to all certificates issued by the CA. Distinguished name suffix values are automatically generated but can be modified.

Common name for this CA:

Distinguished name suffix:

Preview of distinguished name:

Generieren Sie als nächstes einen Certificate Signing Request und speichern Sie ihn lokal ab. Der Rest der Installation kann mit den Standardeinstellungen beendet werden.

Der Results-Tab gibt Auskunft über die nächsten Schritte, die Sie zur Konfiguration unternehmen müssen. Die Meldung, ob weitere Role Services konfiguriert werden sollen, kann mit "No" beantwortet werden. Die verbleibenden Role Services werden später konfiguriert.

The following roles, role services, or features were configured:

**Active Directory Certificate Services**

---

**Certification Authority**  **Configuration succeeded with warnings**

The Active Directory Certificate Services installation is incomplete. To complete the installation, use the request file "C:\WIN-1VQGG7RIS9U.ckteck.com\_ckteck-WIN-1VQGG7RIS9U-CA.req" to obtain a certificate from the parent CA. Then, use the Certification Authority snap-in to install the certificate. To complete this procedure, right-click the node with the name of the CA, and then click Install CA Certificate. The operation completed successfully. 0x0 (WIN32: 0)

[More about CA Configuration](#)

Im Vorfeld dieser Übung wurde vom Assistenzteam bereits eine Root-CA aufgesetzt, von welcher Sie nun ihr Subordinate Certificate lösen können. Navigieren Sie dazu zur folgender URL

<http://islab-ckteck-ca.zh.switchengines.ch/certsrv/>.

Laden Sie als erstes das Zertifikat der Root CA herunter. Sie finden es unter "Download a CA certificate...". **Laden Sie es in BASE64-codierter Version herunter!** Das Zertifikat werden wir später benötigen. Benennen Sie die Datei so, dass Sie später noch wissen welches Zertifikat diese Datei beinhaltet.

Gehen Sie nun zurück auf die Ausgangsseite und navigieren Sie folgendermassen:

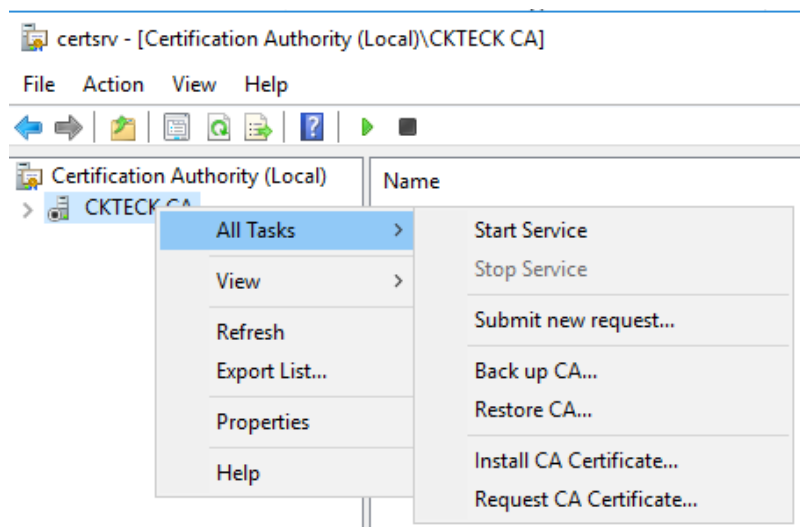
request a certificate -> advanced certification request -> submit a request

Kopieren Sie nun den Inhalt der zuvor generierten **Certificate Signing Request** in das Saved-Request Feld. Theoretisch müssten Sie sich nun auf der Root CA einloggen und den Certificate Request bestätigen. Da aber für alle Kursteilnehmenden nur eine Root CA existiert, wäre das ziemlich chaotisch. Darum wurde diese Root CA so konfiguriert, dass sie Zertifikate automatisch ausstellt. Dies ist sicherheitstechnisch eine üble Sache und sollte ausserhalb dieser Übungsumgebung niemals so konfiguriert werden!

Laden Sie nun das für Sie bereitgestellte **Certificate Chain im BASE64-encodierten Format** herunter.

Als nächstes muss das "Certification Authority" Konfigurationsfenster geöffnet werden. Sie finden dieses im Startmenü unter "Windows Administrative Tools".

Installieren Sie die heruntergeladene Certificate Chain mittels Kontextmenü der Certification Authority -> All Tasks -> Install CA Certificate. Wählen Sie die zuvor beantragte und heruntergeladene Certificate Chain aus.



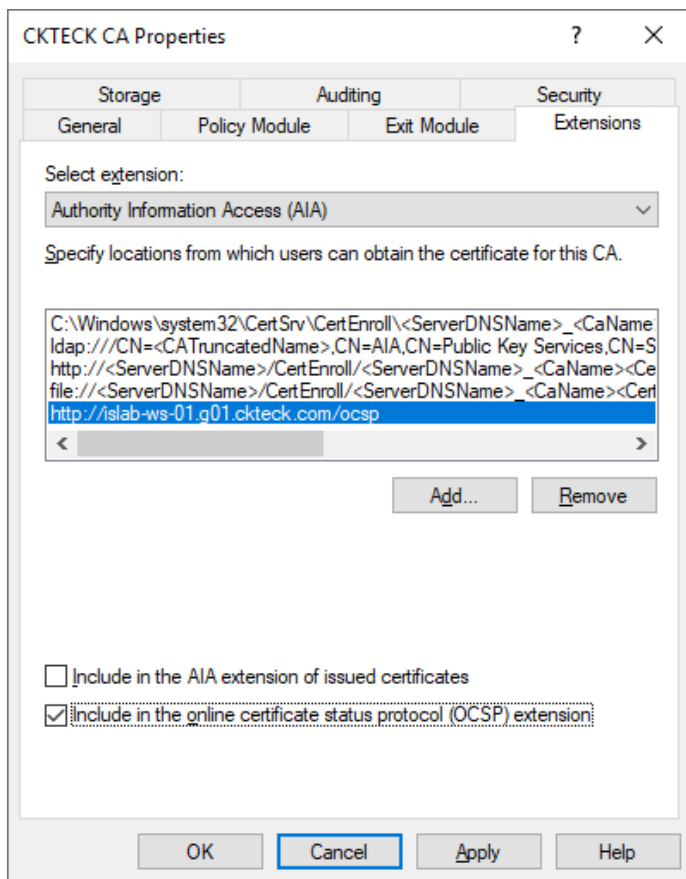
Der nächste Schritt ist von **hoher Wichtigkeit**. Vergewissern Sie sich, dass Sie keine Schreibfehler oder ähnliches gemacht haben! Ansonsten werden im späteren Verlauf der Übung **üble Probleme** auftreten und Sie werden die Übung fast von vorne beginnen müssen!

### Wichtig

Es folgt eine Vorbereitungsarbeit für das später in Angriff genommene Feature "OCSP". Öffnen Sie dazu die Properties der CA per Rechtsklick-Kontextmenü.

Wechseln Sie in den Extensions Tab und fügen Sie in der "**Authority Information Access**" Extension den folgenden Eintrag **auf Ihre Domäne und Hostnamen angepasst** hinzu. **Der einzutragende Host ist Ihr Windows Server / CA-Server!**

Aktivieren Sie zudem die Option "**Include in the online certificate status protocol (OCSP) extension**"



**KONTROLLIEREN SIE NOCHMALS, DASS SIE HOSTNAME UND GRUPPENNUMMER OBEN ANGEPASST HABEN!**

**http://islabs-ws-XX.gXX.ckteck.com/ocsp**

Gönnen Sie sich die zusätzliche Freizeit, indem Sie keine Flüchtigkeitsfehler begehen und darum die Übung nicht nochmals **von vorne beginnen** müssen!

Wird bei der Eingabe der URL ein Fehler gemacht, kann im späteren Verlauf der Übung der OCSP Online Responder nicht gefunden werden. Im Nachhinein kann diese Option **NICHT** mehr angepasst werden.

Nun kann die CA gestartet werden. Dies geschieht über das gleiche Kontextmenü wie auch schon die Installation des CA Zertifikats.

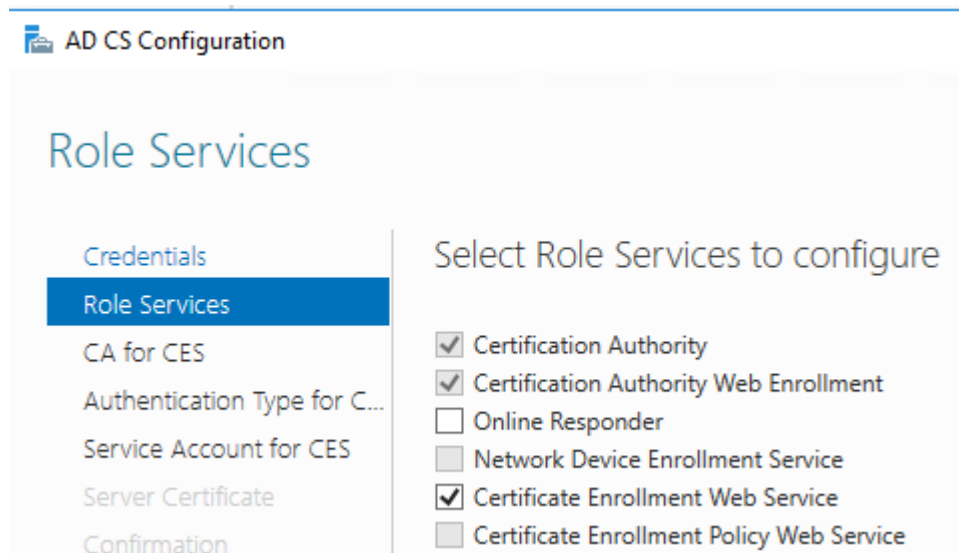
Als nächstes wird der Role Service "Certification Authority Web Enrollment" konfiguriert. Der Konfigurationsdialog kann auch wieder über den Server Manager aufgerufen werden.

Anpassungen an der Default-Konfiguration sind keine möglich.

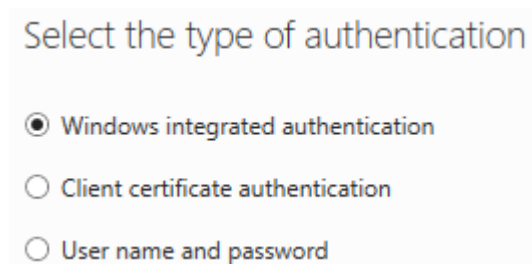
### Select Role Services to configure

- ☒ Certification Authority
- ☒ Certification Authority Web Enrollment
- ☐ Online Responder
- ☐ Network Device Enrollment Service
- ☐ Certificate Enrollment Web Service
- ☐ Certificate Enrollment Policy Web Service

Als nächstes wird der Role Service "Certificate Enrollment Web Service" konfiguriert.



