Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions

Defence
Intrusion
Detection

Eve
(Intruder)
Enemy takes some time to breach each of the levels of defence.

Defence-in-the-depth

- First-level defence
- Second-level defence
- Third-level defence
- Forth-level defence
DMZ – an area where military actions are prohibited

Trusted (our side)

Untrusted (their side)
Hello. How are you? Is this okay?

Even with the best defences, intruders can penetrate them.
Intrusion Detection Systems can help to reduce breaches
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions
CSI (Computer Security Institute) found:
- 70% of organisations had breaches
- 60% of all breaches came from inside their own systems

Internal threats (often a great threat than from outside)

Corporate access
Email access
Web access

Network/ Organisational perimeter

Firewall/ Gateway
(cannot deal with internal threats)

DoS (Denial-of-service)
External hack
Personal abuse
Worms/viruses
Terrorism/ extortion
Fraud

Assets
Users
Systems

DoS (Denial-of-service)

Authors: Prof Bill Buchanan
Defence-in-depth puts as many obstacles in the way of an intruder, so that it becomes harder to penetrate the network, and easier to detect.
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions
**Misuse Detection**
This attempts to model attacks on a system as specific patterns, and then scans for occurrences of these. Its disadvantage is that it struggles to detect new attacks.

**Anomaly Detection.**
This assumes that abnormal behaviour by a user can be correlated with an intrusion. Its advantage is that it can typically react to new attacks, but can often struggle to detect variants of known attacks, particularly if they fit into the normal usage pattern of a user. Another problem is that the intruder can mimic the behavioural pattern of the user.
Intrusion Detection System (IDS)

Log file monitors (LFM)
These monitor log files which are generated by network services, and look for key patterns of change. Swatch is a good example.

System Integrity Verifier
These monitor system files to determine if an intruder has changed them (a backdoor attack). A good example of this is Tripwire. It can also watch other key system components, such as the Windows registry and root/administrator level privileges.

User profiling

Network intrusion detection systems (NIDS)
These monitor packets on the network and tries to determine an intrusion. This is either host based (where it runs on a host), or can listen to the network using a hub, router or probe.
Intruder gains public information about the systems, such as DNS and IP information.

Outside reconnaissance

Intruder gains more specific information such as subnet layout, and networked devices.

Inside reconnaissance

Exploit

Intruder finds a weakness, such as cracking a password, breaching a firewall, and so on.

Profit

Data stealing, system damage, user abuse, and so on.

Foothold

Once into the system, the intruder can then advance up the privilege levels,

From code yellow to code red ...

Eve (Intruder)
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions

Host or Network?
Host or network?

Intrusion Detection System

Audit/logging

Host-based IDS listens to traffic in/out of a host

Network-based IDS listens to some/all network traffic

Public Web Server
Public FTP Server
Public Proxy Server

DMZ

NAT Device

Author: Prof Bill Buchanan
Protecting Intrusion Detection System

NAT Device

Firewall

DMZ

Author: Prof Bill Buchanan
This IDS cannot hear any traffic which is not addressed to it as it connects to a switch.
Using the span port

interface FastEthernet0/1
  port monitor FastEthernet0/2
  port monitor FastEthernet0/5
  port monitor VLAN2
!
interface FastEthernet0/2
!
interface FastEthernet0/3
  switchport access vlan 2
!
interface FastEthernet0/4
  switchport access vlan 2
!
interface FastEthernet0/5
!
interface VLAN1
  ip address 192.168.0.1 255.255.255.0
  no ip directed-broadcast
  no ip route-cache
!
This IDS detects successful attacks against firewall

IDS detects attacks against main firewall

IDS detects attacks against server

Host/Network

This IDS detects attacks against host

IDS detects internal attacks

Intrusion Detection

DMZ

Author: Prof Bill Buchanan

Protecting DMZ

Intrusion Detection

Intrusion Detection

Intrusion Detection

Intrusion Detection

Intrusion Detection

Host

IDS

Author: Prof Bill Buchanan
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions

Author: Prof Bill Buchanan
Agent-based system allows for distributed security

- Security agent
- QoS agent
- Management agent
- Auditing agent
- Reconfig agent
- SNORT agent
Requirements

1. Write SNORT rules
2. Invoke SNORT and get it to read the rules
3. View/process alerts
4. Device reconfiguration

SNORT agent

Security agent

Reconfig agent

Agent-based

alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP CWD ~root attempt"; flow:to_server,established; content:"CWD"; nocase; content:"~root"; nocase; distance:1; pcre:"/CWD\s+~root/SMI"; classtype:bad-unknown; sid:336; rev:7;)


