IDS / SNORT

PacNOG19

28th November - 2nd December 2016

Nadi, Fiji

Issue Date: [31-12-2015]

Revision: [v.1]

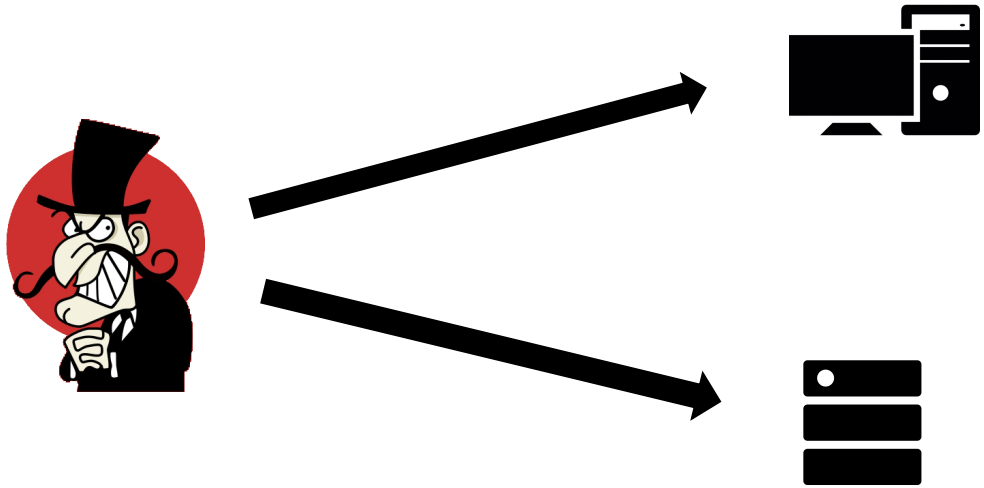
APNIC

Sometimes, Defenses Fail

- Our defenses aren't perfect
  - Patches weren't applied promptly enough
  - Antivirus signatures not up to date
  - 0-days get through
  - Someone brings in an infected USB drive
  - An insider misbehaves
- Now what?
- Most penetrations are never detected
  - This allows continuing abuse, and helps the attackers spread elsewhere

