What I am going to present:
- Review: What are NfSen and nfdump.
- The Tools in Action.
- Plugins.
- Outlook - what’s next.
**What is NetFlow?**

*NetFlow is a traffic monitoring technology developed by Cisco Networks. Flows are unidirectional and contain connection related data such as:*

- Source and destination IP address.
- Source and destination port.
- Source and destination AS.
- Level3 protocol, ToS byte, TCP flags.
- Logical input and output interfaces.
- Bytes and packet counters.

Netflow records never contain any user data!

**With NfSen you can:**

- Display the network traffic situation.
- Easily navigate through the netflow data. (time based)
- Drill down from overview to the details down to the specific flows.
- Profile/monitor specific Networks/Hosts and events.
- Extensively filtering netflow data.
- Analyse the netflow data using the web based as well as the command line based interface.
- Create lots of Top N statistics.
- Post process the netflow data for reporting and alerting.
nfdump 1.4: Main Netflow Processing Tool:
- Stores netflow data in time sliced files.
- CMD line based tool comparable to tcpdump.
- Written in C ⇒ fast.
- Supports netflow format v5 and v7.
- Powerful pcap like filter syntax:
  ‘( tcp and dst net 172.16/16 and src port > 1024 and bytes < 600 ) or ( bps > 1k and …’
- Flexible aggregation.
- Efficient filter engine: > 6 Mio flows/s on 3GHz Intel.
- Lots of fast statistics Top N
- Anonymizing of IP addresses. (Crypto-Pan)
### Nfdump: Statistics Processing:

<table>
<thead>
<tr>
<th>Top N statistics about</th>
<th>Ordered by:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Flow records</td>
<td>• Number of Flows</td>
</tr>
<tr>
<td>• Any IP addr, src IP addr, dst IP addr</td>
<td>• Number of Packets</td>
</tr>
<tr>
<td>• Any Port, src Port, dst Port</td>
<td>• Number of Bytes</td>
</tr>
<tr>
<td>• Any AS, src as, dst as</td>
<td>• pps (packets per second)</td>
</tr>
<tr>
<td></td>
<td>• bps (bits per second)</td>
</tr>
<tr>
<td></td>
<td>• bpp (bytes per packet)</td>
</tr>
</tbody>
</table>

