

IDS

Intrusion Detection System

Inhalt

Allgemein

- Definition
- Arten und Bewertung
- Aufgaben
- Platzierung
- Umgehen eines IDS
- Limits
- False Positives
- IDES
- Produkte

Snort

- Einführung
- Detection Engine
- Plug-Ins
- Modi
- Architektur Version 1.x
- Architektur Version 2.x
- Tools

Activeworx Policy Manager

SnortSnarf

ACID

Allgemeines

Was ist ein *Intrusion Detection System* ? - eher nicht !



Was ist ein *Intrusion Detection System* ? - schon besser !



Was ist ein *Intrusion Detection System* ?

Intrusion Detection System \cong Eindringling Erkennungs-Automat

Die Beschreibung ist nicht korrekt. Eine bessere Bezeichnung wäre:
SCS \cong Signature Compare System \cong Signaturen Vergleichs-Automat

Die Tätigkeit eines IDS ist der

VERGLEICH

von auftretenden Ereignissen mit einem
vorher bestimmten Muster

Die Ergebnisse des Vergleiches sollten grundsätzlich nicht von einem Automaten ausgewertet werden!

Arten von IDS

Hauptarten:

- *Network IDS (NIDS)*
- *Host IDS (HIDS)*

Hybride Arten:

- *Per-Host Network IDS (PH-IDS)*
- *Load Balanced Network IDS (LB-NIDS)*
- *Firewall IDS (FW-IDS)*

NIDS – Network IDS

- Sammelt Datenpakete im promiscuous Mode
- Stichpunkte:
 - *Paket Sammelrate – wie hoch ist der maximale Throughput ?*
 - *Reassembly/defragmentation/spoofing – Unterstützung ?*
 - *Selektive Analyse – kann man bestimmte Ereignisse von dem Vergleichsmechanismus ausschließen ?*

HIDS – Host IDS

- Analysiert Log-Files und Prozesse auf Hosts
- Stichpunkte:
 - *CPU-Last auf dem Host ?*
 - *Wie wird mit netzwerkbasieren Angriffen vorgegangen ?*
 - *Welche Plattformen werden unterstützt ?*

PH-NIDS – Per-Host Network IDS

- NIDS wird im Protokoll-Stack eines jeden Hosts eingefügt
- Stichpunkte:
 - *Besitzt die Eigenschaften eines NIDS*
 - *Aber:*
 - *Nur der Traffic zum/vom Host wird beobachtet*
 - *Besitzt nicht die Leistung eines reinen NIDS*
 - *Aufwendige Überwachung – viele Logs*

LB-NIDS – Load-Balanced Network IDS

- Benutzt einen Loadbalancer, um den Traffic auf mehrere NIDS aufzuteilen
- Stichpunkte:
 - *Skalierbar auf eine nahezu unendlichen Bandbreite*
 - *Teurer als mehrere einzelne NIDS*

FW-IDS – Firewall IDS

- Platzierung einer NIDS-Funktionalität in einer Firewall
- Stichpunkte:
 - *Verringerung der Firewall-Performance*
 - *Probleme mit Stateful-Inspection*

Pros und Cons - Allgemein

- ✓ Kann die Sicherheit allgemein erhöhen
- ✓ Echtzeit-Ansicht der Aktivitäten
- ✗ Falsches Empfinden der Sicherheit
- ✗ Betrieb ist aufwendig und wird schnell vernachlässigt

Pros und Cons – Host IDS

- ✓ Jeder Host ist geschützt
- ✓ Weniger "false positives" als bei NIDS
- ✓ Leicht einzusetzen
- ✗ Einsatz nahezu unmöglich bei großen Web-Shops
- ✗ Kommerzielle Lösungen sind sehr teuer (z.B. Tripwire)
- ✗ Wenn root/admin Zugang geknackt – game over

Pros und Cons – Network IDS

- ✓ Leichter einzusetzen als HIDS
- ✓ Einfacher zu administrieren
- ✓ Freeware Lösungen sind verfügbar
- ✗ Mehr "false positives"
- ✗ Benötigt viele selbstgestrickte Tools, um effektiv zu sein
- ✗ Pattern-matching

Warum ein IDS ?

- Firewall ist „nur ein Rauschfilter“

Viele Angriffe können auch durch Firewalls angewandt werden, z.B. Microsoft Unicode Hacking:

```
/cgi-bin/..%c0%af..%c0%af../winnt/system32/ cmd .exe
```

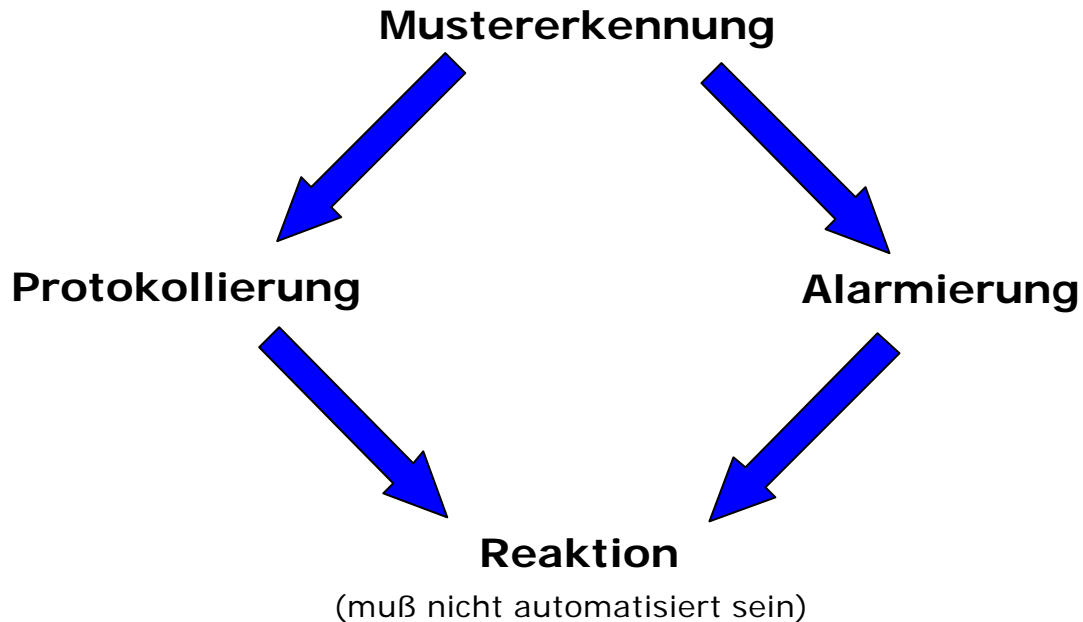
- Kontrolle der Firewall

Taucht ein Alarm sowohl vor als auch nach der Firewall auf, so kann es sich um ein Sicherheitsloch im Regelsatz der Firewall handeln.

- Mitarbeiterüberwachung (Betriebsrat befragen !)

Wird während der Arbeitszeit z.B. über das Netzwerk gespielt ?

