DENIAL OF SERVICE

DoS & DDoS ATTACK TECHNIQUES

Master denial-of-service attack methodologies, distributed attacks, botnet operations, auditing tools, and defensive countermeasures.

2026 Edition

TABLE OF CONTENTS

1. INTRODUCTION TO DENIAL-OF-SERVICE

Denial-of-Service (DoS) attacks aim to make systems, services, or networks unavailable to legitimate users by overwhelming resources or exploiting vulnerabilities. These attacks can cause significant financial and reputational damage to organizations.

What is a Denial-of-Service Attack?

A Denial-of-Service attack is an attempt to disrupt the normal functioning of a targeted server, service, or network by overwhelming it with a flood of traffic or sending information that triggers a crash. The goal is to deny legitimate users access to the resource.

DoS ATTACK OBJECTIVES

- Disrupt business operations and services
- Cause financial losses (downtime costs)
- Damage reputation and customer trust
- Distract security teams during other attacks
- Extortion and ransom demands
- Political or ideological statements
- Competitive sabotage
- Personal revenge or grievance

Impact of DoS Attacks

Financial Impact

- Lost revenue during downtime
- Emergency response costs
- Recovery and remediation expenses
- Potential regulatory fines
- Customer compensation
- **Average Cost:** $20,000 - $40,000 per hour of downtime

Operational Impact

- Service unavailability
- Employee productivity loss
- Resource exhaustion
- Cascading failures
- Delayed projects and deadlines

Reputational Impact

- Customer dissatisfaction
- Loss of trust
- Media attention
- Competitive disadvantage
- Long-term brand damage

DoS ATTACK CATEGORIES

By Attack Vector

- ▶ **Volumetric Attacks –** Flood bandwidth with massive traffic
- ▶ **Protocol Attacks –** Exploit protocol weaknesses
- ▶ **Application Layer –** Target specific applications

By Source

- ▶ **Single Source (DoS) –** One attacking machine
- ▶ **Distributed (DDoS) –** Multiple attacking machines
- ▶ **Reflected –** Using third-party systems
- ▶ **Amplified –** Multiplying attack traffic

ATTACK METRICS

Measuring Attack Strength

- ▶ **Bits per Second (bps) –** Bandwidth consumption
  `Gbps, Tbps for large attacks`
- ▶ **Packets per Second (pps) –** Processing load
  `Millions/billions of packets`
- ▶ **Requests per Second (rps) –** Application load
  `HTTP requests for L7 attacks`
- ▶ **Connections per Second –** State table exhaustion

Attack Scale Reference

- ▶ Small: < 1 Gbps
- ▶ Medium: 1-10 Gbps
- ▶ Large: 10-100 Gbps
- ▶ Massive: 100+ Gbps
- ▶ Record attacks: 3.47 Tbps (2022)

WHO LAUNCHES DoS ATTACKS?

- ▶ **Hacktivists –** Political/social motivation
- ▶ **Cybercriminals –** Extortion, ransom
- ▶ **Nation-States –** Cyberwarfare, disruption
- ▶ **Competitors –** Business sabotage
- ▶ **Script Kiddies –** Notoriety, testing tools
- ▶ **Disgruntled Insiders –** Revenge
- ▶ **DDoS-for-Hire –** Booter/stresser services

2. DoS vs DDoS ATTACKS

Understanding the distinction between single-source DoS and distributed DDoS attacks is essential for implementing appropriate defenses.

DENIAL-OF-SERVICE (DoS)

Single-Source Attacks

Traditional DoS attacks originate from a single system targeting a victim. While simpler, they have limitations.

DoS Characteristics

- Single attacking machine
- Limited bandwidth/resources
- Easier to trace and block
- Often exploits specific vulnerabilities
- May be sufficient for small targets

DoS Advantages (for attacker)

- Simple to execute
- No botnet required
- Direct control over attack
- Lower setup complexity

DoS Limitations

- Limited attack volume
- Single point of failure
- Easy to block source IP
- Traceable to source
- Ineffective against large targets

DISTRIBUTED DENIAL-OF-SERVICE (DDoS)

Multi-Source Attacks

DDoS attacks leverage multiple systems simultaneously, making them far more powerful and difficult to mitigate.

