

Network Intrusion Detection Systems (IDS) Open Source Security Tools for Information Technology Professionals

School of Professional Studies (SPS)

The City University of New York (CUNY)

Aron Trauring
Adjunct Professor
CEO, Zoteca

Class 10 November 28th, 2005

What is an IDS?

- Modified network sniffer
- Sees all traffic on the network
- Tries to sense potential bad traffic
- Sends alert when bad traffic is found

How Does and IDS Work?

1. Compares network traffic to a database of known bad activity (Network IDS)
2. Check integrity of key system files (Host IDS)

Purpose of an IDS

- To detect traffic that gets through a misconfigured firewall
- To detect malicious traffic on ports open to legitimate applications
- To detect attacks coming from inside (behind firewall)

IDS Signature Example

- Nimda and Code Red use the `cmd.exe` attack against IIS
- With IIS having admin privileges, execute `cmd.exe` in a writable directory
- Exploits an IIS buffer overflow in IIS module ISAPI
- No reason for legitimate user to be copying this file over network
- If see network traffic with name "`cmd.exe`" can flag as intrusion

The False Positive Problem

- Generating an alert for normal traffic
- Database contains thousands of signatures
- Can't know what "normal" is on your network
- Default settings may be overly sensitive
- Too many false positives makes IDS useless

Network Monitoring System (NMS) False Positives

- An NMS keeps track of network traffic
- Generate polling and discovery activity
- Use SNMP protocol but may also use ICMP pings
- An NMS can generate thousands of alerts per hour on the IDS
- Set up IDS to ignore traffic to and from NMS

Network Scanning False Positives

- Nmap or Nessus can also set off the IDS
- Best solution is to shut down IDS when doing scanning
- Otherwise alert database will be skewed with false data

User Activity False Positives

- IM and P2P are flagged
- Decide whether or not to ban
- Either comment out flagging rules (leaving yourself exposed) or enforce bans

Application Software False Positives

- Microsoft Exchange behaves just like Nimda worm
- It's webmail interface copies over system files with .eml extension
- Either comment out flagging rules (leaving yourself exposed) or ban Exchange

Long Web Authentication Strings False Positives

- Using long login strings can be a buffer overflow attempt
- Can also be a web application cramming in a lot of information
- Fix your applications if you can

Database Authentication Activity False Positives

- IDS look for a lot of authentication activity on RDBMS
- Assumption is production databases shouldn't often authenticate
- Development databases can cause problems
- Disable alerts in development environments
- Proper IDS Configuration

Customize Settings for your LAN environment

- Shut off categories not relevant (e.g. Unix signatures in an all Microsoft environment)
- Shut off policies not relevant (e.g. IM or P2P as noted above)
- Exempt problematic hosts (e.g. NMS, development databases, admin machine)
- Reducing false positives does increase risks but makes IDS more useful

Constantly Tune

- Analyze alerts and look for more settings that generate false positives
- Get a feel for “normal” traffic on your network
- May take several months to fully tune

Analyze Data

- Logging data can be too easily ignored
- Email alerts can be overwhelming
- Best alternative is to use a database logging and analysis tool (e.g. ACID)

Main Problems with IDS

- False positive noise
- Enumerate badness fallacy
- Open barn door approach
- Don't overly rely on IDS

Anomalous Activity-Based IDS

- Attacks “enumerate badness” approach
- Monitors normal system activity
- Alerts for unusual activity that is outside the norm
- Takes a long time to learn what is “normal”
- Can be exploited by internal users who know network well

Intrusion Prevention Systems (IPS)

- Attacks “open barn door” approach
- Takes action as alerts are generated
- Write on-the-fly custom firewall rules
- Block IP addresses
- Integrate or counter-attack offending system
- Many contend just buzzword — most likely will migrate to firewall

Snort Hardware

- Snort should be run on a stand-alone machine
- Minimal install
- No X-windows or other services
- Several gigabytes of hard disk space in separate partition (in case log overflows)
- 500MHZ Intel machine is fine

NIDS Placement on LAN

- Place behind firewall so can see what firewall let's through
- Windows networking generates many alerts so needs to be tuned
- In switched environment can place in line between firewall and switch
- Can also place a hub between firewall and internal switch to avoid single point of failure
- Better option in larger networks is to mirror all ports to a monitor port (more expensive solution)
- More information <http://www.snort.org/docs/faq/1Q05/node9.html>

NIDS Placement on DMZ

- Servers that are most exposed
- Many alerts due to port scanning etc.
- Focus on application-specific alerts and not reconnaissance
- Switching issues same as on LAN

NIDS Placement Between Firewall and ISP

- Can see everything attacking both public servers and internal LAN
- Won't see internal traffic
- Same reconnaissance traffic issue as in DMZ
- Same switching issue as in DMZ