What is a (D)DoS - technical

2026-06-20

Technical (D)DoS attack vectors: Layer 3/4 floods, BGP hijacking, Layer 7 application DDoS, and operational resilience strategies to keep services available.

The technical follow-up to the non-technical (D)DoS overview. Covers Layer 3/4 network floods (SYN, UDP, amplification), BGP hijacking, Layer 7 application DDoS, and operational resilience — all focused on availability.

Originally published at opensourcerers.org; updated with current attack vectors and technologies.

Series navigation: - Part 1 — Non-technical: Business, Social, Informational - **Part 2 — Technical: Network floods, BGP hijacking, Layer 7, Resilience (this post)** - Part 3 — Application Security

Introduction

The topic “Denial of Service” (DoS) and “Distributed Denial of Service” (DDoS) is always a hot topic because it could happen at any time for any Service at any Layers. To understand what a (D)DoS is, let us explain what a “service” is, what possible attacks are available and why such a denial of service attack could happen at any time for any service.

This is the second or technical part of the first part: What’s a (D)DoS and how to protect against such an attack – non technical.

This article focuses on attacks that target **availability** — making services unreachable or unusable. The attack vectors covered:

- Layer 3/4: Network floods (SYN Flood, UDP Flood, Amplification, IP Spoofing)
- Routing: BGP Hijacking
- Layer 7: Application-level DDoS (expensive requests, crawler traffic)
- Operational resilience as a defense strategy

The picture below shows the different layers of an application request and therefore also an attacker’s request. The TLS protocol sits as an additional layer underneath the application protocol like HTTP.

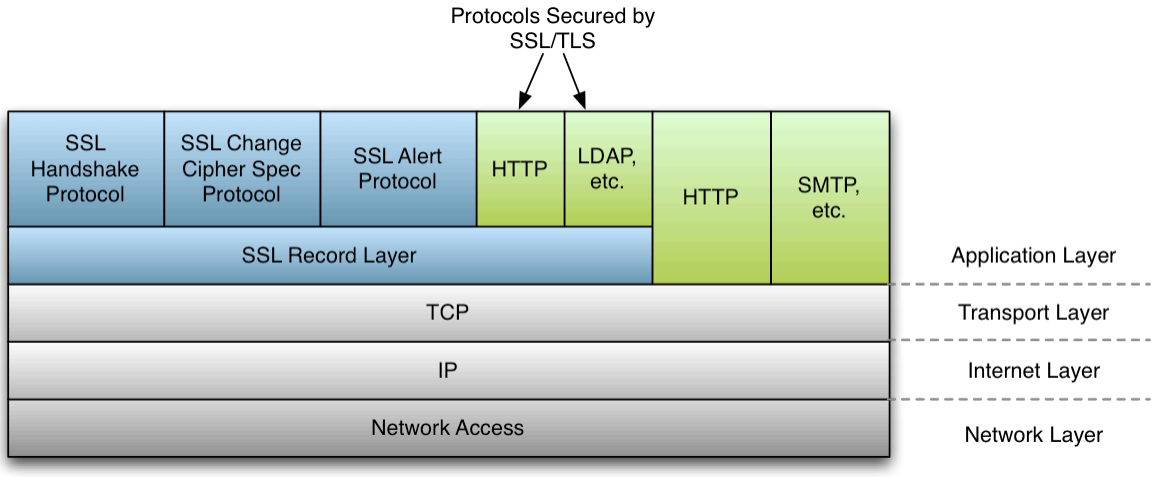

Figure 1: SSL/TLS protocol layer overview

Source: D. SSL/TLS Overview - wolfSSL Manual

TLS termination is itself a potential DDoS surface. A TLS handshake is computationally asymmetric: the client sends a ClientHello, but the server performs the expensive work — certificate validation, key exchange, session setup. An attacker can open many connections and deliberately stall the handshake, or in older TLS versions trigger renegotiation attacks — each renegotiation is cheap for the attacker and expensive for the server. TLS 1.3 eliminates renegotiation, but TLS termination at scale still requires capacity planning and connection-rate limiting at the load balancer.

Overview

The goal of a (D)DoS is to deny the delivery of the service to the end user — through volume, resource exhaustion, or routing manipulation. The full scenario including command-and-control infrastructure is covered in Part 1.

