Praktikum 4 zur Vorlesung IT-Sicherheit

Thema OpenVPN

Im Rahmen dieses Praktikums setzen wir OpenVPN zur Absicherung der Kommunikation zwischen Rechnern ein. OpenVPN ist frei im Internet erhältlich. Bei der Installation wurden die easy-rsa Tools mit installiert. Das ist eine Reihe von Skripten mit denen für OpenVPN benötigte X.509-Zertifikate per OpenSSL vereinfacht erstellt werden können.

In der Praxis sind mobile Firmengeräte häufig so konfiguriert, dass Sie nur eine VPNVerbindung zu einem vorgegebenen Firmenserver zulassen. Alle weiteren Verbindungen, z.B. ins Internet, erfolgen dann vom mobilen Gerät aus über den Firmenserver und die Firewall des Unternehmens.

Für Windows Rechner ist z.B. die OpenVPN Version unter https://www.heise.de/download/product/openvpn-22153 geeignet. Die easy-rsa Tools mit installieren!

Zum Austesten von VPN-Zugriffen wird des Weiteren ein Webserver auf einem der Rechner benötigt. Im Praktikum wird hier die Suite XAMPP verwendet, die Sie z.B. auch auf Heise finden. (Sie dürfen aber auch einen anderen Webserver nehmen):

https://www.heise.de/download/product/xampp-10929 Des Weiteren benötigen Sie Wireshark.

Die Versuchsbeschreibung ist auf den Einsatz PCs ausgelegt. Sie können anstelle dessen auch Virtuelle Maschinen oder Docker nutzen. Es braucht auch kein Windows zu sein.

Aufgabe 1: Peer-to-Peer-Verbindung via Pre-Shared-Keys

# a) Vorbereitung

Wählen Sie zwei PCs, zwischen denen die Absicherung erfolgen soll.

Nutzen Sie die IP-Adressen 192.168.0.1 (Client) und 192.169.0.2 (Server) mit SubnetMaske 255.255.255.0. Stellen Sie Ethernet-Konnektivität zwischen den PCs her und testen Sie die Erreichbarkeit in beide Richtungen per Ping.

Starten Sie auf dem Server-PC den Apache Web-Server unter XAMPP (Zugriff erlauben) und auf dem Client-PC den Firefox-Browser. Geben Sie im Firefox die URL http://192.168.0.2 ein. Die Default Server-Seite des XAMPP sollte geladen werden.

Starten Sie Wireshark mit dem Capture-Filter tcp port 80. Wiederholen Sie nach dem Löschen des Browser-Caches den Seitenaufruf und interpretieren Sie die von Wireshark aufgezeichneten PDUs. Lokalisieren Sie den HTML-Seitenquelltext in den PDUs.

Check O

# b) Erzeugung und Verteilung eines Pre-Shared-Keys

Starten Sie auf einem der PCs ein Kommandozeilenfenster (Eingabeaufforderung) mit Administrationsrechten (Rechte Maustaste → als Admin ausführen).

Wechseln Sie ins Verzeichnis C:\Programme\openvpn\bin.

Einen Pre-Shared-Key erzeugen sie im Kommandozeilenfenster durch das Kommando: openvpn --genkey --secret static.key

Beim Einsatz von Pre-Shared-Keys nutzen die kommunizierenden Rechner denselben

Schlüssel (symmetrische Verschlüsselung). Daher muss die Datei static.key mit dem Schlüssel sowohl auf dem Client als auch auf dem Server kopiert werden. Als Schlüsselverzeichnis wird das Konfigurationsverzeichnis config (alle angegebenen Verzeichnisnamen beziehen sich auf den Basispfad Programme\OpenVPN\)verwendet. Kopieren Sie die Datei static.key auf beiden Rechnern in das Verzeichnis.

# c) Konfiguration von Server und Client

Jetzt müssen Server und Client konfiguriert werden. Dies erfolgt mittels einer

Konfigurationsdatei (\*.ovpn), die im Konfigurationsverzeichnis config stehen muss.

Kopieren Sie hierzu zunächste die Datei sample.ovpn aus dem Verzeichnis sampleconfig in das Verzeichnis config. Bennen sie die Datei passend um in clientPSK.ovpn bzw. serverPSK.ovpn.

Öffnen Sie die Konfigurationsdatei mit einem Text-Editor und passen Sie die Konfiguration wie folgt an:

Ändern Sie remote myremote, in dem Sie myremote durch die IP-Adresse des

Kommunikationspartners ersetzen. Ergänzen Sie darunter die eigene IP-Adresse:

local 192.168.0.x # x=1 bzw. 2

Hierdurch wird OpenVPN nur an das betreffende Netzinterface gebunden.