Aufgaben eines IDS:



Mustererkennung

- anhand vorgefertigter oder selbst erstellter Regeln (Vergleich)
- anhand zeitlich markanter Ereignisse (Portscan)

Protokollierung

- Log-File,
- Syslog,
- SNMP,
- Datenbank,

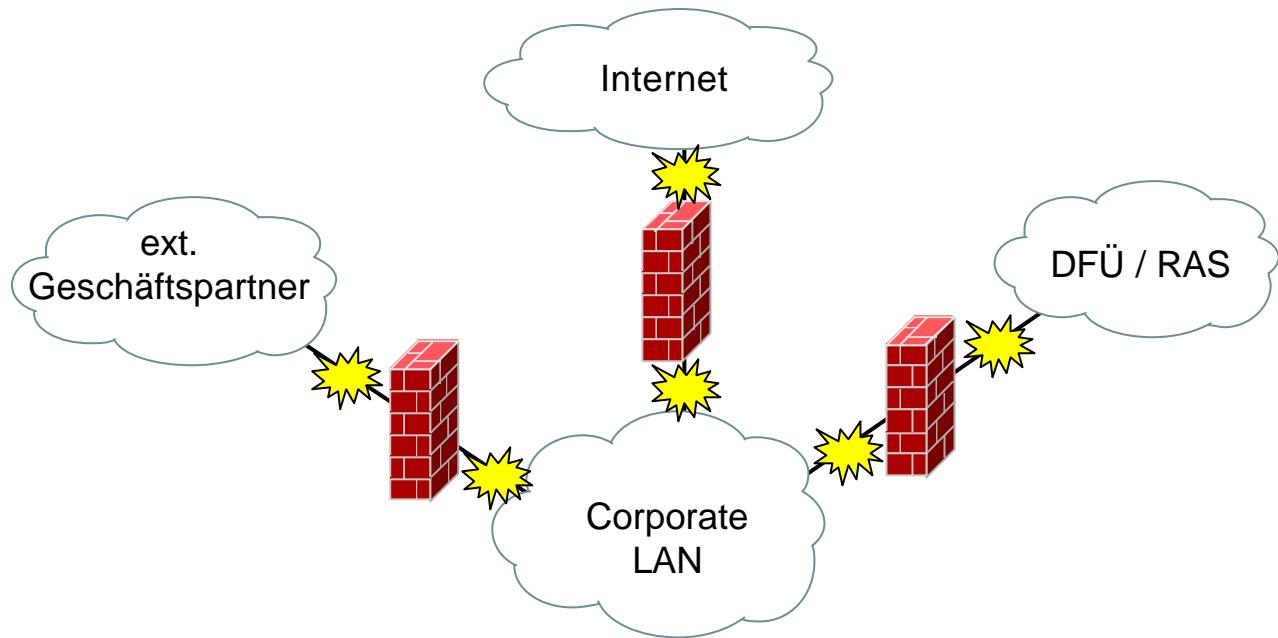
Alarmierung

- SNMP,
- WinPopUp,
- SMS,
- ...

Reaktion

- Regeländerung auf der Firewall (automatisiert/manuell)
- Benachrichtigung von Geschäftspartnern,
- Virenskan,
- Abmahnung von Usern,
- Abschaltung von Systemen,

Platzierung eines IDS



Warum Intern und Extern ?

- Viren, Würme, Trojaner,
(Versuchen sich in das Internet zu verbinden)
- Mitarbeiter,
(Die meisten Sicherheitsverstöße werden durch eigene Mitarbeiter verursacht)
- Kontrolle der Firewall,
(Taucht der gleiche Alarm vor und hinter der Firewall auf, so sollte der Regelsatz überprüft werden)



Umgehen von Intrusion Detection Systems

Möglichkeiten ein IDS zu umgehen:

- IDS Lösungen sind nicht perfekt
- Administratoren sind nicht perfekt
- Sicherheit ist ein Prozess !
 - keine Person
 - kein Produkt
 - IDS/FW sind ein Teil dieses Prozesses

Limits von **NIDS**

- Anzahl der Signaturen
- Qualität der Signaturen
- Performance
- Speicherplatz
- Know-How über die eingesetzten Protokolle
- Verschlüsselung
- TCP-Session Integrität

Limits von **HIDS**

- Management Overhead
- Einschränkungen für NIDS-Funktionalität
- Unter Umständen Kernel-Modifikationen notwendig
- Benötigt Speicherplatz und CPU-Last
- Einsatz nicht möglich bei schnell wechselndem Datenbestand

Techniken zur Umgehung eines IDS

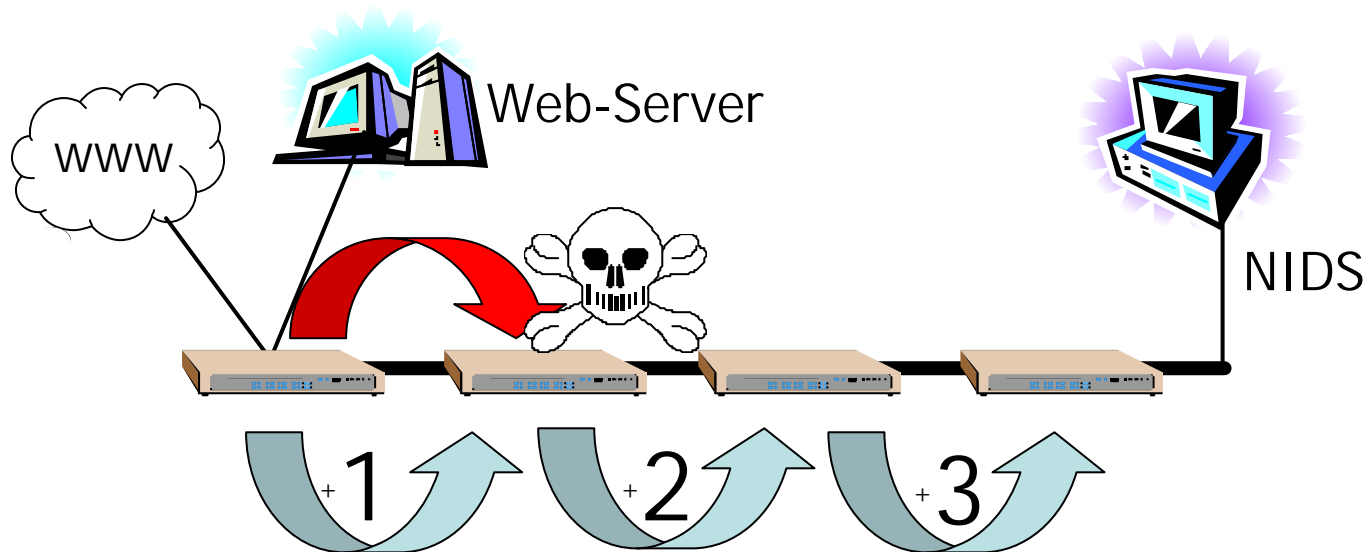
○ NIDS

- fragmentation,
- TCP un-sync,
- Low TTL,
- Max MTU,
- HTTP Protocol,
- Telnet Protocol,

○ HIDS

- Kernel Hacks,
- Library Hacks,
- HTTP Logging,
- Stack Protection,

Umgehen eines NIDS – Beispiel Low TTL



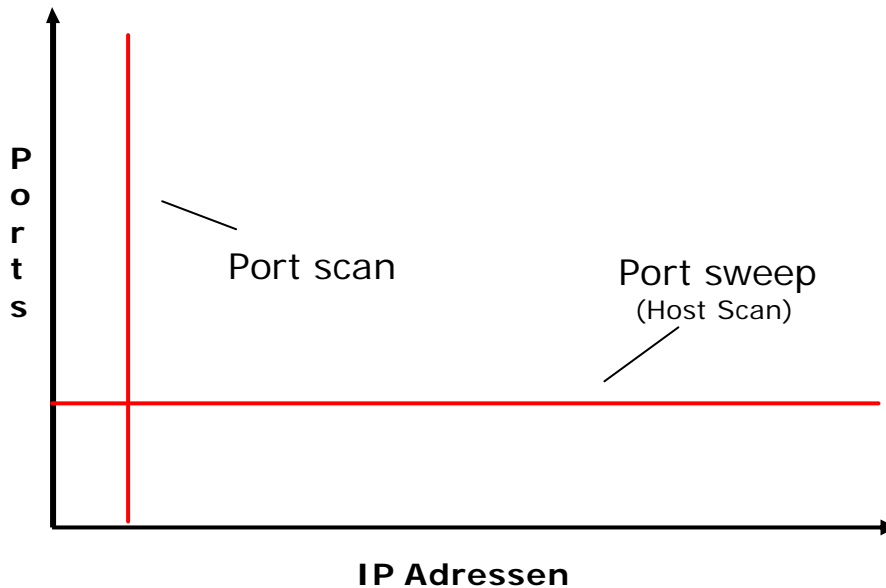
Ein IDS darf nicht zu viele Hops vom Brennpunkt entfernt sein !

