

Zusammenfassung - NASS

SK

19. Januar 2016

Korrektheit und Vollständigkeit der Informationen wird nicht gewährleistet.

Inhaltsverzeichnis

1	Introduction	2
2	Einführung und Rekapitulation	4
3	Paketfilter	7
4	Zustandsbehaftete Firewalls	8
5	Proxy Firewalls	9
6	Policies	9
7	Routers	12
8	Intrusion Detection Systems	12
9	Honeypots	12
10	Virtual Private Networks	12
11	Public Key Infrastructure	12
12	Host security	12
13	Security Design	12
14	Software	12
15	Assessment	12
16	E-Mail Security	12
17	Web Security	12
18	Prozesse für den Grundschutz	12
19	Trusted Architectures	12
20	Computer Forensics	12
21	Computer Forensics – Foundations	12
22	Computer Forensics – File System Forensics	12
23	Sicherheit von Gebäuden mit IT - Infrastruktur	12
24	Timers and timing	12
25	Angriffsbäume	12

26 Sicherheit in mobilen Netzwerken	12
27 Enterprise Authentication	12
28 Beispiele	12
29 Data driven security	12
30 Database Security	12
31 Human Resources Security	12
32 Securing Mail and HTTP protocols	12
33 Ethical Issues of Security Administration	12
34 Continuity Planning Disaster Recovery Planning	12
35 Advanced Crypto Algorithms	12
36 Cloud Security Challenges	12
37 The role of policies in computer security	12
38 Honeypots and tarpits in network security	12
39 Securing hosts and appliances	12
40 Organizational Aspects of Cyber Defense	12
41 Computer Forensics	12

1 Introduction

1.1 Taxonomie der Angreifer

- einzelner Angreifer
 - sozialer Hintergrund
 - öffentliche Aufmerksamkeit als Antrieb
 - evtl pol. Statements
 - geht gewöhnlich niedrige Risiken ein
- organisierte Kriminalität
 - Geld als Antrieb
 - mittlere Risiken
- Terroristen
 - politische oder gesellschaftliche Motivation
 - hohe Risiken
 - Zerstörung/Verwirrung als Ziel
- Konkurrenten
 - möglichst niedriges Risiko der Aufdeckung(abhängig vom wert der Information)
 - Informationsdiebstahl oder Zerstörung als Ziel
- Regierungsorganisationen
 - Industriespionage zum Wohl einheimischer Firmen
 - Militärspionage und hybride Kriegsführung

1.2 Angriffe gegen einen Computer

Informationsdiebstahl führt zu:

- Wettbewerbsvorteilen
- Verwirrung
- Erpressung

Zerstörung führt zu:

- Spaß und Selbstverherrlichung
- Politischen Stellungnahmen

Sammlung von Informationen

- Infos werden zu Angreifer gesendet
- an Netzwerk angeschlossene Rechner mit höherem Risiko
- Zugriff für Angreifer durch:
 - Social engineering
 - Viren/Trojaner/Würmer
 - Physischer Diebstahl von Datenträgern
 - Sniffing

Zerstörung von Infos

- Infos gehen verloren
- physische Angriffe/Feuer/Naturkatastrophen
- Beabsichtigte Löschungen durch
 - Social Engineering
 - Viren/Trojaner/Würmer

Viren

- Infektion von Dateien
- Infektion von System und Boot record
- Zerstörung, Verwirrung und öffentliche Aufmerksamkeit als Ziel

Würmer

- Mailing Worms - Verbreitung durch E-Mails
- Viren/Trojaner evtl als „Nutzlast“
- Network worms - Verbreitung durch Ausnutzung von Softwaremängeln(bspw Bufferoverflows)
- Ablauf:
 - Zielauswahl
 - ausnutzen(exploit)
 - Infektion
 - Verbreitung

Backdoors und Trojaner

- Schadsoftware wird in nützlicher Software versteckt
- mögliche Funktionen:
 - mitschneiden von Daten(logging)

- Zerstörung
- Installation weiterer Software (DoS Clients, root kits etc)
- bedingter Start von Prozessen (time bombs)

Identitäts Spoofing

- Angreifer übernimmt die Identität von jemand anderem
- Angreifer und Ziel müssen normalerweise ein Netzsegment teilen
- Angreifer liefert evtl falsche Infos über Routen oder Namen
- Grundsätzlich sind alle Antworten eines Protokolls potentielle Spoofingsubjekte (subject of spoofing?)