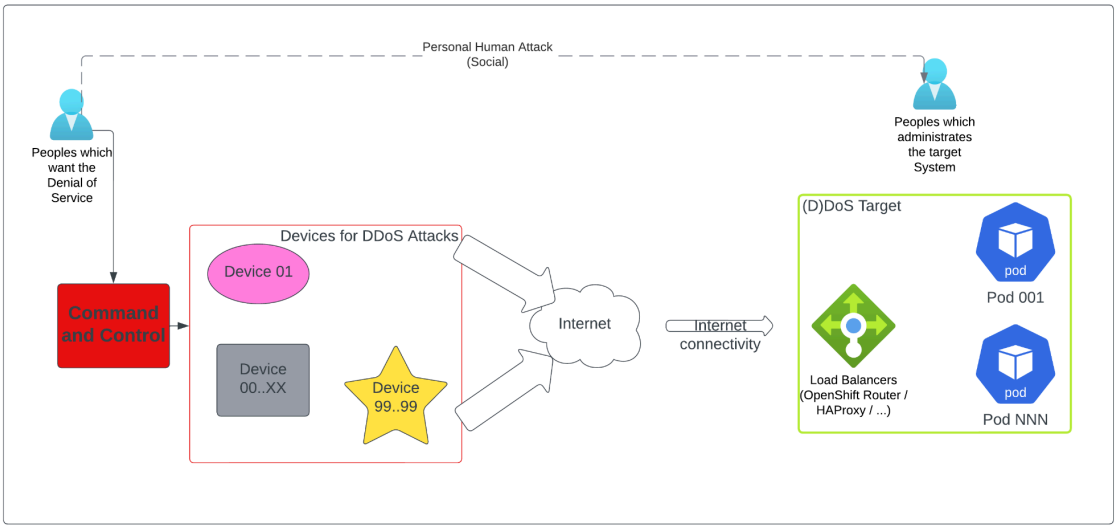

Figure 2: DDoS attack overview: target, devices, command and control structure

Layer 3/4: Network Floods

Network-layer attacks target the IP, TCP, and UDP stack directly — the goal is to exhaust bandwidth or connection-handling capacity before traffic even reaches the application.

Common attack types:

- **SYN Flood**: The attacker sends a large number of TCP SYN packets without completing the three-way handshake. The target allocates state for each half-open connection, exhausting its connection table.
- **UDP Flood**: High-volume UDP packets sent to random ports. The target must process each packet and respond with ICMP "port unreachable" messages, consuming CPU and bandwidth.
- **Amplification attacks**: The attacker sends small requests to open resolvers (DNS, NTP, memcached) with a spoofed source IP pointing to the target. The resolver sends a much larger response to the target — amplification factors of 10–100x are common with DNS and NTP.
- **IP source address spoofing**: The attacker forges the source IP, making traffic appear to come from legitimate addresses and making source-based filtering harder.

From a network point of view the attack can happen on all layers of the TCP/IP model. Nowadays most connections are based on IP and TCP or UDP, which makes these the most common targets.

Protection

- **Firewalls and network routers**: Can filter spoofed source addresses (BCP38 ingress filtering) and limit connection rates per source.
- **Software-defined networking (SDN)**: Moves attack mitigation into the network fabric, but the attack surface remains at the network layer regardless.
- **DDoS scrubbing services**: Cloud providers and specialized services absorb and filter volumetric attacks upstream before traffic reaches the origin — effective for floods that would overwhelm on-premise infrastructure.
- **Cloud provider mitigation**: Many managed infrastructure providers include volumetric DDoS protection at the network edge as part of their base offering.

Routing: BGP Hijacking

Border Gateway Protocol (BGP) Hijacking operates at the routing level. BGP is one of the most widely used routing protocols, necessary for traffic to reach the target autonomous system (AS). BGP works with prefix announcements so that network neighbors know how to reach a destination AS.

When an attacker sends a BGP announcement with a more specific network prefix, traffic gets routed to the attacker's infrastructure instead of the real target — effectively making the original service unreachable from parts of the internet.

Protection

The Resource Public Key Infrastructure (RPKI) allows route originators to cryptographically sign their prefix announcements, enabling downstream routers to reject invalid routes. Adoption is growing but not yet universal — coordination with your upstream provider and network team is essential here.

Layer 7: Application DDoS

Unlike network floods, Layer 7 attacks consume application resources rather than raw bandwidth. The attacker sends requests that are individually legitimate but collectively exhaust server capacity — database connections, CPU time, memory, or thread pools.

