HTTPS-Zertifikate in Velocity

Wie Velocity HTTPS für eine gehostete Website ausliefert - Zertifikate aus einer Datei oder einem Panel, Let's Encrypt und andere ACME-CAs, Archive und PKCS#12-Bündel, der selbstsignierte Ersatz, Austausch im Betrieb…

Source file: `doc/ger-de/hilfe/hosting/server/https-zertifikate-in-velocity.md` Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/hosting/servers/velocity-https-certificates.md

Diese Seite ist für Systemadministratoren bei Hosting-Unternehmen, die Velocity HTTPS für Kunden-Websites ausliefern lassen. Sie beantwortet: welche Quellen für Zertifikate Velocity akzeptiert, wie Sie jede konfigurieren, was bei Erneuerung und Fehlschlag passiert, wie Sie den Ablauf über viele Websites beobachten und was gesichert werden muss. Die Quellen sind Kapitel 8, Abschnitt 8.3.9 und HTTPS und Zertifikate in Velocity.

Inhalt

1. Worauf Sie sich verlassen können
2. Eine Quelle für Zertifikate wählen
3. Ein Zertifikat, das Sie schon haben
4. Let's Encrypt und andere ACME-CAs
5. Strict-Transport-Security
6. Den Ablauf beobachten
7. Was gesichert werden muss
8. Fehlerbehebung
9. Checkliste

Worauf Sie sich verlassen können

Aus der Seite zur Funktion:

- HTTPS startet vor einfachem HTTP, sodass es nie einen Moment gibt, in dem der Server HTTP beantwortet, aber nicht HTTPS.
- Ein kaputtes oder abgelaufenes Zertifikat erreicht nie einen Besucher: Das Paar wird vor der Nutzung geprüft (lässt sich lesen, nicht abgelaufen, der Schlüssel gehört dazu), und ein **selbstsigniertes Zertifikat** springt ein, bis das echte wieder nutzbar ist (der `fallback` der Engine, Standard `self-signed`; `none` schaltet HTTPS stattdessen aus).
- Der Server liest eine private Kopie des Paars, sodass eine Datei, die anderswo neu geschrieben wird, keinen Handshake stören kann. Alle `watchInterval` Sekunden (60) wird die Quelle geprüft, und ein geändertes, gültiges Paar wird im Betrieb ausgetauscht, ohne Neustart und ohne abgebrochene Verbindung.
- Das Zertifikat, das eine Domain abdeckt, wird pro Verbindung gewählt (SNI).
- HTTP/2 kommt mit.

Eine Quelle für Zertifikate wählen

Quelle (`mode`)	Nutzen Sie sie, wenn	Konfiguriert in
`manual` / `files`	das Panel, certbot oder eine CA Ihnen PEM-Dateien gibt	`velocity.ini [HTTPSSettings]`
`letsencrypt` (auch `acme`)	Velocity das Zertifikat selbst beziehen und erneuern soll	einem Snippet in `/etc/vc/conf-available/`
`certbot`	certbot das Zertifikat bereits verwaltet	einem Snippet
`archive`	die CA ein `.zip`, `.tar.gz`, `.7z` oder Ähnliches liefert	einem Snippet
`pkcs12`	Sie ein Bündel `.p12` / `.pfx` haben	einem Snippet
`remote`	das Zertifikat von einer URL heruntergeladen wird	einem Snippet
`self-signed`	nur Entwicklung	einem Snippet

`velocity.ini` bietet nur das Dateipaar an; die anderen Modi werden unter `Q.web.https` in der Konfiguration der Engine gesetzt, wofür eine Datei in `conf-available/` der richtige Ort ist (sie gehört dem Administrator und wird vor der erzeugten Site-Datei eingemischt). Lassen Sie in diesem Fall `[HTTPSSettings] Enabled` auf `false`.