Als nächstes muss die Authentifizierungsart bestimmt werden. Wir verwenden die integrierte Windows Authentifizierung.

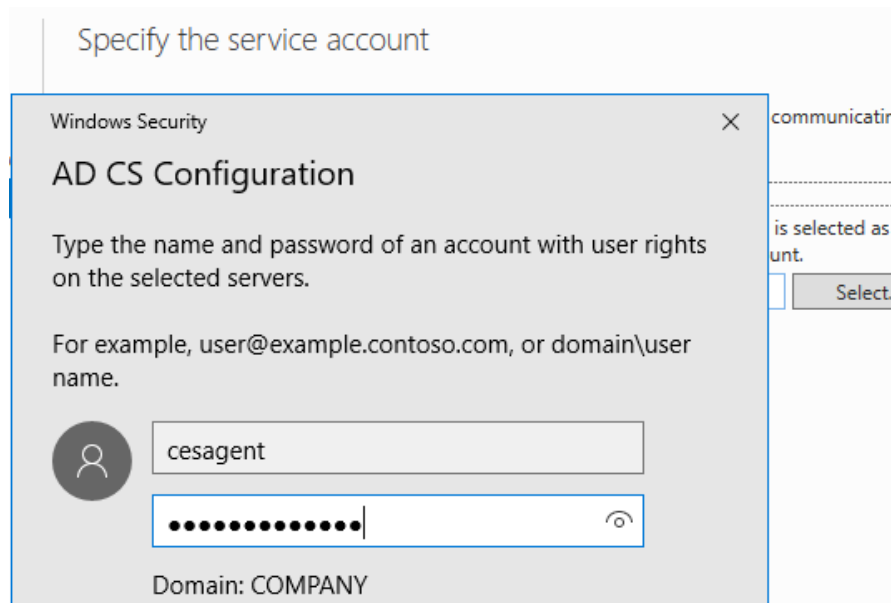


Der Webservice kann entweder mit einem Service User oder einem integrierten Account betrieben werden. Wir entscheiden uns einen Service User zu erstellen.

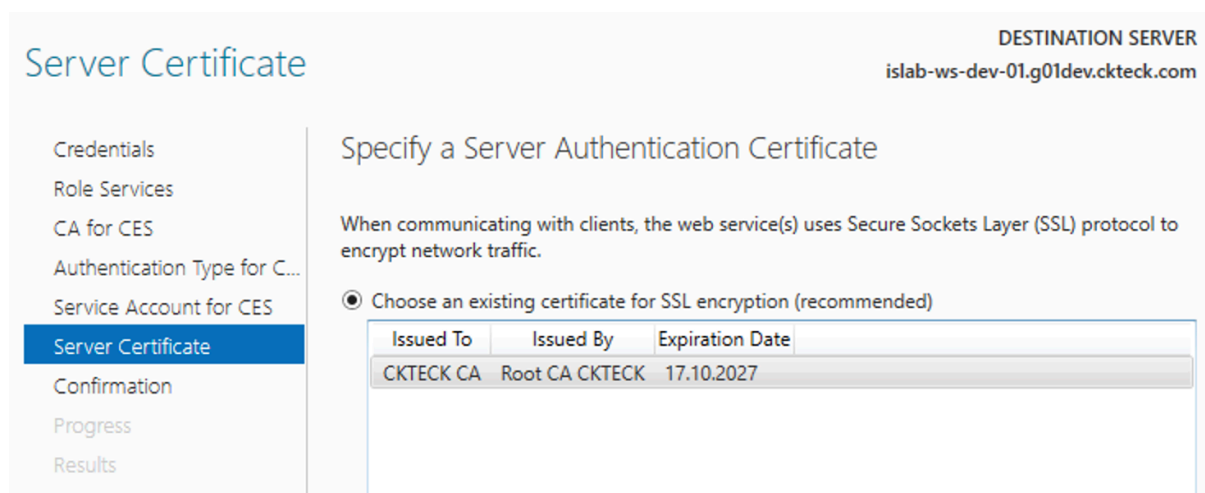
Wechseln Sie ins Active Directory Users and Computers und erstellen Sie einen User namens "cesagent" in der für Service Accounts vorgesehenen OU. Der User muss Mitglied der Gruppe "IIS\_IUSRS" und lokaler Administrator sein. Verwenden Sie die Gruppe "Domain Admins" dazu. Dies ist zwar alles andere als korrekte Vergabe von Rechten, jedoch sparen wir uns die Zeit eine lokale Administratoren-Gruppe zu erstellen und lokal einzutragen.

Beachten Sie beim Erstellen, dass der Benutzer sein Passwort nicht beim nächsten Einloggen ändern muss. Das Passwort soll auch nicht ablaufen.

Wählen Sie den Account im Dialog aus und geben Sie das Passwort für den Account ein.

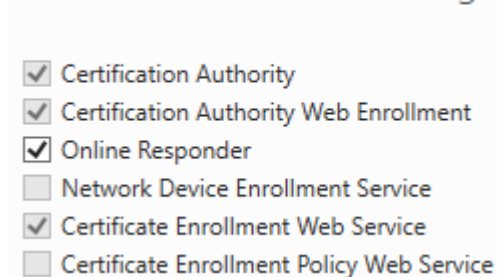


Als nächstes muss ein Zertifikat für den Web Service ausgewählt werden. Wir besitzen bereits eines, welches bei der Installation der CA generiert wurde. Dieses kann auch für SSL verwendet werden. Wählen Sie das bestehende Zertifikat aus.



Als letzter Role Service wird der Online Responder konfiguriert. Dabei ist wieder keine spezielle Konfiguration nötig. Achten Sie darauf, dass Sie den Konfigurationsdialog abschliessen, da der Role Service ansonsten nicht konfiguriert wird und später Probleme bei der Revocation auftreten.

## Select Role Services to configure



Nun ist die Subordinate oder auch Intermediate Certification Authority installiert. Die Root CA, welche unser Subordinate CA Certificate ausgestellt hat, wird im Offline-Modus betrieben. Heisst, der Server ist nicht an die Domäne angebunden. Dies hat zur Folge, dass die Domäne und ihre Mitglieder nichts über das Root Certificate der Subordinate CA wissen.

Damit die Certificate Chain sauber abgebildet werden kann, muss nun das Root CA Certificate dem Active Directory bekanntgemacht werden. Öffnen Sie dazu eine Command prompt **als Administrator** und setzen Sie folgenden Befehl ab. Verweisen Sie dabei auf das zuvor heruntergeladene Root CA Certificate. Je nachdem wie Sie die Datei benannt und wo Sie es abgespeichert haben, verändert sich natürlich die Pfadangabe im Command.

```
1 certutil -dspublish -f C:\rootcacert.cer RootCA
```



### 3. HTTPS aktivieren

In diesem Kapitel wird eine bestehende Webseite im Intranet mittels Zertifikat HTTPS-fähig gemacht.

Heutzutage existiert mit Let's Encrypt eine gute Gratis-Lösung für Web Server Zertifikate, die mit Hilfe des Let's Encrypt Bots mit wenigen Handgriffen installiert ist. In diesem Kapitel wollen wir das Zertifikat jedoch "von Hand" bei unserer CA lösen und installieren und dabei die verschiedenen kryptografischen Vorgänge identifizieren.

Wechseln Sie auf Ihren Linux Client. Dieser dient gleichzeitig auch als unser Apache2 Web Server.

Öffnen Sie ein Terminal und wechseln Sie zum Root User.

Navigieren Sie anschliessend in den folgenden Ordner:

```
1 cd /etc/ssl/private/
```

In diesem Ordner werden die Private Keys aufbewahrt. Als nächstes werden Sie einen Certificate Request erstellen.

Google erzwingt seit Mai 2017, dass Zertifikate über Einträge im Zertifikatsfeld "Subject Alternative Names" verfügen müssen. Ansonsten wird das Zertifikat als invalid angezeigt. Mehr Informationen finden Sie hier:

<https://www.heise.de/security/artikel/Chrome-blockt-Zertifikate-mit-Common-Name-3717594.html>

Obiger Sachverhalt heisst für uns, dass Sie zuerst ein Konfigurationsfile für Ihren Certificate Request erstellen müssen.

```
1 nano request.conf
```

Fügen Sie folgenden Inhalt in die Datei ein. **Vergessen Sie nicht wo nötig Ihre Gruppennummer / Ihre Domäne anzupassen!**

```
1 [req]
2 distinguished_name = req_distinguished_name
3 req_extensions = v3_req
4 prompt = no
5 [req_distinguished_name]
6 C = CH
7 ST = ZG
8 L = Rotkreuz
9 O = CKTECK
10 OU = Switzerland
11 CN = www.gXX.ckteck.com
12 [v3_req]
13 keyUsage = keyEncipherment, dataEncipherment
14 extendedKeyUsage = serverAuth
15 subjectAltName = @alt_names
16 [alt_names]
17 DNS.1 = www.gXX.ckteck.com
```

Studieren Sie den untenstehenden Command und führen Sie ihn aus. Das Passwort für die Private Key Verschlüsselung können Sie selbst wählen. **Merken Sie es sich aber! Sie werden es wieder brauchen!**

```
1 openssl req -newkey rsa:4096 -sha256 -keyout www-private.key -out www.csr -
  config request.conf
```

### Aufgabe 3.1

Was haben Sie hier gerade generiert? Beschreiben Sie!

Hinweis: Es handelt sich um mehrere Komponenten.

### Aufgabe 3.2

Wie funktioniert so ein Certificate Signing Request? Welche Schlüssel kommen wo für was zum Einsatz?

Hinweis: [https://en.wikipedia.org/wiki/Certificate\\_signing\\_request](https://en.wikipedia.org/wiki/Certificate_signing_request)

Öffnen Sie nun eine Firefox-Instanz und navigieren Sie zum Certification Authority Enrollment Web Service. Authentifizieren Sie sich mit dem Domain Admin Account "**labadmin**".

Sie finden den Enrollment Web Service unter folgender URL. Ersetzen Sie Ihre Gruppennummer.

<http://g01.ckteck.com/certsrv>

Navigieren Sie wie folgt:

```
1 request a certificate -> advanced certification request -> submit a request
```

Kopieren Sie nun den Inhalt der zuvor generierten **www.csr** Datei in das Saved-Request Feld und wählen Sie **Web Server** als Certificate Template.