Agent-based
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions

WinPCap

Author: Prof Bill Buchanan
WinPCap

Capture filter

WinPCap Pro

libpcap

Network Interface: Ethernet, Wireless, ADSL, etc

“ip”
“icmp”
“ip and tcp”
“host 192.168.0.1”
“host www.google.com”
“tcp port 8080”
“not tcp port http”
“ether host 00:cc:dd:00:cc:dd”

Wireshark

SNORT

User-defined agent

API Interface

Author: Prof Bill Buchanan

Integrating with WinPCap – capturing packets

Author: Prof Bill Buchanan
WinPCap

Integrating with WinPCap – capturing packets

Network Interface: Ethernet, Wireless, ADSL, etc

User-defined agent

Capture filter

Tamir Code Wrapper (.NET interface)

WinPCap

“ip”
“icmp”
“ip and tcp”
“host 192.168.0.1”
“host www.google.com”
“tcp port 8080”
“not tcp port http”
“ether host 00:cc:dd:00:cc:dd”
using System;
using Tamir.IPLib;
namespace NapierCapture
{
    public class ShowDevices
    {
        public static void Main(string[] args)
        {
            string verWinPCap = null;
            int count = 0;
            PcapDeviceList getNetConnections = SharpPcap.GetAllDevices();
            Console.WriteLine("WinPCap Version: {0}", verWinPCap);
            Console.WriteLine("Connected devices:");
            foreach (PcapDevice net in getNetConnections)
            {
                Console.WriteLine("{0} {1}" , count, net.PcapDescription);
                Console.WriteLine("\tName: \t{0}" , net.PcapName);
                Console.WriteLine("\tMode: \t\t{0}" , net.PcapMode);
                Console.WriteLine("\tIP Address: \t{0}" , net.PcapIpAddress);
                Console.WriteLine("\tLoopback: \t\t{0}" , net.PcapLoopback);
                count++;
            }
            Console.Write("Press any <RETURN> to exit");
            Console.Read();
        }
    }
}
namespace NapierCapture
{
    public class CapturePackets
    {
        public static void Main(string[] args)
        {
            PcapDeviceList getNetConnections = SharpPcap.GetAllDevices();
            NetworkDevice netConn = (NetworkDevice) getNetConnections[1];
            PcapDevice device = netConn;
            device.PcapOnPacketArrival +=
            new SharpPcap.PacketArrivalEvent(device_PcapOnPacketArrival);
            Console.WriteLine("Network connection: {0}",
            device.PcapDescription);
            device.PcapStartCapture();
            Console.Write("Press any <RETURN> to exit");
            Console.Read();
            device.PcapStopCapture();
            device.PcapClose();
        }
        private static void device_PcapOnPacketArrival(object sender,
            Packet packet)
        {
            DateTime time = packet.PcapHeader.Date;
            int len = packet.PcapHeader.PacketLength;
            Console.WriteLine("{0}:{1}:{2},{3} Len={4}",
            time.Hour,
            time.Minute, time.Second, time.Millisecond, len);
        }
    }
}
Intrusion Detection Systems

Introduction

Threats

Types

Host or Network?

Agent-based

Snort

A simple rule

A few intrusions

User profiling

Honeypots

IPS

Conclusions
Snort IDS

Other tools:
- Tcptrace. Identity TCP streams.
- Tcpflow. Reconstruct TCP streams.

SNORT agent

SNORT rules

Event data

Log data

Signature detection.
Identify well-known patterns of attack.

Anomaly detection.
Statistical anomalies, such as user logins, changes to files, and so on.

Other tools:
- Tcptrace. Identity TCP streams.
- Tcpflow. Reconstruct TCP streams.

Author: Prof Bill Buchanan
alert tcp any any -> 192.168.1.0/24 111 (content:"|00 01 86 a5|"; msg:"mountd access");

- **alert** Generate an alert and log packet
- **log** Log packet
- **pass** Ignore the packet
- **activate** Alert and activate another rule
- **Dynamic** Remain idle until activated by an activate rule

**Snort**

**Event data**

**Log data**

**SNORT agent**

**IDS**

**Snort rules**

Author: Prof Bill Buchanan
alert tcp any any -> 192.168.1.0/24 111 (content:"|00 01 86 a5|"; msg:"mountd access");
alert tcp any any -> 192.168.1.0/24 111 (content:"|00 01 86 a5|"; msg:"mountd access");

Payload detection:
Hex sequence  "|00 01 86 a5|
Text sequence  "USER root"

Modifiers:
- rawbytes
- offset
- distance
- within
- uricontent
- bytejump

Event data
Log data

Author: Prof Bill Buchanan
alert tcp any any -> 192.168.1.0/24 111 (content:"|00 01 86 a5|"; msg:"mountd access");

