Übungsblatt 03 – Wireshark

Aufgabe A - Unencrypted Password Sniffing

Nachdem Sie nun auch praktisch mit Wireshark Ihre ersten Erfahrungen gesammelt haben, sollen Sie mithilfe des Sniffers Passwörter im unverschlüsselte Traffic "dumpen". Dazu ist ein kleines Setup notwendig.

1. Um das Passwort-Sniffing etwas zu erleichtern, soll der Netzwerkverkehr über einen neugierigen Router erfolgen. Passen Sie die Routing-Tabelle und das Forwarding wie folgt an:
   - In einer Bankreihe agieren je zwei Web-Server. Diese sollen einfachheitshalber Ihr Default-Gateway auf den sniffenden Rechner legen. Mit folgenden Kommando könne Sie dies realisieren:
   - Die IP-Adresse können Sie mithilfe folgenden Befehls in Erfahrung bringen:

```
ip a s DEV | awk '/inet/ {print $2}'
```

   - Stoppen des Network-Mangers:

```
sudo systemctl stop network-manager
sudo systemctl stop network-manager
```

   - Ändern der Default-Route:

```
# anzeigen der devices
ip l
# anzeigen des routing tables
ip r
# loeschen der default route
sudo ip r del default
# setzen einer neuen default route
sudo ip r add default via XXX.XXX.XXX.XXX dev DEV
```

     Wobei XXX.XXX.XXX.XXX der IP-Adresse des Sniffers entspricht. DEV bezeichnet den Identifier des verwendeten Geräts.
   - Der Sniffer muss das Forwarding aktivieren, sodass Daten weiterhin an Ihre Ziel-Adressen ankommen:

```
sudo sysctl -w net.ipv4.ip_forward=1
```

2. Der Apache Webserver liefert Ihnen nur die Default-Seite.

a) Nehmen Sie für die Konfiguration des Webservers ein Backup vor! Alle Dateien die Sie ändern müssen, sollen zuvor gesichert werden. Kopieren Sie entsprechend die Dateien mit den Ihnen bekannten Kommandozeilenbefehlen im gleichen Ordner. Folglich sollen sich im gleichen Ordner die Backups wie auch die Originaldateien befinden.
Die Kopie kann beispielsweise die Dateiendung *.bck* tragen. [1]

b) Nicht jeder Nutzer soll auf den Inhalt Ihrer Webseite zugreifen dürfen, daher soll eine einfache Passwortabfrage den Inhalt Ihrer Website sichern.
Richten Sie eine Passwortauthentifizierung ein, die auf dem Webserver *A* dem Nutzer `web` und auf Webserver *B* dem User `bew` Zugriff gewährt. Allen anderen Nutzern soll kein Zugriff erlaubt sein!

3. Als Hilfestellung für den Webserver können Sie wie folgt vorgehen:
   - Für das Binding des Webservers muss in der Apache Konfiguration (s. `/etc/apache2/apache2.conf`) die IP-Adresse und optional der Port mit dem Befehl *Listen* gesetzt werden.

```
Listen IP:Port
```

   - Die Passwortauthentifizierung kann mithilfe des Kommandos *htpasswd* eingeleitet werden.

```
sudo htpasswd -c /etc/apache2/.htpasswd YOURUSERNAME
```

   - Anschließend kann in der Datei `/etc/apache2/apache2.conf` entsprechend der Inhalt Ihrer Website geschützt werden.

```
<Directory "/var/www/html">
  AuthType Basic
  AuthName "Speak, friend and enter"
  AuthUserFile "/etc/apache2/.htpasswd"
  Require user YOURUSERNAME

  Order allow,deny
  Allow from all
</Directory>
```

   - Mit dem Tool *apachectl* kann die Konfiguration des Webservers überprüft und anschließend der Apache hochgefahren werden.

```
sudo apachectl configtest
sudo apachectl start
```

4. Der Administrator des Sniffers ist überaus neugierig und soll die verwendeten Nutzernamen/Passwort Kombinationen ausschließlich durch Analyse des Netzwerkverkehrs in Erfahrung bringen. [2]

   a) Analysieren Sie den Traffic! Nach welchem Protokoll müssen Sie suchen?

   b) Stellen Sie entsprechen den Filter in Wireshark ein.

   c) Finden Sie das Tupel aus Nutzernamen und Passwort.
   Wie können Sie im gesamten Verkehr noch weiter filtern, sodass Sie das Paket mitsamt Nutzernamen und Passwort finden?
   Hinweis: Es kann passieren, dass der Browser die Website im Zwischenspeicher behält (cached), sodass Ressourcen gespart werden können. Möglicherweise müssen Sie entsprechend den Browser-Cache leeren, bevor Sie Änderungen im Browser sehen können.