**Microsoft Active Directory Certificate Services – company-ca****Submit a Certificate Request or Renewal Request**

To submit a saved request to the CA, paste a base-64-encoded

**Saved Request:**

Base-64-encoded  
certificate request  
(CMC or  
PKCS #10 or  
PKCS #7):

```
S3lQNnnV1D/HS5sK3iKb9GQHcq0w1P6qDSfLIB
i4YwGtdlWaqDres2VtYm6tNjBwqojDskJx210zi
Vc7lg+oH09GcBJlPgapI4uQE1nexJg+Fa72f0E
HjI=
-----END CERTIFICATE REQUEST-----
```

**Certificate Template:**

Web Server

**Additional Attributes:**

Attributes:

Lassen Sie sich ein Zertifikat generieren. Laden Sie nun das BASE64 encodierte Zertifikat herunter. Öffnen Sie das Zertifikat mit dem Text Editor und kopieren Sie den Inhalt in die Zwischenablage. Wechseln Sie wieder ins Terminal.

Navigieren Sie eine Stufe hoch und wechseln Sie in den Ordner certs.

```
1 cd ..
2 cd certs/
```

Erstellen Sie eine Datei namens `www.cer` und fügen Sie den zuvor in die Zwischenablage kopierten Inhalt ein. Speichern und verlassen Sie die Datei wieder.

```
1 nano www.cer
```

Nun besitzen Sie ein Web Server Zertifikat und haben es an der korrekten Stelle abgelegt.

Apache2 merkt dies jedoch nicht automatisch, sondern muss noch entsprechend konfiguriert werden. Öffnen Sie dazu die Apache2 SSL-Konfigurationsdatei:

```
1 nano /etc/apache2/sites-available/default-ssl.conf
```

Passen Sie die folgenden Zeilen an. Danach speichern und verlassen Sie die Datei wieder. Der Eintrag `ServerName` müssen Sie neu anlegen. Fügen Sie ihn einfach oberhalb der anderen beiden Linien ein. Passen Sie die URL entsprechend Ihrer Gruppennummer an.

```
1 ServerName www.gXX.ckteck.com
2 SSLCertificateFile /etc/ssl/certs/www.cer
3 SSLCertificateKeyFile /etc/ssl/private/www-private.key
```

Apache2 unterstützt Out-of-the-Box kein SSL. Dieses Modul muss zuerst aktiviert werden. Dies geschieht mit den folgenden zwei Commands.

```
1 a2enmod ssl
2 a2ensite default-ssl
```

Nun könnte theoretisch nach einem Neustart des Apache2 Servers auf unsere Webseite mit HTTPS zugegriffen werden, jedoch ist weiterhin der Zugriff mittels HTTP möglich. Dies wollen wir verhindern. Öffnen Sie dazu die folgende Datei:

```
1 nano /etc/apache2/sites-available/000-default.conf
```

Ergänzen Sie die Datei mit dem folgenden Statement innerhalb der VirtualHost Port 80 Definition

```
1 Redirect permanent / https://www.gXX.ckteck.com/
```

Die Konfiguration ist nun abgeschlossen. Damit diese wirksam wird, muss der Apache2 Server neu gestartet werden. Geben Sie das Passwort, welches Sie beim Generieren des Zertifikat-Requests angegeben haben, ein.

```
1 systemctl reload apache2 && systemctl restart apache2
```

Starten Sie zur Sicherheit Ihren Windows Client neu bevor Sie fortfahren. Sie werden gleich erfahren wieso. Loggen Sie sich danach mit dem Benutzer sysing01 ein.

### Aufgabe 3.3

Testen Sie nun die zuvor mit HTTPS ausgerüstete Webseite ([www.gXX.ckteck.com](https://www.gXX.ckteck.com)) von Ihrem Linux Client und Ihrem Windows Client (User sysing01) aus. Warum sieht bei Windows alles gut aus, während auf Ihrem Linux Gerät dem Zertifikat nicht vertraut wird?

Wird dem Webserver-Zertifikat auf Ihrem Windows Client nicht vertraut, müssen Sie eventuell den Windows Client neustarten. Wird dann dem Zertifikat immer noch nicht vertraut, wird etwas mit dem AD-Publish des Zertifikats auf Seite 17 falsch gelaufen sein. Reparieren Sie etwaige Fehler bevor Sie weiterfahren.

## 4. Certificate Authentication

In diesem Kapitel werden Sie die Authentifizierung mit Zertifikat kennenlernen und konfigurieren.

### 4.1. Web Server konfigurieren

Da unsere Webseite interne Informationen beinhaltet, sollen nur Mitarbeiter mit einem gültigen Zertifikat darauf zugreifen können. Dies bietet einen guten Schutz, ohne dass die Mitarbeiter sich aktiv authentifizieren müssen.

Um dieses Feature zu nutzen, muss die Certificate Chain der Certification Authority (CA) dem Web Server bekannt sein. Gegen diese wird der Web Server die User Zertifikate prüfen.

Die CA Certificate Chain kann im Certification Authority Enrollment Web Service heruntergeladen werden. Dies ist der gleiche Ort, wo Sie auch schon das Web Server Zertifikat bezogen haben. Auf der Frontpage finden Sie die Option "Download a CA certificate, certificate chain or CRL".

Wählen Sie BASE64 Encoding und laden Sie die "**CA certificate chain**" herunter. Die Chain kommt im p7b-Format daher. Damit unser Debian dieses lesen kann, muss es ins PEM-Format konvertiert werden. Vereinfacht gesagt, besteht ein PEM File aus mehreren Zertifikaten, die einfach ins gleiche File kopiert wurden.

Passen Sie wo nötig Dateinamen und Pfadangaben auf Ihren Download-Ort an.

```
1 sudo openssl pkcs7 -print_certs -in ca-chain.p7b -out /etc/ssl/certs/ckteck-ca.pem
```

Die dazu nötigen Konfigurationen des Apache2 Servers werden in der folgenden Datei vorgenommen:

```
1 sudo nano /etc/apache2/sites-available/default-ssl.conf
```

Kommentieren Sie folgende Zeilen ein/ergänzen und verändern Sie entsprechend. Danach speichern und verlassen Sie die Datei:

```
1 SSLCACertificateFile "/etc/ssl/certs/ckteck-ca.pem"
2 SSLVerifyClient require
3 SSLVerifyDepth 2
```

Damit die Konfigurationsänderungen wirksam werden, muss der Apache2 Server neu gestartet werden.

```
1 sudo systemctl restart apache2
```

#### Aufgabe 4.1

Versuchen Sie auf Ihrem Windows Client die Webseite erneut aufzumachen. Sollte sich nichts verändern, sind Sie eventuell Opfer der berühmten Cache-Problematik geworden. Am einfachsten öffnen Sie bei jedem Webseiten-Test in dieser Laborübung ein neues InPrivate Browsing Fenster. Dies geschieht mit der Tastenkombination CTRL + Shift + N. Eine weitere Möglichkeit ist das Löschen des Cache.

Kann auf die Webseite zugegriffen werden? Erscheint eine Meldung? Warum funktioniert der Zugriff nicht?

#### Aufgabe 4.2

Beschreiben Sie den Kommunikationsablauf zwischen Client und Web Server bei zertifikatsbasierter Authentifizierung.

Hinweis: Anhang B – Zertifikatsbasierte, gegenseitige Authentifizierung könnte hilfreich sein.

## 4.2. User Certificate Autoenrollment

Der Zugriff auf die Webseite funktioniert nicht mehr, weil der eingeloggte User kein Zertifikat besitzt. Man könnte nun analog zum HTTPS Kapitel ein Certificate Request erstellen und vom Certification Authority Enrollment Web Service ein Zertifikat anfordern. Jedoch ist das vom Otto Normaluser zu viel verlangt und zu aufwendig, diesen Arbeitsschritt durch einen IT-Mitarbeiter bei jedem Benutzer durchführen zu lassen.

Abhilfe schafft das Autoenrollment Feature. Dieses erlaubt es, Zertifikate in einer Windows Domäne automatisiert an die berechtigten User zu verteilen.

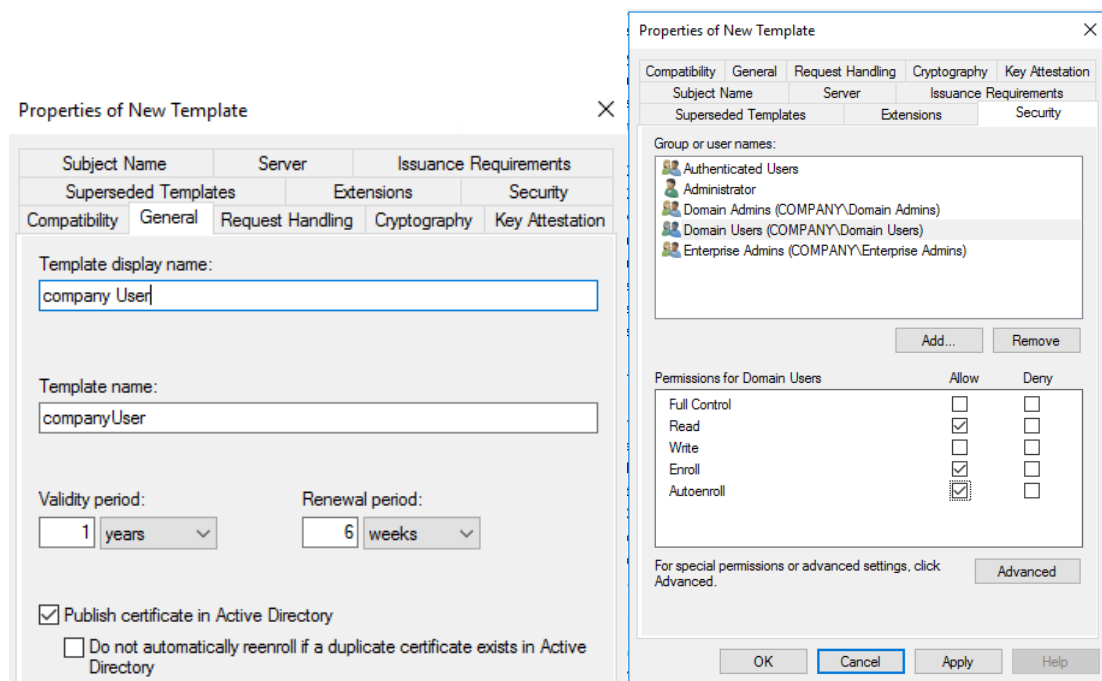
### 4.2.1. Certificate Template erstellen

Wechseln Sie wieder auf Ihren Windows Server und öffnen Sie den Certification Authority Konfigurationsdialog.

