What is DNS?

Introduction

The Domain Name System is a hierarchical, distributed naming system for computers, services, or any other resource connected to the internet or a private network. Whenever you need your browser to locate and connect to a computer service or device, the DNS works behind the scenes to translate an easily-memorized domain name into the numerical Internet Protocol (IP) address for that resource. You could think of the DNS as the internet's phone book: It was created to enable people to easily identify by name all devices and services connected to the internet.

The DNS terminology

Domain Names

A domain name is a user-friendly name associated with an internet source. For example. www.f5.com is a domain name, and the URL is associated with the servers owned by F5.

The subdivision of a domain is known as a subdomain. For example, support.f5.com is the subdomain for support on F5.com. A subdomain is anything to the left of the domain name, followed by a dot.

DNS Lookup

DNS lookup is a process through which a client (such as a web browser) queries a DNS server for a particular domain. The DNS server then replies back with an IP address, which then leads the client to the desired destination.

Domain Name Space

Domain Name Space defines the overall naming structure of the internet. It is a tree-like structure of domain names, with a root domain name at the top. From that root domain, major domains such as .com, .net, .org, and other domains branch out.

Zones

A name space tree is subdivided into zones. It defines the resources available under a specific domain.

Name Servers

Name servers store information about a zone. There are two types of name servers: Primary and Secondary. Every zone has its data stored on both Primary and Secondary name servers.

DNS Resolvers

A DNS resolver is the client side of the DNS. It is responsible for initiating and sequencing the queries that ultimately lead to translation of a domain name into an IP address.

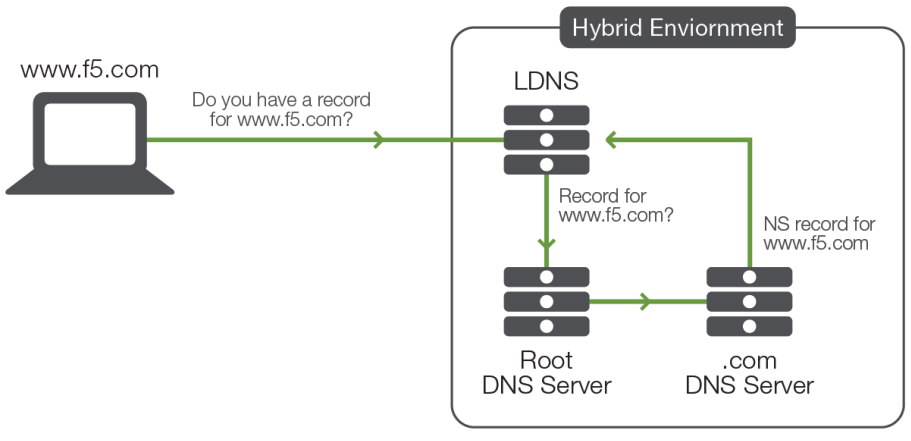
How does DNS Work?

The short version is that a domain name typed by a user in a browser (such as www.f5.com) is translated by a DNS server into an IP address (104.219.105.148). This allows the device to find the resource you are looking for on the internet—in this case, the F5 home page.

Let’s look at this process in more detail:

