Konfiguration Apache Webserver für SSL

Voraussetzung für dieses Tutorial ist ein installierter Apache HTTPD Webserver, wir verwenden die zurzeit aktuelle Version 2.4.16. Dieser beinhaltet OpenSSL in der Version 1.0.2d.

Hinweise:

1. Wir setzen einen anhand der Anleitung „Installation & Konfiguration Apache Webserver + PHP“ voraus – das gilt insbesondere für die Verzeichnisstruktur. Sie können diese selbstverständlich an Ihre Gegebenheiten anpassen.
2. Wir schreiben unsere Tutorials immer im LIVE-Betrieb, d.h. wir schreiben nie blind ohne Ausprobieren eine Anleitung. Hier beim Erzeugen eines Zertifikats ist das natürlich schwierig – wir können nicht willkürlich eins für „meineschule.at“ erzeugen lassen, da wir diese Domäne nicht besitzen. Das Zertifikat wird daher auf „skrejci.com“ ausgestellt. Passen Sie auch hier den Domänennamen jeweils dem Ihrigen an!

Wir bei schultermine.com verwenden, wie viele andere Webseiten auch, Zertifikate von StartCom. Zurzeit sind wir Class 2 zertifiziert.

Wird ein Zertifikat für nur eine Domäne benötigt, reicht eine Class 1 Zertifizierung. Diese ist kostenlos, es genügt eine einfache Registrierung – und daher sehr praktisch für ebendiese Zwecke!

Dieses Tutorial führt Sie nun in fünf Schritten zu einem Webserver, der Webseiten an SSL-Verbindungen bereitstellt:

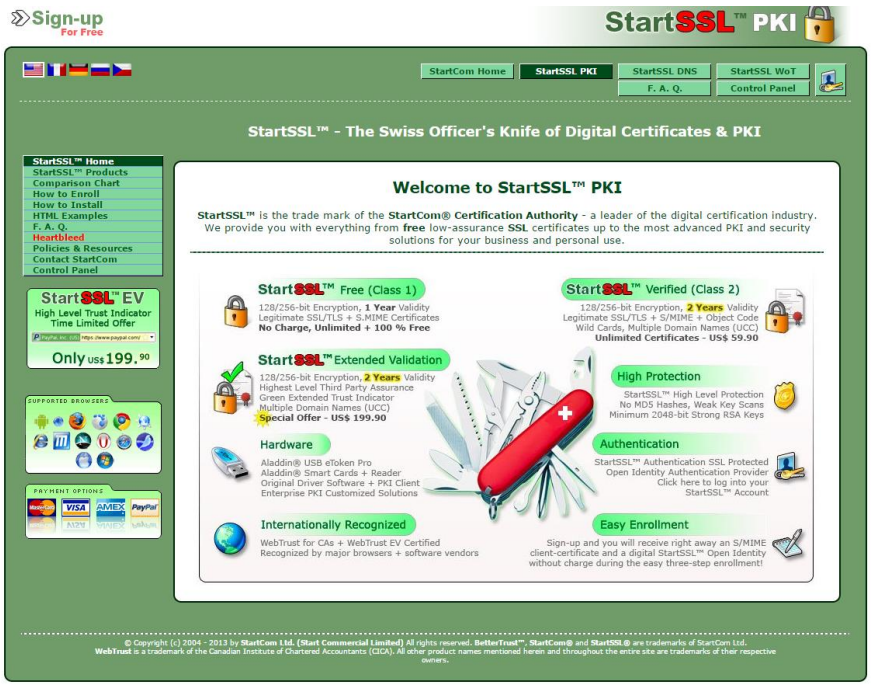
1. StartCom: Email-Adresse validieren
2. StartCom: Domäne validieren
3. StartCom: Private Key generieren und Zertifikat erstellen
4. Apache für SSL konfigurieren
5. Firewall für eingehende SSL-Verbindungen konfigurieren

Um eine SSL-Verbindung auf Ihrem Webserver bereitstellen zu können, benötigen Sie…

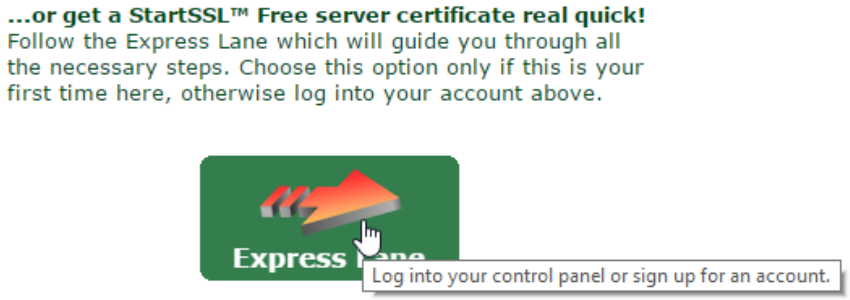
- ein Zertifikat (ssl.crt),
- einen zugehörigen privaten Schlüssel (ssl.key),
- ein Serverzertifikat (Zertifikatskette) sowie
- ein Stammzertifikat (Certificate Authority, CA).

(Die Dateinamen in Klammern sind selbstverständlich beliebig – wir verwenden, wie im Folgenden von StartCom vorgeschlagen, eben diese angegebenen.)

Navigieren Sie zur Seite startssl.com.

