Qassim University جامعة القصيم

College of Computer | كليـــة الحـــاسب

System Administration and Maintenance

Project Part (2)

IT 361

Table of Contents

Introduction

The Linux web server is an important resource in the structure of the World Wide Web. Built on the dependable and open-source Linux operating system, a Linux web server provides the basis for hosting and distributing web content, applications, and services. Their stability, security, and scalability distinguish Linux web servers from traditional servers, making them a popular choice for various organizations and individuals.

A web server is software that receives and responds to requests from web browsers. Hosting web servers in the Linux operating system provides an optimal environment, because of its flexibility and efficiency. By using web server software in conjunction with Linux operating system, people and businesses can create websites, run web applications, and oversee many online services.

Also, Linux web server is designed for storing, processing, and transmitting web data from the internet to users. It is important to note that this content includes static web pages, dynamic websites, multimedia files, and web applications. The Server retrieves the requested files and sends them across the network when clients such as web browsers communicate with it by sending requests. Apart from serving static content, Linux web servers are also configured to run dynamic scripts thereby making creation of interactive and data driven web applications possible.

Web servers have several important uses in the context of hosting websites and serving web content. Here are some common uses of web servers:

- **Hosting Websites:** A web server, on its part, serves as a platform for hosting websites for use by internet users. This is where the site's information, like HTML documents, image, CSS style sheets and so on are kept, and delivered to the clients when they make a demand for them.

- **Serving Web Pages:** The web server, in turn, fetches back the associated HTML file and forwarded to a user's Web browser when a user requests a certain webpage by typing in a URL, or perhaps, clicks a link. The browser reads the HTML and presents the webpage to the user.

- **Handling HTTP Requests:** Web servers handle HTTP requests from clients that could be either GET, POST, PUT or DELETE requests among others. These requests are processed by the web server, which provides an apt response in compliance with the client's demands.

- **Content Delivery:** Web servers ensure that web content is delivered in an efficient manner to the users. These are inclusive of fixed files consisting of image, movie documentation, as well as downloadable files. These web servers are able to manage many simultaneous connections as well as move huge files quickly through the network.

- **Application Hosting:** They can host web application developed upon server-side scripting language such as PHP, Python, Ruby and Node.js. They enable a server to generate dynamic contents, access a database, authenticate users as well as perform any server-sided functions.

Famous Linux Web Servers

1. Apache HTTP Server
one of the most widely recognized and enduring web servers, first introduced in 1995. The Apache Software Foundation develops and maintains it as free and open source software.

2. Nginx Web Server
Igor Sysoev developed Nginx in 2002. The program can also be used as an HTTP cache, load balancer, API gateway, reverse proxy, and IMAP/POP3 proxy server.

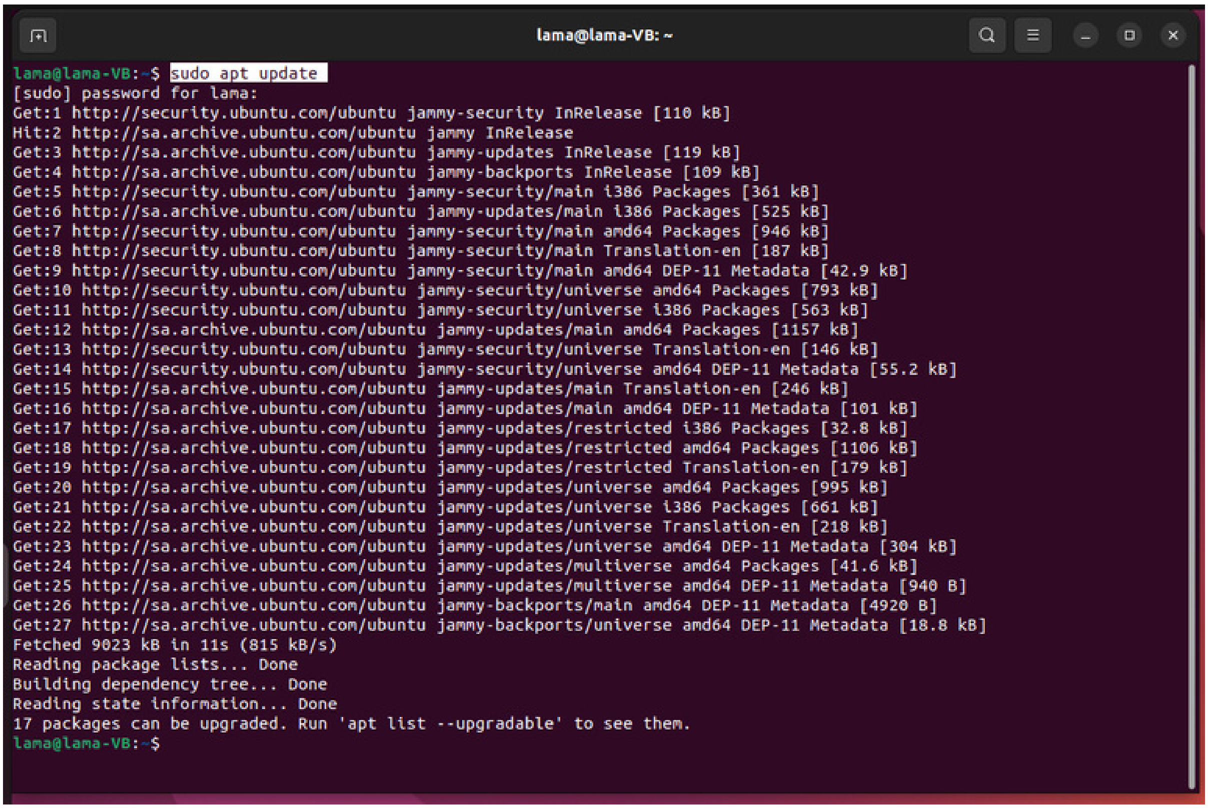
3. Caddy Web Server
one of the efficient cross-platform substitutes for Apache Web Server. It is a quick, dependency-free open-source framework created by Mathew Holt.

4. Lighttpd Web Server
An open-source web server that is quick, safe, and free that has a smaller file size than one megabyte. Speed-critical applications are best suited for the simple-to-install Lighttpd webserver, which was developed by Jan Kneschke.

