"An overview of Denial of Service Issues and Solutions in operators networks"

Olivier Paul
GET/INT/LOR

08/31/2005
1a Escuela de Verano Franco-Mexicana de Telecomunicaciones
Acknowledgements

HFR Project

DDOS Project

FIREFALL
Disclaimer

• This presentation is *not* meant to be complete.
# DoS Attacks definitions

<table>
<thead>
<tr>
<th>Source</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISO 7498-2 [ISO89]</td>
<td>“The prevention of authorised access to resources or the delaying of time of critical operations.”</td>
</tr>
<tr>
<td>NIST [Ols95]</td>
<td>“Actions that prevent a network element from functioning in accordance with its intended purpose. Network elements may be rendered partially or entirely unusable for legitimate users. Denial of service may cause operations which depend on timeliness to be delayed.”</td>
</tr>
<tr>
<td>CERT [CERT97]</td>
<td>“A denial-of-service attack is characterised by an explicit attempt by attackers to prevent legitimate users of a service from using that service”</td>
</tr>
</tbody>
</table>
Agenda

• Introduction
  – DoS attacks taxonomy
  – Some figures.

• Existing DoS mitigation schemes
  – 4 main phases
  – Mainly carriers/operator networks oriented

• Conclusion
DoS taxonomy

- Degree of indirection
DoS taxonomy

- Degree of indirection
DoS taxonomy

• Degree of reflection
DoS classification criterion

- Mirkovic & al., SIGCOMM CCR 04
  - Source Address validity
    - Spoofed Address
    - Valid address
  - Characterization
    - Characterizable.
      - Filterable.
      - Non Filterable.
    - Non characterizable
DoS classification criterion

- Mirkovic & al., SIGCOMM CCR 04
  - Victim type
    - Application
    - Operating system
    - Resource
    - Network
  - Exploited Vulnerability
    - Semantic
      - Design level
      - Implementation level
    - Brute force
DoS classification criterion

- Mirkovic & al., SIGCOMM CCR 04
  - Rate dynamics
    - Constant Rate
    - Variable Rate
      - Increasing.
      - Decreasing.
      - Fluctuating.
  - Impact
    - Disruptive.
      - Self recoverable.
      - Human recoverable.
      - Non recoverable.
    - Degrading.
    - None.
DDoS classification criterion

- Mirkovic & al., SIGCOMM CCR 04
  - Attack Networks Model
    - Agent Based. Slave-Master Model (eg: TFN2K).
    - Server based:
      - IRC Based. Use IRC servers for communications (eg: Agobot).
      - P2P based. Registers with cache servers. Use P2P protocol for communication (eg: Phatbot: Gnutella + WASTE)
  - Degree of automation
    - Relates to the various phases in a DDoS attack:
      - scanning,
      - exploitation,
      - installation,
      - attack control
    - Manual, Semi-Automated, Fully Automated
DDoS Tools

- Usually integrate several types of attacks
- Example: TFN2K, Released in 1999.
  - Integrates:
    - ICMP Flood, UDP Flood, TCP SYN Flood,
    - SMURF. Use broadcast as address destination address and victim address as source.
    - Targa3. Use uncommon IP packets to exploit vulnerabilities in protocol stacks.
    - Able to send mixed attacks.
  - Slave-Master Model
    - Uses CAST-256 encryption between master and slaves.
    - Communication using a protocol (UDP, ICMP, TCP) chosen randomly.
    - Does not use acknowledgements.
Are there a lot of attacks?

- **Inferring DoS Attacks (Moore & al., USENIX 01)**
  - Back-scatter analysis (Random destination address, no previous request)
  - /8 network, one week monitoring.
    - 13k attacks.
    - Mostly TCP and ICMP.
    - Duration from 1s to several days. 50% >10 minutes.
    - 38% > 500 pps, highest 0.7M pps.

- **Trends in DoS Attacks (Nazario, Usenix Security 03)**
  - Back-scatter analysis, /8 routable, unused network, 1 month monitoring.
    - 120k attacks.
    - 2002: 75% TCP, 2003: 90% UDP.
    - Duration from 1s to one day, 10% > 10 minutes.
    - 2% > 100k packets, 0.5% > 1M packets.
An expending business?

• Inferring DoS Attacks (Moore & al., USENIX 2001)
  – Individuals (13%).
  – Infrastructure (7%).
  – No particular geographical preferences.

• From NANOG mailing List:
  Date: Thu, 3 Jun 2004 23:32:19 -0700
  From: <NANOG Mailing list>
  Dear sirs.
  We are glad to you to give qualitative service, on elimination of sites. We can kill any site by our attack, which have name 'DDos attack'<...>
  The prices at us are low, 60 dollars for 6 hours. 150 dollars a day. Destroy any project on the Internet with the help of ours DDos service. Payment prinimaetsja in system WebMoney.
An expending business?