Message-to-display

SNORT agent

Event data

Log data

Author: Prof Bill Buchanan

Payload detection

Log data

Event data

SNORT agent
alert tcp $HOME_NET any -> $EXTERNAL_NET 1863
(msg:"CHAT MSN login attempt"; flow:to_server,established; content:"USR "; depth:4;
nocase; content:" TWN "; distance:1; nocase;
classtype:policy-violation; sid:1991; rev:1;)

The SID and REV represent know Snort rules:

- Less 100 Reserved for future use
- Between 100 and 1,000,000 are rules included with the Snort distribution
- More than 1,000,000 is for local rules

For example: sid:336; rev:7; represents an attempt to change to the system administrator's account in FTP.
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions

A simple rule
alert tcp any any -> any any (content:"the"; msg:"The found ....");

Snort -v -c bill.rules -l /log

16 January 10:27pm

[**] [1:0:0] The found .... [**] [Priority: 0]
DgmLen:347 DF
***AP*** Seq: 0xF842A9D3 Ack: 0x3524EE7B Win: 0x4321 TcpLen: 20

[**] [1:0:0] The found .... [**] [Priority: 0]
DgmLen:394 DF
***AP*** Seq: 0x3524EE7B Ack: 0xF842AB06 Win: 0x42E4 TcpLen: 20

[**] [1:0:0] The found .... [**] [Priority: 0]
01/16-22:27:35.290026 0:60:B3:68:B1:10 -> 0:3:6D:FF:2A:51 type:0x800 len:0x5D
DgmLen:79 DF
***AP*** Seq: 0xF842AB06 Ack: 0x3524EFDD Win: 0x41BF TcpLen: 20
A simple rule

IDS

Payload detection

Version | Header len. | Type of service
---------|-------------|-----------------
|          |             |                  |
Total length
Identification
0 | D | M | Fragment Offset
Time-to-live (TTL) | Protocol
Header Checksum
|          |             |                  |
Source IP Address
|          |             |                  |
Destination IP Address

TCP Source Port
TCP Destination Port
Sequence Number
Acknowledgement Number
Data Offset
Flags/Reserved
Data Offset
Window
Checksum
Urgent Pointer
Ethernet frame

Src MAC address | Dest. MAC address | Type | Length
---------|-----------------|------|-----
IP header | TCP header | Data

[**] [1:0:0] The found .... [**]
Priority: 0
***AP*** Seq: 0xF842A9D3 Ack: 0x3524EE7B Win: 0x4321 TcpLen: 20
A simple rule
IDS

Payload detection
Version Header len. Type of service
Total length
Identification
0 D M Fragment Offset
Time-to-live (TTL) Protocol
Header Checksum
Source IP Address
Destination IP Address

TCP Source Port
TCP Destination Port
Sequence Number
Acknowledgement Number
Data Offset Flags/Reserved
Window Checksum Urgent Pointer

Src MAC address Dest. MAC address Type Length
IP header TCP header Data

Ethernet frame

[**] [1:0:0] The found .... [**]
[PRIORITY: 0]
01/16-22:27:35.286762 0:60:B3:68:B1:10 -> 0:36D:FF:2A:51 type:0x800 len:0x169
***AP*** Seq: 0xF842A9D3 Ack: 0x3524EE7B Win: 0x4321 TcpLen: 20
A simple rule

IDS

Payload detection

Version | Header len. | Type of service
---|---|---

Total length

Identification

0 | D | M | Fragment Offset

Time-to-live (TTL) | Protocol

Header Checksum

Source IP Address

Destination IP Address

TCP Source Port

TCP Destination Port

Sequence Number

Acknowledgement Number

Data Offset | Flags/Reserved

Window

Checksum

Urgent Pointer

Src MAC address | Dest/ MAC address | Type | Length

IP header | TCP header | Data

Ethernet frame

[**] [1:0:0] The found .... [**]

[Priority: 0]


***AP*** Seq: 0xF842A9D3 | Ack: 0x3524EE7B | Win: 0x4321 | TcpLen: 20

Author: Prof Bill Buchanan
Log of traffic between port 3423 and 455 to/from 192.168.0.20
A simple rule

Payload detection

Flags – the flag field is defined as UAPRSF,

- U is the urgent flag (URG).
- A the acknowledgement flag (ACK).
- P the push function (PSH).
- R the reset flag (RST).
- S the sequence synchronize flag (SYN).
- F the end-of-transmission flag (FIN).
The SYN flag identifies a connection.