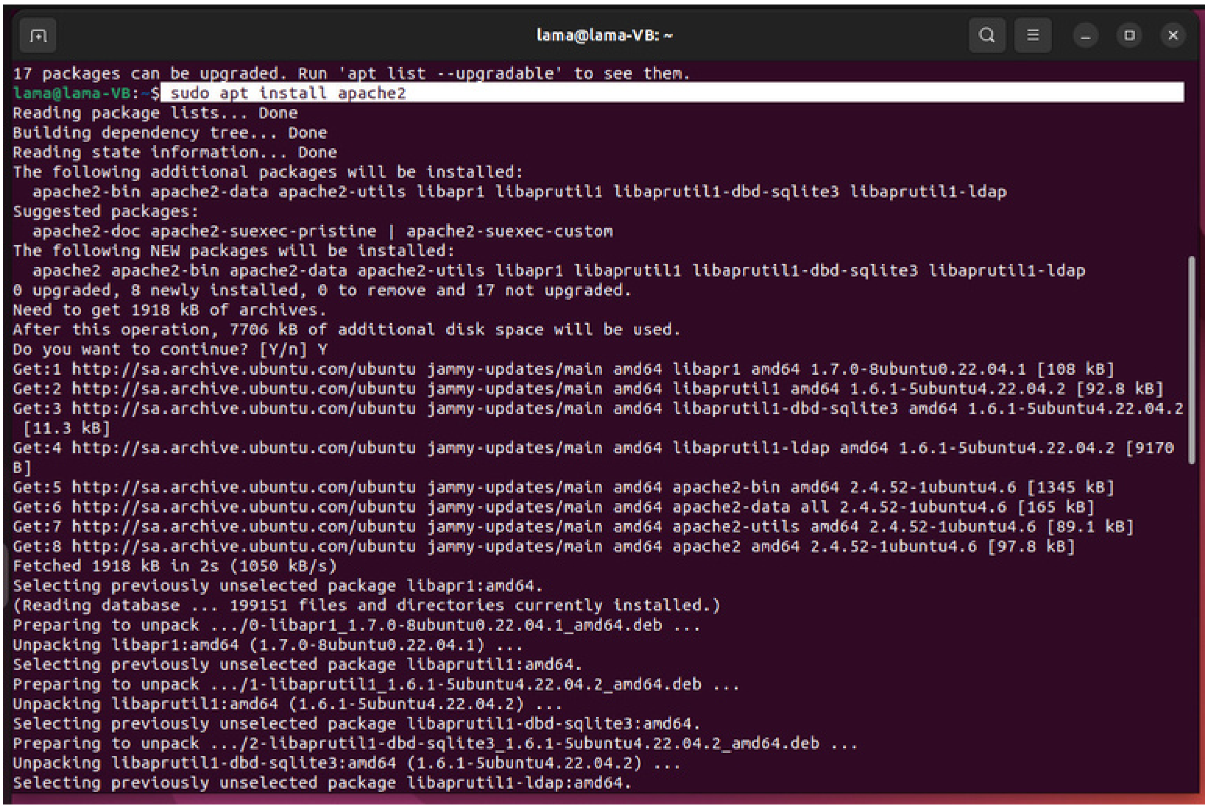
5. Apache Tomcat Web Server
published twenty years later under the Apache License version. Given its excellent performance and scalability, this web server is frequently required by large enterprises. Unlike other web servers like Nginx or Apache, Tomcat is not comparable. It's a Java servlet with plenty of additional features that let you interact with other Java servlets.

Install and Configure The Web Server

- The first thing to do is **updating** to ensure that our package lists are updated then **installing**