Ein Zertifikat, das Sie schon haben

Jedes PEM-Zertifikat von einer CA oder einem Hosting-Panel (Leaf mit Kette und der private Schlüssel):

Shell

```
php bin/php/console exp:velocity config set --apply HTTPSSettings Certificate /etc/ssl/example.co
m/fullchain.pem --allow-root-user
php bin/php/console exp:velocity config set --apply HTTPSSettings Key /etc/ssl/example.com/privke
y.pem --allow-root-user
php bin/php/console exp:velocity config set --apply HTTPSSettings Enabled true --allow-root-user
php bin/php/console exp:velocity restart --allow-root-user
php bin/php/console exp:velocity ssl show --allow-root-user
curl -sI https://example.com/ | head -1
```

- HTTPS wird nur eingeschaltet, wenn `Enabled` `true` ist **und** beide Dateien existieren; ein fehlender Pfad lässt den Server auf einfachem HTTP, und `status` sagt, was es einschaltet.
- Der Serverprozess liest den Schlüssel als root, wenn er als root gestartet wird, sodass der Schlüssel `0600` `root` bleiben kann.
- Eine Erneuerung durch Ihr eigenes Werkzeug (oder das Panel) wird innerhalb einer Minute übernommen, ohne Neustart.
- Auf den ausgelieferten Ports ist HTTPS `8080` und HTTP `8088`, beide lauschen; `velocity.ini` hat keine Einstellung, die einen auf den anderen umleitet. Stellen Sie sicher, dass die Adressen, die die Website erzeugt, HTTPS sind (`site.ini [SiteSettings] SiteURL` und die URLs der Siteaccesses).

Let's Encrypt und andere ACME-CAs

Voraussetzungen für die Challenge HTTP-01: Das DNS jedes Namens zeigt auf diese Maschine, und Port 80 dieses Servers ist aus dem Internet erreichbar (`[ServerSettings] Port 80`). Aus Kapitel 8:

Shell

```
cat > /etc/vc/conf-available/https-acme.conf <<'EOF'
{ "Q": { "web": { "https": {
  "port": 443,
  "mode": "letsencrypt",
  "acme": {
    "email": "hostmaster@example.com",
    "domains": ["example.com", "www.example.com"],
    "directory": "letsencrypt-staging"
  }
} } } }
EOF
php bin/php/console exp:velocity conf enable --apply https-acme --allow-root-user
php bin/php/console exp:velocity restart --allow-root-user
php bin/php/console exp:velocity ssl show --allow-root-user
```

- Beginnen Sie mit `letsencrypt-staging`: Seinen Zertifikaten vertrauen Browser nicht, aber seine Ratenbegrenzungen sind großzügig. Wenn `ssl show` ein Staging-Zertifikat aufführt, entfernen Sie diese Zeile (der Standard ist `letsencrypt`) und starten Sie neu.
- Bis das erste Zertifikat ankommt, wird das selbstsignierte ausgeliefert; das CA-Zertifikat ersetzt es im Betrieb.
- Die Erneuerung läuft im Hintergrund, wenn ein Drittel der Laufzeit übrig ist (`renewAt` 0.33), mit wachsenden Pausen nach einem Fehlschlag (5 Minuten, verdoppelt bis zu einem Tag), sodass Ratenbegrenzungen nie aufgebraucht werden. `exp:velocity ssl renew [host...]` fordert sie sofort an.
- `directory` nimmt auch `zerossl`, `google`, `google-staging` oder jede URL eines ACME-Verzeichnisses; `eab` enthält die External Account Binding. `keyType` ist standardmäßig `ec256`.
- Wildcards brauchen `challenge: dns-01` mit einem Programm `dnsHook`. Wenn ein anderer Server Port 80 besitzt, setzen Sie `webroot` auf ein Verzeichnis, das dieser Server ausliefert.
- `start` meldet, dass die Konfiguration wegen des eingeschalteten Snippets von `velocity.ini` abweicht; das ist zu erwarten.

> **Tipp** Nutzen Sie auf einem Host mit vielen Websites ein Snippet pro Installation und nur die Domains des Kunden. Ein geteiltes Snippet, das die Namen eines anderen Kunden aufführt, würde für sie Zertifikate von der falschen Maschine anfordern.

Strict-Transport-Security

INI

```
[HTTPSSettings]
HSTSMaxAge=300              # seconds; 0 sends no header
HSTSIncludeSubDomains=disabled
HSTSPreload=disabled
```

Der Header geht bei jeder HTTPS-Antwort hinaus und nie über einfaches HTTP. Ein Browser, der ihn gesehen hat, verweigert für den ganzen Zeitraum einfaches HTTP zu diesem Host, selbst wenn Sie den Wert später senken. Erhöhen Sie auf ein Jahr (`31536000`) erst, wenn jeder Name der Website dauerhaft über HTTPS ausgeliefert wird; Preload braucht `HSTSIncludeSubDomains` und mindestens ein Jahr. Vereinbaren Sie das mit dem Kunden: Es überdauert einen Umzug zu einem anderen Hoster.