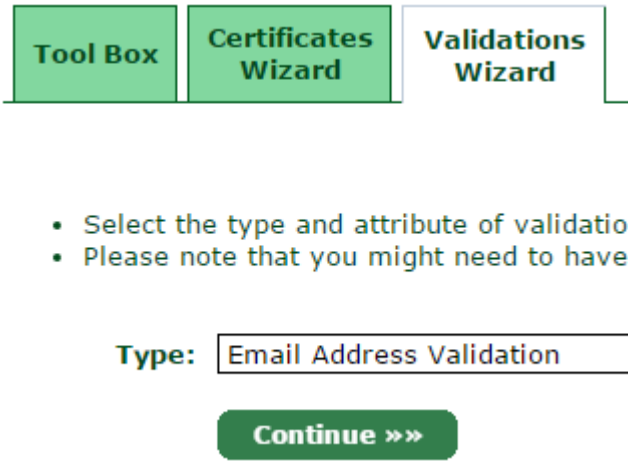


Klicken Sie rechts oben auf das Tür-Schlüsselsymbol. Klicken Sie anschließend auf „Express Lane“.

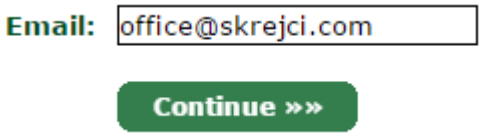


Schritt 1: EMail-Adresse validieren

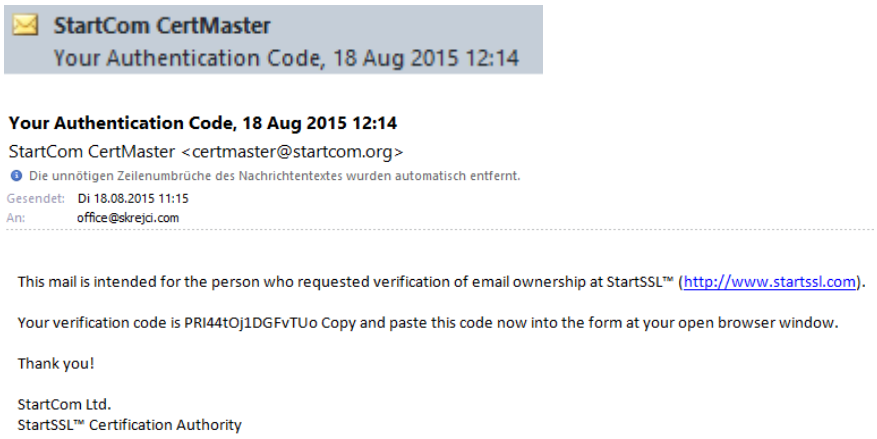
Nach der Registrierung müssen Sie über den „Validations Wizard“ Ihre Email-Adresse verfizieren. Klicken Sie dazu auf „Validations Wizard“ und wählen unter „Type“ die „Email Address Validation“ aus.



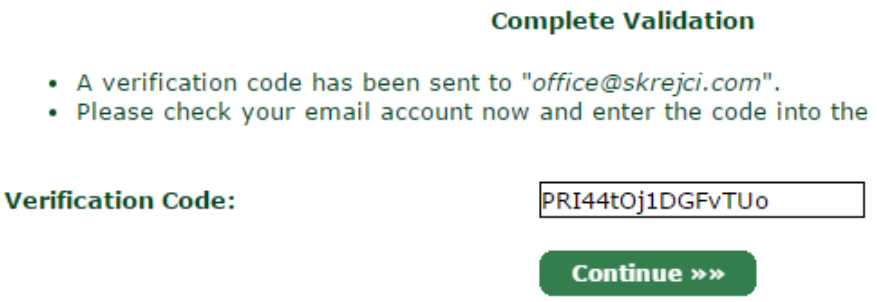
Geben Sie Ihre Email-Adresse ein und klicken Sie auf Continue. Bitte bedenken Sie, dass diese Email-Adresse später im Zertifikat für Ihren Webserver aufscheinen wird – verwenden Sie besser eine offizielle (Schul-)Email-Adresse als eine private.



Sie bekommen im Anschluss ein Email mit einem Verification Code zugeschickt.



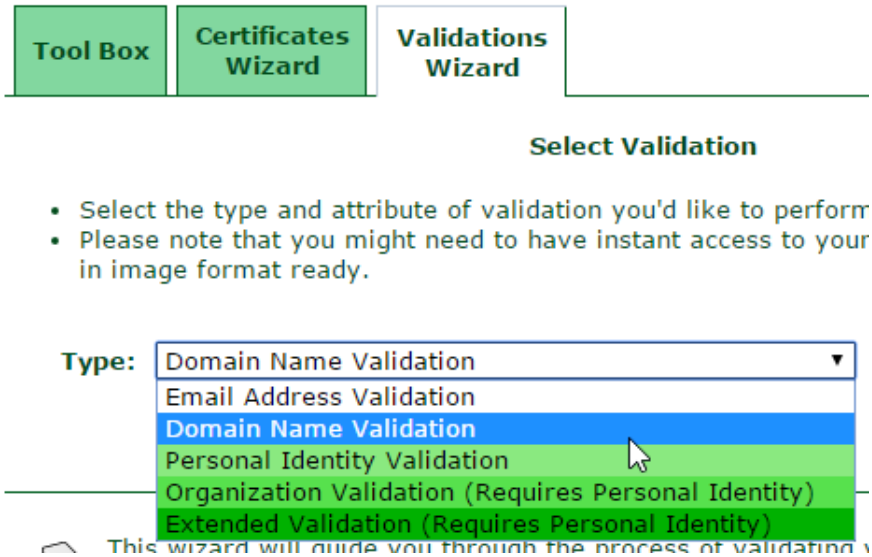
Kopieren Sie den Code und fügen Sie ihn im entsprechenden Feld



Klicken Sie auf Continue und anschließend auf Finish.