DoS

- Angreifer möchte einen Dienst der von einem Rechner oder Gerät angeboten wird überladen
- Angriffe gegen Konkurrenten, als pol/gesellschaftliche Aussage oder um andere Aktivitäten zu verbergen
- bösartige Anfragen sind nicht von normalen Anfragen zu unterscheiden
- BSP: HTTP, DNS DoS, SYN Flooding

Bot Network

- Fernsteuerung mehrerer Rechner um bösartige Aktionen auszuführen
- bsp: DDoS, aufwändige Entschlüsselungen berechnen

Password/Schlüssel Attacken

- Brute Force
- Raten/ Wörterbuchangriffe
- Mängel in der Implementation (z.B. Password als Klartext gespeichert)

Port/Network Scanning

- während der Aufklärungsphase um Sicherheitslücken und geeignete Ziele zu finden
- Angreifer möchte Informationen über das System erlangen
- Sniffing/Mapping/Port Scans

Session Hijacking

- Angreifer bricht in eine bestehende Session ein ohne sich einloggen zu müssen.

ZSF: viele unterschiedliche Angriffsmöglichkeiten \Rightarrow unüberschaubare Anzahl an verschiedenen Attacken
 Angreifer unterscheiden sich in Motivation und Möglichkeiten.

2 Einführung und Rekapitulation

2.1 Sicherheitskomponenten

- Firewalls
- Intrusion Detection/Prevention Systems
- Proxies
- interne oder private Netzwerke / Netzwerkzonen / Entmilitarisierte Zonen
- VPNs

2.2 Firewalls

entscheidet ob Verkehr ins Netzwerk gelangen darf oder nicht Typen:

- Paketfilter
- Zustandsbehaftete Firewall
- Proxyfirewall

2.3 Intrusion Detection Systems

Identifizierung von Attacken / verdächtigem Verkehr

Hilfe beim Einrichten/ konfigurieren von Firewalls

Normalerweise transparent für Nutzer und Angreifer.

hauptsächlich 2 Arten:

- Mustererkennung
- Anomalieerkennung

2.4 Proxies

strikte Trennung von internen und externen Netz

Üblicherweise auf Application Layer. Verhindert dass bestimmte Informationen(Viren, Pornos, illegale Infos) in das interne Netz gesandt werden.

Verhindert, dass bestimmte Informationen nach außen gesendet werden.

Kombinationen mit anderen Systemen(Virenfilter/ Spamfilter / IDS...)

2.5 VPN

VPNs erschaffen einen gemeinsamen Addressraum.

VPNs schützen die Kommunikation über ungesicherte Netzwerke als würde sie in einem Netzwerk stattfinden.

Gegenseitige Authentifizierung der Kommunikationspartner.

VPNs bieten signifikante Einsparungen über dedizierte Verbindungen.

2.6 Zonen - DMZ

kleine Netzwerke, welche öffentlich erreichbare Dienste beinhalten (z.B. HTTP)

DMZ oft durch Firewalls etc geschützt.

DMZ befinden sich außerhalb des internen Netzes.

sind unsicherer als das interne Netz.

Abgeschirmte Teilnetze sind isolierte Netze innerhalb des internen Netzes

2.7 internes Netz

eingeschränkter Zugriff auf das externe Netz nur über gut bekannte Ports

Internes Angriffsrisiko hängt ab von:

- Anzahl der Nutzer
- Vertrauen in die Nutzer
- Zugriffswege der Nutzer(Notebooks?)
- Fähigkeiten der Nutzer

Hosts müssen trotzdem noch mit firewalls etc geschützt werden

2.8 Basis Kryptografie

Kerkhoff's Prinzip: Sicherheit hängt nur von Schlüssel ab und nicht von der Kenntnis der kryptografischen Funktion.

2.8.1 symmetrische Verschlüsselung

Beide Teilnehmer benutzen zum ver- und entschlüsseln denselben Schlüssel.

Stromchiffren: Klartext wird Zeichen für Zeichen ver- und entschlüsselt.

Blockchiffren: arbeitet mit festen Blockgrößen und entschlüsselt mehrere Zeichen in einem Schritt.

One-Time Pads Stromchiffre deren Schlüsselstrom ein Strom aus echten Zufallsbits ist

Uneingeschränkt sicher(einziges bisher „bewiesen“ sicheres Verfahren).

Schlüssel muss zu verschlüsseln mindestens so lang sein wie der Klartext.

Jeder Schlüssel darf nur einmal verwendet werden.

Nachteil: viel Speicherbedarf für Schlüssel.

2.8.2 asymmetrische Verschlüsselung

Sender und Empfänger nutzen jeweils unterschiedliche Schlüssel.

