Scalable DDoS Protection

Rajesh Talpade
Director
Information Assurance Group
rtalpade@telcordia.com
+1 732 699 2832

An SAIC Company
High-level ISP Network Topology

ISP: Internet Service Provider
BR: Border Router
FW: Customer Firewall
SOC: Security Operations Center

ISP 1
ISP IP Backbone
ISP 2
ISP 3
Customer’s Server Farm
DDoS Attacks

Multiple DDoS tools exist on the WWW that can be trivially downloaded and installed on unprotected PCs for launching attacks.

With rapid spread of broadband and Internet access internationally, DDoS attacks are assuming epidemic proportions:
- CSI/FBI 2003 survey indicates DoS attacks moved up to 2nd place from 4th, and more than tripled in financial impact from 2002.

Traditional firewalls may stop DDoS attack, but customer’s network becomes inaccessible to even “legitimate” traffic since access link is over-loaded; firewall-based solution causes a denial of service!

Detection and trace-back difficult since source IP addresses are “spoofed”

Existing solutions (e.g. Arbor, Mazu, Captus, Riverhead) require use of specialized hardware placed along-side in-service routers, providing non-scalable and expensive solutions that are not applicable to typical broadband customer base.
DDoS Protection

- A scalable solution for automated detection, trace-back and mitigation of DDoS attacks that combines hardware-based COTS products with policy-based control software
  - Detect at edge and Mitigate in core
  - Based on UUNET’s CenterTrack approach
DDoS Attack

Original DDoS Traffic

Original non-DDoS Traffic

“DDoS Target”
Reaction after DDoS Detection

1. SNMP Trap
   1a. IDMEF Message
2. Install Local Route
3. BGP Route Propagation

“DDoS Target”
Mitigation of DDoS Attack

Non-DDoS Traffic reaches user; DDoS Traffic is filtered out by FE
Sensor

• Prototype software sensor is Linux-based traffic sniffer and analysis for Ethernet (10/100 mbps) interface
  – Provided by ISP to customer as software executable on host with visibility to all traffic in customer network
  – Standards-based IDMEF messages over XML communication between Sensor & Analysis Engine (AE)
  – Tracks information about current TCP, UDP, ICMP, IP packet and bit rate flowing into customer’s network from ISP’s network
  – Performs signature-based detection of both DDoS control traffic and flood traffic, and volume-based detection of DDoS flood traffic
    • Signatures can be updated similar to email virus approach; signatures currently available for Stacheldraht and TFN2K DDoS tools
    • Can handle previously unknown DDoS floods using volume-based detection with user-configurable thresholds

• We have also developed interface between AE and 3rd party network-based sensor (e.g. Arbor Networks’ peakFlow) to obtain DDoS detection information
Filtering Engine (FE)

• Combination of high-end router and packet filtering firewall
  – Could use separate router and firewall components for same effect
  – Prototype uses Cisco 3600 and firewall software

• FE maintains an IP-in-IP tunnel and an eBGP session with every BR

• InFilters (explained later) can be pre-provisioned at each incoming tunnel in FE

• Signature-based and valid-header DDoS flood filters are pre-provisioned for all traffic transiting FE
  – Valid-Header: Finite set of “legal” values for TCP/UDP/ICMP/IP headers

• FE is accessible for automated configuration from AE; necessary security mechanisms are used to protect this access

• Multiple FEs can be used depending on expected usage, without any requirements for inter-FE or inter-BR synchronization
**Analysis Engine (AE)**

- Prototype is Java-based analysis and reaction software
- Receives DDoS **control** traffic notification from software sensor
  - ISP may inform customer off-line about existence & identity of DDoS client/agent in their network
- Receives DDoS **flood** traffic from software sensor or Arbor sensor
  - AE automatically configures FE to advertise route for DDoS-ed customer network; FE uses eBGP to convey this info to BRs, which forward traffic destined for DDoS-ed customer over IP-in-IP tunnels to FE
  - Signature-based filters in FE drop all traffic that matches known DDoS signatures
  - Remaining DDoS traffic is dropped by InFilters and valid-header filters
  - Packet drop counter from these filters will be periodically polled by AE to determine the BRs, and hence peer ISPs, which are being used by DDoS flood
- Can modulate mitigation action based on current network defense posture
Key Features