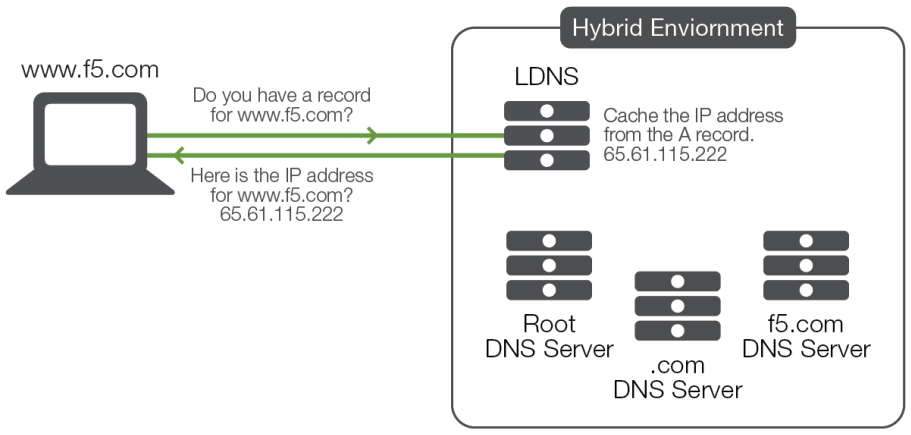
Say a user types the domain name www.f5.com in a browser. Because the browser has no clue where www.F5.com is, it sends a request to the Local DNS Server (LDNS) asking if it has the record for the website. If the LDNS has no record for that particular site, it starts a recursive search of the internet domains to find out who owns www.F5.com .

First, the LDNS goes to one of the root servers, which directs it to the .com DNS server. The .com server then determines the owner of www.F5.com and notifies the LDNS with a name server (NS) record for F5.com.



The LDNS then queries the F5.com DNS server NS record. The f5.com DNS server looks up the name www.F5.com . If it finds the name, it returns an Address (A) record to the LDNS. This A record contains the request name, the IP address assigned to that name in the LDNS, and the Time to Live (or TTL) for the name. The TTL tells the LDNS how long to maintain the A record before it asks the F5.com DNS server again.

When the LDNS receives the A record, it caches the IP address information for the time specified in the TTL; if any other client needs the same information, the LDNS will answer the query from its own cache of names before sending it out. Because it can hold on to the info locally, it doesn't need to keep asking the f5.com DNS server, making future connections to that resource faster.



The browser then uses the IP address to open a connection to www.F5.com :80 and sends a GET /.... which then leads to the web server returning the web page response.

Now, in practice, DNS is a lot more complicated than what the example above shows—but it should give you a fair idea of how it works.

DNS Records and Its Common Types

DNS records are mapping files which tell the DNS server which IP address is associated with which domain name. It also tells the DNS server how to handle those requests. There are various types of DNS records, but all DNS records for a specific domain are contained in something called a DNS Zone. Think of the DNS Zone as a container which allows the internet to look up the IP address for one, and only one, particular domain.

The common DNS record types are as follows:

A and AAAA Records

Address or A records (also known as host records) are the central records of the DNS. These records link a domain to an IP Address. AAAA record is same as A record—but instead of a 32-bit IPv4 IP address, it returns a 128-bit IPv6 address.

NS Record

Name Server (NS) records determine which servers communicate DNS information for a domain. Generally, you will have primary and secondary name server records for your domain.

MX Record

Mail Exchange records direct email messages to the servers for a particular domain. Multiple MX records can be defined for a domain, each with a different priority. The lowest number is the highest priority. If mail can't be delivered using the first priority record, the second priority record is used, and so on.

TXT Record

Text or TXT records may contain arbitrary text, but can also be used to define machine readable text.

CNAME Record

Canonical NAME or CNAME records link an alias name to another canonical domain name. For instance, alias.example.com might link to example.com.

DNS Importance and Limitations

The DNS is one of the primary technology enabling the internet. It is also a vital component in the networking infrastructure. Because having an available, intelligent, secure and scalable DNS infrastructure is critical, DNS doesn't simply deliver content and applications: it manages a distributed and redundant architecture, ensuring high availability and quality user response time. If DNS fails, most web applications will fail to function properly. This not only makes DNS critical, but also a prime target for attacks. If you don't have a proper DNS infrastructure, customers won't be able to reach your applications or content—which might lead to them turning elsewhere for their needs.

However, there are certain limitations to the standard DNS services. First, even though DNS makes your application/website/content available, DNS doesn't really care whether it's up and running, or even exists.

In addition, DNS has no real ability to distribute load. It will continue to use all the IP addresses, even if the application supported by that IP is overloaded or down.

