Übungsblatt 4 – Wireshark

Aufgabe A – Wireshark

Nachdem Sie sich in den letzten Übungen vor allem mit dem „handwerklichen“ Aufsetzen von Netzwerken beschäftigt haben, schauen Sie sich in der kommenden Übung genauer an, was im Netzwerkstack vor sich geht.
Sie analysieren was im Inneren eines Netzwerkes passiert, wie die Realisierung der Protokolle aussieht und ob sich die theoretischen Ideen aus der Vorlesung auch in der Praxis wiederfinden lassen.
Um all dies in Angriff nehmen zu können, nutzen wir den Netzwerk-Sniffer *Wireshark*. Wireshark ist eine Open-Source-Software mithilfe dessen Analysen, Fehlerbehebungen, Software- und Protokollkommunikation untersucht werden können. *Wireshark* ähnelt *tcpdump* in gewisser Weise, jedoch bietet *Wireshark* ein grafisches Frontend (GUI), sodass die Analysen visuell ansprechend dargestellt werden können.
Hilfreiche Links:

- https://en.wikipedia.org/wiki/Wireshark
- https://www.lifewire.com/wireshark-tutorial-4143298
- https://www.wireshark.org/download/docs/user-guide.pdf
- https://wiki.wireshark.org/
- https://wiki.wireshark.org/CaptureFilters
- https://wiki.wireshark.org/DisplayFilters

1.) Rekapitulieren Sie Ihr Wissen zum OSI-Modell.

2.) Erläutern Sie was in Netzwerken unter Datenkapselung verstanden wird.

3.) Lesen Sie folgendes Tutorial in Hinblick auf die Fragen in Aufgabe 4.): https://tinyurl.com/yby2kukf [1]

4.) Nachdem Sie die Tutorials abgearbeitet haben:

a.) Was ist ein *Network-Sniffer*?

b.) Wozu kann ein Netzwerk-Sniffer genutzt werden?

c.) Verschaffen Sie sich einen Überblick, sodass Sie einen Überblick haben wo was zu finden ist, bzw. wo Sie Hilfe finden können.

[1] Lohnenswert ist das Wireshark 101 Buch im PDF Format.

d.) Recherchieren Sie wozu Filter in *Wireshark* eingesetzt werden.

e.) Bringen Sie in Erfahrung wie Filter genutzt werden.

f.) Welche beiden unterschiedlichen Mitschnitt-Modi (Caputre Modes) bietet *Wireshark*? Worin unterscheiden sich diese?

5.) Erläutern Sie anhand von Beispielen den grundlegende Umgang mit *Wireshark*.

a.) Wie stellen Sie Netzwerkinterfaces ein – auf welchem Interface soll der Mitschnitt laufen.

b.) Wie filtern Sie nach Protokollen?

c.) Wie filtern Sie *MAC*-Adressen?

d.) Wie filtern Sie *IP*-Adressen?

6.) **Fakultativ:** Wenn Sie mögen, können Sie *Wireshark* auf Ihre(n) Gerät(en) installieren oder in der virtuellen Maschine laufen lassen und ihren Netzwerkverkehr ein wenig analysieren.

- https://www.wireshark.org/download.html
- https://www.wireshark.org/docs/wsug_html_chunked/ChapterBuildInstall.html

Aufgabe B – Address Resolution Protocol (ARP) & Neighbor Discovery Protocol (NDP)

Es sollte Ihnen aufgefallen sein, dass in der zweiten Übung (Switched LAN) Ihr Netzwerk in der Planung zwar IP-Adressen nutzt, aber kein Router Verwendung fand. Der verwendete Switch ist ein `OSI-Layer 2` Gerät und kommt ohne IP-Adressen zurecht. Ihre Raspberry Pis verlangen jedoch zwingend eine IP-Adresse von Ihnen.
Um den Knoten ein wenig zu lösen, schauen Sie sich das *Address Resolution Protocol (ARP)* an.

1.) Recherchieren Sie mithilfe folgenden links: https://en.wikipedia.org/wiki/Address_Resolution_Protocol, was *ARP* ist und wie dies funktioniert.

2.) Wie adressiert ein Switch die Pakte zwischen den Endknoten (also den Raspberry Pis)?

3.) Erläutern Sie das Adressschema von *MAC*-Adressen. Kann dieses Adressschema auch zu Problemen führen?

4.) Da unser Uplink (Gateway des Labors) „nur“ das alte *IPv4* spricht ist *ARP* notwendig. Unter *IPv6* gibt es kein *ARP*, wie wird dies dort gehandhabt?

5.) Erklären Sie wie die Adressauflösung mittels *NDP* aussieht? Welche Schritten sind hier notwendig?

6.) Recherchieren Sie wie die Werkzeuge *arp* und *ip neigh* in unixoiden Betriebssystemen genutzt werden können, sowie deren Syntax.