DDoS Characteristics

- Multiple attacking sources
- Massive combined bandwidth
- Difficult to distinguish from legitimate traffic
- Challenging to block all sources
- Often uses compromised systems (botnet)

DDoS Attack Sources

- **Botnets –** Networks of compromised devices
- **Voluntary Networks –** Hacktivism (LOIC)
- **Reflected Traffic –** Spoofed source addresses
- **Cloud Resources –** Rented/compromised VMs
- **IoT Devices –** Insecure connected devices

DDoS ATTACK ARCHITECTURE

Traditional Botnet Model

Hierarchical command and control structure:

- Attacker controls Command & Control (C2) server
- C2 communicates with bot handlers
- Handlers control groups of bots
- Bots (zombies) execute attack commands
- Attack targets victim infrastructure

Modern Attack Infrastructure

- Peer-to-peer botnets (no central C2)
- Cloud-based attack platforms
- DDoS-as-a-Service offerings
- Bulletproof hosting
- Cryptocurrency payments for anonymity

DDoS ATTACK TYPES OVERVIEW

Layer 3/4 Attacks (Network/Transport)

- Target network infrastructure
- Consume bandwidth and connections
- Measured in Gbps/Tbps and pps
- Examples: UDP flood, SYN flood, ICMP flood

Layer 7 Attacks (Application)

- Target application services
- Consume server resources
- Measured in requests per second
- Examples: HTTP flood, Slowloris

COMPARING DoS AND DDoS

DoS

- Source: Single machine
- Scale: Limited (Mbps-Gbps)
- Complexity: Low
- Cost: Minimal
- Detection: Easier
- Mitigation: Block source IP

DDoS

- Source: Thousands/millions of machines
- Scale: Massive (Tbps possible)
- Complexity: High
- Cost: Botnet or service required
- Detection: Harder (distributed)
- Mitigation: Complex, multi-layered

3. VOLUMETRIC ATTACKS

Volumetric attacks aim to consume all available bandwidth between the target and the internet. These are the most common DDoS attacks, measured in bits per second (bps).

VOLUMETRIC ATTACK OBJECTIVES

- Saturate target's internet bandwidth
- Overwhelm network infrastructure
- Exhaust ISP resources
- Cause complete network unavailability
- Simple but effective approach

UDP FLOOD

Attack Mechanism

Sends massive amounts of UDP packets to random ports on the target system.

How UDP Flood Works

- Attacker sends UDP packets to random ports
- Target checks for listening application
- No application = ICMP Destination Unreachable
- Volume overwhelms target's capacity
- Legitimate traffic cannot get through

UDP Flood Characteristics

- Connectionless - no handshake required
- Easy to spoof source addresses
- High packet rate possible
- Stateless, low overhead for attacker

```
hping3 --udp -p 53 --flood target_ip
```

ICMP FLOOD (PING FLOOD)

Attack Mechanism

Overwhelm target with ICMP Echo Request (ping) packets.

How ICMP Flood Works

- Send massive ICMP Echo Requests
- Target must process each request
- Echo Reply consumes outbound bandwidth
- Both inbound and outbound saturation

```
hping3 --icmp --flood target_ip
```

Smurf Attack (Legacy)

Amplified ICMP attack using broadcast:

- Send ICMP to network broadcast address
- Spoof source as victim's IP
- All hosts reply to victim
- Amplification based on network size

📌 Mostly mitigated in modern networks

DNS FLOOD

Attack Mechanism

Flood DNS servers with requests to exhaust resources.

DNS Flood Types

- ▶ **Direct DNS Flood –** Massive queries to DNS server
- ▶ **DNS Amplification –** Reflected, amplified attack
- ▶ **Random Subdomain –** Queries for non-existent records
- ▶ **NXDOMAIN Attack –** Exhaust cache with invalid domains

Impact

- ▶ DNS server overwhelmed
- ▶ Legitimate lookups fail
- ▶ Services become unreachable
- ▶ Cascade effect on dependent services

FRAGMENTATION ATTACKS

IP Fragmentation

Exploit packet reassembly process:

- ▶ **Teardrop –** Overlapping fragment offsets
- ▶ **Ping of Death –** Oversized ICMP packets
- ▶ **Fragment Flood –** High volume of fragments
- ▶ Target exhausts memory for reassembly
- ▶ CPU intensive to process

VOLUMETRIC ATTACK CHARACTERISTICS

Common Features

- ▶ High bandwidth consumption
- ▶ Often spoofed source IPs
- ▶ Measured in Gbps/Tbps
- ▶ May use amplification
- ▶ Target: Network layer
- ▶ Defense: Upstream filtering, scrubbing

Detection Indicators

- ▶ Sudden bandwidth spike
- ▶ Traffic from unusual sources
- ▶ Single protocol dominance
- ▶ Geographic anomalies
- ▶ Packet size patterns

VOLUMETRIC ATTACK MITIGATION

- ▶ Over-provision bandwidth capacity
- ▶ Use DDoS mitigation services
- ▶ Deploy traffic scrubbing
- ▶ Implement rate limiting
- ▶ Block traffic from known bad sources
- ▶ Use anycast distribution
- ▶ Filter at ISP level

4. PROTOCOL ATTACKS

Protocol attacks exploit weaknesses in network protocols to consume server resources, connection state tables, or infrastructure capacity. These attacks target Layer 3 and Layer 4 of the OSI model.

SYN FLOOD

TCP Handshake Exploitation

The most common protocol attack, exploiting the TCP three-way handshake mechanism.