#### Example:

```
nfdump -r ... -s record/flows -s ip/packets/bytes/bps -s dstport/flows
```

---

### Nfdump: Statistics Processing:

<table>
<thead>
<tr>
<th>Top 20 IP Addr ordered by bps:</th>
<th>Flow records</th>
<th>Packets</th>
<th>Bytes</th>
<th>pps</th>
<th>bps</th>
<th>bpp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date first seen</td>
<td>Duration</td>
<td>IP Addr</td>
<td>Flows</td>
<td>Packets</td>
<td>Bytes</td>
<td>pps</td>
</tr>
<tr>
<td>2005-09-06 21:56:02.973</td>
<td>146.122.12.8</td>
<td>111.122.22.8</td>
<td>706</td>
<td>2.3 M</td>
<td>2.4 G</td>
<td>10135</td>
</tr>
<tr>
<td>2005-09-06 21:56:11.730</td>
<td>178.622</td>
<td>126.122.17.9</td>
<td>706</td>
<td>2.3 M</td>
<td>2.4 G</td>
<td>13498</td>
</tr>
<tr>
<td>2005-09-06 21:56:15.245</td>
<td>150.885</td>
<td>124.103.24.244</td>
<td>767</td>
<td>2.0 M</td>
<td>2.0 G</td>
<td>13974</td>
</tr>
<tr>
<td>2005-09-06 21:54:10.936</td>
<td>280.895</td>
<td>126.122.41.11</td>
<td>679</td>
<td>2.5 M</td>
<td>2.6 G</td>
<td>9511</td>
</tr>
<tr>
<td>2005-09-06 21:55:07.622</td>
<td>0.004</td>
<td>37.35.251.55</td>
<td>70</td>
<td>70</td>
<td>33180</td>
<td>17499</td>
</tr>
<tr>
<td>2005-09-06 21:53:28.265</td>
<td>387.974</td>
<td>65.118.28.242</td>
<td>939</td>
<td>2.7 M</td>
<td>2.8 G</td>
<td>7389</td>
</tr>
<tr>
<td>2005-09-06 21:58:35.484</td>
<td>0.004</td>
<td>34.195.119.50</td>
<td>67</td>
<td>67</td>
<td>31500</td>
<td>16749</td>
</tr>
<tr>
<td>2005-09-06 21:54:36.207</td>
<td>291.392</td>
<td>121.100.232.10</td>
<td>725</td>
<td>2.1 M</td>
<td>2.1 G</td>
<td>7476</td>
</tr>
<tr>
<td>2005-09-06 21:56:20.489</td>
<td>0.001</td>
<td>107.38.46.178</td>
<td>2</td>
<td>9</td>
<td>6646</td>
<td>8999</td>
</tr>
<tr>
<td>2005-09-06 21:54:41.364</td>
<td>312.061</td>
<td>193.110.8.220</td>
<td>164</td>
<td>1.6 M</td>
<td>1.5 G</td>
<td>5214</td>
</tr>
<tr>
<td>2005-09-06 21:52:15.122</td>
<td>459.647</td>
<td>190.122.25.243</td>
<td>497</td>
<td>2.0 M</td>
<td>2.1 G</td>
<td>4627</td>
</tr>
<tr>
<td>2005-09-06 21:54:10.936</td>
<td>342.076</td>
<td>69.80.133.221</td>
<td>182</td>
<td>1.6 M</td>
<td>1.6 G</td>
<td>4903</td>
</tr>
<tr>
<td>2005-09-06 21:54:49.170</td>
<td>297.345</td>
<td>193.84.200.191</td>
<td>226</td>
<td>1.3 M</td>
<td>1.3 G</td>
<td>4691</td>
</tr>
<tr>
<td>2005-09-06 21:54:40.708</td>
<td>314.428</td>
<td>139.148.122.209</td>
<td>339</td>
<td>1.4 M</td>
<td>1.4 G</td>
<td>4695</td>
</tr>
<tr>
<td>2005-09-06 21:59:05.744</td>
<td>0.897</td>
<td>215.5.42.34</td>
<td>2</td>
<td>0.0 M</td>
<td>0.9 M</td>
<td>4521</td>
</tr>
<tr>
<td>2005-09-06 21:54:47.764</td>
<td>284.158</td>
<td>69.111.200.217</td>
<td>226</td>
<td>1.2 M</td>
<td>1.1 G</td>
<td>4321</td>
</tr>
<tr>
<td>2005-09-06 21:54:41.235</td>
<td>285.182</td>
<td>49.103.221.218</td>
<td>220</td>
<td>1.1 M</td>
<td>1.0 G</td>
<td>3897</td>
</tr>
<tr>
<td>2005-09-06 21:57:44.271</td>
<td>6.404</td>
<td>49.110.199.199</td>
<td>24</td>
<td>23995</td>
<td>23.5 M</td>
<td>3764</td>
</tr>
<tr>
<td>2005-09-06 21:57:00.563</td>
<td>4.863</td>
<td>49.75.122.44</td>
<td>20</td>
<td>17912</td>
<td>16.7 M</td>
<td>3683</td>
</tr>
</tbody>
</table>

**IP addresses are anonymized**


Flows analyzed: 1307442 matched: 1307442, Bytes read: 63839112
NfSen in Action

Overview - Details - Flows

Overview Profile: live
NfSen in Action

Overview - Details - Flows

Profiles:
- A profile is a specific view on the netflow data with nfdump filters applied.
- The profile applies to the graphical as well as to the numerical view.
- Profiles can be created from data in the past. (static)
- Profiles can be created from incoming data (continuous)
- Any views or processing options are available.

Profiles:
- A profile is a specific view on the netflow data with nfdump filters applied.
- The profile applies to the graphical as well as to the numerical view.
- Profiles can be created from data in the past. (static)
- Profiles can be created from incoming data (continuous)
- Any views or processing options are available.

Profiles:
- A profile is a specific view on the netflow data with nfdump filters applied.
- The profile applies to the graphical as well as to the numerical view.
- Profiles can be created from data in the past. (static)
- Profiles can be created from incoming data (continuous)
- Any views or processing options are available.