Die Absicherung soll über Port 1194 und UDP erfolgen.

Als virtuelles Netzwerkadapter soll „TUN“ verwendet werden. TUN und TAP sind virtuelle Netzwerk-Kerneltreiber. TUN simuliert ein Layer-3 Interface, während TAP ein EthernetInterface simuliert (Layer 2).

Der Client erhält die virtuelle Adresse 10.3.0.1 und der Server 10.3.0.2. Passen Sie ifconfig … entsprechend an. Auch tun-mtu 1500 auf beiden Seiten auskommentieren.

Geben Sie die korrekte Schlüsseldatei an: secret …

Die anderen Einstellungen können unverändert übernommen werden.

# d) Freischalten der benötigten Kommunikationsbeziehungen auf der Firewall

Bevor Sie OpenVPN starten müssen Sie sicherstellen, dass eine Kommunikation (UDP) über den genutzten Port (1194) möglich ist. Hierzu fügen Sie eine neue Regel in die Windows-Firewall ein, die eingehenden Verkehr auf UDP Port 1194 erlaubt.

Da die Windows 10 Defender Firewall auch die Verbindungen zwischen den virtuellen Adressen blockt, dürfen Sie für den Versuch die Firewall komplett deaktivieren (sowohl fürs private Netzwerk als auch für das öffentliche Netzwerk). In der Praxis sollten Sie das jedoch vermeiden!

# e) Starten von OpenVPN

Das Starten erfolgt entweder auf der Kommandozeile durch openvpn xxx.ovpn, wobei xxx.ovpn der Name der entsprechenden Konfigurationsdatei ist, oder durch Rechtsclick auf die Konfigurationsdatei und Auswählen von Start OpenVPN on this config file.

Im Verbindungsfenster sollte dann angezeigt werden:

Peer Connection Initiated with ...

...

Initialization Sequence Completed

Falls das nicht der Fall ist, suchen Sie den Fehler und starten Sie OpenVPN erneut.

Mit welchem Algorithmus und mit

welcher Schlüssellänge wird verschlüsselt? BF-CBC Schlüssellänge: 128

Recherchieren Sie, wofür die 2 Buchstaben Abkürzung steht: Blowfish

Welcher Algorithmus wird für die Authentisierung verwendet? SHA-1

# f) Nutzung von AES und SHA-256

Stoppen Sie die laufende VPN-Verbindung und ergänzen Sie am Ende in beiden

Konfigurationsdateien die Zeilen cipher AES-256-CBC auth SHA256

Starten Sie OpenVPN erneut und kontrollieren Sie, ob die konfigurierten Verfahren eingesetzt werden:

Verschlüsselung mit AES-256-CBC: Check O

(Nachrichten-)Authentisierung mit SHA-256: Check O

# g) Test von OpenVPN und Beobachten der Kommunikation

OpenVPN hat den Rechnern ein neues (virtuelles) Interface hinzugefügt. Schauen Sie sich Details zum virtuellen Interface mit ipconfig /all an. Die geschützte Kommunikation verläuft über dieses Interface.

Starten Sie Wireshark für die Netzwerkkarte, über die die PCs physisch miteinander verbunden sind (also nicht das virtuelle Interface), mit dem Capture-Filter tcp port 80. Geben Sie im Client-Browser die URL http:\\10.3.0.2 des Servers ein. Was beobachten Sie und warum ist das so?

|  |
| --- |
| Nichts, da die Anfrage über das VPN läuft |

Konfigurieren Sie jetzt für die Netzwerkkarte als Capture-Filter udp port 1194, löschen Sie den Browser-Cache und Laden Sie die Seite erneut.

Was beobachten Sie und warum ist das so?

|  |
| --- |
| Die Verschlüsselten Pakete, da OpenVPN die Daten ja trotzdem irgendwie übertragen muss |

Aufgabe 2: Einsatz von Zertifikaten zur

Authentisierung und Schlüsselaushandlung

Um Zertifikate server- und clientseitig einzusetzen wird eine PKI benötigt, d.h. eine Zertifizierungsstelle (CA), die sowohl das Serverzertifikat als auch die Clientzertifikate ausgestellt hat. Zur Erzeugung der Zertifikate wird OpenSSL verwendet, das Bestandteil von OpenVPN ist. Zur Vereinfachung der Bedienung gibt es in OpenSSL im Verzeichnis easy-rsa einige \*.bat-Skripte.

Verwenden Sie einen der beiden Rechner als CA. Sie können den Versuch auch mit mehr als einem Client durchführen. Die Versuchsbeschreibung im Folgenden bezieht sich auf die Verwendung von 4 Rechnern: eines Rechners als CA, einem Server und 2 Clients.