- Automated DDoS detection, reporting & trace-back mechanisms
- Automated diversion & filtering of DDoS flood traffic destined for target
  - Minimal impact on “legitimate” traffic flow to DDoS target
- Highly scalable architecture, for handling multiple 10Ks of end-users, that can be incrementally deployed
- Low-cost of deployment
  - Uses COTS routers and firewalls, and IETF-standard protocols for DDoS reporting and traffic diversion
  - COTS router- and firewall-vendor independent
- High ISP network stability
  - No dynamic provisioning of in-service network routers is required
  - Border routers forwarding DDoS flood can be quickly identified without need for “logging-on”, permitting mitigation negotiations with peer service providers
  - No impact on traffic to other networks/hosts in service provider’s network
- Rapid updates
  - Signatures for newly discovered DDoS attack tools can be rapidly and scalably added to detectors and filters
- Policy-based modulation of DDoS flood mitigation
- Detection of both DDoS Control and Flood traffic
Demonstration Test-bed

DoS Source 1.2

BR1: 3600

CR: 2600

ER: 2600

Analysis Engine

ISP Customer Network “DDoS Target”

Subnet Prefix: 10.10

All Fast Ethernet Interfaces

Sensor 6.3

Firewall 8.1

FE: 3600

5/4

Subnet Prefix: 10.10

All Fast Ethernet Interfaces

Subnet Prefix: 10.10

All Fast Ethernet Interfaces
Virtual Interfaces & IP-in-IP Tunnels
Demonstration Screen-shots

Traffic to customer routed over “normal” path

Low (1 msec average) ping delay and 0% packet loss
DDoS Flood Attack

Very high (83 msec average) ping delay and 33% packet loss
DDoS Notification to AE

Customers Under Attack Details - Signatures (cumulative)

<table>
<thead>
<tr>
<th>Customer</th>
<th>Last Update</th>
<th>Ctrl SKILLZ</th>
<th>Ctrl FICKEN</th>
<th>Ctrl 3333</th>
<th>Ctrl SPOOF</th>
<th>Str TCP</th>
<th>Str UDP</th>
<th>Str ICMP</th>
<th>Tfn TCP</th>
<th>Tfn UDP</th>
</tr>
</thead>
<tbody>
<tr>
<td>30.10.6.2</td>
<td>Sep 23, 2002 11:38:37 AM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>160989</td>
</tr>
<tr>
<td>30.10.6.0</td>
<td>Sep 23, 2002 11:38:47 AM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Customers Under Attack Details - Throughputs (instantaneous)

<table>
<thead>
<tr>
<th>Customer</th>
<th>Last Update</th>
<th>Total pkts</th>
<th>Total bps</th>
<th>TCP pkts</th>
<th>TCP bps</th>
<th>UDP pkts</th>
<th>UDP bps</th>
<th>ICMP pkts</th>
<th>ICMP bps</th>
</tr>
</thead>
<tbody>
<tr>
<td>30.10.6.2</td>
<td>Sep 23, 2002 11:38:58 AM</td>
<td>4776</td>
<td>8302771</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30.10.6.0</td>
<td>Sep 23, 2002 11:38:57 AM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>48352</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Effect of DDoS Mitigation

Path to customer passes thru FE (150.150.8.2), which filters DDoS flood.