Flags – the flag field is defined as UAPRSF,

- U is the urgent flag (URG).
- A the acknowledgement flag (ACK).
- P the push function (PSH).
- R the reset flag (RST).
- S the sequence synchronize flag (SYN).
- F the end-of-transmission flag (FIN).
An incoming SYN flag is important in detecting the start of a connection. The main flags are:

F FIN
S SYN
R RST
P PSH
A ACK
U URG

The following modifiers can be set to change the match criteria:

+ match on the specified bits, plus any others
* match if any of the specified bits are set
! match if the specified bits are not set

Example to test for SYN flag:

alert tcp any any -> any any (flags:S;)

It is often important to know the flow direction. The main flow rules options are:

- to_client. Used for server responses to client.
- to_server Used for client requests to server.
- from_client. Used on client responses.
- from_server. Used on server responses.
- established. Established TCP connections.

Example to test for an FTP connection to the users computer:

alert tcp any any -> $HOME_NET 21 (flow: from_client; content: "CWD incoming"; nocase;
A simple rule

IDS

Payload detection

TCP TTL:128 TOS:0x0 ID:975 IpLen:20 DgmLen:48 DF

****S** Seq: 0x26885B8B Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK
=+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

TCP TTL:128 TOS:0x0 ID:653 IpLen:20 DgmLen:48 DF

***A**S** Seq: 0xE9A4004C Ack: 0x26885B8C Win: 0x4470 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK
=+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

TCP TTL:128 TOS:0x0 ID:977 IpLen:20 DgmLen:40 DF

***A**** Seq: 0x26885B8C Ack: 0xE9A4004D Win: 0x4470 TcpLen: 20
=+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++=
Devices can only communicate directly if they have the MAC address and IP address.

ARP request: Who has 192.168.0.168?

ARP request is broadcast to the network

ARP information:

- 01/16-09:31:08.785149 ARP who-has 192.168.0.168 tell 192.168.0.22
- 01/16-09:45:59.458607 ARP who-has 192.168.0.42 tell 192.168.0.216
- 01/16-09:45:59.459159 ARP reply 192.168.0.42 is-at 0:20:18:38:B8:63
- 01/16-09:46:03.857325 ARP who-has 192.168.0.104 tell 192.168.0.198
- 01/16-09:46:10.125715 ARP who-has 192.168.0.15 tell 192.168.0.38
- 01/16-09:46:10.125930 ARP who-has 192.168.0.38 tell 192.168.0.15
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions

A few intrusions
A few intrusions

IDS

Policy Definition

Aims/objectives of the organisation

Legal, moral and social responsibilities

Technical feasibility

Policy Implementation

Evaluation

Verification

Audit

Policy Definition

Operating System rights

Firewall rules

Domain rights

Event log definition

Application rights
A few intrusions

IDS

alert tcp $HOME_NET any -> $EXTERNAL_NET 1863
(msg:"CHAT MSN login attempt"; flow:to_server,established; content:"USR "; depth:4; nocase; content:" TWN "; distance:1; nocase;
classtype:policy-violation; sid:1991; rev:1;)

No. Source Destination Protocol Info
1 192.168.0.3 207.46.28.93 TCP 5398 > 1863 [SYN] Seq=0 Len=0 MSS=1460
2 207.46.28.93 192.168.0.3 TCP 1863 > 5398 [SYN, ACK] Seq=0 Ack=1 Win=5840
3 192.168.0.3 207.46.28.93 TCP 5398 > 1863 [ACK] Seq=1 Ack=1 Win=17520
4 192.168.0.3 207.46.111.39 MSNMS USR 2 TWN I test@hotmail.com
5 207.46.111.39 192.168.0.3 MSNMS USR 26 OK test@hotmail.com 1 0

private static void device_PcapOnPacketArrival(...)
{
    if (packet is TCPPacket)
    {
        TCPPacket tcp = (TCPPacket)packet;
        int destPort = tcp.SourcePort; byte [] b = tcp.Data;
        ASCIIEncoding format = new ASCIIEncoding();
        string s = format.GetString(b); s=s.ToLower();
        if (destPort==1863 && (s.StartsWith("usr ")) && s.IndexOf(" twn ")>0 )
            Console.WriteLine("MSN Messenger Login");
    }
}
A few intrusions

IDS

alert tcp $HOME_NET any -> $EXTERNAL_NET 1863

(msg:"CHAT MSN login attempt";

flow:to_server,established;

content: "USR "; depth 4; nocase;

costent:" TWN ";distance:1; nocase;

classstype:policy-violation; sid:1991; rev:1;)