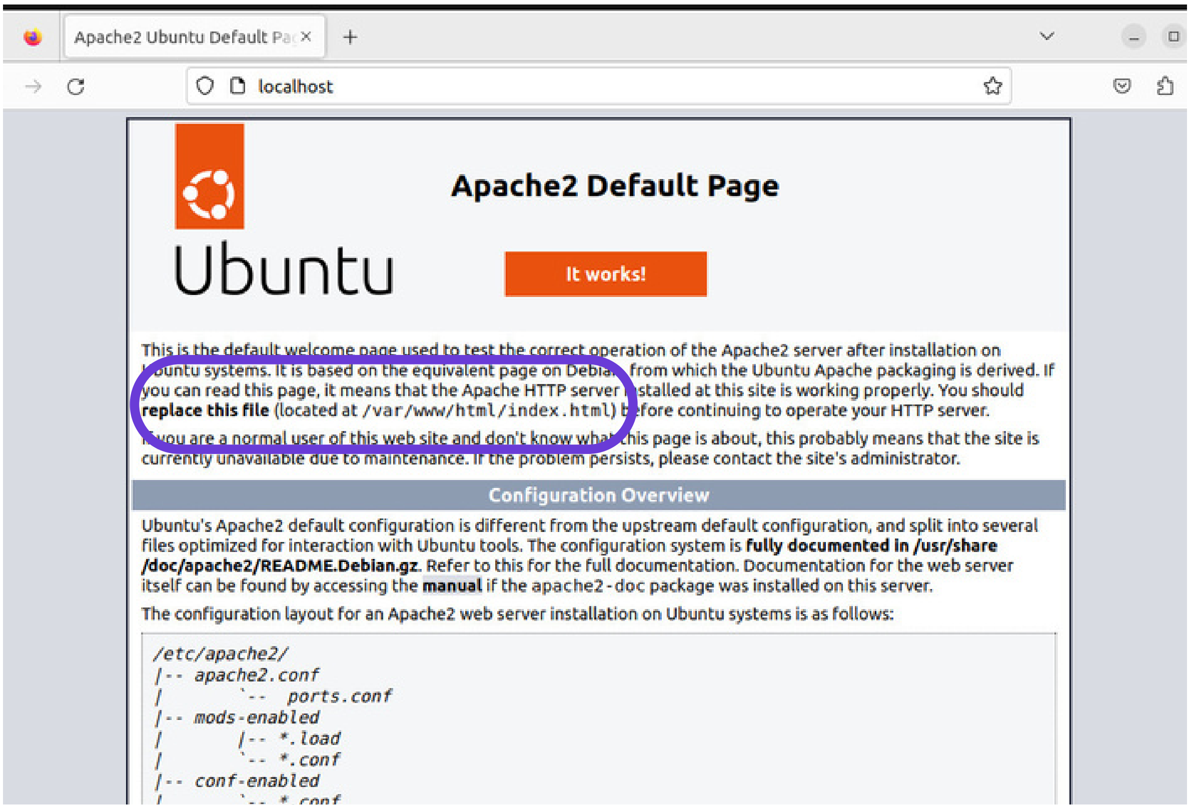


```
lama@lama-VB:~$ sudo apt update
[sudo] password for lama:
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:2 http://sa.archive.ubuntu.com/ubuntu jammy InRelease
Get:3 http://sa.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:4 http://sa.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/main i386 Packages [361 kB]
Get:6 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main i386 Packages [525 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [946 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [187 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/main amd64 DEP-11 Metadata [42.9 kB]
Get:10 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [793 kB]
Get:11 http://security.ubuntu.com/ubuntu jammy-security/universe i386 Packages [563 kB]
Get:12 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1157 kB]
Get:13 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [146 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 DEP-11 Metadata [55.2 kB]
Get:15 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [246 kB]
Get:16 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 DEP-11 Metadata [101 kB]
Get:17 http://sa.archive.ubuntu.com/ubuntu jammy-updates/restricted i386 Packages [32.8 kB]
Get:18 http://sa.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1106 kB]
Get:19 http://sa.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [179 kB]
Get:20 http://sa.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [995 kB]
Get:21 http://sa.archive.ubuntu.com/ubuntu jammy-updates/universe i386 Packages [661 kB]
Get:22 http://sa.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [218 kB]
Get:23 http://sa.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 DEP-11 Metadata [304 kB]
Get:24 http://sa.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [41.6 kB]
Get:25 http://sa.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 DEP-11 Metadata [940 B]
Get:26 http://sa.archive.ubuntu.com/ubuntu jammy-backports/main amd64 DEP-11 Metadata [4920 B]
Get:27 http://sa.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 DEP-11 Metadata [18.8 kB]
Fetched 9023 kB in 11s (815 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
17 packages can be upgraded. Run 'apt list --upgradable' to see them.
lama@lama-VB:~$
```



```
17 packages can be upgraded. Run 'apt list --upgradable' to see them.
lama@lama-VB:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap
0 upgraded, 8 newly installed, 0 to remove and 17 not upgraded.
Need to get 1918 kB of archives.
After this operation, 7706 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libapr1 amd64 1.7.0-8ubuntu0.22.04.1 [108 kB]
Get:2 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprutil1 amd64 1.6.1-5ubuntu4.22.04.2 [92.8 kB]
Get:3 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.1-5ubuntu4.22.04.2
 [11.3 kB]
Get:4 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprutil1-ldap amd64 1.6.1-5ubuntu4.22.04.2 [9170
B]
Get:5 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apache2-bin amd64 2.4.52-1ubuntu4.6 [1345 kB]
Get:6 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apache2-data all 2.4.52-1ubuntu4.6 [165 kB]
Get:7 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apache2-utils amd64 2.4.52-1ubuntu4.6 [89.1 kB]
Get:8 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apache2 amd64 2.4.52-1ubuntu4.6 [97.8 kB]
Fetched 1918 kB in 2s (1050 kB/s)
Selecting previously unselected package libapr1:amd64.
(Reading database ... 199151 files and directories currently installed.)
Preparing to unpack .../0-libapr1_1.7.0-8ubuntu0.22.04.1_amd64.deb ...
Unpacking libapr1:amd64 (1.7.0-8ubuntu0.22.04.1) ...
Selecting previously unselected package libaprutil1:amd64.
Preparing to unpack .../1-libaprutil1_1.6.1-5ubuntu4.22.04.2_amd64.deb ...
Unpacking libaprutil1:amd64 (1.6.1-5ubuntu4.22.04.2) ...
Selecting previously unselected package libaprutil1-dbd-sqlite3:amd64.
Preparing to unpack .../2-libaprutil1-dbd-sqlite3_1.6.1-5ubuntu4.22.04.2_amd64.deb ...
Unpacking libaprutil1-dbd-sqlite3:amd64 (1.6.1-5ubuntu4.22.04.2) ...
Selecting previously unselected package libaprutil1-ldap:amd64.
```

- search in the browser for "**localhost**". this page indicates a successful installation.
- then we need to accesses **index.html file** as shown here the file in path **(/var/www/html/)**



Apache2 Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

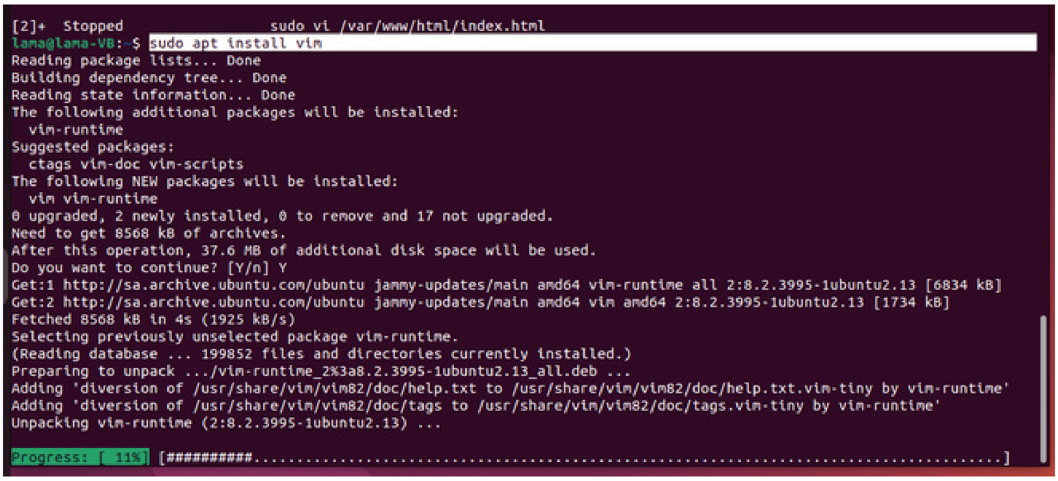
Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share /doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