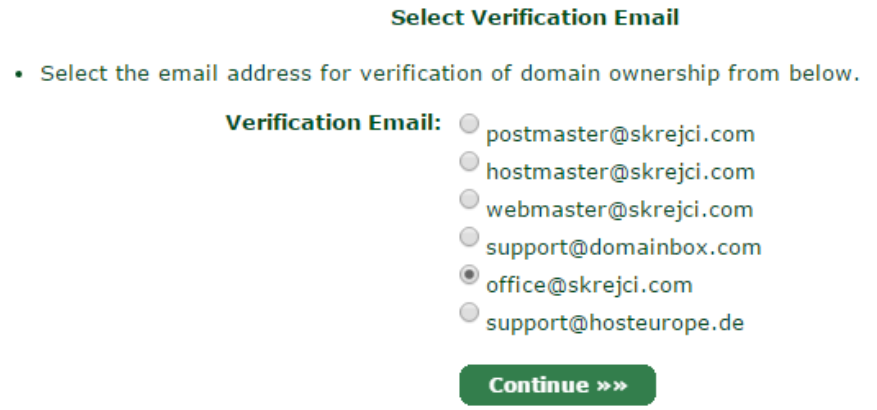
Schritt 2: Domäne validieren

Für die Validierung Ihrer Domäne gehen Sie analog wie bei der Validierung für die Email-Adresse vor. Wählen Sie im „Validations Wizard“ nun „Domain Name Validation“ aus.

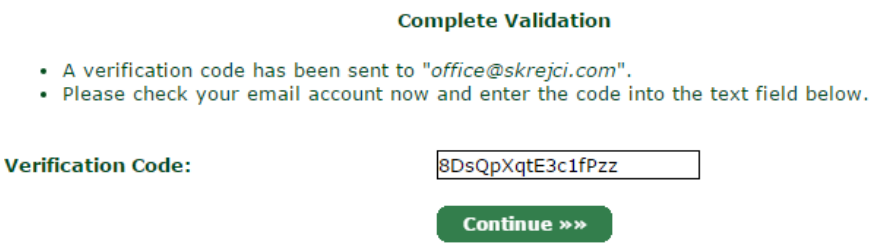


Geben Sie nun Ihre Domäne ein und klicken Sie auf Continue.

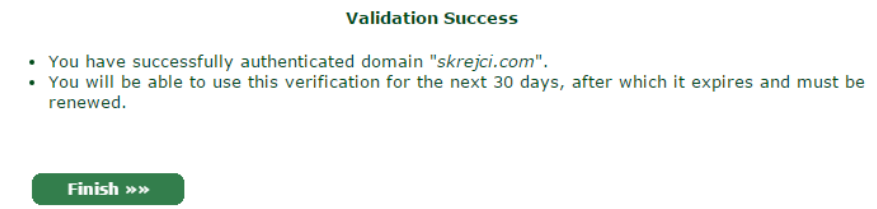
Sie erhalten eine Liste aller beim Registrar eingetragenen Email-Kontaktadressen. Wählen Sie diejenige aus, auf die Sie Zugriff haben und klicken Sie auf „Continue“.



Sie erhalten wiederum ein Email mit einem Verification Code. Kopieren Sie diesen Code in die Zwischenablage und fügen Sie ihn auf der Homepage ein.

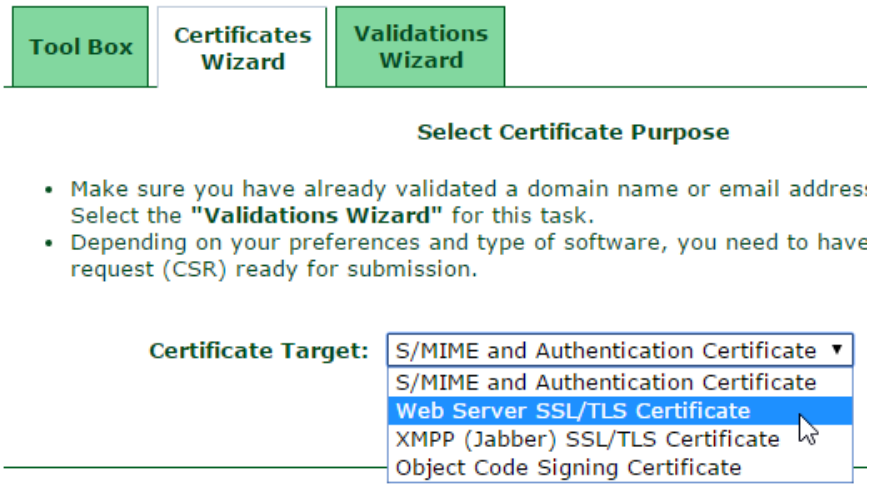


Diese Domänen-Validierung ist nun 30 Tage lang gültig. In diesem Zeitraum müssen Sie das Zertifikat ausstellen, ansonsten müssen Sie die Domäne erneut validieren.