“legitimate” ping traffic reaches customer, and ping delay and loss restored to pre-DDoS levels.
Post-attack Processing

On manual “deletion” of customer networks from AE by ISP admin after DDoS attack, path from Peer1 to customer is restored to pre-DDoS state

```
Peer1:~ # traceroute MDSensor
trace route to MDSensor (30.10.6.2), 30 hops max, 40 byte packets
  1 BR1 (10.10.1.1) 1 ms 1 ms 1 ms
  2 CR (10.10.2.2) 3 ms 3 ms 2 ms
  3 ER (10.10.5.1) 5 ms 4 ms 5 ms
  4 MDSensor (30.10.6.2) 1 ms 1 ms 2 ms
```
Experimental Evaluation
AE Scalability

- Evaluate scalability of AE with number of sensors
- Sensor Driver emulates “N” number of sensors
  - Messages sent at sensor default of 1 message per 10 seconds
  - Messages are not halted after normal Filtering Engine (FE) mitigation
Some Results

- AE CPU usage linear increase up to 140 sensors; 100% over 140
  - Pentium3, 700 MHz, 128 MB RAM
- AE maximum throughput at 170 sensors; All AE interfaces OK until 170
  - Information displayed correctly at AE GUI (Visually checked)
  - Static routes installed correctly at FE (Traffic route checked)
  - Log files
- Actual sensors supportable by AE in a network deployment >> 170
  - Messages sent only to AE when DoS detected
  - CAIDA estimated 4000 DDoS attacks per week; 0.007 per second
  - Single prototype AE can support 24000 sensors (10% under DDoS attack)
Effectiveness of Signature- and Valid-Header-based Filtering

• False-Negatives
  – How many DDoS (TFN2K and Stacheldraht) packets make it thru the filter?
  – 3 runs of >1M packets each, for each of TCP, UDP and ICMP
  – TCP: 0, UDP: 50, ICMP: 0

• False-Positives
  – How many “non-DDoS” packets get caught in filter?
  – 90M packets collected from cable-modem and DSL networks
  – 2 TCP & 111 UDP packets caught by TFN2k & Stacheldraht filters
  – Few thousand ICMP packets caught; All had destination addresses from small set of 6, implying that these were packets from actual attacks in-progress at time of packet capture
What about new DDoS attacks?

- Focus on “spoofed” IP packets
  - Significant proportion of Internet-based attacks involve use of “spoofed” IP packets i.e. packets with incorrect source IP address
  - Use of spoofed packets provides anonymity to attacker, and reduces effectiveness of attack protection mechanisms
- Necessary to detect spoofed IP packets in large IP networks, and trace-back the spoofed IP packets to the administrative boundaries of these networks
**InFilter Hypothesis**

<table>
<thead>
<tr>
<th>Peer AS – BR</th>
<th>Expected IP Addresses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peer AS₁ – BR₁</td>
<td>IP_{A₁}, IP_{B₁}</td>
</tr>
<tr>
<td>Peer AS₂ – BR₂</td>
<td>IP_{A₂}, IP_{B₂}</td>
</tr>
<tr>
<td>Peer AS₄ – BR₂</td>
<td>IP_{A₄}, IP_{B₄}</td>
</tr>
<tr>
<td>Peer AS₃ – BR₃</td>
<td>IP_{A₃}, IP_{B₃}, IP_{C₃}</td>
</tr>
</tbody>
</table>

AS: Autonomous System  
BR: Border Router

- **Predictive Ingress Filtering (InFilter)**
  - Peer AS-BR pair (last AS hop) used by traffic from any IP address to enter “Target Network” remains relatively static.
  - Can associate “Expected IP Address” (EIA) set with each Peer AS-BR pair.
  - Consider only last hop (instead of entire path) from source to target.
Empirical Validation
Traceroute-based Validation

- Looking Glass (LG) sites permit traceroute from LG site to any Internet destination
- Periodic traceroute from LG sites to Target sites
  - 24 Looking Glass sites: 5 US, 19 international
  - 20 Target sites: IP address in large and medium sized ISP networks
# Looking Glass Sites and Targets