Wenn nur 2 Rechner zur Verfügung stehen, betreiben Sie auf dem OpenVPN Server auch die CA und realisieren nur einen OpenVPN Client.

# a) Erstellen eines CA-Schlüssels und eines CA-Zertifikats

Öffnen Sie auf dem CA-Rechner ein Kommandozeilenfenster als Administrator und wechseln Sie in das Verzeichnis Programme\OpenVPN\easy-rsa. Führen Sie dort init-config aus, das eine Kopie der Datei vars.bat.sample erstellt: copy vars.bat.sample vars.bat

Openssl-1.0.0.cnf ist eine für OpenVPN vordefinierte OpenSSL-Konfigurationsdatei. In dem Skript vars.bat werden Variablen vorbelegt, die bei der Erzeugung der verschiedenen Zertifikate genutzt werden. Neben Verzeichnis- und Dateiangaben sind dies insbesondere die RSA-Schlüssellänge (1024 Bit) und Bestandteile des X.500Distinguished Names (DN), die in den verschiedenen Zertifikaten übereinstimmen müssen.

Editieren Sie die Datei vars.bat (Rechte Maustaste => Bearbeiten) und Ersetzen Sie die DN-Vorbelegungen durch:

set KEY\_COUNTRY=DE # Staat

set KEY\_PROVINCE=Niedersachsen # Bundesland set KEY\_CITY=Osnabrueck # Stadt set KEY\_ORG=HS-Osnabrueck # Firma/Organisation set KEY\_EMAIL=admin@fhos.de # E-Mail-Adresse set KEY\_CN=CA set KEY\_NAME=CA set KEY\_OU=KN-Labor set PKCS11\_MODULE\_PATH=CA

Zum Setzen der Variablen rufen Sie danach vars in der Kommandozeile auf.

Danach wird das Skript clean-all in der Kommandozeile aufgerufen, das ein ggf. schon vorhandenes key-Verzeichnis löscht, ein neues key-Verzeichnis anlegt und in dem Verzeichnis eine leere CA-Datenbank (index.txt) und eine Datei serial erzeugt, in der die aktuelle Zerifikats-Seriennummer gespeichert wird.

Nach diesen vorbereitenden Schritten kann jetzt durch Aufruf von build-ca in der Kommandozeile der geheime CA-Schlüssel und das CA-Zertifikat erzeugt werden.

Übernehmen Sie die durch vars.bat eingestellten DN-Vorbelegungen und geben Sie als CommonName (CN) OpenVPN-CA und als E-Mail Adresse openvpn-ca@fhos.de ein.

Im Unterverzeichnis keys sollten jetzt der CA-Schlüssel ca.key und das selbstsignierte CA-Zertifikat ca.crt stehen. Öffnen Sie ca.crt und kontrollieren Sie die Einträge.

Mit welchem Algorithmus ist das Zertifikat unterzeichnet? sha256RSA

Welcher öffentliche RSA-Exponent wird verwendet

(letzen 3 Byte des öff. Schlüssels)? 01 00 01

# b) Erstellen eines Serverschlüssels und -zertifikats

Die Erzeugung des Serverschlüssels und -zertifikat erfolgt durch das Skript build-keyserver.bat, dass zunächst einen (selbstsignierten) Zertifizierungsantrag (request) erzeugt.

Editieren Sie build-key-server.bat:

Welches OpenSSL-Kommando dient zum Erzeugen eines Schlüssels und des Zertifizierungsantrags? openssl req -days 3650 -nodes -new -keyout %KEY\_DIR%\%1.key -out %KEY\_DIR%\%1.csr -config %KEY\_CONFIG%

Welche Endung hat die Datei, in der der Schlüssel gespeichert wird? .key

Welche Endung hat die Datei, in der der Zertifizierungsantrag gespeichert wird? .csr

Mit einem zweiten OpenSSL-Kommando in build-key-server.bat wird dann der Zertifizierungsantrag von der CA signiert.

Wie lautet das OpenSSL-Kommando inkl. der Option mit der die Gültigkeit des ausgestellten Zertifikats festgelegt wird? openssl ca -days 3650

Wie lange sollte das Serverzertifikat demnach gültig sein? 10 Jahre

Welche Endung hat die Datei, in der das Zertifikat gespeichert wird? .crt

Rufen Sie jetzt build-key-server server in der Kommandozeile im Verzeichnis easy-rsa auf. Bei der Ausführung Übernehmen Sie wiederum die durch vars.bat eingestellten DN-Vorbelegungen und wählen als CN einen Servernamen (z. B. Server). Die Fragen Sign the certificate? [y/n] und 1 out of 1 certificate requests certified, commit? [y/n] bestätigen Sie mit y.