TCP Three-Way Handshake (Normal)

- ▶ Client sends SYN (synchronize)
- ▶ Server responds with SYN-ACK
- ▶ Client sends ACK (connection established)
- ▶ Server allocates resources for connection

SYN Flood Attack

- ▶ Attacker sends flood of SYN packets
- ▶ Often with spoofed source IPs
- ▶ Server sends SYN-ACK, waits for ACK
- ▶ ACK never arrives (spoofed IP)
- ▶ Half-open connections fill state table
- ▶ Legitimate connections rejected

```
hping3 -S --flood -p 80 target_ip
```

SYN Flood Impact

- ▶ Connection table exhaustion
- ▶ Memory depletion
- ▶ CPU overload processing requests
- ▶ Service becomes unavailable

ACK FLOOD

Attack Mechanism

Flood target with TCP ACK packets:

- ▶ Send massive ACK packets
- ▶ Target must process each packet
- ▶ Check against existing connections
- ▶ Generate RST for invalid ACKs
- ▶ Resource exhaustion from processing

RST FLOOD

Flood with TCP Reset packets:

- ▶ Disrupt existing connections
- ▶ Cause connection termination
- ▶ Application-level disruption
- ▶ May require valid sequence numbers

TCP STATE EXHAUSTION

Connection Table Attacks

Exhaust connection tracking resources on firewalls and servers.

Attack Methods

- Open many legitimate connections
- Keep connections alive (idle)
- Fill connection tracking tables
- New connections rejected
- Affects stateful firewalls heavily

Sockstress Attack

Specialized TCP state exhaustion:

- Complete TCP handshake normally
- Advertise zero window size
- Server waits with open connection
- Persistent resource consumption

PUSH AND ACK FLOOD

Send TCP packets with PSH and ACK flags:

- Forces immediate data processing
- Bypasses some TCP flood detection
- Strains application layer
- May trigger application responses

SESSION ATTACKS

Connection Establishment Flood

- Complete handshake, immediate disconnect
- Rapid connection cycling
- State table churn
- Resource allocation/deallocation overhead

SSL/TLS Exhaustion

Exploit resource-intensive encryption:

- TLS handshake is CPU intensive
- Asymmetric cost (client vs server)
- Server does more work
- SSL renegotiation attacks
- THC-SSL-DOS tool exploitation

PROTOCOL ATTACK MITIGATION

SYN Flood Defenses

- **SYN Cookies –** Stateless connection validation
- **SYN Proxy –** Validate before passing to server
- **Rate Limiting –** Limit SYN packets per source
- **Timeout Reduction –** Faster half-open expiration
- **Increased Backlog –** Larger connection queue

```
sysctl -w net.ipv4.tcp_syncookies=1
```

General Protocol Defenses

- Increase connection table capacity

- Deploy stateless packet filtering
- Use TCP proxy/load balancer
- Implement connection rate limiting
- Configure appropriate timeouts
- Monitor connection states

5. APPLICATION LAYER ATTACKS

Application layer attacks (Layer 7) target specific services and applications. They require fewer resources than volumetric attacks but can be equally devastating by exhausting application server resources.

APPLICATION LAYER ATTACK CHARACTERISTICS

- Target specific applications/services
- Lower bandwidth requirement
- Harder to distinguish from legitimate traffic
- Measured in requests per second (rps)
- Exploit application logic
- Can bypass network-layer defenses
- Often require valid sessions

HTTP FLOOD

Web Server Exhaustion

Overwhelm web servers with HTTP requests.

HTTP GET Flood

- Request same resource repeatedly
- Or random URLs to bypass caching
- Server must process each request
- Database queries may be triggered
- Can appear as legitimate traffic

```
GET /index.php HTTP/1.1
GET /search?q=random123 HTTP/1.1
```

HTTP POST Flood

- Submit form data repeatedly
- Often targets login, search, forms
- Server-side processing intensive
- May trigger database operations
- Harder to cache POST responses

HTTP Request Characteristics

- Valid HTTP headers
- Complete TCP handshake
- May include cookies/sessions
- Mimics real browser behavior
- User-Agent rotation

SLOWLORIS

Slow Connection Attack

Keep connections open by sending partial requests very slowly.

How Slowloris Works

- Open multiple connections to server
- Send partial HTTP request
- Never complete the request

- Send additional headers periodically
- Server waits for complete request
- Connection pool exhausted
- Legitimate users cannot connect

Slowloris Technique

```
GET / HTTP/1.1
Host: target.com
X-Header: [send more headers slowly...]
```

Never send final blank line to complete request

Slowloris Characteristics

- Low bandwidth requirement
- Single machine can be effective
- Targets connection limits
- Apache particularly vulnerable
- Difficult to detect

SLOW POST (R.U.D.Y.)

R.U.D.Y. - R-U-Dead-Yet?