Öffnen Sie das Kontextmenü des Eintrags "Certificate Templates" mittels Rechtsklick und wählen Sie "Manage".

Suchen Sie nach dem vordefinierten Template "User" und erstellen Sie eine Kopie davon mit dem Kontextmenü-Eintrag "Duplicate".

Vergeben Sie dem Template einen Namen und fügen Sie im Tab Security die Gruppe "Domain Users" hinzu. Diese soll die folgenden Berechtigungen erhalten.



Speichern Sie Ihr neues Template und kehren Sie in den Certification Authority Dialog zurück.

Nun ist ein neues Template für User Zertifikate erstellt, dieses ist jedoch noch nicht aktiv.

Öffnen Sie das Kontextmenü des Eintrages "Certificate Templates" erneut und klicken Sie auf "new -> Certificate Template to issue..."

Wählen Sie Ihr neu erstelltes Template aus.

#### 4.2.2. GPO konfigurieren

Um die Konfiguration zu vervollständigen, muss eine GPO konfiguriert werden, die den Domänen-Clients mitteilt, das Autoenrollment für Zertifikate verwendet werden soll.

Öffnen Sie das Group Policy Management.

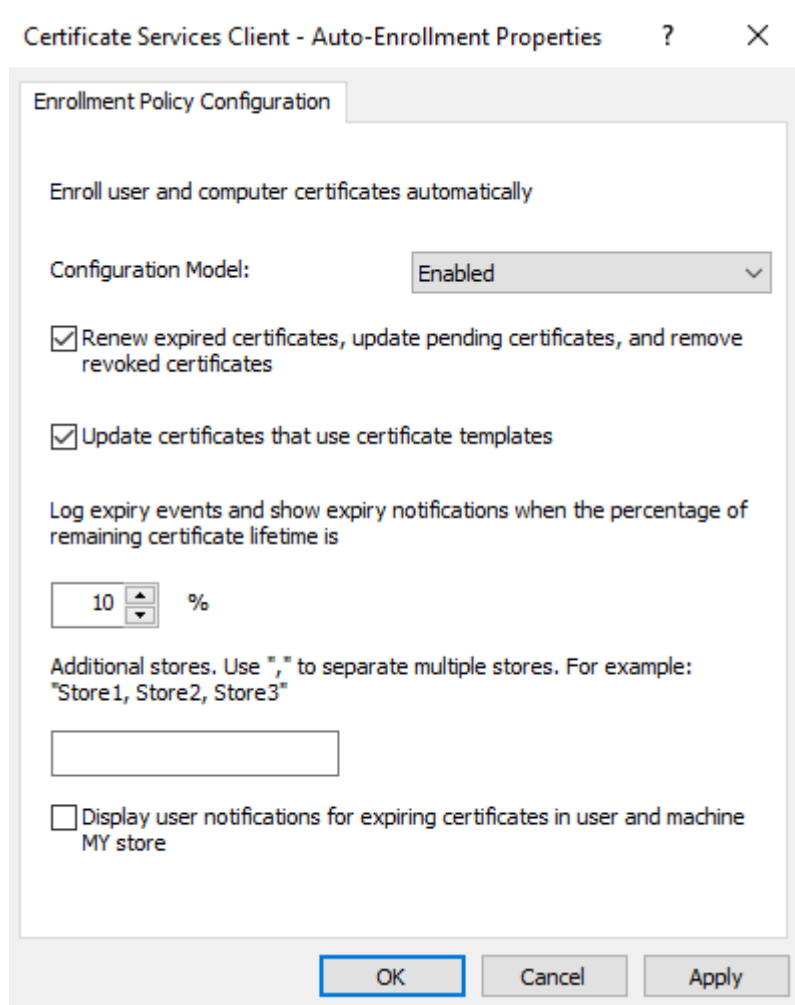
Erstellen Sie eine neue GPO auf dem Switzerland Ordner, damit alle Mitarbeiter (auch Admins) von der GPO betroffen sind und verlinken Sie diese darauf. Verwenden Sie einen sprechenden Namen für Ihre GPO.

Editieren Sie das erstellte Objekt und navigieren Sie in der GPO Baumstruktur an folgenden Ort:

1 User Configuration -> Policies -> Windows Settings -> Security Settings -> Public Key Policies

Die für uns spannende Einstellung heisst "Certificate Services Client – Auto-Enrollment"

Konfigurieren Sie diese wie folgt:



Melden Sie sich nun mit dem Domain User am Windows Client ab und wieder neu an. Versuchen Sie nun unsere Webseite nochmals zu öffnen.

Die Webseite kann nun nach erfolgreicher Authentifizierung mit Zertifikat geöffnet werden. Teils muss auch nochmals aktualisiert werden. Dies wegen der Browser Caching Funktion.

Wechseln Sie wieder in den "Certification Authority" Dialog und schauen Sie sich das soeben ausgestellte Zertifikat unter "Issued Certificates" an.

#### Aufgabe 4.3

Wechseln Sie in den Tab Details. Für welche Zwecke kann dieses Zertifikat alles verwendet werden?

### 4.3. Certificate Revocation

Stellen Sie sich vor, Sie arbeiten in der IT eines mittleren Unternehmens. Wie im Berufsleben so üblich kommt es vor, dass Mitarbeitende kündigen. Dabei ist es gang und gäbe, dass nach dem Austritt eines Mitarbeitenden sein/ihr Windows etc. Account gesperrt wird.



So weit so gut. Nun haben Sie aber im vorherigen Schritt Authentifizierung mit Zertifikat eingerichtet. Dieses Zertifikat kann unabhängig von den Windows Login Daten des Mitarbeiters verwendet werden. Zudem sind Zertifikate meistens für ein Jahr gültig. Im schlimmsten Fall ist also ein Zertifikat weiterhin für ein Jahr gültig!

Es muss ein Mechanismus her, der es erlaubt Zertifikate für ungültig zu erklären. Eine Möglichkeit bieten Certificate Revocation Lists (CRL). Wird ein Zertifikat für ungültig erklärt, gibt es einen Eintrag in der CRL. Bei jeder Anfrage wird die CRL konsultiert und sequenziell nach dem vom Client übertragenen Zertifikat durchsucht.

#### Aufgabe 4.4

Wieso könnte die oben beschriebene Funktionsweise von CRLs bei hoher Fluktuation innerhalb der Firma zu Problemen führen?

Um das obige Problem zu umgehen, wurde eine neue Methode eingeführt: Online Certificate Status Protocol, kurz OCSP. Dieses funktioniert nach dem Client-Server Prinzip, wobei der Web Server jeweils die Issuing CA nach der Gültigkeit des übermittelten Zertifikats abfragt.

#### 4.3.1. OCSP konfigurieren

Wechseln Sie wieder in den Certification Authority Dialog auf Ihrem Windows Server.

Öffnen Sie das Kontextmenü von "Revoked Certificates" und publizieren Sie die für ungültig erklärten Zertifikate manuell. Momentan sind das noch keine, es wird jedoch eine initiale, leere Certificate Revocation List (CRL) generiert.

Öffnen Sie wieder das Kontextmenü der "Certificate Templates" und wählen Sie "Manage".

Als nächstes muss das Template "OCSP Response Signing" dupliziert werden. Dies geschieht analog zum User Zertifikat.

Im Tab "Security" muss das Computerobjekt des Online Responders hinzugefügt werden (in unserem Fall der Domain Controller). Würde der Online Responder auf einem anderen Server installiert (grundsätzlich empfohlen), müsste dasjenige Computerobjekt hinzugefügt werden.

#### Hinweis

Damit das Computer Objekt gefunden werden kann, müssen Computer Objekte im Dialog "Object Types..." hinzugefügt werden.

Properties of New Template ✕

Compatibility General Request Handling Cryptography Key Attestation

Subject Name Server Issuance Requirements

Superseded Templates Extensions Security

Group or user names:

- Authenticated Users
- labadmin (labadmin@g01dev.ckteck.com)
- Domain Admins (G01DEV\Domain Admins)
- Enterprise Admins (G01DEV\Enterprise Admins)
- ISLAB-WS-DEV-01 (G01DEV\ISLAB-WS-DEV-01\$)

Add... Remove

Permissions for ISLAB-WS-DEV-01

	Allow	Deny
Full Control	<input type="checkbox"/>	<input type="checkbox"/>
Read	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Write	<input type="checkbox"/>	<input type="checkbox"/>
Enroll	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Autoenroll	<input type="checkbox"/>	<input type="checkbox"/>

Aktivieren Sie das neu erstellte Template über das Kontextmenü der Certificate Templates mittels "New Template to Issue..."

Als nächstes muss der Online Responder konfiguriert werden. Das dazu benötigte Tool befindet sich ebenfalls in den "Windows Administrative Tools" und nennt sich "Online Responder Management".

Erstellen Sie eine neue Revocation Configuration über das Kontextmenü der "Revocation Configuration". Da sich die Certification Authority auf demselben Computer und in derselben Domäne befindet, wird der Radio Button "Existing enterprise CA" ausgewählt.

Add Revocation Configuration ? ✕

**Select CA Certificate Location**

Getting started with addi...

Name the Revocation Co...

Select CA Certificate Loca...

Choose CA Certificate

Select Signing Certificate

Specify the location of the CA certificate that you want to associate with this revocation configuration.

☒ Select a certificate for an Existing enterprise CA

Select this option if your CA certificate is available in Active Directory or on the CA computer

Wählen Sie über die Browse Funktion das bestehende CA Certificate aus.

