Zertifikate

S/MIME Zertifikate (Nutzerzertifikate)

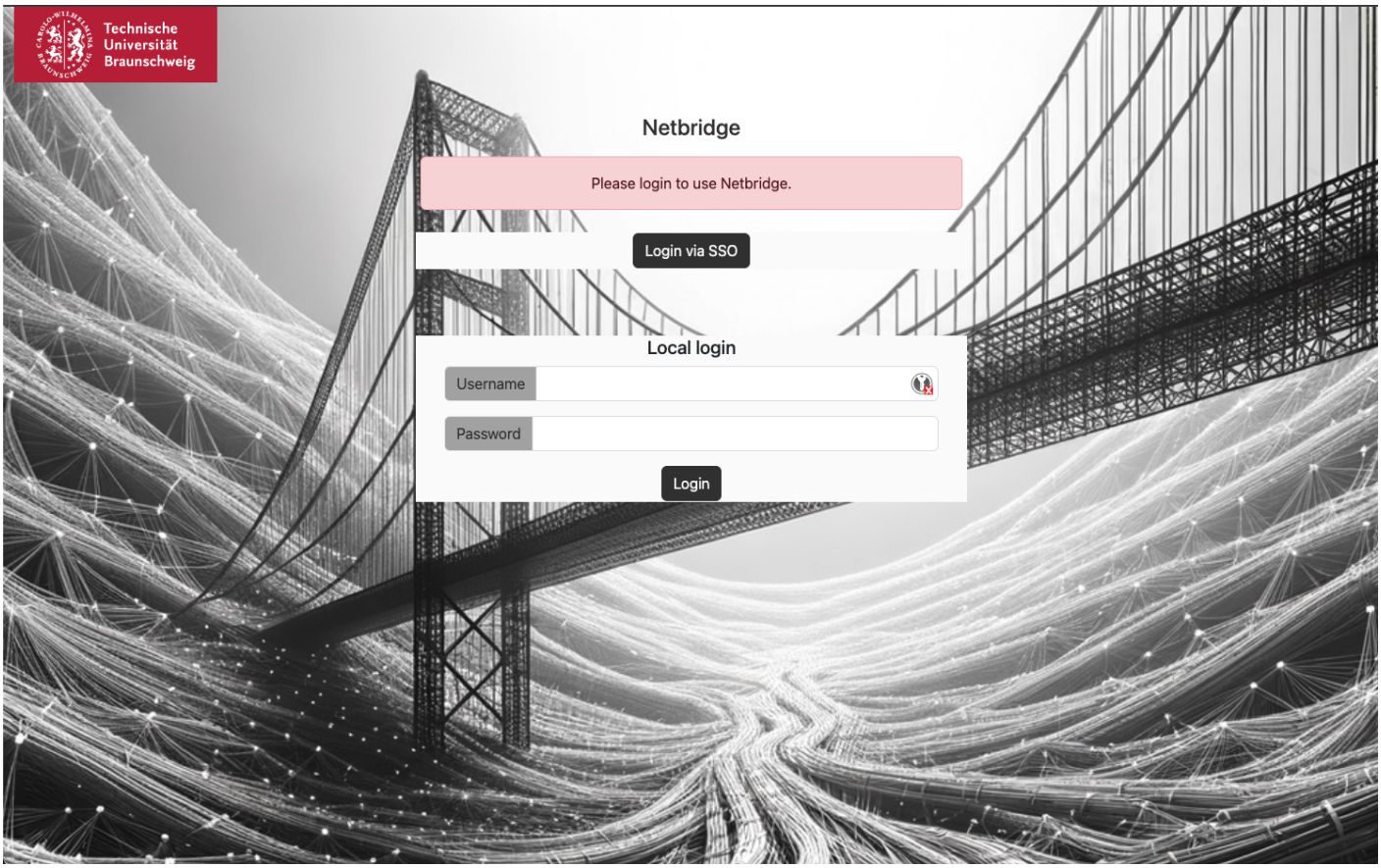
S/MIME Zertifikate (Nutzerzertifikate)

Allgemeines zu S/MIME Zertifikaten

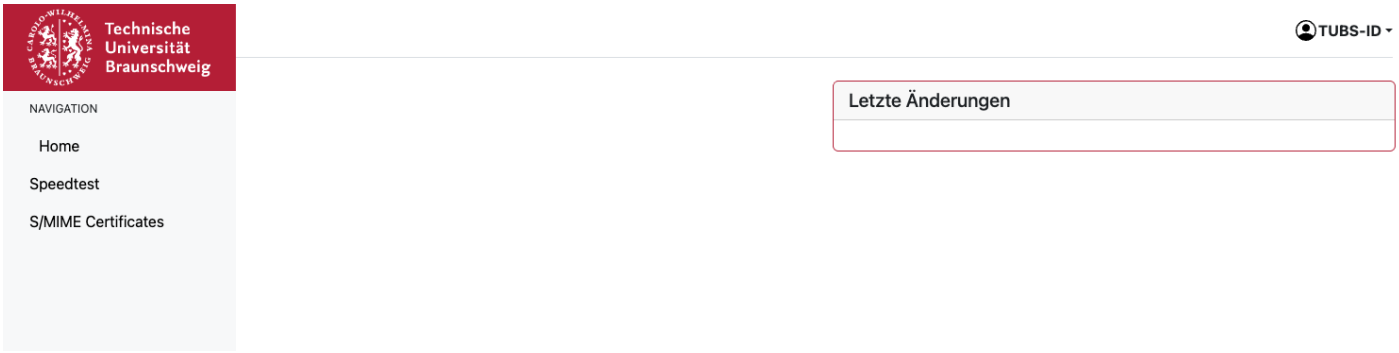
Mit einem S/MIME lässt sich nachweisen, wer der Absender einer E-Mail ist oder von welchem Autor eine Textdatei stammt. Diese sogenannten Nutzerzertifikate können weiterhin eingesetzt werden, um E-Mails zu verschlüsseln, eine Identität im Internet nachzuweisen oder aber sich an Webseiten oder Diensten anzumelden.
An der TU-Braunschweig bieten wir über die Public Key Infrastructure (PKI) des Deutschen Forschungsnetzes (DFN) und den Zertifikatsaussteller HARICA S/MIME Zertifikate an. Diese Zertifikate decken die meisten der oben genannten Anwendungsgebiete ab, eignen sich aber nicht zum Signieren von Dokumenten.

Übersicht der vorhanden S/MIME Zertifikate

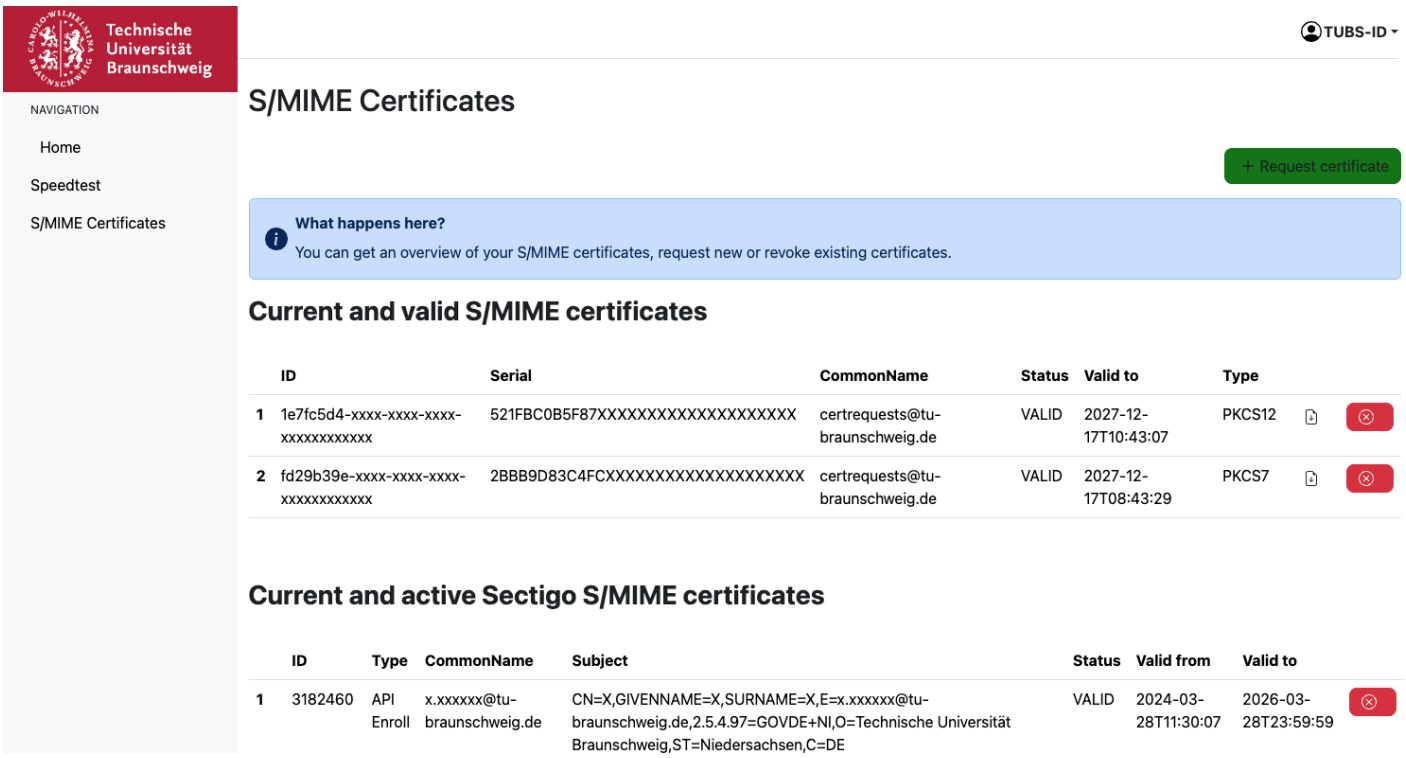
Um in die Zertifikatsverwaltung für S/MIME Zertifikate zu gelangen, navigieren Sie bitte zu folgender URL: https://netbridge.rz.tu-bs.de/ und loggen sich per SSO ein.



