Versuch 1: Network Sniffing + Portscanning

Inhalt:

Mit dem Begriff *Sniffing* (Deutsch: schnüffeln) wird in der Netzwerktechnik das Mitlesen von Daten in einem Netzwerk bezeichnet, wobei die mit gelesenen Daten in der Regel nicht für den *sniffenden* Rechner bestimmt sind. Sniffing wird zum Einen zur Fehlersuche in Netzwerken und Netzwerkbasierter Software verwendet, andererseits kann ein in ein Netzwerk eingedrungener Hacker durch Sniffing an für ihn interessante Daten wie z.B. Passwörter gelangen, wenn diese beim Transport über das Netzwerk nicht verschlüsselt werden.

In paketbasierten Netzwerken werden mit einem Sniffer in der Regel einzelne Pakete aufgezeichnet und analysiert. Dabei beinhaltet ein Sniffer in der Regel mindestens einen Protokollanalysator für das TCP/IP Protokoll, mit dem einzelne Pakete einer Verbindung analysiert werden können.

In diesem Versuch soll der Umgang mit dem Open Source Sniffer *Wireshark* ehemals *Ethereal* geübt werden. Dabei soll sowohl der Bereich Fehlersuche im Netzwerk abgedeckt werden, als auch durch Sniffen von Passwörtern über Klartextprotokolle (z.B. FTP und Telnet) die Problematik unverschlüsselter Verbindungen nochmals verdeutlicht werden.

Portscanning ist gewissermaßen das Gegenstück zum Sniffing. Hier wird nicht passiv nach Informationen gelauscht, sondern es wird ein Rechner aktiv auf vorhandene Netzwerkdienste geprüft. Da ein erfolgreicher Angriff gegen einen entfernten Rechner über eine eventuell vorhandene Schwachstelle in einem Netzwerkdienst erfolgen muss, stellt ein Portscan die erste Stufe eines Angriffs, nämlich die Zielaufklärung dar.

Die Technik eines Portscans hängt entscheidend davon ab, ob TCP oder UDP Dienste identifiziert werden sollen. Dies ist deshalb so, da TCP im Gegensatz zu UDP ein verbindungsorientiertes Protokoll ist. Beiden gemeinsam ist, dass ein Verbindungsversuch auf den zu testenden Ports erfolgt und das Ergebnis ausgewertet wird.

Vorwissen: Netzwerktechnologie Ethernet und TCP/IP, UDP und ICMP.
http://www.wireshark.org/
http://www.tcpdump.org/tcpdump_man.html
http://acs.lbl.gov/~jason/tcpdump_advanced_filters.txt
http://insecure.org/nmap/

Praktische Durchführung des Networksniffing und Portscannings

Zunächst sollen Sie sich mit der Bedienung von Wireshark vertraut machen. Nach dem starten sehen sie das Hauptfenster des Sniffers. Für uns ist vor allem das Menü *Capture* von Interesse, in dem die Parameter für das Sniffing gesetzt werden können und der Sniffer gestartet werden kann. Gehen sie auf den Unterpunkt O*ptions*. Nun sehen sie das Fenster mit den Sniffer-Einstellungen, diese sollten zum großen Teil selbsterklärend sein. Für uns wichtig ist es zunächst unter *Name Resolution* alle Rechnernamensauflösungen zu deaktivieren.

Frage: Was sind diese im Einzelnen und zu welchen Problemen kann es kommen, wenn sie aktiviert bleiben?

Überprüfen sie die Netzwerkkonfiguration ihres Rechners. Nun können sie durch Wahl eines Interfaces an ihrem Rechner (z.B. eth0) und Starten des Sniffers den von und zu ihrem Rechner gehenden Netzwerkverkehr mitlesen. Erzeugen sie Traffic! Erkunden sie das umliegende Netzwerk, verwenden sie ping, traceroute usw. Testen sie die auf den PC vorhandenen Netzwerkdienste (telnet, ssh, ftp, http).