Profiles:
- A profile is a specific view on the netflow data with nfdump filters applied.
- The profile applies to the graphical as well as to the numerical view.
- Profiles can be created from data in the past. (static)
- Profiles can be created from incoming data (continuous)
- Any views or processing options are available.

Profiles:
- A profile is a specific view on the netflow data with nfdump filters applied.
- The profile applies to the graphical as well as to the numerical view.
- Profiles can be created from data in the past. (static)
- Profiles can be created from incoming data (continuous)
- Any views or processing options are available.

Profiles:
- A profile is a specific view on the netflow data with nfdump filters applied.
- The profile applies to the graphical as well as to the numerical view.
- Profiles can be created from data in the past. (static)
- Profiles can be created from incoming data (continuous)
- Any views or processing options are available.
Example Profiles:

Filter: ‘( udp or tcp ) and port 53’

Filter: ‘bytes < 100’

Filters may be as complex as the filter syntax of nfdump allows.
Example: ‘((src net 172.16/16 and src port > 1024 ) or dst host 192.168.16.17 and dst port 80) and packets > 1000 and pps > 150’
NfSen/nfdump - Plugins

Plugins - what for?
• Monitoring and alerting.
• Track for known botnet masters and send notifications.
• Track for possible scanners or DoS attacks, not necessarily visible in the graph.
• Port Tracking.

Backend Plugins are:
• Simple Perl modules hooked into the NfSen backend.
• Automatically called at regular 5 Min intervals.

Frontend Plugins are:
• Simple PHP modules hooked into NfSen frontend.
• Called by selecting the tab.

NfSen/nfdump

NfSen Plugins:

Web Front-end
Frontend Plugins

Post Processing
Backend Plugins

Periodic Update

2005 © SWITCH
Example Candidates for scanning activities:

```perl
# Define a nice filter:
# We like to see flows containing more than xxx packets
my $limit = 6000;
my $nf_filter = 'duration < 3500 and packets < 3 and bps < 100 and src as 559';

# Periodic function
# input: profilename
# timeslot. Format yyyyMMddHHMM e.g. 200503012000
sub run {
    my $profile = shift;
    my $timeslot = shift;

    syslog('debug', 'CatchScanners run: Profile: $profile, Time: $timeslot');

    my $profileinfo = NfSen::ReadProfile($profile);
    my $netflow_sources = $profileinfo->{'source_list'};

    # process all sources of this profile at once
    my @output = `$nfdump -M PROFILES_DIR/$profile/$netflow_sources -r nfcapd.$timeslot -a -H tcp -l $limit -f $nf_filter`;

    # Process the output and notify the duty team
}
```

(IP addresses anonymised)
NfSen/nfdump

Example Candidates for scanning activities:
The plugin processes data with nfdump arguments and filter:

```
Packet limit: > 6000 packets
Date flow start Duration Proto Src IP Addr:Port Dst IP Addr:Port Packets Bytes Flows
2005-09-06 16:29:55.639 297.585 TCP 182.115.108.131:0 -> 0.0.0.0:135 6531 313488 6531
2005-09-06 16:29:55.638 297.495 TCP 188.149.88.218:0 -> 0.0.0.0:135 6523 313104 6523
2005-09-06 16:29:27.096 328.073 UDP 254.115.137.234:0 -> 0.0.0.0:123 71827 5.2 M 71369
2005-09-06 16:29:23.958 328.445 TCP 188.171.59.109:0 -> 0.0.0.0:445 15693 746619 9477
2005-09-06 16:32:11.059 149.213 TCP 105.113.32.55:0 -> 0.0.0.0:445 12930 620640 6706
2005-09-06 16:32:10.660 141.883 TCP 195.238.53.231:0 -> 0.0.0.0:3306 114891 5.3 M 58328
2005-09-06 16:29:24.868 330.240 TCP 32.2.148.33:0 -> 0.0.0.0:1080 40174 1.8 M 33093
2005-09-06 16:29:25.126 92.930 TCP 161.194.253.63:0 -> 0.0.0.0:5900 16408 787584 8212
2005-09-06 16:29:26.085 141.633 TCP 236.65.30.98:0 -> 0.0.0.0:445 6976 334848 3488
2005-09-06 16:29:23.641 73.978 TCP 181.100.194.207:0 -> 0.0.0.0:4000 11489 551472 6075
2005-09-06 16:29:23.897 324.474 TCP 109.116.193.215:0 -> 0.0.0.0:445 16452 789676 8337
2005-09-06 16:29:23.704 161.342 TCP 110.77.48.188:0 -> 0.0.0.0:80 3640 654720 6821
2005-09-06 16:29:23.641 300.476 TCP 109.116.145.116:0 -> 0.0.0.0:445 6219 298512 3115
2005-09-06 16:29:25.622 297.916 TCP 109.116.193.215:0 -> 0.0.0.0:139 9715 440400 4604
2005-09-06 16:29:26.007 326.525 TCP 170.122.25.137:0 -> 0.0.0.0:6129 6849 3.1 M 47258
2005-09-06 16:29:26.328 259.071 TCP 109.116.226.66:0 -> 0.0.0.0:445 6835 328076 3446
2005-09-06 16:29:28.742 321.102 TCP 188.171.111.125:0 -> 0.0.0.0:445 13714 658234 7006
```