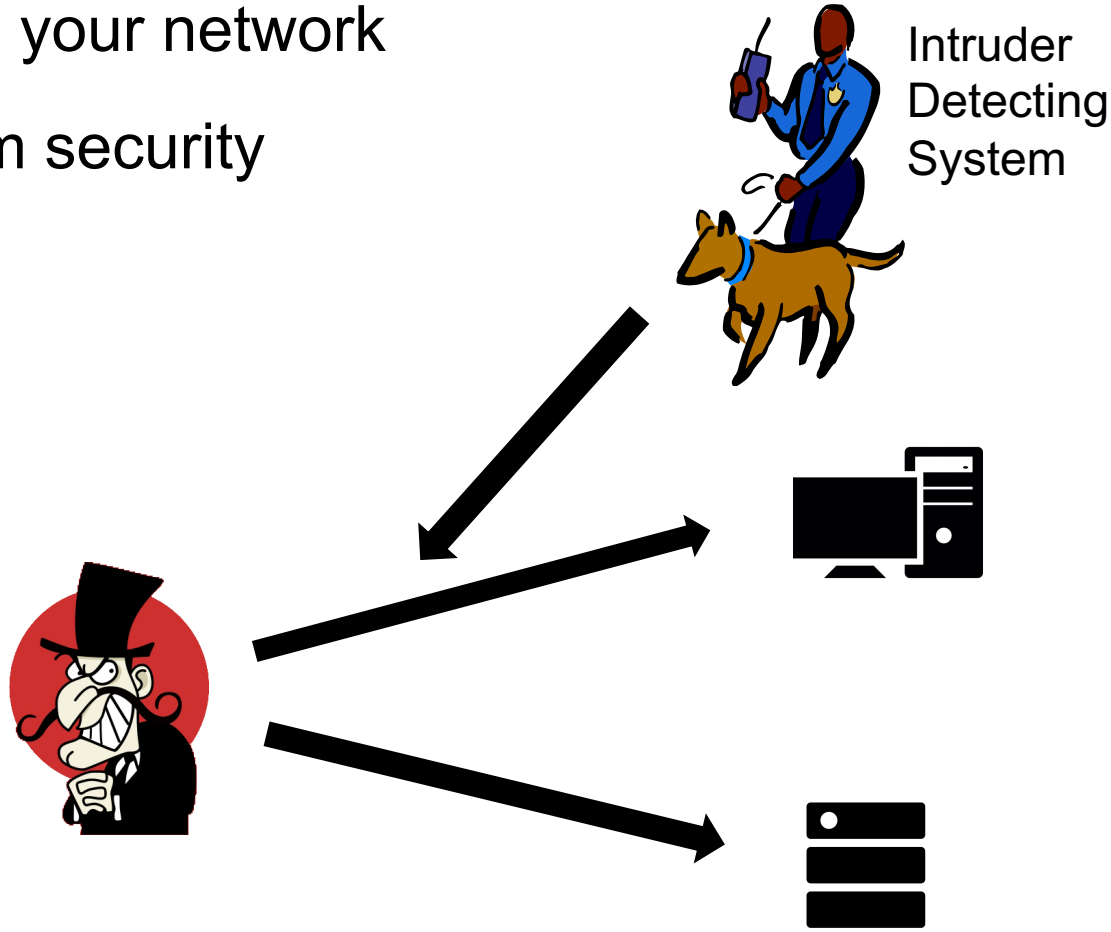
Unexpected Activity

•There could be an intruder even if you have security practice in place



Additional Monitoring

- Activity in your network
- To confirm security



What can IDS realistically do

- Detect successful attacks
- Look for various things that shouldn't be there
- Infected files
- Attacks on other machines
- Packets that shouldn't exist
- Strange patterns of behavior
- Contain attacks before they spread further
- Clean up penetrated machines—because you'll know they're infected
- Recognition of pattern reflecting known attacks
- Statistical analysis for abnormal activites