Common patterns:

- **Expensive HTTP requests**: Targeting endpoints that trigger heavy computation — complex search queries, report generation, deep pagination, or image resizing. Each request is syntactically valid but resource-intensive.
- **Login endpoint abuse**: High-volume credential stuffing or brute-force attempts that hammer authentication infrastructure and often trigger backend database or session store load.
- **Slow HTTP attacks (Slowloris)**: Keeping many connections open by sending data very slowly, exhausting the server's connection pool without generating high traffic volume.
- **API complexity attacks**: GraphQL queries with deeply nested resolvers or recursive fragments can trigger unbounded backend computation. LLM inference endpoints are a newer variant — a single completion request consumes orders of magnitude more compute than a static file request, making them a natural target for resource exhaustion attacks. Per-user rate limiting and request quotas at the gateway layer are essential; see LiteLLM API Gateway for a practical example.
- **AI crawler traffic**: Aggressive indexing bots from AI training pipelines generate high request volumes from distributed infrastructure. For smaller sites, this can look and feel identical to a volumetric DDoS — many IPs, sustained load, difficult to distinguish from legitimate traffic by volume alone.

The OpenShift Router (HAProxy-based) provides rate limiting via Route-specific annotations. This works when the Router is at the edge and receives the real client IP directly, or when an upstream load balancer forwards it via Proxy Protocol. The Proxy Protocol pass-through can be configured in the OpenShift Ingress Operator.

Protection

- **Rate limiting**: Per-IP or per-endpoint request rate caps at the ingress layer (HAProxy, Envoy, cloud load balancer). Essential for login endpoints and expensive search paths. For a concrete implementation see Envoy Gateway Global Rate Limiting and the HAProxy SPOE agent.
- **Connection limits**: Cap concurrent connections per source IP to defend against slow HTTP attacks.
- **Caching**: Serving static or semi-static responses from cache eliminates backend load for repeated expensive requests.
- **Web Application Firewall (WAF)**: Can identify and block known attack patterns and abusive user agents. Effective only when actively configured for the specific application — a WAF deployed with default rules provides limited protection.
- **robots.txt and crawler controls**: `robots.txt` can signal intent to polite crawlers — but `Crawl-delay` is ignored by Googlebot and unreliably followed by AI indexers. The effective lever is rate-limiting and user-agent-based blocking at the edge, which enforces behavior regardless of whether the crawler respects `robots.txt`.

Operational Resilience

No single mitigation is sufficient against a sustained, well-resourced DDoS. Operational resilience means designing the system so that partial overload doesn't cause a complete outage.

Key strategies:

- **Geographic redundancy**: Running the service in multiple regions so that a volumetric attack on one region doesn't take down the entire service. Traffic can be shifted via DNS or load balancers to healthy regions.
- **Anycast routing**: Publishing the same IP prefix from multiple network locations. Traffic is automatically routed to the nearest point of presence, distributing attack load across the infrastructure.
- **Failover and health checks**: Automated failover between regions or availability zones, triggered by health checks, reduces the window between an attack and a mitigation response.

Protection

The most effective resilience architecture combines upstream scrubbing (absorb the volume before it reaches you) with geographic distribution (no single point of failure) and automated failover (minimize human reaction time). These are infrastructure-level decisions that should be planned before an attack, not during one.

Person (OSI Layer 8)

This layer isn't really technical but I wanted to highlight that people are involved in several levels of a (D)DoS attack and also in the technical level. The section Social in the first part of this series describes the attack on this level.

Application Security: Out of Scope

Several topics that often appear alongside DDoS discussions — SQL Injection, Log4Shell (CVE-2021-44228), OWASP Top 10, Input Validation — are **application security** topics, not DDoS vectors. They primarily affect **confidentiality and integrity** (data theft, remote code execution), not availability. A successful SQL Injection attack extracts data; it does not typically make the service unavailable.

These topics are covered in a follow-up post on Application Security.

Conclusion

The attack vectors covered here — volumetric floods, BGP hijacking, Layer 7 exhaustion — all target availability. No single control addresses all of them. The effective posture combines upstream scrubbing at the network edge, rate limiting and WAF at the application layer, and geographic distribution for resilience.

The boundary between an intentional attack and accidental overload is also blurring: AI crawlers generate distributed, high-volume request patterns that are functionally indistinguishable from a Layer 7 DDoS. The same infrastructure decisions that defend against one defend against the other.