No. Source Destination Protocol Info
1 192.168.0.3 207.46.28.93 TCP 1863 > 5398 [SYN, ACK] Seq=0 Ack=1 Win=5840
2 192.168.0.3 207.46.28.93 TCP 5398 > 1863 [ACK] Seq=1 Ack=1 Win=17520
3 207.46.28.93 192.168.0.3 TCP USR 2 TWN I test@hotmail.com
4 192.168.0.3 207.46.28.93 TCP USR 26 OK test@hotmail.com 1 0
5 192.168.0.3 207.46.111.39 MSNMS
6 207.46.111.39 192.168.0.3 MSNMS

DEPTH. Modifier for previous content ... defines to look within 4 bytes of the payload

NOCASE; Modifier for previous content ... ignore the case of the content

DISTANCE. Modified for previous content and defines how far into the payload it should search (in bytes)
A few intrusions

Intrusions/Policy Violations

alert tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"CHAT MSN login attempt"; flow:to_server,established; content: "USR "; depth:4; nocase; content:" TWN "; distance:1; nocase; classtype:policy-violation; sid:1991)

No. Source Destination Protocol Info
1 192.168.0.3 207.46.28.93 TCP SYN Seq = 0 Len = 0 MSS = 1460
2 192.168.0.3 207.46.28.93 TCP SYN Seq = 0 Len = 0 MSS = 1460
3 207.46.28.93 192.168.0.3 TCP SYN,ACK Seq = 0 Ack = 1 Win = 5840
4 192.168.0.3 207.46.28.93 TCP ACK Seq = 1 Ack = 1 Win = 17520
5 192.168.0.3 207.46.28.93 HTTP MSNMS USR 26 OK test hotmail.com
6 207.46.111.39 192.168.0.3 HTTP MSNMS USR 26 OK test hotmail.com

...
alert tcp $EXTERNAL_NET any -> $HOME_NET 21
(msg:"FTP CWD ~root attempt"; flow:to_server,established; content:"CWD";
nocase; content:"~root"; nocase; distance:1; pcre:"/^CWD\s+~root/smi";
class:unknown; sid:336; rev:7;)

private static void device_PcapOnPacketArrival()
{
    if(packet is TCPPacket)
    {
        TCPPacket tcp = (TCPPacket)packet;
        int destPort = tcp.SourcePort;
        byte [] b = tcp.Data;
        ASCIIEncoding format = new ASCIIEncoding();
        string s = format.GetString(b); s=s.ToLower();
        if (destPort==21 && s.IndexOf("cwd")>0 && s.IndexOf("~root")==0 )
            Console.WriteLine("FTP CWD ~root attempt");
    }
}
A few intrusions

Intrusion Detection System

FTP server (listening on port 21)

telnet ftp.test.com 21

220-Microsoft FTP Service
220 NTXPW35

User bill

331 Password required for ******.

Password: ******

230-FTP Server User bill logged in.

help

214-The following commands are recognized

ABOR ACCT ALLO APPE CDUP
CWD DELE FEAT HELP
LIST
MDTM MKD MODE NLST NOOP
OPTS PASS PASV PORT PWD
QUIT REIN REST RETR
RMD RNFR RNTO SITE
SIZE SMNT STAT STOR
STOU STRU SYST TYPE
USER XCUP XCWD XMKD
XPWD XRMD

214 HELP command successful.

pwd

257 "/bill" is current directory.

cwd /

250 CWD command successful.

pwd

257 "/" is current directory.

cwd ~root

250 CWD command successful.

list

150 Opening ASCII mode data connection for /

bin/ls.

private static void device_PcapOnPacketArrival(...)

if (packet is TCPPacket)

TCPPacket tcp = (TCPPacket) packet;

int destPort = tcp.SourcePort;

byte[] b = tcp.Data;

ASCIIEncoding format = new ASCIIEncoding();

string s = format.GetString(b);

if (destPort == 21 && s.IndexOf("cwd") > 0 && s.IndexOf("~root") > 0)

Console.WriteLine("FTP CWD ~root attempt");

cwd ~root

250 CWD command successful.

list

150 Opening ASCII mode data connection for /

bin/ls.
A few intrusions

IDS

alert tcp $HOME_NET any -> $EXTERNAL_NET 8888  
(msg:"P2P napster login"; flow:to_server,established;  
content:"00 0200"; offset:1; depth:3;  
classtype:policy-violation; sid:549; rev:6;)

private static void device_PcapOnPacketArrival(...)  
{
  if(packet is TCPPacket)  
  {
    TCPPacket tcp = (TCPPacket)packet;  
    int destPort = tcp.SourcePort;  
    byte [] b = tcp.Data;  
    if (destPort==8888 && b[0]==0x00 && b[1]==0x20 && b[2]==0x00)  
      Console.WriteLine("P2P napster login");
  }
}
alert tcp $HOME_NET any -> $EXTERNAL_NET 80
(msg:"MULTIMEDIA Quicktime User Agent access"; flow:to_server,established;
content:"User-Agent\: Quicktime";
class:policy-violation; sid:1436; rev:2;)