Es ist schwierig den Entschlüsselungsschlüssel(k') aus dem Verschlüsselungsschlüssel(k) zu berechnen

k kann öffentlich gemacht werden (public-key-Verschlüsselung).

Nachteil: Verteilung der Schlüssel

2.8.3 hybride Verschlüsselung

Kombination aus symmetrischer und asymmetrischer Verschlüsselung.

symmetrischer Session Key mit dem die Daten symmetrisch verschlüsselt werden.

Session Key wird asymmetrisch mit public Key des Empfängers verschlüsselt.

löst Verteilungsproblem der asymmetrischen und behält Geschwindigkeit der symmetrischen Verschlüsselung

2.8.4 kryptografische Hashfunktion

Anforderungen:

- einseitig: wenn Hashwert y gegeben ist, ist es rechnerisch unmöglich eine Nachricht x zu finden, sodass $h(x) = y$
- schwacher Kollisionswiderstand: bei gegebener Nachricht ist es rechnerisch unmöglich eine andere Nachricht mit gleichem Hashwert zu finden
- starker Kollisionswiderstand: Es ist rechnerisch unmöglich zwei Nachrichten mit gleichem Hashwert zu finden.

Hash und Signaturen Von Nachricht wird Hash gebildet. Dieser wird verschlüsselt und als Signatur an die Nachricht gehängt.

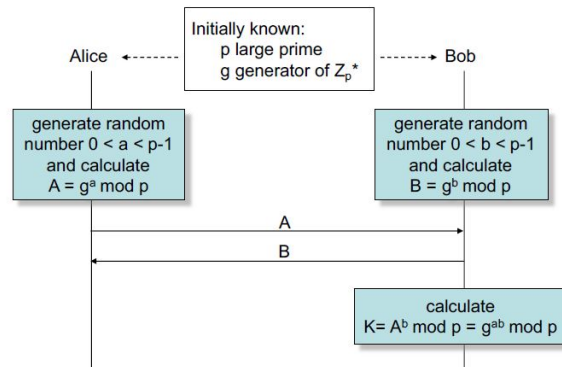
Empfänger entschlüsselt Signatur mit public key des Senders und vergleicht mit dem hash der Nachricht. Wenn gleich dann ist alles gut, wenn nicht dann wurde was verändert.

2.8.5 Diffie-Hellman Schlüsselaustausch

2.8.6 Zertifikate

Zertifikat ist eine Datenstruktur welche folgendes enthält:

- Öffentlichen Schlüssel
- Namen des Eigentümers des öff Schlüssels
- Namen des Ausstellers
- Ausstellungsdatum
- Ablaufdatum
- Möglicherweise andere Daten
- Signatur des Ausstellers



2.8.7 Certification Authorities (CA)

stellen Zertifikate aus.

sind normalerweise vertrauenswürdige Dritte.

Zertifikate werden über online Datenbanken verteilt (Certificate Directories) denen vertraut werden muss.

3 Paketfilter

3.1 Funktionsweise von Paketfiltern

Netzwerkpakete werden akzeptiert oder zurückgewiesen anhand von Parametern wie:

- Quelladresse/Ports
- Zieladresse/Ports
- Flags

3.2 Paketfilterregeln

Regeln können bzgl. Flags, Adressen und Ports angewandt werden.

Paketfilter können auch bezüglich des Inhaltes von Paketen angewandt werden.

Zulassende Regeln:

- explizites Erlauben von Zugriff
- sämtlicher anderer Verkehr wird verhindert

Verhindernde/ablehnende Regeln:

- bestimmter Verkehr wird explizit abgelehnt.
- sämtlicher anderer Verkehr wird für gewöhnlich zugelassen.

Wichtig:

- Reihenfolge der Regeln ist wichtig. (erst alles verhindern und dann einige zulassen ist was anderes als erst einige zulassen und dann alles zu verhindern!)
- große Anzahl an Regeln kann verwirrend sein.
- „alles verbieten und solange es nicht explizit benötigt wird“ kann gute Herangehensweise sein.

3.2.1 Ingress-Filter

Filtern ankommende Pakete

blockieren Zugriff von verdächtigen Quelladressen.

3.2.2 Egress-Filter

Filtern ausgehenden Verkehr.

Nur Pakete mit Quelladresse im Netzwerk dürfen das Netzwerk verlassen so lange keine andere Regel greift.

Quellen von abgewiesenen Paketen sind gute Kandidaten für Überprüfung.

