Aktuelles von IO::Socket::SSL

Steffen Ullrich, genua mbH
Deutscher Perl-Workshop 2013, Berlin

- #1 SSL library in Perl
- Interface analog zu IO::Socket
- setzt auf Net::SSLeay auf
- me: Maintainer seit 2008

Client Side SNI

- Server Name Indication
  - mehrere SSL Zertifikate hinter einer IP
  - in SSLext sendet Client den Namen
- transparent genutzt, wenn Name bekannt (PeerHost)
- kann explizit gesetzt werden
  - SSL_hostname => 'foo.bar'
  - disable: SSL_hostname => ''
- wegen OpenSSL Bug erst ab openssl 1.0

Server Side SNI

- ein Zertifikat: SSL_cert_file => path
- per Hostname: SSL_cert_file => %hash
  - 'host' => path
  - '' => default_path

- Next Protocol Negotiation
- typischerweise SPDY inerhalb SSL

```
my $server = IO::Socket::SSL->new(
   Listen => ...,
   SSL_npn_protocols => ['foo','bar'],
   ...,
);
my $client = $server->accept;
my $want = $client->next_proto_negotiated;
-------
my $sock = IO::Socket::SSL->new(
   PeerAddr => ...,
   SSL_npn_protocols => ['bar','foobar']
);
my $can = $sock->next_proto_negotiated;
```

The Most Dangerous Code in the World: Validating SSL Certificates in Non-Browser Software

Martin Georgiev
The University of Texas at Austin

Subodh Iyengar
Stanford University

Suman Jana
The University of Texas at Austin

Rishita Anubhai
Stanford University

Dan Boneh
Stanford University

Vitaly Shmatikov
The University of Texas at Austin

- Analyse APIs von SSL Libs und SSL/Web-APIs in Java, PHP, Python...
- Perl nicht dabei
- Resultat: alles unsicher
- Perl wäre etwas besser gewesen
  - hostname verification
  - LWP - Mozilla CAs, Hostname-Checking per Default
  - **auch kein gescheites CRL/OCSP handling**

- bisher: SSL_verify_mode SSL_VERIFY_NONE per Default

```
*******************************************************************
 Using the default of SSL_verify_mode of SSL_VERIFY_NONE for client
 is deprecated! Please set SSL_verify_mode to SSL_VERIFY_PEER
 together with SSL_ca_file|SSL_ca_path for verification.
 If you really don't want to verify the certificate and keep the
 connection open to Man-In-The-Middle attacks please set
 SSL_verify_mode explicitly to SSL_VERIFY_NONE in your application.
*******************************************************************
```

- demnächst SSL_verify_mode 1 per Default

- BEAST, SSLv2, MD5...
- SSL_version => 'SSLv23:!SSLv2'
  - festgezurrter Syntax brach diverse Software
- SSL_cipher_list => 'ALL:!LOW'
- BEAST:
  - SSL_honor_cipher_order => 1,
  - SSL_cipher_list => 'RC4-SHA:ALL:!ADH:!LOW',

Next?

- SSL_verify_mode default auf 1 setzen
- OCSP/CRL ?