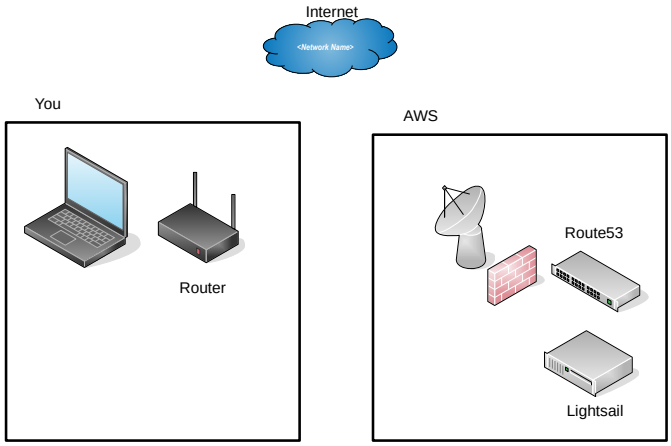
From URL to Page Load

Ever wanted to know what happens when you enter a URL into your browser? In this presentation I will show the main steps involved.

Note that this has been heavily simplified as the full details would take years and thousands of pages to present. Also I am ignoring caching, in reality there are caches at every step. If the cache can answer the request it will and the request will not proceed further.

The network



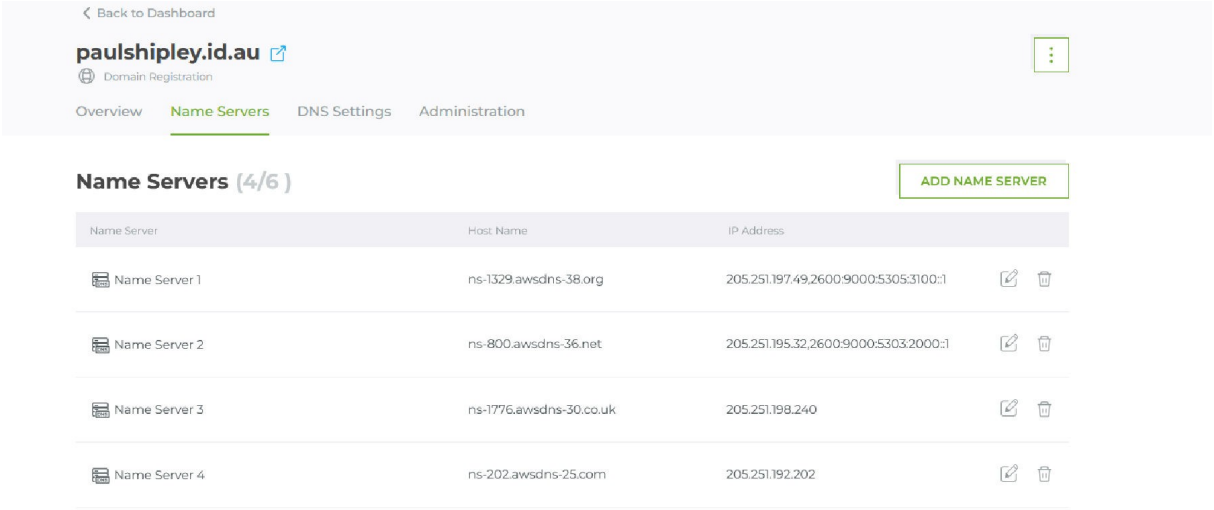
URL

Either enter a URL in the address bar of your browser, or click a link.