DNS also has no concept of stateful application: it cannot guarantee that a user goes back to the same IP address. For example, if you go to a particular data center and build a shopping cart that is maintained in that data center, there is no guarantee that next time you resolve the name, you will get the same IP.

Finally, standard DNS servers can only answer a limited number of DNS queries per second, making them vulnerable to distributed denial-of-service (DDoS) attacks.

Security Issues

DNS is the backbone of the internet, but it is also one of the most vulnerable points in your network—which makes it a high-value target. DDoS attacks can flood your DNS servers to the point of hijack or failure, leading to redirecting the requests to a malicious server. To prevent this, a high performing, distributed, secure, architecture must be integrated into the network. Companies should also add more DNS servers during DNS surges and DDoS attacks.

Even though DNS servers and cloud services can handle varying amounts of requests per second, with costs increasing as the queries increase, this solution often requires manual intervention when changes are needed. And since new vulnerabilities keep coming, traditional DNS servers require frequent maintenance and patching, making it even more costly.

The Role of Application Delivery Controllers in DNS Infrastructure

Now that we've established that DNS is prone to serious attacks, let's talk about how Application Delivery Controllers (ADCs) help shield the DNS infrastructure. ADCs can balance the loads of multiple DNS servers and cache responses, providing scale and enabling DNS servers to handle large amounts of traffic and massive attacks. This functionality enables customers to deploy many DNS servers at the same time, which lets them maximize application availability, provide greater speed, and improve performance. ADCs also quickly detect DDoS attacks and route those connections away from servers – or reject them completely. ADCs support DNSSEC, and allow organizations to defend against threats such as cache poisoning and man-in-the-middle attacks. Because of all this, ADCs reduce customers' total cost of ownership by reducing the need for extra DNS servers to be provisioned as backups in case of overload, or attack.

In short, a high-performance ADC can not only protect DNS servers from various attacks but can also provide scale, improve performance, and reduce TCO, while enabling DNS servers to handle heavy traffic loads.

The Future

With the growth of mobile apps and newer technologies such as Internet of Things (IoT) devices, DNS is also growing. In addition, the number of applications is increasing rapidly, as is the volume of traffic accessing those applications. In the last 5 years, the volume of DNS queries has doubled for .com and other addresses. More than 10 million domain

more cloud, mobile and IoT implementations are deployed, the DNS is expected to grow at an even faster rate. Some recent studies done on global internet traffic show that the number of internet users will rise to 4.1 billion by the end of 2020. Because DNS servers are so critical to the internet, without a well-functioning DNS, the internet would be practically useless.

Just like IoT, the popularity of cloud services has increased immensely over the past several years. As a result, it is more important than ever to think about your DNS infrastructure, and the benefits and threats associated with it. On one hand, cloud-based authoritative DNS offers better performance, high availability, security, and scalability. On the other, it is also vulnerable to threats such as DNS infrastructure and DDoS attacks.

These threats have the potential to significantly disrupt access to websites, applications, cloud services, and other resources. Planning and managing your IT infrastructure effectively is absolutely vital to keep these attacks at bay, and continue giving your employees and customers access to the resources they need, whenever and wherever they need them.

PUBLISHED OCTOBER 05, 2017

CONNECT WITH F5

F5 LABS

The latest in application threat intelligence.

Go to F5 Labs

DEVCENTRAL

The F5 community for discussion forums and expert articles.

Go to DevCentral

F5 NEWSROOM

News, F5 blogs, and more.

Go to the newsroom

Secure and Deliver Extraordinary Digital Experiences

F5's portfolio of automation, security, performance, and insight capabilities empowers our customers to create, secure, and operate adaptive applications that reduce costs, improve operations, and better protect users. Learn more ›

WHAT WE OFFER

RESOURCES

SUPPORT

PARTNERS

COMPANY

CONNECT WITH US