#42

Tackling Network DoS on Transit Networks

David Harmelin

DANTE IN PRINT is a track record of papers and articles published by, or on behalf of DANTE. HTML and Postscript versions are available from: http://www.dante.net/pubs/dip

For more information about DANTE or *DANTE IN PRINT* please contact:

DANTE
Francis House
112 Hills Road
Cambridge CB2 1PQ
United Kingdom

Tel: +44 1223 302992
Fax: +44 1223 303005
E-mail: dante@dante.org.uk

Tackling Network DoS on Transit Networks

David Harmelin

Abstract

Denial of service attacks often cross multiple transit IP networks, like TEN-155 and GÉANT, before reaching their victims. Besides, such transit IP networks being, by definition, between end sites, carry more attacks than leaf networks in average.

It is therefore natural to expect transit IP networks to be the firsts to enable mechanisms to fight DoS attacks. Especially since, even if such networks are usually not the targets themselves, DoS attacks have a cost on resources of all networks they cross.

The following paper describes an approach to implementing such mechanisms to detect DoS attacks.

KEYWORDS: Denial of Service / DoS, transit networks.

1. Introduction and definitions

In the case of *network DoS*, quite often, targeted victims are not the only network elements to be affected by the *DoS* attack. Most *transit networks* (and their resources), between the targeted victim and the sources of the *DoS*, suffer from the attack.

Networks, used to generate *DoS* attacks, are usually, themselves, victims (misconfigured, or compromised hosts).

Most papers available, on the *DoS* phenomenon, are not targeted to *transit networks*. This document intends to fill this void.

David Harmelin is a Network Engineer in the Network Engineering and Planning team at DANTE, where he is also in charge of security-related matters. His email address is <david.harmelin@dante.org.uk>.

This document:

- is aimed at *transit networks* managers, operations desks and security personnel as a tutorial, or starting point, for addressing the *DoS* phenomenon on *transit network*.
- addresses what *transit networks* can do to trace (and reduce) the effect of generic *network DoS* traffic, transiting through their network
- does not address in detail the various types of *DoS* attacks

The following terms have the following meanings, in the scope of this document.

DoS: Denial of Service attacks. Such attacks aim to starve a resource (or resources), usually in order to make a service unavailable.

DDoS: Distributed DoS. This subset of DoS attacks involves multiple sources, usually within multiple administrative domains.

Network DoS: this refers to DoS attacks where the resource being starved is a network element, and the attack is conducted through one (or more) network(s). In all the following document, *DoS* actually refers to *network DoS*.

Transit network: a network that provides transit between other networks.

Forged source IP address: IP packets crossing IP networks carry information on the source host that originated them. Forging the source IP address in a packet consists in lying about the host that originated the packet, in order to hide it.

2. Various DoS cases:

In all the following cases:

- TGT is the targeted victim
- The attacker's purpose is that TGT becomes unreachable
- He achieves it by:
  * starving TGT's kernel capacity to deal with the traffic
  * starving the network interface's forwarding capacity of TGT
  * starving the circuit capacity between the network N and the transit network
  * starving the forwarding (or switching) capacities of N
  * starving the forwarding (or switching) capacities of the transit network

It is enough that one of those resources is successfully starved for the attack to be successful. Quite often, which resource is starved does not interest the attackers, as long as the goal is obtained (TGT being unreachable).

2.1. Ping DoS:

For simplicity reasons, it is assumed traffic is ICMP, generated with PING, but reports exist of cases where other routed protocols were used, with other traffic generators.

This case can result in a DoS if SRC1 has more forwarding capacity than TGT.