Sie können parallel dazu die Zahl der gesnifften Pakete in dem *Capture* Fenster von Wireshark ablesen. Wenn sie genügend Pakete aufgefangen haben, können die das sniffen durch den *stop* Button beenden.

Sie sehen nun die aufgezeichneten Pakete im Analysefenster. Sie können die einzelnen Pakete mit der Maus anwählen und sehen die Protokollanalyse im darunter liegenden Fenster. Wenn sie auf die einzelnen Protokollfelder klicken, werden die dazugehörigen *Rohdaten* im untersten Teilfenster hervorgehoben.

Um nun auch Pakete, die nicht direkt von oder zu ihrem Rechner gehen mitsniffen zu können, wird nun das Netzwerk von Switches auf Hubs umgeschaltet. Es entsteht ein großes Netzwerksegment. Um weiterhin effektiv arbeiten zu können, müssen sie nun die mitzusniffenden Pakete eingrenzen.

Gehen sie erneut auf *Capture* um einen neuen Sniffinglauf zu starten. Ein wichtiges Instrument zur Eingrenzung der zu sniffenden Pakete ist der *Capture Filter*, den sie im Parameterfenster für die Sniffing-Einstellungen sehen. Hier können sie einen Filter durch Verknüpfungen (Schlüsselwörter *and*, *or*, *not*) von den einzelnen Paketauswahlkriterien erstellen. Experimentieren sie mit den Paketauswahlkriterien.

Die Portscans werden mit dem Tool *nmap* durchgeführt. Zu diesem Scanner gibt es eine detailierte Manpage. Lesen sie die Manpage und machen sich sich mit der Benutzung von nmap und dessen Optionen vertraut.

Aufgaben:

a.) Analyse von typischem Netzwerkverkehr

Bauen Sie eine TCP Verbindung zu einem Dienst (z.B. http(s), ftp, smb, ssh, telnet) ihrer Wahl auf und beschreiben sie den Ablauf des Verbindungsaufbau/Verbindungsabbau (Sequenzdiagramm).

Was passiert bei einem Verbindungsversuch auf einem geschlossenen TCP Port (Sequenzdiagramm), z.B. telnet auf Port 161 (Befehl: telnet <ip> 161).

Sniffen Sie den DNS Traffic aus dem Netzwerkverkehr heraus, was beobachten Sie?

Filtern Sie den nicht IP basierten Traffic heraus. Was bleibt nun noch übrig?

b.) Sniffing von Passwörtern

Verbinden Sie sich mit anderen Telnet und FTP Servern, nutzen Sie den PC Ihres Nachbarn!

Versuchen Sie Passwörter und Benutzernamen anderer Benutzer, mitzusniffen. Filtern Sie gezielt die entsprechenden Pakete heraus.

Beschreiben Sie den Unterschied zwischen dem FTP und Telnet Protokoll bezüglich der Passwort Übermittlung.

c.) Portscanning

Scannen Sie das gesamte Labor Netz um aktive Systeme zu erkennen. Nehmen Sie einen Rechner genauer unter die Lupe und führen Sie einen detaillierten Portscan sowohl für TCP als auch für UDP Dienste durch.

Beobachten Sie die unterschiedlichen Scan-Varianten (Syn, Fin, Xmas, etc.) in Wireshark und analysieren Sie diese.

Nachbereitung und Protokoll

Das Protokoll sollte neben den genauen Versuchsbeschreibungen mit Netzwerkskizzen und Ergebnissen folgende Punkte umfassen:

- Grundlagen des Network Sniffing, Unterschied HUB und Switch
- Mitgelesene Daten mit Kommentar und Analyse
- Die jeweils verwendeten Filter mit Begründung
- Sequenzdiagramme für eine beispielhafte TCP/UDP- und FTP-/Telnet-Verbindung
- Grundsätzlicher Ablauf eines Protscans (UDP und TCP)