Tutorial

# A DNS Deep Dive with dig, netstat, and /etc/hosts

This tutorial will guide you through the inner workings of the Domain Name System (DNS) using common Linux tools. We’ll explore how your computer translates human-readable domain names (like google.com) into machine-readable IP addresses.

## Introduction: Why DNS Matters

Imagine having to memorize the IP address of every website you visit! DNS acts like a phonebook for the internet, converting user-friendly domain names into the numerical IP addresses computers use to communicate.

## dig: The DNS Detective

The dig (Domain Information Groper) command is a powerful tool for querying DNS servers. Let’s try it out:

$ dig google.com  
  
; <<>> DiG 9.18.28-0ubuntu0.24.04.1-Ubuntu <<>> google.com  
;; global options: +cmd  
;; Got answer:  
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25954  
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1  
  
;; OPT PSEUDOSECTION:  
; EDNS: version: 0, flags:; udp: 65494  
;; QUESTION SECTION:  
;google.com. IN A  
  
;; ANSWER SECTION:  
google.com. 300 IN A 142.250.80.78  
  
;; Query time: 47 msec  
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)  
;; WHEN: Wed Oct 30 12:06:39 EDT 2024  
;; MSG SIZE rcvd: 55

This command asks a DNS server for the IP address associated with “google.com”. Observe the output:

* **HEADER**: Information about the query itself.
* **QUESTION**: What you asked (the domain name).
* **ANSWER**: The IP address(es) associated with the domain.
* **AUTHORITY**: Information about the authoritative name servers for the domain.
* **ADDITIONAL**: Extra information, like the IP addresses of those name servers.

## Exploring Record Types with dig

DNS stores different types of records. Here are a few key ones:

* **A**: Maps a domain name to an IPv4 address.
* **AAAA**: Maps a domain name to an IPv6 address.
* **CNAME**: Creates an alias for an existing domain name.
* **MX**: Specifies the mail server for a domain.
* **NS**: Identifies the authoritative name servers for a domain.

Try these commands to see different record types:

$ dig google.com AAAA # Get the IPv6 address  
$ dig www.google.com CNAME # See if it's an alias  
$ dig google.com MX # Find the mail server

## netstat: Peeking into Network Connections

The netstat command displays active network connections. Let’s see which DNS server your system is using:

netstat -rn | grep "^0.0.0.0"

This command filters the routing table to show the default gateway and associated DNS server.

Try this command to see addresses and ports your applications are using. Use the same command and run it in the root account. Notice the differences. (As a user you will not see the program names and process id’s that are running under somebody else’s account)

$ netstat -tulpn  
$ sudo netstat -tulpn

## /etc/hosts: Your Local DNS Override

The /etc/hosts file allows you to manually map domain names to IP addresses on your local machine. This can be useful for testing or blocking websites.

Open the file with a text editor:

sudo nano /etc/hosts

Add a line like this to block Example.com:

127.0.0.1 example.com

Now, your computer will resolve “example.com” to your local loopback address, effectively blocking access. Remember to remove this line when you’re done!

## Putting it All Together: Tracing a DNS Lookup

Browser Request: You type “www.example.com” in your browser. Check /etc/hosts: Your computer first checks if the domain is in /etc/hosts. Query DNS Server: If not found locally, your computer queries the DNS server specified in your network settings (which you found using netstat). Recursive Resolution: The DNS server will contact other DNS servers if necessary to find the IP address (using dig behind the scenes!). Caching: The result is cached for faster future lookups. Browser Connects: Finally, your browser uses the resolved IP address to connect to the web server.

# Key Takeaways:

DNS is a critical component of internet infrastructure.

* dig allows you to query DNS servers and explore different record types.
* netstat helps you understand your network connections and identify your DNS server.
* /etc/hosts provides a way to override DNS locally.

By understanding DNS and these essential tools, you gain valuable insights into how the internet functions and how to troubleshoot network issues.