Sie gelangen auf eine Startseite, auf der Sie im linken Menü den Unterpunkt "S/MIME Certificates" auswählen.



Je nach Anzahl der vorhandenen Zertifikate kann es einige Zeit dauern, bis die Übersichtsseite geladen ist. Hier sehen Sie eine Auflistung der aktuellen und gültigen Zertifikate des aktuellen (HARICA) und des vorherigen Zertifikatsanbieters (Sectigo). Zurückgerufene oder abgelaufene Zertifikate werden nicht aufgelistet. Beide Listen sind unterschiedlich aufgebaut und an die jeweiligen Informationen angepasst. Ein Zertifikat von HARICA wird mit einer ID, der Seriennummer, dem sogenannten Common Name, dem Status, dem Ablaufdatum und dem Typ angegeben. Der Typ unterscheidet sich, je nachdem, ob Sie das Zertifikat mit Passwort (PKCS12) oder mit CSR (PKSC7) beantragt haben. Weiterhin können Sie HARICA Zertifikate herunterladen oder zurückrufen.



(Das rote "Entfernen-Symbol" bei den Sectigo-Zertifikaten wird noch entfernt.)
Mit einem Klick auf "+ Request certificate" gelangen Sie auf die Beantragungseite, wo sie zwischen Beantragung per Passwort und per CSR wählen können.

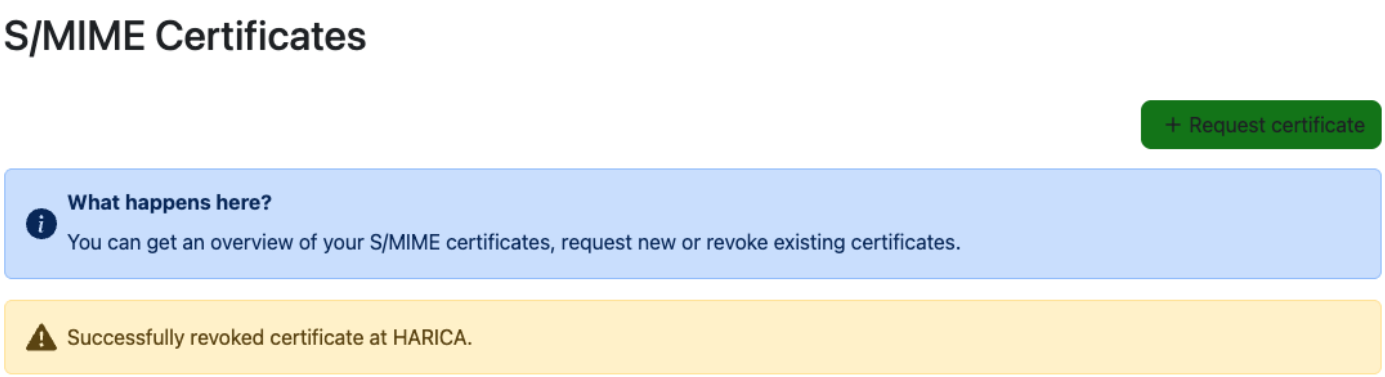
Beantragen eines S/MIME Zertifikates

Für die Beantragung von Nutzerzertifikaten stehen zwei Arten zur Verfügung, die Beantragung per Passwort und die Beantragung per CSR. Entscheidet man sich für die Beantragung mit Passworteingabe, so werden serverseitig ein CSR (Zertifikatsantrag) und ein privater Schlüssel erstellt. Der erstellte CSR wird dann bei dem Zertifikatsaussteller (HARICA) eingereicht. Der Zertifikatsaussteller erzeugt eine Datei, in der das digitale Zertifikat und die Zwischenzertifikate enthalten sind (P7B-Format). Mit dieser Datei, dem zuvor erzeugten privaten Schlüssel und dem eingegebenen Passwort wird dann ein P12-Zertifikat erstellt. Der private Key und alle Zwischendateien werden serverseitig nicht gespeichert. Lediglich das P12-Zertifikat wird gespeichert, um es dem Antragsteller jederzeit zum Download anzubieten. Der Antragsteller ist für die sichere Verwahrung des Passworts zuständig. Bei Passwortverlust kann das Zertifikat nicht neu eingebunden werden und verschlüsselte Daten gehen verloren. Eine detaillierte Anleitung zur Beantragung durch Passworteingabe erhalten Sie hier.

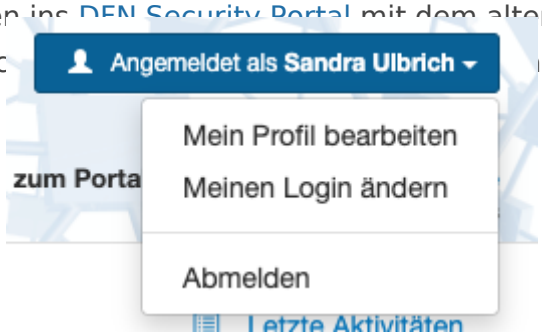
Wird die Beantragung per CSR gewählt, so werden der private Schlüssel und der Zertifikatsantrag auf dem Nutzersystem erstellt. Auch das Zusammenfügen des privaten Schlüssels und des P7B-Zertifikates erfolgt auf dem Nutzersystem. Lediglich das P7B-Zertifikat wird serverseitig gespeichert. Der Antragsteller ist für eine sichere Verwahrung des privaten Schlüssels verantwortlich. Bei Verlust des privaten Schlüssels kann das P12-Zertifikat nicht neu erstellt werden. Eventuell verschlüsselte Daten gehen verloren. Eine detaillierte Anleitung zur Beantragung nach Erzeugen eines CSRs erhalten Sie hier

Zurückziehen eines Zertifikates

Durch Klicken auf das rote Kreuz am Ende einer Zeile, können Sie ein Zertifikat zurückrufen. Nach erfolgreichem Rückruf gelangen Sie wieder auf die Übersichtsseite. Das Zertifikat ist aus der Auflistung verschwunden.
Sectigo Zertifikate lassen sich nicht mehr automatisiert zurückziehen. Dazu wenden Sie sich bitte per Email an noc@tu-braunschweig.de.



Ändern des Zertifikates für das DFN.Security-Portal

1. Bevor das alte Zertifikat ungültig wird, muss ein neues beantragt werden!
2. Einloggen ins DFN Security Portal mit dem alten Zertifikat!
3. Oben rec 

men und auf ***Meinen Login ändern*** klicken.
4. Es erscheint ein PopUp-Fenster. Hier bitte auf ***Zertifikat ändern*** klicken. Man erhält eine Email an die im Portal eingerichtete Email-Adresse mit weiteren Anweisungen.

Die Email-Adresse im Zertifikat muss mit der Email-Adresse übereinstimmen, mit der man im DFN.Security Portal angemeldet ist.