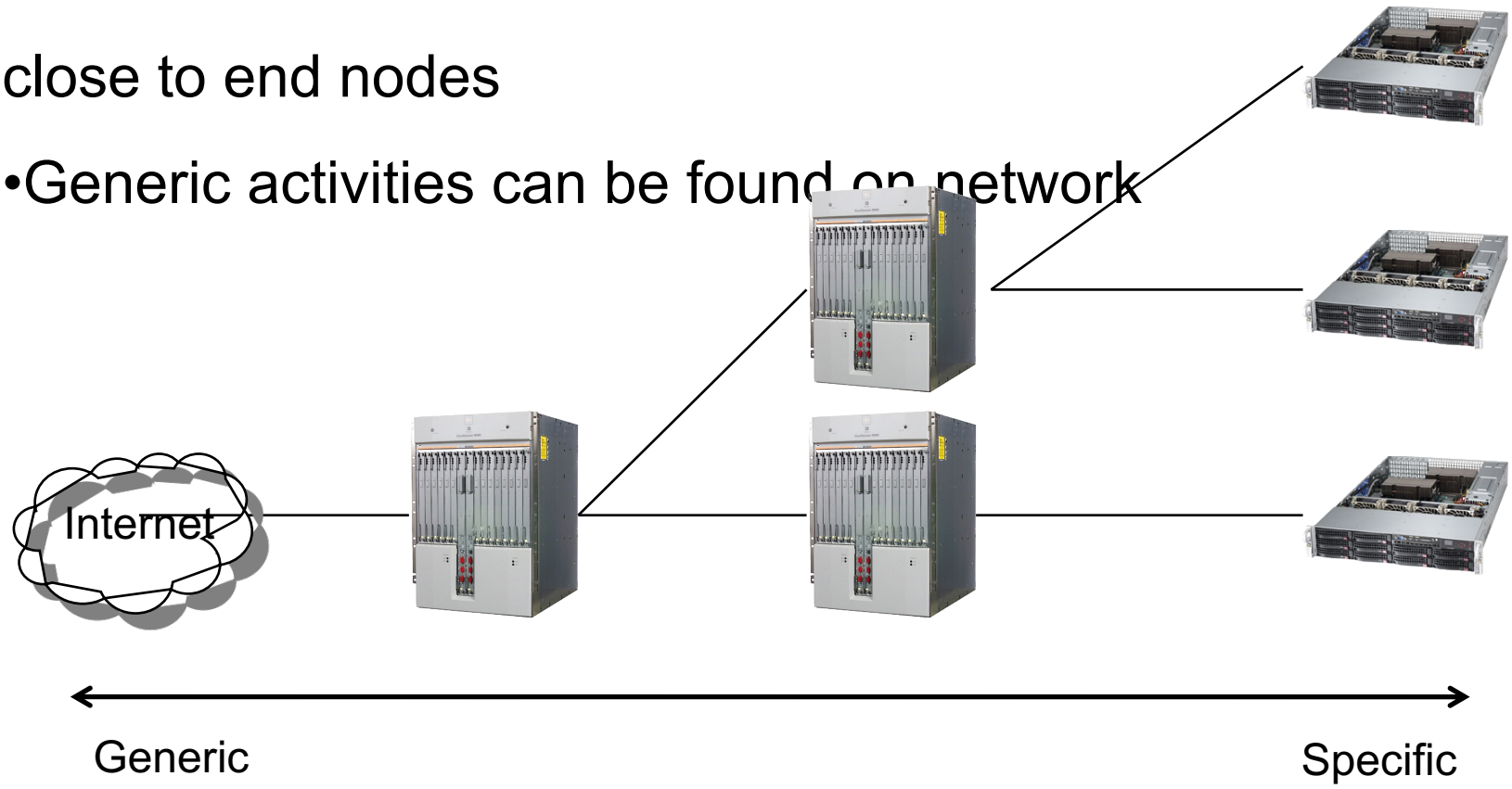
What IDS can’t do

- Compensate for weak authentication & identification mechanisms
- Investigate attacks without human intervention
- Guess the content of your organization security policy
- Compensate for weakness in networking protocols, for example IP Spoofing