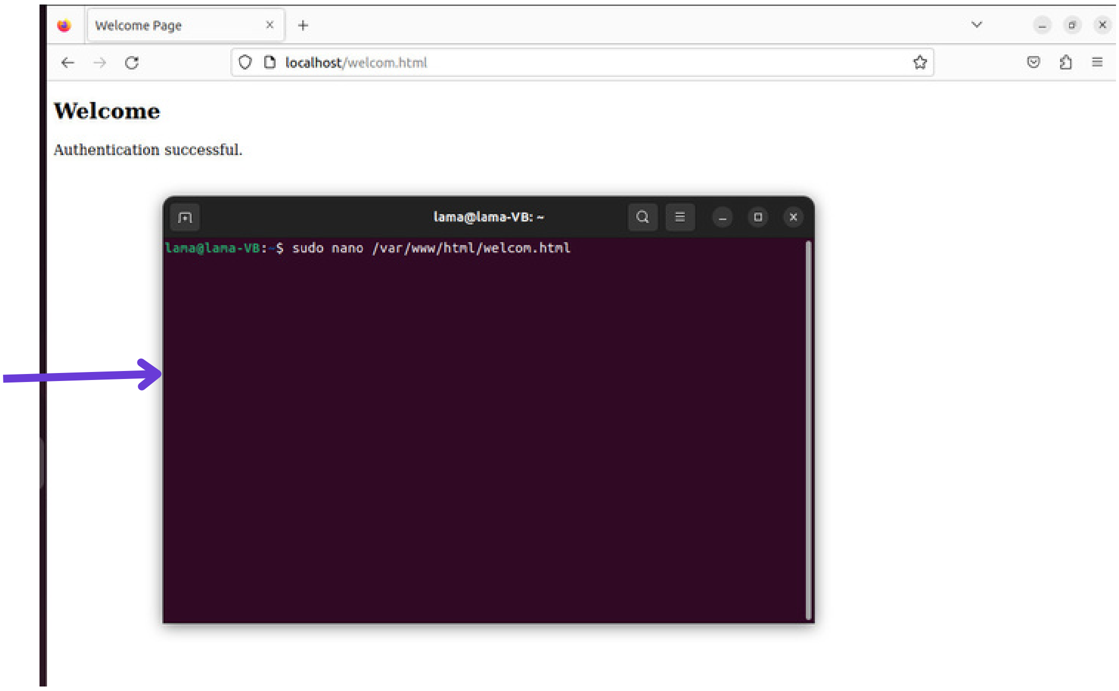
```
/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
```

- Since Vim is a highly configurable and powerful text editor we use it here and commonly used for editing configuration files



```
[2]+  Stopped                 sudo vi /var/www/html/index.html
lama@lama-VB:~$ sudo apt install vim
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  vim-runtime
Suggested packages:
  ctags vim-doc vim-scripts
The following NEW packages will be installed:
  vim vim-runtime
0 upgraded, 2 newly installed, 0 to remove and 17 not upgraded.
Need to get 8568 kB of archives.
After this operation, 37.6 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 vim-runtime all 2:8.2.3995-1ubuntu2.13 [6834 kB]
Get:2 http://sa.archive.ubuntu.com/ubuntu jammy-updates/main amd64 vim amd64 2:8.2.3995-1ubuntu2.13 [1734 kB]
Fetched 8568 kB in 4s (1925 kB/s)
Selecting previously unselected package vim-runtime.
(Reading database ... 199852 files and directories currently installed.)
Preparing to unpack .../vim-runtime_2%3a8.2.3995-1ubuntu2.13_all.deb ...
Adding 'diversion of /usr/share/vim/vim82/doc/help.txt to /usr/share/vim/vim82/doc/help.txt.vim-tiny by vim-runtime'
Adding 'diversion of /usr/share/vim/vim82/doc/tags to /usr/share/vim/vim82/doc/tags.vim-tiny by vim-runtime'
Unpacking vim-runtime (2:8.2.3995-1ubuntu2.13) ...

Progress: [ 11%] [##########.............................................................................]
```

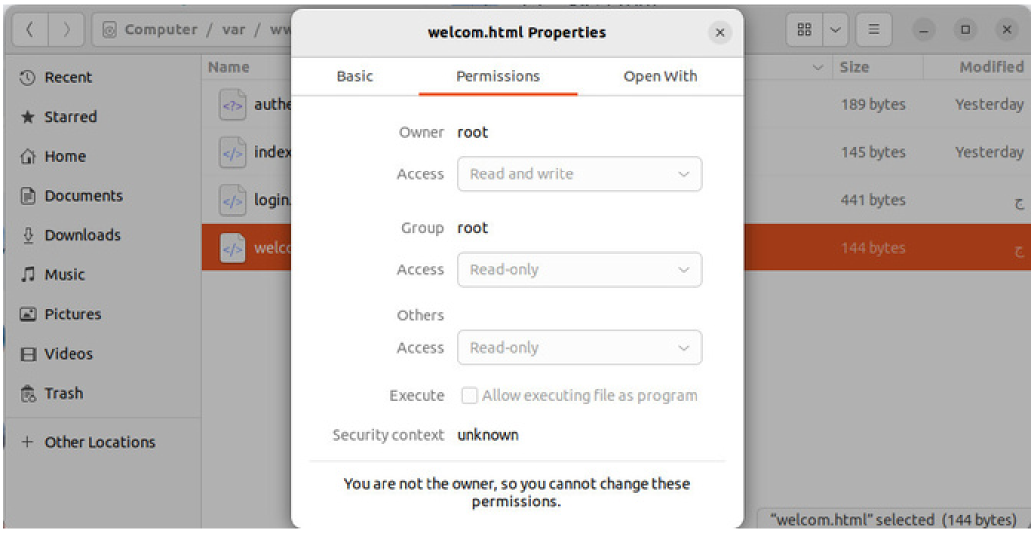
- we create a 'welcome' page and access it with sudo because it is read only



Welcome

Authentication successful.

```
lama@lama-VB:~$ sudo nano /var/www/html/welcom.html
```



Name	Size	Modified
authe	189 bytes	Yesterday
index	145 bytes	Yesterday
login	441 bytes	ح
welc	144 bytes	ح

welcom.html Properties

Owner root

Access Read and write

Group root

Access Read-only

Others

Access Read-only

Execute ☐ Allow executing file as program

Security context unknown

You are not the owner, so you cannot change these permissions.

"welcom.html" selected (144 bytes)

now to ensure that no one can access our welcome page except a **authenticated user** we do the following steps