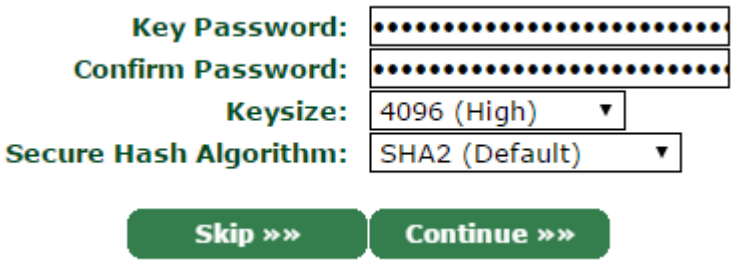


Schritt 3: Zertifikat anfordern

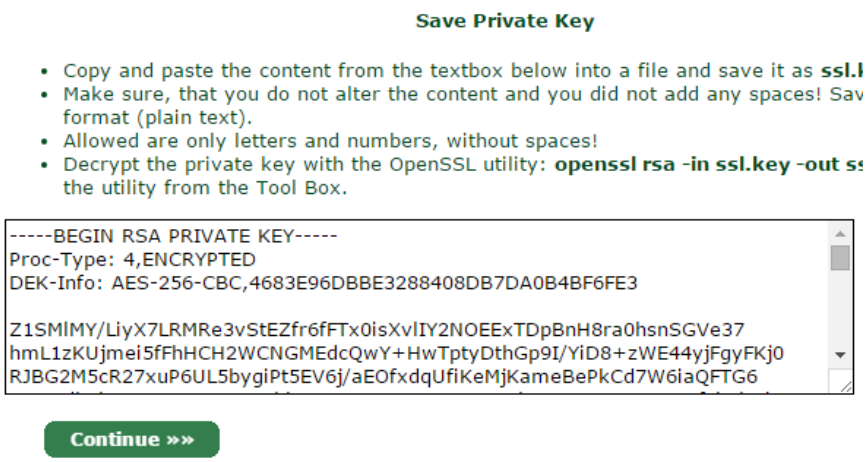
Um ein Zertifikat für Ihre Domäne zu erstellen, klicken Sie nun auf „Certificates Wizard“ und wählen aus der Liste „Web Server SSL/TLS Certificate“ aus.

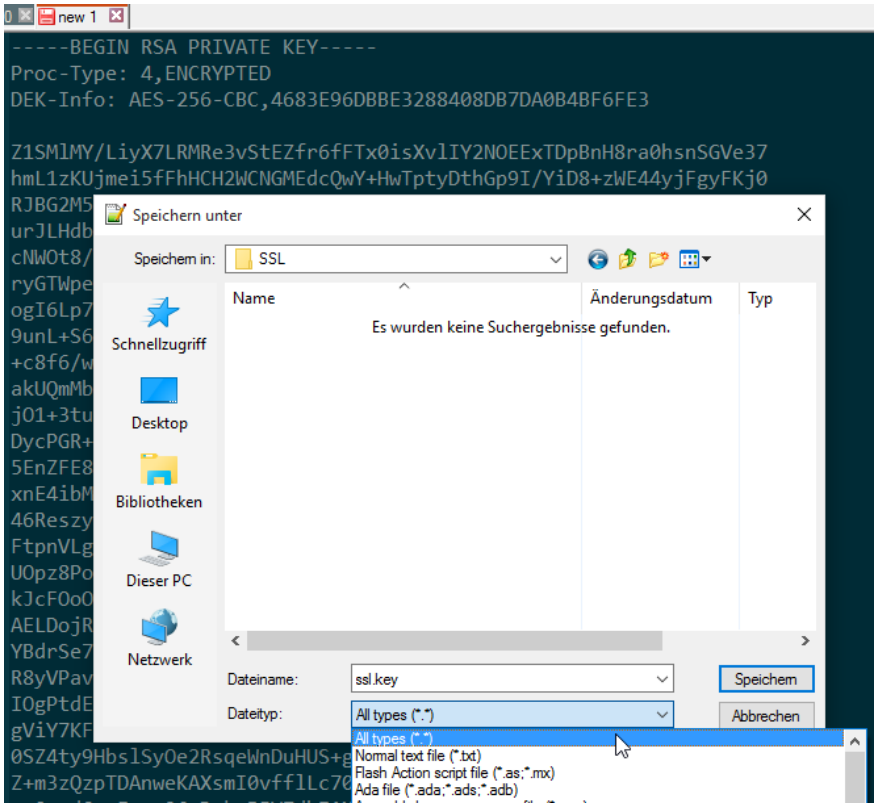


Im nächsten Schritt wird Ihr privater Schlüssel erzeugt. Sie benötigen ein Passwort (Länge zwischen 10 und 32 alphanumerische Zeichen). Notieren Sie sich dieses Passwort gut! Belassen Sie aus Sicherheitsgründen die Keysize unbedingt bei 4096 Byte und den Hash-Algorithmus bei sha2!



Der private Key wird nun generiert und in einer Textarea dargestellt. Kopieren Sie den gesamten Inhalt in einen Texteditor Ihrer Wahl und speichern Sie die Datei unter dem Namen ssl.key ab!