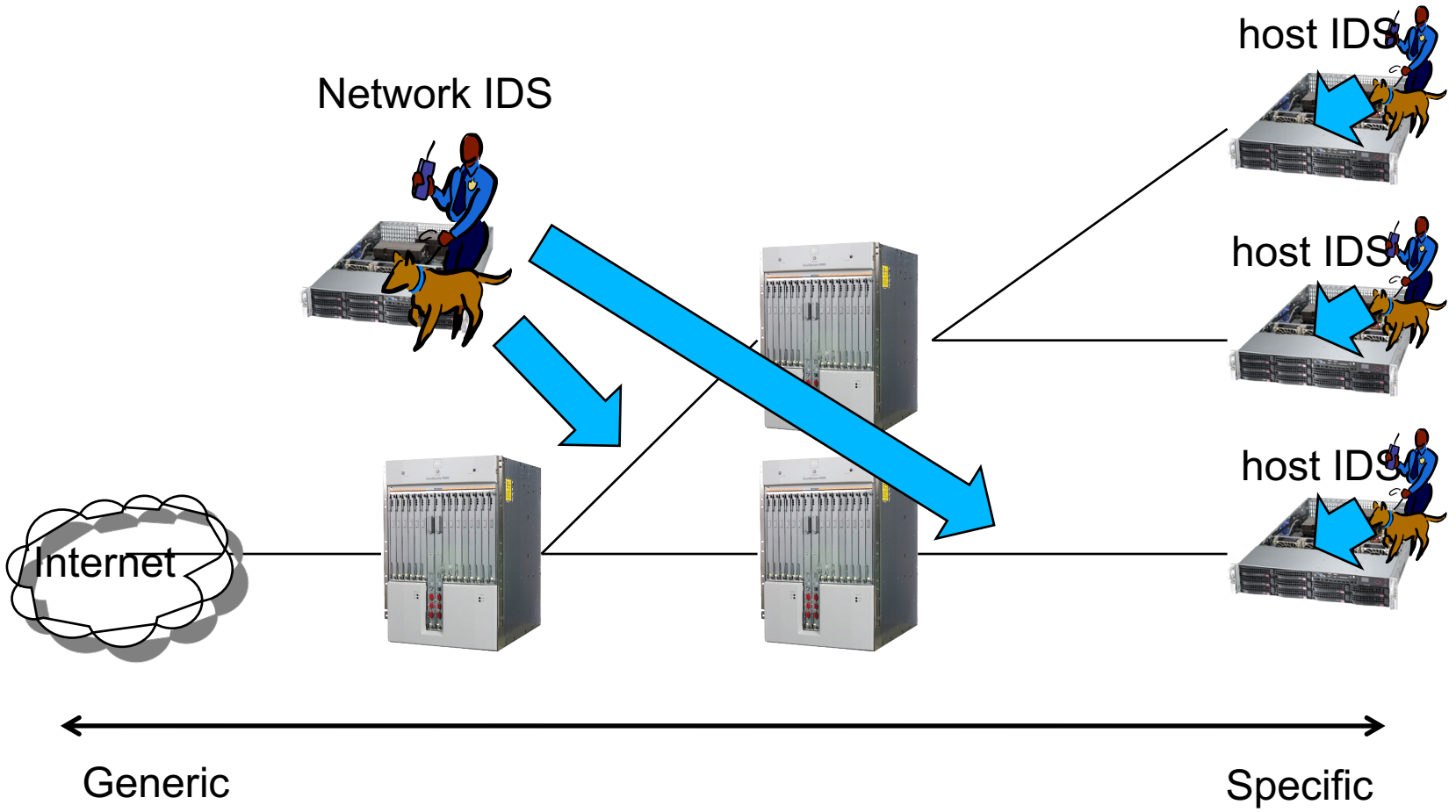
Monitoring Point

•More specific rules can be applied for a point close to end nodes

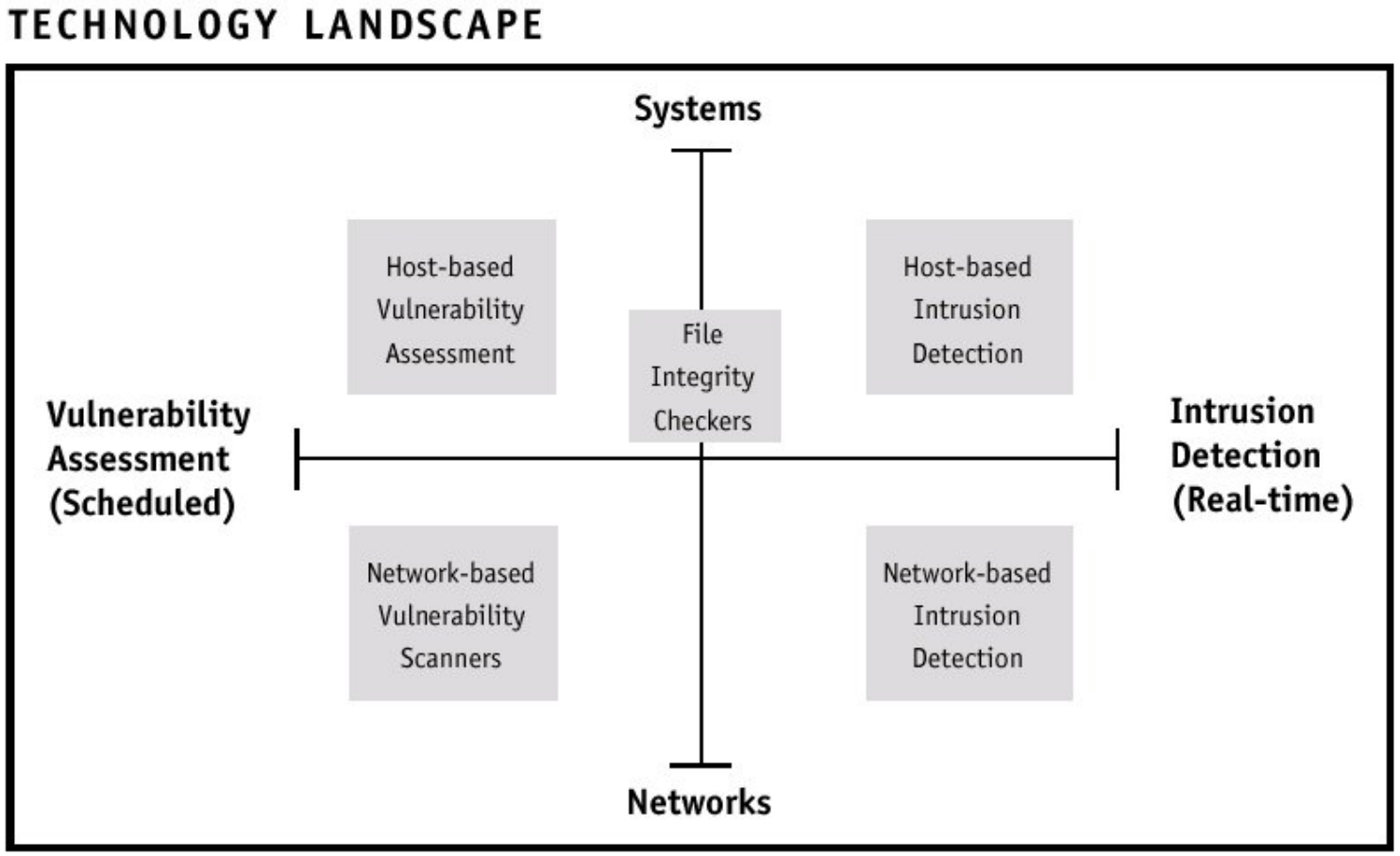
•Generic activities can be found on network



