Scalable Denial of Service Attack Prevention on the Internet

Kihong Park  Hyojeong Kim  Bhagya Bethala
Network Systems Lab
Department of Computer Sciences
Purdue University

Team: Ali Selcuk, Humayun Khan, Heejo Lee
Outline

- Background & set-up [Kihong Park]
- Proactive & reactive protection performance [Bhagya Bethala]
- Network security tools [Hyojeong Kim]
Network Security Projects at NSL

Scalable defenses for:

- Distributed denial of service (DDoS) attack prevention
- Virus/worm attack prevention

Supported by:

[DARPA] [XEROX] [intel] [Electronics and Telecommunications Research Institute] [CERIAS]
Distributed DoS Attack

- Source localization or traceback

→ spoofing: forge IP source address
→ hinders source localization & delays reactive response
→ more and more against network infrastructure
Virus/Worm Attack

David Moore (dmoore@caida.org)
Solution Approach

- **DDoS**: route-based filtering
  
  Where are you coming from? “unde venis”
  
  Where are you headed? “quo vadis”
  
  → if the packet lies, discard

- **Virus/worm**: content-based filtering
  
  → if the packet contains worm/virus, discard
Objective

- **Containment**
  - DDoS: prevent attack packets from reaching their targets in the first place
  - Worm/virus: stop spreading in its tracks
    → proactive protection

- **Traceback**
  - DDoS: when infeasible to do so, locate attack source
  - Worm/virus: adapt to new viruses
    → reactive protection
Key Issue

- Partial deployment & effective placement

→ distributed filtering vs. local filtering (aka firewall)
Performance Evaluation
Internet Connectivity: Power-law Nature

Key features exploited by placement strategy

- Bias toward high degree nodes
  - locally star-like topology
  - efficiency
  - simple
Power-law Connectivity

- Power-law vs. random graph

\[ \Pr[\text{deg}(x) = k] \approx k^{-\alpha} \]

\[ \Pr[\text{deg}(x) = k] \approx 2^{-k} \]
Demo: DDoS Attack Configuration
Demo: Without DPF - Snapshot 1
Demo: Without DPF - Snapshot 2
Demo: With DPF - Snapshot
Results: DDoS Attack

Protection  ~99%

Traceback

4 or less
Worm Attack: Containment Pockets

- Containment Pocket: Definition

![Diagram showing containment pockets and infected nodes]
Worm Attack: Containment Pockets
Worm Attack: Containment Pockets
Worm Attack: Containment Pockets
Results: Worm Attack
Tools and System Building
Dynamic DPF Simulator

- Built using DaSSFNet
  - DaSSF
    - Scalable simulation on workstation clusters
  - DaSSFNet
    - C++ SSFNet with improved synchronization
- Parallel DML support
- Collaborative effort with David Nicol (DaSSF PI)
Dynamic DPF Simulator

- Key Extensions
  - Protocol stack:
    - DPF-lookup, IP, TCP, UDP
    - BGP-4, DPF-update
    - Attack configuration, traffic generation suite
    - Measurement subsystem
  - Meta-DML
    - Automatic DML script generation
    - Topology-aware partitioning scheme
  - Trace-driven visualization tool
Dynamic DPF Simulator: Architecture

Network Partition
Topology
Protocol Stack
Attack Configuration

Applications
DPF Update
Socket API
BC
UDP
IP

DaSSF Kernel
MPI
DML

Meta-DML

CBR, Poisson, self-similar, MMPP, file transfer
{attackers, traffic generators, fault generators, …}
Dynamic DPF Simulator: Demo

- Network Topology
  - 300-node subgraph of 1997 Internet AS topology
- DDoS Attack Configuration
  - 106 attackers / 1 victim
  - IP source address spoofing
- Filter deployment
  - 15 nodes (5%)
- Timeline
  - Setup: BGP & DPF convergence
  - Synchronized Attack
  - Attack Duration: 4.0 sec

(without DPF)
Memory and Speed-up Benchmarks

- Workstation cluster
  - x86 PCs, 2 GHz, 1 GB, Linux 2.4+
  - DaSSFNet, MPI
Memory and Speed-up Benchmarks

- Nov. 1997, 3023-node Internet AS topology
- 10 machines
Memory and Speed-up Benchmarks

- Nov. 1997, 3023-node Internet AS topology
- 10 machines
Network Processor Prototyping

- 7-node Intel IXP1200 NP testbed
- Teja development environment
Conclusion

- DDoS attack/Internet Worm attack
  - constrained by Internet connectivity
- Distributed Packet Filtering
  - exploits Internet connectivity
- Effective performance with strategic/partial placement
- Scalable simulation tool for evaluating time dynamics
- Network processor based prototype system

http://www.cs.purdue.edu/nsl