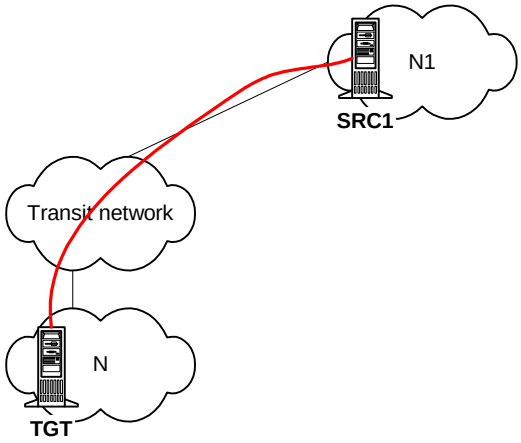

Figure 1. Ping DoS

2.2. Ping DDoS:

Here also, for simplicity reasons, it is assumed traffic is ICMP, generated with PING, but reports exist of cases where other routed protocols were used, with other traffic generators.

More sources send traffic. TGT (or a network element in N or the transit network) fails.

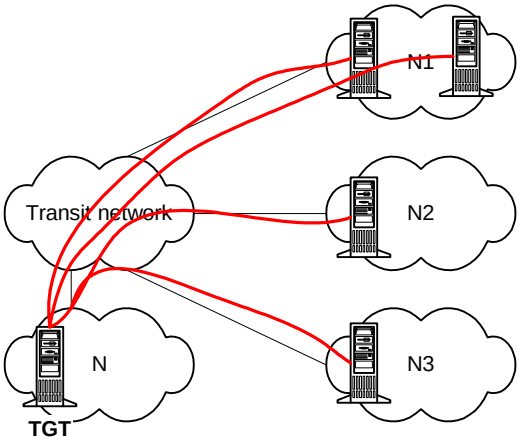

Figure 2. Ping DDoS

2.3. Smurf:

The attacker sends ICMP ECHO packets to the broadcast address of (SRC1, SRC2, SRC3, SRC4) LAN, with the source IP address set to the victim's address. As a result, all hosts in N1 send an ICMP ECHO_REPLY packet towards the victim.

This DoS attack is very simple to achieve, as the attacker only needs access to a slightly modified PING program and a list of subnets that forward network-prefix-directed broadcast.

The real problem resides in N1, which should not forward network-prefix-directed broadcast requests from the outside to the LAN. N1 is called a *smurf amplifier*.

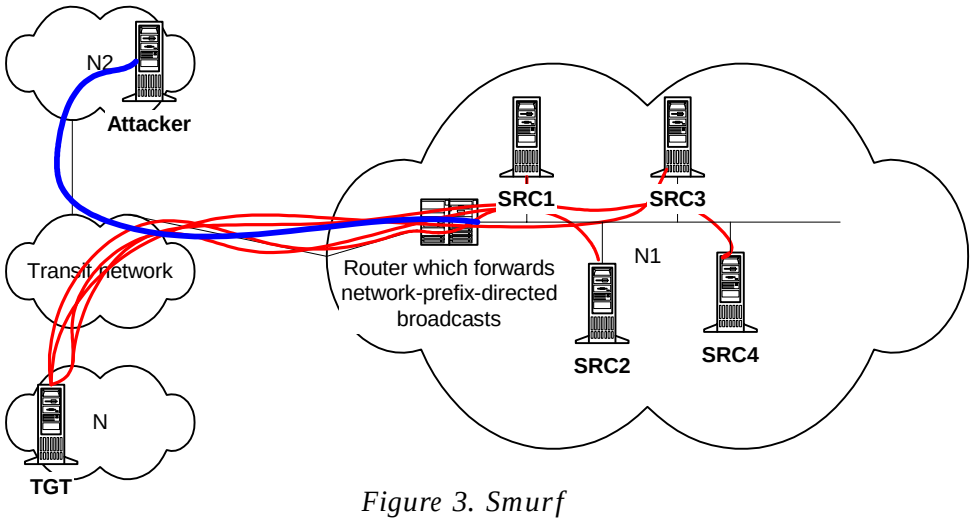

Figure 3. Smurf

2.4. DDoS, masters, handlers and slaves

Many DDoS tools exist, and the most frequent types of DoS detected consist of such attacks.

Most of them use the ICMP, UDP or TCP (SYN, ACK packets) protocols. The set up shown in Figure 4 below is just a "simple" example of what can be done.