Network and Host IDS



IDS Technology landscape



Preventive

Real Time

Alert

- You may receive tons of millions of alerts
  - Depending on your detection rules
  - There are many suspicious activities in the Internet today
- You should notice a critical one at least
  - Detection rule is important!

Alert

- False Positive / Type I Error:
  - is the incorrect rejection of a true null hypothesis
  - is when a system raises an incorrect alert
- False Negative / Type II Error:
  - is the failure to reject a false null hypothesis
  - is when an attack pass undetected

Types of Detection

- Signature Based
  - –Match patterns against known attacks
  - –Catch the intrusions in terms of the characteristics of known attacks or system vulnerabilities
- Anomaly Based
  - –Look for unusual behavior
  - –Detect any action that significantly deviates from the normal behavior

Intrusion Detection for ISPs

- Monitor your own network—but that's no different than any other enterprise
- Monitor your customers
  - –Good: you can help them by detecting problems
  - –Good: you can prevent them from clogging your infrastructure
  - –Bad: it can be privacy-invasive

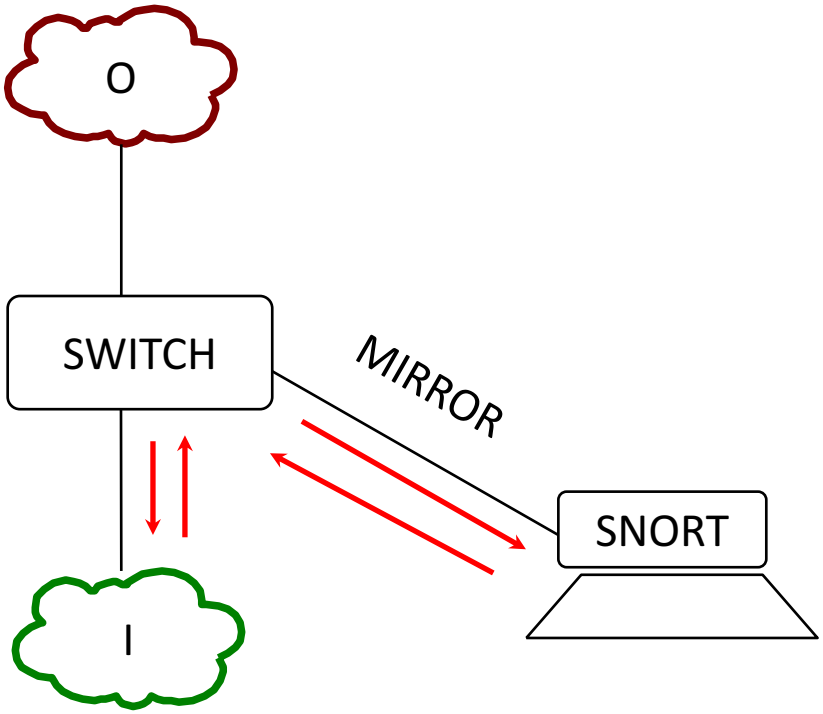
SNORT

- Snort is an open source IDS, and one of the oldest ones
- Hundreds of thousands of users
- Active development of rules by the community make Snort up to date, and often more so than commercial alternatives
- Snort is fast! It can run at Gbit/s rates with the right hardware and proper tuning

Getting Snort to see the network

- You could run Snort in multiple ways
  - –As a device “in line” behind or after the firewall/router
    - But this adds one more element that can fail in your connectivity
  - –Or you could use a span/mirror port to send traffic to Snort
  - –Or you can use an “optical splitter” to “mirror” or “tap into” traffic from a fiber optic link
    - This method and the previous are the most recommended

Getting Snort to see the network



Getting Snort to see the network

•Be careful not to overload your switch port – If you mirror a gigabit port to another gigabit port, the monitoring port (the receiving port) can drop packets if the total traffic exceeds 1 Gbit/s