3.2.3 Protokollfilter

Dienste haben festgelegte Protokolle

Daumenregel: nur Verkehr zu Diensten zulassen die wirklich benötigt werden.

3.2.4 Probleme

Zugriff für bestimmte Netze zulassen

Gefahr des Spoofings: Angreifer nutzt evtl falsche Quelladressen

Source Routing:

- Pakete enthalten evtl Infos über die Route zurück zum Urheber
- Überschreiben die Routingtabelle des Routers

Gefahr das Filterregeln umgangen werden

Fragmentierung:

- Paketfilter untersuchen Headerinfos
- Paket wird so aufgeteilt dass der Header geteilt wird und Adresse und Ports nicht gefiltert werden können.

Löcher:

- Dienste müssen erreichbar bleiben für externe Netzwerke
- entsprechende Ports müssen geöffnet werden.

3.3 dynamische Paketfilter

Filterregeln werden on the fly so erstellt wie sie benötigt werden und nach schließen der Verbindung wieder gelöscht.

Filter beobachten ausgehenden Verkehr und erstellen zurückwirkende Regeln.(Ausgehender Verkehr zu einer Adresse bewirkt Regel dass eingehender Verkehr von dieser Adresse erlaubt wird.)

Probleme:

- Regeln sind angreifbar z.B. durch senden falscher reset Pakete
- ausgehender Verkehr wird nicht gefiltert. Gefahr von Trojanern/Viren.

4 Zustandsbehaftete Firewalls

4.1 Funktionsweise

Kennen den Zustand von Verbindungen und wissen welche Pakete in welchem Zustand erwartet werden.

Es können Regeln angewandt werden die nur in bestimmten Zuständen wirksam sind.

Untersuchen hauptsächlich OSI 4 (transport layer), aber auch höhere Schichten.

4.2 Probleme

Hohe Leistung benötigt teilweise geclusterte Hardware. Zustandsbehaftete Firewalls lassen sich nicht einfach clustern.

Zustandslose Protokolle (UDP, ICMP, DNS, HTTP)

4.2.1 Zustandslose Protokolle

Zustandslose Protokolle definieren trotzdem welche Pakete erwartet werden.

Timeouts werden genutzt um Pseudo-Verbindungen zu erzeugen.

4.3 Multi-Layer Inspection

Die meisten Protokolle basieren auf Protokollen aus niedrigeren Layern. BSP: HTTP nutzt TCP Verbindungen. Zustandsbehaftete Firewalls können beide Layer beobachten.

5 Proxy Firewalls

6 Policies

6.1 Was sind Policies?

Eine Sicherheitsrichtlinie(security policy) beschreibt was getan werden muss um auf einem Rechner gespeicherte Infos zu schützen.

Richtlinie definiert was gemacht werden muss und und wie es ausgewertet werden kann.

Werden normalerweise aufgeschrieben.

6.2 Komplexität von Richtlinien

werden von Menschen definiert.

müssen verständlich formuliert sein.

zu komplexe (aber auch zu einfache. bsp „keine rechner benutzen!“) Richtlinien sind nicht durchsetzbar.

6.3 Entwicklung von Richtlinien

beste Vorgehensweise:

- Risiken identifizieren
 - Sicherheitsanalyse
 - * kritische Daten und Systeme identifizieren
 - * normale Nutzung des Netzwerkes feststellen
 - aufschreiben
- Funde kommunizieren
 - Dem Management berichten
 - * einfach
 - * ausgeglichen
 - * präzise
 - * Zeigen auf einzelne vermeiden
 - * Allgemein halten
- Richtlinie erstellen oder aktualisieren
 - aufschreiben
 - Genauheit und Klarheit
 - * Was muss getan werden?
 - * Warum?
 - * Wer ist verantwortlich?
 - Knappheit: Niemand liest mehr als 10 Seiten.
 - Realismus
- Einhaltung der Richtlinie kontrollieren
 - Wenn die Einhaltung einer Regel nicht kontrolliert werden kann ist sie nicht durchsetzbar
 - Stichproben, Log Analysen, Festplattendurchsuchungen
- Versuchen eine „Kultur“ zur Einhaltung der Richtlinie einzuführen
 - Anpassung an die Richtlinie hängt stark vom Verhalten der Nutzer ab
 - Mit Nutzern über Risiken sprechen
 - Richtlinien vor der Einführung erklären
 - Anweisungen und autoritäres Verhalten vermeiden

6.4 ungeschriebene Richtlinien

versteckte Regeln existieren.
nutzen des gesunden Menschenverstandes
Versuchen ein Sicherheitsbewusstsein im Betrieb zu etablieren.
Wissen verbreiten, aber vorsichtig und sensibel.

