“It was DNS”

Understanding and attacking DNS

`@__timk`

Why talk about DNS?

DNS can get fairly complicated, and people make assumptions about the way it works. It is rarely fully understood.

It is super important to grasp if you want to understand how many services work, or you are troubleshooting.

Totally nothing at all to do with this:

“<codingo> if someone does a DNS CFP that gets accepted I will buy their beers”

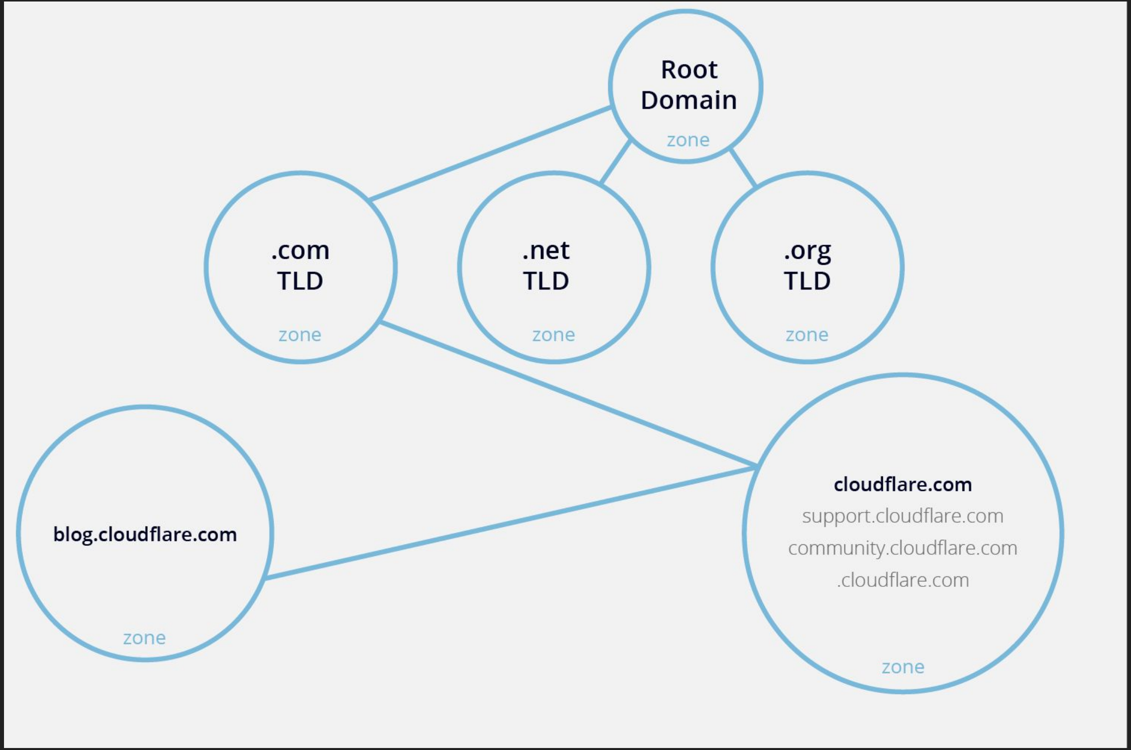
DNS Overview

DNS stands for Domain Name System.

DNS is a distributed database, with a tree structure starting from a root node.

To oversimplify it, it is a phonebook for IP addresses.

https://www.cloudflare.com/en-au/learning/dns/glossary/dns-zone/



Why does it exist?

It started with a bunch of host files being copied around ARPANET, but this did not scale.

System resolver:

- /etc/hosts
- /etc/nsswitch.conf
- /etc/resolv.conf

A few details

DNS mostly uses UDP port 53, although it can use TCP for larger queries or zone transfers.

It is unencrypted and unauthenticated.

Not to be confused with mDNS which uses UDP port 5353.

Also confusing differing standards:

- DNS over TLS uses UDP port 853
- DNS over HTTPS uses TCP port 443

Nodes are called zones. Each zone has a bunch of Resource Records within it.

Common record types:

- Start of authority (SOA)
- Name server (NS)
- Address (A)
- IPv6 address (AAAA)
- Canonical name (CNAME)
- Mail exchange (MX)
- Pointer record (PTR)

Traps I have seen first hand

Web developers delegating domain to new nameservers, breaking email and everything else (I have seen this happen way too many times).

Setting your nameserver records to hosts within the domain itself, without glue.

Incorrect records on old nameservers that think they are still authoritative.

Making changes to zone but not incrementing SOA serial, so secondary servers stay out of date.

Handing out invalid DNS resolver IP addresses via DHCP, causing frustration and rage at CrikeyCon CTF. Not mentioning names.

Security considerations

- Malware hijacking resolver settings
- On-path rewriting records or tracking users (dodgy ISP/VPN, free wifi, etc)
- DNS amplification attacks
- DHCP server performing dynamic DNS updates
- DNS cache poisoning
- DNS rebind attacks
- DNSSEC evasion
- Zone transfers (AXFR)
- Dangling records
- DNS as a transport for exfil/C2 as it usually gets a free pass

Demo time

- DNS amplification attack
- DNS cache poisoning
- DNS rebind attack

Attacker: 172.16.235.4

Victim: 172.16.235.5

Server: 172.16.235.6

Mitigations

Ensure query recursion is limited to clients you trust.

Ensure zone transfers are limited to listed servers only.

Lock down dynamic updates of records.

Cryptographically random transaction IDs have helped to some degree with cache poisoning, along with a greater use of TLS.

DNSSEC helps by providing integrity but not confidentiality, so anyone on-path can still see the records you look up.

DNS over HTTP and DNS over TLS provide confidentiality if you trust your resolvers.

Thanks!

Contact me:

Twitter: `@__timk`

SecTalks Slack: timk

https://github.com/timkent