- Drone Armies Command and Control Centers.
  - June 2005 report (from Gadi Evron monthly report)

<table>
<thead>
<tr>
<th>Domain</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>SERVER4YOU - Server4You Inc.</td>
<td>49</td>
</tr>
<tr>
<td>UNITEDCOLO-AS Autonomous Syste</td>
<td>44</td>
</tr>
<tr>
<td>SAGONET-TPA - Sago Networks</td>
<td>80</td>
</tr>
<tr>
<td>MFNX MFN - Metromedia Fiber Ne</td>
<td>61</td>
</tr>
<tr>
<td>NOC - Network Operations Cente</td>
<td>39</td>
</tr>
<tr>
<td>AS13680 Hostway Corporation Ta</td>
<td>22</td>
</tr>
<tr>
<td>FDCSERVERS - FDCservers.net LL</td>
<td>42</td>
</tr>
<tr>
<td>NEBRIX-CA - Nebrix Communicati</td>
<td>33</td>
</tr>
<tr>
<td>ASN-NA-MSG-01 - Managed Soluti</td>
<td>31</td>
</tr>
<tr>
<td>LAMBDANET-AS European Backbone</td>
<td>15</td>
</tr>
<tr>
<td>INFOLINK-MIA-US - Infolink Inf</td>
<td>28</td>
</tr>
<tr>
<td>Lycos Europe GmbH</td>
<td>17</td>
</tr>
</tbody>
</table>

Note: Each C&C can command 100k+ slaves!
DoS mitigation techniques

- Usually divided in four phases:
  - Prevention
  - Detection
  - Tracking
  - Suppression
  - (Post Mortem)
Prevention

• Customer side
  • Design more resilient protocols.
  • Implement more resilient software.
  • Protect end hosts (patching, anti virus, firewall).

• Public network side
  • Block well known protocol/ports/address ranges during outbreaks.
  • Address spoofing prevention.
    – Ingress filtering/RPF.
    – Hop count based filtering (Jin & al. CCS 2003).
    – Source Address Validation (Li & al. INFOCOM 2002).
Ingress Filtering

- IETF BCP 38
  - Restrict the scope of addresses that can be used by the attacker.
  - Use ACL on routers.
Ingress Filtering

- **IETF BCP 38**
  - High in the network. Increase spoofing ability.
  - Low in the network. Increase management burden.
  - Some protocol need to use foreign address (e.g. Mobile IP)
  - Some other protocols need to use special addresses (e.g. BOOTP)
  - Some performance issues.
  - Not widely implemented.
uRPF

- Unicast Reverse Path Forwarding
  - Or … how to use routers architecture smartly
**uRPF**

- **Unicast Reverse Path Forwarding**
  - Use FIB information to build filtering rules automatically.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Outgoing Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>I2</td>
</tr>
<tr>
<td>R</td>
<td>I1</td>
</tr>
</tbody>
</table>

- No rules at all, Just use the FIB!
- No performance issues.
Hop Count Filtering

- Jin & al. ACM CCS 2003:
  - Number of hops between end hosts should remain the same.

For each packet (SA, TTL)
- Infer original TTL’ for source SA.
- Compute hop count HC = TTL’ - TTL.
- Retrieve stored hop count for SA:HC[SA].
- HC ≠ HC[SA] flag packet as spoofed.
Hop Count Filtering

- Jin & al. ACM CCS 2003:
  - Assumptions:
    - Ability to infer original TTL.
      - Based on the fact that only a few original TTLs values exist in practice.
    - Routes stability.
    - Not all existing sources have the same hop count.
    - Attacker
      - is not located close to the victim.
      - is not able to know hop count between arbitrary hosts and victim.
  - Implementation:
    - Use prefix based aggregation to limit size of HC[] table.
DoS mitigation techniques

- Usually divided in four phases:
  - Prevention
  - Detection
  - Tracking
  - Suppression
  - (Post Mortem)
Detection Goals

- Decide whether an attack is going on.
- Identify the traffic related to the attack.
  - Generate a signature that will unambiguously identify the traffic generated by the attacker.
  - This signature will later be used in other mitigation phases.
Detection Taxonomy

- Detection Model
- Detection Location
- Detection Co-operation
  - Standalone (Most existing techniques).
  - Sharing information
    - Speed up/allow detection.
      - Peng & al. (ACISP 03).
      - Lakhina & al. (INFOCOM 04).
Detection Taxonomy

- Detection location
  - Close to victim.
    - Detection is easy but result comes too late.
  - Close to attacker.
    - Because of distribution, attack events may be scarce.
  - Intermediate networks.
    - Need to interact with existing devices.
    - High speed processing.
Detection Taxonomy

• Detection Model
  – Knowledge based
    • Know what an attack looks like.
    • Mainly applies to vulnerabilities oriented attacks
      – eg: Snort rule for malformed request DoS attack on REAL audio server:
        ```
        ```
  – But also works for some poorly coded attack tools:
    – eg: Snort rule for *shaft* SYN Flooding attack:
      ```
      ```
Detection Taxonomy

- **Detection Model**
  - Knowledge based main challenges
    - Obfuscation
    - Speed:
      - Faster pattern matching algorithms (1Gb/s).
      - FPGA based implementations (~5Gb/s).
  - Signatures construction
    - Discover new attacks (Honeypots, Sinkholes, Network telescopes).
    - Build new signatures rapidly and automatically.
      » Network/Transport based methods (eg. Estan SIGCOMM 02)
      » Application level (eg. Singh & al. OSDI 04, Kim & al. USENIX SEC 04)
      » Implementation at UCSD ~200Mb/s.
Detection Taxonomy

