

Trace and Stop Attacks

- ∅ Figure out which machines attacks come from
- ∅ Go to those machines (or near them) and stop the attacks
- ∅ Tracing is trivial if IP source addresses aren't spoofed
 - ∅ Tracing may be possible even if they are spoofed
- ∅ May not have ability/authority to do anything once you've found the attack machines
- ∅ Not too helpful if attacker has a vast supply of machines

1

Filtering Attack Streams

- ∅ The basis for most defensive approaches
- ∅ Addresses the core of the problem by limiting the amount of work presented to target
- ∅ Key question is:
 - ∅ What do you drop?
- ∅ Good solutions drop all (and only) attack traffic
- ∅ Less good solutions drop some (or all) of everything

2

Filtering Versus Rate Limiting

- ∅ Filtering drops packets with particular characteristics
 - ∅ If you get the characteristics right, you do little collateral damage
 - ∅ But no guarantee you have dropped enough
- ∅ Rate limiting drops packets on basis of amount of traffic
 - ∅ Can thus assure target is not overwhelmed
 - ∅ But may drop some good traffic
- ∅ Not really a hard-and-fast distinction

3

Implications of Filtering Location Choices

- ∅ Near target
- ∅ Near source
- ∅ In core

4

Implications of Filtering Location Choices

- ∅ Near target
 - ∅ Easier to detect attack
 - ∅ Sees everything
 - ∅ May be hard to prevent collateral damage
 - ∅ May be hard to handle attack volume
- ∅ Near source
- ∅ In core

5

Implications of Filtering Location Choices

- ∅ Near target
- ∅ Near source
 - ∅ May be hard to detect attack
 - ∅ Doesn't see everything
 - ∅ Easier to prevent collateral damage
 - ∅ Easier to handle attack volume
- ∅ In core

6

Implications of Filtering Location Choices

- ∅ Near target
- ∅ Near source
- ∅ In core
 - ∅ Easier to handle attack volume
 - ∅ Sees everything (with sufficient deployment)
 - ∅ May be hard to prevent collateral damage
 - ∅ May be hard to detect attack

7

How Do You Detect Attacks?

- ∅ Have database of attack signatures
- ∅ Detect anomalous behavior
 - ∅ By measuring some parameters for a long time and setting a baseline
 - ∅ Detecting when their values are abnormally high
 - ∅ By defining which behavior must be obeyed starting from some protocol specification

8

How Do You Filter?

- ∅ Devise filters that encompass most of anomalous traffic
- ∅ Drop everything but give priority to legitimate-looking traffic
 - ∅ It has some parameter values
 - ∅ It has certain behavior

9

DDoS Defense Challenges

- ∅ Need for a distributed response
- ∅ Economic and social factors
- ∅ Lack of detailed attack information
- ∅ Lack of defense system benchmarks
- ∅ Difficulty of large-scale testing

10

Sample Research Approaches

- ∅ Pushback
- ∅ Traceback
- ∅ D-WARD
- ∅ Netbouncer
- ∅ SOS
- ∅ Proof-of-work systems
- ∅ Distributed solutions
 - ∅ Cossack
 - ∅ DefCOM

11

Pushback¹

¹"Controlling high bandwidth aggregates in the network."
Mahajan, Bellovin, Floyd, Paxson, Shenker, ACM CCR, July 2002

- ∅ Goal: Preferentially drop attack traffic to relieve congestion
- ∅ Local ACC: Enable core routers to respond to congestion locally by:
 - ∅ Profiling traffic dropped by RED
 - ∅ Identifying high-bandwidth aggregates
 - ∅ Preferentially dropping aggregate traffic to enforce desired bandwidth limit
- ∅ Pushback: A router identifies the upstream neighbors that forward the aggregate traffic to it, requests that they deploy rate-limit

12

Can it work?

- ∅ Even a few core routers are able to control high-volume attacks
- ∅ Separation of traffic aggregates improves current situation
 - ∅ Only traffic for the victim is dropped
 - ∅ Drops affect a portion containing the attack traffic
- ∅ Likely to successfully control the attack, relieving congestion in the Internet
- ∅ Will inflict collateral damage on legitimate traffic

13

Advantages and Limitations

- + Routers are well equipped to handle high traffic volumes
- + Deployment at a few core routers can affect many traffic flows, due to core topology
- + Simple operation, no overhead for routers
- + Pushback minimizes collateral damage by placing response close to the sources
- Pushback only works in contiguous deployment
- Collateral damage is inflicted by response, whenever attack traffic is not clearly different than legitimate traffic
- Deployment requires modification of existing core routers and likely purchase of new hardware

14

Traceback¹

¹“Practical network support for IP Traceback,” Savage, Wetherall, Karlin, Anderson, ACM SIGCOMM 2000

- ∅ Goal: locate the agent machines
- ∅ Each packet header may carry a mark, containing:
 - ∅ *EdgeID* (IP addresses of the routers) specifying an edge it has traversed
 - ∅ The distance from the edge
- ∅ Routers mark packets probabilistically
- ∅ If a router detects half-marked packet (containing only one IP address) it will complete the mark
- ∅ Due to limited space in IP header (fragment offset field) *EdgeID* is fragmented
- ∅ Victim under attack reconstructs the path from the marked packets

15

Traceback and IP Spoofing

- ∅ Strictly speaking, traceback does nothing to stop DDoS attacks
- ∅ It only identifies attackers’ true IP addresses
 - ∅ Within a subnet, at least
- ∅ If IP spoofing were not possible in the Internet, traceback would not be necessary
- ∅ There are approaches under development to largely prevent IP spoofing

16

Can it work?

