1 Analyse des TCP-Verkehrs

Anhand der gegebenen Wireshark-Daten analysieren wir die TCP-Kommunikation zwischen den IP-Adressen 192.168.0.84 und 44.195.140.190.

1.1 Verbindungsaufbau (Drei-Wege-Handshake)

- **Paket 2:**
  - **Source:** 192.168.0.84
  - **Destination:** 44.195.140.190
  - **Details:** 59035 → 443 [SYN]
  - Dieses Paket initiiert die TCP-Verbindung. Der Client (192.168.0.84) sendet ein SYN-Paket an den Server (44.195.140.190) auf Port 443.
- **Paket 3:**
  - **Source:** 44.195.140.190
  - **Destination:** 192.168.0.84
  - **Details:** 443 → 59035 [SYN, ACK]
  - Der Server antwortet mit einem SYN-ACK-Paket, das den Empfang des SYN-Pakets bestätigt und selbst ein SYN sendet.
- **Paket 4:**
  - **Source:** 192.168.0.84
  - **Destination:** 44.195.140.190
  - **Details:** 59035 → 443 [ACK]
  - Der Client bestätigt den Empfang des SYN-ACK-Pakets mit einem ACK-Paket. Die Verbindung ist nun hergestellt.

1.2 Datenübertragung

- **Paket 5:**
  - **Source:** 192.168.0.84
  - **Destination:** 44.195.140.190
  - **Details:** TLSv1.2 Client Hello (SNI=ase.autodesk.com)
  - Nachdem die Verbindung hergestellt ist, sendet der Client eine TLS Client Hello-Nachricht, um eine sichere Verbindung aufzubauen.
- **Pakete 13 bis 20:**
  - Mehrere Pakete, die TLS-Daten und Handshake-Nachrichten übertragen. Diese beinhalten die Client Key Exchange, Change Cipher Spec und Encrypted Handshake Messages.

- **Paket 21:**
  - **Source:** 44.195.140.190
  - **Destination:** 192.168.0.84
  - **Details:** 443 → 59035 [ACK]
  - Der Server bestätigt den Empfang eines früheren Pakets mit einem ACK-Paket.
- **Pakete 22 bis 26:**
  - Mehrere Pakete, die verschlüsselte TLS-Anwendungsdaten übertragen. Diese beinhalten Change Cipher Spec und Encrypted Handshake Messages sowie Anwendungsdaten.

1.3 Verbindungsabbau

In den gegebenen Paketen ist der Verbindungsabbau nicht vollständig zu erkennen, aber wir können einige FIN-ACK-Pakete beobachten, die darauf hindeuten, dass Verbindungen geschlossen werden:

- **Paket 7:**
  - **Source:** 2a01:b740:a30:f000::207
  - **Destination:** 2a02:810b:48c0:2104:7594:474d:3520:5c51
  - **Details:** 443 → 51393 [FIN, ACK]
  - Der Server initiiert den Verbindungsabbau mit einem FIN-ACK-Paket.
- **Paket 11:**
  - **Source:** 2a02:810b:48c0:2104:7594:474d:3520:5c51
  - **Destination:** 2a01:b740:a30:f000::207
  - **Details:** 51393 → 443 [FIN, ACK]
  - Der Client bestätigt den Empfang und sendet seinerseits ein FIN-ACK-Paket, um die Verbindung zu schließen.

2 Zusammenfassung der TCP-Funktionalitäten in der PCAP-Datei

- **Verbindungsaufbau:** SYN, SYN-ACK, ACK (Pakete 2, 3, 4)
- **Datenübertragung:** Client Hello, TLS-Daten und Handshake-Nachrichten (Pakete 5, 13-20)
- **Verbindungsabbau:** FIN, ACK (Pakete 7, 11)

3 Analyse des UDP-Verkehrs

Da die UDP-Kommunikation in den bereitgestellten Paketen nicht enthalten ist, kann ich keine konkrete Analyse für den UDP-Verkehr durchführen. Falls Sie weitere UDP-Daten zur Verfügung stellen können, kann ich die Unterschiede zwischen TCP und UDP detaillierter darstellen.

4 Vergleich zwischen TCP und UDP

4.1 Gemeinsamkeiten

- Beide Protokolle dienen der Datenübertragung im Netzwerk.
- Beide verwenden Ports, um verschiedene Dienste auf den Endpunkten zu adressieren.

4.2 Unterschiede

- **Verbindungsorientierung:**
  - **TCP:** Verbindungsorientiert. Es wird ein Verbindungsaufbau (Drei-Wege-Handshake) durchgeführt, bevor Daten gesendet werden.
  - **UDP:** Verbindungslos. Daten werden ohne vorherigen Verbindungsaufbau gesendet.
- **Zuverlässigkeit:**
  - **TCP:** Bietet Zuverlässigkeit, indem es die Zustellung und Reihenfolge der Pakete sicherstellt und verlorene Pakete neu sendet.
  - **UDP:** Bietet keine Garantie für die Zustellung oder Reihenfolge der Pakete. Es gibt keine Mechanismen für erneutes Senden oder Fehlerkorrektur.
- **Overhead:**
  - **TCP:** Höherer Overhead durch Verbindungsaufbau, Zustellungsgarantie und Flusskontrolle.
  - **UDP:** Geringerer Overhead, da keine Verbindungsverwaltung und Fehlerkorrektur stattfinden.

Durch die detaillierte Analyse der TCP-Pakete in der PCAP-Datei wird deutlich, wie TCP Verbindungen aufbaut, Daten zuverlässig überträgt und die Verbindung wieder abbaut. UDP hingegen würde einfach Pakete senden, ohne diese zusätzlichen Schritte.

Rechnernetze$_u$3

Mirko Raber

May 2024

4.3 U4

10001111