Send POST body data extremely slowly:

- Start legitimate POST request
- Specify large Content-Length
- Send body data one byte at a time
- Server waits for complete body
- Connection held open indefinitely

SLOW READ ATTACK

Read server responses very slowly:

- Make legitimate request
- Advertise small receive window
- Server buffers response data
- Memory exhaustion on server
- Connection pool depletion

APPLICATION-SPECIFIC ATTACKS

WordPress XML-RPC

- Pingback amplification
- Brute force via XML-RPC
- System.multicall abuse

DNS Application Attacks

- Random subdomain queries
- NXDOMAIN flooding
- Recursive query exhaustion

Database Exhaustion

- Complex query flooding
- Search functionality abuse
- Heavy report generation

APPLICATION LAYER DEFENSE

□ Implement Web Application Firewall (WAF)

□ Use rate limiting per IP/session

□ Deploy CAPTCHA for suspicious traffic

□ Set connection and request timeouts

□ Use reverse proxy/load balancer
□ Implement request queuing
□ Monitor application performance
□ Cache static content aggressively

6. AMPLIFICATION AND REFLECTION ATTACKS

Amplification attacks exploit protocols that generate large responses to small requests, allowing attackers to multiply their traffic volume. Reflection uses third-party servers to hide attack origin.

REFLECTION ATTACK CONCEPT

How Reflection Works

Use third-party servers to deliver attack traffic:

- Attacker spoofs source IP as victim's
- Sends requests to vulnerable servers
- Servers respond to spoofed address (victim)
- Victim receives flood of responses
- Attack origin is hidden

Benefits for Attackers

- Hide true attack source
- Leverage third-party resources
- Difficult to trace back
- Combine with amplification for power

AMPLIFICATION CONCEPT

Bandwidth Amplification Factor (BAF)

Ratio of response size to request size:

- Small request generates large response
- Multiplies attacker's bandwidth
- Higher factor = more powerful attack

```
BAF = Response Size / Request Size
```

Example Amplification Factors

- **Memcached:** 51,000x amplification
- **NTP:** 556x amplification
- **DNS:** 28-54x amplification
- **SSDP:** 30x amplification
- **SNMP:** 6x amplification
- **Chargen:** 358x amplification

DNS AMPLIFICATION

Attack Mechanism

Exploit DNS servers for traffic amplification:

Attack Process

- Find open DNS resolvers
- Craft DNS query with spoofed source (victim)
- Request records with large responses (ANY, TXT)
- DNS server sends large response to victim
- Amplification factor: 28-54x typical

Query Example

```
Small query: ~60 bytes
Large response: ~3000 bytes (ANY record)
Amplification: 50x
```

NTP AMPLIFICATION

NTP Monlist Attack

Exploit NTP servers' monitoring command:

Attack Details

- Send MONLIST command to NTP server
- Server returns list of last 600 clients
- Small request, massive response
- Amplification: up to 556x
- Widely used in major attacks

```
ntpdc -c monlist ntp_server
```

Mitigation

- Disable monlist on NTP servers
- Upgrade to NTP 4.2.7+ (removed)
- Block NTP from untrusted sources

MEMCACHED AMPLIFICATION

Record-Breaking Amplification

Memcached servers abused for massive amplification:

Attack Characteristics

- UDP port 11211 exploitation
- Store large data, request retrieval
- Amplification up to 51,000x
- Used in 1.7 Tbps attack (2018)
- Highest known amplification factor

Mitigation

- Disable UDP on Memcached
- Bind to localhost only
- Firewall port 11211
- Enable SASL authentication

OTHER AMPLIFICATION VECTORS

SSDP Amplification

- Simple Service Discovery Protocol
- UPnP device discovery
- Port 1900/UDP
- 30x amplification
- IoT devices commonly vulnerable

SNMP Amplification

- GetBulk requests
- Community string: 'public'
- 6x amplification
- Network device exploitation

Chargen Amplification

- Character Generator Protocol
- Port 19/UDP

- ▶ 358x amplification
- ▶ Legacy service, should be disabled

AMPLIFICATION DEFENSE

□ Block spoofed traffic (BCP38/BCP84)
□ Disable unnecessary UDP services
□ Configure services to require authentication
□ Rate limit responses
□ Use response rate limiting (RRL)
□ Monitor for amplification indicators
□ Deploy DDoS mitigation services

7. BOTNETS AND ATTACK INFRASTRUCTURE

Botnets are networks of compromised computers controlled by attackers. They provide the distributed infrastructure needed for large-scale DDoS attacks and are a critical component of the DDoS ecosystem.

BOTNET FUNDAMENTALS

What is a Botnet?

A botnet is a collection of internet-connected devices infected with malware that allows remote control by an attacker (botmaster). Each infected device is called a 'bot' or 'zombie.'