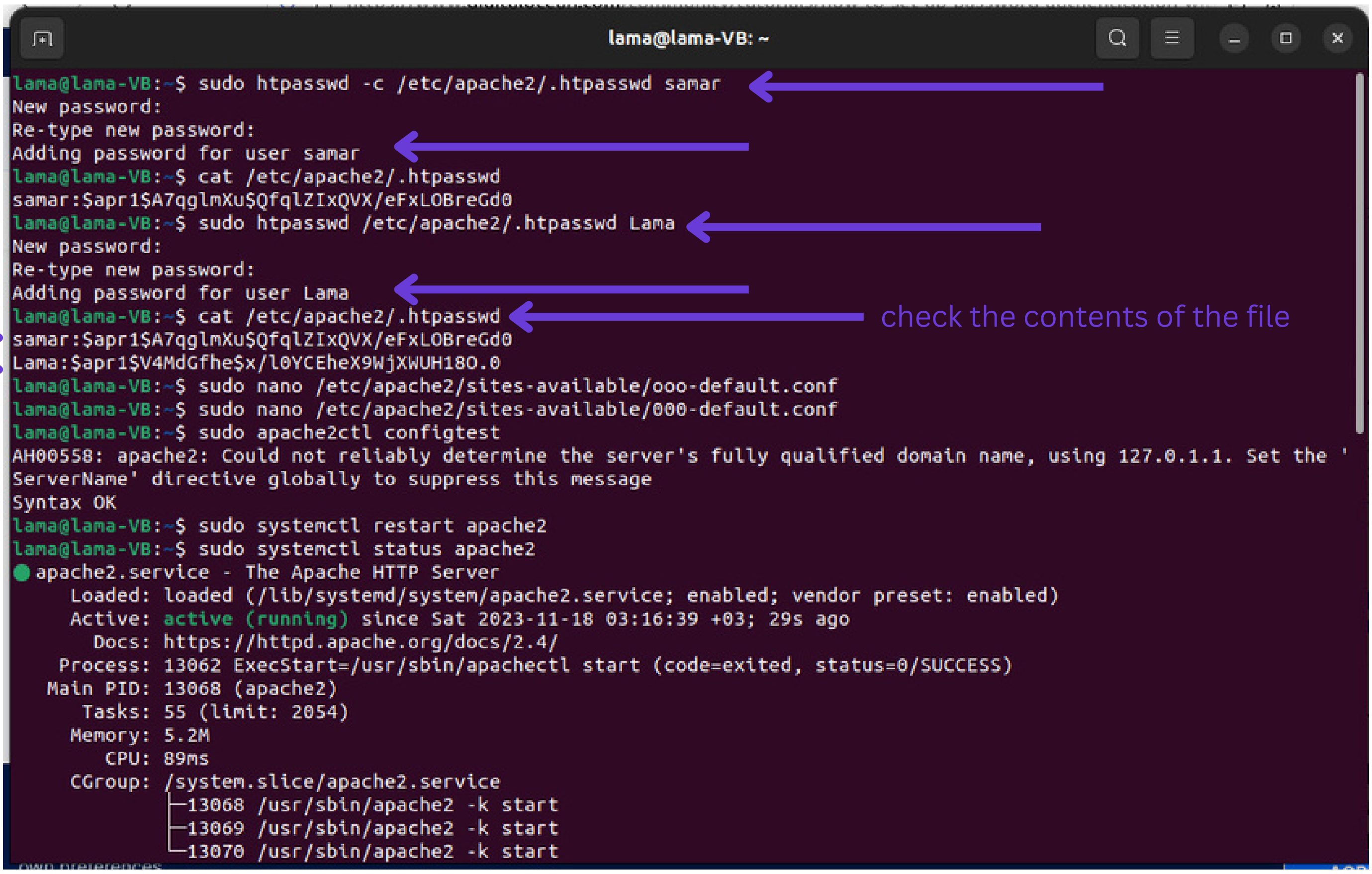
1)Creating the Password File

sudo htpasswd -c /etc/apache2/.htpasswd samar

The htpasswd command allows us to create a password file that Apache can use to authenticate users. we create a hidden file for this purpose called .htpasswd within our /etc/apache2 configuration directory. since we use this utility first time , the -c option to create the specified .htpasswd file. Here, we specify a username (**samar**) at the end of the command to create a new entry within the file:

sudo htpasswd /etc/apache2/.htpasswd Lama

(**Lama7**) here is an additional user



lama@lama-VB: ~

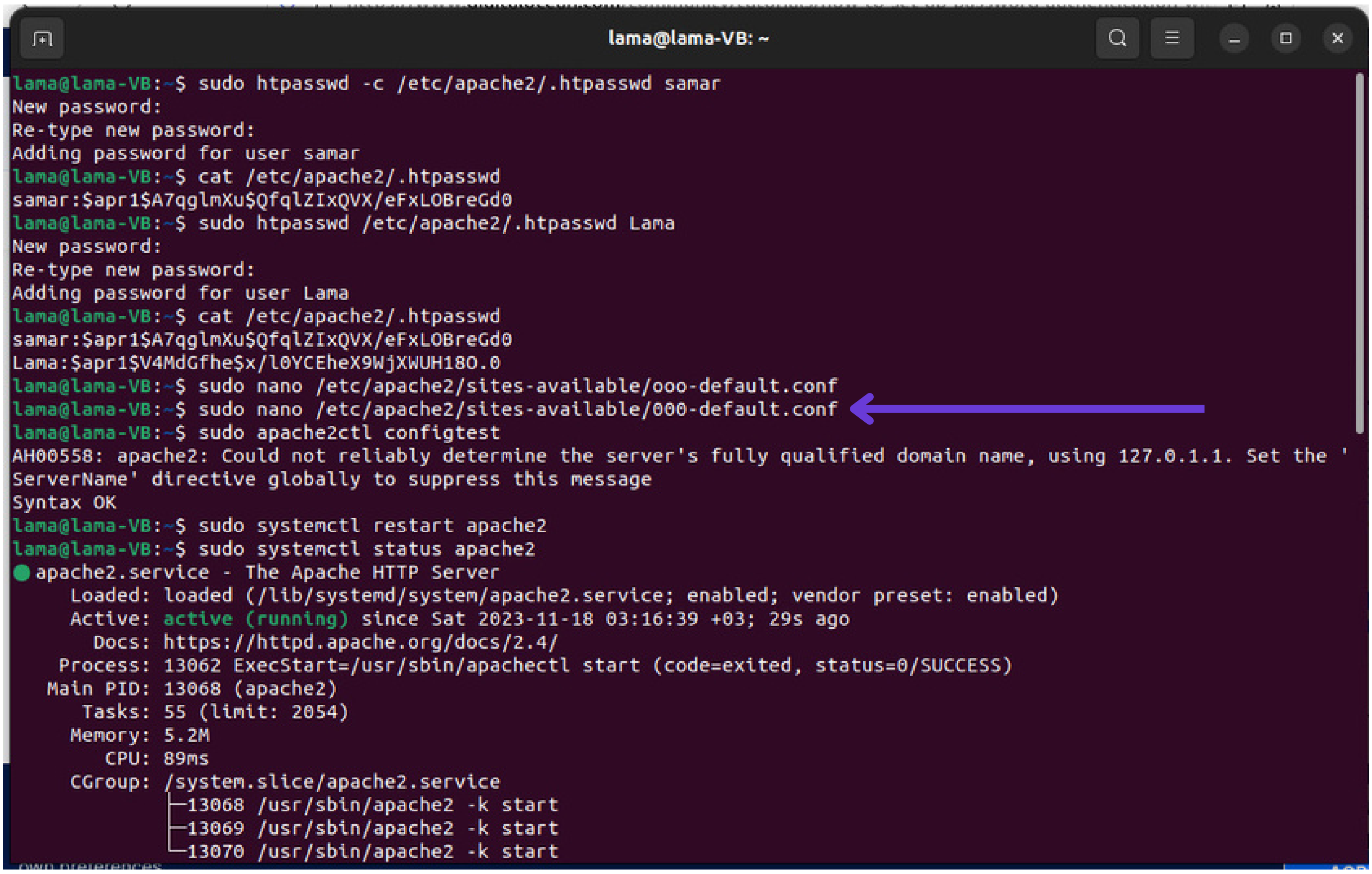
```
lama@lama-VB:~$ sudo htpasswd -c /etc/apache2/.htpasswd samar
New password:
Re-type new password:
Adding password for user samar
lama@lama-VB:~$ cat /etc/apache2/.htpasswd
samar:$apr1$A7qglmXu$QfqlZIxQVX/eFxLOBreGd0
lama@lama-VB:~$ sudo htpasswd /etc/apache2/.htpasswd Lama
New password:
Re-type new password:
Adding password for user Lama
lama@lama-VB:~$ cat /etc/apache2/.htpasswd
samar:$apr1$A7qglmXu$QfqlZIxQVX/eFxLOBreGd0
Lama:$apr1$V4MdGfhe$x/l0YCEheX9WjXWUH18O.0
lama@lama-VB:~$ sudo nano /etc/apache2/sites-available/ooo-default.conf
lama@lama-VB:~$ sudo nano /etc/apache2/sites-available/000-default.conf
lama@lama-VB:~$ sudo apache2ctl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the '
ServerName' directive globally to suppress this message
Syntax OK
lama@lama-VB:~$ sudo systemctl restart apache2
lama@lama-VB:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2023-11-18 03:16:39 +03; 29s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 13062 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 13068 (apache2)
      Tasks: 55 (limit: 2054)
     Memory: 5.2M
        CPU: 89ms
     CGroup: /system.slice/apache2.service
             ├─13068 /usr/sbin/apache2 -k start
             ├─13069 /usr/sbin/apache2 -k start
             └─13070 /usr/sbin/apache2 -k start
```