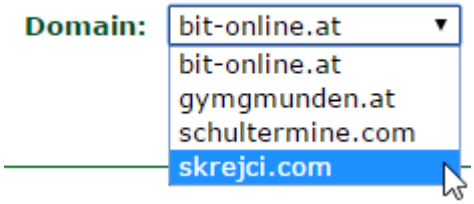




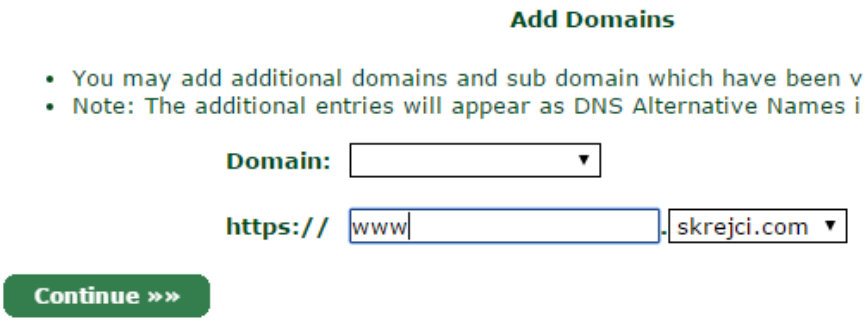
Je nach verwendetem Editor müssen Sie den Dateityp auf „All types (*.*)“ umstellen, damit die Datei nicht fälschlicherweise den Dateinamen „ssl.key.txt“ erhält.

Klicken Sie nun auf „Continue“.

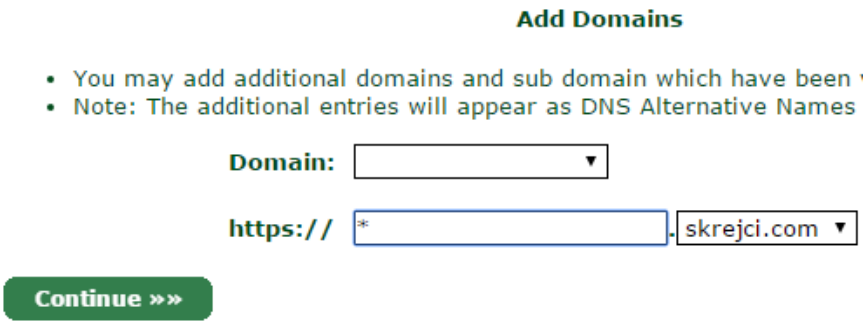
Nach der Erstellung des Private Keys wird jetzt das Zertifikat für Ihre Domäne beantragt. Wählen Sie aus der Drop Down Liste Ihre zuvor validierte Domäne aus.



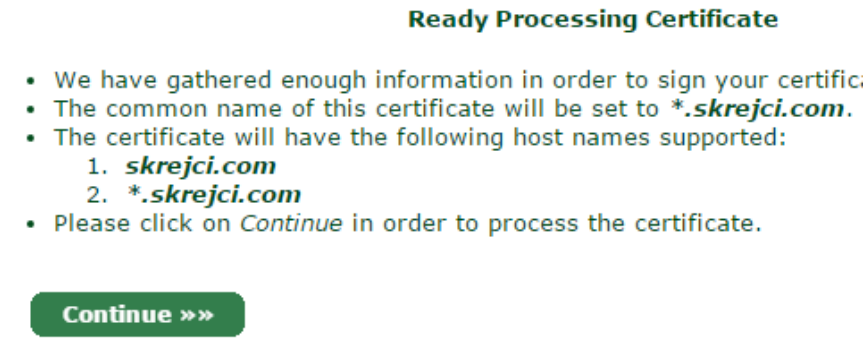
Das nächste Dialogfeld sieht je nach bei StartSSL gewähltem Paket bzw. je nach Anzahl der verfügbaren Domänen verschieden aus. Je nachdem wie Ihre Domäne lautet, müssen Sie hier die Subdomain eintragen – im Allgemeinen ist dies „www“. Damit wird das Zertifikat automatisch auf „skrejci.com“ und „www.skrejci.com“ ausgestellt.



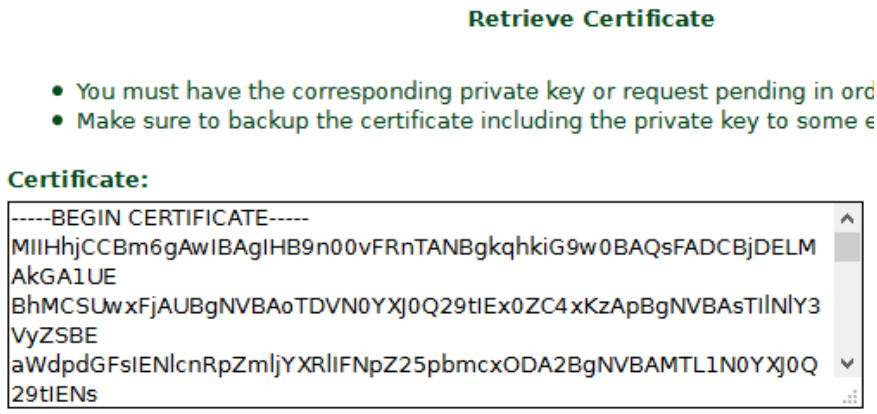
Sind Sie berechtigt, ein Wildcard-Zertifikat auszustellen (ein Zertifikat, dass für alle Subomains Ihrer Domäne gilt), so tragen Sie statt www ein Sternchen * ein.



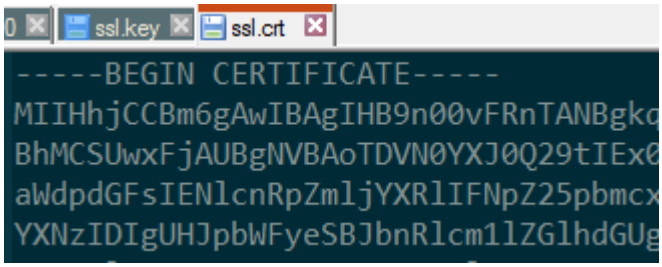
Klicken Sie anschließend auf „Continue“. Es wird eine Zusammenfassung angezeigt.