<table>
<thead>
<tr>
<th>24 Looking Glass Sites</th>
<th>20 Target Sites</th>
</tr>
</thead>
<tbody>
<tr>
<td>Australia (Telstra)</td>
<td>America Online (AOL) (Time Warner)</td>
</tr>
<tr>
<td>Brasil (Matrix Internet)</td>
<td>Earthlink (ELNK)</td>
</tr>
<tr>
<td>Canada (Tera Byte)</td>
<td>Prodigy (Acquired By SBC) (PRGY)</td>
</tr>
<tr>
<td>Columbia (AT&amp;T Columbia)</td>
<td>RCN (RCNC)</td>
</tr>
<tr>
<td>Denmark (Tele Danmark Internet Data Division)</td>
<td>Concentric (XOXO)</td>
</tr>
<tr>
<td>France (Oleane)</td>
<td>Global Crossing (GBLX)</td>
</tr>
<tr>
<td>Germany (TeliaNet Germany)</td>
<td>Broadwing Inc (BRW)</td>
</tr>
<tr>
<td>Greece (National Technical University of Athens)</td>
<td>McleodUSA, Inc (MCLD)</td>
</tr>
<tr>
<td>Hong Kong (Hong Kong Telecom Netplus)</td>
<td>Nextlink Communications (XOXO)</td>
</tr>
<tr>
<td>India (National Centre for Software Technology)</td>
<td>Pac West Telecom, Inc. (PACW)</td>
</tr>
<tr>
<td>Japan (MutugotoNet)</td>
<td>Cable and Wireless (CWP)</td>
</tr>
<tr>
<td>Thailand, (Internet Thailand)</td>
<td>Level 3 (LVLT)</td>
</tr>
<tr>
<td>Netherlands (Surfnet)</td>
<td>Qwest (Q)</td>
</tr>
<tr>
<td>Russia (Relcom)</td>
<td>Sprint (FON)</td>
</tr>
<tr>
<td>Singapore (SingNet)</td>
<td>UUNET (WCOM)</td>
</tr>
<tr>
<td>Sweden (Royal Institute of Technology)</td>
<td>Williams (WCG)</td>
</tr>
<tr>
<td>Switzerland (CERN)</td>
<td>SBC Communications (SBC)</td>
</tr>
<tr>
<td>Taiwan (Ministry of Education Taiwan)</td>
<td>Verizon (VZ)</td>
</tr>
<tr>
<td>United Kingdom (Jellybaby Networks)</td>
<td>Focal Communications Corp (FCOM)</td>
</tr>
<tr>
<td>USA (University of California, University of Maryland, Princeton University, Iowa State, Texas A&amp;M)</td>
<td>Level 3 (LVLT)</td>
</tr>
</tbody>
</table>
Validation Tool Design

• Java code to communicate with LG sites' traceroute cgi-bin scripts at configurable intervals
  – Filter extraneous output from output i.e. HTML, eye candy
  – Find Target string(s) unique for each customer site
    • e.g. swbell.net and sbcglobal.net
  – Indicate "change" or "no change" to previous result
  – Record “Peer” and Border Router
    • Raw: 183.ATM7-0.GW5.NYC8.ALTER.NET (152.63.23.225)
    • Abbreviated: ALTER.NET

• Challenges
  – LG sites have unique cgi-bin scripts
  – Many not in English, and selections needed
  – Targets may be one or two each per customer i.e swbell.com and sbcglobal.com
  – Unresolved IP addresses
  – *s in traceroute result
  – Load balancing effects
  – Route changes, but previous AS does not change
Traceroute Validation Results

- **24 hour run**
  - Each LG site hits each target site every 30 minutes; total of 10,000 samples
  - Non-aggregated: 4.8%
  - Aggregated: 0.4%
- **4 day run**
  - Each LG site hits each target site every 60 minutes; total of 31,000 samples
  - Non-aggregated: 6.4%
  - Aggregated: 0.6%
- **Last-hop on path from LG sites to Target is stable** *(we do not consider complete path)*
  - EIA set more stable than previously thought
BGP-based Validation