check the contents of the file

now to ensure that no one can access our welcome page except a **authenticated user** we do the following steps

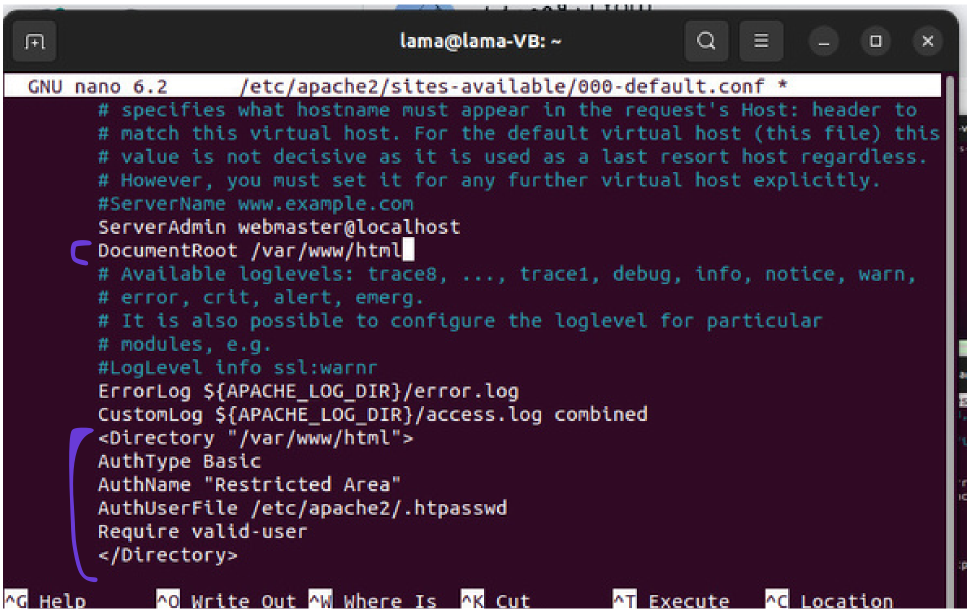
2)Configuring Apache Password Authentication

- we need to access (000-deault.conf)



```
lama@lama-VB:~$ sudo htpasswd -c /etc/apache2/.htpasswd samar
New password:
Re-type new password:
Adding password for user samar
lama@lama-VB:~$ cat /etc/apache2/.htpasswd
samar:$apr1$A7qglmXu$QfqlZIxQVX/eFxLOBreGd0
lama@lama-VB:~$ sudo htpasswd /etc/apache2/.htpasswd Lama
New password:
Re-type new password:
Adding password for user Lama
lama@lama-VB:~$ cat /etc/apache2/.htpasswd
samar:$apr1$A7qglmXu$QfqlZIxQVX/eFxLOBreGd0
Lama:$apr1$V4MdGfhe$x/l0YCEheX9WjXWUH18O.0
lama@lama-VB:~$ sudo nano /etc/apache2/sites-available/ooo-default.conf
lama@lama-VB:~$ sudo nano /etc/apache2/sites-available/000-default.conf
lama@lama-VB:~$ sudo apache2ctl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the '
ServerName' directive globally to suppress this message
Syntax OK
lama@lama-VB:~$ sudo systemctl restart apache2
lama@lama-VB:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2023-11-18 03:16:39 +03; 29s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 13062 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 13068 (apache2)
      Tasks: 55 (limit: 2054)
     Memory: 5.2M
        CPU: 89ms
     CGroup: /system.slice/apache2.service
             ├─13068 /usr/sbin/apache2 -k start
             ├─13069 /usr/sbin/apache2 -k start
             └─13070 /usr/sbin/apache2 -k start
```

