# DDoS-blocker:

# Detection and Blocking of Distributed Denial of Service Attack

Sugih Jamin
EECS Department
University of Michigan

*jamin@eecs.umich.edu*

## Internet Design Goals

Key design goals of Internet protocols:

- resiliency
- availability
- scalability

Security has not been a priority until recently.

## Security Attacks

Two types of security attacks:

1. Against information content: secrecy, integrity, authentication, authorization, privacy, anonymity
2. Against the infrastructure: system intrusion, denial of service

## Counter Measures

Fundamental tools:

- Content attack: cryptography
- Intrusion detection: border checkpoints (firewall systems) to monitor traffic for known attack patterns
- Denial of service: **no effective counter measure**

## Denial of Service (DoS) Attack

An *attacker* inundates its *victim* with otherwise legitimate service requests or traffic such that victim's resources are overloaded and overwhelmed to the point that the victim can perform no useful work.

## Distributed Denial of Service (DDoS) Attack

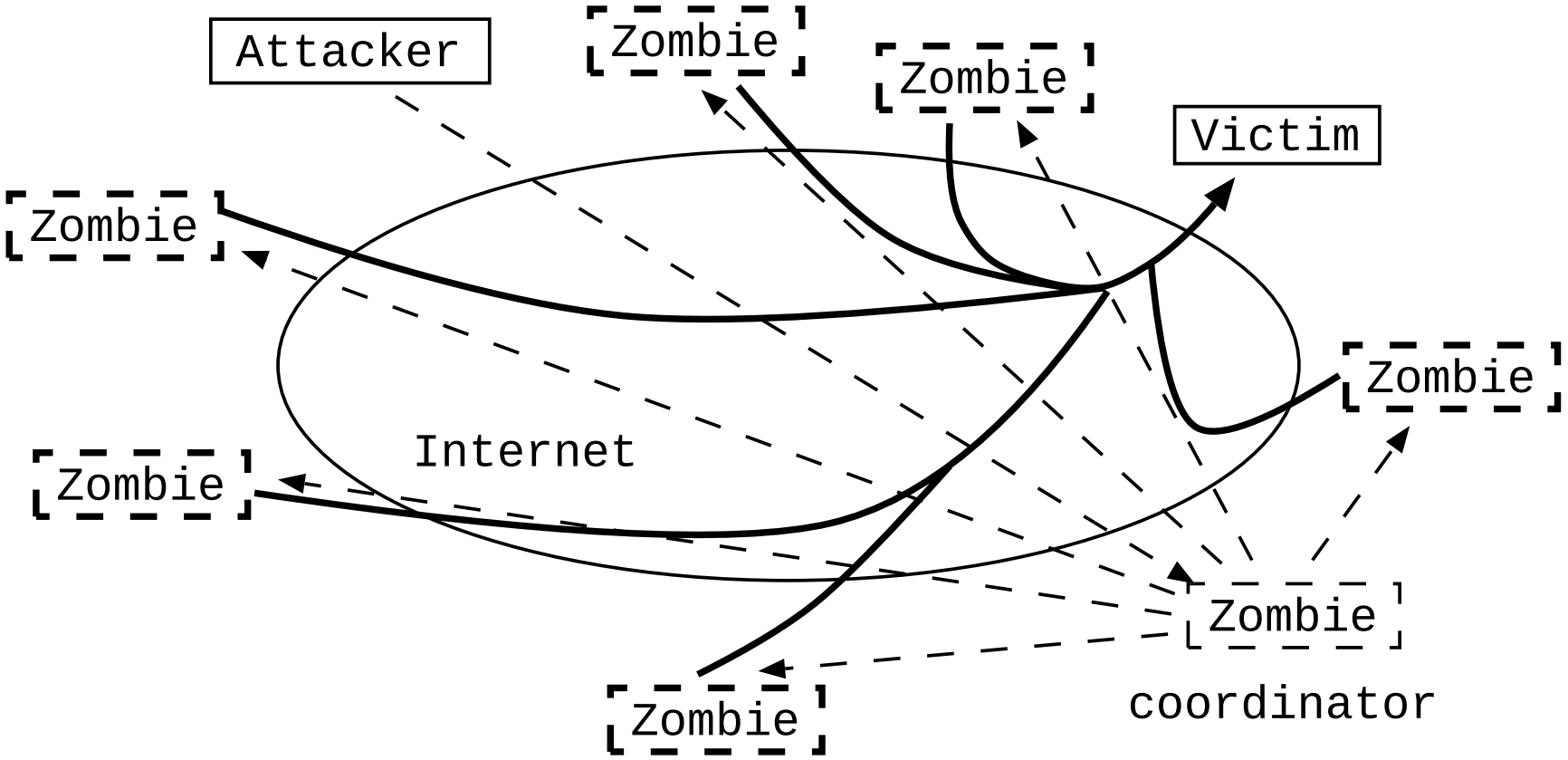
A newly emerging, particularly virulent strain of DoS attack enabled by the wide deployment of the Internet.

