htw
Hochschule für Technik
und Wirtschaft Berlin
University of Applied Sciences

Übungsblatt 6 – Application Layer

Hinweis: Versuchen Sie die Übungsblätter soweit wie möglich ohne Hilfe von Google, Stackoverflow, Stackexchange zu lösen. Sie sollen eigene Lösungswege finden und nicht professionell Suchmaschinen bedien können. Ausnahmen sind natürlich Aufgaben, in denen explizit recherchiert werden soll

Aufgabe A – Secure Shell mit openSSH

Die folgenden Aufgaben stellen verschiedene Arten der verschlüsselten Kommunikation zwischen Prozessen oder Rechnern dar – von einer einfachen Verschlüsselung einzelner Ports zum sicheren Zugriff auf einzelne Dienste. Auch das Aufsetzen eines kompletten VPN, das transparent für alle darüber laufenden Services ist, wäre mit SSH möglich.

Voraussetzungen: Da Sie bis jetzt Ihr eigenes Netzwerk aufgesetzt haben und u.U. kein DNS und Uplink ins Internet vorhanden ist, sollte dies geändert werden. In diesem Fall können Sie alle persistierten Einträge löschen und anschließend die Raspberry Pis mit DHCP betreiben.
Sie sollte hierfür die Konfigurationen in der */etc/network/interfaces* überprüfen (default in Raspbian – s. Listing).

```
1 # interfaces(5) file used by ifup(8) and ifdown(8)
2
3 # Please note that this file is written to be used with dhcpcd
4 # For static IP, consult /etc/dhcpcd.conf and 'man dhcpcd.conf'
5
6 # Include files from /etc/network/interfaces.d:
7 source-directory /etc/network/interfaces.d
```

Wenn Sie den DNS-Dienst selbst konfiguriert haben, ändern Sie entsprechend die */etc/resolv.conf*. Anschließend muss der Networking-Service abgeschaltet und das DHCP wieder eingeschaltet werden. Achten Sie darauf, dass Ihre Hostnames korrekt gesetzt wurden, d.h. sowohl in der */etc/hostname* als auch in der */etc/hosts*.

```
1 #Abschalten des Networking-Service
2 sudo systemctl status networking.service
3 sudo systemctl stop networking.service
4 sudo systemctl disable networking.service
5 # Einschalten des DHCP
6 sudo systemctl enable dhcpcd
```

```
reboot
# Nach dem Reboot
sudo systemctl status dhcpdcd
ping -c 1 google.de
```

1.) Loggen Sie sich auf dem Raspberry Pi ein. Auf den Raspberry Pis kann mit *startx* die GUI gestartet werden, sodass Sie die Verbindungen auch via Wireshark analysieren können.

a.) Loggen Sie sich via SSH auf dem Uranus-Server (`uranus.f4.htw-berlin.de`) ein!
Alternativ können Sie sich auch auf einen anderen Raspberry Pi einloggen.

b.) Was bedeuten die Abfragen zur „authenticity“ die Ihnen beim ersten mal gestellt wird.

c.) Wie können Sie den Fingerprint prüfen? Mit welchem Programm können Sie sich diesen anzeigen lassen?
Bspw.:. *SHA256:KsUg4lOc91/iJBYFkQhxeI/YGkcnKV2uKUXFNP1ymiw root@xen (ECDSA)*)

d.) Starten Sie in Wireshark einen neuen Traffic-Mitschnitt auf dem Netzwerk-Interface *eth0*. Anschließend soll eine neue SSH-Session von einem anderen Rechner gestartet werden. Analysieren Sie auszugsweise die entsprechenden Pakete! Was wird von Traffic verschlüsselt, was können Sie einsehen?

Finden Sie das OSI-Modell bei der Analyse wieder? D.h. ist dort eine Art Hierarchie/ Verschachtelung wiederzuerkennen?

Sie müssen sich bis jetzt immer via Passwort authentifizieren, d.h. Ihr Login erfolgt aufgrund eines Passworts. Ist Ihr Passwort in einem der ersten Pakete zu finden? Wenn es nicht zu finden ist, wie können Sie sich dennoch erfolgreich anmelden?

e.) Wenn Sie die entsprechenden Wireshark Mitschnitte ausgewertet haben, ist Ihnen aufgefallen, dass dort ein „Key Exchange“stattfindet. Welches kryptografische Verfahren wird dort verwendet und ist dies eine symmetrisches oder asymmetrisches Kryptografieverfahren?

2.) Ermöglichen Sie nun das Login mittels SSH zum Linux-SSH-Server **ohne** das Nutzerpasswort angeben zu müssen. **Achtung:** Wenn Sie sich auf dem Uranus ohne Passwort anmelden wollen, muss eine bereits existierende SSH-Verbindung auf dem Uranus-Server vorhanden sein, da ihr Home-Directory erst im Anschluss gemountet wird und ihr hinterlegter Public-Key ansprechbar ist.

a.) Generieren Sie sich einen SSH-Schlüssel! Recherchieren Sie **kurz** welche Schlüssellänge und welche Schlüsselarten für Ihren Einsatz im Labor sinnvoll sind. Wie hängen Schlüssellänge und Sicherheit zusammen?

