Automatic anomaly detection using NfSen

Wim Biemolt, SURFnet

Werner Schram, SURFnet
Automatic anomaly detection using NfSen

- SURFnet and netflow anomaly detection
  - NERD
  - NfSen
  - PeakFlow SP
- Currently used detection methods
  - DDos
  - Botnet
  - Holt-Winters aberrant behavior
SURFnet and netflow anomaly detection

- NERD v1
  - Developed by TNO
  - Based on cflowd
  - cflowd is no longer supported

- NERD v2
  - Initially developed by TNO
  - Has serious performance problems
  - NfSen can do the same but without the performance problems
- Netflow Sensor (NfSen) is a
  - network statistics tool
  - Developed by Peter Haag
  - Currently in active development
  - Alert plug-in system
  - Generic plug-in system
  - Some plug-ins already available
Overview Profile: live, Group: (nogroup)
DDoS detection

- Simple flow analysis
  - based on NERD v1 DDoS detection
  - using a low threshold and a high threshold
  - Rules for traffic between those thresholds
  - Custom thresholds for high load services
Expected traffic
Definitely Conspicuous Traffic
Border cases
High load servers
Custom thresholds

SURFnet – Automatic anomaly detection using NfSen
The DDoS alarms between 2007-12-07 and 2007-12-15

<table>
<thead>
<tr>
<th>ID</th>
<th>Destination</th>
<th>Flows per 5 minutes</th>
<th>Average packets/flow</th>
<th>Average bytes/flow</th>
<th>Starttime</th>
<th>Stoptime</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>#50598</td>
<td></td>
<td>7772</td>
<td>5054</td>
<td></td>
<td>4 2007-12-14 08:55:00</td>
<td>2007-12-14 16:32:50</td>
<td>1 Delete</td>
</tr>
<tr>
<td>#50596</td>
<td></td>
<td>10620</td>
<td>3859</td>
<td></td>
<td>4 2007-12-14 08:39:54</td>
<td>2007-12-14 16:32:50</td>
<td>1 Delete</td>
</tr>
<tr>
<td>#50594</td>
<td></td>
<td>9510</td>
<td>3147</td>
<td></td>
<td>3 2007-12-14 08:25:01</td>
<td>2007-12-14 16:32:50</td>
<td>1 Delete</td>
</tr>
<tr>
<td>#50592</td>
<td></td>
<td>12951</td>
<td>129</td>
<td></td>
<td>2 2007-12-14 08:24:58</td>
<td>2007-12-14 16:32:50</td>
<td>1 Delete</td>
</tr>
<tr>
<td>#50490</td>
<td></td>
<td>9517</td>
<td>73</td>
<td></td>
<td>1 2007-12-13 06:13:41</td>
<td>2007-12-14 16:32:50</td>
<td>1 Delete</td>
</tr>
<tr>
<td>#49200</td>
<td></td>
<td>281618</td>
<td>163</td>
<td></td>
<td>1 2007-12-04 14:47:47</td>
<td>2007-12-14 16:32:50</td>
<td>1 Delete</td>
</tr>
<tr>
<td>#49074</td>
<td></td>
<td>22047</td>
<td>171</td>
<td></td>
<td>2 2007-11-26 13:32:20</td>
<td>2007-12-14 16:32:50</td>
<td>1 Delete</td>
</tr>
<tr>
<td>#50656</td>
<td></td>
<td>5222</td>
<td>2550</td>
<td></td>
<td>3 2007-12-14 16:20:07</td>
<td>2007-12-14 16:29:56</td>
<td>1 Delete</td>
</tr>
<tr>
<td>#50635</td>
<td></td>
<td>6031</td>
<td>1155</td>
<td></td>
<td>7 2007-12-14 11:44:53</td>
<td>2007-12-14 16:22:51</td>
<td>1 Delete</td>
</tr>
</tbody>
</table>
DDos interface: Details

Top 10 flows per 5 minutes at 2007-12-14 16:37:40:

<table>
<thead>
<tr>
<th>address</th>
<th>Bytes</th>
<th>port usage</th>
<th>last scan</th>
<th>actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1379</td>
<td>2947950</td>
<td>1046, mac: 65508</td>
<td>2007-12-14 12:37:51</td>
<td>Report port scan, analyse</td>
</tr>
<tr>
<td>1353</td>
<td>2897466</td>
<td>1038, mac: 65509</td>
<td>2007-12-14 12:53:00</td>
<td>Report port scan, analyse</td>
</tr>
<tr>
<td>1342</td>
<td>2963856</td>
<td>1071, mac: 65502</td>
<td>2007-12-14 13:03:01</td>
<td>Report port scan, analyse</td>
</tr>
<tr>
<td>1341</td>
<td>2997262</td>
<td>16971, mac: 56329</td>
<td>2007-12-14 13:17:59</td>
<td>Report port scan, analyse</td>
</tr>
</tbody>
</table>

SURFnet – Automatic anomaly detection using NfSen
Botnet detection

- Hosts infected by viruses connect to hosts known as botnet controllers
- List of botnet controllers are available, for example: [http://www.bleedingthreats.net/rules/bleeding-botcc.rules](http://www.bleedingthreats.net/rules/bleeding-botcc.rules)
- Our plug-in logs all hosts that connect to known botnet controllers
- Automatically reports to incident report system using IODEF
SURFnet – Automatic anomaly detection using NfSen

Botnet IODEF reports

<?xml version="1.0" encoding="iso-8859-1"?>
  <io:Incident purpose="reporting">
    <io:IncidentID name="overflow.surfnet.nl">#33408</io:IncidentID>
    <io:StartTime>2007-08-13T15:07:47+02:00</io:StartTime>
    <io:EndTime>2007-08-13T21:06:12+02:00</io:EndTime>
    <io:Assessment>
      <io:Impact type="user"/>
    </io:Assessment>
    <io:Contact>
      <io:ContactName>Werner Schram</io:ContactName>
    </io:Contact>
    <io:EventData>
      <io:Method>
        <io:Reference>
          <io:ReferenceName>botnet</io:ReferenceName>
        </io:Reference>
      </io:Method>
      <io:Flow>
        <io:System category="source">
          <io:Node>
            <io:Address category="ipv4-addr">192.168.1.1</io:Address>
            <io:Counter type="flow">20</io:Counter>
          </io:Node>
        </io:System>
        <io:System category="target">
          <io:Node>
            <io:Address category="ipv4-addr">192.168.1.2</io:Address>
          </io:Node>
          <io:Service ip_version="4" ip_protocol="6">
            <io:Port>80</io:Port>
          </io:Service>
        </io:System>
      </io:Flow>
    </io:EventData>
  </io:Incident>
</io:IODEF-Document>

SURFnet – Automatic anomaly detection using NfSen
Holt-Winters aberrant behavior detection

- Uses information about periodic data to predict aberrant behavior.
Holt-Winters: Example
Holt-Winters: Original implementation

- Trend
- Periodic information
- Noise

Prediction
Limitations of the original implementation

- The original algorithm has three parameters which define:
  - the weight of historical data
  - the weight of the trend
  - the amount of expected noise
- The original algorithm has a constant learning rate
  - If a low learning rate is used, the selection of the initial values is critical. This will introduce false positives for a long time.
  - With a high learning rate, the model will likely be overfitted. This will introduce false negatives
- The trend parameter has no significant influence with the resolution we are using
Network traffic time series often show multiple recurring patterns, for example a weekly trend:
Holt-Winters: Multiple periods

Daily Period

Weekly period

Noise
Learning rate

Fixed learning rate:
The first pattern is overweighted

Adaptive learning rate:
The weight of the first pattern is relative to the rest
Real data example
Holt Winters: Usage Example

Normal ICMP Traffic

Aberrant ICMP Traffic: Caused by DDoS attack by Stormworm botnet
Holt Winters: Other possible uses

Common SMTP Traffic

Last week SMTP Traffic