Übungsblatt 6 – Netzwerk-Security

Über das gesamt Semester hinweg haben Sie ein fundiertes Grundlagenwissen zu Netzwerken erworben. Sie haben mit dem Aufbau der Infrastruktur begonnen, indem Sie eigene kleine und etwas komplexere Netzwerke aufgebaut und betrieben haben. Daran anknüpfend sind Sie in die Phase gegangen in der Sie sich die vielmals verwendeten Protokolle analysierten. Wodurch sich offenbare wie Geräte und Anwendungen in Netzwerken arbeiten.
Da Sie als angehende InformatikerInnen sich bewusst sind, dass Daten von Ihnen selbst, als auch Daten anderer Nutzer, schützenswert sind, beschäftigen Sie sich im letzten Übungsblatt mit den Themen Kryptografie und Netzwerksicherheit.

Aufgabe A – Kryptografie Grundlagen

Da Sie mit großer Wahrscheinlichkeit keine ausgebildeten Mathematiker sind, beginnen Sie zunächst mit einer kurzen Recherchephase. Dies soll Ihnen helfen Licht ins Dunkel zu bringen.
Hilfreiche Links:

- https://de.wikipedia.org/wiki/Kryptologie
- https://en.wikipedia.org/wiki/Cryptography
- https://www.cryptool.org → sehr schönes Tool! Zeigt & visualisiert Chiffren & Verfahren etc.

1.) Begriffsklärung Kryptografie & Chiffren:

a.) Was sind die grundlegende Sicherheitszielklassen im Bereich der IT-Security?

b.) Recherchieren Sie was sich hinter den Begriffen Kryptologie, Kryptografie und Kryptoanalyse verbirgt.

c.) Worin besteht der maßgebliche Unterschied zwischen symmetrischen und asymmetrischen Kryptosystemen?

d.) Welche Aufgaben könne kryptografisch per symmetrischen Chiffren bewältigt werden?

e.) Welche Aufgaben könne kryptografisch per asymmetrischen Chiffren bewältigt werden?

f.) Nennen Sie mindestens drei asymmetrische Krypto-Verfahren.

g.) Nennen Sie mindestens drei symmetrische Chiffrierverfahren.

h.) Was ist der Unterschied zwischen Stromchiffren und Blockchiffren?

i.) Recherchieren Sie kurz welche Aufgabe der Diffie-Hellmann-Algorithmus und der Elgamal-Algorithmus haben? Wozu werden diese Verfahren für gewöhnlich genutzt?

j.) Beschreiben Sie die Gefahr des Man-in-the-Middle-Angriffs bei Diffie-Hellmann.

k.) Recherchieren Sie zunächst was unter einer Hashfunktion verstanden wird. Im Anschluss daran: Was wird unter einer kryptografischen Hashfunktion verstanden?

l.) Was ist die Aufgabe einer kryptografischen Hashfunktion?

m.) Woran bemisst sich die Qualität einer kryptografischen Hashfunktion?

n.) Nennen Sie mindestens drei (aktuell ungebrochene) kryptografische Hashfunktionen.

o.) Fakultativ: Warum kann nicht gezeigt werden (d.h. mathematisch bewiesen werden), dass ein Hashfunktion Kollisionsresistent ist? (https://en.wikipedia.org/wiki/Collision_resistance)

2.) Public-Key-Kryptografie

a.) Was wird unter dem Begriff Public-Key-Kryptografie verstanden?

b.) Warum werden Public-Key-Verfahren eingesetzt? Warum sollte ein solches Verfahren genutzt werden?

c.) Erläutern Sie den Ablauf eines Public-Key-Verfahren im groben? Sie müssen hier keine mathematischen Feinheiten beachten, wichtig ist der grundlegende Gedanke!

3.) Kryptografische Zertifikate & Public-Key-Infrastruktur

a.) Was wird unter einem kryptografische Zertifikat verstanden? Welchen Nutzen hat dieses Zertifikat?

b.) Recherchieren Sie was unter einer Public-Key-Infrastruktur *PKI* verstanden wird.

c.) Im letzten Übungsblatt ist Ihnen diese des öfteren über den Weg gelaufen. Mit welcher *PKI* hatten Sie es zu tun? Was war die Aufgabe der *PKI*?

d.) Recherchieren Sie was im Zusammenhang mit *PKI*s unter dem Namen *Chain-Of-Trust* verstanden wird.