Monitoring Port…

- On Cisco Catalyst, this is a “SPAN” port
- You can SPAN one port to another, a group of ports to one port, or an entire VLAN to a port
- Sample config:

```
interface FastEthernet 0/1
# port monitor FastEthernet 0/2
```

- This would copy any packet received on F0/2 to F0/1

Snort configuration file

- By default, /etc/snort/snort.conf
- It's a long file – 900+ lines
- If you browse it, you will notice many "preprocessor" entries
- Snort has a number of "preprocessors" which will analyze the network traffic and possibly clean it up before passing it to the rules

SNORT Rules

- Snort rules are plain text files
- Adding new rules to snort is as simple as dropping the files into /etc/snort/rules/
- Groups of rules can be loaded from snort.conf using the “include” statement
- Rules can match anything
- Technical – web attacks, buffer overflow, portscan, etc…
- Policy/user oriented – URL filtering, keyword, forbidden applications, etc…

Tailoring the rules

- Not all rules will make sense in your network
- You will want to customize which rules you want to run
- Otherwise you will get many false positives, which will lead you to ignore Snort, or simply turn it of…
- It doesn't help to have logs full of junk alerts you don't want
- To avoid this, rules can be suppressed (disabled)

Updating Snort rules

•The commercially maintained snort rules are available for free with a 30 day delay from http://www.snort.org/start/rules

•Other rules are maintained by some volunteers at emerging threats: http://rules.emergingthreats.net/open/

•The updating of rules can be automated with a tool called “Pulled Pork”, which is located at http://code.google.com/p/pulledpork/

Snort rules

- Snort rules are divided into two logical sections:
  - **Rule Header** : The rule header contains the rule's action, protocol, source and destination IP addresses and netmasks, and the source and destination ports information.
  - **Rule Options** : The rule option section contains alert messages and information on which parts of the packet should be inspected to determine if the rule action should be taken.

Snort rules

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 22
(msg: "SSH Detected"; sid:10; rev:1;)
```

The text up to the first parenthesis is the rule header and the section enclosed in parenthesis contains the rule options. The words before the colons in the rule options section are called option *keywords*.

Snort rules header

•alert - generate an alert using the selected alert method, and then log the packet

•log - log the packet

•pass - ignore the packet

•activate - alert and then turn on another dynamic rule

•dynamic - remain idle until activated by an activate rule , then act as a log rule

•drop - block and log the packet

•reject - block the packet, log it, and then send a TCP reset if the protocol is TCP or an ICMP port unreachable message if the protocol is UDP.

•sdrop - block the packet but do not log it.

Snort rules : The Direction Operator

- The direction operator **->** indicates the orientation, or direction, of the traffic that the rule applies to.
- There is no **<-** operator.
- Bidirectional operator **<>**

Snort rules : sid

- The sid keyword is used to add a “Snort ID” to rules
  - –Range 0-99 is reserved for future use
  - –Range 100-1,000,000 is reserved for rules that come with Snort distribution
  - –All numbers above 1,000,000 can be used for local rules

Snort rules : classtype

- Rules can be assigned classifications and priority numbers to group and distinguish them
  - /etc/snort/classification.config