Attacker commandeers systems (*zombies*) distributed across the Internet to send correlated service requests or traffic to the victim to overload the victim.

## DDoS Example

February 7th and 8th, 2000: several large web sites such as Yahoo!, Amazon, Buy.com, eBay, CNN.com, etc. were taken offline for several hours, costing the victims several millions of dollars.

# Mechanism of DDoS [Bellovin 2000]



## The Making of a Zombie

Zombies, the commandeered systems, are usually taken over by exploiting program bugs or backdoors left by the programmers.

Zombie intrusion done by sending harmless looking legitimate code or data containing hidden malicious code (Trojan Horse) to the vulnerable candidate.

Once a host is taken over and made a zombie, the malicious code will run as a background process (*daemon*) that performs the actual attack.

(DDoS problem expected to get worse as more home systems come online and remain connected on DSL or cable modems.)

## Weapons of Choice

**Spoofing:** faking the return address of a service request.

**SYN flood:** ties up access to a web server with a large number of false connections with spoofed client addresses.

**Smurfing:** requests a large number of machines on the Internet to send an echo of a message to a spoofed return address.

New DDoS tools uses encryption to make detection and tracking of the perpretator harder.

## Current Approaches to Counter DoS

Rely on the criminal justice system

Practice good computer hygiene

Limit usage and convenience of usage

## Tracking the Perp

The US Computer Fraud and Abuse Act makes it a crime to knowingly transmit a program or command that intentionally damages a computer.

To help track down perpetrators, systems on the Internet keep traffic logs as a standard operating procedure.

## Good Computer Hygiene

Install operating system patches that fix bugs and close backdoors.

Install intrusion detection software to prevent systems from being taken over.

Continually update intrusion detection software to incorporate all the latest known attack patterns.

Regularly monitor system logs, network traffic, and system configurations to detect anomalies.

## Limiting Harm

Install traffic filters to detect spoofed addresses.

Install new traffic limiting (congestion control) mechanisms in routers.

Deploy new versions of Internet protocols that incorporate authentication.

## Limitations of Current Approaches

Intrusion detection effective only against known attack patterns.

Traffic logs generate large amount of data during normal operation.

Traffic filters and traffic limiting mechanisms are expensive to deploy, both financially and performance wise.

## Limitations of Current Approaches (cont)

There is a fundamental trade-off between security vs. convenience: at one extreme, the most secure, least convenient system is not networked and is placed in a secure locked room.

**Post facto tracking down of DDoS perpetrator does not help the victim.**

## The Challenge

Can one detect correlated traffic streams across the Internet in a user-transparent manner, in real-time, before they merge into a denial of service attack?

## Solution Requirements

Fundamental requirement: preserve the original design goals of the Internet, that the Internet continues to be resilient, available, and scalable.

More specifically:

1. be transparent to legitimate use of the Internet,
2. detect anomalies in time to devise and deploy effective counter measures to neutralize the attack,
3. require zero or minimal coordination between monitoring/control sites, and
4. be self-configuring with zero or minimal human interaction.

## Solution Parameters

Internet traffic is very bursty, with self-similarity detected on traffic load over various timescales

Internet topology continues to grow and expand

There is no central authority to dictate uniform deployment of any mechanism

## Summary

Analogy to river networks: Can one detect increases in water volume at tributaries, maybe based on historical data, to predict an upcoming inundation downstream on the main stream?

Furthermore, can one do this in real-time, with minimal coordination, and minimal false positives?

Sugih Jamin (jamin@eecs.umich.edu)

## Further Readings

Garber, L., “Denial-of-Service Attacks Rip the Internet,” *IEEE Computer*, April 2000, pp. 12-17.

Bellovin, S.M., “Distributed Denial of Service Attacks,” http://www.research.att.com/~smb, 2000.