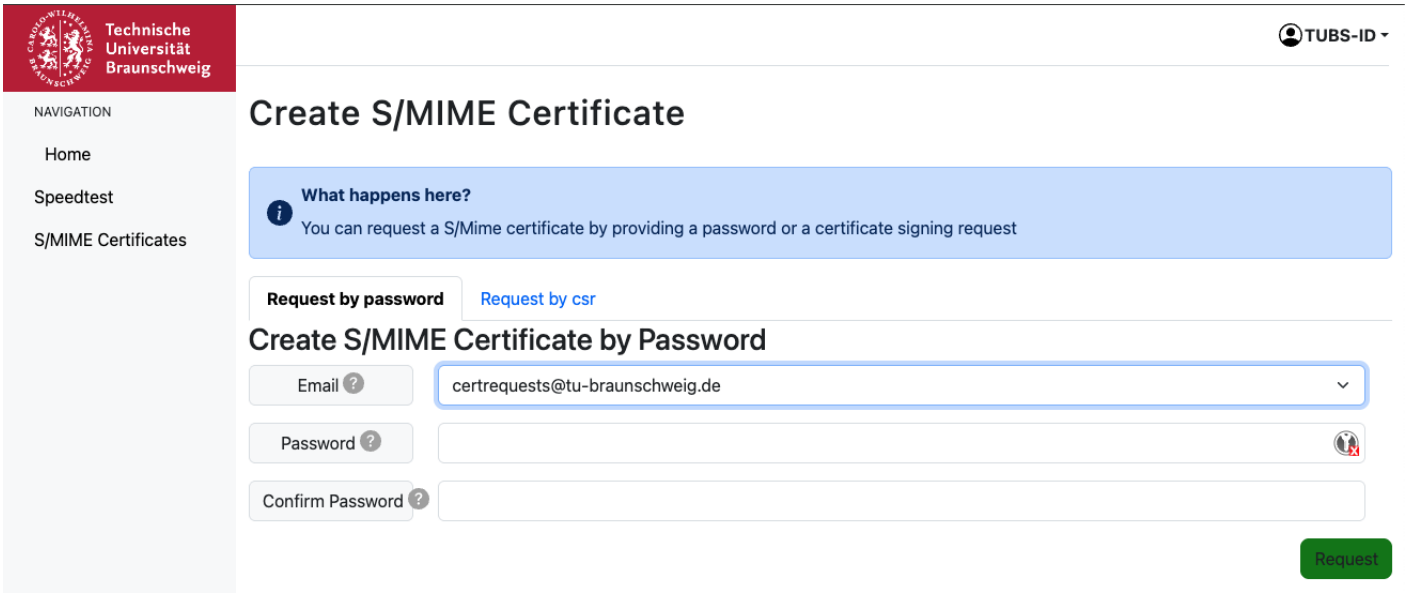
Verwenden Nutzerzertifikates

Eine Anleitung zum Einbinden des erhaltenen Zertifikates in Outlook finden Sie hier:

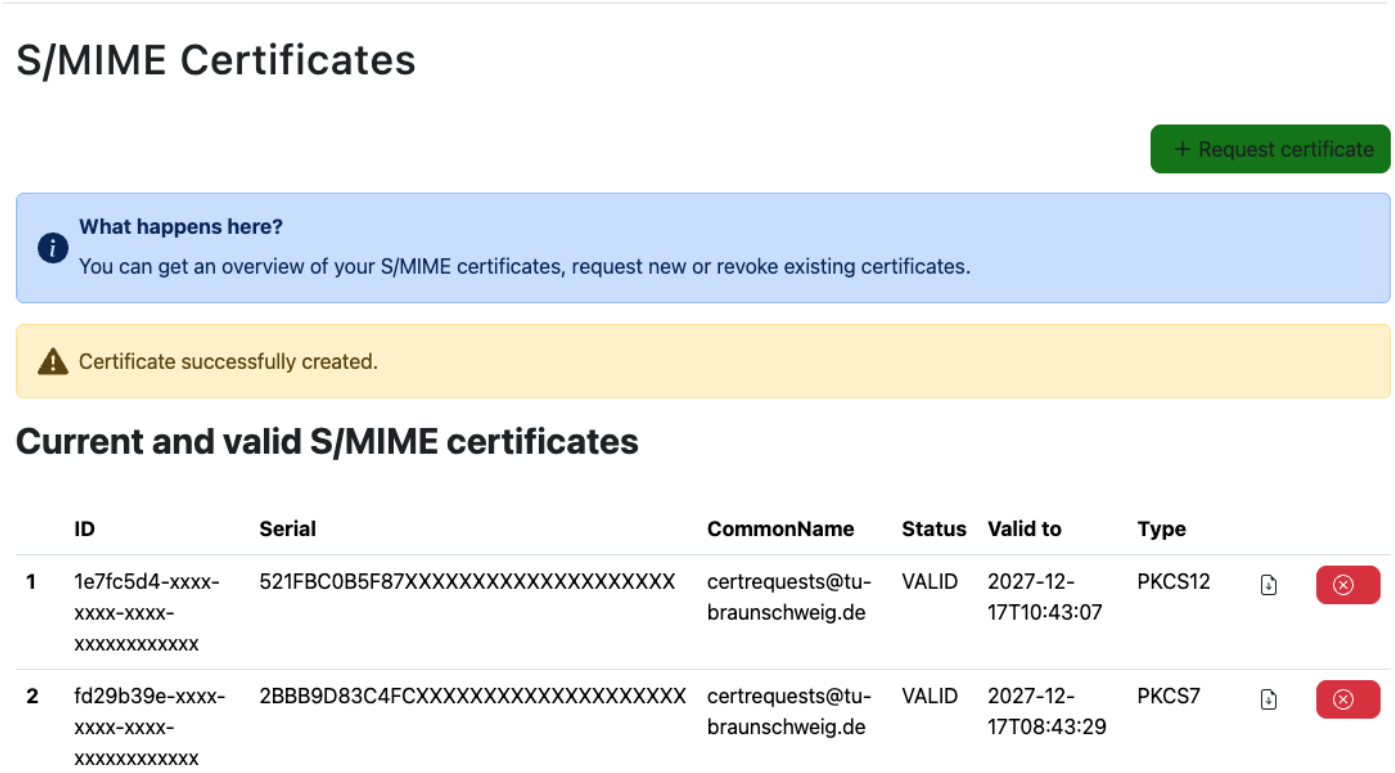
Zertifikatsimport S/MIME Outlook

Beantragung eines S/MIME Zertifikates durch Passworteingabe

Beantragung per Passwort



Um ein Zertifikat mit Eingabe eines Passwortes zu beantragen, wählen Sie bitte eine Email-Adresse aus. Nur personenbezogene Email-Adressen sind berechtigt und werden hier aufgelistet. Fehlt eine Ihrer Email-Adressen, wenden Sie sich bitte per Email an noc@tu-braunschweig.de. Im Folgenden vergeben Sie bitte ein Passwort und geben es ein zweites Mal ein. Bitte verwahren Sie dieses Passwort sicher. Sie benötigen es, um das Zertifikat später in den Zertifikatsspeicher ihres Betriebssystems zu importieren.



Nach erfolgreicher Beantragung, die einen Moment dauern kann, gelangen Sie wieder auf die Übersichtsseite. Dort können Sie ihr P12-Zertifikat herunterladen und sofort nutzen.

Beantragung mit einem Zertifikatsantrag (CSR)

Erstellen einer CSR-Datei

Haben Sie die Beantragung durch einen Zertifikatsantrag gewählt, müssen Sie zuerst einen solchen erstellen. Dazu gehen Sie bitte wie folgt vor.

