

TCS_ServerCert

Usage

With this script, you can generate a certificate request that you can submit manually to Terena TCS service. It's possible to include multiple SubjectAltName -s in the request, such as `aai.niif.hu` and `www.aai.niif.hu`.

This script creates the following files in your current working directory:

- `hostname.you.provided.first.org.key` (private key)
- `hostname.you.provided.first.org.csr` (certificate request)

Program code

```
#!/usr/bin/perl -w

print "Please enter the fqdn's of the hosts one at a line\n";
print "Press Ctrl-D when done or Ctrl-C to cancel\n";

my $h;
my @hosts;

while ($h=<STDIN>) {
    chomp ($h);
    #XXX sanity check
    push @hosts,$h;
}

my $tmpfile=`mktemp`;
chomp $tmpfile;

my $defaulthost=$hosts[0];
```

```
my @opensslReqCmd=("openssl","req","-new","-nodes","-config","$tmpfile","-out","$defaulthost.csr");
```

```
#for re-key, you'd use:
```

```
#if (-r "$defaulthost.key") {  
    #push @opensslReqCmd,("-key","$defaulthost.key");  
#}
```

```
my @opensslVerifyCmd=("openssl","req","-text","-in","$defaulthost.csr");
```

```
&mkConfig($tmpfile,@hosts);
```

```
umask 0077;
```

```
system @opensslReqCmd;
```

```
system @opensslVerifyCmd;
```

```
unlink $tmpfile;
```

```
sub mkConfig(@) {  
    my $out=shift;  
    my @hosts=@_  
    my $defaulthost=$hosts[0];  
  
    open (CONF,">$out") or die "$!";
```

```
    print CONF <<EOS;
```

```
[ req ]
```

```
default_bits          = 2048
```

```
default_keyfile       = $defaulthost.key
```

```
default_days         = 1095 # 3x365 days
```

```
default_md            = sha256
```

```
distinguished_name    = req_distinguished_name
```

```
req_extensions        = v3_req
```

```
prompt               = no
```

```
[ req_distinguished_name ]
```

```
CN                   = $defaulthost
```

```
[ v3_req ]
```

```
subjectAltName       = \@alt_names
```

```
[alt_names]
```

```
EOS
```

```
for (my $i=1; $i<=$#hosts+1; $i++) {  
    print CONF "DNS." . $i . "          = " . $hosts[$i-1] . "\n";  
}  
close CONF;  
}
```

Apache config

This is how you can instruct Apache to use the new cert

```
SSLCertificateFile /path/to/your/pki/hostname.you.provided.first.crt  
SSLCertificateKeyFile /path/to/your/pki/hostname.you.provided.first.key  
SSLCertificateChainFile /path/to/your/pki/hostname.you.provided.first-chain.crt
```

Self-signed

It's not recommended to use CA-signed certificates with your IdPs or SPs. It has no benefits and has some drawbacks (ie. some older versions of mod_ssl refuse to work with expired SP certs).

Instead, you should generate a self-signed certificate with the following commands (please adjust the subject):

```
export host=your.host.name  
openssl req -new -newkey rsa:2048 -subj "/C=HU/O=NIIF/OU=AAI/CN=$host" -days 10000 -nodes \  
-keyout $host-fed.key -out $host-fed.csr  
openssl x509 -in $host-fed.csr -out $host-fed.crt -req -signkey $host-fed.key
```

Változat #3

cziernorbert hozta létre 9 április 2025 16:31:01

cziernorbert frissítette 10 április 2025 09:49:50