Umgehen eines NIDS – Beispiel HTTP Protokol

- '/' padding: `"/cgi-bin///phf"`
- Self referencing directories: `"/cgi-bin/./phf"`
- URL Encoding: `"%2fcgi-bin/phf"`
- Reverse Traversal: `"/cgi-bin/here/../phf"`
- TAB anstatt SPACES
- DOS/Win Syntax: `"/cgi-bin\phf"`
- Null method: `"GET%00/cgi-bin/phf"`

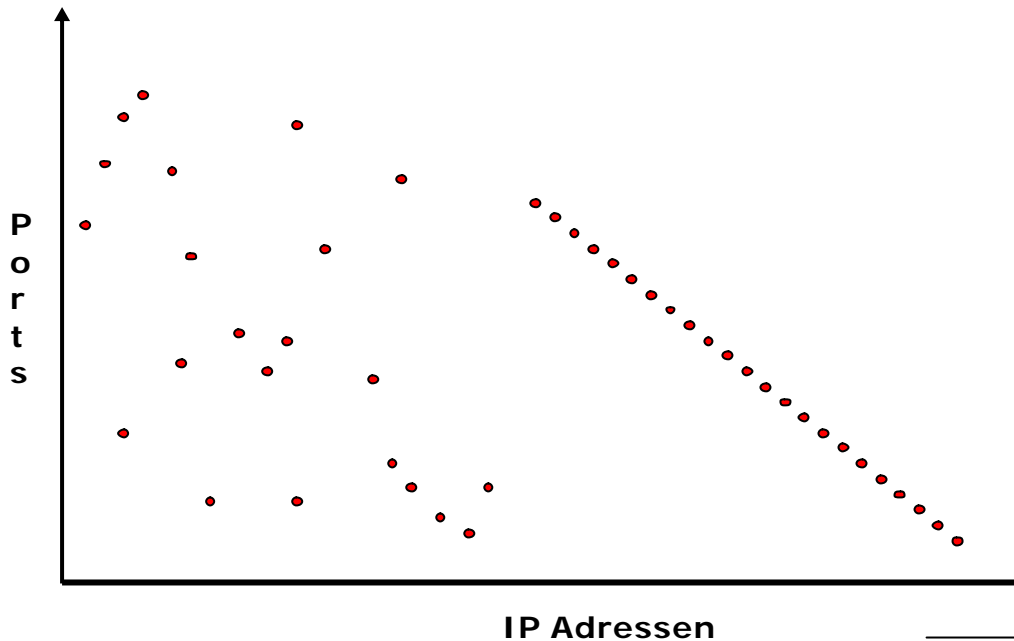
Umgehen eines NIDS – Beispiel Verteilte Scans

"Normaler" Scan (Wird von den meisten IDS erkannt)



Umgehen eines NIDS – Beispiel Verteilte Scans

Verteilter Scan (Wird von nahezu keinem IDS erkannt)



Umgehen eines NIDS – Beispiel TELNET Protokol

- Automatische Proxies, welche Zufallszeichen gefolgt von Backspaces einfügen

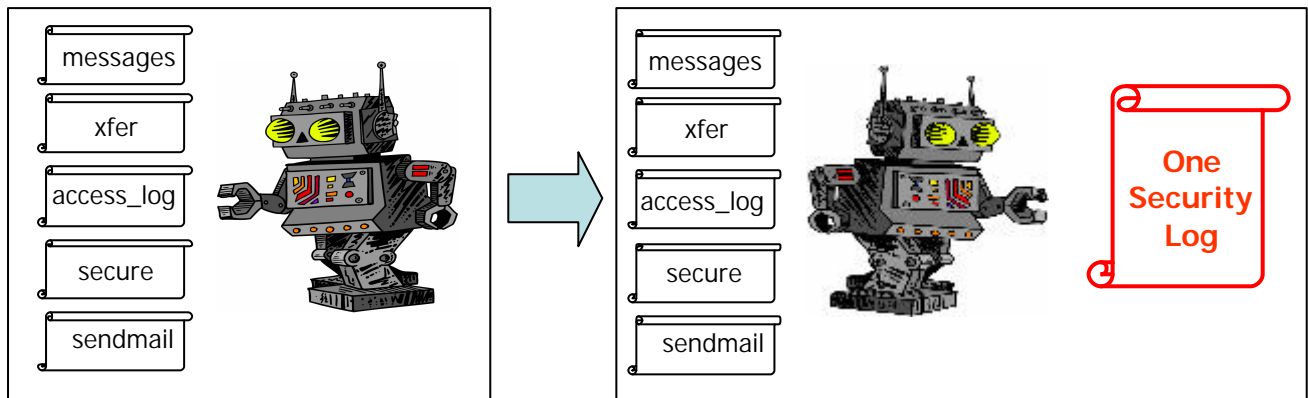
```
"su X{backspace}root"
```

Umgehen eines NIDS - Quellen

- Whisker – Rain Forest Puppy
<http://www.wiretrip.net/rfp/p/doc.asp?id=21&iface=2>
- Fragrouter - Dug Song
<http://www.anzen.com/research/nidsbench/>
- Congestant - horizon,
[Phrack 54](#)

Umgehen eines HIDS – Beispiel Logfile

Bei den meisten HIDS wird zu der Vielzahl der Logfiles noch ein weiteres hinzugefügt.



Die Aufmerksamkeit des Administrators liegt jetzt nur noch bei dem IDS-Logfile ! (Social Attack)

Umgehen eines HIDS – Beispiel Kernel/Library Hacks

- Für Windows NT gibt es einen 4 Byte Patch, der jegliche Security Mechanismen abschaltet
- Umgebungsvariablen, welche direkt auf ein Kernel-Modul zeigen
- Root-Kits

Umgehen eines HIDS – Quellen

- Stackguard - <http://www.immunix.org/documentation.html>
- Phrack 51 ("Shared Library Redirection Techniques")
- Phrack 52 ("Weakening the Linux Kernel")
- Phrack 55 ("A real NT Rootkit, patching the NT Kernel")
- Phrack 56 ("Backdooring Binary Objects")

False Positives

False Positives sind Meldungen, die nicht zu den *event of interests* gehören.

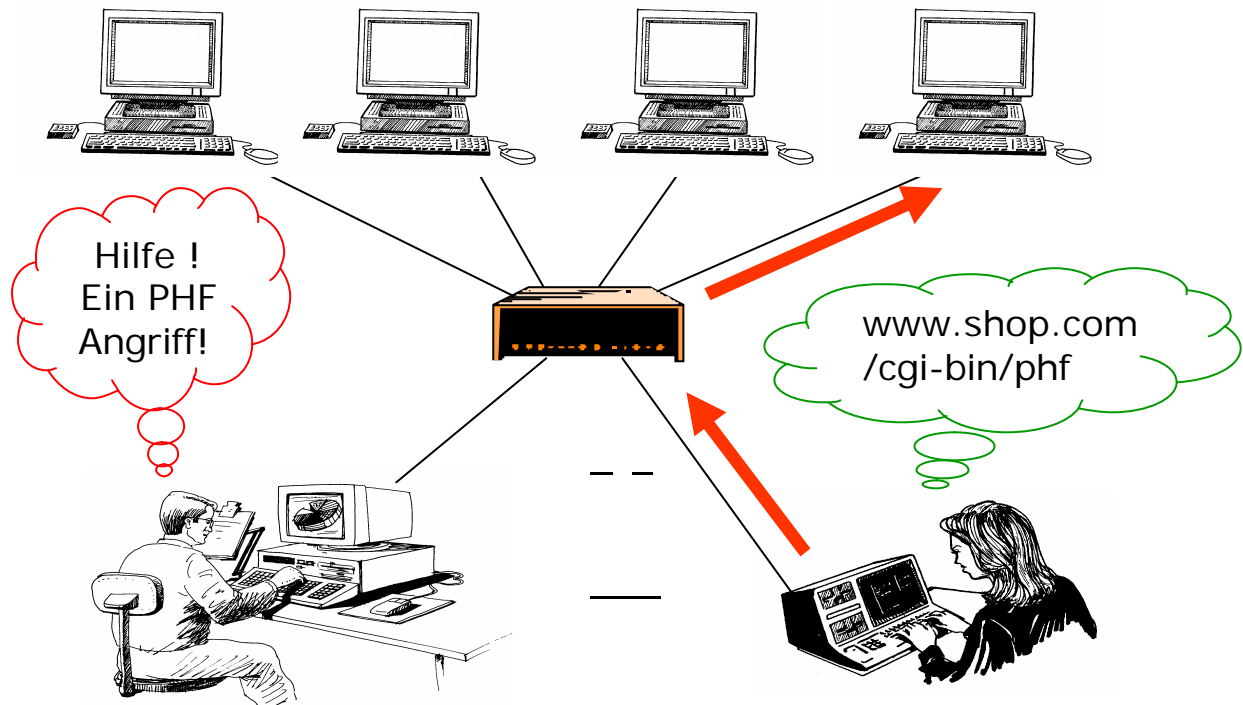
Sie entstehen bei:

- einer automatischen Auswertung der Log-Files,
- schlecht konfigurierter Policy,
- zu vielen Policies,

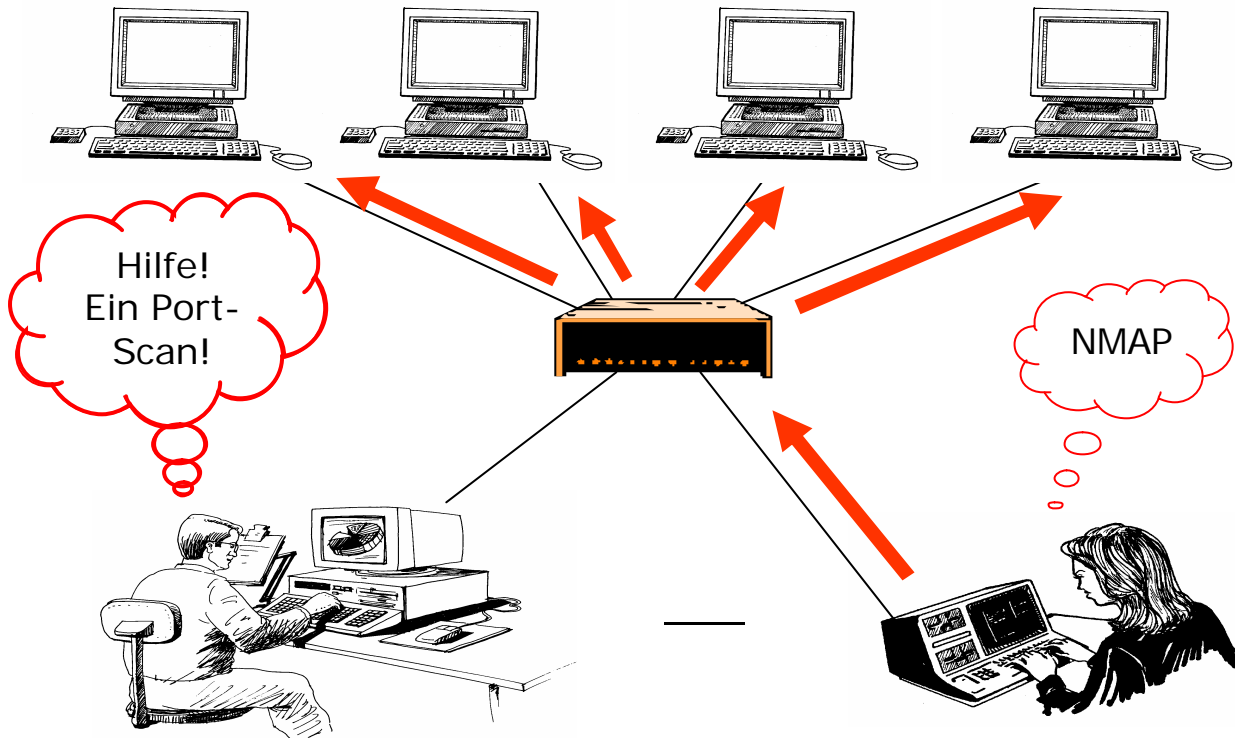
False Positives treten immer auf !

Nur ein Administrator/Analytiker kann false postivies ausmerzen !

False Positives ? – Beispiel 1



False Positives ? – Beispiel 2



IDES – An Intrusion Detection Model

Dorothy Denning veröffentlichte 1986:

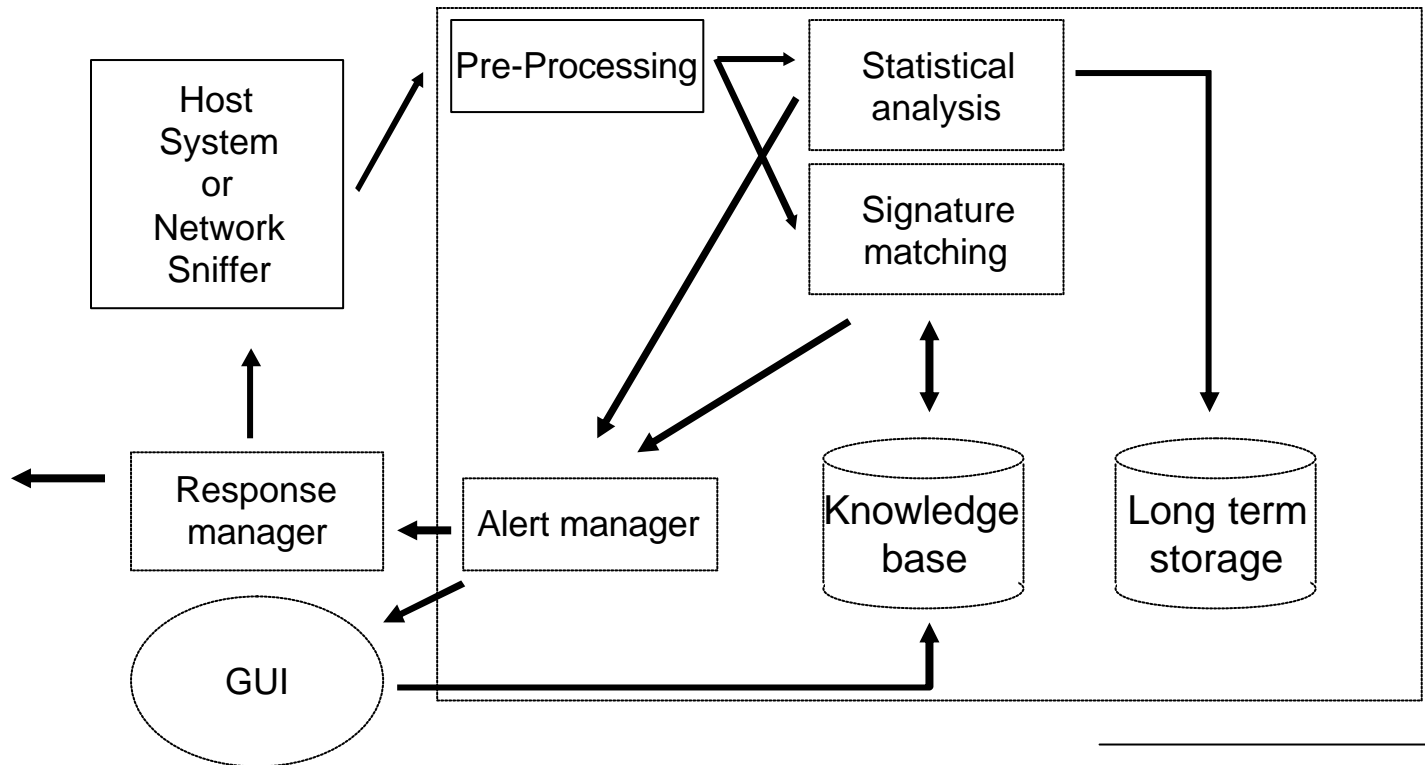
"An Intrusion Detection Model"

Welches (ansatzweise) die Grundlage der meisten IDS ist.