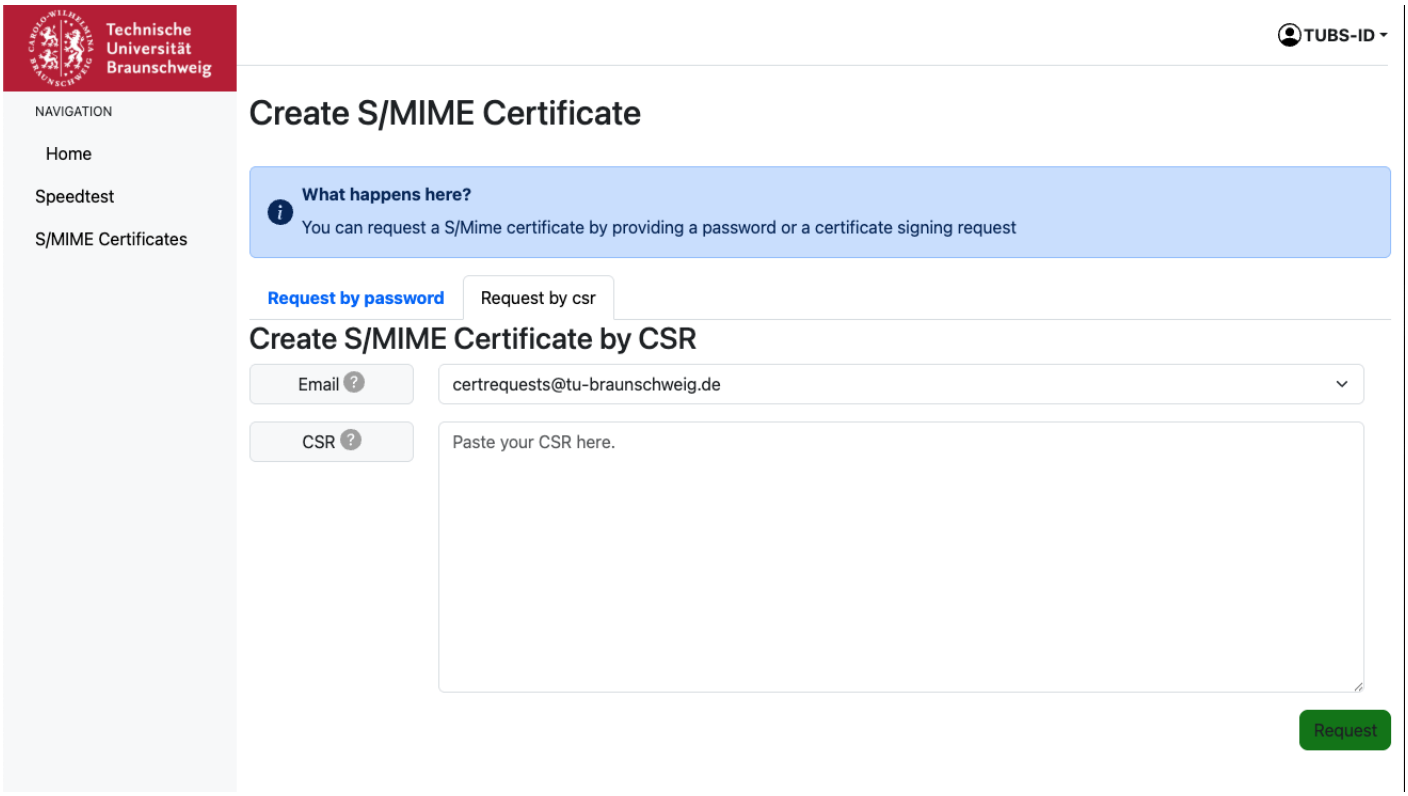
1. Öffnen Sie das Terminal auf Ihrem Computer.
2. Mit folgendem Befehl erstellen Sie eine CSR- Datei:
   openssl req -nodes -newkey rsa:4096 -keyout <certificate_key>.key -out <certificate_request>.csr
   Geben Sie bei Aufforderung folgende Daten ein:
   Country Name (2 letter code) [AU]: **DE**
   State or Province Name (full name) [Some State]: **Niedersachsen**
   Locality Name (eg, city) []: **Braunschweig**
   Organization Naem (eg, company) [Internet Widgits Pty Ltd]: **Technische Universitaet Braunschweig**
   Organizational Unit Name (eg, section) []: ***Ihr(e) Institut/ Abteilung***
   Common Name (e.g. server FQDN or YOUR name) []: ***Ihr Name laut Personalausweis***
   Email Address []: ***Ihre EMail-Adresse, für die sie das Zertifikat erstellen wollen***
   A challenge password []: ***Bitte leer lassen*** *
   An optional company name []: ***Bitte leer lassen***

   * Das Challenge Passwort dient dazu, das Zertifikat selbst direkt bei HARICA zu widerrufen. Ist ein Challenge Passwort gesetzt, können Sie es nicht mehr über die Selbstverwaltungsoberfläche zurückrufen.

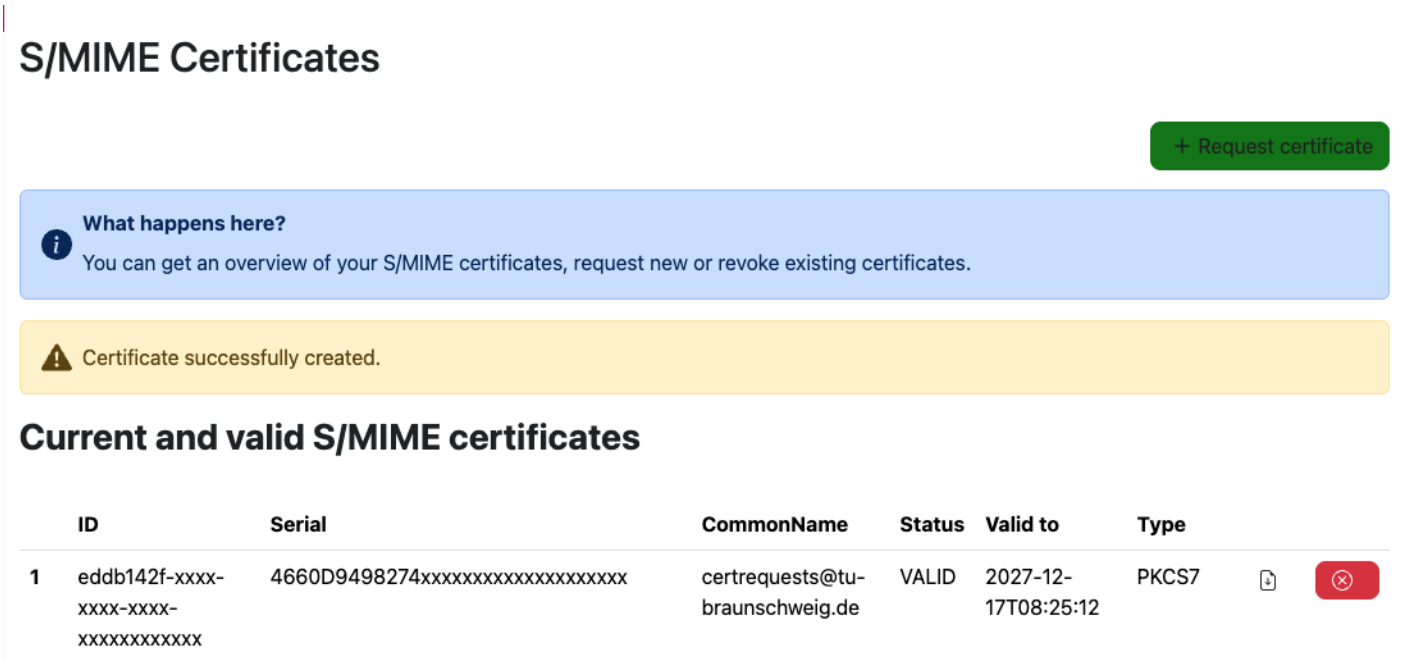
   Die beiden Dateien <certificate_key>.key und <certificate_request>.csr finden Sie nun in dem Verzeichnis, aus dem Sie den Befehl ausgeführt haben.

Beantragung des Zertifikates

Kehren Sie nun zur Beantragungsoberfläche zurück, wählen Sie eine entsprechende Email-Adresse aus und kopieren Sie den CSR in das dafür vorgesehene Feld.



Nach erfolgreicher Beantragung gelangen Sie auch hier wieder zurück auf die Übersichtsseite und können dort Ihr P7B-Zertifikat herunterladen. Um dieses Zertifikat nutzen zu können, müssen sie zusammen mit dem privaten Schlüssel zuerst ein P12-Zertifikat erstellen. Die Anleitung dazu finden Sie weiter unten.



Konvertieren der erhaltenen P7B-Datei

Nach erfolgreicher Beantragung eines Zertifikats mit eigenem CSR erhalten sie eine Datei mit der Endung ".p7b". Der Dateiname entspricht der ID des erstellten Zertifikates. Hierbei handelt es sich noch nicht um ein vollständiges Zertifikat. Sie müssen aus dieser Datei und dem bei Erstellung des CSRs erzeugten privaten Schlüssel erst noch eine Datei mit der Endung ".p12" erstellen. Dazu gehen Sie wie folgt vor:

1. Konvertieren der P7B-Datei in das PEM Format:
   openssl pkcs7 -print_certs -in <cert_id>.p7b -out <cert_id>.pem -inform der
2. Exportieren der PEM Datein in das P12 Format:
   openssl pkcs12 -export -certpbe PBE-SHA1-3DES -keypbe PBE-SHA1-3DES -nomac -inkey <certificate_key>.key -in <cert_id>.pem -out <cert_id>.p12

   Bei diesem Befehl werden Sie aufgefordert, ein Passwort einzugeben. Dieses Passwort schützt Ihr Zertifikat vor Missbrauch. Es wird gebraucht, um es in den Zertifikatsspeicher Ihres Betriebssystems zu importieren.