GET /napierstream
HTTP/1.0 User-Agent: QuickTime/7.1 (qtver=7.1;os=Windows NT 5.1)
Accept: application/x-rtsp-tunnelled
Pragma: no-cache Cache-Control: no-cache

private static void device_PcapOnPacketArrival(..)
{
    if (packet is TCPPacket)
    {
        TCPPacket tcp = (TCPPacket)packet;
        int destPort = tcp.SourcePort;
        byte [] b = tcp.Data;
        ASCIIEncoding format = new ASCIIEncoding();
        string s = format.GetString(b); s=s.ToLower();
        if (destPort==80 && s.StartsWith("User-Agent\: Quicktime "))
            Console.WriteLine("MULTIMEDIA Quicktime User Agent access ");
    }
}
A few intrusions

Intrusions/Policy Violations

alert tcp $HOME_NET any -> $EXTERNAL_NET ...

private static void device_PcapOnPacketArrival(...)

File: C:\DOCUME~1\WILLIA~1\LOCALS~1\Temp\ebuffer\34b3428\14149.KB.00:03:...

Author: Prof Bill Buchanan
alert tcp $SMTP_SERVERS any -> $EXTERNAL_NET 25
(msg:"VIRUS OUTBOUND .exe file attachment";
flow:to_server,established; content:"Content-Disposition|3a|";
content:"filename=|22|"; distance:0; within:30;
content":.exe|22|"; distance:0; within:30; nocase;
class type:suspicious-filename-detect; sid:2160; rev:1;)

private static void device_PcapOnPacketArrival(..)
{
    TCPPacket tcp = (TCPPacket)packet;
    if(packet is TCPPacket)
    {
        int destPort = tcp.SourcePort; byte [] b = tcp.Data;
        ASCIIEncoding format = new ASCIIEncoding();
        string s = format.GetString(b); s=s.ToLower();
        if (destPort==25 && s.IndexOf("Content-Disposition;")>0
            && s.IndexOf("filename=\"")>0 && s.IndexOf(".exe\"")>0 )
            Console.WriteLine("VIRUS OUTBOUND .exe file attachment");
    }
}
A few intrusions

private static void device_PcapOnPacketArrival(...)
{
    if(packet is TCPPacket)
    {
        TCPPacket tcp = (TCPPacket)packet;
        if (tcp.Syn=true && tcp.Fin=true)
            Console.WriteLine("SYN FIN Scan");
    }
}
alert tcp $TELNET_SERVERS 23 -> $EXTERNAL_NET any (msg:"TELNET root login"; flow:from_server,established; content:"login|3A| root"; classtype:suspicious-login; sid:719; rev:7;)

A few intrusions

IDS

Intrusions/Policy Violations
A few intrusions

IDS

Author: Prof Bill Buchanan

UDP/TCP Port Scans

A particular threat is the TCP/UDP port scanner, which scans for open ports on a host.

If an intruder finds one, it may try and connect to it.

An open port is in the LISTEN state.

C:\log>netstat -a
Active Connections
Proto   Local Address          Foreign Address        State
TCP     bills:epmap            bills:0                LISTENING
TCP     bills:microsoft-ds    bills:0                LISTENING
TCP     bills:1035             bills:0                LISTENING
TCP     bills:3389             bills:0                LISTENING

Typical scans:
Ping sweeps.
TCP scans.
UDP scans.
OS identification scans.
Account scans.

Open port 10?
Open port 11?
..
Open port 8888?
A particular threat is the ping port scanner, which pings multiple hosts to see which ones are alive. If an intruder finds one, they may try and connect to it. Often ping (ICMP) is blocked on the gateway of the network.

Typical scans:
- Ping sweeps.
- TCP scans.
- UDP scans.
- OS identification scans.
- Account scans.

Ping 192.168.0.1?
Ping 192.168.0.1?
...
Ping 192.168.0.253?
Ping 192.168.0.254?
TSeq. This is where SYN packets are sent, and the TCP sequence numbers are analysed.