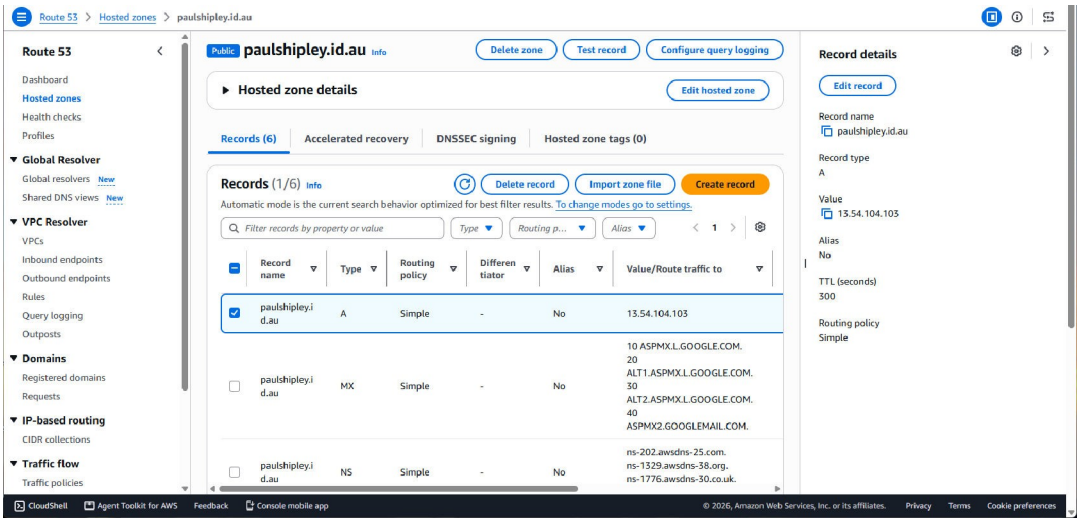


The first step is to find the website address from DNS.

As this domain is with Crazy Domains, DNS will go there, which will give it the Name Servers.



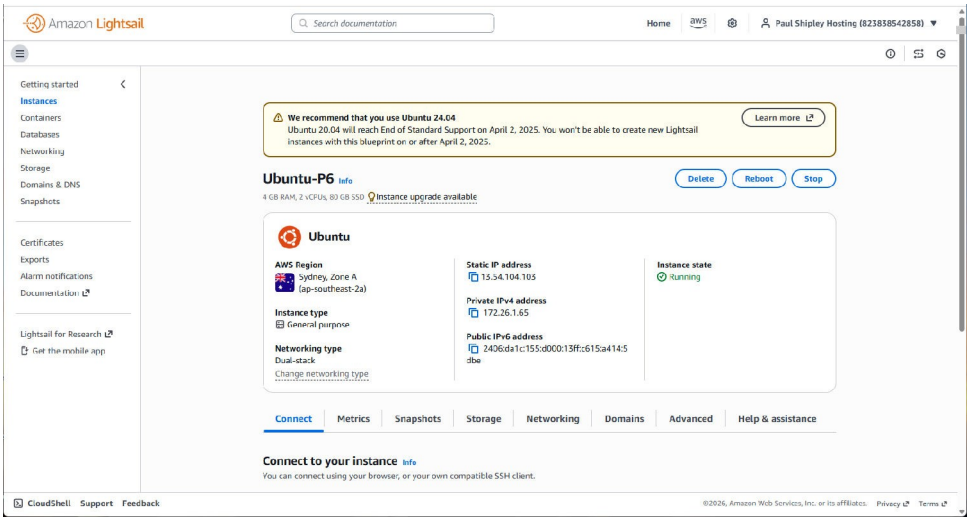
These Name Servers all point to AWS Route 53, which is the AWS DNS service.



This gives an address for the website of 13.54.104.103, which is an AWS Lightsail VM (Virtual Machine) host.

The address is returned to the browser and it starts making content requests from the host.

