



Falko Dressler Tobias Limmer Thomas Schneider

Netzwerksicherheit WS 2007/08

6. Übungsblatt

11.12.2007

Diese Übung soll Sie mit der Verwendung der Kommandozeilentools sowie der API von OpenSSL vertraut machen und dient als Vorbereitung für die darauffolgende Übung.

Für diese Übung müssen Sie zunächst ein von der NetSecCA (`cacert.pem`) signiertes persönliches GroupCA Zertifikat (`groupcert.pem`) beantragen. Hierzu wie in der Übung besprochen einen **eigenen** privaten Schlüssel für ihre GroupCA (`groupkey.pem`) generieren. Für diesen dann einen Certificate Signing Request (CSR) (`groupcsr.pem`) erstellen mit **C=DE, ST=Bavaria, L=Erlangen, O=FAU, OU=CS 7 / NetSec, CN=Group XX/ emailAddress=GROUPCONTACT@informatik.stud.uni-erlangen.de**. Diesen CSR dann per Mail senden an `thomas.schneider@informatik.stud.uni-erlangen.de` und 24 Stunden warten, bis ihr persönliches GroupCA Zertifikat (`groupcert.pem`) erstellt und per Mail zurückgeschickt wurde. Mit diesem GroupCA Zertifikat können Sie dann wie in der Übung besprochen einen eigenen Server Key (`serverkey.pem`) erstellen, den zugehörige CSRs erzeugen und diesen signieren, um ein gültiges Server Zertifikat (`servercert.pem`) zu erstellen.

Aufgabenstellung: Erweitern Sie den in der Übung erklärten HTTPS Server (`wserver`) und Client (`wclient`) von <http://www.rtfm.com/openssl-examples/>. Aktuelle Zertifikate und Dummy-Keys hierfür stehen auf der NetSec Seite zur Verfügung:

- “NetSecCA Zertifikat” (`cacert.pem`)
- Dummy Server Zertifikat (`servercert.pem`) & Key (`serverkey.pem`)
- Optional (`wserver2`, `wclient2`): Client Zertifikat (`clientcert.pem`) & Key (`clientkey.pem`)

Ihre Erweiterung soll folgende zusätzlichen Funktionalitäten bieten:

- Server+Client: Unterstützung hierarchischer Zertifikate. Zumindest `cacert.pem`, `groupcert.pem` und `server.pem`.
- Server: Abfrage des Passworts für den private Key beim Programmstart, falls Kommandozeilenoption “-p” angegeben wurde.
- Client: Anzeige aller Zertifikate des Zertifizierungspfades, falls Kommandozeilenoption “-v” angegeben wurde.
- Client: Falls das Root-Zertifikat nicht das “NetSecCA Zertifikat”, sondern ein Self Signed Root Zertifikat ist, wird dem Benutzer das Self Signed Root Zertifikat und eine Warnmeldung ausgegeben. Der Benutzer wird in diesem Fall gefragt, ob er die Verbindung trotzdem aufbauen oder abbrechen will.

Technische Hinweise:

- Die Aufgaben sind in C/C++ mit OpenSSL oder Java (no support !) zu erstellen.

- Der Code muss auf dem Betriebssystem Debian/Linux im CIP-Pool der Informatik mit der dort installierten Version des GNU C/C++- oder Java-Compilers kompilieren und lauffähig sein.
- Es ist ein Makefile oder ein ausführbares Skript zu bereitzustellen, welches temporäre Daten löschen (Target „clean“) und eine Binärdatei (Target „all“) erstellen kann.
- Es ist eine Dokumentation im ASCII-Format zu erstellen, welche einen typischen Programmstart dokumentiert und über Besonderheiten des Programms berichtet.

Fragen und Abgabe der Lösungen bis Montag, den 17.12.2007 als Archiv im Format `.tar.(gz|bz2)` oder `.zip` per Mail an `tobias.limmer@informatik.uni-erlangen.de` und `thomas.schneider@informatik.stud.uni-erlangen.de`.