- to add the following

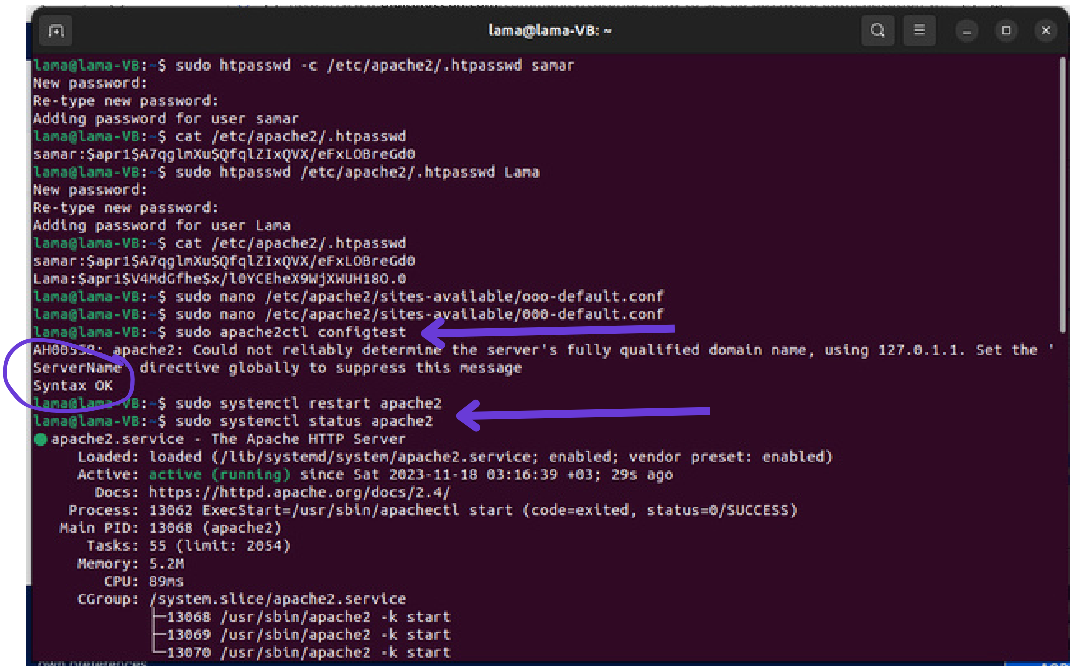


```
  GNU nano 6.2        /etc/apache2/sites-available/000-default.conf *
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com
        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html
        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warnr
        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
        <Directory "/var/www/html">
        AuthType Basic
        AuthName "Restricted Area"
        AuthUserFile /etc/apache2/.htpasswd
        Require valid-user
        </Directory>
^G Help      ^O Write Out ^W Where Is  ^K Cut        ^T Execute   ^C Location
```

now to ensure that no one can access our welcome page except a **authenticated user** we do the following steps

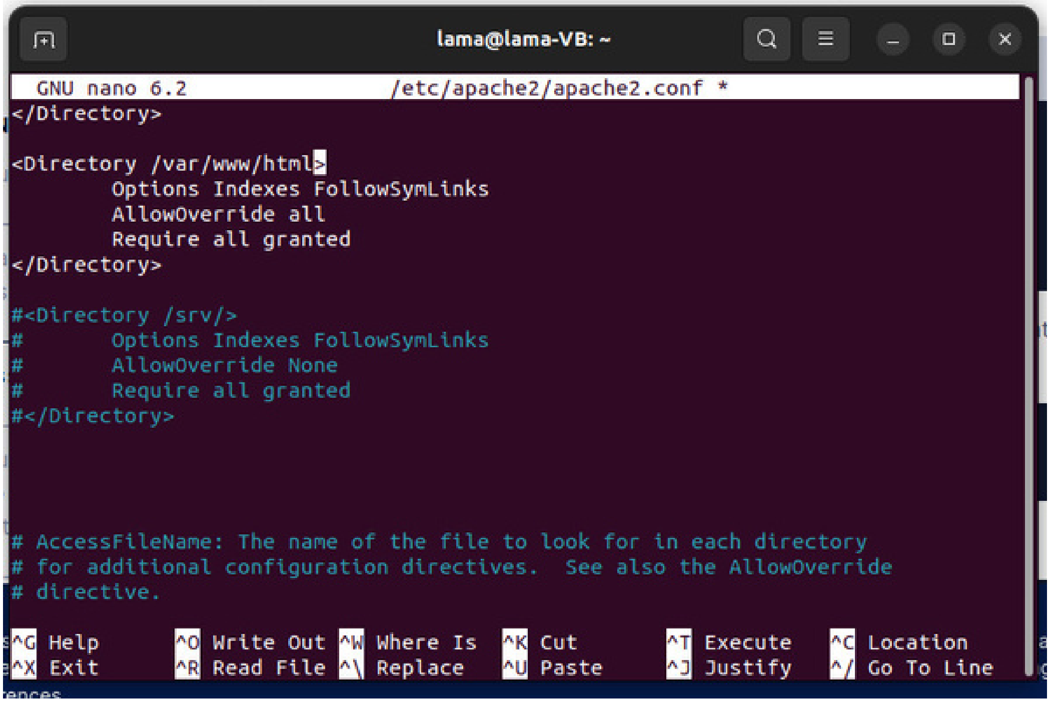
2)<u>Configuring Apache Password Authentication</u>

- Before restarting the web server, we need to check the configuration if its OK



```
lama@lama-VB:~$ sudo htpasswd -c /etc/apache2/.htpasswd samar
New password:
Re-type new password:
Adding password for user samar
lama@lama-VB:~$ cat /etc/apache2/.htpasswd
samar:$apr1$A7qglmXu$QfqlZIxQVX/eFxLOBreGd0
lama@lama-VB:~$ sudo htpasswd /etc/apache2/.htpasswd Lama
New password:
Re-type new password:
Adding password for user Lama
lama@lama-VB:~$ cat /etc/apache2/.htpasswd
samar:$apr1$A7qglmXu$QfqlZIxQVX/eFxLOBreGd0
Lama:$apr1$V4MdGfhe$x/l0YCEheX9WjXWUH18O.0
lama@lama-VB:~$ sudo nano /etc/apache2/sites-available/ooo-default.conf
lama@lama-VB:~$ sudo nano /etc/apache2/sites-available/000-default.conf
lama@lama-VB:~$ sudo apache2ctl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the '
ServerName' directive globally to suppress this message
Syntax OK
lama@lama-VB:~$ sudo systemctl restart apache2
lama@lama-VB:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2023-11-18 03:16:39 +03; 29s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 13062 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 13068 (apache2)
      Tasks: 55 (limit: 2054)
     Memory: 5.2M
        CPU: 89ms
     CGroup: /system.slice/apache2.service
             ├─13068 /usr/sbin/apache2 -k start
             ├─13069 /usr/sbin/apache2 -k start
             └─13070 /usr/sbin/apache2 -k start
```

- we chang the AllowOverride directive within that block from None to All.

