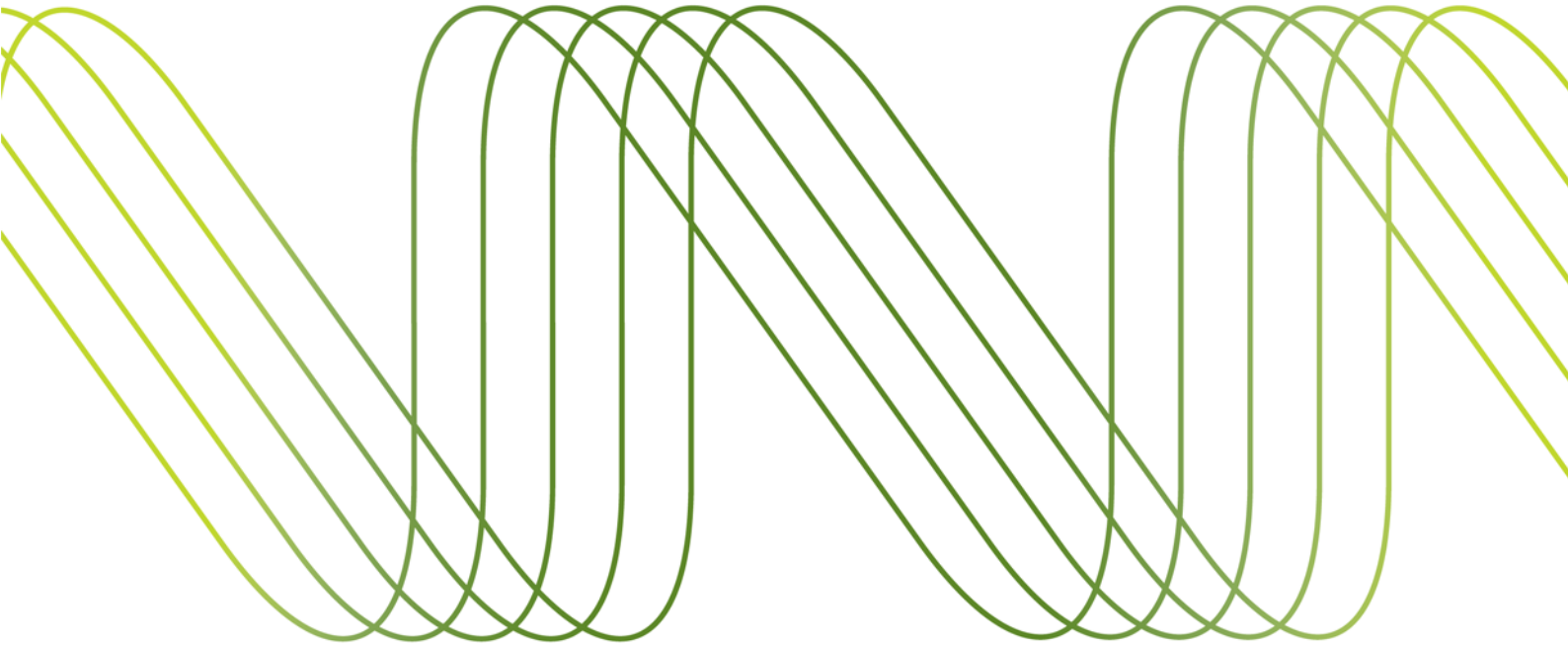
T: 0191 222 5000
E: enquiries@netskills.ac.uk
www.netskills.ac.uk

JISC netskills

Configuring Apache

CSC3504 Assignment 2



Access and Authentication

Using passwords

Create a content area for your website, to appear at:

`http://vm-eliot-NNN.ncl.ac.uk/secure-area`

Restrict access to this area to users with valid username and password details as follows:

- Give access to *yourself* using your campus username e.g. `a123456`

  but...**DO NOT USE YOUR CAMPUS PASSWORD!** ...as you will be submitting the details in a file to me.

- Also give access to the following `username:password` combination:

  `bob:umDolxdRNr4aY`

This is the username and *hashed* password – you'll need to work out how to add this to your setup, but you do not need to know the actual password that "bob" would type in ☺

Access by host/IP

Create a content area for your website, to appear at:

`http://vm-eliot-NNN.ncl.ac.uk/restricted-area`

Prevent users from specific IP ranges as follows:

- Use an `.htaccess` file to prevent access to the content delivered from this IP address:

  `128.240.148.134`

HINT You will need to test this by also (temporarily!) denying access to a machine/IP that *you* can use. Remember to remove the temporary block when you have finished.

Possible test options could be to use your VM details and use Lynx to access your site or, look up the IP of the PC you are working on and try that. You may however find it hard to test this from off campus over RAS!

Encryption & SSL

Creating keys and certificates

Use OpenSSL on your VM to create:

- A Private Key
- A CSR (Certificate signing request)
- A self-signed x509 certificate

Your certificate should be issued to the following subject (with the appropriate details completed):

```
vm-eliot-NNN.ncl.ac.uk
your.email@ncl.ac.uk
CSC3504
Newcastle University
Newcastle
Tyne & Wear
GB
```

Configure Apache for SSL

Enable SSL on your Apache server – remember this will involve:

- Compiling in/checking if you have already compiled in the correct module first
- Moving your key and certificate to your `conf` directory
- Configuring a virtual host to handle the SSL requests
- Restarting your server

You should then be able to visit:

```
https://vm-eliot-NNN.ncl.ac.uk
```

Check all your other URLs also work under SSL.

REMEMBER You will need to let your browser accept the self-signed certificate (you might want to permanently add the exception).