• Detection Model
  – Behaviour based
    • Know how system usually behaves.
    • Changes in behaviour can only be explained by attack.
  – Behaviour based main challenges
    • Performance (depending on its location), Speed of detection.
    • Adaptation to existing monitoring/network devices.
    • Accuracy.
Detection Taxonomy

- Behaviour based Detection Model Characteristics
  - Uncorrelated/Autocorrelated Data
    - Variation around mean is random.
    - Variation depends on previous signal values.
  - Able to detect large/small variations
    - Small variations: adapted to entry points.
    - Large variations: closer to victim.
  - Level of aggregation supported.
    - More aggregation => lower processing overhead.
    - More aggregation => less precise attack signatures.
  - Univariate/Multivariate
    - Univariate (number of bytes/packets, number of addresses).
    - Multivariate (addresses, bytes, any combination of bytes in the packet).
Intermediate networks

- Asymmetry monitoring (Gil & al. Usenix 01).
  - Basic idea:
    - Normal operation yield fixed forward/backward traffic volume ratio.
    - DDoS generate variations of this ratio.

![Graph showing server capacity and requests served vs requests received]

- Introduction
- Mitigation
- Prevention
- Detection
- Tracking
- Suppression
Intermediate networks

- Asymmetry monitoring (Gil & al. Usenix 01).
  - Basic idea:
    - It is not feasible nor effective to keep a state for each destination.
    - Create a data structure that allows asymmetry to be kept.
  - Structure represents forward/backward ratio.
  - Root stores ratio for /8 prefixes.
  - If ratio for w/8 passes over a given level we create a leaf zooming on w.x/16.
  - If ratio for w.x/16 passes below a given level we delete the w.x/16 leaf and collapse the results in w/8.
  - Structure size is limited.
Intermediate networks

- Power Spectral Analysis (Cheng & al. GLOBECOM 02)
  - Basic idea:
    - Well regulated TCP flows can be considered as legitimate.
    - Keeping a state for each TCP flow is not feasible.
    - Need to find this information per aggregate.
  - Spectral Analysis.
    - Let’s consider a packet arrival process
      - \( X(t) = \text{number of packets received during } [t-d,t], \) (d~10ms)
    - We can evaluate periodicity of the packet arrival process using an autocorrelation function.
      - \( R(l,t) = \text{degree of correlation between signal at } t \text{ and signal at } t+l \)
      - Maximise \( R(l,t) \)
Intermediate networks

- Why should we get some periodicity?
Intermediate networks

- Power Spectral Analysis (Cheng & al. GLOBECOM 02)
  - Example (3 TCP flows, F1(p=3,s=0), F2(p=3,s=1), F3(p=1,s=0)):

```
Poor correlation
(low R(I1,t))
```
Intermediate networks

- Power Spectral Analysis (Cheng & al. GLOBECOM 02)
  - Example (3 TCP flows, F1(p=3, s=0), F2(p=3, s=1), F3(p=1, s=0)):

  ![Diagram of power spectral analysis](image)

Some correlation
(Average R(12, t))
Intermediate networks

- Power Spectral Analysis (Cheng & al. GLOBECOM 02)
  - Example (3 TCP flows, F1(p=3,s=0), F2(p=3,s=1), F3(p=1,s=0)):

Very Good correlation
(High R(I3,t))
Intermediate networks

- **Power Spectral Analysis** (Cheng & al. GLOBECOM 02)
  - Represent “power” of various periods using periodogram.
    - Frequency representation should present “peaks” around 1/RTT values.
    - Attacks flatten these peaks by distributing frequency power evenly.
Intermediate networks

• Power Spectral Analysis (Cheng & al. GLOBECOM 02)
  – Detection
    • Selects an aggregate of flows from the same prefix.
      – Assumption 1: Flows from the same prefix have the similar RTT values.
      – Assumption 2: Most flows are TCP.
      – Assumption 3: Most packets are from long TCP flows.
    • Use previously defined method to compute Power Spectral Density.
    • Compare power of strongest frequency to power of other frequencies:
      – Small difference: Not TCP traffic.
      – Large difference: TCP traffic.
  – Efficiency varies with TTL values/variations, degree of aggregation.
  – Roughly 80% correct classification with real life normal traffic.
Intermediate networks

- Wavelet based detection.
  - Representing time and frequencies power.
    - Huang & al. (SIGCOMM 01).
    - Barford & al. (IMW 02).
Close to the attacker

- **Challenges:**
  - Instant variations sometimes do not provide enough information to detect attacks.
    - Variations are small.
    - Attack diluted over time.
- **Statistical Process Control techniques**
  - Developed to control quality of manufacturing processes.
- **Well known techniques to detect small changes**
  - Cumulative Sum (CuSum) Charts.
  - Exponentially Weighted Moving Averages (EWMA) Charts.
Close to the attacker

- Monitoring changes in the mean.
  - Change small compared to mean or variance values.
  - Need to take deviation history into account.
  - Use cumulative sum of deviations vs expected mean.