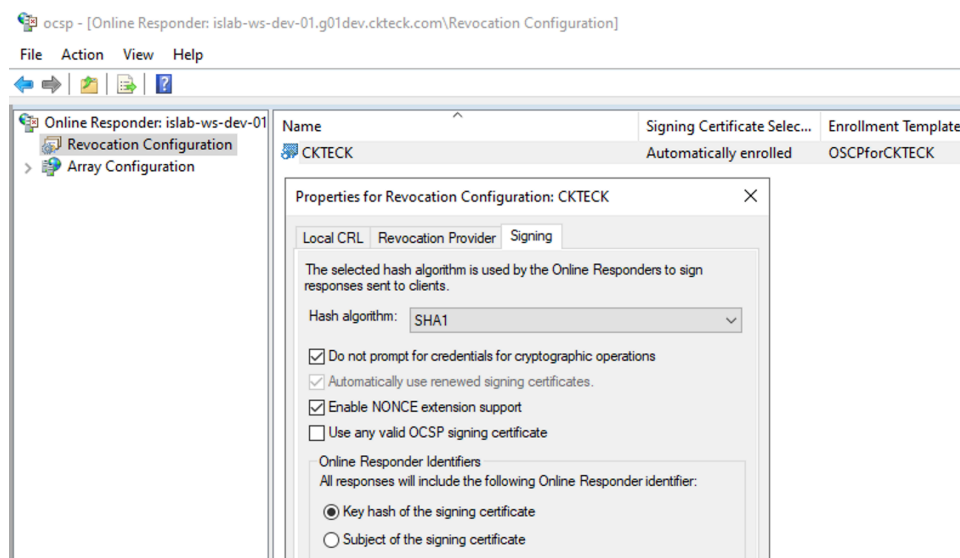
In order to check the status of a certificate, a revocation configuration for the Online Responder must identify the CA that issued the certificate. You can identify this CA by selecting a CA certificate published in Active Directory or by locating a CA computer.

☒ Browse CA certificates published in Active Directory Browse...

Überprüfen Sie, dass das zuvor erstellte Certificate Template ausgewählt ist. Dies sollte automatisch geschehen.

☒ Automatically select a signing certificate  
☒ Auto-Enroll for an OSCP signing certificate  
 Certification authority: islab-ws-dev-01.g01dev.ckteck.com\CKTECK CA  
 Browse...  
 Certificate Template: OSCPforCKTECK

Öffnen Sie die Properties der Revocation Configuration und wechseln Sie in den Tab "Signing". Aktivieren Sie dort den "NONCE extension support".



#### Aufgabe 4.5

Was hat die OSCP NONCE Extension für eine Funktion?

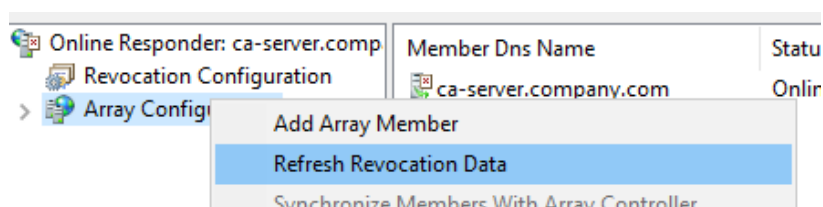
Hinweis: der folgende Wikipedia Artikel hilft Ihnen:

[https://en.wikipedia.org/wiki/Online\\_Certificate\\_Status\\_Protocol](https://en.wikipedia.org/wiki/Online_Certificate_Status_Protocol)

Da der Apache2 Web Server in der Default Konfiguration die NONCE Extension verlangt, muss diese im Windows Online Responder aktiviert werden. Ansonsten funktioniert die Kommunikation nicht korrekt.

Folgender Artikel zeigt Ihnen, wie Sie auch seitens Web Server die NONCE Extension ausschalten könnten: [https://httpd.apache.org/docs/trunk/mod/mod\\_ssl.html#sslopensslrequestnonce](https://httpd.apache.org/docs/trunk/mod/mod_ssl.html#sslopensslrequestnonce)

Laden Sie die Revocation Data über das Kontextmenü der Array Configuration neu.



Überprüfen Sie die durchgeführte Konfiguration mittels des Tools "pkiview.msc". Geben Sie dazu "pkiview.msc" in ein "Ausführen..."-Fenster ein. Alle Werte sollten "OK" sein. Warnings mit Status "Expiring" wo das Expiration Date in der Zukunft liegt sind nicht weiter schlimm und werden sich selbst beheben, sobald die neue CRL automatisch gepushed wird.

#### 4.3.2. Web Server konfigurieren

Nun muss noch dem Web Server mitgeteilt werden, dass er die Gültigkeit von Zertifikaten mittels OCSP prüfen soll.

Um die komplette Certificate Chain prüfen zu können, muss die Certificate Chain dem Apache Web Server bekannt gegeben werden. Dies haben Sie bereits früher in dieser Übung gemacht. Sie erinnern sich daran, wie Sie die P7B Certificate Chain in das PEM-Format umgewandelt haben etc.

Als nächstes muss die Konfiguration des Apache Web Servers angepasst werden. Öffnen Sie dazu erneut die folgende Datei auf ihrem Linux Client.

```
1 nano /etc/apache2/sites-available/default-ssl.conf
```

Fügen Sie die folgenden Einträge ein. Die Commands befinden sich nicht als Vorlage in der Konfigurationsdatei und müssen selbst ergänzt werden.

```
1 SSLOCSPEnable on
2 SSLOCSPDefaultResponder http://islab-ws-XX.gXX.ckteck.com/ocsp
3 SSLOCSPUseRequestNonce on
```

Starten Sie als letztes den Apache2 Web Server neu, damit die Konfiguration wirksam wird. Mittlerweile kennen Sie wahrscheinlich den Befehl bereits auswendig.

#### 4.3.3. Revocation testen

Wechseln Sie nun zurück in den Certification Authority Dialog und wechseln Sie zu Issued Certificates. Erklären Sie mittels Kontextmenü das zuvor für den User Account sysing01 ausgestellte Zertifikat für ungültig.

Publizieren Sie nochmals die CRL Liste. Dies würde in bestimmten Intervallen auch automatisch passieren, jedoch kann die Aktualisierung auch manuell getriggert werden, um weniger lange zu warten.

Öffnen Sie auch nochmals das Online Responder Management und aktualisieren Sie die Revocation Data. Dies würde auch automatisch geschehen – wir beschleunigen die Sache nur ein wenig.

Wechseln Sie zum Windows Client. Vergewissern Sie sich, dass der Cache etc. ihres Browsers geleert wurde. Ansonsten ist es gut möglich, dass Ihnen vom Browser eine lokal gecachte Version der Webseite angezeigt wird. Versuchen Sie die geschützte Webseite nochmals zu öffnen.

#### Aufgabe 4.6

Was ist nun passiert? Können Sie sich weiter die Webseite anzeigen lassen?

Öffnen Sie den Apache2 Error Log und sehen Sie sich die Meldungen darin an (zu finden unter `"/var/log/apache2/error.log"`).

Welche Meldung weist darauf hin, dass das vom Web Server erhaltene Zertifikat nicht mehr gültig ist?

## 5. Code Signing

Um die Herkunft sowie die Integrität einer kompilierten Software zu beweisen, kann diese mit einem Zertifikat signiert werden. Viele Softwarehersteller signieren ihre Software bereits, jedoch gibt es auch noch unzählige die dies nicht tun. Ohne Signatur können potenziell bösartige Veränderungen durch Dritte an einer Software vorgenommen werden, ohne dass jemand etwas davon merkt. Lukrative Ziele sind zum Beispiel bekannte OpenSource Softwares, die zum freien Download im Internet angeboten werden.

Im oben beschriebenen Beispiel müssen global gültige Zertifikate eingesetzt werden. Die lokal von unserer PKI ausgestellten Zertifikate sind natürlich im GBI (Grosses Böses Internet) unbekannt und somit ungültig. Die dahinterstehenden Mechanismen sind jedoch identisch.

In dieser Übung werden Sie ein PowerShell Script signieren. Auch Scripts können signiert und somit vor unbemerkten Änderungen geschützt werden.

### 5.1. Windows Scripting Standardverhalten feststellen

Loggen Sie sich mit dem Benutzer sysing01 am Windows Client ein.

Erstellen Sie auf dem Desktop eine neue Textdatei und ändern Sie die Endung der Datei auf ".ps1" um diese als PowerShell Script zu deklarieren.

#### Hinweis

Falls Sie die Endungen der Dateien nicht sehen können, ist das ein Microsoft Feature namens "Hide extensions for known file types". Da dieses Feature jedoch alles andere als hilfreich ist, wird empfohlen dieses in den Ordneroptionen auszuschalten.

Fügen Sie folgendes kleines Script in das erstellte PowerShell Script ein und speichern Sie.

```
1 $wshShell = New-Object -comObject WScript.Shell
2 $shortcut = $wshShell.CreateShortcut($env:HOMEPATH+"\Desktop\Intranet.lnk")
3 $shortcut.TargetPath = "https://www.gXX.ckteck.com/"
4 $shortcut.Save()
```

#### Aufgabe 5.1

Was macht das Script? Was erwarten Sie als Resultat, wenn das Script ausgeführt wurde?

**Öffnen Sie eine PowerShell Console.** Führen Sie das Script aus.

Funktioniert das Script? / Was für eine Meldung tritt auf?

Wie Sie in der Fehlermeldung lesen können, hat das Problem etwas mit Execution Policies zu tun. PowerShell kennt fünf verschiedene Execution Policies. Machen Sie sich mit Hilfe der folgenden Webseite mit den möglichen Execution Policies vertraut und beschreiben Sie kurz deren Unterschiede:

[https://docs.microsoft.com/de-de/powershell/module/microsoft.powershell.core/about/about\\_execution\\_policies?view=powershell-6](https://docs.microsoft.com/de-de/powershell/module/microsoft.powershell.core/about/about_execution_policies?view=powershell-6)

#### Aufgabe 5.2

Restricted:

#### Aufgabe 5.3

AllSigned:

#### Aufgabe 5.4

RemoteSigned:

#### Aufgabe 5.5

Unrestricted:

#### Aufgabe 5.6

Bypass:

Lesen Sie die aktuell angewendete Execution Policy mit folgendem Commandlet aus:

## 1 Get-ExecutionPolicy

### 5.2. Execution Policy anpassen

Ihr Systemadministrator sieht nach einem verbalen Schlagabtausch beim Kaffee ein, dass sie als PowerShell-Developer Scripts auf ihrem lokalen Arbeitsgerät ausführen müssen. Da Ihre Scripts aber später in der ganzen Organisation verteilt und ausgeführt werden, muss eine langfristige und sicherheitstechnisch saubere Lösung her.

Damit nur Scripts ausgeführt werden können, welche aus vertrauenswürdigen Quellen stammen (eigene Entwickler), wird die Execution Policy "AllSigned" gewählt.

Damit alle Computer in der ganzen Unternehmung zentral umgeschaltet werden können, wird die Änderung über das Group Policy Management des Active Directorys realisiert.