- ∅ Incrementally deployable, a few disjoint routers can provide beneficial information
- ∅ Moderate router overhead (packet modification)
- ∅ A few thousand packets are needed even for long path reconstruction
- ∅ Does not work well for highly distributed attacks
- ∅ Path reassembly is computationally demanding, and is not 100% accurate:
 - ∅ Path information cannot be used for legal purposes
 - ∅ Routers close to the sources can efficiently block attack traffic, minimizing collateral damage

17

Advantages and Limitations

- + Incrementally deployable
- + Effective for non-distributed attacks and for highly overlapping attack paths
- + Facilitates locating routers close to the sources
- Packet marking incurs overhead at routers, must be performed at slow path
- Path reassembly is complex and prone to errors
- Reassembly of distributed attack paths is prohibitively expensive
- Packet marks can be forged by the attacker
- Only identifies the agent machines

18

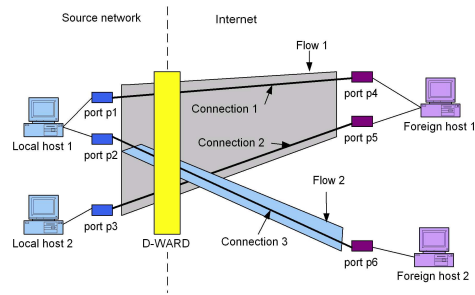
D-WARD¹

¹"Attacking DDoS at the source." Mirkovic, Prier, Reiher, ICNP 2002

- ∅ Goal: detect attacks, reduce the attack traffic, recognize and favor the legitimate traffic
- ∅ Source-end, inline defense system
- ∅ Gathers statistics on flows and connections, compares them with protocol-based models:
 - ∅ Mismatching flow statistics indicate attack
 - ∅ Matching connection statistics indicate legitimate traffic
- ∅ Dynamic and selective rate-limit algorithm:
 - ∅ Fast decrease to relieve the victim
 - ∅ Fast increase when the attack stops and on false alarms
 - ∅ Detects and forwards legitimate connection packets

19

Flows and Connections



20

Can it work?

- ∅ Extensive experiments indicate:
 - ∅ Fast detection of a wide range of attacks
 - ∅ Effective control of the attack traffic
 - ∅ Extremely low collateral damage
 - ∅ Fast removal of rate limit when attack stops
- ∅ Small processing and memory overhead
- ∅ Effectively stops attacks from deploying networks
- ∅ Only effective in actually stopping attacks if deployed at most/all potential attacking networks
 - ∅ May provide synergistic benefits with other defenses

21

Advantages and Limitations

- + Fast detection and control of wide range of attacks
- + Extremely low collateral damage
- + Low number of false positives
- + Stops attacks as soon as possible
- Attackers can perform successful attacks from unprotected networks
- Deployment motivation is low

22

Netbouncer¹

¹"NetBouncer: Client-Legitimacy-based High-performance DDoS Filtering," Thomas, Mark, Johnson, Croall, DISCEX 2003

- ∅ Goal: detect legitimate clients and only serve their packets
- ∅ Victim-end, inline defense system deployed in front of the choke point
- ∅ Keeps a list of legitimate clients:
 - ∅ Only packets from these clients are served
 - ∅ Unknown clients receive a challenge to prove their legitimacy, several levels of legitimacy tests
 - ∅ Various QoS techniques are applied to assure fair sharing of resources by legitimate client traffic
 - ∅ Legitimacy of a client expires after a certain interval

23

Can it work?

- ∅ Successfully defeats spoofed attacks
- ∅ Ensures fair sharing of resources among clients that have proved to be legitimate
- ∅ All legitimacy tests are stateless – defense system cannot be target of state-consumption attacks
- ∅ Some legitimate clients do not support certain legitimacy tests (i.e. ping test)
- ∅ Legitimate client identity can be misused for attacks
- ∅ Large number of agents can still degrade service to legitimate clients, creating "flash crowd" effect

24

Advantages and Limitations

- + Ensures good service to legitimate clients in the majority of cases
- + Does not require modifications of clients or servers
- + Stateless legitimacy tests ensure resiliency to DoS attacks on Netbouncer
- + Realistic deployment model:
 - Autonomous solution, close to the victim
 - Attackers can perform successful attacks by:
 - Misusing identities of legitimate clients
 - Recruiting a large number of agents
 - Some legitimate clients will not be validated
 - Challenge generation may exhaust defense