

# CIA Lab 6 Assignment: Web Servers and Directory Service

## Introduction

In this lab you will be deploying your web servers. Please do not use docker-ready images, try to install yourself.

Note: TAs might check your demo

## Web Server

In this section of the lab, you will learn how to deploy and configure the web service. You will also learn best security practices such as enabling and tuning SSL/TLS and playing with your web server security.

Choose one of the below web servers.

- **Apache**
- **Nginx**
- **lighttpd**

## Task 1 - Install & Configure Virtual Hosts

1. Download, verify, build and install the webserver daemon from the source.  
Note: Some features of your web server may be built-in or modularized. Enable at least SSL/TLS during your installation.
2. Define the root directory and then two virtual hosts (and configure DNS records or wildcard accordingly):

aaa.stX.sne22.ru  
bbb.stX.sne22.ru

3. Create a simple, unique HTML page for each virtual host to make sure that the server can correctly serve it.

### Testing:

4. Check the configuration syntax, start the daemon and enable it at boot time.
5. Use curl to display the contents of a full HTTP/1.1 session served by your server.
6. Explain the meaning of each request and reply header.

## **Task 2 - SSL/TLS**

1. Enable SSL/TLS and tune the various settings to make it as secure as possible
2. Describe how you created your own certificate(s) e.g. with Let's encrypt (certbot) or self-signed and re-validate every virtual-host . Explain your security tuning process.

## **Task 3 (bonus) - Choose one of the options from the following:**

### **1. Web Server Security**

Create a new virtual host and a HTML page with content (i.e Administrative area). Enable basic authentication in your web server for your new virtual host. Use password file creation utility and create two users. Verify your work by authenticating against your webpage.

### **2. Web Server Performance**

Investigate what configuration options there are that can potentially improve the performance of the web server. Also look at how you can check the (current) load on the web server using. Using a standard benchmarking tool (e.g. ab, siege, etc.) evaluate the performance of your server before and after optimizations. Try to maximize the number of requests per second. Explain all the changes made.

### **3. Logging**

Choose a log Analyzer, run some tests from various client locations and User-Agents, and produce statistics. Checking the logging information on your web server is important to discover problems and/or attacks on your web server. Define your own log format containing information you deem important. Use Conditional logging to add User-Agent and Referrer information to request logs that generated an error.

### **4. GeoIP**

Enable GeoIP on your chosen web-server (is only NGINX capable to do this?) and show how to take advantage of it with real examples.