```
config classification: DoS,Denial of Service Attack,2
```

Name Description Priority

- You can distinguish between high- and low-risk alerts

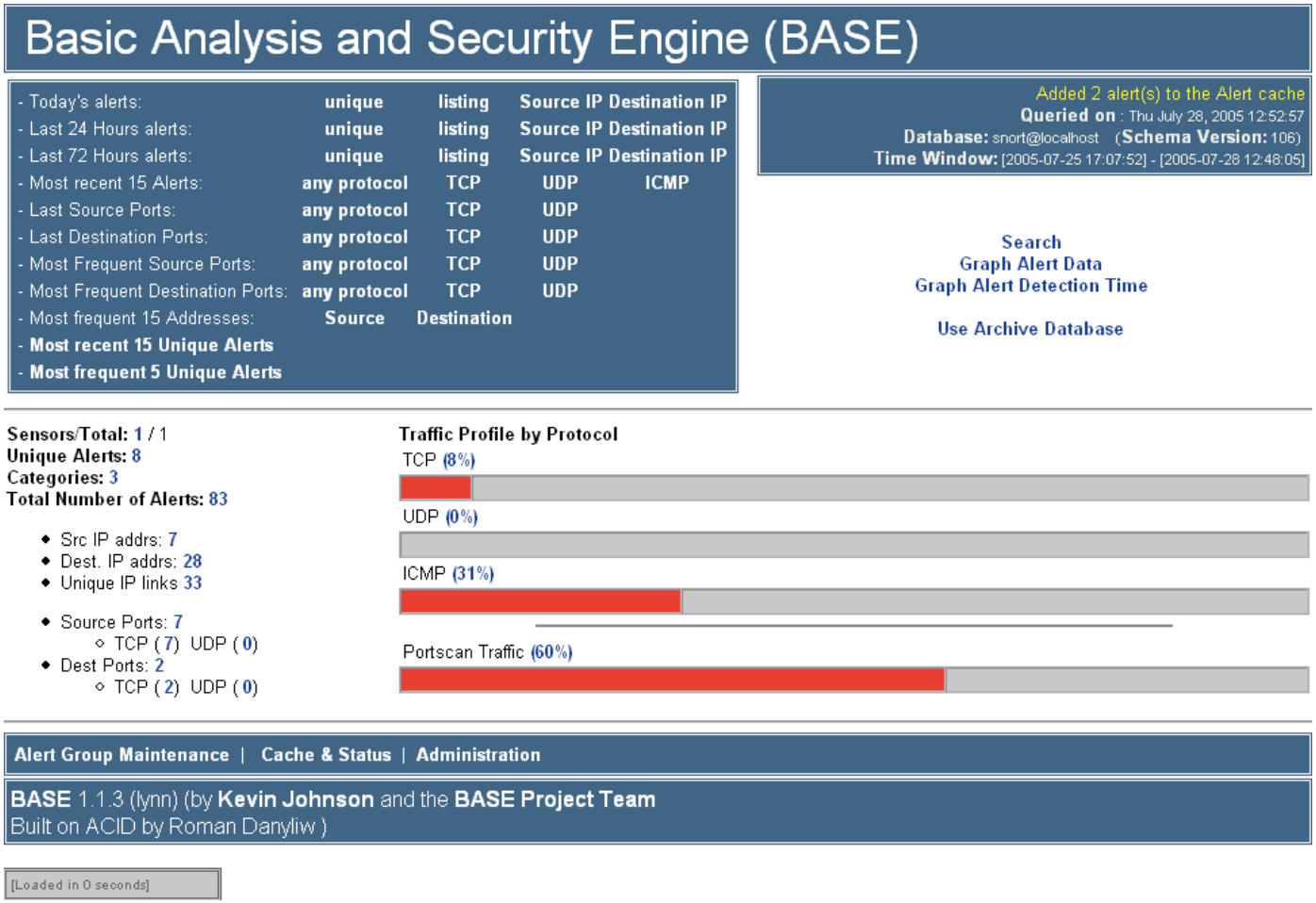
Sample rules

```
alert tcp msg:"MYSQL root login attempt";
flow:to_server,established; content:"|0A 00 00 01 85 04 00 00
80|root|00|"; classtype:protocol-command-decode; sid:1775;
rev:2;)

alert tcp $EXTERNAL_NET any -> $SQL_SERVERS 3306 (msg:"MYSQL
show databases attempt"; flow:to_server,established;
content:"|0F 00 00 00 03|show databases"; classtype:protocol-
command-decode; sid:1776; rev:2;)