Aufgabe C – MITM & ARP-Cache

Aufgrund Ihres Wissen ahnen Sie schon, dass mit ein wenig List viele Netzwerke leicht manipulierbar sind. *ARP* besitzt keinerlei Mechanismen, um die Nutzer vor Angriffen zu schützen. Mit der folgende Aufgabe sollen Sie herausfinden, wie hoch der Aufwand für eine solche Manipulation ist.
Hinweis: Die hier vorgestellten Techniken sollen Ihnen ermöglichen Angriffszenarien zu verstehen. Nicht diese fremde Infrastrukturen anzuwenden. Die Skripte sollten Sie nur in den eigenen Umgebungen bzw. der entsprechenden Laborübung nutzen!

1.) Vergegenwärtigen Sie sich, was ein Cache ist und wozu dieser eingesetzt wird. Anschließend daran, recherchieren Sie was es mit dem ARP-Cache auf sich hat.

2.) Erläutern Sie wie der ARP-Cache-Mechanismus funktioniert.

3.) Recherchieren Sie was ein Man-In-The-Middle-Angriff (*MITM*) ist.

4.) Da der ARP-Cache keinerlei Validierungsmöglichkeiten hat, ist ein Manipulation des ARP-Caches möglich. Überlegen Sie sich zunächst, welche Schritte hierfür notwendig wären, wenn Sie als Angreifer den Cache eines anderen Systems verändern wollen.

- Welche Voraussetzungen müssen gegeben sein?
- Welche Informationen über das Angriffsziel benötigen Sie?
- Welche Schritten müssen für den Angriff erfolgen. Denken Sie zunächst abstrakt darüber nach.
- **Anmerkung:** Falls Sie die vorige Aufgabe nicht bewerkstelligen könne, lesen Sie ein Tutorial zu ARP-Cache-Poisoning.

5.) Im Moodle (sowie auf den Raspberry Pis) steht ein Angrifftool für das sogenannte ARP-Spoofing bzw. ARP-Cache-Poisoning bereit. Lesen dieses Skript und versuchen Sie diese weitestgehend zu verstehen.
Notieren Sie sich den Ablauf! Stellen Sie Fragen zu den Stellen im Quellcode, die Sie nicht verstehen.

Aufgabe D – Ethernet & UDP|TCP/IP

Da Sie in der kommenden Übung mithilfe *Wiresharks* Ihren Netzwerkverkehr untersuchen sollen, müssen Sie zumindest grundlegend verstanden haben, auf welche Protokolle Sie dort stoßen werden. Natürlich können wir uns nicht alle Protokolle en détail anschauen, die wichtigsten, zumindest für die kommende Übung, sollten Sie jedoch kennen.

1.) Recherchieren Sie die Funktion, sowie den Aufbau von Ethernet (*IEEE 802.3* Protokollfamilie).

- **a.)** Auf welcher Ebene im OSI-Modell arbeitet *Ethernet*?
- **b.)** Welche Aufgabe übernimmt das oben genannte Protokoll?
- **c.)** Aus welchen Segmenten besteht ein Ethernet-Frame?
- **d.)** Zeigen Sie beispielhaft den Aufbau eine Ethernet-Frames.

2.) Recherchieren Sie die Funktion, sowie den Aufbau des IP-Protokolls (*IPv4*).

- **a.)** Auf welcher Ebene im OSI-Modell arbeitet IP?
- **b.)** Welche Aufgabe übernimmt das oben genannte Protokoll?
- **c.)** Aus welchen Segmenten besteht ein IPv4-Paket?
- **d.)** Aus welchen Segmenten besteht ein IPv6-Paket?
- **e.)** Zeigen Sie beispielhaft den Aufbau eines IPv4-Pakets.
- **f.)** Zeigen Sie beispielhaft den Aufbau eines IPv6-Pakets.

3.) Recherchieren Sie die Funktion, sowie den Aufbau des *TCP*-Protokolls.

- **a.)** Auf welcher Ebene im OSI-Modell arbeitet *TCP*?
- **b.)** Welche Aufgabe übernimmt das oben genannte Protokoll?
- **c.)** Aus welchen Segmenten besteht ein *TCP*-Datagram?
- **d.)** Zeigen Sie beispielhaft den Aufbau eines *TCP*-Datagram.

4.) Recherchieren Sie die Funktion, sowie den Aufbau des *UDP*-Protokolls.

- **a.)** Auf welcher Ebene im OSI-Modell arbeitet *UDP*?
- **b.)** Welche Aufgabe übernimmt das oben genannte Protokoll?
- **c.)** Aus welchen Segmenten besteht ein *UDP*-Datagram?
- **d.)** Zeigen Sie beispielhaft den Aufbau eines *UDP*-Datagram.

5.) Worin unterscheiden sich *TCP* und *UDP* grundlegend?