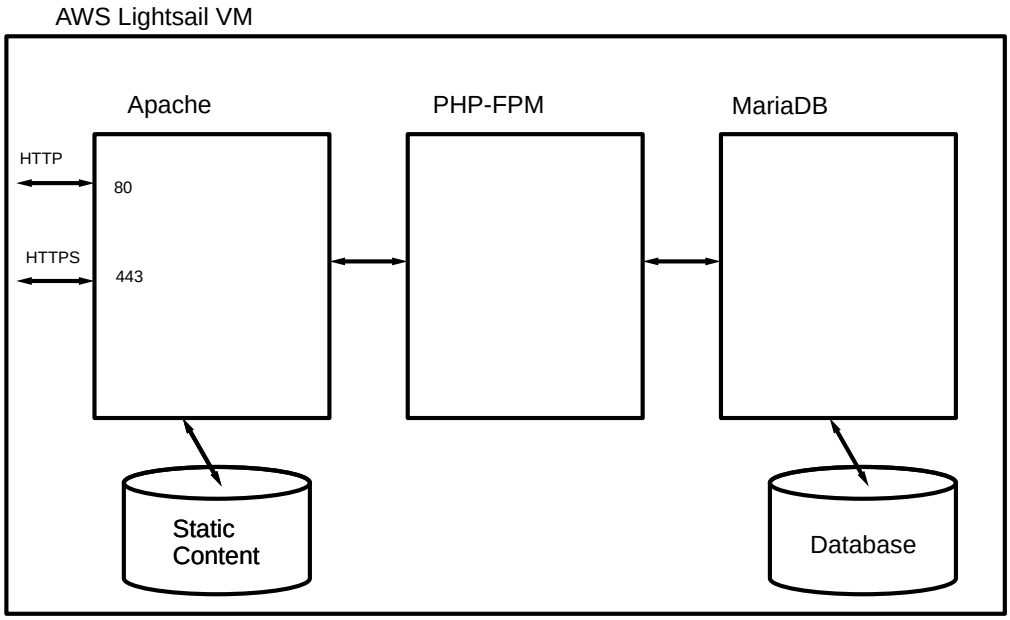
This is the AMS Lightsail configuration for this host (below).



All WordPress websites have (at least) three components:

- A web server, usually Apache or Nginx
- A PHP server, either the one built into Apache or PHP-FPM
- A database, usually MariaDB (which is a fork of MySQL)

All requests will be initially processed by the web server, with static content (eg HTML, images, CSS, Javascript, etc) being returned directly. Dynamic content (ie PHP) will be passed to the PHP server for processing. The PHP server will run the code to generate the HTML content, querying the database as required.



Each host is often serving multiple websites. These are identified is by the URL.

In the Apache configuration is a file for each of the websites, located in the /sites-available directory.

/etc/apache2/sites-available/paulshipleyidau.conf

```
<VirtualHost *:80>
 ServerName paulshipley.id.au
 ServerAlias www.paulshipley.id.au

 RewriteEngine on
 RewriteCond %{SERVER_NAME} =paulshipley.id.au
 RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI} [END,NE,R=permanent]
 RewriteCond %{HTTP_HOST} ^www\.(.*)$ [NC]
 RewriteRule ^(.*)$ https://%1/$1 [R=301,L]
</VirtualHost>

<VirtualHost *:443>
 ServerName paulshipley.id.au
 ServerAlias www.paulshipley.id.au

 RewriteEngine on
  RewriteCond %{HTTP_HOST} ^www\.(.*)$ [NC]
  RewriteRule ^(.*)$ https://%1/$1 [R=301,L]
```

```
  DocumentRoot "/var/www/paulshipleyidau"
  <Directory "/var/www/paulshipleyidau">
    Options Indexes FollowSymLinks Includes ExecCGI
    AllowOverride All
    Order allow,deny
    Allow from all
  </Directory>

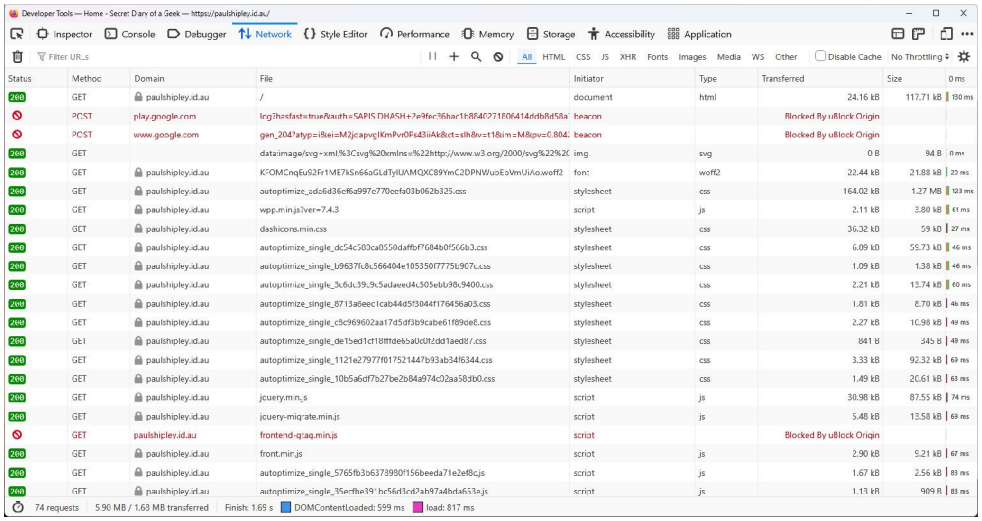
  SSLEngine on
  Include /etc/letsencrypt/options-ssl-apache.conf
  SSLCertificateFile /etc/letsencrypt/live/dance4fitness.com.au-0001/fullchain.pem
  SSLCertificateKeyFile /etc/letsencrypt/live/dance4fitness.com.au-0001/privkey.pem
</VirtualHost>
```