T1. This is a SYN packet with certain options (WNMTE) set is sent to an open TCP port.
T2. This is a NULL packet with options (WNMTE) and is sent to an open TCP port.
T3. This is a SYN,FIN,PSH,URG packet with options (WNMTE), and sent to an open TCP port.
T4. This is an ACK packet with options (WNMTE) and is sent to an open TCP port.
T5. This is a SYN packet with options (WNMTE) and is sent to a closed TCP port.
T6. This is an ACK packet with options (WNMTE) and is sent to a closed TCP port.
T7. This is a FIN,PSH,URG packet with options (WNMTE) and is sent to a closed TCP port.
PU. This is a packet sent to a closed UDP port.
A few intrusions

**IDS**

**Author:** Prof Bill Buchanan

**User account scans**

**Typical problems:**
- Anonymous logins
- Using the same password as user ID
- Using password as password
- Using root login
- Using system default logins
- Weak passwords
- Well-known passwords
- Social Engineering

**Typical scans:**
- Ping sweeps.
- TCP scans.
- UDP scans.
- OS identification scans.
- Account scans.
A few intrusions

User account scans

FTP Username/

Password scan

IDS

FTP Username/
A few intrusions

IDS

User account scans

Successful login to FTP server
NOTE:
The NMAP program should only be used on machines which you are under control of, and in a local, and isolated environment. It should only be used to determine possible weaknesses and vulnerabilities.
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions

User Profiling
User profiling for on-line purchases

Name: Fiona Smith
Nationality: British
Location: Edinburgh
Gender: Female
Typical purchase: Computer equipment
Average Purchases/week: 5
Average Value of purchases: £30
Browser used: Mozilla
Date of last purchase: 6 May 2008
Email address: f.smith@nowhere

Name: Fred McLean
Nationality: USA
Location: Washington
Gender: Male
Typical purchase: Fish Food
Average Purchases/week: 50
Average Value of purchases: $4
Browser used: IE
Date of last purchase: 18 Sept 2008
Email address: f.mclean@usa

Name: Michel Weber
Nationality: German
Location: Munich
Gender: Male
Typical purchase: Flowers
Average Purchases/week: 0.005
Average Value of purchases: €43
Browser used: Opera
Date of last purchase: 1 Mar 2007
Email address: m_weber@de

Name: Amélie Cheney
Nationality: French
Location: Paris
Gender: Female
Typical purchase: Clothes
Average Purchases/week: 70
Average Value of purchases: €13
Browser used: Mozilla
Date of last purchase: 16 Sept 2008
Email address: a.cheney@fr.edu

Name: A.N.Other
Nationality: Any
Location: Nowhere
Gender: Female/Male
Typical purchase: High-value goods
Average Purchases/week: 1000
Average Value of purchases: $9999
Browser used: Not known
Date of last purchase: Today
Email address: doesnt.exist

Transactions are checked against user profile

User profiler (such as bank transaction agent)

User/behaviour profiling is especially useful in fraud detection

Author: Prof Bill Buchanan

User profiling

Profiles
User profiling for local access/usage

1. On-login, the user profile is uploaded to the local machine.

2. Agent on each machine analyses the current user, and reports on differences of behaviour.

Profiles

User profiler (monitors alerts from user agents)
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions

Honeypots

Author: Prof Bill Buchanan
This device has all the required weaknesses, such as:

- Default administrator/password.
- Dummy users with weak passwords.
- Ports open for connection.
- React to virus/worm systems (but simulate conditions).
Honeypot types

High-interaction honeypot. This simulates all the aspects of the operating system.

Low-interaction honeypot. This simulates only part of the network stack (such as for Honeyd).
- can be virtual (from a virtual machine) or simulated by another machine.

Open ports: 110 (POP-3), 80 (HTTP), 21 (FTP), 22 (SSH)

Honeyd.conf

create default
set default personality "Windows XP"
set default default tcp action reset
add default tcp port 110 "sh scripts/pop.sh"
add default tcp port 80 "perl scripts/iis-0.95/main.pl"
add default tcp port 25 block
add default tcp port 21 "sh scripts/ftp.sh"
add default tcp port 22 proxy $ipsrc:22
add default udp port 139 drop
set default uptime 3284460

### Cisco router
create router
set router personality "Cisco PIX Firewall (PixOS 5.2 - 6.1)"
add router tcp port 23 "/usr/bin/perl scripts/router-telnet.pl"
set router default tcp action reset
set router uid 32767 gid 32767
set router uptime 1327650
# Bind specific templates to specific IP address
# If not bound, default to Windows template
bind 192.168.1.150 router
while ($string)

"Users (authorized or unauthorized) have no explicit or implicit expectation of privacy. Any or all uses of this system may be intercepted, monitored, recorded, copied, audited, inspected, and disclosed to authorized site, and law enforcement personnel, as well as to authorized officials of other agencies, both domestic and foreign. By using this system, the user consents to such interception, monitoring, recording, copying, auditing, inspection, and disclosure at the discretion of authorized site. Unauthorized or improper use of this system may result in administrative disciplinary action and civil and criminal penalties. By continuing to use this system you indicate your awareness of and consent to these terms and conditions of use. LOG OFF IMMEDIATELY if you do not agree to the conditions stated in this warning."