SSL Zertifikate (Serverzertifikate)

Allgemeines zu SSL-Zertifikaten

Diese Seite richtet sich an DV-Koordinatoren von Instituten und Einrichtungen, die SSL/TLS/HTTPS-Zertifikate (nach dem X.509-Standard) für den Betrieb von Servern benötigen. Die TU Braunschweig setzt Zertifikate aus der DFN-PKI ein. Diese sind für den dienstlichen Einsatz vorgesehen und können kostenfrei beantragt werden. Das Verwenden von Zertifikaten anderer Anbieter, wie z.B. Let's Encrypt ist nicht erlaubt.

> Die Beantragung von SSL-Zertifikaten wird über den KDD bereitgestellt. Die entsprechende Dokumentation finden Sie hier.

Zertifikatsantrag (CSR) erzeugen

Der folgende Abschnitt beschreibt Details zur Erzeugung von privatem Schlüssel und Zertifikatsantrag.

> Wenn Sie mehr Details benötigen, finden Sie eine umfänglichere Anleitung zu diesem Thema auf den Webseiten der DFN-PKI und darüber hinaus die vollständige Anleitung zu OpenSSL unter https://www.openssl.org/docs/

Zertifikatsantrag und privaten Schlüssel erzeugen

Wenn Sie noch kein SSL auf Ihrem Server einsetzen, verfahren Sie bitte, wie im Folgenden beschrieben. Zertifikatsanträge müssen in einem maschinell verarbeitbaren Format bei der RA eingereicht werden, dem Public Key Cryptography Standard Nr 10, PKCS#10, und werden bei Verwendung eines geeigneten Werkzeuges automatisch erstellt. Der Zertifikatsantrag setzt sich aus vier Teilen zusammen:

1. Dem „Subject Distinguished Name“, kurz DN, die teilweise durch die Policy vorgegeben sind, wie Landesbezeichnung, Country, C=de; die Ortsbezeichnung O=Technische Universitaet Braunschweig.
2. Den optionalen Attributen.
3. Einer digitalen Signatur mit dem privaten Schlüssel (der später auch auf dem Server abgelegt wird).
4. Einer Kennzeichnung des verwendeten Signaturalgorithmus.

Insbesondere wenn Sie mehrere Zertifikate beantragen, sollten Sie sich beispielsweise eine Konfigurationsdatei für OpenSSL erstellen, in der die wesentlichen Angaben bereits enthalten sind. Im Unterabschnitt „OpenSSL-Konfigurationsdatei“ ist ein Beispiel angegeben.

Falls Sie Ihren Dienst bisher ohne Zertifikat betreiben, müssen Sie ein entsprechendes Schlüsselpaar nach dem RSAVerfahren erzeugen. Das folgende openssl-Kommando führt die Schlüsselgenerierung, die Abfrage eines Passwortes für diesen Schlüssel, die Attribut-Abfrage und die anschliessende Erstellung des Zertifikatsantrages durch:

```
openssl req -config myopenssl.conf -newkey rsa:4096 -outform pem -out certreq.pem
```

Die einzelnen Befehlsteile bedeuten folgendes:

1. openssl: Programmname des Openssl-Kommandos, evtl. muss der komplette Pfad z.B. /usr/local/bin/openssl angegeben werden.
2. req: erstellt einen PKCS#10-Antrag
3. -config: verweist ggf. auf die ssl-Konfig-Datei für Voreinstellungen, siehe Abschnitt „OpenSSL Konfigurationsdatei“
4. -newkey rsa:4096: neues Schlüsselpaar mit RSA und einer Key-Länge von 4096 Bit
5. -outform PEM: Privacy Enhanced Mail Ausgabeformat
6. -out certreq.pem: speichert den Antrag unter certreq.pem. Achtung: die Konfigurationsdatei enthält die Direktive, den neu zu erstellenden privaten Schlüssel in server-key.pem zu speichern. Sollen mehrere Anträge hintereinander erstellt werden, müssen die Dateien nach jedem Durchgang umbenannt werden, um ein Überschreiben zu vermeiden.

Zertifikatsantrag ohne erneute Erzeugung eines privaten Schlüssels erzeugen

Setzen Sie bereits SSL auf Ihrem Server ein und möchten Sie den dafür verwendeten private key weiter verwenden, so erzeugen Sie den Zertifikatsrequest wie in diesem Abschnitt beschrieben.

Läuft der Dienst mit einem selbst signierten Zertifikat und haben Sie bereits einen RSA-Schlüssel erstellt mit einer Länge von 2048 Bit, können Sie diesen Schlüssel für Ihr neues Zertifikat weiter verwenden. Mit folgendem Kommando kann dann eine Zertifizierungsanfrage für den vorhandenen Key (server.key) erstellt werden:

```
openssl req -new -config myopenssl.conf -key server-key.pem -keyform PEM -outform pem -out certr
```

Mit -keyform PEM bzw. -keyform DER sollten Sie das Format des vorhandenen Schlüssels berücksichtigen.

Entfernen der Passphrase für automatischen Start von Diensten

Beachten Sie bitte auch, dass Sie den Dienst den Sie mit dem so beantragten Zertifikat ausstatten, beim Start/Neustart stets die Angabe der Passphrase erfordert. Um einen Dienst auch ohne manuelle Eingabe der Passphrase starten zu können, muss man ggf. auf die Passphrase verzichten. Für den Betrieb von Diensten, die auch nach einem ungeplanten Neustart des Servers (z.B. Stromausfall, Softwarefehler) sofort und ohne manuelle Intervention erreichbar sein sollen, muss die Passphrase entfernt werden. Darüber hinaus setzen manche Produkte voraus, dass man zuvor die Passphrase entfernt.

Das Entfernen der Passphrase ist kritisch und aus Sicht der IT-Sicherheit nicht empfehlenswert. Das Zertifikat auf dem Server ist dann ungeschützt hinterlegt und kann bei einer Kompromittierung des Servers besonders einfach kopiert und für Angriffe genutzt werden. Grundsätzlich - ob mit oder ohne Passphrase - gilt: Sollte der Server gehackt werden, so ist das Zertifikat samt private key nicht mehr vertrauenswürdig. Das Zertifikat muss dann umgehend gesperrt werden. Sie müssen in einem solchen Fall sowohl einen neuen private key generieren, als auch ein neues Zertifikat beantragen.

Das Entfernen der Passphrase kann mit dem folgenden Kommando durchgeführt werden:

```
openssl rsa -in server-key.pem -out server-key_nopass.pem
```

In Abhängigkeit von der Version von OpenSSL kann die genaue Kommandosyntax leicht abweichen.

OpenSSL Konfigurationsdatei

In manchen der zuvor aufgeführten Kommandos wird eine Konfigurationsdatei referenziert:

```
  ... -config myopenssl.conf ...
```

(ohne Alternative DNS Names)

Sollte der Server nur über genau einen Hostnamen erreicht werden, könnte der Inhalt einer solchen Datei wie weiter unten beschrieben aussehen.

(mit Alternative DNS Names)

Sollte der Server mehr als einen Namen (z.B. CNAMES/Aliase) haben, so können/sollten alle Namen entsprechend im Zertifikat hinterlegt werden. Zu diesem Zweck benutzen Sie bitte das Eingabefeld „Subject Alternative Names“ des Webformulars wie oben beschrieben.

Alternativ, aber aufwendiger, lassen sich die Alternativnamen auch über die Konfigurationsdatei angeben. Dazu ändern Sie, wie weiter unten beschrieben, am Ende die Beispiel-Hostnamen unterhalb von “[subject_alt_name]„ in die Aliase um, über die der Server zusätzlich zum eigentlichen Hostnamen (wird bei der Erstellung des Zertifikatsantrags erfragt oder kann bei „commonName“ angegeben werden) noch erreicht werden soll. Außerdem können Sie weitere Namen in entsprechender Weise hinzufügen.

Zertifikat anzeigen

Der Datei, die Sie per E-Mail aus der CA zugesendet bekommen, können Sie die in ihr enthaltenen Informationen nicht ansehen. Um die Zertifikatsinformationen einer Zertifikatsdatei anzeigen zu können, verwenden Sie:

```
  openssl x509 -text -noout -in server-crt.pem
```

Zertifikat sperren

Durch die Beantragung des Zertifikates akzeptieren Sie Handlungsanweisungen, dass Sie Zertifikate bei Bedarf auch wieder sperren. Gründe für eine Sperrung könnten sein:

- Ein Zertifikat soll ersetzt oder erneuert werden (das vorherige ist zu sperren).
- Der private Key wurde offen gelegt.
- Der Server wurde kompromittiert.