Den Ablauf beobachten

Shell

```
php bin/php/console exp:velocity ssl show --allow-root-user          # served certificate, every
known one by expiry
php bin/php/console exp:velocity ssl show --json --allow-root-user   # for monitoring: watch days
Left
```

Ein CA-Zertifikat mit 90 Tagen wird erneuert, wenn etwa 30 Tage übrig sind; eines mit weniger als 14 übrigen Tagen bedeutet also, dass die Erneuerung seit mehr als zwei Wochen scheitert; der letzte Fehler, den `ssl show` mit dem ACME-Zustand ausgibt, sagt warum. Halten Sie NTP am Laufen: Zertifikate werden gegen die Systemuhr geprüft. Der Reiter **SSL** des Kontrollzentrums zeigt dasselbe und kann aus einem Browser erneuern und neu laden. Lösen Sie für jede Kunden-Website einen Alarm bei `daysLeft` unter 14 aus; siehe Ablauf von Zertifikaten.

Was gesichert werden muss

Sichern Sie das Verzeichnis `ssl/` des Konfigurationsverzeichnisses, mit dem der Server läuft. Alles darin lässt sich neu erstellen außer den **Schlüsseln der ACME-Konten** (`acme/account-<ca>.pem`). Der Aufbau:

```
ssl/
  self-signed.pem, self-signed.key      the self-signed pair
  active-443-<fp>-<pid>.pem/.key        the copy one server reads (do not edit)
  imported/                             pairs imported from DER, archives, bundles, remote
  acme/account-<ca>.pem                 one account key per CA: keep it
  acme/<first domain>/fullchain.pem, privkey.pem, state.json
```

Private Schlüssel werden mit `0600` geschrieben, atomar, und nie protokolliert.

Fehlerbehebung

Symptom	Ursache	Abhilfe
`status` meldet HTTPS als aus	Enabled ist nicht `true`, oder eine genannte Datei existiert nicht	die Pfade mit `ls -l` prüfen
der Browser warnt vor dem Zertifikat	der selbstsignierte Ersatz wird ausgeliefert	`ssl show`; das Startlog nennt den Grund
ACME `Invalid response from http://.../.well-known/acme-challenge/...`	Port 80 erreicht diesen Server nicht	Port 80 öffnen oder `acme.webroot` für den Server setzen, dem er gehört

Symptom	Ursache	Abhilfe
`none of the certificates belongs to the private key`	Schlüssel und Zertifikat aus verschiedenen Anfragen	den Schlüssel nutzen, für den das Zertifikat ausgestellt wurde
`no private key found (is it encrypted? ...)`	verschlüsselter Schlüssel, kein Passwort angegeben	`keyPassword`, `keyPasswordFile` oder `keyPasswordEnv` setzen

Checkliste

- ☐ Jede Website liefert ein CA-Zertifikat aus (`ssl show`), nicht den selbstsignierten Ersatz.
- ☐ ACME zuerst gegen Staging versucht; danach das Produktionsverzeichnis gesetzt.
- ☐ Die Überwachung schlägt bei `daysLeft` unter 14 Alarm.
- ☐ NTP läuft auf der Maschine.
- ☐ `ssl/` gesichert, einschließlich der Schlüssel der ACME-Konten.
- ☐ `HSTSMaxAge` nur mit Zustimmung des Kunden erhöht und mit jedem Namen auf HTTPS.

Verwandte Seiten

- Velocity für Hoster
- TLS und HTTPS
- TLS-Zertifikate in Panels
- Ablauf von Zertifikaten
- HTTPS und Zertifikate in Velocity
- Freie Dokumentation: Diese Seite darf unter der GNU FDL 1.3 frei kopiert, übersetzt und verbessert werden.

> **Werden Sie Exponential-Partner**
>
> Partner teilen ihre Informationen, stellen Mittel bereit, geben Beiträge zurück und entwickeln die Produkte mit uns, und so wächst Exponential. Eine Partner-Website und ein Dashboard sind für Mitte 2027 geplant: Werden Sie noch heute Partner, damit wir schneller wachsen und neue Funktionen früher liefern.
>
> Partner: hier beginnen | Wie Sie sich bewerben

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.