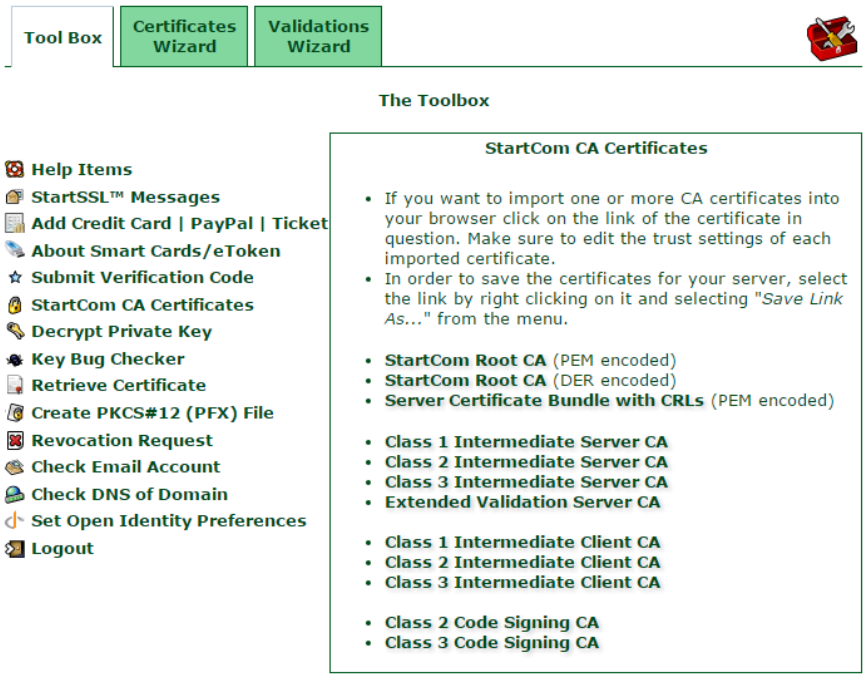
Mit einem weiteren Klick auf „Continue“ wird das Zertifikat erzeugt.



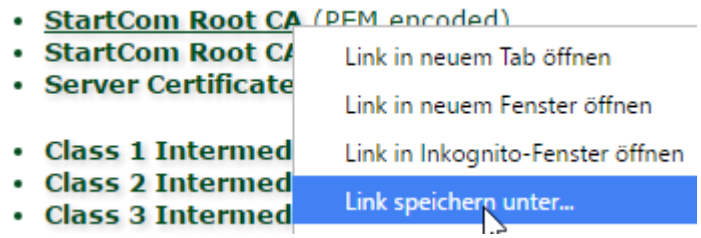
Speichern Sie das Zertifikat wie vorher in einer Textdatei, diesmal unter dem Namen „ssl.crt“.



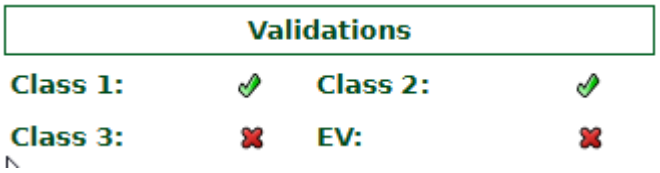
Nun benötigen Sie noch die Root-Zertifikate. Wählen Sie unter „Toolbox“ den Menüeintrag „StartCom CA Certificates“ aus.



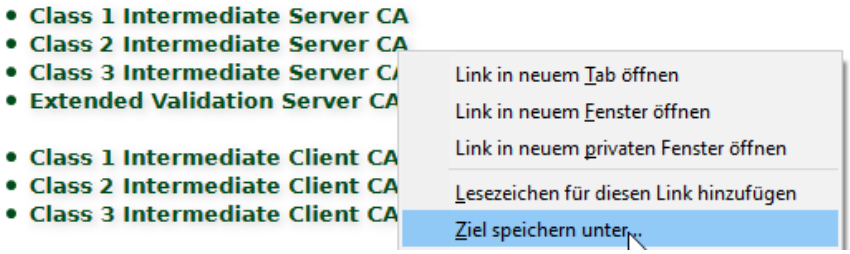
Laden Sie als erstes das „StartCom Root CA (PEM encoded)“ herunter. Dieses erhält den Namen „ca.pem“.



Anschließend wählen Sie das Server Zertifikat der entsprechenden Klasse aus. Diese finden Sie rechts unter „Validations“ je nach Anzahl der grünen Häckchen.

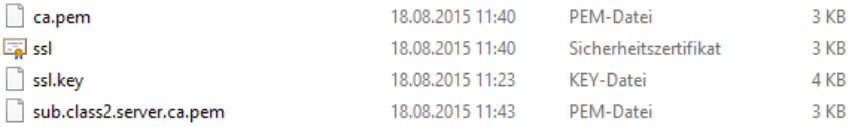


In diesem Fall (grünes Häckchen bei Class 1 und Class 2) wird das Serverzertifikat „Class 2 Intermediate Server CA“ benötigt.



Sind Sie „nur“ Class 1 validiert, wählen Sie bitte das „Class 1 Intermediate Server CA“. Speichern Sie dieses Zertifikat unter dem vorgeschlagenen Dateinamen, z.B. „sub.class2.server.ca.pem“).

Sie haben nun insgesamt vier Dateien erzeugt bzw. heruntergeladen.