Botnet Components

- **Bots/Zombies –** Compromised devices
- **Botmaster –** Attacker controlling the botnet
- **C2 Server –** Command and Control infrastructure
- **C2 Protocol –** Communication method

Botnet Capabilities

- DDoS attacks
- Spam distribution
- Credential theft
- Cryptocurrency mining
- Click fraud
- Ransomware distribution
- Proxy services

BOTNET ARCHITECTURES

Centralized (Client-Server)

Traditional hierarchical structure:

- Single C2 server controls all bots
- Easy to manage for attacker
- Single point of failure
- Easier to disrupt by taking down C2
- IRC, HTTP, custom protocols

Decentralized (Peer-to-Peer)

Distributed control structure:

- No central C2 server
- Bots communicate peer-to-peer
- Highly resilient to takedown
- Harder to trace command origin
- More complex to manage
- Examples: Gameover Zeus, Hajime

Hybrid Architecture

- Combination of centralized and P2P
- Fallback mechanisms
- Domain generation algorithms (DGA)
- Fast-flux DNS

IOT BOTNETS

Internet of Things Exploitation

IoT devices have become primary botnet targets due to weak security.

Why IoT Devices?

- ▶ Default/weak credentials
- ▶ Rarely updated firmware
- ▶ Always-on connectivity
- ▶ Limited security capabilities
- ▶ Massive number of devices
- ▶ Users unaware of compromise

Notable IoT Botnets

- ▶ **Mirai –** 2016, DVRs/cameras, 620 Gbps attack
- ▶ **Mozi –** 2019, routers, DVRs, P2P-based
- ▶ **Gafgyt/Bashlite –** Routers, IoT devices
- ▶ **Meris –** 2021, MikroTik routers, 22M rps
- ▶ **Mantis –** 2022, virtual machines, 26M rps

Mirai Impact

- ▶ Attacked Dyn DNS (October 2016)
- ▶ Major internet outage
- ▶ Twitter, Netflix, GitHub affected
- ▶ Source code released publicly
- ▶ Spawned many variants

DDoS-FOR-HIRE SERVICES

Booter/Stresser Services

Commercial DDoS attack services available for purchase:

Service Characteristics

- ▶ Web-based interface
- ▶ Subscription-based pricing
- ▶ Various attack methods
- ▶ Anonymous cryptocurrency payment
- ▶ Customer support offered
- ▶ Marketed as 'stress testing'

Pricing Examples

- ▶ $10-50/month for basic access
- ▶ $100-500/month for premium
- ▶ Pay-per-attack options
- ▶ Volume discounts

⚠ Using booter services is illegal regardless of marketing claims

BULLETPROOF HOSTING

Infrastructure providers that ignore abuse complaints:

- ▶ Host C2 servers and attack infrastructure
- ▶ Located in permissive jurisdictions
- ▶ Ignore takedown requests
- ▶ Anonymous registration

- ▶ Cryptocurrency payment

BOTNET DEFENSE

- ☐ Keep all devices patched and updated
- ☐ Change default credentials
- ☐ Implement network segmentation
- ☐ Monitor for C2 traffic patterns
- ☐ Deploy endpoint protection
- ☐ Participate in botnet takedown efforts

8. DoS/DDoS ATTACK TOOLS

Various tools exist for conducting and testing DoS/DDoS attacks. Understanding these tools is essential for both red team testing and defensive preparation.

> ⚠ These tools should only be used with explicit authorization on systems you own or have permission to test

NETWORK LAYER TOOLS

hping3

Versatile packet crafting and testing tool:

- ▶ TCP, UDP, ICMP, RAW-IP modes
- ▶ Custom packet construction
- ▶ Flood mode for testing
- ▶ Port scanning capabilities

```
hping3 -S --flood -p 80 target
hping3 --udp --flood -p 53 target
hping3 --icmp --flood target
```

LOIC (Low Orbit Ion Cannon)

GUI-based flood testing tool:

- ▶ TCP, UDP, HTTP flood modes
- ▶ Originated from 4chan/Anonymous
- ▶ Easy to use interface
- ▶ Hivemind mode for coordination
- ▶ Easily detected/blocked

📌 Source IP is not hidden - easily traced

HOIC (High Orbit Ion Cannon)

Enhanced version of LOIC:

- ▶ HTTP-based attacks only
- ▶ Booster scripts for customization
- ▶ Higher request rates
- ▶ Randomized headers
- ▶ Still exposes source IP

APPLICATION LAYER TOOLS

Slowloris

Slow HTTP attack tool:

- ▶ Perl/Python implementations
- ▶ Holds connections open
- ▶ Low bandwidth requirement
- ▶ Effective against Apache

```
slowloris.pl -dns target.com -port 80
```

R.U.D.Y. (R-U-Dead-Yet)

Slow POST attack tool:

- ▶ Targets web forms
- ▶ Sends body slowly
- ▶ Low resource requirement

```
python rudy.py -t target.com
```