Darin sind folgende Bausteine eines IDS definiert:

- *Subjekte* – Initiator einer Aktivität
- *Objekte* – Ziele einer Aktivität
- *Audit Records* - Trace-Informationen
- *Alarm* – Informationen, die zur Handlung aufrufen

IDES – An Intrusion Detection Model



Produkte

Kommerzielle Tools

- ISS RealSecure
- Axent NetPowler
- Network Flight Recorder
- Cisco NetRanger
- GOTS
- EPIC2
- Network Intrusion Detector
- Shadow

Freie Tools

- **Snort**



Lightweight Intrusion Detection

Einführung in Snort (Metrics)

Snort ist:

- klein (*ca. 110K Source Code*)
- portable (*Linux, Solaris, *BSD, HP-UX, Windows*)
- konfigurierbar (*Leicht zu erlernende Sprache für Regeln, sehr viele reporting/logging Optionen*)
- kostenlos (*GPL/Open Source Software*)
- NIDS (eingeschränkt HIDS)
- Populär

Einführung in Snort

Snort Design:

- Paket sniffing network intrusion detection system
- Libpcap basierendes Interface
- Regelbasierende detection engine
- Plug-in System
- Mehrere Ausgabe Optionen
 - Decoded logs, tcpdump log,
 - Echtzeit Alarmierung in Syslog, SNMP und WinPopup

Detection Engine

- Regeln nach Signaturen,
- Modularer Aufbau der Detection Engine,
- Abnormale Aktivitäten werden erkannt
(stealth scan, OS fingerprints, invalid ICMP codes, ...)
- Regelsystem ist sehr flexibel und einfach anzupassen

Plug-Ins

- Präprozessor

Pakete können überprüft/manipuliert werden, bevor sie von der detection engine untersucht werden.

- Erkennung/Vergleich

Die Möglichkeit, einen einfachen Test an einem Paket durchzuführen (z.B. Flags)

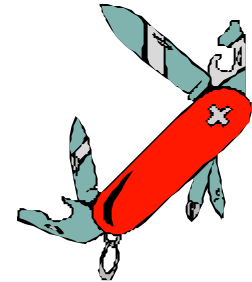
- Ausgabe

Viele Möglichkeiten zur Ausgabe an andere Tools/Plug-Ins



Snort Modi

- Drei Haupt-Modi
 - Sniffer Modus
 - Packet Logger Mode
 - NIDS Mode
- Optionale Modi können per Command-Line geändert werden



Standardmäßig startet Snort im NIDS-Modus

Sniffer Modus

- Eigenschaften wie tcpdump
- Dekodiert Pakete und gibt sie nach stdout aus
- Filterung durch gleichen Syntax wie bei tcpdump möglich



Sniffer Modus - Ausgabe

○ Snort Ausgabe

```

=====
13/07-11:12:02.954779 192.168.1.6:1032 -> 172.17.1.8:23
TCP TTL:128 TOS:0x0 ID:31237 IpLen:20 DgmLen:59 DF
***AP*** Seq: 0x16B6DA Ack: 0x1AF156C2 Win: 0x2217 TcpLen: 20
FF FC 23 FF FC 27 FF FC 24 FF FA 18 00 41 4E 53 ..#..'..$.ANS
49 FF F0 I..
=====

```

○ Zum Vergleich: tcpdump Ausgabe

```

11:16:35.648944 172.17.1.8.23 > 192.168.1.6.1033: P 16:34(18) ack 16
                               win 8760 (DF) (ttl 255, id 49913)
                               4500 003a c2f9 4000 ff06 a2b4 0a01 0108
                               0a01 0106 0017 0409 1cf9 e7f6 001a e050
                               5018 2238 31c6 0000 fffe 1fff fe23 fffe
                               27ff fe24 fffa

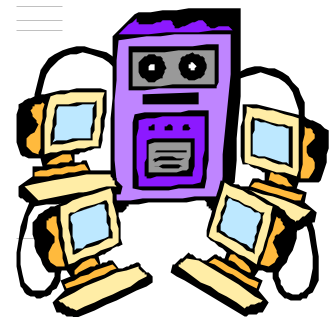
```

Packet Logger Modus

- Speicherung der kompletten Pakete
- Multi-Mode Logging Optionen
 - ASCII,
 - tcpdump,
 - XML,
 - MySQL,

NIDS Modus

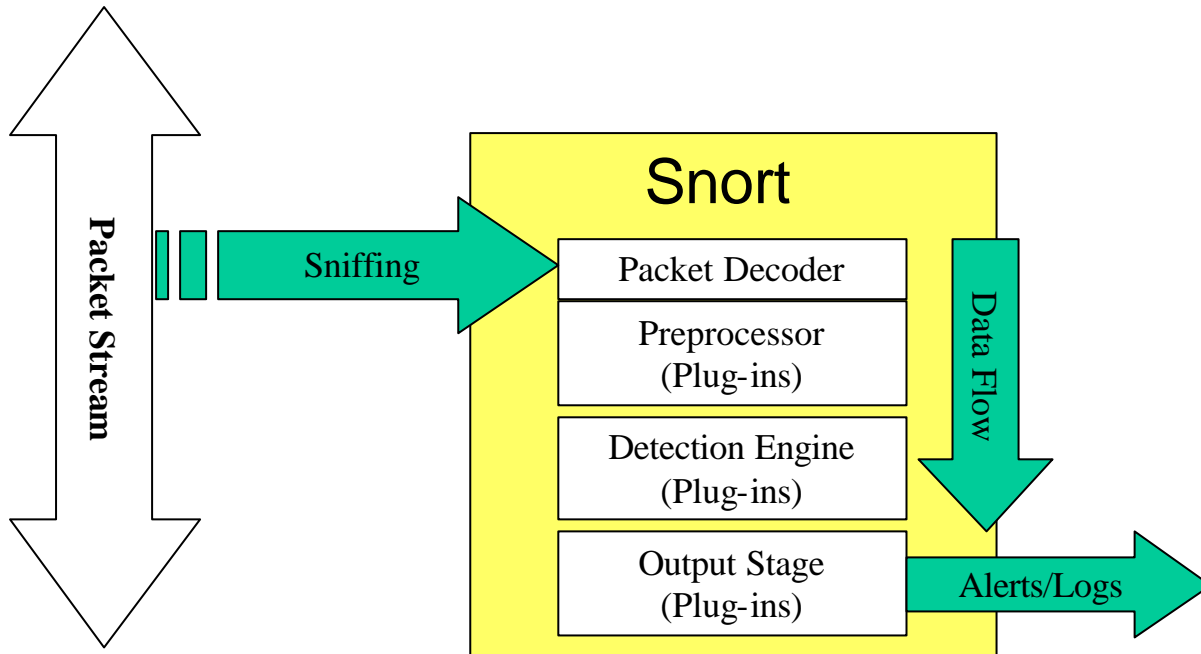
- Alle Phasen von Snort (Präprozessor + Plug-Ins)
- Zusätzlicher Scan nach Anomalien
 - Portscan,
 - Defragmentation,
 - TCP Streams,
 - Reassembly,
 - Application Layer Analyse (Telnet, HTTP, ...)
- Ausgabe
 - Datenbank (MySQL, Oracle, unixODBC, ...)
 - XML,
 - tcpdump,
 - Snort proprietär,
 - ASCII, syslog, WinPopup
 - ...