In this case:

1. Attacker -> Master: 1 UDP packet (type of attack requested, duration, target)
2. Master -> each handler: 1 UDP packet (type of attack requested, duration, target)
3. Each handler -> each of its registered DoS Slaves: 1 UDP packet (type of attack requested, duration, target)
4. Each DoS Slave -> TGT: generates packets for the duration requested

In such a case, Master, Handlers and Slaves are usually compromised hosts, with the DoS software installed.

The attack will stop either after a timer expires, or when the attacker issues another command (1 packet) to the master.

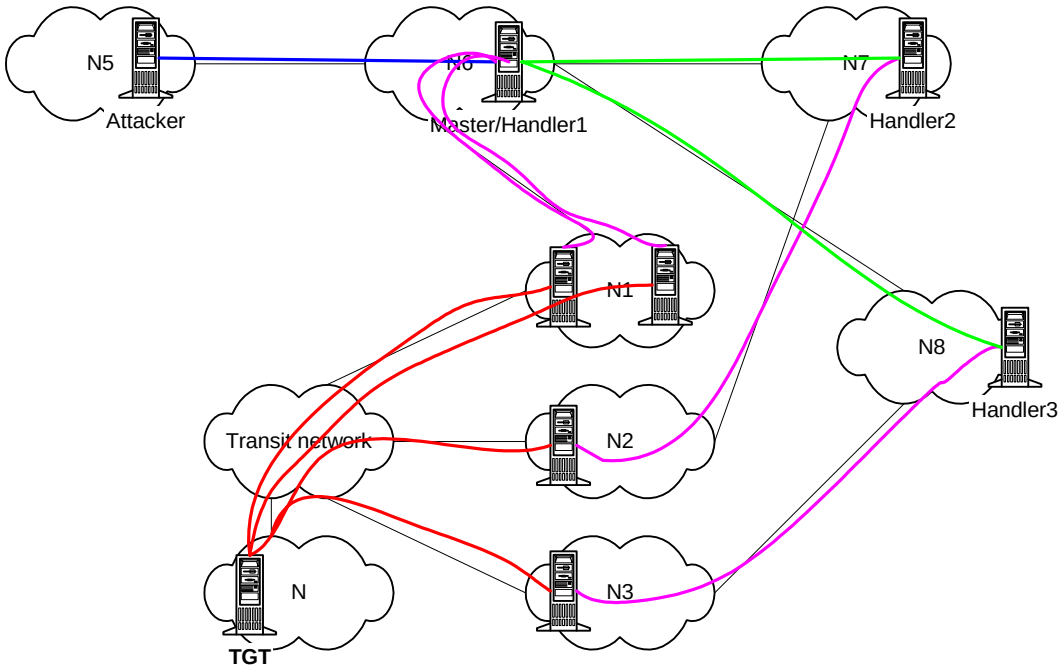

Figure 4. DDoS

In such a scenario, the attack will most likely be only detected by the transit network and the N network.

2.5. Forged DDoS:

In many cases, the DDoS applications on the hosts in N1, N2 and N3 (see example above) are generating packets with forged IP addresses (sometimes, completely random).

Due to the setup, only the transit network, N and TGT are susceptible to notice the change in the amount of traffic shipped by the attack, when it occurs.
If the source IP address field is forged, the transit network can only trace the attack up to the interfaces towards N1, N2 and N3.

Cases could occur where the attacker would use forged IP addresses to generate traffic from hosts in N1, but which those forged IP addresses would still belong to the N1 network.

Therefore, if the *transit network* traces the traffic up to the interface towards N1, it can only be up to N1's administrations to assess whether the source IP addresses are forged or not. If they are, it is also N1's responsibility to get sites within N1 not to allow forged traffic to leave their site.

3. Characteristics of DoS attacks:

3.1. Forged random source IP address:

In many cases, when the software has been installed on compromised hosts, the software will generate packets with source IP addresses forged.

When a DDoS flow with random source IP addresses is first noticed on a given router, it can only be traced backwards router after router, by looking at which interfaces the traffic actually comes from. It is doable by looking at the flow cache, or netflow exports. This is often difficult in a multi-administrative environment (more than one network involved).