Beim generieren des Schlüssels werden Sie aufgefordert eine Passphrase einzugeben. Was ist das und ist die Passphrase gleichzusetzen mit dem Schlüssel oder Passwort?

b.) Verbindung von Linux zu Linux: Verbinden Sie sich von Rechner zu Rechner ohne ein Passwort zu nutzen. Wenn Ihr Raspberry Pi vom Laborrechner aus erreichbar ist können Sie von dort auch versuchen sich einzuloggen (Das wäre auch unter Windows mit dem Tool PuTTy möglich!)

c.) Lassen sich die SSH-Schlüssel zwischen den verschiedenen Clients weiterverwenden/konvertieren? Oder muss andernfalls für jeden Client ein eigener Schlüssel generiert werden.

d.) Wie kann aus Sicherheitsgründen ein Login ohne Passwort eingeschränkt werden, so das nur bestimmte Kommandos via SSH ausgeführt werden können?

e.) In manchen Fällen ist es ratsam den Zugriff via SSH nur auf einige Nutzer zu beschränken. Recherchieren Sie wie das aussehen müsste.

f.) Setzen Sie die Anzahl der maximalen Login-Fehlversuche auf drei!

g.) Erlauben Sie dem Nutzer Pi nur noch das Auflisten des Home Verzeichnis, wenn er sich via SSH verbunden hat.

h.) Setzen Sie als Anmeldeverfahren SSH auf reine Public-Key-Kryptografie. Hat dies eventuell auch Nachteile?

3.) Mit SSH können Sie beliebige TCP-Verbindungen über die verschlüsselte SSH-Verbindung „tunneln“. Somit wird es Ihnen möglich, Server zu erreichen, zu denen Sie ansonsten direkt keinen Zugriff haben, weil sie hinter einer Firewall stehen oder der Datenverkehr anderweitig gefiltert wird.
Konfigurieren Sie das Portforwarding unter SSH. Ermöglichen Sie dazu folgende Zugriffe:

a.) Recherchieren Sie kurz welche Weiterleitungsmöglichkeiten SSH Ihnen bietet.

b.) Sie sollen von Ihrem Raspberry Pi aus ein lokales Portforwarding auf die Seite der HTW vornehmen. Hierzu soll ein SSH-Tunnel aufgebaut werden mit den Source-Port 8080 und dem HTTP-Port 80 für den Ziel-Port.

c.) Ihr Raspberry Pi logt sich per SSH auf anderen Raspberry Pi SSH-Server ein und leitet den lokalen Port 2200 auf den Port 22 des dortigen Systems weiter. Danach sollten Sie sich mit SSH über den lokalen Port mit dem SSH-Server des fremden SSH-Server verbinden können.

Aufgabe B – Domain Name System – DNS

Das Domain Name System ist ein dezentrales System, dessen primäre Aufgabe die Adressauflösung von Domain zu IP-Adresse. M.a.W. DNS bietet eine Abbildung von Domain auf IP-Adresse. Im Laufe der Jahre sind hierzu einige Tools entwickelt worden: whois, host, dig, nslookup.

1.) DNS Informationen abfragen

a.) Recherchieren Sie kurz wie die einzelnen Tools zu benutzen sind! Wird eine Empfehlung abgegeben, welche Tools heute nicht mehr genutzt werden (sollten)?

b.) Fragen Sie mit jedem der vier Tools auf der Kommandozeile jeweils einmal einen Hostnamen (bspw. `www.htw-berlin.de`), einen Domainnamen (htw-berlin.de) und eine IP-Adresse 141.45.5.100 ab.

c.) Schauen Sie sich die Ausgabe von *dig* bei der Abfrage der IP-Adresse genauer an – dort werden Sie in der „Question Section“ sehen, das nach dem A-Resource-Record mit dem Namen 141.45.5.100 gefragt wurde. Wenn Sie den Namen zu dieser IP-Adresse suchen – welchen Resource-Record müssen Sie dann anstelle des A-Records erfragen?

d.) In welcher Form müssen Sie dann die IP-Adresse angeben? (Test mit dig -t <record-type> <richtiges-format-ip-adresse>).

e.) Denken Sie sich einen Domainnamen aus, den es wahrscheinlich geben könnte, aber den noch niemand vom Netzwerk der HTW-Berlin aus innerhalb der letzten Stunden angefragt hat (z.B. www.uriminzokkiri.com oder www.northkoreatech.org). Erfragen Sie diesen Namen zweimal kurz hintereinander via dig und vergleichen Sie die beiden Ausgaben. Worin unterscheiden sich beide Einträge? Begründen Sie diese Unterschiede!

f.) Erfragen Sie mit host, dig und nslookup den zuständigen Mail-Server für die Domain `htw-berlin.de`.

g.) Erzwingen Sie mit *host, dig* und *nslookup*, das die Namensauflösung nicht mit dem Standard-Nameserver des Betriebssystems, sondern mit dem öffentlichen Nameserver (bspw.: 9.9.9.9) erfolgt. Testen Sie am Besten zuerst mit dig oder nslookup , da diese Ihnen immer sagen, welche Nameserver sie genutzt haben. *host* liefert diese Information nur, wenn Sie explizit eigene Server angefordert haben.

2.) DNS-Resolver: Das Listing zeigt die „resolv.conf“eines Servers.

```
1 # Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
2 # DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
3 nameserver 141.45.3.100
4 search f4.htw-berlin.de
```

a.) Was bedeuten die Einträge mit den Schlüsselwörtern: „nameserver“und „search“?