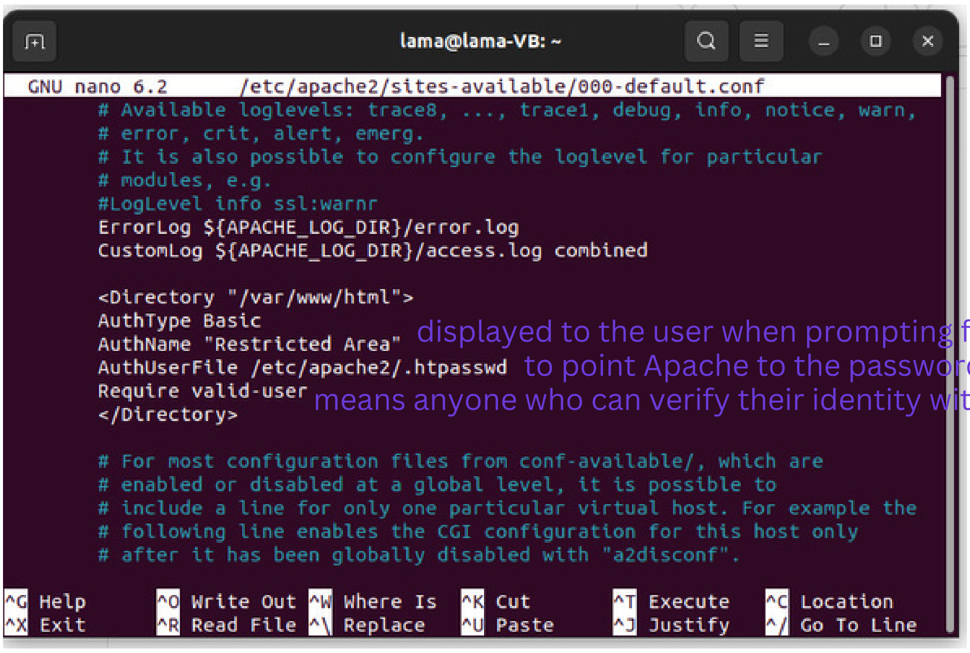


```
  GNU nano 6.2                    /etc/apache2/apache2.conf *
</Directory>

<Directory /var/www/html>
        Options Indexes FollowSymLinks
        AllowOverride all
        Require all granted
</Directory>

#<Directory /srv/>
#       Options Indexes FollowSymLinks
#       AllowOverride None
#       Require all granted
#</Directory>



# AccessFileName: The name of the file to look for in each directory
# for additional configuration directives.  See also the AllowOverride
# directive.

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line
```

now to ensure that no one can access our welcome page except a **authenticated user** we do the following steps

2)Configuring Apache Password Authentication

- Next, we add a .htaccess file to the directory we are wish to restrict. In our demonstration, we'll restrict the entire document root (the entire website) which is based at /var/www/html
- sudo nano /var/www/html/.htaccess



```
  GNU nano 6.2        /etc/apache2/sites-available/000-default.conf
        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warnr
        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        <Directory "/var/www/html">
        AuthType Basic
        AuthName "Restricted Area"
        AuthUserFile /etc/apache2/.htpasswd
        Require valid-user
        </Directory>

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".

^G Help       ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit       ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line
```

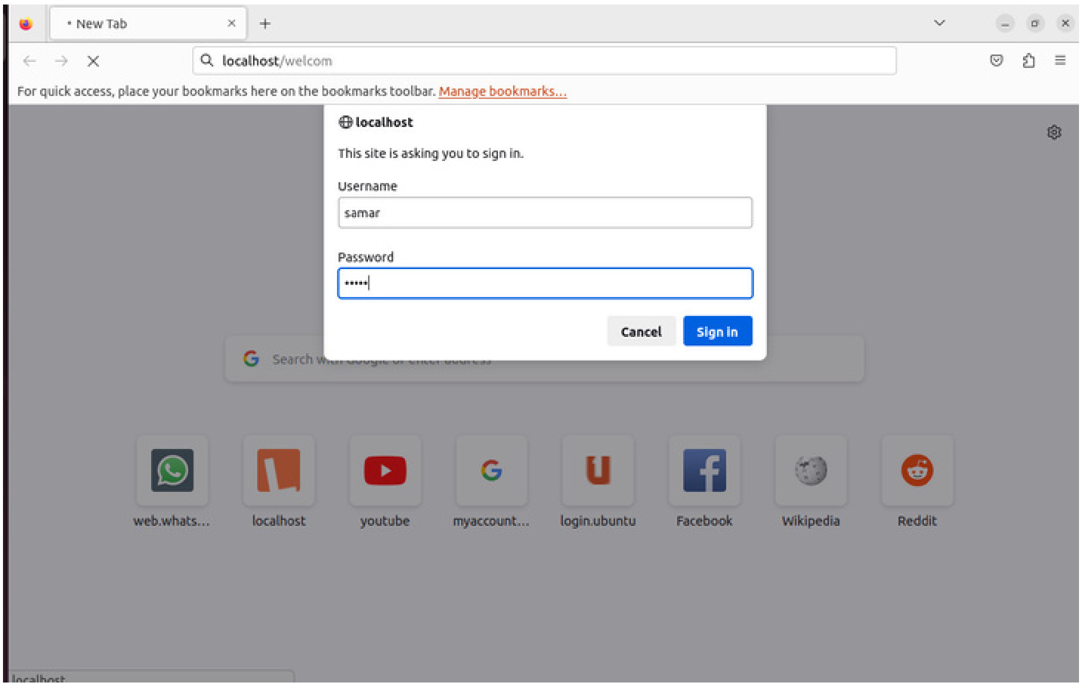
displayed to the user when prompting for credentials

to point Apache to the password file we created

means anyone who can verify their identity with a password will be allowed in

now to ensure that no one can access our welcome page except a **authenticated user** we do the following steps

3)Confirming Password Authentication



localhost/welcom

For quick access, place your bookmarks here on the bookmarks toolbar. Manage bookmarks...

localhost

This site is asking you to sign in.

Username: samar

Password: •••••

Cancel | Sign in

scan a barcode to better see the case of correct password and NOT

Summary

Linux's robust architecture and open-source nature allow for extensive customization and configuration options, making it an ideal platform for web hosting. The inherent security features of Linux, including user and permission management, firewall capabilities, and access controls, contribute to creating a secure hosting environment.

While Linux distributions offer regular security updates and patches, the management of software updates and dependencies can be demanding. Compatibility issues may arise when integrating with other open-source technologies like PHP, Python, MySQL, and PostgreSQL, requiring meticulous attention to ensure seamless functionality.

In addition to these challenges, despite supporting various web protocols, including HTTP/1.1, HTTP/2, and HTTPS, Linux web servers may face difficulties in scenarios requiring specialized protocols or when compatibility issues arise between different protocol versions. Despite these limitations, Linux remains a reliable, scalable, and customizable hosting platform, making it a popular choice for businesses and individuals alike.

REFERENCES

how-to-set-up-password-authentication-with-apache-on-ubuntu-20-04

How web server works

How To Host Web Server

Famous web servers