GoldenEye

HTTP DoS test tool:

- HTTP GET/POST floods
- Keep-alive connections
- Randomized user agents
- Multiple workers

```
python goldeneye.py http://target.com
```

HULK (HTTP Unbearable Load King)

HTTP flood testing tool:

- Obfuscated traffic generation
- Unique requests per hit
- Bypasses caching
- Python-based

STRESS TESTING FRAMEWORKS

Siege

Legitimate load testing tool:

- HTTP load testing
- Multiple concurrent users
- Transaction timing
- Benchmarking mode

```
siege -c 100 -t 1M http://target.com
```

Apache JMeter

Full-featured load testing:

- GUI and CLI modes
- Multiple protocols
- Distributed testing
- Detailed reporting
- Enterprise-grade tool

Locust

Python-based load testing:

- User behavior scripting
- Web UI for monitoring
- Distributed mode
- Modern architecture

AMPLIFICATION TOOLS

NTPDDoS

NTP amplification testing

DNS amplification scripts

Custom scripts for DNS testing

Memcrashed

Memcached amplification PoC

TOOL DETECTION

Security teams should recognize attack tool signatures:

- LOIC - distinctive traffic patterns
- Slowloris - partial HTTP headers
- Known user-agent strings
- Request rate patterns
- Payload signatures

LEGITIMATE TESTING CONSIDERATIONS

□ Obtain written authorization
□ Test in isolated environment
□ Coordinate with ISP/hosting
□ Have rollback plan
□ Monitor for collateral impact
□ Document all testing

9. CASE STUDIES: MAJOR DDoS ATTACKS

Examining notable DDoS attacks provides insight into attack evolution, attacker motivations, and lessons for defenders.

DYN DNS ATTACK (2016)

Attack Overview

- **Date:** October 21, 2016
- **Target:** Dyn DNS infrastructure
- **Method:** Mirai botnet IoT DDoS
- **Peak Volume:** 1.2 Tbps
- **Duration:** Multiple waves over one day

Impact

- Major internet outage
- Twitter, Netflix, Reddit, GitHub affected
- PayPal, Spotify, PlayStation unavailable
- East Coast US primary impact
- Millions of users affected

Key Lessons

- IoT devices as attack weapons
- DNS infrastructure criticality
- Single points of failure risk
- Need for redundant DNS

GITHUB ATTACK (2018)

Attack Overview

- **Date:** February 28, 2018
- **Target:** GitHub
- **Method:** Memcached amplification
- **Peak Volume:** 1.35 Tbps
- **Duration:** ~20 minutes

Attack Details

- Largest attack at that time
- Memcached UDP exploitation
- 51,000x amplification factor
- Mitigated within 10 minutes
- Akamai Prolexic scrubbing

Key Lessons

- Memcached UDP danger
- Importance of DDoS protection
- Quick mitigation possible
- Need to secure UDP services

AWS SHIELD ATTACK (2020)

Attack Overview

- **Date:** February 2020
- **Target:** AWS customer
- **Method:** CLDAP reflection
- **Peak Volume:** 2.3 Tbps
- **Duration:** Three days

Significance

- Largest reported attack to date (at time)
- CLDAP (Connection-less LDAP) abuse
- 56-70x amplification
- Successfully mitigated by AWS Shield

GOOGLE ATTACK (2017, DISCLOSED 2020)

Attack Overview

- **Date:** September 2017
- **Target:** Google infrastructure
- **Method:** UDP amplification
- **Peak Volume:** 2.54 Tbps
- **Duration:** Six months of campaigns

Details

- State-sponsored (China attributed)
- Multiple attack campaigns
- 180,000 exposed servers used
- Disclosed three years later

CLOUDFLARE RECORD ATTACKS (2022-2023)

HTTP/2 Rapid Reset (2023)

- **Peak:** 201 million requests per second
- **Method:** HTTP/2 protocol abuse
- **CVE:** CVE-2023-44487
- New L7 attack technique
- Exploited stream multiplexing

Mirai Variant Attack (2022)

- **Peak:** 26 million requests per second
- Cloudflare customer targeted
- 5,067 devices in botnet
- ~5,200 rps per device

ATTACK EVOLUTION TRENDS

- Increasing attack volumes (Tbps normal)
- More sophisticated L7 attacks
- IoT botnet proliferation
- Protocol vulnerability exploitation
- Multi-vector attacks
- Ransom DDoS (RDoS) increase
- State-sponsored attacks

LESSONS FROM CASE STUDIES

□ DDoS protection is essential
□ Redundant infrastructure needed
□ Quick detection and response critical
□ Monitor for new attack vectors
□ Patch amplification vulnerabilities
□ Plan for multi-day attacks

10. DETECTION AND MONITORING

Early detection of DDoS attacks enables faster response and mitigation. Effective monitoring combines traffic analysis, anomaly detection, and alerting systems.