Random(x)
Expected mean: 0.5

Random(x)+0.1
Expected mean: 0.5
Close to the attacker

- Cumulative SUM techniques (Wang & al. INFOCOM 02).
  - Basic Idea:
    - TCP connections:
      - Started with SYN packet
      - Closed with FIN / RST packets.
    - Ratio SYN/(FIN+RST) should be fixed.
      - Measure SYN at time $t$.
      - Measure FIN+RST at time $t+d$ where $d$ is average duration of TCP flow.
      - Assumption: traffic is symmetric.
    - In the case of TCP SYN Flooding:
      - No answer when attack is successful.
      - Ratio should increase.
    - Use CUSUM to use scheme close to attacker.
Close to the attacker

- Cumulative SUM techniques (Wang & al. INFOCOM 02).
  - Processing:
    - $X_n$ normalised version of the measured value. # $0 \leq X_n \leq 1$
    - $c = E(X_n)$, $a = \max (c)$. # $c$ is the mean for $X_n$
    - $X'_n = X_n - a$ # $X'_n < 0$ when normal
    - $y_n = (y_{n-1} + X'_n)^+$ # $y_n > 0$
  - Detection:
    - $D_n = 1$ iff $y_n > T$ # $T$ attack threshold
    - $D_n = 0$ otherwise.
  - Results:
    - Traffic: 1500-4000 SYN packets/s.
    - Detects attacks as low as 35 SYN per second.
Close to the attacker

• Cumulative SUM techniques.
  – Measurable parameters.
    • Must exhibit some natural stability.
      – Ratio TCP SYN/FIN(RST) packets (Wang & al. INFOCOM 02).
      – Number of source IP addresses in use in a specific network (Peng & al.
        GLOBECOM 03)
    • Or transformed to exhibit some stability (Siris & al. TCP SYN,
      GLOBECOM 04)

• EWMA techniques
  – Measurable parameters.
    • Bits/s. EWMA for autocorrelated data (Ye & al, IEEE Transactions on
      Reliability 03)
    • Bits/s. Holt-Winters Forecasting (Jake Brutlag, USENIX LISA 00).
Multivariate approaches:

- Problem: curse of dimensionality.
- Start appearing:
  - 2 papers at SIGCOMM 05.
Detection on the operator side

• Conclusion
  – Some products now implement similar techniques.
  – However:
    • Detection is today mostly performed on customer side.
    • Or close to customer.
  – Need to have a feedback to other mitigation phases.
Link between customer and operator

- Mostly phone oriented

My web server is going down: Help!

Attack signature
Link between customer and operator

- Remote Triggered Blackhole Capability (Sprint, UUnet)
  - Triggers blackhole creation inside ISP network.
  - By sending a new route advertisement through BGP.
Link between customer and operator

- IETF ID Message Exchange working group IDMF

```
<?xml version="1.0" encoding="UTF-8"?>
<idmef version="1.0" xmlns:idmef="http://iana.org/idmef">
<alert messageid="42760">
  <analyzer analyzerid="HTTP Overload Module0.1">
    <model> HTTP Overload Module </model>
    <version> 0.1 </version>
  </analyzer>
  <create_time ntpstamp="0">
    <detect_time> 111755724 </detect_time>
  </create_time>
  <source spoofed="No">
    <node>
      <address category="ipv4-addr">
        <address>192.168.0.2</address>
      </address>
    </node>
    <port> 1057 </port>
  </source>
  <assessment>
    <impact>
      <severity> 1.096633 </severity>
      <completion> 1 </completion>
      <type> 1 </type>
    </impact>
  </assessment>
</idmef>
```

Attack signature
DoS mitigation techniques

- Usually divided in four phases:
  - Prevention
  - Detection
  - Tracking
  - Suppression
  - (Post Mortem)
• Goals and problems in operator network
  – In theory: Find the source of packets matching a specific “pattern”.
    • Find responsible to bring him in front of a court.
    • Charge him for damages.
  – In practice: Find the entry point of these packets in operator networks.
    • Block packets as soon as possible to avoid useless traffic in operator network.
  – Why not use source address?
    • Spoofed addresses.
    • Asymmetric routes.
Tracking

• Existing approaches
  – Extend flow information.
    • On each network device, send some identifying information to the destination along with the flow.
    • Destination recovers identification information to guess the path to packets sources.
  – Trace back flows.
    • From the destination, find sources among immediate neighbors.
    • Ask neighbors to perform similar job.
Extend flow information

- Two classes of approaches
  - Packet marking.
    - Use specific packet selection strategy.
    - Store identification information into IP packet.
    - Storing space is limited.
  - ICMP message.
    - Use common packet selection strategy.
    - Build ICMP packet including captured packet and identification information.
    - Send both packets to the destination.
Packet Marking