5. Setzen Sie die vorgenommenen Änderungen wieder zurück. Schalten Sie auch den Apache ab

```
# abschalten des Apaches
sudo apachectl graceful-stop
# löschen der default route
sudo ip r del default
# dhcp neu starten
sudo systemctl restart dhcpcd
# Forwarding deaktivieren
sudo sysctl -w net.ipv4.ip_forward=0
```

Aufgabe B – TCP: 3-Way-Handshake

Nachdem Sie sich bereits theoretisch mit dem 3-Way-Handshake auseinandergesetzt haben, sollen Sie nun schauen, ob der TCP-Handshake tatsächlich wie theoretisch beschrieben arbeitet.

1. Überlegen Sie sich eine Anfragen an eine Website (dies sollte TCP nutzen, wie HTTP!), die Sie noch nicht von der VM aus getätigt haben. Da ansonsten bestimmte Inhalte bereits gecacht vorliegen könnten oder über andere Verfahren eine TCP-Handshake vereiteln könnten.

2. Starten Sie Wireshark, richten Sie Interface und Protokoll-Type ein. Filtern Sie nur auf eine speziellen Request!

3. Lösen Sie den Handshake durch aufrufen der Website (oder Ressource) aus, während Wireshark den Netzverkehr mitschneidet.

[2] Dieses Szenario ist sehr fingiert, soll aber nur verdeutlichen, dass ohne Schutz unverschlüsselte Daten leicht einsehbar sind! Dies ist auch der Fall, wenn Daten nicht direkt über einen Rechner gehen – s. Promiscous-Mode oder im WiFi-Verkehr

4. Analysieren Sie den 3-Way-Handshake!
5. Zum Vergleich: Analysieren Sie ihren Mitschnitt mit folgender Aufzeichnung: https://wiki.wireshark.org/TCP_3_way_handshaking?action=AttachFile&do=view&target=3-way+handshake.pcap

Aufgabe C – ICMP

Da die Befehle *ping* und *traceroute ICMP* nutzen, sollen Sie mit Wireshark solche Request mitverfolgen.

1. Setzen Sie alle notwendigen Parameter um Wireshark mitlaufen zu lassen, sodass Sie die ICMP-Nachrichten mitverfolgen können.
2. Pingen Sie einen Rechner mit seinem Namen an (bspw.: mi.fu-berlin.de).
3. Ping auf eine IP-Adresse (bspw.: 160.45.117.199).
4. Ping auf die IP-Adresse des Laborrouters (IP: 10.10.10.254).
5. Ping auf meine eigene IP-Adresse.
6. Starten Sie traceroute auf eine beliebige Adresse und verfolgen Sie dabei den Ausgabe auf der Konsole als auch in Wireshark. Spiegeln sich die Einträge in Wireshark mit denen auf der Kommandozeile?

Aufgabe D – Routing & Traceroute

Nachdem Sie recherchiert haben, wie *traceroute* arbeitet, welche Kritik an Traceroute geäußert wurde und wie diese mit dem Tool Paris-Traceroute abgestellt wurden, sollen beide Tools hier kurz erprobt werden.

1. Überlegen Sie sich zunächst anhand Ihrer Recherche was *traceroute* in etwa ausgeben müsste, wenn Sie im Labor eine Route von einem Rechner A zu einem Rechner B verfolgen würden. Wobei beide Rechner zu unterschiedlichen Netzwerken gehören (d.h. unterschiedlichen Tischreihen).
2. Nutzen Sie anschließend *traceroute* um sich die Router zwischen zwei Laborrechnern anzeigen zu lassen. Stimmen Ihre theoretische Überlegungen mit denen von *traceroute* überein? Falls nicht, sollten Sie analysieren woran dies liegen könnte.
3. Vergleichen Sie die Ausgaben von *traceroute* und *paris-traceroute* für folgende IP-Adressen:
   a) 41.231.21.44

b) 91.198.174.192

c) 37.220.21.130

d) 80.239.142.229

Hinweis: Für *paris-traceroute* sollten Sie den „exhaustive algorithm“ Nutzen (in machen Versionen als Parameter: `-na exhaustive`)

4. Analysieren Sie anschließend die Ausgabe beider Tools.

5. Warum wurde Ihnen eine Liste von IP-Adressen genannt anstelle von Domainnamen? Nennen Sie mindestens zwei Gründe!