Log Files & Analytics

Check the log format

Check your server configuration and make sure the log format is set to Combined Log Format (you should have done this in the first assignment).

Prepare the log file destination

Create a new directory in your web root, to deliver content at:

```
http://vm-eliot-NNN.ncl.ac.uk/my-server-log/
```

Set up Analog

Make a *copy* of the default Analog configuration file (`/etc/analog.cfg`) and put it in your home directory.

Edit your copy of the configuration file to allow Analog run with the following criteria:

- Read data from *your* access log
- Direct the output to the location you created above
- Resolve hostnames by performing a DNS look-up
- Include the **Host** report
- Correctly return the **Web Server Statistics for...** line
- *Exclude* the hits from the marking server `128.240.148.134` (`vs-web-cs3.ncl.ac.uk`)

You should find that Analog will create your report HTML file and a handful of PNG files, containing the graphs it refers to. The rest of the images used in the report are generic and need to be copied to a location that the finished report can read them from.

You should be able to work out how to achieve this, through a combination of:

- Putting the files in a suitable location in your site
- Configuring Analog to write the correct links into the report

The original files (for you to copy into your web site) are in:

```
/var/www/html/images/
```

If it all works

You should be able to visit:

```
http://v-eliot-NN.ncl.ac.uk/my-server-log/
```

And see your analysed access log.

Error pages and favicons

Creating custom error pages

Create *different* custom error pages for the following HTTP codes:

- `401`
- `403`
- `404`

Ensure that the final content of each error page is > `512` bytes (shouldn't be hard!). You can be as creative as you like but make sure you:

- Make it clear which error the page is reporting
- Include a link to your server home page
- Keep it decent please, and do not use copyrighted images ☺

Save your files as `401.html` etc. and put them in the root of your web site i.e. so you *could* access them directly at:

`http://vm-eliot-NNN.ncl.ac.uk/401.html` etc.

Configure Apache to deliver the correct custom error page for each of the three circumstances above (i.e. 401, 403 and 404 errors). You should be able to figure out how to test them for yourselves...

Making a favicon

Find a suitable square image (again keep it decent and copyright free!) and use IcoFX portable to create a 16x16 icon file from your image, called `favicon.ico`

You can download **IcoFX portable** from the **Tools** section for this module in Blackboard:

Just unzip the contents and run the executable. Delete the unzipped files to remove the app.

To create a .ico from your image:

- Select `File > Import Image` and find your original
- In the next dialog pick **Colors:** `True Color (24 bits)` and **Size:** `16 x 16` and click **OK**
- Click **OK** in the `Advanced Import` dialog
- Then `File > Save As…. favicon.ico`

Using the favicon

To use your favicon, copy it to your site root. You should see it appear in the URL bar and/or bookmarks for your web site.

Figure 1. Favicon (Chrome)

Submitting Your Work

Although you may have been submitting files as you go along, the *only* URLs and files that will be marked will be those included in the *last submission you make before the deadline*.

This means that you will need to submit the unique files and URLs for all the components together.

Below you will find a checklist for the whole assessment and the filenames and URLs you will need to include.

Please pay close attention as *some of your files may need to be renamed in order to submit successfully*.

Also not you only need to submit one copy of each file for the final submission. For example the final version of your `httpd.conf` should contain all the changes required for all of the sub-components.

URLs to be auto-marked

These URLs should be submitted to the **URL** submission area for this assignment:

```
http://vm-eliot-NNN.ncl.ac.uk

http://vm-eliot-NNN.ncl.ac.uk/secure-area/

http://vm-eliot-NNN.ncl.ac.uk/restricted-area/

https://vm-eliot-NNN.ncl.ac.uk

http://vm-eliot-NNN.ncl.ac.uk/my-server-log/
```

A script will access these URLs, check they are delivering the required content, functionality or error message as per the assignment details and store a copy of the output.

You should ensure that they all deliver the correct content when accessed... and that your server is on and running!

Files to submit

Submit the files required for marking in *a single zip archive* to the **FILES** submission area for this assignment.

Name your zip file:

```
a123456789-vm-eliot-NNN.zip
```

Where `a123456789` is your student number and **NNN** is your VM number

A checklist of the files and how they should be renamed in the zip is on the next page.

File Checklist

Your file...	Renamed as...	✗/✓
Your password file for `/secure-area`	`vm-eliot-NNN-passwords.txt`	
Your `.htaccess` file for `/restricted-area`	`vm-eliot-NNN-access.htaccess.txt`	
`vm-eliot-NNN.csr`	`vm-eliot-NNN.csr.txt`	
`vm-eliot-NNN.crt`	`vm-eliot-NNN.crt.txt`	
`httpd-ssl.conf`	`vm-eliot-NNN-httpd-ssl.conf.txt`	
`vm-eliot-NNN-analog.cfg`	`vm-eliot-NNN-analog.cfg.txt`	
Zipped up contents of your `/my-server-log` directory *	`vm-eliot-NNN-log-report.zip`	
All your error pages (and any images they contain) in a single zip file **	`vm-eliot-NNN-error-pages.zip`	
`favicon.ico`	`vm-eliot-NNN-favicon.ico`	
`httpd.conf`	`vm-eliot-NNN-httpd.conf.txt`	

*From your home directory (`cd ~` to get there) use the Linux `zip` command e.g.

```
zip -r wherever.zip /usr/local/www/whatever
```

This will recursively zip up the contents of ***`whatever`*** and write them to a file called ***`wherever.zip`*** in your home directory – you should be able to work it out from there... ☺