- Determine mapping of peer ASs to source ASs for Target 4.2.101.20 on Genuity's network (AS 1)
- Relevant data from Routeviews in "show ip bgp" format

<table>
<thead>
<tr>
<th>Network</th>
<th>Next Hop</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>* 4.0.0.0</td>
<td>193.0.0.56</td>
<td>3333 9057 3356 1 i</td>
</tr>
<tr>
<td>* 217.75.96.60</td>
<td>16150 8434 286 1 1</td>
<td></td>
</tr>
<tr>
<td>* 141.142.12.1</td>
<td>1224 38 10514 3356 1 i</td>
<td></td>
</tr>
<tr>
<td>* 4.2.101.0/24</td>
<td>141.142.12.1</td>
<td>1224 38 6325 1 i</td>
</tr>
<tr>
<td>* 202.249.2.86</td>
<td>7500 2497 1 i</td>
<td></td>
</tr>
<tr>
<td>* 203.194.0.5</td>
<td>9942 1 i</td>
<td></td>
</tr>
<tr>
<td>* 66.203.205.62</td>
<td>852 1 i</td>
<td></td>
</tr>
<tr>
<td>* 167.142.3.6</td>
<td>5056 1 e</td>
<td></td>
</tr>
<tr>
<td>* 206.220.240.95</td>
<td>10764 1 i</td>
<td></td>
</tr>
<tr>
<td>* 157.130.182.254</td>
<td>19092 1 i</td>
<td></td>
</tr>
<tr>
<td>* 203.62.252.26</td>
<td>1221 4637 1 i</td>
<td></td>
</tr>
<tr>
<td>* 202.232.1.91</td>
<td>2497 1 i</td>
<td></td>
</tr>
<tr>
<td>*&gt; 4.0.4.90</td>
<td>1 i</td>
<td></td>
</tr>
</tbody>
</table>

Target AS: 1

Relevant IP address blocks: 4.0.0.0, 4.2.101.0/24
Peer ASs: 3356, 286, 6325, 2497, 9942, 852, 5056, 10764, 19092, 1221, 2497
Source ASs: 3333, 9057, 16150, 8434, 1224, 38, 10514, 7500, 1221

Mapping from source ASs to peer ASs:

<table>
<thead>
<tr>
<th>Peer AS</th>
<th>Source AS set</th>
</tr>
</thead>
<tbody>
<tr>
<td>3356</td>
<td>3333, 9057, 10514</td>
</tr>
<tr>
<td>286</td>
<td>16150, 8434</td>
</tr>
<tr>
<td>6325</td>
<td>1224, 38</td>
</tr>
<tr>
<td>2497</td>
<td>7500</td>
</tr>
<tr>
<td>4637</td>
<td>1221</td>
</tr>
</tbody>
</table>

“Expected IP address” set for a peer AS are all addresses belonging to source ASs mapping to that peer AS
Peer and Source ASs for Genuity (now Level3)
BGP Validation Results

- Processed routeviews BGP data for determining stability of source AS set
- Same set of 20 target ISPs as Looking Glass analysis, now at AS level
- For each target AS, determined set of peer ASs and corresponding source AS set over 30 days, every 2 hours: 346 data points
- **Average change** in source AS set is 1.6%; Max is 5%
- Mapping between peer ASs and source ASs is stable
  - EIA set more stable than previously thought
# Detailed Results