This defines what Apache should do when it receives a request.

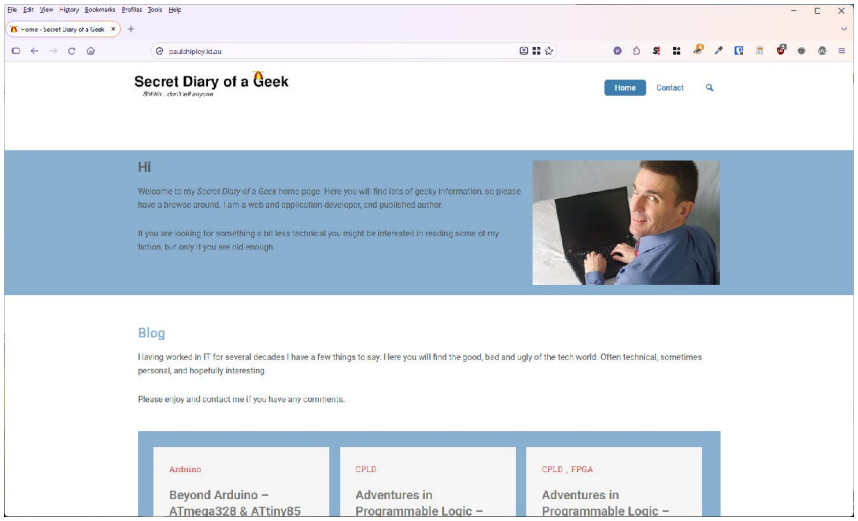
The first part handles HTTP traffic on Port 80 and redirects it to HTTPS on Port 443.

The second part tells Apache how to handle the HTTPS (443) traffic.

Apache then sends the various files to the browser.



Which the browser then displays


Secret Diary of a Geek
Home
Contact
Hi
Welcome to my Secret Diary of a Geek home page. Here you will find lots of geeky information, so please have a browse around. I am a web and application developer, and published author.
If you are looking for something a bit less technical you might be interested in reading some of my fiction, but only if you are old enough.
Blog
Having worked in IT for several decades I have a few things to say. Here you will find the good, bad and ugly of the tech world. Often technical, sometimes personal, and hopefully interesting.
Please enjoy and contact me if you have any comments.
Arduino
Beyond Arduino – ATmega328 & ATtiny85
CPLD
Adventures in Programmable Logic –
CPLD , FPGA
Adventures in Programmable Logic –