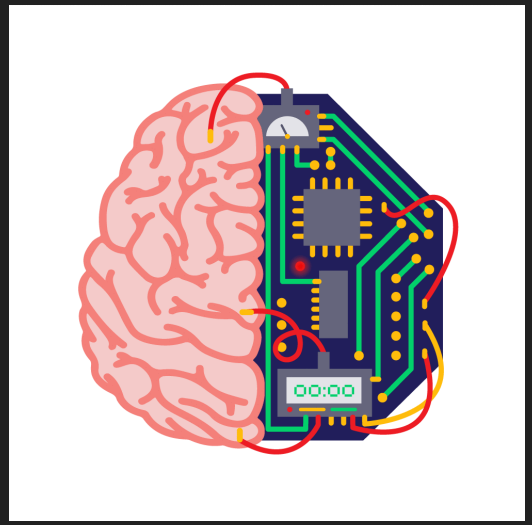
Route53 – AWS DNS

An **EXPRESS** Session

Route53 Basics….

- Translates friendly web addresses to unfriendly IP's or other DNS
- Domain Registration
- Different routing options
  - Geo-routing
  - Latency-based
  - Weighted



Activity Time!

Match the Term with the Explanation

1. Domain Name
2. Subdomain
3. Top-Level Domains
4. Hosted Zone
5. Name Servers
6. A Record
7. CNAME
8. MX Record
9. TTL

A. Used for resolving mail servers
B. What a user types into their browers e.g. *awesome-website.com*
C. Resolves a DNS name to an IP
D. Subset of a Domain Name eg. *example.awesome-website.com*
E. Servers that help look up IP's for a domain
F. Records how traffic is routed for a domain and subdomains
G. The number of seconds you want to your Hosted Zone values to be cached
H. At the top of the DNS hierarchy, can be ".com" or ".net"
I. Resolves a DNS to another DNS name (can be used with non-AWS DNS services)

Security considerations for Route53

- DNS Takeover
- Dangling DNS Records
- Misconfiguring private/public hosted zones

How can this be harmful?

How do I prevent Service Takeover?

Service takeover can happen if your domain expires and is not renewed. Someone may pick it up.

Route53 allows you to 'lock' your domain, preventing someone else from transferring the domain to another registrar without your permission.

This can be done via the console
You can also enable auto-renew

Dangling DNS - EIP
(The overly simplified example!)

You create an Elastic IP (EIP) and associate it with an AWS resource

You update your hosted zone and add a record for example.awesome-website.com to point to your EIP

You delete the EIP and it goes back into the AWS pool of IPs (and now it's up for grabs!)

Someone else  gets allocated your deleted IP and now they can take over your example.awesome-website.com

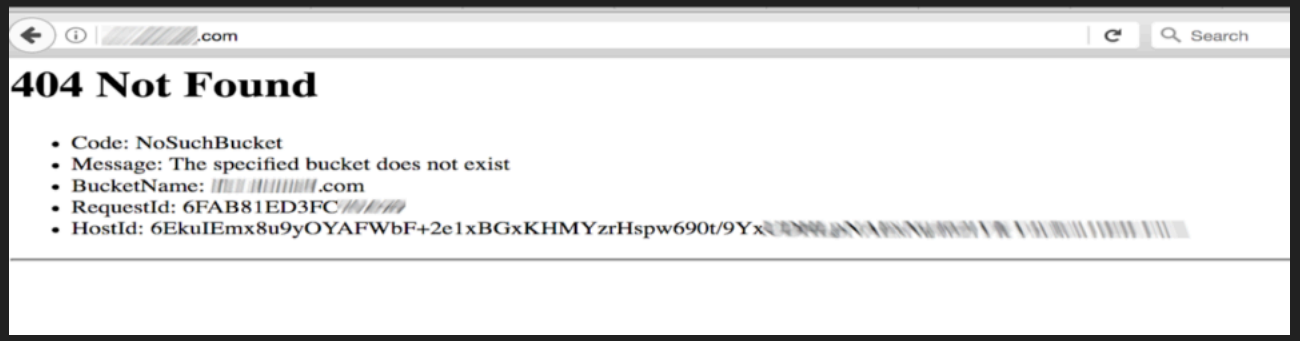
Dangling DNS – s3
(The overly simplified example!)

You create an s3 bucket for webhosting and associate it with Cloudfront

You update your hosted zone and add a record for example.awesome-website.com to point to your Cloudfront DNS

You delete the s3 bucket and

now people see this message:



Someone else 🦹 creates their own bucket with the same name and Cloudfront picks that up and now example.awesome-website.com is displaying a malicious site 😈

So, how do I prevent this?

- Be aware of what you're deleting!
- SecOps the crap out of it!

Examples!

- Bug Bounty for Shopify: https://hackerone.com/reports/365853
- Microsoft subdomains: https://www.zdnet.com/article/microsoft-has-a-subdomain-hijacking-problem/
- Roblox.com: https://hackerone.com/reports/264494