<table>
<thead>
<tr>
<th>target</th>
<th>#peers</th>
<th>#sources</th>
<th>changes</th>
<th>no_change</th>
<th>percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>aol</td>
<td>2</td>
<td>52</td>
<td>267</td>
<td>34623</td>
<td>0.76526226</td>
</tr>
<tr>
<td>broadwing</td>
<td>23</td>
<td>25</td>
<td>181</td>
<td>11081</td>
<td>1.6071746</td>
</tr>
<tr>
<td>concentric</td>
<td>24</td>
<td>49</td>
<td>424</td>
<td>25305</td>
<td>1.6479459</td>
</tr>
<tr>
<td>cw</td>
<td>28</td>
<td>49</td>
<td>909</td>
<td>31714</td>
<td>2.7863776</td>
</tr>
<tr>
<td>earthlink</td>
<td>1</td>
<td>47</td>
<td>42</td>
<td>16894</td>
<td>0.24799244</td>
</tr>
<tr>
<td>focal</td>
<td>6</td>
<td>53</td>
<td>1189</td>
<td>46238</td>
<td>2.5070108</td>
</tr>
<tr>
<td>global</td>
<td>37</td>
<td>25</td>
<td>1404</td>
<td>26287</td>
<td>5.0702393</td>
</tr>
<tr>
<td>globix</td>
<td>32</td>
<td>16</td>
<td>181</td>
<td>7133</td>
<td>2.474706</td>
</tr>
<tr>
<td>level3</td>
<td>30</td>
<td>21</td>
<td>164</td>
<td>10225</td>
<td>1.5785927</td>
</tr>
<tr>
<td>mcleod</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>null</td>
</tr>
<tr>
<td>nextlink</td>
<td>1</td>
<td>44</td>
<td>37</td>
<td>15253</td>
<td>0.24198822</td>
</tr>
<tr>
<td>pacwest</td>
<td>4</td>
<td>47</td>
<td>193</td>
<td>51296</td>
<td>0.37483734</td>
</tr>
<tr>
<td>prodigy</td>
<td>3</td>
<td>46</td>
<td>83</td>
<td>16663</td>
<td>0.49564075</td>
</tr>
<tr>
<td>qwest</td>
<td>26</td>
<td>50</td>
<td>1061</td>
<td>38179</td>
<td>2.7038736</td>
</tr>
<tr>
<td>rcn</td>
<td>21</td>
<td>46</td>
<td>509</td>
<td>38954</td>
<td>1.2898157</td>
</tr>
<tr>
<td>sbc</td>
<td>17</td>
<td>47</td>
<td>137</td>
<td>17972</td>
<td>0.7565299</td>
</tr>
<tr>
<td>sprint</td>
<td>40</td>
<td>59</td>
<td>1634</td>
<td>158836</td>
<td>1.0182588</td>
</tr>
<tr>
<td>uunet</td>
<td>56</td>
<td>66</td>
<td>4649</td>
<td>180047</td>
<td>2.5171092</td>
</tr>
<tr>
<td>verizon</td>
<td>5</td>
<td>55</td>
<td>265</td>
<td>82468</td>
<td>0.3203075</td>
</tr>
<tr>
<td>williams</td>
<td>18</td>
<td>33</td>
<td>128</td>
<td>13119</td>
<td>0.9662565</td>
</tr>
<tr>
<td><strong>total</strong></td>
<td><strong>13457</strong></td>
<td><strong>822287</strong></td>
<td><strong>13457</strong></td>
<td><strong>822287</strong></td>
<td><strong>1.610182</strong></td>
</tr>
</tbody>
</table>
InFilter System Architecture

Target Network

- Peer AS₁
- Peer AS₂
- BR₁ (NetFlow Enabled)
- Peer AS₃
- BR₃ (NetFlow Enabled)
- Peer AS₄

Flow-tools software
EIA set, Clustering & Nearest Neig. software
Flow-tools software
Information Flow Diagram

**Initialization Phase**

- IP Traffic → BR (Netflow)
- Netflow reports → Flow-Tools
- IP Flow statistics → EIA set generation software

- EIA set for each BR

**Normal Phase**

- IP Traffic → BR (Netflow)
- Netflow reports → Flow-Tools
- IP Flow statistics → EIA set analysis software

- Suspect Flows → Clustering software
- Clusters for known attack & normal traffic → Nearest Neighbor software
- Attack Flows (IDMEF-based Reports)