DETECTION CHALLENGES

- Distinguishing attacks from legitimate traffic
- Flash crowds vs DDoS
- Slow attacks harder to detect
- Encrypted traffic visibility
- Distributed attack sources
- Evolving attack techniques

TRAFFIC ANALYSIS

Baseline Establishment

Understanding normal traffic patterns:

- Average bandwidth utilization
- Peak traffic periods
- Geographic traffic sources
- Protocol distribution
- Request patterns and rates
- Seasonal variations

Anomaly Indicators

- Sudden bandwidth spikes
- Unusual protocol ratios
- Geographic anomalies
- Time-based anomalies
- Request rate spikes
- Error rate increases

DETECTION METHODS

Signature-Based Detection

Match known attack patterns:

- Known tool signatures
- Attack payload patterns
- Protocol violations
- Fast but limited to known attacks

Anomaly-Based Detection

Identify deviations from normal:

- Statistical analysis
- Machine learning models
- Behavioral baselines
- Can detect unknown attacks
- May generate false positives

Rate-Based Detection

Monitor traffic rates:

- Packets per second thresholds

- Connections per second limits
- Requests per second monitoring
- Per-source rate tracking

MONITORING TOOLS AND SYSTEMS

Network Monitoring

- **NetFlow/sFlow/IPFIX –** Flow-based traffic analysis
- **SNMP –** Bandwidth and interface monitoring
- **Wireshark –** Packet capture analysis
- **ntopng –** Real-time network monitoring

Infrastructure Monitoring

- **Nagios/Zabbix –** Server and service monitoring
- **Grafana –** Visualization dashboards
- **Prometheus –** Metrics collection
- **ELK Stack –** Log aggregation and analysis

DDoS-Specific Tools

- **FastNetMon –** DDoS detection toolkit
- **Kentik –** Network intelligence platform
- **Arbor Networks –** DDoS detection appliances
- Cloud provider monitoring dashboards

KEY METRICS TO MONITOR

Network Metrics

- Bandwidth utilization (inbound/outbound)
- Packets per second
- Protocol distribution
- Source IP diversity
- Geographic distribution

Server Metrics

- CPU utilization
- Memory usage
- Connection counts
- Request rates
- Response times
- Error rates

Application Metrics

- HTTP response codes
- Page load times
- Transaction completion
- User experience metrics
- API response times

ALERTING AND RESPONSE

Alert Configuration

- Threshold-based alerts
- Rate-of-change alerts
- Compound condition alerts
- Tiered severity levels
- Escalation procedures

Response Integration

- Automated mitigation triggers
- Incident response team notification

- ▶ Status page updates
- ▶ Customer communication
- ▶ Post-incident analysis

DETECTION CHECKLIST

- ☐ Establish traffic baselines
- ☐ Deploy flow-based monitoring
- ☐ Implement anomaly detection
- ☐ Configure rate-based alerts
- ☐ Monitor server resources
- ☐ Track application metrics
- ☐ Set up automated alerting
- ☐ Test detection regularly

11. MITIGATION STRATEGIES

DDoS mitigation involves multiple layers of defense working together to absorb, filter, and deflect attack traffic while maintaining service availability.

MITIGATION APPROACHES

On-Premise Mitigation

Local defense equipment and techniques:

- Hardware appliances
- Firewall rules
- Load balancer filtering
- Router ACLs
- Limited by local bandwidth
- Best for smaller attacks

Cloud-Based Mitigation

Scrubbing centers and CDN protection:

- Massive bandwidth capacity
- Global distribution
- Always-on or on-demand
- Traffic scrubbing
- Expert management
- Handles large-scale attacks

Hybrid Approach

Combination of on-premise and cloud:

- On-premise for small attacks
- Cloud for volumetric attacks
- Automatic failover
- Best of both worlds

TRAFFIC SCRUBBING

How Scrubbing Works

Divert traffic through cleaning center:

- BGP announcement redirects traffic
- All traffic routed to scrubbing center
- Malicious traffic identified and dropped
- Clean traffic forwarded to origin
- Return path may be direct or tunneled

Scrubbing Techniques

- Rate limiting
- IP reputation filtering
- Geographic filtering
- Protocol validation
- Behavioral analysis
- Challenge-response (CAPTCHA, JS)

CDN AND ANYCAST

Content Delivery Network Benefits

- Distribute content globally
- Absorb traffic at edge
- Cache static content
- Reduce origin load
- Built-in DDoS protection

Anycast Distribution

Single IP address, multiple locations:

- Traffic routed to nearest location
- Absorbs attacks across network
- No single point of failure
- Geographic distribution of load
- Used by major DNS providers

RATE LIMITING

Types of Rate Limiting

- **Per-IP Rate Limit –** Limit requests from single source
- **Per-Endpoint –** Limit requests to specific URLs
- **Connection Limit –** Maximum concurrent connections
- **Bandwidth Limit –** Cap data transfer rates