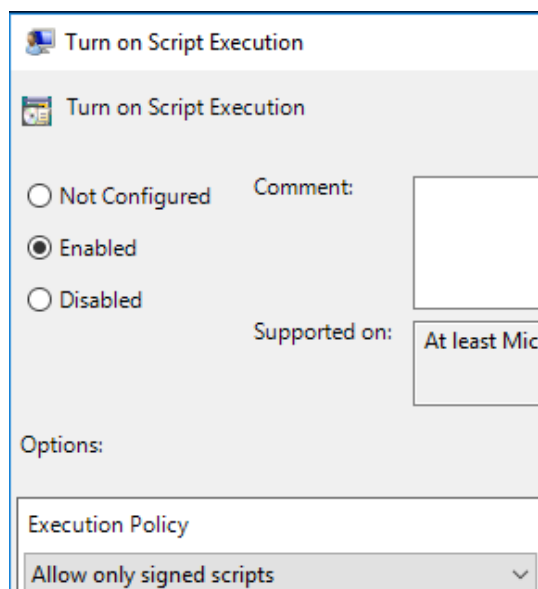
Öffnen Sie das Group Policy Management und überlegen Sie sich, ob Sie eine bestehende GPO verwenden oder eine neue anlegen wollen. Überlegen Sie auch, auf welche OU Sie die GPO applizieren wollen.

Setzen Sie Ihre Überlegungen um.

Aktivieren Sie in der GPO die folgende Einstellung:

- 1 Computer Configuration -> Policies -> Administrative Templates -> Windows Components -> Windows PowerShell -> Turn on
- 2 Script Execution

Wählen Sie "Allow only signed scripts".



Da es sich um eine Einstellung auf Computer-Ebene handelt, muss der Computer neugestartet werden, damit die Änderung wirksam wird.

Überprüfen Sie nach dem Neustart nochmals die Execution Policy und vergewissern Sie sich, dass diese den korrekten Wert vorweist.

## 1 Get-ExecutionPolicy



### Aufgabe 5.7

Führen Sie nochmals das zuvor erstellte PowerShell Script aus. Was für eine Meldung erscheint nun?

## 5.3. Code Signing Zertifikate ausstellen

Grundsätzlich gibt es zwei Möglichkeiten Code Signing zu betreiben. Crypt32 beschreibt diese auf serverfault.com (eine Schwesterseite von stackoverflow.com) wie folgt:

1) issue a personal signing certificate to each developer. Assign CodeSigning certificate template to CA and grant permissions to dev group.

pros: each developer has his own signing certificate.

cons: too many certificates. May be ok for internal (development and testing) purposes only. Is not suitable when you deliver signed binaries to your customers or other 3rd parties.

2) issue a single certificate to developer groups. Export to PFX and provide a copy of this certificate to all devs.

pros: only one certificate is used for signing. Best suitable to sign binaries for external parties.

cons: you are not controlling who signed the file. You see it is signed by a particular certificate, but can't tell who exactly signed the file. It is security degradation.

as per best practices, products for external parties shall be signed by a single and authorized digital certificate that identifies the company. This certificate shall not have copies and should be stored in a secure container, for example smart cards. However, for internal use and if your policy allows this, I would go with option 1, to issue personalized code signing certificates to each developer.

Quelle: <https://serverfault.com/questions/679334/enterprise-root-ca-create-codesigning-certificate-for-multiple-users>

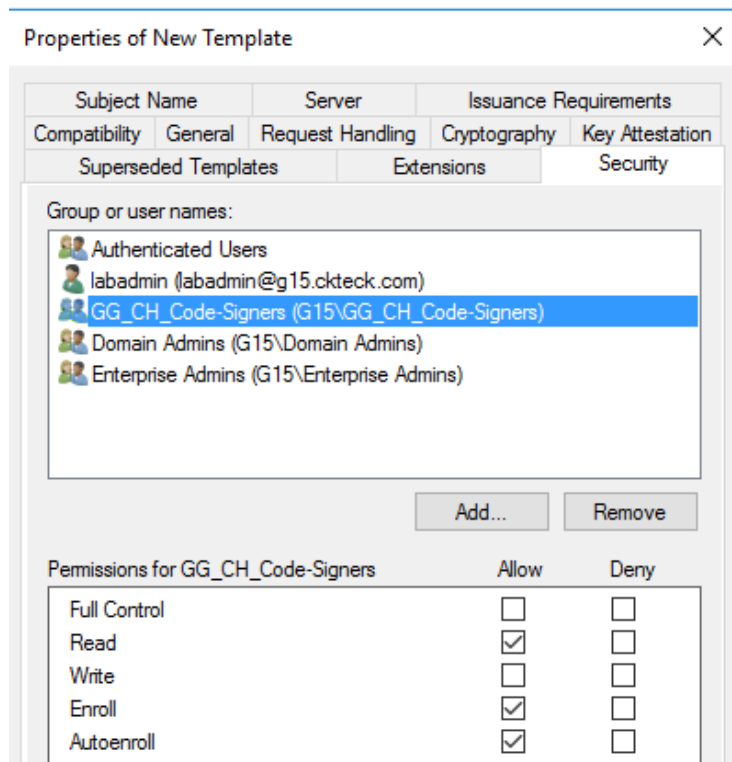
Wir werden in diesem Versuch mit Variante 1: persönliche Code Signing Zertifikate arbeiten.

Wechseln Sie auf den Windows Server.

Damit das Script signiert werden kann, muss zuerst ein Code Signing Zertifikat erlangt werden. Dieses wird auf ähnliche Weise wie die anderen in diesem Versuch ausgestellten Zertifikaten gemacht.

Das heisst, es wird wiederum eine Gruppe benötigt, welcher es erlaubt ist, ein solches Zertifikat zu erlangen. Erstellen Sie dazu eine neue Gruppe im Active Directory mit dem Namen **"GG\_CH\_Codesigners"**. Wie Sie das machen, haben Sie bereits im Access Management Versuch gelernt. Fügen Sie der Gruppe den Benutzer Sysing01 hinzu.

Öffnen Sie den Certification Authority Dialog. Rechtsklick auf Certificate Templates > Manage und duplizieren Sie das "Code Signing" Template.



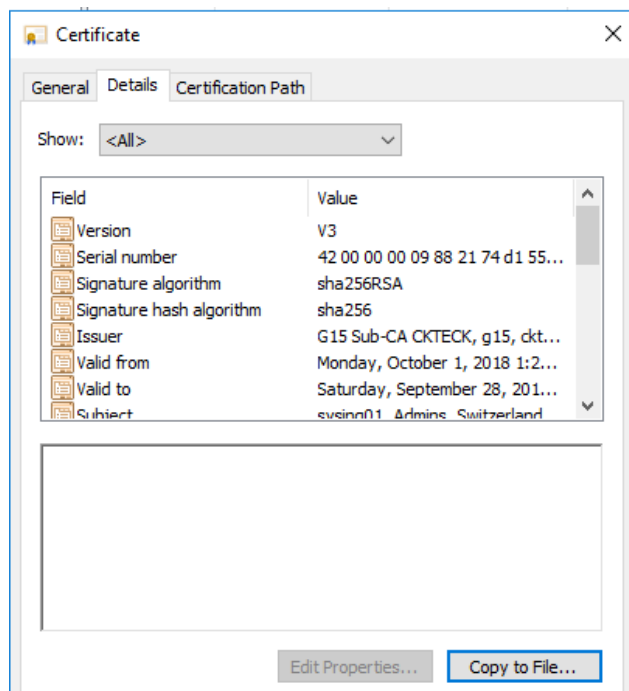
Aktivieren Sie das neu erstellte Template mittels Kontextmenü der "Certificate Templates -> new -> Certificate to issue".

Melden Sie sich neu am Windows Client an, damit Sie das Zertifikat erhalten. Kontrollieren Sie in Ihrem **User Certificate Store** -> Personal, ob Sie das Code Signing Zertifikat erhalten haben.

Um Sie vor der nächsten Stolperfalle zu bewahren, wird im Folgenden ein wenig vorgegriffen. Sie können nun zwar Scripts signieren, jedoch gelten Sie noch nicht als "Trusted Publisher" auf Ihrem Windows Client.

Dazu muss Ihr Zertifikat zuerst verteilt und auf allen Windows Clients in die "Trusted Publishers" aufgenommen werden. Dies kann zum Beispiel mittels Group Policy Management passieren.

Zuerst muss das gefragte Zertifikat bereitgestellt werden. Wechseln Sie auf ihrem Windows Server in den Certificate Authority Dialog und sehen Sie sich die ausgestellten Zertifikate an. Identifizieren Sie das Code Signing Certificate, welches für Sysing01 ausgestellt wurde. Exportieren Sie das Zertifikat im BASE64-Format.



Öffnen Sie nochmals den Group Policy Management Editor. Navigieren Sie wie folgt:

- 1 Computer Configuration -> Policies -> Windows Settings -> Security Settings -> Public Key Policies -> Trusted Publishers

Starten Sie mittels Kontextmenü den Import-Dialog und importieren Sie das gerade eben exportierte Code Signing Certificate.

Nun ist ein **gpupdate /force** notwendig, um die Einstellung auf dem Windows Client zu aktualisieren. Kontrollieren Sie anschliessend Local Computer Certificate Store, ob das Zertifikat am korrekten Ort erscheint.

## 5.4. Script signieren

Nun wollen wir das Script signieren. Öffnen Sie eine PowerShell Console und geben Sie folgenden Befehl ein. Passen Sie den Pfad zu Ihrem Script wenn nötig an.

- 1 Set-AuthenticodeSignature -FilePath .\script.ps1 -Certificate (Get-ChildItem -Path Cert:\CurrentUser\My\ -CodeSigningCert) -TimeStampServer http://timestamp.digicert.com

### Aufgabe 5.8

Öffnen Sie das Script in einem Editor Ihrer Wahl. Sehen Sie sich das Script an. Was wurde hinzugefügt?

Führen Sie das Script nochmals aus. Es sollte nun funktionieren.

### Aufgabe 5.9

Öffnen Sie das Script erneut und verändern Sie etwas. Das kann zum Beispiel das Einfügen eines Kommentars mittels `"#"`-Zeichen sein. Speichern Sie.

Führen Sie das Script nochmals aus. Funktioniert das Script noch? Was für eine Meldung erscheint?

Signieren Sie das Script nochmals und führen Sie es nochmals aus. Stellen Sie sicher, dass das Ausführen des Scripts wieder funktioniert, bevor Sie weiterfahren.

## 5.5. Revocation Problematik

Was passiert aber wenn ein Mitarbeiter, welcher Code signiert hat, die Unternehmung verlässt? Was passiert mit dem von diesem Mitarbeiter signierten Code? In diesem Kapitel werden Sie genau diesem Umstand nachgehen.