EIA: Expected IP Address
BR: Border Router
NetFlow

- Standardized network accounting technology
- Involves:
  - Collection of traffic statistics at a router
  - Export of statistics to consumer applications over UDP
- Developed by Cisco
**Flow-tools**

- Suite of programs used to capture and analyze NetFlow exports
- Developed at Ohio State University by Mark Fullmer
- Flow-Capture
  - Listens for NetFlow exports on a UDP port
  - Stores received records in binary format
  - Possible to control the number of files into which data is written as well as the volume of data in all the files
- Flow-Report
  - Analyzes flow data captured by Flow-capture
  - Generates ASCII text reports containing statistical information computed from flows
- Other tools to aggregate flow files, filter information in flow files, redistribute flows to other destinations and export to various formats
EIA Set Generation

How is EIA set membership change handled during Initialization phase?

- Include flow in EIA set of last ingress BR used by the flow
- Validation experiments have demonstrated low change in EIA set membership

Receive data on IP flow from Flow-Tools

Source IP in EIA set of ingress BR?

Source IP in EIA set of other BR?

Remove source IP from EIA set of other BR

Add to EIA set of ingress BR
Clustering Algorithm

- Used in the Initialization Phase for each BR
- Part I – Processing flow data
  - Map attributes of each flow to a single data point
    - E.g. pps, bps, duration, ToD, average packet size, average number of octets, average number of packets
    - Cannot use source IP as cluster attribute since address allocation may not be contiguous
  - Data point representation permits analysis of how similar flows are to each other
Clustering Algorithm

- Part II – Post-processing of collected information
- Combination of signature- and anomaly-based detection
  - Create arbitrary number of clusters of normal and attack flows using agglomerative clustering
- Agglomeration can be based on well-known services or port numbers
  - Possible to characterize “normal” behavior of well-known services, such as HTTP, FTP, SNMP, etc.
  - Each “normal” behavior cluster can represent a specific service

Read all clusters and cluster centers from disk

Process Clusters

Generate Proximity table using points from all clusters

Divide normal behavior cluster into smaller clusters using “Agglomeration”

Write the proximity table and the new clusters to disk
EIA Set Analysis

- EIA sets may not be complete during Initialization phase
  - New members of EIA sets can be added during Normal phase, even after Initialization phase

Receive data on IP flow from Flow-Tools

Source IP in EIA set of ingress BR?
- Yes
  - OK
- No
  - Flow classified as suspicious by NN search?
    - Yes
      - Display on GUI using IDMEF
    - No
      - Source IP in EIA set of other BR?
        - Yes
          - Display on GUI using IDMEF
        - No
          - Add to EIA set of ingress BR

Remove source IP from EIA set of other BR

Flow classified as suspicious by NN search?
- Yes
  - Display on GUI using IDMEF
- No
  - Source IP in EIA set of other BR?
Nearest Neighbor Algorithm

- Used in the Normal Phase
InFilters to Detect and Filter new DDoS Attacks

- InFilter can be used to detect new DDoS attacks using spoofed IP addresses
  - BRs have NetFlow enabled
  - Flow-tools software deployed on COTS workstations to receive NetFlow data exported by BRs
  - Single Flow-tools instance can handle exports from multiple BRs
  - All Flow-tools enabled workstations transfer relevant traffic statistics to workstation running Clustering Analysis software
  - Multiple Flow-tools instances can be deployed for load sharing

- InFilter can be used to filter out DDoS traffic with spoofed IP addresses
  - For each BR, install EIA-set-based filters at FE, on IP-in-IP tunnel between that BR and FE
DDoS Detection and Protection Service

BR: Border Router
FE: Filtering Engine
IP-in-IP Tunnels & eBGP sessions
Summary

- DDoS detection and mitigation solution could be used by ISP as service to broadband customers
  - Incremental deployment possible
  - Scalable to tens of thousands of customers
  - Ensures backbone IP network stability by obviating need for configuring in-network routers/switches, and reducing impact on non-DDoS traffic
  - Enables automated DDoS detection, reporting, mitigation, and trace-back