
SSL certificate chain resolver

build failing licence MIT

This application downloads all intermediate CA certificates for a given SSL server certificate. It can help you fix the *incomplete certificate chain* issue, also reported as *Extra download* by Qualys SSL Server Test.

See Releases for prebuilt binaries or build it yourself.

NOTE: In case of any troubles with Go you can try the deprecated shell script in shell branch.

Usage

```
1 NAME:
2   cert-chain-resolver - SSL certificate chain resolver
3
4 USAGE:
5   cert-chain-resolver [global options] [INPUT_FILE]
6
7 VERSION:
8   1.0.4
9
10 GLOBAL OPTIONS:
11   --output OUTPUT_FILE, -o OUTPUT_FILE  output to OUTPUT_FILE (default
12   : stdout)
13   --intermediate-only, -i                output intermediate
14   certificates only
15   --der, -d                             output DER format
16   --include-system, -s                   include root CA from system in
17   output
18   --version, -v                         print the version
```

Example

```
1 $ cert-chain-resolver -o domain.bundle.pem domain.pem
2 1: *.xxx.com
3 2: COMODO RSA Domain Validation Secure Server CA
4 3: COMODO RSA Certification Authority
5 Certificate chain complete.
6 Total 3 certificate(s) found.
```

Build

Dependencies:


- Go >= 1.12

```
1 go mod download
2 go build
```

Tests


```
1 go test ./...
2 tests/run.sh
```

Background



Additional Certificates (if supplied)

Certificates provided	1 (1359 bytes)
Chain issues	Incomplete



Certification Paths

Path #1: Trusted

1	Sent by server	Fingerprint: [redacted] RSA 2048 bits (e 65537) / SHA256withRSA
2	Extra download	COMODO RSA Domain Validation Secure Server CA Fingerprint: 339cdd57cfd5b141169b615ff31428782d1da639 RSA 2048 bits (e 65537) / SHA384withRSA
3	Extra download	COMODO RSA Certification Authority Fingerprint: f5ad0bcc1ad56cd150725b1c866c30ad92ef21b0 RSA 4096 bits (e 65537) / SHA384withRSA
4	In trust store	AddTrust External CA Root Self-signed Fingerprint: 02faf3e291435468607857694df5e45b68851868 RSA 2048 bits (e 65537) / SHA1withRSA Weak or insecure signature, but no impact on root certificate

All operating systems contain a set of default trusted root certificates. But CAs usually don't use their root certificate to sign customer certificates. Instead of they use so called intermediate certificates, because they can be rotated more frequently.

A certificate can contain a special *Authority Information Access* extension (RFC-3280) with URL to issuer's certificate. Most browsers can use the AIA extension to download missing intermediate certificate to complete the certificate chain. This is the exact meaning of the *Extra download* message. But some clients (mobile browsers, OpenSSL) don't support this extension, so they report such certificate as untrusted.

A server should always send a complete chain, which means concatenated all certificates from the certificate to the trusted root certificate (exclusive, in this order), to prevent such issues. Note, the trusted root certificate should not be there, as it is already included in the system's root certificate store.

You should be able to fetch intermediate certificates from the issuer and concat them together by yourself, this script helps you automatize it by looping over certificate's AIA extension field.

Other implementations

- deprecated shell script (shell)
- freekmurze/ssl-certificate-chain-resolver (PHP)

Licence

The MIT License (MIT). See LICENCE file for more information. TL;DR

If you use my code in some interesting project, I'd be happy to know about it.