Prüfen Sie, ob die erzeugten Dateien den obigen Antworten entsprechen. Check O

Für einen etwaigen Diffie-Hellman-Schlüsselaustausch benötigt der Server noch DHParameter. Diese werden durch Ausführung von build-dh in der Kommandozeile im Verzeichnis easy-rsa erzeugt: Datei dh2048.pem

# c) Erstellen von Clientschlüsseln und –zertifikaten

Für jeden Client muss ein Schlüsselpaar erzeugt und zertifiziert werden.

Erstellen Sie Schlüssel und Zertifikate für zwei Clients durch Aufruf von build-key client1 und build-key client2 in der Kommandozeile im Verzeichnis easy-rsa. Wählen Sie als CN jeweils eindeutige Namen (z. B. Client1 und Client2). Das weitere Vorgehen entspricht dem Vorgehen beim der Erstellung des Serverzertifikats.

Prüfen Sie die erzeugten Zertifikate. Check O

Editieren Sie build-key.bat. Die Client-Zertifizierung unterscheidet sich nur in einer Option von der Server-Zertifizierung. Wie lautet die zugehörige Option und für welche Zertifizierung (Server oder Client) ist sie erforderlich?

|  |
| --- |
| -extensions server |

d) Verteilung von Schlüsseln und Zertifikaten auf die Rechner.

Überlegen Sie anhand der Informationen aus der Vorlesung, welche Schlüssel/Zertifikate von wem benötigt werden und welche der Informationen geheim sind:

|  |  |  |  |
| --- | --- | --- | --- |
| Dateiname | enthält | Wird benötigt von | geheim? |
| ca.crt | CA Zertifikat | Client/Server | Nein |
| ca.key | CA Schlüssel | Ca | Ja |
| dh2048.pem | Diffie Hellman Parameter | Server | Nein |
| server.crt | Server Zertifikat | Client | Nein |
| server.key | Server Schlüssel | Server | Ja |
| client1.crt | Client1 Zertifikat | Server | Nein |
| client1.key | Client1 Schlüssel | Client 1 | Ja |
| client2.crt | Client2 Zertifikat | Server | Nein |
| client2.key | Client2 Schlüssel | Client 2 | Ja |

Kopieren Sie nur die benötigten Dateien in das Verzeichnis config der jeweiligen Rechner.

# e) OpenVPN Server- und Clientkonfiguration

Kopieren Sie die Konfigurationsdatei client.ovpn bzw. server.ovpn aus dem Verzeichnis sample-config in das Verzeichnis config. Anpassung der Datei server.ovpn:

local 192.168.0.2 #eigene IP des Servers

Kommunikation über UDP Port 1194 und ein TUN Interface.

Anpassung der Dateinamen des CA-Zertifikats, des Serverzertifikats, des Serverschlüssels und der DH-Datei.

Aktivierung der Option client-to-client, damit eine Kommunikation zwischen Clients möglich ist.

tls-auth ta.key 0 deaktivieren (=> ;tls-auth ta.key 0) topology subnet aktivieren (auskommentieren)

Auswahl einer Verschlüsselung: cipher … (identisch in Client-Konfig ergänzen!) Was bewirkt die Einstellung server 10.8.0.0 255.255.255.0 ?

|  |
| --- |
| VPN hat Ip 10.8.0.0 mit Subnetzmaske 255.255.255.0 |

Anpassung/Kontrolle der Konfigurationsdateien der Clients:

Identifizierung als Client

Kommunikation über UDP und TUN Interface

Server IP-Adr. und Portnummer: remote 192.168.0.2 1194

Anpassung der Dateinamen des CA-Zertifikats, des Clientzertifikats und des Clientschlüssels.

Angabe der Verschlüsselung: cipher … (identisch zum Server!) tls-auth ta.key 1 auskommentieren (=> ;tls-auth ta.key 1)

# f) Start und Test von OpenVPN

Starten Sie OpenVPN auf den Rechnern wie bei Aufgabe 1 beschrieben.

Falls der Verbindungsaufbau fehlschlägt gehen Sie auf Fehlersuche.

Ermitteln Sie die vom VPN-Server zugeteilten Adressen der virtuellen Interfaces.

|  |  |
| --- | --- |
| Phsyische Adresse | Virtuelle Adresse |
| 192.168.0.106 | 10.8.0.2 |
| 192.168.0.121 | 10.8.0.1 |
| n/a |  |

Rufen Sie die virtuelle IP-Adresse des Servers als URL im Browser

der Clients auf. Check O

Kontrollieren Sie mit Wireshark, dass eine Verschlüsselung erfolgt. Check O

Bei mehreren Client-Rechnern: Testen Sie mit Ping die Erreichbarkeit der Clients

untereinander über die virtuellen Adressen. Check O