htw saar

Studiengang Kommunikationsinformatik (Master)
Prof. Dr.–Ing. Damian Weber

Sicherheit und Kryptographie – Praktische Übung 4

Die folgenden Aufgaben können Sie auf praktisch allen ISL–Rechnern lösen, beispielsweise `isl-s-01`, `isl-s-02` oder `isl-s-04`.

Aufgabe 1 (OpenSSL Frontend)

Diese Aufgabe soll Sie mit dem OpenSSL-Frontend vertraut machen.

Lesen Sie die Manualpage zu openssl(1), z.B. unter

`https://www.openssl.org/docs/apps/openssl.html`

Aufgabe 2 (OpenSSL Zufallsbits)

Lesen Sie über die Erzeugung von Zufallsbits mittels

```
openssl rand -h
```

Erzeugen Sie eine BASE64–kodierte Ausgabe von 128 Zufallsbits.

Aufgabe 3 (OpenSSL Message-Digest–Kommandos)

Listen Sie alle Message-Digest–Kommandos (sichere Hashfunktionen) auf.

Bilden Sie den SHA–1 Hashwert der Datei `/etc/passwd` mit Hilfe des openssl–Tools.

Das UNIX–System FreeBSD besitzt eigene Kommandos für die Anwendung von sicheren Hashfunktionen. Bilden Sie den SHA–1 Hashwert der Datei `/etc/passwd` mit Hilfe des `sha1`–Kommandos.

Bilden Sie die Hashwerte für diese Datei mit den Hashfunktionen SHA–1, SHA–256, RIPEMD–160,

Es gibt FreeBSD–eigene Kommandos für die Funktionen SHA–1, SHA-256, nutzen Sie diese zur Kontrolle: `sha1`, `sha256`. NB: Linux–User nutzen `sha1sum`.

Wieviele Bits haben die Hashwerte der benutzten Funktionen?

Aufgabe 4 (OpenSSL MACs)

Ein MAC ist ein Message–Authentication–Code, eine Hashfunktion, die durch einen geheimen Schlüssel parametrisiert ist. Zwei Teilnehmer die den gleichen geheimen Schlüssel benutzen können in dieser Weise eine symmetrische Variante der digitalen Signatur realisieren.

Bilden Sie einen SHA–1–HMAC (bzw. SHA–256–HMAC) für verschiedene Dateien in Ihrem Homeverzeichnis.

Aufgabe 5 (OpenSSL Passwort-Hashes)

Listen Sie die Möglichkeiten des `openssl passwd` Kommandos mit

```
openssl passwd -h
```

Hashen Sie das Paßwort ,,abcd" mit Hilfe des `openssl passwd` Kommandos (standard UNIX Algorithmus).

Hashen Sie das Paßwort ,,abcd" mit dem Startwert (salt) ,,eQ".

Hashen Sie das Paßwort ,,abcd" mit dem MD5–basierten Algorithmus.

Aufgabe 6 (OpenSSL symmetrische Verschlüsselung)

Listen Sie verfügbaren Verschlüsselungsalgorithmen auf.

Verschlüsseln Sie die Datei `/etc/passwd` mit dem aes-128-cbc Verfahren und speichern Sie diese als `passwd.enc`

Entschlüsseln Sie `passwd.enc`.

Wenn dies funktioniert, senden Sie verschlüsselte e–Mail–Nachrichten (als Attachment) an einen Ihnen wohlgesonnenen Kursteilnehmer und verraten ihm das Geheimnis zur Entschlüsselung auf einem zweiten Kanal (z.B. mündlich ;-).

Testen Sie hierbei auch noch folgende Verschlüsselungsmethoden: aes-256-cfb, cast5-ofb

Aufgabe 7 (OpenSSL asymmetrische Verfahren)

Rufen Sie die `genrsa` Hilfe auf: `openssl genrsa -h`

Erzeugen Sie einen RSA–Schlüssel von 2048 Bit und schreiben ihn dabei in die Datei `rsakey.pem`.

Rufen Sie die `rsa` Hilfe auf: `openssl rsa -h`

Extrahieren Sie den Public Key aus `rsakey.pem`.

Rufen Sie die `dgst` Hilfe auf: `openssl dgst -h`

Signieren Sie die Datei `/etc/passwd` mit Ihrem RSA-Key und dem SHA–1 Hashverfahren.

```
openssl dgst -sha1 -sign rsakey.pem -out signature1 someInputFile
```

Verifizieren Sie die erzeugte Signatur mit dem oben extrahierten Public Key.

Führen Sie einen ähnlichen Vorgang mit DSA–Schlüsseln durch (diese sind abgeleitet vom ElGamal–Verfahren).

Aufgabe 8 (OpenSSL BASE64-Kodierung)

Finden Sie heraus, wie mit `openssl` eine Datei in BASE64–Kodierung umgewandelt werden kann und umgekehrt.

Hinweis: `openssl enc -h`

Erläutern Sie das Padding in BASE64 mit Hilfe des Zeichens „=".