Durch die Sperrung eines Zertifikats wird dessen eindeutige Seriennummer auf einer Sperrliste/Widerrufsliste (certificate revocation list, CRL) veröffentlicht. Bezüglich der Umsetzung der Sperrung von Zertifikaten bei Clients ist zur Zeit in manchen Fällen noch einiges im Argen. Im Idealfall ist bei einem gesperrten Zertifikat die weitere Verwendung des Zertifikates aufgrund der Prüfung der Sperrliste seitens der Clients verhindert.

Die Sperrung eines Zertifikats lässt sich nicht rückgängig machen. Sollte ein Zertifikat irrtümlich gesperrt worden sein, müssen Sie ein neues Zertifikat beantragen.

Für die Durchführung einer Sperrung gibt es drei Möglichkeiten:

1. Sperren über Webformular: Hier müssen Sie die sogenannte „certificate ID“ angeben. Diese ID finden Sie als „Self-Enrollment Certificate ID“ in der ersten E-Mail, die Sie nach Beantragung über das Webformular erhalten haben (Betreff: GÉANT TCS: Awaiting approval for certificate *FQDN*). Weiterhin benötigen Sie die „Annual Renewal Passphrase“, die Sie bei Beantragung über das Webformular vergeben haben (siehe oben).
2. Sperrantrag im KDD über den entsprechenden Reiter (verfügbar ab Ende 2022)
3. Kontakt zum IT-Service-Desk. Geben Sie die Antragsnummer und/oder die Seriennummer und den Common Name an.

Installation eines Zertifikats für Apache

Die Konfiguration der Zertifikate ist prinzipiell bei allen Apache-Installationen sehr ähnlich – hier wird nur grundlegend dargelegt, wie dies vorzunehmen ist. Für versionsspezifische Abweichungen konsultieren Sie bitte die Hilfe über die man-pages Ihrer Linux-Distribution. Folgende Dateien werden benötigt:

- common_name.pem: die Zertifikatsdatei, die Sie über den KDD als "PEM" heruntergeladen haben
- common_name_interm.cer: Diese Datei enthält alle „über“ dem Server liegenden Zertifikate bis zum Root Zertifikat des Zertifikatanbieters. Diese Datei laden Sie als "Bundle" über den KDD herunter.
- privkey.pem: Der private Schlüssel des Servers, den Sie zur Erstellung des Zertikatantrags (CSR) genutzt haben.

Zur Installation sind die folgenden Schritte nötig:

1. Stoppen Sie den eventuell gestarteten Apache Webserver Dienst (z.B. „service httpd stop“).
2. Kopieren Sie die drei Zertifikatsdateien in einen Unterordner des Apache Webservers (z.B. :/etc/apache2/ssl.key/).
3. Editieren Sie bzw. legen Sie eine neue V-Hosts-Datei unter “/etc/apache2/vhosts.d„ an. In der Konfigurationsdatei des V-Hosts werden unter anderem die Zertifikate mit angegeben.
4. Wichtig ist, dass „ServerName“ genau mit dem Namen angegeben wird, für den Sie das Zertifikat beantragt haben.
5. Den Aufbau der Datei (Beispielkonfiguration) finden Sie weiter unten.
6. Aktivieren Sie ggf. noch MOD_SSL für den Apache-Webserver.
7. Starten Sie den Apache neu.
8. Besuchen Sie die Startseite Ihres Servers, kontrollieren Sie die Logfiles.

Installation eines Zertifikats für NGINX

Folgen Sie prinzipiell den Anweisungen zur Installation bei Apache (siehe oben). Lediglich die Dateien, die Sie herunterladen müssen unterscheiden sich.

- common_name_cert.cer: Eine Zertifikatsdatei nur für Ihren Server, die Sie von einem der Links in der E-Mail bezogen haben. Für NGINX nutzen Sie bitte den ersten Link in der E-Mail („as Certificate only, PEM encoded“).
- common_name_interm.cer: Diese Datei enthält alle „über“ dem Server liegenden Zertifikate vom Root Zertifikat des Zertifikatanbieters abwärts. Einen Link zu dieser Zertifikatskette erhalten Sie sie mit der E-Mail, in der auch die Links zu verschiedenen Zertifikatsformaten enthalten sind (Betreff:„GÉANT TCS certificate information: *Common name ihres Servers*“). Bitte nutzen Sie dazu den vorletzten Link („as Root/Intermediate(s) only, PEM encoded“) in der E-Mail, die sie erhalten haben.
- privkey.pem: Der private Schlüssel des Servers, den Sie zur Erstellung des Zertikatantrags (CSR) genutzt haben.

Fügen Sie nun die beiden erhaltenen Zertifikatsdateien mittels „cat“ im Linux Terminal oder einem ähnlichen Programm zusammen:

cat common_name.cer common_name_interm.cer > common_name_cert_chain.cer

Bitte nutzen Sie zum Zusammenfügen der beiden Dateien keine Office-Programme, wie MS Office, Open Office oder Libre Office. In Ihrer NGINX Konfigurationsdatei geben Sie nun den Pfad zu der neu erstellten Datei (common_name_cert_chain.cer) als SSL_CERTIFICATE_PATH und den Pfad zur privaten Schlüsseldatei Ihres Servers (privkey.pem) als SSL_KEY_PATH an.

Installation eines Zertifikats (allgemein)

Für die Konfiguration eines Zertifikates sollten Sie sich zuerst mit den Konfigurationsdateien Ihres Webservers vertraut machen. Diese finden Sie in der Regel unter */etc/<Produkt>/* . *(/etc/nginx, /etc/apache2, /etc/httpd, ...)*.
Welche Änderungen Sie zur Absicherung Ihres Webservers an Ihrer Konfigurationsdatei vornehmen müssen, können Sie im kostenlosen Konfigurationsgenerator von Mozilla (https://ssl-config.mozilla.org/) sich anzeigen lassen. Nachdem Sie im Konfigurationsgenerator Ihre Einstellungen ausgewählt haben, können Sie mit der angezeigte Konfiguration Ihre Webserver-Konfigurationsdatei anpassen.
Die Webserver-Konfigurationsdatei finden Sie direkt in Ihrem Konfigurationsordner, meist werden auch hier Produktnamen als Dateinamen verwendet. *(apache2.conf, htttpd.conf, nginx.conf, ...)*
Nach der Anpassung Ihrer Konfiguration sollten Sie die verwendeten Platzhalter *(/path/to/...)* an Ihre lokalen Gegebenheiten anpassen.
Anschließend können Sie die Syntax Ihrer Konfiguration mit einem Test *(apachectl configtest , nginx -t , ...)* überprüfen und bei Erfolg den Webserver neu starten.

Skizze einer Beispielkonfiguration: vhost-ssl.conf

```
[...]
<VirtualHost _default_:443>
DocumentRoot "/srv/www/htdocs"
ServerName example.inst.tu-bs.de:443
[...]
SSLEngine on
SSLCertificateFile etc/apache2/ssl.key/common_name.pem
SSLCertificateKeyFile /etc/apache2/ssl.key/privkey.pem
SSLCertificateChainFile /etc/apache2/ssl.key/common_name_interm.cer
[...]
</VirtualHost>
[...]
```

OpenSSL Konfigurationsdatei ohne Alternativnamen

```
#
# myopenssl.conf
#
HOME                        = .
RANDFILE                = $ENV::HOME/.rnd
[ req ]
default_bits                = 4096
default_keyfile         = server-key.pem
```

```
distinguished_name        = req_distinguished_name
attributes                = req_attributes
string_mask = nombstr
req_extensions = v3_req

[ req_distinguished_name ]
countryName = Laendername (bitte nicht aendern)
countryName_default = DE
countryName_min = 2
countryName_max = 2

0.organizationName = Name der Organisation (bitte nicht aendern)
0.organizationName_default = Technische Universitaet Braunschweig

0.organizationalUnitName = Offizieller Einrichtungsname (ohne Umlaute)
0.organizationalUnitName_default =

1.organizationalUnitName = Optional Abteilung oder AG im Institut
1.organizationalUnitName_default =

stateOrProvinceName = Bundesland (ausgeschrieben)
stateOrProvinceName_default = Niedersachsen
localityName = Locality Name - Stadt (Sitz der TU Braunschweig)
localityName_default = Braunschweig

commonName  = Voller DNS-Name unter dem der Service erreichbar ist
commonName_max = 64

emailAddress = Support E-Mail Adresse der Einrichtung (bevorzugt)
emailAddress_max = 40
emailAddress_default =

[ req_attributes ]
[ v3_req ]
basicConstraints = CA:FALSE
keyUsage = nonRepudiation, digitalSignature, keyEncipherment
```