- Packet Marking basics
Packet Marking

- Packet Marking basics
Packet Marking

• Packet Marking basics:
  – Issue#0: Distance attenuation
Packet Marking

- Packet Marking basics:
  - Issue#1: Packet Overwrite
Packet Marking

• Packet Marking basics
  – Issue#2: Accidental collision
Packet Marking

- Packet Marking basics
  - Issue#3: Voluntary collision
Packet Marking

- Mathematical problems that play a role in packet marking:
  - Birthday paradox:
    - Take n elements $x_1 \ldots x_n$ from a set made of N with a probability $p = 1/N$.
    - the probability that $\exists j,k$ such that $x_j = x_k$ is
      \[ P_{\text{collision}(n,N)} = 1 - (1 - 1/N)^{(n \cdot (n-1))/2} \]
  - Coupon collector problem:
    - Let’s consider a set of n stamps.
      - We have to collect n stamps $s_1 \ldots s_n$ to win.
      - The probability to get stamp $s_i$ on a box is $1/n$
    - The number of boxes you have to buy to get n coupons is:
      \[ E(b) = n \cdot (\log(n)+1) \]
Packet Marking

- Mathematical problems that play a role in packet marking:
  - Packet marking probabilities
    - Let’s consider packets with a single stamp.
    - Marking probability by each router is $p$.
    - Marking probability by router $R_d$ at $d$ hops from receiver is:
      \[
      P_{Rd} = p \cdot (1 - p)^{d-1}
      \]
    - Probability that marking by routers $R'_d$ at more $d$ hops from receiver is not overwritten is:
      \[
      P_{R'd} = (1 - p)^d
      \]
      (Note: $p > 1/2 \Rightarrow P_{Rd} < P_{R'd}$, $p < 1/2 \Rightarrow P_{R'd} > P_{Rd}$)
Packet Marking

• Probabilistic packet marking (Savage & al. SIGCOMM 00)
  – Only use 1 field and store field in IP header ID field.
    • May be treated by ASICS: prevent packet punting to GP processor.
    • Packet marking is still probabilistic (P = 1/r).
      – Performance issues.
      – Limit impact on fragmentation (ID field modification generates packet rejection).
  • Why ?:
    – Fragmented traffic constitutes a small part of overall traffic (~0.45% Shannon & al. ACM TON 2002)
    – Marking is probabilistic (not every packet is marked).
    – Recall \( P_{\text{notmarked}} = (1-1/r)^r, r=30 \Rightarrow P_{\text{notmarked}} = 0.37 \)
    – \( P_{\text{loss}} = P_{\text{marked}} \cdot P_{\text{frag}} = (1 - 0.37) \cdot 0.0045 = 0.0028 \) (0.28%)
    – Average packet loss on the internet: 0.011 (1.1%).
    – How can you store addresses in a 16bits field?
Packet Marking

- Probabilistic packet marking (Savage & al. SIGCOMM 00)
  - How do you store addresses in a 16bits field?
    - Store edges (couple of addresses) instead of addresses.
    - Need to capture packets that were captured by previous router.
  - Divide space requirements by 2 using xor property:
    - \( A \oplus B \oplus A = B, B \oplus A \oplus B = A \).
Packet Marking

• Probabilistic packet marking (Savage & al. SIGCOMM 00)
  – How do you know B is between A and C?
    • Include distance field to indicate distance between B and V.
    • Distance initialized to 0 by edge router
    • Incremented by each marking router.
  – We still need a lot of bits!
    • Reduce space requirements by using n address slices
    • Space requirement divided by n but packets# required multiplied by n
    • Use log n to indicate slice id.
      – 16-log n bits for address slice.
      – log n bits for slice id.

<table>
<thead>
<tr>
<th>Distance</th>
<th>Slide id</th>
<th>Address slice</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>3</td>
<td>8</td>
</tr>
</tbody>
</table>
Packet Marking

• Probabilistic packet marking (Savage & al. SIGCOMM 00)
  – However IP address chunks are not necessarily unique:
    \[
    \begin{array}{c}
    192.200.128.12 \\
    192.133.128.12 \\
    \end{array}
    \]
    Mainly due to addressing method in the Internet
    3 identical chunks
  
  – XOR of IP address chunks bears the same problem.
  
  – You need to break IP addresses structure in the chunk.
    • Idea: “mix” another values with bit interleaving to break the structure.
    • You cannot choose a random number (no way to find back the address).
    • Idea: Use hash function (pseudo random).
Packet Marking

• Probabilistic packet marking (Savage & al. SIGCOMM 00)
  – How do you perform reconstruction?
    • Store n stamps.
    • Order stamps according to distance into d bins.