User Access Verification

syswrite STDOUT, $string;
$count = 0;
while ($count < 3) {
    do {
        $count++;
        syswrite STDOUT, "$\n"
        $word = read_word("Username: ", 1);
    } while (!$word || $count < 3);
    if ($count >= 3 && !$word) {
        exit;
    }
    $password = read_word("Password: ", 0);
    if (!$password) {
        syswrite STDOUT, "% Login invalid\n"
    } else {
        syswrite STDERR, "Attempted login: $word/$password";
        syswrite STDOUT, "% Access denied\n";
    }
}

exit;
sub read_word {
    local $prompt = shift;
    local $echo = shift;
    local $word;

    $prompt = ";
    $notification = ";
    $proxy = "127.0.0.1"

    if ($proxy)
}

$alarm = ";
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)

$word = ";
$alarm = 0;
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)

$word = ";
$alarm = 0;
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)

$word = ";
$alarm = 0;
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)

$word = ";
$alarm = 0;
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)

$word = ";
$alarm = 0;
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)

$word = ";
$alarm = 0;
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)

$word = ";
$alarm = 0;
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)

$word = ";
$alarm = 0;
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)

$word = ";
$alarm = 0;
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)

$word = ";
$alarm = 0;
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)

$word = ";
$alarm = 0;
$notification = ";
$proxy = "127.0.0.1"

if ($alarm)
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions

IPS and In/out-line
In-line IDS, which can decide to drop a packet, alarm (send an alert/log) or reset a connection.

Out-of-line IDS, which passively listens to traffic and cannot actually drop packets (unless there is an IPS).

In-line IDS has the advantage that they can act on the intrusion, but it has a performance impact. The signatures are also difficult to change/upgrade.

Out-of-line IDS has the advantage of being able to more easily craft an IDS rule, but cannot take actions, directly.
Example Cisco IDS signatures
1001 – Bad IP Options (Info)
1100 – IP Fragment (Attack)
2000 - ICMP echo reply (Info)
2154 – Ping of death (Attack)
3041 – SYN/FIN Packet (Attack)
3040 – NULL TCP Packet (Attack)
3050 – Half open SYN (Attack)
3152 – CWD Root on FTP (Info)

Syslog Server (stores alerts/logs/etc)

((config)# ip audit ?
   attack Specify default action for attack signatures
   info Specify default action for informational signatures
   name Specify an IDS audit rule
   notify Specify the notification mechanisms (nr-director or log) for the alarms
   po Specify nr-director's PostOffice information (for sending events to the nr-directors
   signature Add a policy to a signature
   smtp Specify SMTP Mail spam threshold
((config)# ip audit notify ?
   log Send events as syslog messages
   nr-director Send events to the nr-directors
((config)# ip audit notify log
((config)# logging 132.191.125.3
((config)# ip audit ?
   attack Specify default action for attack signatures
   info Specify default action for informational signatures
   name Specify an IDS audit rule
   notify Specify the notification mechanisms (nr-director or log) for the alarms
   po Specify nr-director's PostOffice information (for sending events to the nr-directors
   signature Add a policy to a signature
   smtp Specify SMTP Mail spam threshold
((config)# ip audit info ?
   action Specify the actions
((config)# ip audit info action ?
   alarm Generate events for matching signatures
   drop Drop packets matching signatures
   reset Reset the connection (if applicable)
((config)# ip audit info action drop
((config)# ip audit attack action reset
((config)# ip audit signature ?
   <1-65535> Signature to be configured
((config)# ip audit signature 1005 disable
((config)# ip audit smtp ?
   spam Specify the threshold for spam signature
   <cr>
((config)# ip audit smtp spam ?
   <1-65535> Threshold of correspondents to trigger alarm
((config)# ip audit smtp spam 4
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions

Anomaly Detection
Defence
Intrusion Detection
Defence

Author: Prof Bill Buchanan
**Anomaly detection:**

Learn normal activity, such as:

- User activity.
- System activity
- Server activity
- Network activity
- Application activity
- And so on
User anomaly:
- Typing speed
- Packages used
- Working hours
- Emails sent/hr
- Web sites visited

System anomaly:
- CPU Usage/min
- Threads/min
- Disk writes/min

Bob
Network anomaly
IP packets (%)
TCP packets (%)
HTTP (%)
FTP (%)

---

FTP threshold (2%)
Intrusion Detection Systems

Introduction
Threats
Types
Host or Network?
Agent-based
Snort
A simple rule
A few intrusions
User profiling
Honeypots
IPS
Conclusions

Conclusions