Once the compromised hosts are found, the first step should be to deny forged traffic (with ANY source address) to leave the end-sites concerned.

3.2. Forged source IP addresses within the address range of the generating machine

As it is recommended to end-sites to filter egress traffic based on their LANs IP addresses, some DDoS software only generate forged IP addresses within a configurable address range (usually configured to be the /24 of the machine the software is running of).

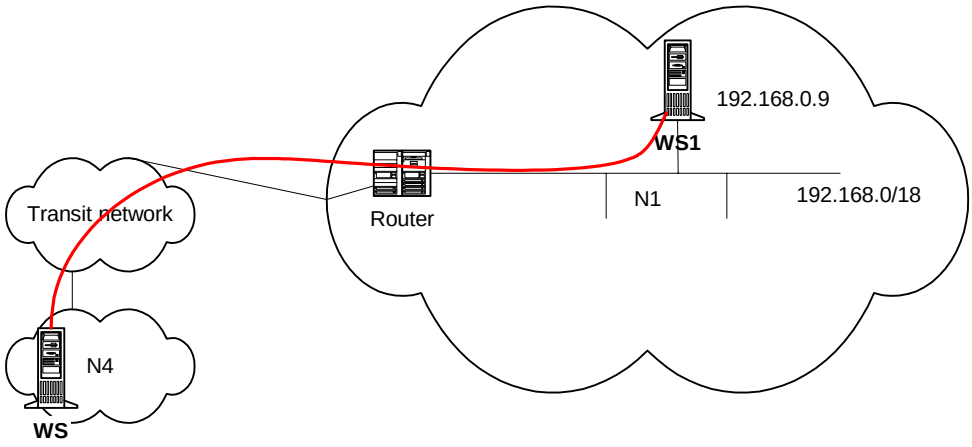

Figure 5. Forged IP addresses within prefix range of machines

In the case shown above in Figure 5, although WS1's IP address is 192.168.0.9, WS1 DDoS software will generate packets from 192.168.0/24 (for example), so that the router does not drop the traffic.

Only N1 can investigate which machine is exactly generating the traffic.

3.3. Forged source IP addresses within the address range of site

As shown on Figure 6 below, if only Router1 does egress filtering on forged source IP addresses, WS1 may only forge packets with source IP addresses from 192.168.64.0/18 or 192.168.128.0/18, in order to make it more difficult for N1 to trace the traffic to WS1.

Many DoS clients have various options that allow the attack to choose which type of spoofed packets to generate. In particular, a given compromised machine may generate various types of DoS attacks.

4. DoS attacks on transit networks:

DoS packets often cross multiple networks before reaching their victims.

Because *DDoS* attacks often involve packet generators from various sites, *transit networks* between the packet generators and the victim will often be the first administrative entity (in the IP path) to notice a DoS attack is on-going.

Also, large *transit networks* often interconnect many smaller customer networks which are, either being attacked, or used to launch attacks. As such, *transit networks* are more likely to suffer from extremely frequent (sometimes constantly) and multiple on-going DoS attacks.
As such, the DoS phenomenon influences the cost of the network, due to the frequent (sometimes constant) noise traffic, this, in turn, inducing more frequent hardware and circuit upgrades.

It is therefor essential that *transit networks* implement ways to notice, trace and reduce the DoS attacks and their effects, to reduce those attacks cost.

5. Detection tools:

5.1. Flow based:

Because *transit networks* carry a lot of traffic, which cannot be analyzed at line rate, it is recommended to take advantage of sampling, when coding flow-based tools.

Due to the uncertainty of the source IP addresses in the packets, tracing an attack often requires to check router/interface after router/interface, until the entry points of the traffic to the *transit network* are found.