1  2  3  ...  d

• Order stamps according to chunk id i.
  • You need d bins (Vertical coupon collector problem)
  • With k chunks (Horizontal coupon collector problem)
Packet Marking

- Probabilistic packet marking (Savage & al. SIGCOMM 00)
  - How do you perform reconstruction?
    - For stamps S in bin 1 (Stamps only include last node before victim)
      
      \[
      H(O\_1\_2) = E\_1\_2
      \]
      
      \[
      H(O\_2\_1) = E\_2\_1
      \]
      
      \[
      H(O\_1\_2) = E\_1\_2
      \]
      
      \[
      H(O\_2\_1) = E\_2\_1
      \]

      ... 

      \[(m[1^k]^2 combinations)\]

      ![Diagram showing packet marking process](image)
Packet Marking

- Probabilistic packet marking (Savage & al. SIGCOMM 00)
  - How do you perform reconstruction?
    - For stamps S in bin n

\[ H(01 \oplus 12) = E^{12} \]
\[ H(01 \oplus 12) = E^{12} \]
\[ H(01 \oplus 12) = E^{12} \]
\[ H(01 \oplus 12) = E^{12} \]

... \[ m[n-1].(m[n]^k)^2 \] combinations

Set of valid addresses for n-1:
- 1 2
- 1 2

Chunks received
- 2 1 1 2 2
- with distance=n

De-interleaving
- O 2
- address
- E 2
- hash

2 chunks
Improvements

• Make more space
  – No addresses, use hash values and network map (Song & al. INFOCOM 01).
  – Use short space for address and larger space for checksum (Goodrich CCS 02).

• Use other coding techniques
  – Algebraic polynomial coding (Dean & al INFOCOM 01).
  – Probabilistic coding (Adler STOC 03).

• Change probabilistic selection
  – Adjusted probabilistic selection (Peng & al. Networking 02)
ICMP Message

- ICMP message basics
ICMP Message

• ICMP message basics
  – Packet collision issues similar to marking issues
    • Packet attenuation:
      – Far worse than packet marking (probability to generate packet is much lower).
    • Packet overwrite:
      – Does not apply.
  • Accidental Collisions: Two routers generate the same packet.
    – Very unlikely. Code size is much bigger.
  • Voluntary collisions: Attacker generates false ICMP packets.
    – Much easier than with packet marking.
ICMP Message

  - Basic idea:
    - Capture packets with very low probability (1/20000).
    - Send ICMP message including captured packet to the destination.
      - Source of packet is capturing router.
  - Packet format (based on version 3 of the draft):
    - Structure of packet:
      - Type
      - Code=0
      - Checksum
      - Message body
      - Length of message body
ICMP Message

  - Tag values:

  0x01  Back Link
  0x02  Forward Link
  0x03  Interface Name
  0x04  IPv4 Address Pair
  0x05  IPv6 Address Pair
  0x06  MAC Address Pair
  0x07  Operator-Defined Link Identifier
  0x08  Timestamp
  0x09  Traced Packet Contents
  0x0A  Probability
  0x0B  RouterId
  0x0C  HMAC Authentication Data
    0x0D  Key Disclosure List
    0x0E  Key Disclosure
    0x0F  Public-Key Information
ICMP Message

  - Authentication scheme

- Key n is authenticated using public key infrastructure.
ICMP Message

- Existing ICMP message approaches:
  - ICMP traceback (itrace, Bellovin 2000-2003)
    - Itrace IETF wg (2000)
    - Intention based traceback (Mankin & al. 2001)
    - Reverse traceback (Barros 2000)
    - Active traceback (Yamada 2002)
  - Mostly dead since mid-2002.
  - Regain activity since beginning 2003.
    - Version 3 of original draft.
  - Closed beginning 2004.
  - Still people are working on improving ICMP traceback
Trace back flows

• Basics
Trace back flows

• Existing traceback approaches
  – Reactive
    • Manual
      – Input debugging (Widely used in operators networks).
        » Netflow, sflow, cflow, ...
        » Various improvements in ACLs.
        » IP packet tracer (Cisco).
    • Automated
      – Without pattern re-evaluation:
        » Dostrack, (MCI 1997).
        » Centertrack (UUnet 1999).
        » ICMP backscatter (UUnet, 2002).
        » Backhacking (Burch & al. 1999).
      – With pattern re-evaluation (Pushback, Floyd & al. 2001).
Trace back flows

- Existing traceback approaches
  - Reactive
    - Hop by hop.
    - Edge tracking.
      - Mainly extensions of hop by hop techniques.
  - Preventive
    - Sample packet capture
      - Trajectory sampling (Duffield & al. 2001)
    - General packet capture
      - Hash based IP traceback (Snoeren & al. 2002)
Reactive approaches

• Manual flow tracking
  – Basic idea:
    • From a description of the traffic generating the attack
      – Destination, specific transport protocol, specific packet headers.
    • Find the neighbours from which most of the traffic is coming.
      – Most of: Parameter to be determined on a case by case basis.
  – Find the neighbours:
    • Most links in operator networks are point to point.
    • Find incoming interface ⇔ find the neighbours.
Reactive approaches

- Manual flow tracking
  - Find incoming interfaces.
- Flow capture functionality
  - Several flavours:
    » Juniper: cflowd.
    » Foundry networks: Sflow.
    » Cisco: Netflow.
    » Howto: Activate netflow on a specific interface.

Retrieve incoming interface for flows matching pattern.