NIDS Modus ...

- Einsatz der Regeln
 - aktuell (07/2002) ca. 1300 !
- Multiple Erkennungsmodi
 - Rules/Signaturen
 - Protokoll Überprüfung
 - Statistische Überprüfung

Snort Architektur (Version 1.X) - Datenfluss

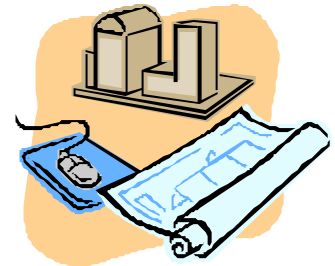


Graphic Copyright



Snort Architektur (Version 1.X) ...

- Snort Entwicklung
 - Sniffer -> Paket Logger -> NIDS
- Performance der Subsysteme
 - Decoder => sehr schnell
 - Detection Engine => schnell
 - Ausgabe/Präprozessor => langsam
- Implementierung als 3-dimensionale verkettete Liste
 - Dimension 1 und 2: Zeiger auf Felder im aktuell zu prüfenden Paket
 - Dimension 3: Verkettete Liste mit Zeigern auf Signaturen
 - sehr schnell/robust
 - Signaturen werden rekursiv geprüft



Snort Architektur (Version 1.X) ...

```
Alert tcp
  !192.168.1.0/24 any -> 192.168.1.0/24 any
  (flags: SF; msg: „SYN-FIN Scan“;)
```

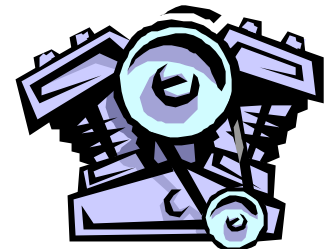
Regel ist in zwei Sektionen geteilt:

- Regel-Header:

```
Alert tcp !192.168.1.0/24 any -> 192.168.1.0/24 any
```

- Regel-Optionen:

```
(flags: SF; msg: „SYN-FIN Scan“;)
```



Snort Architektur (Version 1.X) - Optionen im Regel-Header

IP-Adresse

- Negation,
- Subnetting,

TCP/UDP ports

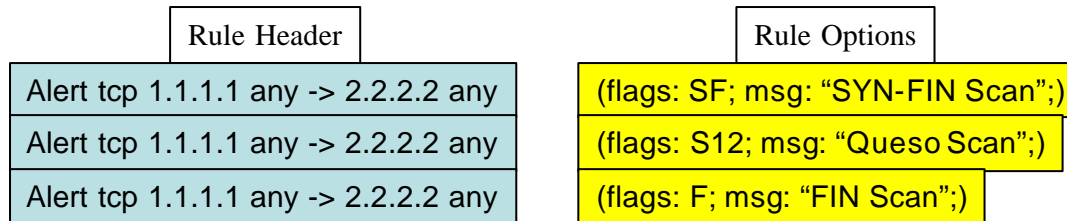
- Negation,
- Bereiche,
- größer/kleiner als,

Snort Architektur (Version 1.X) - Regel-Optionen

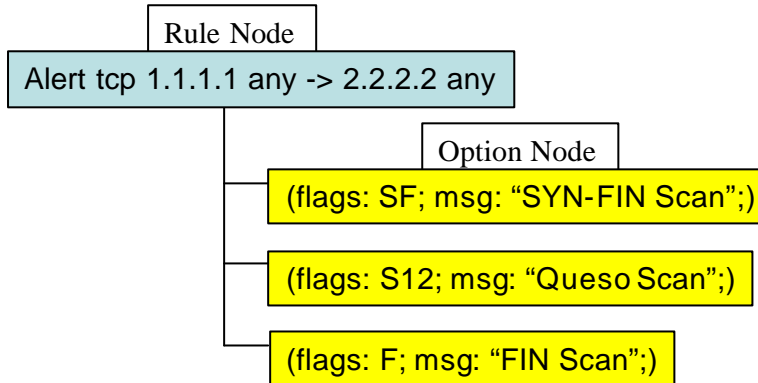
- IP TTL
- IP ID
- Fragment Size
- TCP flags
- TCP Ack number
- TCP Seq number
- Payload size
- Content
- Content offset
- ICMP type
- ICMP code

Snort Architektur (Version 1.X) ...

○ Benutzerpräsentation der Regeln

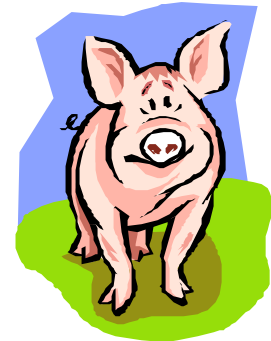


○ Interne Darstellung



Snort Architektur (Version 1.X) - Limits

- IPv4 basierend
- Paket Analyse
 - IP defragmentierung und TCP Streams nur über Präprozessoren
 - Interne Datenstrukturen sind nicht skalierbar
 - Nur Single-CPU Support
 - Application Layer wird nicht von der detection Engine decodiert (nur über Plug-In oder Präprozessor)



Snort Architektur (Version 1.X) – Limits ...

- Detection Engine & Präprozessor
 - Hinzufügen von weiteren Protokollen ist nicht möglich
 - Regel-Beschreibung ist nur auf Protokoll-Level möglich

- Ausgabe
 - Ausgabe Plug-Ins sehr langsam
 - Keine Möglichkeit um die CPU-Last aufzuteilen

Snort Architektur (Version 2.X)

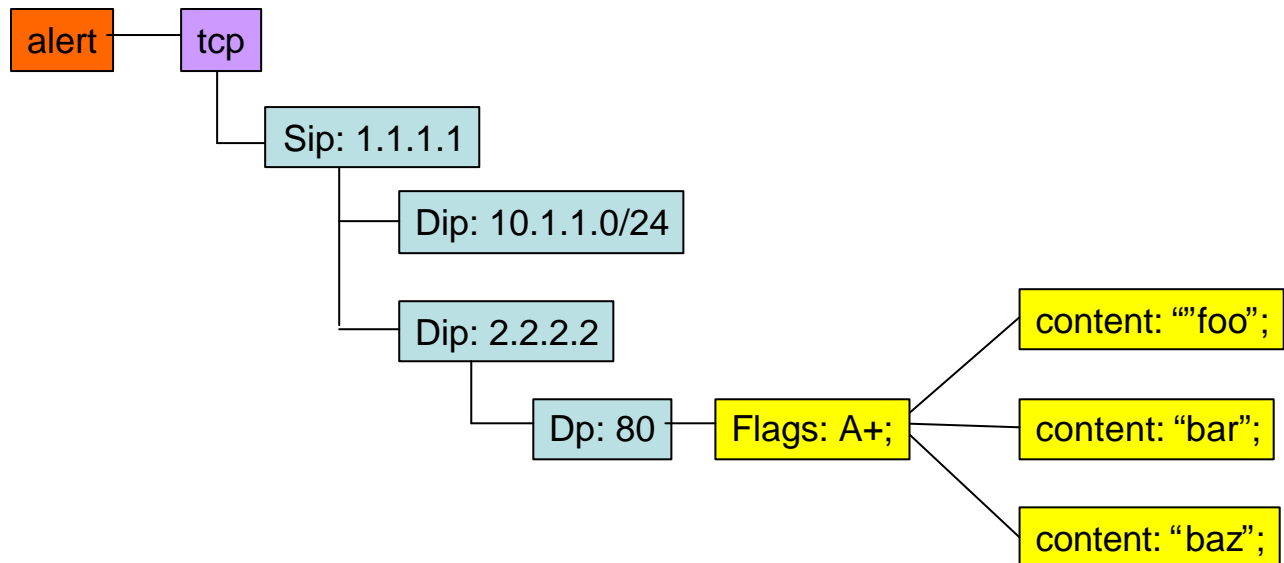
- Nur (vorerst) als Beta verfügbar
- Grundsatzliche Änderungen
 - Schneller,
 - Bessere Erweiterbarkeit,
 - Besserer Protokoll Unterstützung,
- Plug-Ins
 - Multi-Format für Regeln (DB, XML, ...)
 - Plug-Ins für Applikation Level Decoder,
 - OPSEC-Schnittstelle,