alert tcp $EXTERNAL_NET any -> $SQL_SERVERS 3306 (msg:"MYSQL
4.0 root login attempt"; flow:to_server,established;
content:"|01|"; within:1; distance:3; content:"root|00|";
within:5; distance:5; nocase; classtype:protocol-command-
decode; sid:3456; rev:2;)
```

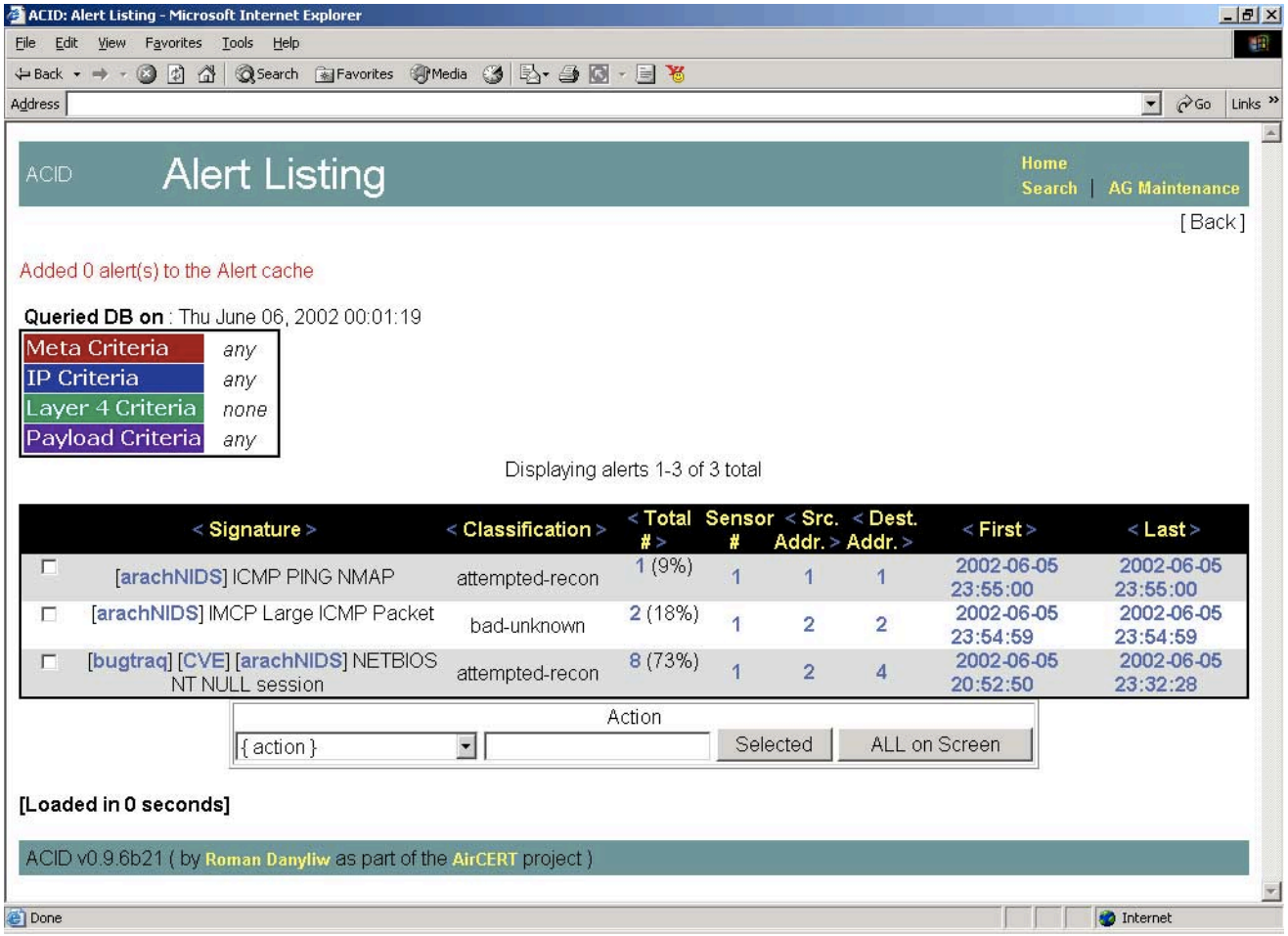
Reporting and logging

- Snort can be made to log alerts to an SQL database, for easier searching
- A web front-end for Snort, BASE, allows one to browse security alerts graphically

BASE (Basic Analysis and Security Engine)



BASE (Basic Analysis and Security Engine)



References and documentation

- Snort preprocessors:
  - http://www.informit.com/articles/article.aspx?p=101148&seqNum=2
- Snort documentation
  - http://www.snort.org/docs
- An install guide for Ubuntu 10.04:
  - http://www.snort.org/assets/158/014-snortinstallguide292.pdf
- Writing SNORT Rules
  - http://manual.snort.org/node27.html

Exercise

SNORT Setup

•Follow lab manual to install SNORT and check the basic SNORT rules.

Exercise : 1

- Write a rules to check XMAS scan on your server
  - Clue XMAS scan sets the FIN, PSH, and URG flags
  - Check the rules with nmap
    - `nmap -sX SERVER_IP`

Exercise : 2

- Write a rules to check any external network access your webserver /admin pages
  - Match content

Exercise : 3

- Write a rules to check SSH brute force attack and log IP trying to connect more than 3 times in 60 seconds.
  - –threshold:type threshold, track by_src, count 3, seconds 60;