It could be good practice that all *transit networks* implement similar sampling-based

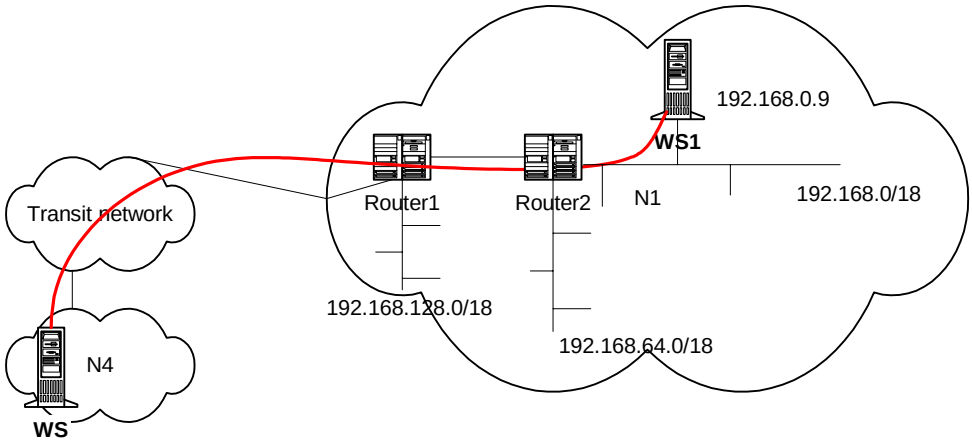

Figure 6. Forged IP addresses within prefix range of sites

logging tools, at least on their border routers, so that when alerts are received, timestamped samples can be consulted and lead to the next network in the chain, towards the packet generators of the DoS attack.

It is important to log the samples that have generated the alerts, so that further post-mortem analysis can be conducted.

With traffic analysis based on netflow, DANTE has been able to develop such a system, proving extremely efficient to generate alerts upon DoS attacks.

The tool takes advantage of the flow export functionality, available with most routers. It is recommended to acquire hardware with such functionality, when possible.

Should all networks run similar tools, post-mortem analysis all the way to the source would be easier.

5.2. itrace:

An IETF working group is working on defining a new ICMP message (ICMP traceback), also based on sampling (packets, this time) and generating information to allow end-users to know the real IP path taken by IP packets (even forged ones).

The proposal is also aimed at detecting DoS attacks and their sources.

More information is available at: http://www.ietf.org/html.charters/itrace-charter.html

5.3. Resources usage:

Tools based on resources usage:

- must make sure to monitor all resources: amount of packets forwarded, bandwidth used, cpu;
- must provide mechanisms to trace attacks AFTER they have occurred.

Further readings:

DANTE DoS resources
(http://www.dante.net/sf/dos/)

CERT – Denial of Service attacks
(http://www.cert.org/tech_tips/denial_of_service.html)

SANS - Global Incident Analysis Center - Special Notice - Consensus Roadmap for Defeating Distributed Denial of Service Attacks
(http://www.sans.org/ddos_roadmap.htm)

SANS - Help Defeat Denial of Service Attacks: Step-by-Step
(http://www.sans.org/dosstep/index.htm)

CISCO – Strategies to Protect Against Distributed Denial of Service (DDoS) Attacks
(http://www.cisco.com/warp/public/707/newsflash.html)

JANET – Investigating a Denial of Service attack (at the end site level)
(http://www.ja.net/documents/gn_ddos.pdf)

Rob Thomas - Tracking Spoofed IP Addresses
(http://www.cymru.com/%7Erobt/Docs/Articles/tracking-spoofed.html)

Craig A. Huegen - “Smurf” attack information
(http://www.pentics.net/denial-of-service/white-papers/smurf.cgi)

Dietrich, Long, Dittrich – Analyzing distributed denial of service tools – The Shaft case
(http://netsec.gsfc.nasa.gov/~spock/lisa2000-shaft.pdf)

Dave Dittrich - Distributed Denial of Service (DDoS) Attacks/tools
(http://staff.washington.edu/dittrich/misc/ddos)

NIPC / STAU – find Distributed Denial of Service
(http://www.nipc.gov/warnings/alerts/1999/README)

Richard A Steenbergen - Some thoughts on modern denial of service
(http://www.e-gerbil.net/ras/dos.txt)