<table>
<thead>
<tr>
<th>src_ip</th>
<th>dst_ip</th>
<th>in_if</th>
<th>out_if</th>
<th>s_port</th>
<th>d_port</th>
<th>pkts</th>
<th>bytes</th>
<th>prot</th>
<th>src_as</th>
<th>dst_as</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.xx.xxx.69</td>
<td>194.yyy.yyy.2</td>
<td>29</td>
<td>49</td>
<td>1308</td>
<td>77</td>
<td>1</td>
<td>40</td>
<td>6</td>
<td>xxx</td>
<td>ddd</td>
</tr>
<tr>
<td>192.xx.xxx.222</td>
<td>194.yyy.yyy.2</td>
<td>29</td>
<td>49</td>
<td>1774</td>
<td>1243</td>
<td>1</td>
<td>40</td>
<td>6</td>
<td>xxx</td>
<td>ddd</td>
</tr>
<tr>
<td>192.xx.xxx.108</td>
<td>194.yyy.yyy.2</td>
<td>29</td>
<td>49</td>
<td>1869</td>
<td>1076</td>
<td>1</td>
<td>40</td>
<td>6</td>
<td>xxx</td>
<td>ddd</td>
</tr>
<tr>
<td>192.xx.xxx.159</td>
<td>194.yyy.yyy.2</td>
<td>29</td>
<td>49</td>
<td>1050</td>
<td>903</td>
<td>1</td>
<td>40</td>
<td>6</td>
<td>xxx</td>
<td>ddd</td>
</tr>
<tr>
<td>192.xx.xxx.54</td>
<td>194.yyy.yyy.2</td>
<td>29</td>
<td>49</td>
<td>2018</td>
<td>730</td>
<td>1</td>
<td>40</td>
<td>6</td>
<td>xxx</td>
<td>ddd</td>
</tr>
</tbody>
</table>
Reactive approaches

• Automated flow tracking.
  – Pushback (Floyd & al. 2001)
    • Implemented in routers. (no remote configuration)
    • Combines 3 phases:
      – (Detection, pattern elaboration), Tracking, Suppression.
  • Detection.
    – Monitors line-card output queues (output queues recover traffics from several incoming line-cards).
    – When packet loss bypass a given level, we recover rejected packets.
      » Rejected packets are the cause of the problem (independently from number of packets).
      Small flows may cause high rejection rate.
      » Packets recovered constitute a fair representation of rejected packets.
    – From recovered packets we can generate patterns describing aggregates.
      » Select the aggregate generating most of the packet loss.
Reactive approaches

• Automated flow tracking.
  – Pushback (Floyd & al. 2001)
  • Tracking (local):
    – From the general packet loss rate $r$ and number of packets recovered we can estimate the packet rate matching the pattern ($p$).
    – Compute reasonable rate limit of the aggregate : $p’ = p - p \cdot r$.
    – Compute the share of each incoming interface ($p’_1…p’_n$).
  • Suppression:
    – Select interface with the largest contribution $I_1…I_k$.
    – Rate limit the aggregate matching the pattern on each contributing interface $I_j$ to $p’_i$.
    – Note that the rate limit does not alter conditions for sender/receiver !
Reactive approaches

- Automated flow tracking.
  - Pushback (Floyd & al. 2001)
    - Tracking (remote):
      - For each interface $I_1 \ldots I_k$ determine adjacent router $R_1 \ldots R_k$
      - Send aggregate description (pattern) and rate limit to routers $R_1 \ldots R_k$.
      - $R_i$ has to re-evaluate aggregate pattern.
        » Sharpen destination address definition to only include traffic headed to the outbound router:
          195.269/16 -> 195.269.153/26
      - $R_i$ performs local tracking, suppression and remote tracking operations.
**Reactive approaches**

- Automated flow tracking.
  - Pushback (Floyd & al. 2001)
    - Problems:
      - When do you stop tracking?
        » No more router to call.
        » Depth specified in tracking request.
      - When do you stop rate limiting?
        » Expiration time specified in tracking request.
      - How do you know the attack has not stopped?
        » R must re-evaluate rate-limit based on:
          * Local packet rejection rate on incoming rate limited interfaces.
          * Remote packet rejection rate on remote incoming rate limited interfaces.
          * Packet loss rate on outgoing interface.
Preventive Approaches

• General packet capture (Snoeren & al. SIGCOMM 01)
  – Outcome:
    • IP packet Traceback (IPPT, partridge, 2002)
    • Plan to form IETF wg (beginning 2002).
    • An Architecture for IP Packet Tracing (draft, Keeny, 2002).
    • However change of chair in itrace end 2002 seemed to have soften the will to create a new wg on the same topic.
  – Basic ideas:
    • Keep all packets. Don’t sample traffic.
    • Don’t send packets to collector. Keep packets on routers.
    • Problems:
      – 10 line cards routers x 100Mpkts/s = 1Gpkts/s.
      – Average: 300 bytes/pkt = 300G bytes/s to store !
      – Keep data 60 s ⇒ 1.8Tbytes to store.
Preventive Approaches

- General packet capture (Snoeren & al. SIGCOMM 01)
  - Reducing data size.
    - Keep only invariant part of the packet (I).
    - Idea: Use a bloom filter.