Implementation

```
nginx: limit_req zone=one burst=5 nodelay;
iptables: -m limit --limit 25/second --limit-burst 50
```

- Web server configuration
- Firewall rules
- Load balancer settings
- API gateway limits

BLACK HOLE ROUTING

Traffic Discard

Drop traffic destined for attacked IP:

- Last resort mitigation
- Announce black hole route to ISP
- All traffic to IP dropped
- Prevents collateral damage
- Target becomes unreachable
- Attacker achieves goal

Remote Triggered Black Hole (RTBH)

- Customer-triggered black hole
- BGP community signaling
- ISP drops traffic upstream
- Protects infrastructure

SINKHOLING

Redirect attack traffic to analysis:

- Route traffic to controlled server
- Analyze attack patterns

- Doesn't disrupt target
- Used for C2 disruption

ISP COORDINATION

- Upstream filtering requests
- Black hole routing
- Bandwidth provisioning
- Attack reporting
- Incident coordination

12. COUNTERMEASURES AND DEFENSE

Comprehensive DDoS defense requires preparation, multiple layers of protection, and tested response procedures. Prevention and resilience are key.

INFRASTRUCTURE HARDENING

Network Architecture

- Overprovision bandwidth capacity
- Multiple ISP connections (multihoming)
- Anycast for critical services
- Geographic distribution
- Redundant infrastructure
- Network segmentation

Server Hardening

- Tune TCP/IP stack parameters
- Increase connection limits
- Configure SYN cookies
- Reduce timeouts
- Disable unnecessary services

```
sysctl -w net.ipv4.tcp_syncookies=1
sysctl -w net.ipv4.tcp_max_syn_backlog=2048
sysctl -w net.ipv4.tcp_synack_retries=2
```

Application Hardening

- Efficient code and database queries
- Caching implementation
- Connection pooling
- Request queuing
- Graceful degradation

NETWORK SECURITY CONTROLS

Firewall Configuration

- Block invalid packets
- Rate limit connection attempts
- Geographic blocking if applicable
- Protocol filtering
- Stateful inspection limits

Access Control Lists (ACLs)

- Block known bad sources
- Allow only required protocols
- Filter RFC 1918 addresses inbound
- Block amplification ports

```
deny udp any any eq 11211 (Memcached)
deny udp any any eq 19 (Chargen)
```

Load Balancing

- Distribute traffic across servers
- Health checking
- Geographic load balancing
- Automatic failover
- Connection limiting per source

DDOS PROTECTION SERVICES

Cloud-Based Solutions

- **Cloudflare –** CDN with DDoS protection
- **AWS Shield –** AWS native protection
- **Akamai Prolexic –** Enterprise scrubbing
- **Azure DDoS Protection –** Azure native solution
- **Google Cloud Armor –** GCP protection
- **Imperva –** Application protection

Selection Criteria

- Capacity and bandwidth
- Time to mitigation
- Global coverage
- Protocol support
- Always-on vs on-demand
- Cost structure
- SLA guarantees

INCIDENT RESPONSE PLANNING

DDoS Response Plan

- Define roles and responsibilities
- Establish communication channels
- Document escalation procedures
- List mitigation contacts
- Create runbooks for attack types
- Plan for customer communication

During Attack

- Identify attack type and vector
- Implement appropriate mitigation
- Communicate with stakeholders
- Document timeline and actions
- Monitor mitigation effectiveness
- Coordinate with providers

Post-Attack

- Conduct post-mortem analysis
- Document lessons learned
- Update protection measures
- Review and improve procedures
- Train staff on findings

BUSINESS CONTINUITY

- Redundant service deployment
- Failover procedures
- Status page communication

- Alternative service delivery
- Customer notification process

DEFENSE IN DEPTH CHECKLIST

- ☐ Conduct DDoS risk assessment
- ☐ Implement network monitoring
- ☐ Deploy DDoS protection service
- ☐ Configure firewall and ACLs
- ☐ Harden servers and applications
- ☐ Implement rate limiting
- ☐ Set up CDN for public services
- ☐ Create incident response plan
- ☐ Test mitigation procedures
- ☐ Train staff on response
- ☐ Establish ISP relationships
- ☐ Maintain emergency contacts
- ☐ Document architecture thoroughly
- ☐ Review and update regularly

KEY DEFENSE PRINCIPLES

Remember core strategies:

- ▶ Absorb: Have capacity to handle attacks
- ▶ Detect: Know when attacks occur quickly
- ▶ Deflect: Redirect malicious traffic
- ▶ Filter: Remove bad traffic surgically
- ▶ Respond: Execute practiced procedures
- ▶ Recover: Return to normal operations
- ▶ Learn: Improve from each incident

https://eph4.ai/