Wechseln Sie in den Certification Authority auf Ihrem Windows Server und erklären Sie das vorhin ausgestellte Code Signing Zertifikat als ungültig. Gehen Sie analog zum vorhergehenden Kapitel mit User Certificates vor (CRL publizieren, Online Responder Revocation Data aktualisieren, um nicht warten zu müssen etc.). Damit simulieren wir den Austritt des Mitarbeiters.

Wechseln Sie nun wieder zurück zum Windows Client. Versuchen Sie das Script nochmals auszuführen.

### Aufgabe 5.10

Was passiert? Kann das Script noch erfolgreich ausgeführt werden? Warum ist das festgestellte Verhalten so? Hinweis: Sehen Sie sich die Stammzertifikate von Microsoft im lokalen Certificate Store an. Warum hat es da schon längst abgelaufene Zertifikate drin? Die könnte man doch schon lange rauslöschen, oder etwa doch nicht?



Signieren Sie nun das Script erneut und speichern Sie es z.B. unter `C:\temp\`. Melden Sie sich als Sysing01 vom Windows Client ab. Loggen Sie sich mit einem anderen Benutzer Ihrer Wahl ein (kein Domain Admin (labadmin) verwenden!).

### Aufgabe 5.11

Öffnen Sie eine PowerShell Console und versuchen Sie das Script erneut auszuführen. Wie sieht es jetzt aus? Kann das Script ausgeführt werden? Warum ja/nicht?

## 6. SSL Interception

### Aufgabe 6.1

Beginnen wir dieses Kapitel mit einer Frage: Was bedeutet für Sie das "Grünes Schloss"-Symbol in der Adressleiste des Browsers?   <https://>

In diesem Versuch möchten wir Ihnen aufzeigen, dass E-Banking oder ähnliches nur in Ihnen bekannten und vertrauenswürdigen Netzwerken gemacht werden sollte. Warum? – dies sehen Sie gleich.

Um SSL Interception (zu Deutsch "Abfangen") machen zu können, ist ein Man-in-the-Middle (MitM) Aufbau nötig. Heisst, der gesamte Netzwerkverkehr muss über eine Zwischenstation laufen, bevor die Zielstelle (z.B. Webseite) angegangen wird. Der klassische MitM-Fall ist der Hacker (Yuri) der sich mittels Spoofing Methoden zwischen das Opfer und sein Ziel drängt. Es muss jedoch nicht immer ein Hacker sein, sondern man kann z.B. das Default Gateway eines jeden Geräts durchaus auch als MitM-Gerät betrachten, da jeglicher Datenverkehr nach externen Netzwerken über dieses eine Gerät geleitet wird. Diesen Umstand werden wir mit Hilfe der OpenSource Software "mitmproxy" zu Nutze machen.

Unser Szenario spielt sich im internen Firmenumfeld ab. Das Feindbild hier ist "Der Systemadmin".

Obiges Szenario lässt sich jedoch auch gut auf z.B. öffentliche Flughäfen anwenden. Mehr dazu später.

### 6.1. mitmproxy

Die Webseite von mitmproxy beschreibt die Software mit folgenden Worten:

"mitmproxy is your swiss-army knife for debugging, testing, privacy measurements, and penetration testing. It can be used to intercept, inspect, modify and replay web traffic such as HTTP/1, HTTP/2, WebSockets, or any other SSL/TLS-protected protocols. You can prettify and decode a variety of message types ranging from HTML to Protobuf, intercept specific messages on-the-fly, modify them before they reach their destination, and replay them to a client or server later on."

Mitmproxy ist ein sehr mächtiges Werkzeug. In diesem Versuch werden wir nur einen Teil der gebotenen Funktionalität verwenden.

#### 6.1.1. mitmproxy installieren

Wir werden den mitmproxy über den Python Package Manager pip (pip installs packages) installieren. Man könnte mitmproxy auch aus dem apt-get Repository installieren, zum Zeitpunkt der Übungserstellung war die mitmproxy Version aus dem apt-get Repository jedoch veraltet. Darum wird hier auf pip zurückgegriffen.

Vor der Installation des Pakets soll pip aktualisiert werden.

```
1 sudo pip3 install -U pip
```

Installieren Sie nun mittels pip den mitmproxy. Ignorieren Sie dependency Errors falls Sie solche erhalten.

```
1 sudo python3 -m pip install mitmproxy
```

### 6.1.2. IP Forwarding aktivieren

Damit der Linux Client Netzwerkverkehr anderer Geräte weiterleiten kann, müssen einige Konfigurationen vorgenommen werden.

Ändern Sie in der Datei "/etc/sysctl.conf" folgende Einträge entsprechend ab.

```
1 net.ipv4.ip_forward=1
2 net.ipv6.conf.all.forwarding=1
3 net.ipv4.conf.all.send_redirects=0
```

Um die durchgeführten Änderungen anzuwenden, muss folgender Command ausgeführt werden.

```
1 sudo sysctl -p /etc/sysctl.conf
```

Zusätzlich ist ein Reboot des Linux Clients notwendig. Machen Sie einen Reboot über das Terminal. Wenn Sie das GUI verwenden, dann startet sich nur das GUI und nicht die ganze VM neu.

```
1 sudo reboot
```

In der momentanen Situation wird der Web Traffic einfach nur weitergeleitet. Es findet keine Interception statt.

Um den interessanten Traffic an den mitmproxy weiterzuleiten, müssen folgende nftables Rules definiert werden. Der mitmproxy horcht lokal auf dem TCP Port 8080.

```
1 sudo nft add table inet nat
2 sudo nft add chain inet nat prerouting { type nat hook prerouting priority
  0 \; }
3 sudo nft add rule inet nat prerouting iifname "ens3" tcp dport 80 counter
  redirect to :8080
4 sudo nft add rule inet nat prerouting iifname "ens3" tcp dport 443 counter
  redirect to :8080
```

### Aufgabe 6.2

Warum haben Sie nftables Einträge für inet (IPv4 und IPv6) konfiguriert?

Erklären Sie das Verhalten Ihres Browsers, wenn Sie eine Webseite mit Dual Stack-Kompatibilität öffnen!

Hinweis: [https://en.wikipedia.org/wiki/Happy\\_Eyeballs](https://en.wikipedia.org/wiki/Happy_Eyeballs)

Starten Sie den mitmproxy im interaktiven Modus mit folgendem Befehl.

```
1 /usr/local/bin/mitmproxy --mode transparent
```

## 6.2. Windows Client vorbereiten

Loggen Sie sich als erstes wieder mit dem User Sysing01 ein.

### Wichtig

Passen Sie auf, dass Sie beim Default Gateway keine Tippfehler machen! Ansonsten sperren Sie sich selbst aus!

Lassen Sie sich die derzeitige IPv4 Netzwerkeinstellungen anzeigen (IP Adresse und Subnet), z.B. mit `ipconfig`. Legen Sie nun die IP Adresse in den Adaptereinstellungen statisch fest. Verwenden Sie dieselbe IP Adresse und Subnet Maske wie zuvor. Für das Default Gateway wird die private IPv4 Adresse des Linux Clients verwendet.

## 6.3. Testing

Wechseln Sie auf Ihren Windows Client. Öffnen Sie eine Webseite, die HTTPS erzwingt. Heutzutage sind dies die meisten. Ein Beispiel ist <https://meteo.ch>.

### Aufgabe 6.3

Wie sieht es aus? Funktioniert es? Treten Probleme auf?

Schauen Sie sich das Resultat im Windows Client und in der mitmproxy Console an.

## 6.4. Eigene Certificate Authority einbinden

Wie Sie gemerkt haben, läuft noch nicht alles rund. Der Client sieht sofort, dass hier etwas nicht stimmt. Ihnen wird eine grosse Zertifikatswarnung angezeigt. In diesem Kapitel werden Sie eine Subordinate CA unter Ihrer eigenen Subordinate CA erstellen.

Zuerst muss erneut ein Certificate Request erstellt werden. Führen Sie den Befehl auf dem Debian System aus. Unter anderem werden Sie aufgefordert, ein Verschlüsselungspasswort für den Private Key einzugeben. Merken Sie sich dieses, da Sie es gleich wieder brauchen werden.

```
1 openssl req -new -newkey rsa:4096 -keyout mitmproxyc-private.key -out  
subca.csr
```

Öffnen Sie nun wieder den Certificate Enrollment Web Service. Sie finden diesen wieder unter <http://gXX.ckteck.com/certsrv>. Gehen Sie gleich wie beim Request für das Webserver Zertifikat vor, wählen Sie jedoch diesmal **"Subordinate Certification Authority"** als Certificate Template.

Laden Sie die **Certificate Chain im BASE64-encodierten Format** herunter. Gleich wie schon beim Webserver muss das Format der Certificate Chain angepasst werden. Verwenden Sie dazu nochmals folgenden Befehl (passen Sie Dateinamen etc. auf Ihre Umgebung an):



```
1 sudo openssl pkcs7 -print_certs -in subcachain.p7b -out /etc/ssl/certs/  
mitmproxy-ca.pem
```

Mitmproxy benötigt das Zertifikat plus den Private Key im gleichen PEM-File. Dies bedeutet für Sie, dass Sie den Inhalt des zuvor erstellten Private Key Files unten an die Datei "mitmproxy-ca.pem" anhängen müssen. Vergessen Sie nicht, die Datei zu speichern.

Starten Sie als nächstes den mitmproxy mit folgendem Befehl:

```
1 /usr/local/bin/mitmproxy --mode transparent --set confdir=/etc/ssl/certs --  
set cert_passphrase=<Verschlüsselungspasswort>
```

Falls ein Fehler auftritt, prüfen Sie, ob Sie wie oben beschrieben, den Private Key dem PEM-File angehängt haben. Beim Testing haben die Probanden diesen Schritt häufig vergessen. Vergewissern Sie sich ebenfalls, dass Sie die vorherige Instanz von mitmproxy beendet haben, bevor Sie die Software erneut starten.

## 6.5. Testing zum Zweiten

Navigieren Sie nun vom Windows Client zu der bei Studierenden sehr beliebten Webseite <https://elearning.hslu.ch/>. Loggen Sie sich mit **falschen** Angaben in der ILIAS-Login Maske ein.

### Aufgabe 6.4

Wechseln Sie zur mitmproxy Console und suchen Sie im mitmproxy Log nach Ihrem Login Versuch. Mit "Enter", "Q" und den Pfeiltasten können Sie in den einzelnen Requests navigieren. Finden Sie Ihre zuvor eingegeben Login Daten? Wo finden Sie diese? Beschreiben Sie!