Snort Architektur (Version 2.X) - Verbesserungen

- Verbesserte detection engine
 - Aho-Corasick/Boyer-Moore Automat
 - ca. 500% schnellerer Vergleichs-Automat
- Gespoolte Ausgabe
 - Ausgabelogs können verwertet werden ohne den Prozess stoppen zu müssen
- Selbstoptimierung der Policy
 - Regeln werden weiter "getreed"
 - Mehrere Vergleiche laufen parallel ab
- Multi-CPU Unterstützung



Snort Architektur (Version 2.X) – Optimierung der Regeln



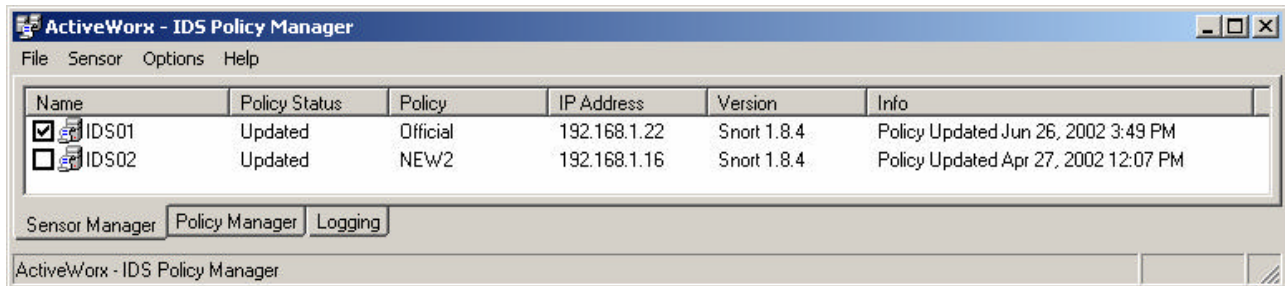
Tools für Snort

- Activeworx IDS Policy Manager
(Management von mehreren Sensoren, Policy Verwaltung, Änderung von Policies, Klick-Tool, kostenlos)
- SnortSnarf
(Logfile Analyse, Perl-basierend, Ausgabe im HTML-Format, kostenlos)
- ACID (Analysis Console for Intrusion Databases)
(Zentralisiertes Logging von mehreren Sensoren in eine MySQL-Datenbank, Zugriff über PHP-Web-GUI)
- Viele viele mehr (www.snort.org)

Activeworx IDS Policy Manager

Activeworx IDS Policy Manager

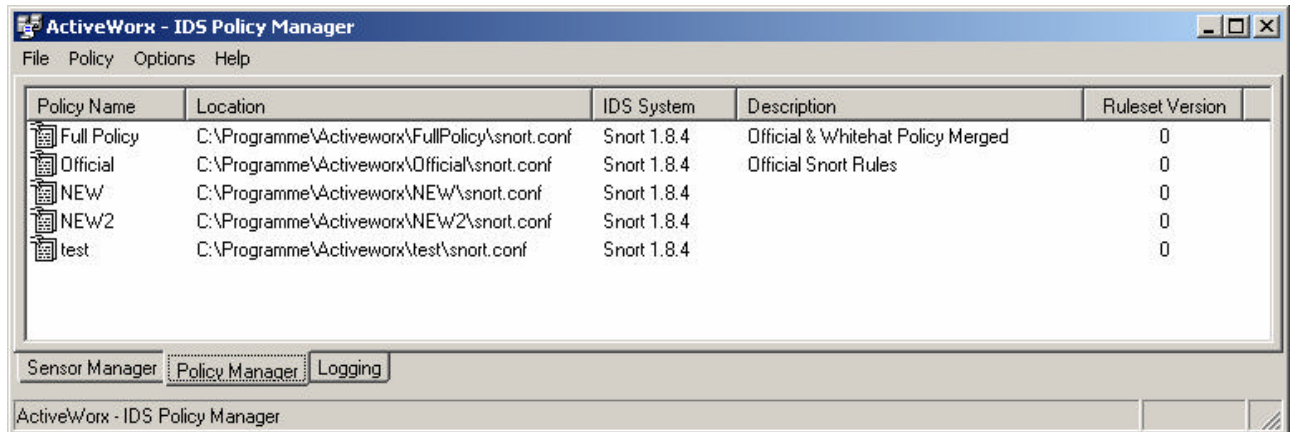
Sensor Manager



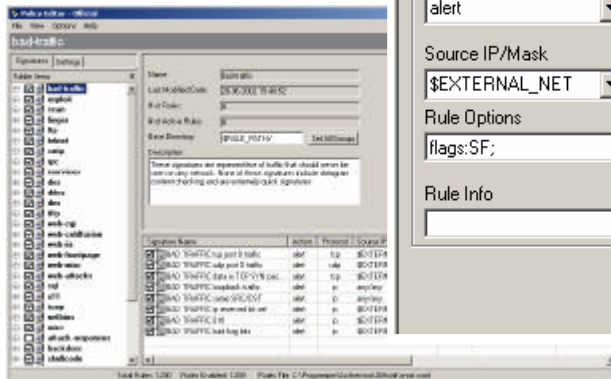
Möglichkeit zum Verwalten und Verteilen von unterschiedlichen Policies auf verschiedene Sensoren


Activeworx IDS Policy Manager

Policy Manager



Activeworx IDS Policy Manager - Policy Editor



Signature  arachnids

Name:

Signature ID: Signature Revision:

Reference Type	Value
arachnids	198

Signature

Action	Protocol	Classification	Priority
<input type="text" value="alert"/>	<input type="text" value="tcp"/>	<input type="text" value="attempted-recon"/>	<input type="text" value="2"/>

Source IP/Mask	Port	Direction	Destination IP/Mask	Port
<input type="text" value="\$EXTERNAL_NET"/>	<input type="text" value="any"/>	<input type="text" value="->"/>	<input type="text" value="\$HOME_NET"/>	<input type="text" value="any"/>

Rule Options

Rule Info

SnortSnarf

SnortSnarf

- Sammlung von Perl-Skripten,
- Auswertung des Alert-Log-Files (full/fast),
- Liste der angesprochenen Regeln,
- Liste der Top 20 Quell-IPs,
- Liste der Top 20 Ziel-IPs,

SnortSnarf ...

SnortSnarf: Snort signatures in /var/log/snort/snort_fast.txt et al - Microsoft Internet Explorer bereitgestellt von Lycos Euro

Datei Bearbeiten Ansicht Favoriten Extras ?