Aufgabe B – Grundlagen: Secure Shell (SSH) mit openSSH

Das gesamte Semester über haben Sie überwiegend lokal auf der Kommandozeile gearbeitet, also relativ nah an der eigentlich Hardware (abstandsmäßig :)). Viele Netzwerk-

und Serverkomponenten sind jedoch nicht lokal verfügbar (d.h. direkt, physisch), da diese in Rechenzentren unter besonderen Bedingungen ihren Dienst verrichten.[1] Die Administration der Rechner muss also auch entfernt möglich sein – remote.
Früher haben dies die sogenannten *r-Tools* ermöglicht, dies jedoch ohne kryptografische Schutzmaßnahmen. Heute übernehmen gesicherte Tools wie *SSH* mit verschiedensten Implementierungen, wie *openSSH*, diese Aufgabe.

1.) Lesen Sie folgendes SSH-Tutorial: https://support.suso.com/supki/SSH_Tutorial_for_Linux

2.) Welche vier Aufgaben, d.h. Zusicherungen in Bezug auf die Sicherheit von Daten, kann *SSH* mithilfe von kryptografischen Verfahren gewährleisten?

3.) Notieren Sie sich an welchen Orten die verschiedenen Konfigurationsdateien für Server und Client im Normalfall (default) unter einem Debian-Linux liegen. Notieren sie sich deren Zweck.

4.) Recherchieren Sie, was ein „Fingerprint“ im Sinne von *SSH* ist und welche Aufgabe dieser übernimmt.

5.) *SSH* kommt ohne Passwörter aus, es können Public-Key-Verfahren genutzt werden. D.h. Sie können *SSH* auch ohne Zugangspasswort nutzen.[2]
Recherchieren Sie welche Verfahren *openSSH* hierfür anbietet.

6.) Recherchieren Sie, wie die Schlüsselgenerierung in openSSH erfolgt. Wie sind Verfahren, Schlüsselgröße und zu speichernden Ort zu wählen? Notieren Sie sich die entsprechende Syntax!

7.) Welche Schlüssellänge und welche Schlüsselarten sind für Ihren Einsatz im Labor sinnvoll?
Wie hängen Schlüssellänge und Sicherheit zusammen?

8.) Lassen sich die *SSH*-Schlüssel zwischen verschiedenen Clients (Windows, Linux, Solaris,...) weiterverwenden/konvertieren? Oder muss andernfalls für jeden Client ein eigener Schlüssel generiert werden?

9.) Recherchieren Sie die Bedeutung der Passphrase. Ist die Passphrase mit dem Passwort gleichzusetzen?

[1] Sie würden bestimmt nicht direkt in einem Rechenzentrum Ihre Arbeit als Administrator ausführen! Da die Temperaturen oft unangenehm und die Lautstärke recht hoch ist. Auch die Sicherheitsbestimmungen sind enorm hoch.
[2] Eigentlich ist es ratsam auf Passwörter zu verzichten, da das Brechen von kryptografischen Schlüsseln momentan fast unmöglich ist.

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

11.) In manchen Fällen ist es ratsam den Zugriff via *SSH* nur auf einige Nutzer zu beschränken. Wie muss dies unter *openSSH* anhand eines Beispiels aussehen.

Aufgabe C – SSH Port-Forwarding

Mit *SSH* können Sie beliebige TCP-Verbindungen über die verschlüsselte *SSH*-Verbindung „tunneln“. [3] Somit wird es Ihnen möglich Server zu erreichen, zu denen Sie ansonsten direkt keinen zugriff hätten, weil sie hinter einer Firewall stehen oder der Datenverkehr anderweitig gefiltert wird.
openSSH kann nicht nur beliebige TCP-Verbindungen weiterleiten, sondern ein komplettes VPN aufbauen, in dem alle Datenverbindungen, egal ob TCP, UDP oder ICMP über die verschlüsselte *SSH*-Verbindung weitergeleitet werden.
Der Nachteil hierbei ist jedoch, das es, im Gegensatz zum SSH-Port-Forwarding, nur durch den *root*-Nutzer eingerichtet werden kann. Den Tunnel verwenden kann jeder Nutzer/jedes Programm, konfigurieren muss dies jedoch der Administrator. Normale Port-Forwardings hingegen kann jeder Nutzer für sich selber nach Bedarf einrichten.
Nützliche Links:

- https://www.ssh.com/ssh/tunneling/example
- https://blog.trackets.com/2014/05/17/ssh-tunnel-local-and-remote-port-forwarding-expl
  html?utm_source=cronweekly.com
- https://marius.bloggt-in-braunschweig.de/2016/01/02/vds-schnell-ein-vpn-aufsetzen/
- https://marius.bloggt-in-braunschweig.de/2016/04/12/ssh-vpn-mit-den-iproute2-tools/
- https://debian-administration.org/article/539/Setting_up_a_Layer_3_tunneling_
  VPN_with_using_OpenSSH

Hinweis: Sie können die oben genannten Links nutzen!

- Recherchieren Sie was „Tunneling“ im Sinne von *SSH* bedeutet.
- Recherchieren Sie was unter Port-Forwarding verstanden wird.

a.) Welche Arten von Port-Forwarding gibt es bzw. welche können mit SSH realisiert werden? Für welche Einsatzszenarien kann welches Forwarding genutzt werden?

[3] https://de.wikipedia.org/wiki/Tunnel_(Rechnernetz)

b.) Verdeutlichen Sie sich jeweils anhand eines Beispiels wie Forwarding genutzt werden kann.

c.) Finden Sie heraus wie Port-Forwarding unter Linux und SSH funktioniert.
Sie sollten schauen, was die Vorbedingungen sind, und welche Kommandos für das Forwarding notwendig sind.
Notieren Sie sich entsprechende Kommandos, sowie deren Bedeutung!

Aufgabe D – TLS

Ein Großteil der von Ihnen genutzten Verbindungen nutzen *TLS* (früher *SSL*), sodass Ihr Datenverkehr absichert zwischen den Sockets fließen kann. In der vorigen Übung haben Sie schon einige male *TLS* genutzt, nun sollen Sie dazu ein wenig mehr in Erfahrung bringen.

1.) Der englische Wikipedia-Artikel ist ein guter Einstieg in das *TLS*-Protokoll.
https://en.wikipedia.org/wiki/Transport_Layer_Security

a.) Für welche Zwecke kann *TLS* genutzt werden?

b.) Auf welcher Schicht des OSI-Modells würden Sie *TLS* einordnen und welche Schicht soll dieses Protokoll absichern?

2.) Beschreiben Sie kurz den wesentlichen Aufbau von *TLS*.

3.) Vollziehen Sie alles Schritte einer *TLS*-Session nach!
Hinweis: unter https://tls.ulfheim.net/ gibt es eine schöne Illustration.

Aufgabe E – VPN via Wireguard

Virtual Private Networks (VPN) sind in sich geschlossene Kommunikationsnetze, die wie der Name schon sagt, virtuell sind. Sinn eines solches Netzwerkes ist es physisch nicht lokale Teilnehmer in die eigene Netzwerkinfrastruktur aufzunehmen. Somit ist es den VPN Teilnehmern möglich die Infrastruktur trotz entfernten Zugriff zu nutzen.

- https://emanuelduss.ch/2018/09/wireguard-vpn-road-warrior-setup/
- https://www.linux-magazin.de/ausgaben/2018/01/wireguard/
- https://www.linux.org/threads/how-to-create-a-vpn-tunnel-with-wireguard.21496/

1.) Unter den oben gelisteten Links befinden sich einige Tutorials für den Einstieg in Wireguard.

a.) Der Installationsschritt kann ausgelassen werden, da Wireguard bereits vorinstalliert ist.

b.) Recherchieren Sie, wie die privaten und öffentlichen Schlüssel zu generieren sind und welche Besonderheiten zu beachten sind.

c.) Was wird unter einen virtuellen Interface verstanden? Warum muss ein solches für das VPN konfiguriert werden?

d.) Darauf aufbauend: Wie müssen die virtuellen Interfaces konfiguriert werden? Notieren Sie sich alle notwendigen Schritte!

e.) Recherchieren Sie wie eine *Wireguard*-Konfiguration automatisiert werden könnte.