5 DNS Attack Types and How To Prevent Them

What Is a Domain Name Server (DNS) Attack?

DNS is a fundamental form of communication. It takes user-inputted domains and matches them with an IP address. DNS attacks use this mechanism in order to perform malicious activities.

For example, DNS tunneling techniques enable threat actors to compromise network connectivity and gain remote access to a targeted server. Other forms of DNS attacks can enable threat actors to take down servers, steal data, lead users to fraudulent sites, and perform Distributed Denial of Service (DDoS) attacks.

This is part of an extensive series of guides about Cybersecurity.

In this article:

- Keep DNS Resolver Private and Protected
- Configure Your DNS Against Cache Poisoning
- Securely Manage Your DNS servers
- Test Your Web Applications and APIs for DNS Vulnerabilities

What Is DNS?

Domain name system (DNS) is a protocol that translates a domain name, such as website.com, into an IP address such as 208.38.05.149.

When users type the domain name website.com into a browser, a DNS resolver (a program in the operating system) searches for the numerical IP address or website.com. Here is how it works:

- The DNS resolver looks up the IP address in its local cache.
- If the DNS resolver does not find the address in the cache, it queries a DNS server.
- The recursive nature of DNS servers enables them to query one another to find a DNS server that has the correct IP address or to find an authoritative DNS server that stores the canonical mapping of the domain name to its IP address.
- Once the resolver finds the IP address, it returns it to the requesting program and also caches the address for future use.

Why Perform an Attack on the DNS?

DNS is a fundamental service of the IP network and the internet. This means DNS is required during most exchanges. Communication generally begins with a DNS resolution. If the resolution service becomes unavailable, the majority of applications can no longer function.

Attackers often try to deny the DNS service by bypassing the protocol standard function, or using bug exploits and flaws. DNS is accepted by all security tools with limited verification on the protocol or the usage. This can open doors to tunneling, data exfiltration and other exploits employing underground communications.

What Are the 5 Major DNS Attack Types?

Here are some of the techniques used for DNS attacks.

1. DNS Tunneling

DNS tunneling involves encoding the data of other programs or protocols within DNS queries and responses. It usually features data payloads that can take over a DNS server and allow attackers to manage the remote server and applications.

DNS tunneling often relies on the external network connectivity of a compromised system, which provides a way into an internal DNS server with network access. It also requires controlling a server and a domain, which functions as an authoritative server that carries out data payload executable programs as well as server-side tunneling.

Related content: Read our guide to DNS tunneling

2. DNS Amplification

DNS amplification attacks perform Distributed Denial of Service (DDoS) on a targeted server. This involves exploiting open DNS servers that are publicly available, in order to overwhelm a target with DNS response traffic.

Typically, an attack starts with the threat actor sending a DNS lookup request to the open DNS server, spoofing the source address to become the target address. Once the DNS server returns the DNS record response, it is passed to the new target, which is controlled by the attacker.

Learn more in our detailed guide to DNS amplification attacks

3. DNS Flood Attack

DNS flood attacks involve using the DNS protocol to carry out a user datagram protocol (UDP) flood. Threat actors deploy valid (but spoofed) DNS request packets at an extremely high packet rate and then create a massive group of source IP addresses.

Since the requests look valid, the DNS servers of the target start responding to all requests. Next, the DNS server can become overwhelmed by the massive amount of requests. A DNS attack requires a great amount of network resources, which tire out the targeted DNS infrastructure until it is taken offline. As a result, the target's internet access also goes down.

4. DNS Spoofing

DNS spoofing, or DNS cache poisoning, involves using altered DNS records to redirect online traffic to a fraudulent site that impersonates the intended destination. Once users reach the fraudulent destination, they are prompted to login into their account.

Once they enter the information, they essentially give the threat actor the opportunity to steal access credentials as well as any sensitive information typed into the fraudulent login form. Additionally, these malicious websites are often used to install viruses or worms on end users' computers, providing the threat actor with long-term access to the machine and any data it stores.

Learn more in our detailed guide to DNS flood attacks

5. NXDOMAIN Attack

A DNS NXDOMAIN flood DDoS attack attempts to overwhelm the DNS server using a large volume of requests for invalid or non-existent records. These attacks are often handled by a DNS proxy server that uses up most (or all) of its resources to query the DNS authoritative server. This causes both the DNS Authoritative server and the DNS proxy server to use up all their time handling bad requests. As a result, the response time for legitimate requests slows down until it eventually stops altogether.

DNS Attack Prevention

Here are several ways that can help you protect your organization against DNS attacks:

Keep DNS Resolver Private and Protected

Restrict DNS resolver usage to only users on the network and never leave it open to external users. This can prevent its cache from being poisoned by external actors.

Configure Your DNS Against Cache Poisoning

Configure security into your DNS software in order to protect your organization against cache poisoning. You can add variability to outgoing requests in order to make it difficult for threat actors to slip in a bogus response and get it accepted. Try randomizing the query ID, for example, or use a random source port instead of UDP port 53.

Securely Manage Your DNS servers

Authoritative servers can be hosted in-house, by a service provider, or through the help of a domain registrar. If you have the required skills and expertise for in-house hosting, you can have full control. If you do not have the required skills and scale, you might benefit from outsourcing this aspect.

Test Your Web Applications and APIs for DNS Vulnerabilities

Bright automatically scans your apps and APIs for hundreds of vulnerabilities, including DNS security issues.

BOOK A DAST DEMO!

The generated reports are false-positive free, as Bright validates every finding before reporting it to you. The reports come with clear remediation guidelines for your team. Thanks to Bright's integration with ticketing tools like JIRA, it is easy to assign issues directly to your developers, for rapid remediation.

Sign up for a FREE Bright account and start automating your application and API security testing

See Our Additional Guides on Key Cybersecurity Topics

Together with our content partners, we have authored in-depth guides on several other topics that can also be useful as you explore the world of Cybersecurity

Security Misconfiguration

Learn how security misconfigurations can expose sensitive systems and data to attackers.

- Misconfiguration Attacks: 5 Real-Life Attacks and Lessons Learned
- Directory Traversal: Examples, Testing, and Prevention
- Directory Traversal Attack: Real-life Attacks and Code Examples

Command Injection

Learn about command injection attacks, in which attackers run malicious code directly within operating systems and applications.

- Command Injection: How it Works and 5 Ways to Protect Yourself
- Code Injection Example: A Guide to Discovering and Preventing attacks

Deserialization

Learn about deserialization mechanisms and how attackers can use it to compromise vulnerable systems.

- Deserialization: How it Works and Protecting Your Apps
- Deserialization in Java and How Attackers Exploit It

Penetration Testing

Learn about penetration testing, a proactive security technique that can help organizations identify security weaknesses and fix them.

- Penetration Testing Tools: 10 Tools to Supercharge Your Pentests
- Web Application Penetration Testing: A Practical Guide
- What is Penetration Testing as a Service (PTaaS)?

Publication:

May 29, 2022

Writer:

Admir Dizdar

Related Articles:

Benefits of AppSec Education and Gamification

Activities and Opportunities at RSA Conference 2023

Web Application Scanning: Why You Need it and Choosing a Tool

Shift Left Testing: Why You Need It and 4 Tips for Success

Introducing 2023 Guide to AppSec Testing Tools

Category:

AppSec Testing

Datasheets
Whitepapers

Legal

Terms of Use
Privacy Policy
Cookies Policy

Get Started

Login
Sign Up

BOOK A DEMO