OpenSSL Konfigurationsdatei mit Alternativnamen

```
#
# myopenssl.conf
#
HOME                      = .
RANDFILE                = $ENV::HOME/.rnd
[ req ]
default_bits              = 4096
default_keyfile         = server-key.pem
distinguished_name        = req_distinguished_name
```

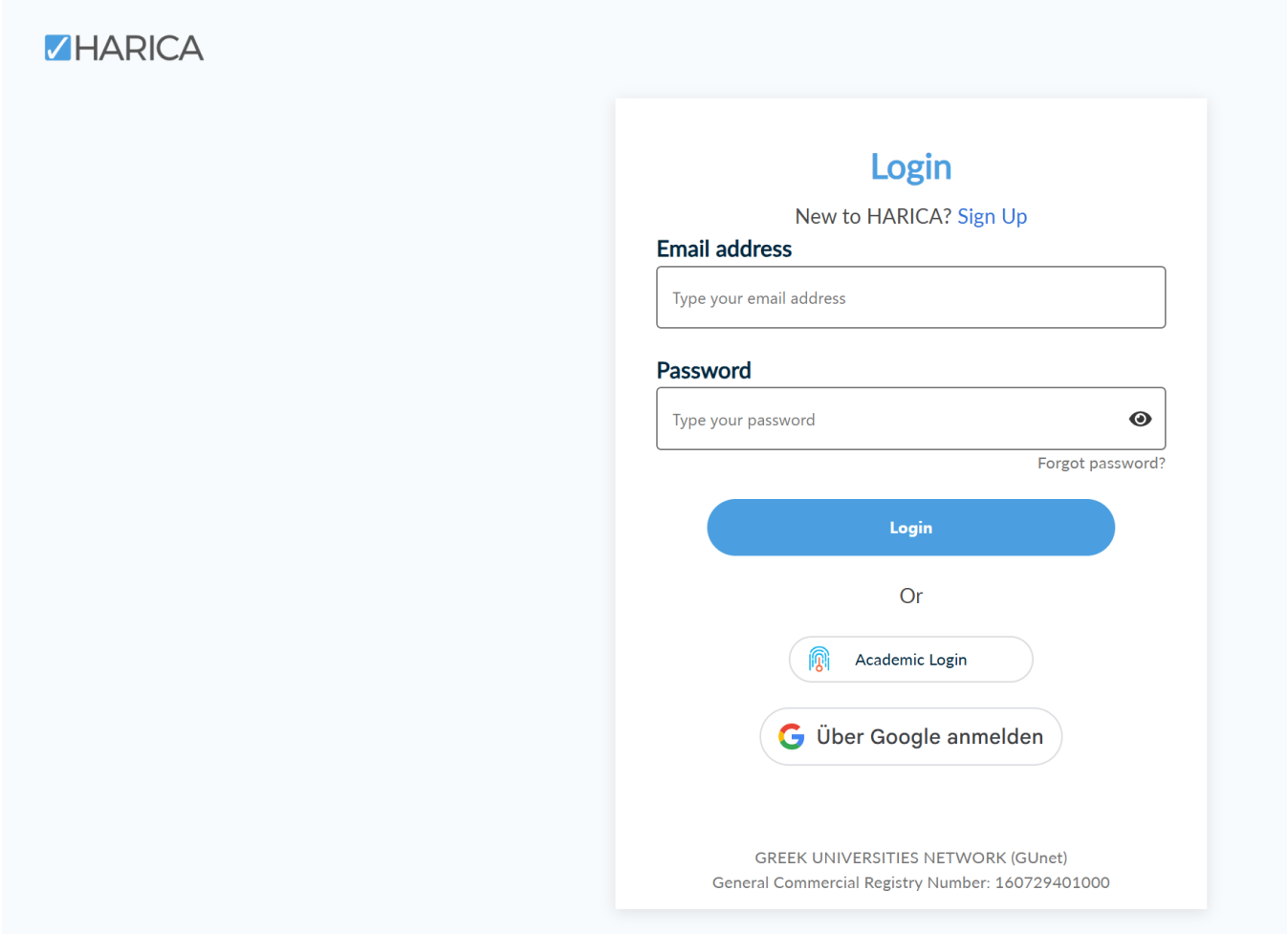
```
attributes                = req_attributes
string_mask = nombstr
req_extensions = v3_req

[ req_distinguished_name ]
countryName = Laendername (bitte nicht aendern)
countryName_default = DE
countryName_min = 2
countryName_max = 2

0.organizationName = Name der Organisation (bitte nicht aendern)
0.organizationName_default = Technische Universitaet Braunschweig

0.organizationalUnitName = Offizieller Einrichtungsname (ohne Umlaute)
0.organizationalUnitName_default =

1.organizationalUnitName = Optional Abteilung oder AG im Institut
1.organizationalUnitName_default =

stateOrProvinceName = Bundesland (ausgeschrieben)
stateOrProvinceName_default = Niedersachsen
localityName = Locality Name - Stadt (Sitz der TU Braunschweig)
localityName_default = Braunschweig

commonName  = Voller DNS-Name unter dem der Service erreichbar ist
commonName_max = 64

emailAddress = Support E-Mail Adresse der Einrichtung (bevorzugt)
emailAddress_max = 40
emailAddress_default =

[ req_attributes ]
[ v3_req ]
basicConstraints = CA:FALSE
keyUsage = nonRepudiation, digitalSignature, keyEncipherment
subjectAltName=@subject_alt_name

[ subject_alt_name ]
DNS.1=*.tu-bs.de
DNS.2=*.tu-braunschweig.de
```

Validierung von externen Domains

Wenn Sie ein SSL-Zertifikat über die DFN-PKI für eine externe Domain (z. B. `example.org`) beantragen möchten, die nicht auf `tu-braunschweig.de` oder `tu-bs.de` endet, müssen Sie die Inhaberschaft der Domain gegenüber dem NOC (Netz- und Systembetrieb der TU Braunschweig) nachweisen. Dieser Prozess erfolgt in zwei Schritten**:**