## 6.6. Funktionsanalyse

Nachdem Sie gesehen haben, dass mit Hilfe der mitmproxy Software HTTPS (SSL) Verbindungen abgehört werden können, werden wir in diesem Kapitel die Funktionsweise der Software genauer unter die Lupe nehmen.

### Aufgabe 6.5

Wo könnten Sie mit der Funktionsanalyse der Software beginnen? Wo könnten Sie gegebenenfalls überall ansetzen? Nennen Sie ein paar Beispiele!

### Aufgabe 6.6

Untersuchen Sie die Funktionsweise an den von Ihnen oben genannten Stellen. Dokumentieren Sie Ihre Findings!

Sollten Sie verunsichert sein, wo Sie anfangen sollen, starten Sie mit der Analyse des Zertifikats der ILIAS Webseite auf Ihrem Windows Client. Beantworten Sie folgende Fragen.

### Aufgabe 6.7

Wer ist der Issuer des Zertifikats?

### Aufgabe 6.8

Wie sieht der Certification Path aus?

### Aufgabe 6.9

Machen die obigen Werte im Zusammenhang mit einer HSLU-Webseite Sinn?

Öffnen Sie noch ein paar weitere bekannte Webseiten Ihrer Wahl und studieren Sie die HTTPS-Zertifikate der Webseiten.

Wir schlussfolgern also, dass mitmproxy für jede besuchte Website ein neues Zertifikat anlegt. Diesem wird von unserem Client vertraut, da der ausstellende CA grundsätzlich vertraut wird.

## 6.7. Aufräumen

Um Konflikte mit darauffolgenden Übungen zu vermeiden, entfernen Sie die nftables Regeln wieder.

```
1 sudo nft delete table inet nat
```

## 6.8. Schlusswort

Kommen wir auf den einführenden Teil dieses Kapitels zurück. Wie Sie in diesem Versuch gesehen haben, bedeutet das Schloss-Symbol im Browser nicht automatisch das eine absolut sichere Verbindung zwischen Ihnen und der von Ihnen an gesurften Webseite besteht.

Wie Sie auch gesehen haben, wird einem wildfremden Zertifikat des Proxy Servers glücklicherweise nicht einfach blind vertraut.

Wird jedoch ein Zertifikat verwendet, welches bereits als vertrauenswürdig gespeichert ist, merkt man ohne genaueres Hinschauen nicht, ob die SSL Verbindung wirklich sicher ist. Diesen Fall haben Sie in diesem Laborversuch mittels interner CA umgesetzt.

Heisst das, wenn man mit seinem privaten Gerät ein wenig vorsichtig ist und rote Meldungen beachtet ist man sicher? Nein! Weit gefehlt! Lesen Sie den folgenden Abschnitt:

Im Jahre 2011 wurde die holländische Certification Authority DigiNotar Opfer eines Hackerangriffs. Dabei wurden über 500 betrügerische Zertifikate für Google, Yahoo!, Mozilla, Wordpress etc. ausgestellt.

Lesen Sie den Wikipedia Artikel zu obigem Vorfall: <https://en.wikipedia.org/wiki/DigiNotar>

Diese Zertifikate waren nicht von originalen Zertifikaten zu unterscheiden, da sie von einer legitimen Stelle ausgestellt wurden. Jedoch waren die jeweiligen Zertifikate nur für eine Domain gültig.

Zertifikatsfälschungen sind also durchaus möglich! Dazu kommen Certificate Authorities, die in Ländern mit restriktiver Überwachung als wir sie kennen, zu Hause sind. Es ist nicht zu kontrollieren, für wen diese CAs welche Zertifikate ausstellen.

Lesen Sie den folgenden Artikel. Dieser zeigt, dass das oben beschriebene Beispiel durchaus in der Praxis vorkommen kann.

<http://www.h-online.com/security/news/item/Trustwave-issued-a-man-in-the-middle-certificate-1429982.html>

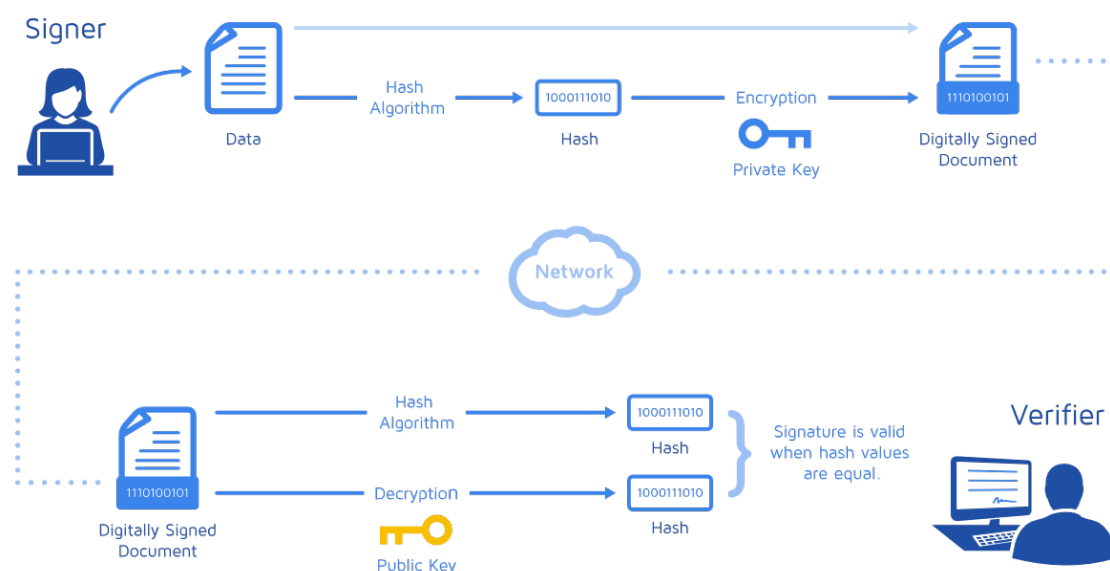
## 7. Anhang

### Anhang A – Ablauf Signieren

Folgende Grafik zeigt den Prozess einer digitalen Signatur. Der Unterschreibende wendet einen Hash-Algorithmus auf die zu signierenden Daten an und verschlüsselt diesen Hash mit seinem eigenen Private Key. Diese Message wird Signatur genannt und an die Daten angeheftet.

Der Empfänger der Nachricht verwendet den gleichen Hash-Algorithmus und rechnet diesen über dieselben Daten. Er erhält nun ein eigenes Resultat für den Hash-Algorithmus und vergleicht dieses Resultat mit dem vom Absender angehängten. Dabei wird der öffentliche Schlüssel des Signierenden verwendet, um den Hash zu entschlüsseln.

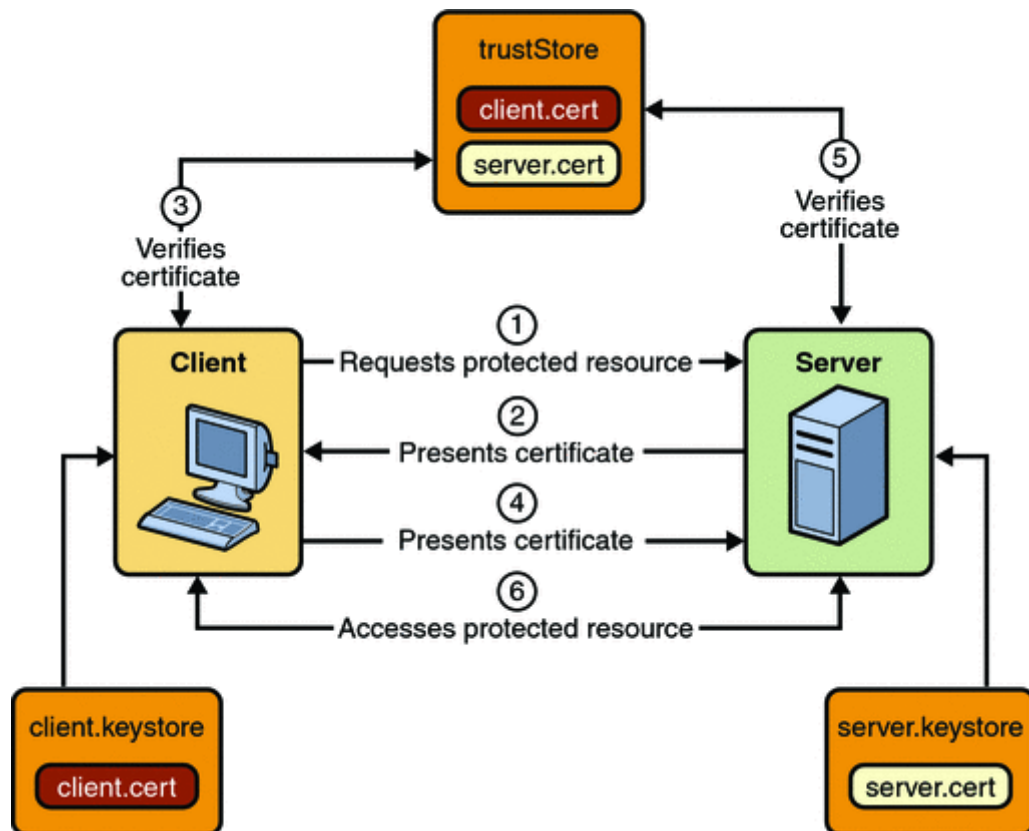
Stimmt der selbst gerechnete plus der mitgesendete Hash überein, kann davon ausgegangen werden, dass sich die Daten seit dem Unterzeichnen nicht verändert haben und die Nachricht wirklich vom Signierenden stammt.



**Abbildung 2:** Ablauf Signieren (<https://medium.com/@meruja/digital-signature-generation-75cc63b7e1b4>)

### Anhang B – Zertifikatsbasierte, gegenseitige Authentifizierung

Folgende Grafik zeigt den Prozess der zertifikatsbasierenden, gegenseitigen Authentifizierung. Dabei überprüft der Client zuerst das Zertifikat des Web Servers. Danach übergibt der Client sein Client Zertifikat dem Web Server, welcher dieser wiederum gegenüber der ausstellenden CA verifiziert. In der Grafik unten, wird die ausstellende CA für Web Server Zertifikat und Client Zertifikat als die gleiche dargestellt. Dies muss jedoch nicht zwingend der Fall sein und ist in den meisten Fällen eine andere CA.



**Abbildung 3:** Zertifikatsbasierte, gegenseitige Authentifizierung  
(<https://docs.oracle.com/cd/E19226-01/820-7627/bncbs/index.html>)

## **Notizen**