Zurück Suchen Favoriten Medien

Adresse <http://192.168.1.22> Wechseln zu

 **SnortSnarf start page**

All Snort signatures

[SnortSnarf](#) v020516.1

[Signature section \(61\)](#) [Top 20 source IPs](#) [Top 20 dest IPs](#)

61 alerts found using input module SnortFileInput, with sources:

- /var/log/snort/snort_fast.txt

Earliest alert at **16:46:37.093415** on 06/26/2002
Latest alert at **19:23:44.990252** on 06/26/2002

Internet



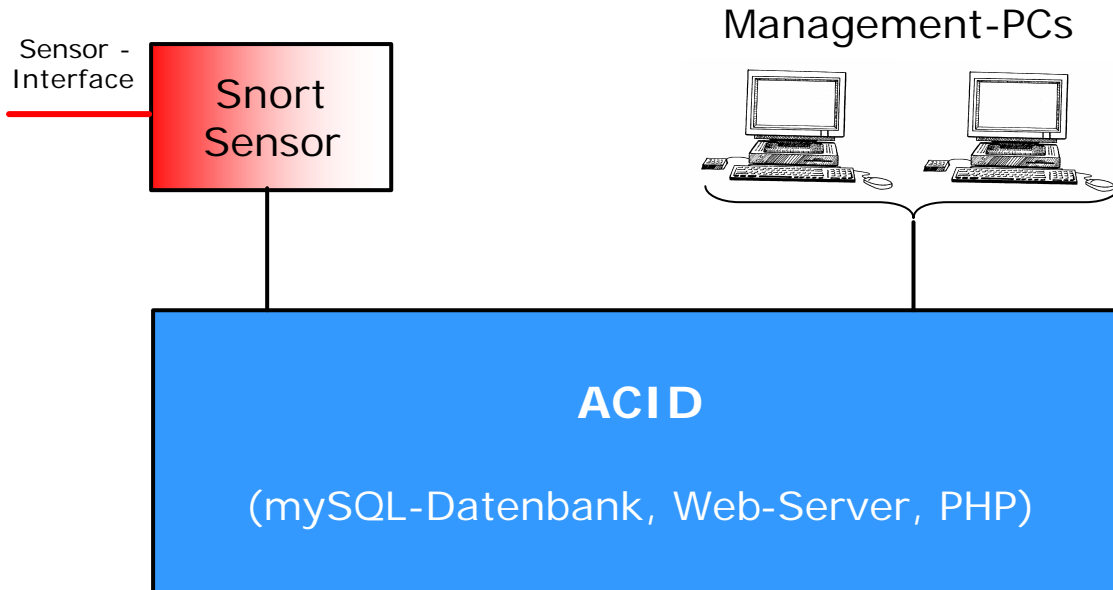
ACID

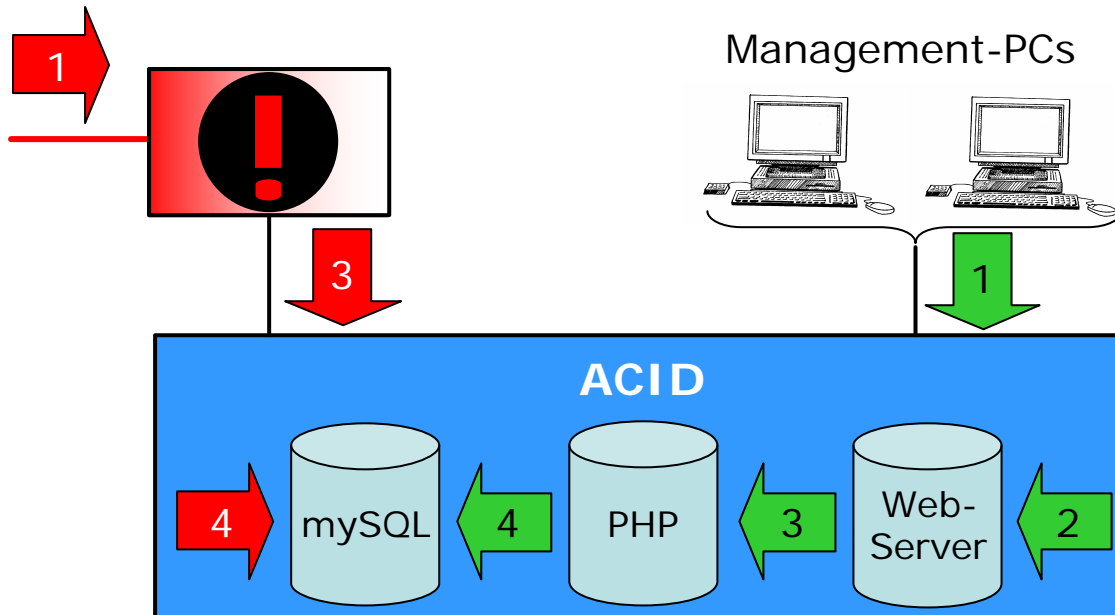
(Analysis Console for Intrusion Databases)

ACID

- Entwickelt von www.cert.org,
- Sammlung von PHP-Skripten,
- Echtzeit-Ansicht der Meldungen,
- Zentrales Logging in eine mySQL-Datenbank,
- Umfassende Auswertemöglichkeiten,

ACID ... Prinzipieller Aufbau



ACID ... Prinzipieller Ablauf

ACID ... Hauptmenü

Analysis Console for Intrusion Databases

Added 0 alert(s) to the Alert cache

Queried on: Fri September 13, 2002 15:29:04
Database: snort@localhost (schema version: 105)
Time window: [2002-08-05 10:48:57] - [2002-09-13 15:16:00]

<p>Sensors: 6</p> <p>Unique Alerts: 62 (10 categories)</p> <p>Total Number of Alerts: 3827</p> <ul style="list-style-type: none"> • Source IP addresses: 232 • Dest. IP addresses: 97 • Unique IP links: 477 • Source Ports: 664 <ul style="list-style-type: none"> ◦ TCP (654) UDP (20) • Dest. Ports: 181 <ul style="list-style-type: none"> ◦ TCP (178) UDP (3) 	<p>Traffic Profile by Protocol</p> <p>TCP (97%)</p> <p>UDP (3%)</p> <p>ICMP (1%)</p> <p>Portscan Traffic (0%)</p>
--	--

- [Search](#)
- [Graph Alert data](#) (EXPERIMENTAL)
- **Snapshot**
 - Most recent Alerts: [any protocol](#), [TCP](#), [UDP](#), [ICMP](#)
 - Today's alerts: [unique](#), [listing](#), [IP src / dst](#)
 - Last 24 Hours: alerts [unique](#), [listing](#), [IP](#)
 - Most [frequent 5 Alerts](#)
 - Most Frequent Source Ports: [any](#), [TCP](#), [UDP](#)
 - Most Frequent Destination Ports: [any](#), [TCP](#), [UDP](#)

ACID ... Chronologische Auflistung der Meldungen + Einsicht in die Protokoll-Ebene

The screenshot displays the ACID interface. On the left, a list of alerts is shown with columns for ID and Signature. The selected alert is #0-(2-60485) with the signature [arachNIDS] [CVE] TELNET access. On the right, the detailed view of this alert is shown, organized into protocol layers:

- Meta (Red):**
 - ID #:** 2 - 60485
 - Time:** 2002-09-05 10:48:57
 - Triggered Signature:** [arachNIDS] [CVE] TELNET access
 - Sensor:**

name	interface	filter
strids01	eth1	none
 - Alert Group:** none
- IP (Blue):**
 - source addr:** 53.142.12.22
 - dest addr:** 53.142.30.27
 - Ver:** 4
 - Hdr Len:** 5
 - TOS:** 0
 - length:** 55
 - ID:** 35282
 - flags:** 0
 - offset:** 0
 - TTL:** 58
 - chksum:** B610
 - FQDN:**

Source Name	Dest. Name
strsun30.debitel.de	Unable to resolve address
 - Options:** none
- TCP (Green):**
 - source port:** 23
 - dest port:** 1045
 - RST:** X
 - URG:** X
 - ACK:** X
 - PSH:** X
 - SYN:** X
 - FIN:** X
 - seq #:** 1589011408
 - ack:** 218645
 - offset:** 5
 - res:** 0
 - window:** 24820
 - urp:** 0
 - chksum:** 16752
 - Options:** none
- Payload (Purple):**
 - length:** 15
 - hex:** 000 : FF ED 18 FE FD 1F FE FD 23 FF FD 27 FF FD 24

ACID ...

- Häufigsten Meldungen,
- Häufigsten Quell-/Ziel-Adressen,
- Häufigsten Quell-/Ziel-Ports,
- Häufigsten Protokolle,
- Kategorien,
- Statistische und zeitliche Auswertungen,
- ...

Fragen ???

Bewertung/Meinungen !!!