\[
\begin{align*}
H_1(I) &= 1000110001\ldots\, 1 \\
H_i(I) &= 0100010001\ldots\, 0 \\
H_k(I) &= 1100010101\ldots\, 1
\end{align*}
\]
Preventive Approaches

- General packet capture (Snoeren & al. SIGCOMM 01)
  - Reducing data size.
    - Keep only invariant part of the packet (I).
    - Idea: Use a bloom filter.

\[
\begin{align*}
H_1(I_j) &= 0010110001...1 \\
H_i(I_j) &= 1110010001...0 \\
H_k(I_j) &= 0000010101...1
\end{align*}
\]
Preventive Approaches

- General packet capture
  - Reducing data size.
    - After n packets the probability to have a collision is:
      \[ P = \left(1 - \left(1 - \frac{1}{2^m}\right)^{k \cdot n}\right)^k \]
    - \( k = 3, m = 4, n = 3 \)
    - \( P = 8.5\% \).
    - Average data reduction ratio:
      \[ N = 300 \times 8 \times 3/2^4 = 450 \]
Preventive Approaches

• General packet capture (Snoeren & al. SIGCOMM 01)
  – Tracking architecture.
  • 3 levels
    – Area Managers: STM (SPIE Traceback Manager).
    – Collectors: SCAR (SPIE Collection and Reduction Agents)
    – Routers: DGA (Data Generation Agents)
DoS mitigation techniques

- Usually divided in four phases:
  - Prevention
  - Detection
  - Tracking
  - Suppression
  - (Post Mortem)
Suppressing attacks

- At Victim premises
  - Filtering routers.
    - TCP_Interceptor (Cisco)
    - Firewalls filtering well known DoS attack signatures.

- On operator network
  - Redirection.
  - Ingress filtering/rate limiting.
    - ACLs, CAR.
TCP Intercept

• Basic idea:
  • Limit the number of TCP SYN requests directed to a host.
  • Queue TCP SYN request

  – Problems in the core/edge:
    • Are you sure you are receiving ACKs from the destination?
    • Delay. Much higher than in a LAN.
      – Queue size has to be much larger.
      – Synchronisation effect.
Redirection

• Steps
  – Choose a not so important blackhole router.
  – Modify address-name mapping on customer side.
    • DNS update.
  – Announce more specific route to blackhole.
    • Using BGP or IGP.
  – Make sure route does not leak.
    • Configure appropriate route-maps on peering points.
Redirect

• Steps
  – Choose a not so important blackhole router.
  – Modify address-name mapping on customer side.
    • DNS update.
  – Announce more specific route to blackhole.
    • Using BGP or IGP.
  – Make sure route does not leak.
    • Configure appropriate route-maps on peering points.
Dropping/Rate limiting

- **Access Control lists**
  - Set of conditions carrying on packet headers values.
  - Action (permit/drop).
- **Rate limiting**
  - Use ACL to specify traffic. Actions more elaborate (drop, mark, permit).
  - Token Bucket (Burst Size, Mean Rate) to decide if flow is too aggressive.
  - No buffers involved.
  - Can be distributed using BGP.
Existing products

- Usually combine several mitigation phases.
- Distributed in nature.
- Some are used by network operators
  - Eg: AT&T (Riverhead), Qwest (Arbor), TELUS (Arbor).
  - Some operators develop in house products (eg: Sprint, France Telecom).
- Main companies:
  - Riverhead networks (now bought by Cisco system).
  - Arbor networks.
  - Mazu networks.
  - Many others.
- Often funded/spin offs of big router companies (Cisco/Juniper).
Existing products

- Riverhead networks.
  - Used by AT&T.

Monitors traffic (BGP)
In line
- anti-spoofing,
- anomaly recognition,
- protocol analysis,
- rate limiting

Monitors copy of traffic
Builds traffic Profiles
Detects changes in traffic profile
Alerts RG when attack is detected
Existing products

- Sprint cleaning center (Agarwal & al., Sprint lab TR, 2004)
Final words

- Will we see DoS mitigation measures implemented in routers?
  - Technical perspective:
    - A lot of tools are already there.
      - ACLs, Netflow, NBar, QoS services, uRPF ...
    - Most mitigation schemes make use of them.
    - But mitigators also use some more complex tools:
      - Pattern matching, stateful filtering, protocol parsing, normalisation, packet marking …
Final words

- Will we see DoS mitigation measures implemented in routers?
  - Technical perspective:

Source: Mckeown, HPSR 2002
Final words

- Will we see DoS mitigation measures implemented in routers?
  - Economical perspective:
    - Carriers point of view.
      - DoS benefit network operators ...
        » Customer pay for traffic received.
        » Customer pay to stop attacks.
        » Operator pays for outbound-inbound ratio.
      - … that are able to stop at the edges ...
        » Traffic uses unnecessary resources.
      - … BIG attacks.
        » BIG attacks target mostly BIG customers.
        » BIG attacks are easier to stop and do not occur so often.
    - However!
      » DoS often associated with poor QoS by smaller customers.
      » no DoS mitigation scheme by operator pushes customer to competitors.
Final words

*Cleverness has never been seen associated with long delays.*

Sun Tzu, *The Art of War.*