(IP addresses anonymised)

2005 © SWITCH

NfSen/nfdump

Backend / Frontend Plugins:

```perl
# package CatchDos;
use strict;
#
# sub Init {  
#   # Init plugin
# }  # End of Init

sub run {  
  my $profile = shift;
  my $timeout = shift;
  }  # End of run

nfsen.conf

register

output

Runs automatically every 5 min

Backend Plugin

Frontend Plugin

/* Port Tracker */

Plugin

Function

PortTracker_Run($plugin) {
  $plugin = GetTopN($plugin_id, 
  _SESSION['$plugin_id'])

} // End of PortTracker_Run
```

2005 © SWITCH
**NfSen Plugins: Port Tracker**

**Planned Plugin: Host behaviour based worm detection:**
Result of a PhD network security research work in the context of the DDoSVax project at Swiss Federal Institute of Technology Zurich: [http://www.tik.ee.ethz.ch/~ddosvax/](http://www.tik.ee.ethz.ch/~ddosvax/)

*Ideas:*
- **Infected hosts show a different behaviour and can be put into different classes:**
  - **Traffic** class:
    Worm infected hosts tend to send considerably more traffic than they receive.
  - **Responder** class:
    If many more hosts start to answer requests, they probably are victims of a worm.
  - **Connector** class:
    Worm infected hosts typically open many connections.
Host behaviour based worm detection:

Example: MyDoom.A

Most interesting for worm detection are cardinalities of class combinations.

Figures @ SWITCH:
- Server: 2 x 3GHz 2GB Ram. Debian Linux Kernel 2.6.10
- 3TB (2TB + 1TB) AXUS Disk Raid
- XFS file system.
- Gigabit Ethernet interfaces.
- 5min workload avg. ca. 5%.
- 25GB Netflow data / day.
- About 41 days of netflow data available.
NfSen/nfdump

Next Steps - Todo list:
NfSen:
• More Plugins:
  – Alerting Plugin.
  – Anomaly Detection Plugin. (DDoS VAX Project ETH Zürich)
  – Portocol, AS Tracker.
  – ....

nfdump:
• Netflow v9 IPv6
• Related filters: ‘Worm Footprint Tracking’
  first { dst ip <A> dst port 445 bytes > 600 }
  then { src ip <A> and dst ip 172.16.17.18 and dst port 80 }
• Post Processing Filters ... etc.
• ...

Summary:
• Good and flexible tools for all sort of netflow tasks.
  – Network monitoring.
  – Incident Handling.
  – All sort of tracking ...
• Open Source Tools under BSD License.
• Cmd line tool: nfdump
  – Written in C. Runs on most *nix.
    Tested on Linux Kernel 2.4.*, and 2.6.*, FreeBSD, OpenBSD, Solaris.
  – Available at http://nfdump.sourceforge.net
• Web based frontend: NfSen
  – Written in PHP and Perl.
  – Extendable using plugins.
  – Available at http://nfsen.sourceforge.net
• Possible candidate for the toolset in GN2/JRA2.
Thank you for your attention.
Any Questions?