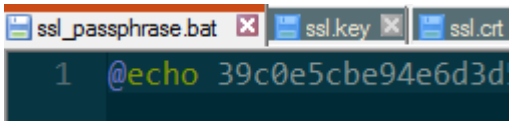
Schritt 4: Installation SSL am Webserver

Diese vier Dateien kopieren Sie nun auf Ihren Webserver in das Verzeichnis „c:\Webserver\Configuration“.

Erstellen Sie nun in diesem Verzeichnis eine Datei namens „ssl_passphrase.bat“ als Textdatei und öffnen Sie diese zur Bearbeitung. Schreiben Sie in diese Datei folgende Zeile:

@echo XXXX

Ersetzen Sie das XXXX dabei durch das Passwort des privaten Schlüssels! Bei Aufruf dieser Batch-Datei wird mittels dem echo-Befehl das Passwort auf der Kommandozeile ausgegeben, so bekommt der Webserver Zugriff auf den privaten Schlüssel. Speichern Sie die Datei und schließen Sie sie!



Öffnen Sie nun am Webserver die Apache-Konfigurationsdatei „httpd.conf“ im Verzeichnis „c:\Webserver\Apache24\conf“.

Navigieren Sie zu der Stelle in der Datei, an der die zusätzlichen Module geladen werden. Binden Sie die beiden Module „mod_ssl“ und „mod_socache_shmcb“ ein.

```
165 LoadModule socache_shmcb_module modules/mod_socache_shmcb.so
166 LoadModule ssl_module modules/mod_ssl.so
```

Navigieren Sie nun nach(fast ganz) unten zur Zeile „Include conf/extra/httpd-ssl.conf“. Entfernen Sie das Raute-Symbol davor, sodass die Zeile interpretiert und die httpd-ssl.conf – Datei mit in die Konfiguration aufgenommen wird.

```
Include conf/extra/httpd-ssl.conf
```

Stellen Sie auch sicher, dass die folgenden Zeilen etwas unterhalb nicht auskommentiert sind.

```
<IfModule ssl_module>
  SSLRandomSeed startup builtin
  SSLRandomSeed connect builtin
</IfModule>
```

Speichern und schließen Sie diese Datei, öffnen Sie statt dessen die Datei „httpd-ssl.con“ im Verzeichnis „c:\Webserver\Apache24\conf\extra“.

In dieser Datei werden – ähnlich wie in der httpd-vhosts.conf – Datei – die Virtual Hosts für die SSL-Verbindungen angebeben. Zuerst folgt ein Block mit den allgemeinen SSL-Einstellungen, hier im folgenden wieder eine Beispielkonfiguration ohne Gewähr:

```
Listen *:443
SSLCipherSuite ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-
AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-DSS-
AES128-GCM-SHA256:kEDH+AESGCM:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-SHA256:ECDHE-
RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES256-
SHA384:ECDHE-RSA-AES256-SHA:ECDHE-ECDSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-
AES128-SHA:DHE-DSS-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-DSS-AES256-SHA:DHE-RSA-
AES256-SHA:AES128-GCM-SHA256:AES256-GCM-
SHA384:AES128:AES256:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!3DES:!MD5:!PSK:!ECDHE-RSA-RC4-
SHA:!ECDHE-ECDSA-RC4-SHA:!RC4-SHA
SSLHonorCipherOrder on
SSLProtocol all -SSLv3
SSLProxyProtocol all -SSLv3
SSLPassPhraseDialog "exec:C:/WebServer/Configuration/ssl_passphrase.bat"
SSLSessionCache        "shmcb:c:/WebServer/Logs/ssl_scache(512000)"
SSLSessionCacheTimeout  300
```

Hier kommt die zuvor erstellte „ssl_passphrase.bat“ zum Einsatz.

Anschließend an diese allgemeinen Konfigurationen folgt die Definition der einzelnen Hosts. Auch hier geben wir Ihnen wieder eine mögliche Beispielkonfiguration zur Vorlage wider:

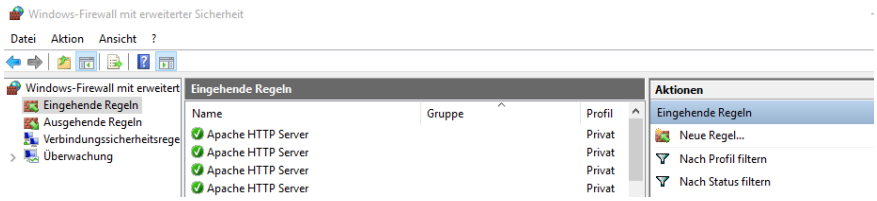
```
<VirtualHost *:443>
  DocumentRoot "c:/WebServer/wwwRoot/www.skrejci.com.ssl"
  ServerName www.skrejci.com:443
  ServerAdmin postmaster@skrejci.com
  ErrorLog "c:/WebServer/Logs/www.skrejci.com.ssl.error.log"
  TransferLog "c:/WebServer/Logs/www.skrejci.com.ssl.access.log"
  SSLEngine on
  SSLCertificateFile "c:/WebServer/Configuration/ssl.crt"
  SSLCertificateKeyFile "c:/WebServer/Configuration/ssl.key"
  SSLCertificateChainFile "c:/WebServer/Configuration/sub.class2.server.ca.pem"
  SSLCACertificateFile "c:/WebServer/Configuration/ca.pem"
  <Directory "c:/WebServer/wwwRoot/www.skrejci.com.ssl ">
    Options None
    AllowOverride All
    Require all granted
    DirectoryIndex index.php index.html
  </Directory>
  <FilesMatch "\.(cgi|shtml|phtml|php)$">
    SSLOptions +StdEnvVars
  </FilesMatch>
  <Directory "c:/WebServer/Apache24/cgi-bin">
    SSLOptions +StdEnvVars
  </Directory>
  BrowserMatch "MSIE [2-5]" \
         nokeepalive ssl-unclean-shutdown \
         downgrade-1.0 force-response-1.0
  CustomLog "c:/WebServer/Logs/www.skrejci.com.ssl_request.log" \
          "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
</VirtualHost>
```