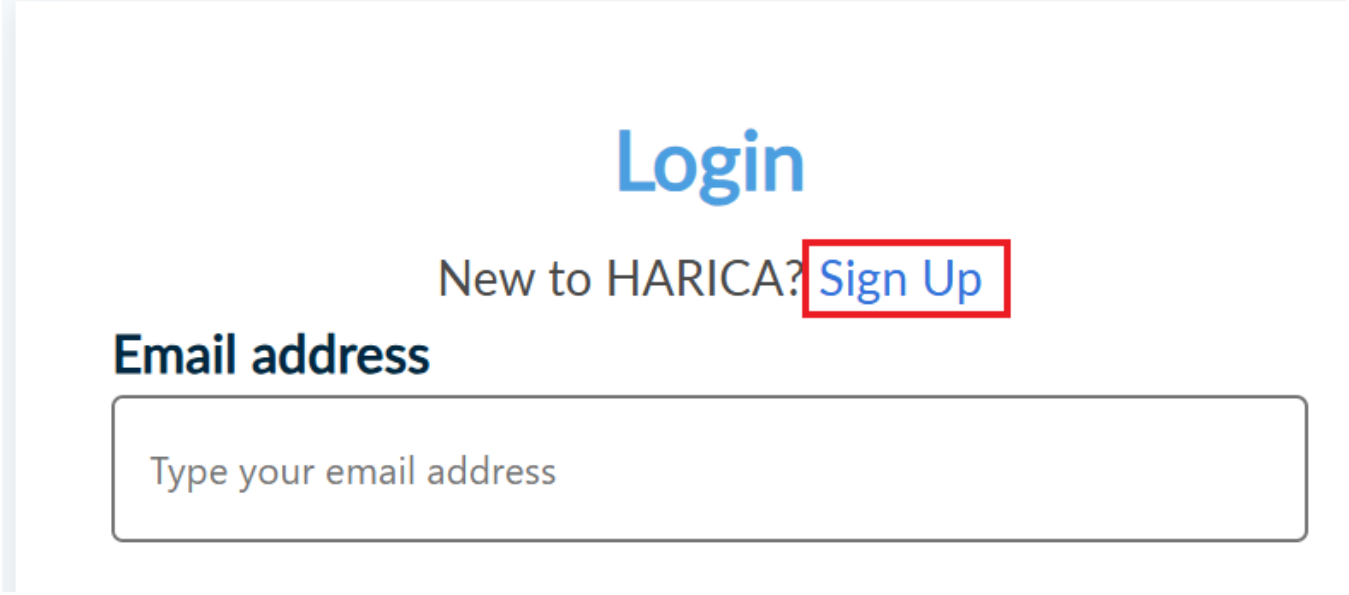
1. Account bei Harica anlegen

1. Öffnen Sie im Browser die Seite

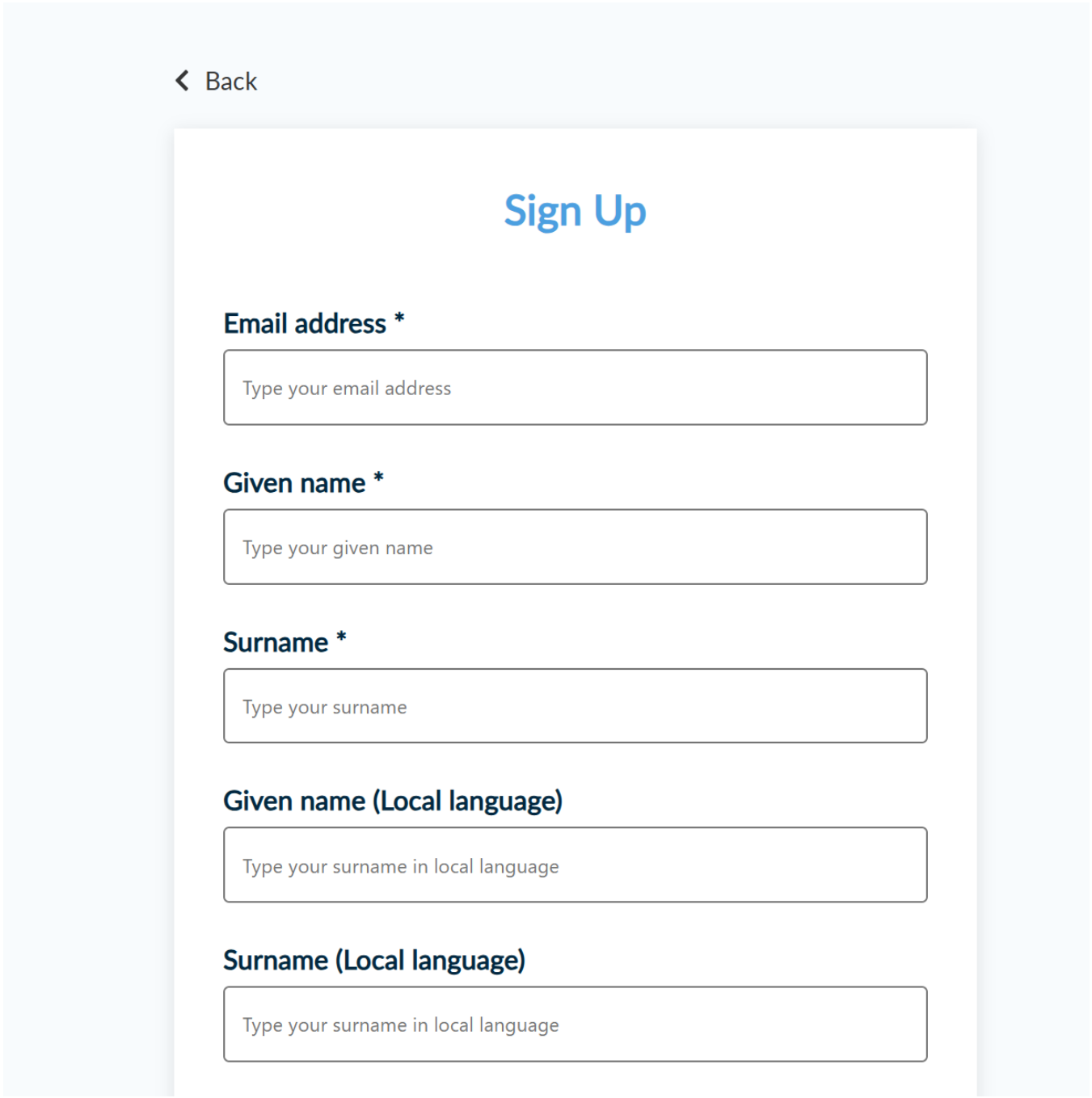


2. Klicken Sie auf „Sign Up“ und wählen Sie die normale E-Mail-Registrierung **(nicht**

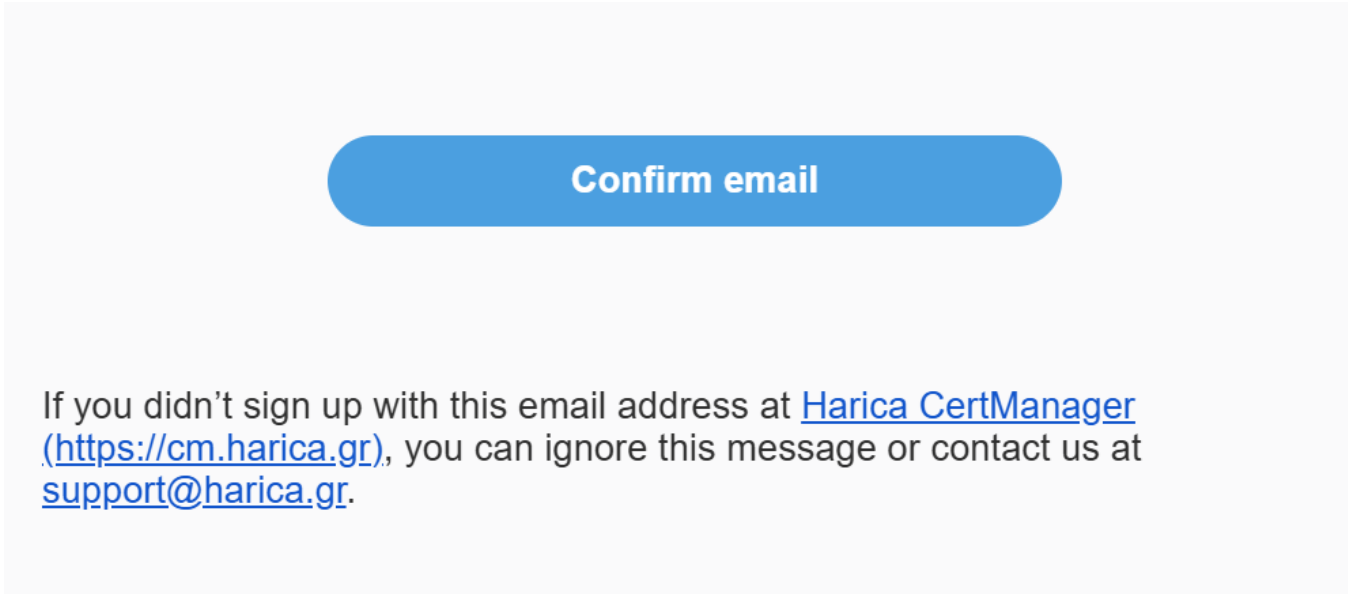
Academic-Login oder Google)



3. Füllen Sie die Felder (Name, E-Mail, Passwort) aus und bestätigen Sie. Nutzen Sie bitte eine TU-Email, damit Ihr Account der TU zugeordnet werden kann.



4. Öffnen Sie Ihr Email-Postfach und klicken Sie auf den Bestätigungslink in der Harica-Mail.



2. E-Mail an das NOC der TU Braunschweig

Senden Sie eine E-Mail an **noc@tu-braunschweig.de**, um die Freigabe Ihrer externen Domain für die DFN-PKI zu beantragen.

Inhalt der E-Mail:

- Eine **Kopie der Rechnung oder des Registrierungsvertrags**, aus der hervorgeht, dass die Domain der **TU Braunschweig** gehört
  *(Domains auf Privatpersonen sind **nicht zulässig**!)*
- Die **E-Mail-Adresse**, mit der Sie sich bei HARICA registriert haben
- Die **Challenge-Response-E-Mail-Adresse**, über die HARICA die automatisierte Validierung durchführt

Zulässige Challenge-Response-E-Mail-Adressen:

Für die Domainvalidierung per E-Mail sind ausschließlich folgende generische Adressen zulässig:

- `hostmaster@<domain>`
- `postmaster@<domain>`
- `admin@<domain>`
- `administrator@<domain>`
- `webmaster@<domain>`

Nach Prüfung der Angaben kontaktiert das NOC die HARICA-CA, um die Domainfreigabe anzustoßen. Der weitere Ablauf erfolgt über das Challenge-Response-Verfahren der CA direkt mit der angegebenen Validierungsadresse.