System Administration (LTAT.06.003)

Lab 4
Apache web server
Web servers

- Webpages are an integral part of the internet - while DNS acts as an address book, web servers work to provide the content

- In our course we will be using Apache as the example as it is currently the most popular web server
  - Other alternatives might be haproxy, nginx, cloudflare, node.js etc.
  - Each alternative has its use-cases, weaknesses and strengths

- The purpose of the web server is to handle HTTP requests and act as an intermediary between the browser and an application

- An integral part of web server is to also handle basic security and compatibility
HTTP requests

• All web content displaying starts with a request, a query to the server to access data
• After the request (GET, POST, UPDATE) the server replies with a return code (200 OK, 404 Not Found) and the appropriate content for the request
• Extra data can also be embedded into both the reply and query
Basic internet infrastructure

• Not all of the webpage content must be served from the same server, cross-site requests can also be used
• Most commonly ads, images and scripts are hosted on “Third-party” servers
• This practice can interfere with both the security of a webpage and the privacy of the client
  • Loading scripts from different sites might make your site vulnerable to cross-site-scripting (XSS)
  • Sharing cookies to third-parties might send private user information to other parties, hence the banners warning of “Third-party cookies”
Web server uses

In this lab we will be looking at some different applications web servers can be used for

• The most common one is proxying applications - dynamic web apps are written in python, java spring or other languages
  • The web server redirects necessary content, handles basic security, load-balancing and sessions
• Static content can also be hosted, similar to a poster on the wall
• A content management system (CMS) can also be configured, allowing a single server to host multiple websites
  • Popular examples include wordpress, drupal, joomla
This week's lab

- Setting up CNAME entries in our nameserver so we can host multiple pages on a single machine
- Setting up a static welcome-page for simple HTML web pages
- Proxying a simple application
- Installing a WordPress to act as our CMS
- Configuring extra logging and security parameters for our web server
- As always, using Ansible to do most of the work for us and keeping our progress safe