Wie bei den „normalen“ Virtual Hosts werden die SSL-Pendants über die Direktive „ServerName“ identifiziert. Die Zertifikate bzw. der private Key werden über die entsprechenden SSLCertificate- bzw. SSLCACertificate-Direktiven eingebunden. Anschließend werden die Optionen für den DocumentRoot genauso wie in der httpd-vhosts.conf festgelegt.

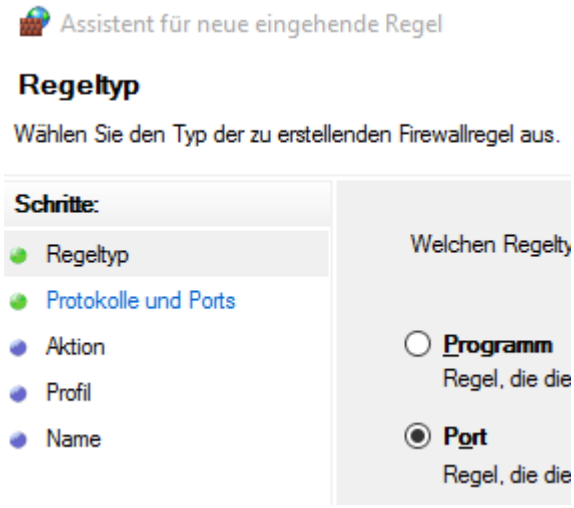
Schritt 5: Firewall für SSL einrichten

Damit eingehende SSL-Anforderungen nicht geblockt werden, muss standardmäßig Port 443 auf der lokalen Firewall (und selbstverständlich auch auf allfälligen vorgeschalteten Firewalls durchgeroutet und) freigeschaltet werden.

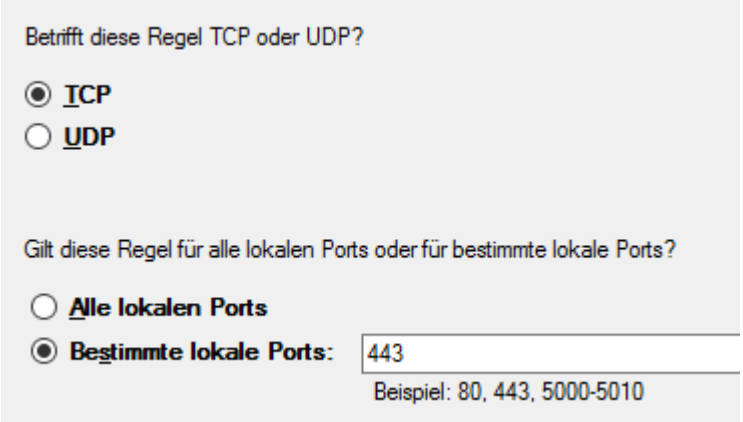
Öffnen Sie dazu die „Windows-Firewall mit erweiterter Sicherheit“ auf dem Server. Navigieren Sie zu „eingehende Regeln“ und klicken Sie rechts auf „Neue Regel…“.



Aktivieren Sie im ersten Schritt „Port“,



im nächsten Dialogfenster geben Sie bei „bestimmte lokale Ports“ die Zahl 443 an.



Im nächsten Dialogfeld belassen Sie „Verbindung zulassen“, im wiederum nächsten Dialogfeld belassen Sie alle drei Profiltypen „Domäne“, „Privat“ und „Öffentlich“ aktiviert. Im letzten Schritt vergeben Sie eine eindeutige Bezeichnung, und klicken Sie auf Fertig stellen.

Ab sofort lässt Ihr Server Anfragen auf dem Standard-SSL Port 443 zu.

Fertig!

Testen Sie den Zugriff nun, indem Sie Ihren Server mit vorangestelltem „https“ aufrufen!

Wichtiger Hinweis: In näherer Vergangenheit sind immer wieder teil gravierende Sicherheitsprobleme im Umgang mit SSL aufgetreten. Prüfen Sie daher in halbwegs regelmäßigen Abständen Ihren SSL-Server auf Sicherheit! Navigieren Sie z.B. zur Seite www.ssllabs.com/ssltest. Geben Sie hier Ihren Domänennamen an und klicken Sie auf „Submit“. Nach einiger Zeit erhalten Sie wichtige Informationen bzgl. der Sicherheit Ihrer Konfiguration!

Hier das Ergebnis des Tests der Konfiguration, die hier im Tutorial beschrieben ist (Stand 18.08.2015):

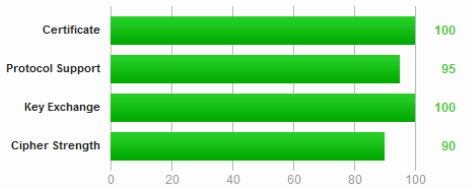
Assessed on: Tue, 18 Aug 2015 10:14:42 UTC | Clear cache

Scan Anot

Summary

Overall Rating

A



Visit our documentation page for more information, configuration guides, and books. Known issues are documented here.

This server supports TLS_FALLBACK_SCSV to prevent protocol downgrade attacks.