7 Intrusion Detection Systems (IDS)

Sind dazu bestimmt Angriffe zu erkennen und nicht um diese zu verhindern.
Netzwerk IDS Sensor liest Traffic mit uns analysiert diesen indem nach Zeichen für:

- Scans/Sonden
- Aufklärungsaktivitäten
- Exploits

Ist normalerweise komplett transparent (nicht zu entdecken).

7.1 Motivation

Ohne IDS würde ein Admin die meisten Angriffe nicht bemerken und kann auf diese somit nicht reagieren.
Fehlende Infos ohne IDS:

- Welche Hosts wurden angegriffen?
- Welche Daten wurden kompromittiert?
- Mit welcher Methode wurde angegriffen?

Nachfolgende Schritte eines Angriffes können verhindert werden.

7.2 Methoden- Anomalieerkennung

Statistische Analyse um „unnormalen“ Verkehr erkennen zu können.
Parameter wie Herkunft, Datenrate, Ports und Zeit werden berücksichtigt und gegen eine Statistik geprüft, jedoch nicht nach bestimmten Mustern.
Berechnung der Wahrscheinlichkeit dafür das Verkehr unnormale ist mithilfe von Bayesschen Filtern.
Training der Filter mit Verkehr der als normal betrachtet wird.
Wenn eine bestimmte Grenze überschritten wird, wird Alarm ausgelöst.

7.3 Methoden- Signaturerkennung

Analyse von Paketen anhand von gegebenen Mustern.
Adressen und timing Pattern werden berücksichtigt.
bestimmte Mustern lösen einen Alarm aus.

7.4 Probleme mit IDS

Fehlalarme (false positives) und nicht erkannte Alarme (false negatives).

- Viele Fehlalarme werden zu einer bedeutend höheren Ignoranz seitens des Admins führen.
- Reduzierung der false positives führt gewöhnlich zu mehr false negatives.
- IDS Evasion um false positives zu verringern
- Als ersten Filter ein allgemeines Muster Nutzer
- Spezifischere Untersuchung der Pakete die den ersten Filter durchlaufen haben.

7.5 Ansätze für IDS

globale vs lokale Ansätze:

- betrachtet die Auflösung der Referenzmenge bezüglich ob ein bestimmtes Datenobjekt als Außenseiter erkannt wurde.
- globale Ansätze:
 - Referenzmenge enthält alle Daten.
 - Basisannahme: es gibt nur einen normalen Mechanismus (normal mechanism?).
 - Problem: andere Ausreißer sind auch in der Menge und können das Ergebnis verfälschen.
- lokale Ansätze:
 - Die Referenz enthält lediglich eine Teilmenge der Daten.
 - keine Annahme über die Anzahl von normalen Mechanismen (normal mechanism?).
 - Problem: Wie wählt man eine geeignete Referenzmenge

Einige Ansätze befinden sich irgendwo dazwischen.

Die Auflösung der Referenzmenge kann automatisch oder durch eine Nutzereingabe verändert werden.

7.6 Statistische Tests

Idee:

- Gegebene Wahrscheinlichkeitsverteilung (z.B. Gauss)
- Parameter berechnen unter der Annahme, dass alle Datenpunkte durch eine solche Wahrscheinlichkeitsverteilung generiert wurden.
- Ausreißer sind die Punkte, die eine geringe Wahrscheinlichkeit haben von der Verteilung generiert worden zu sein (z.B. weicht mehr als 3 mal von der Standardabweichung ab).

Annahme:

- Normale Daten folgen einer Verteilung und treten in einer Region des Models mit hoher Wahrscheinlichkeit auf.
- Ausreißer weichen stark von dieser Verteilung ab.

Viele verschiedene Tests (unterschiedliche Verteilung, Menge der Variablen, Menge der Verteilungen....)

7.7 Tiefenbasierte Ansätze

Idee:

- Suche nach Ausreißern an den Grenzen des Datenraums aber unabhängig von Wahrscheinlichkeitsverteilungen.
- organisieren Daten in Schichten von konvexen Hüllen
- Ausreißer sind Objekte in äußeren Schichten.

Annahme:

- Ausreißer befinden sich an den Grenzen des Datenraums und normale Daten im Zentrum.

- 8 Honeypots and Tarpits
- 9 Public Key Infrastructure
- 10 Enterprise Authentication
- 11 Securing Hosts and Appliances
- 12 Organisational Aspects
- 13 Computer Forensics