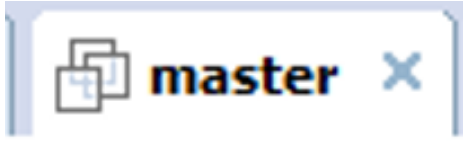
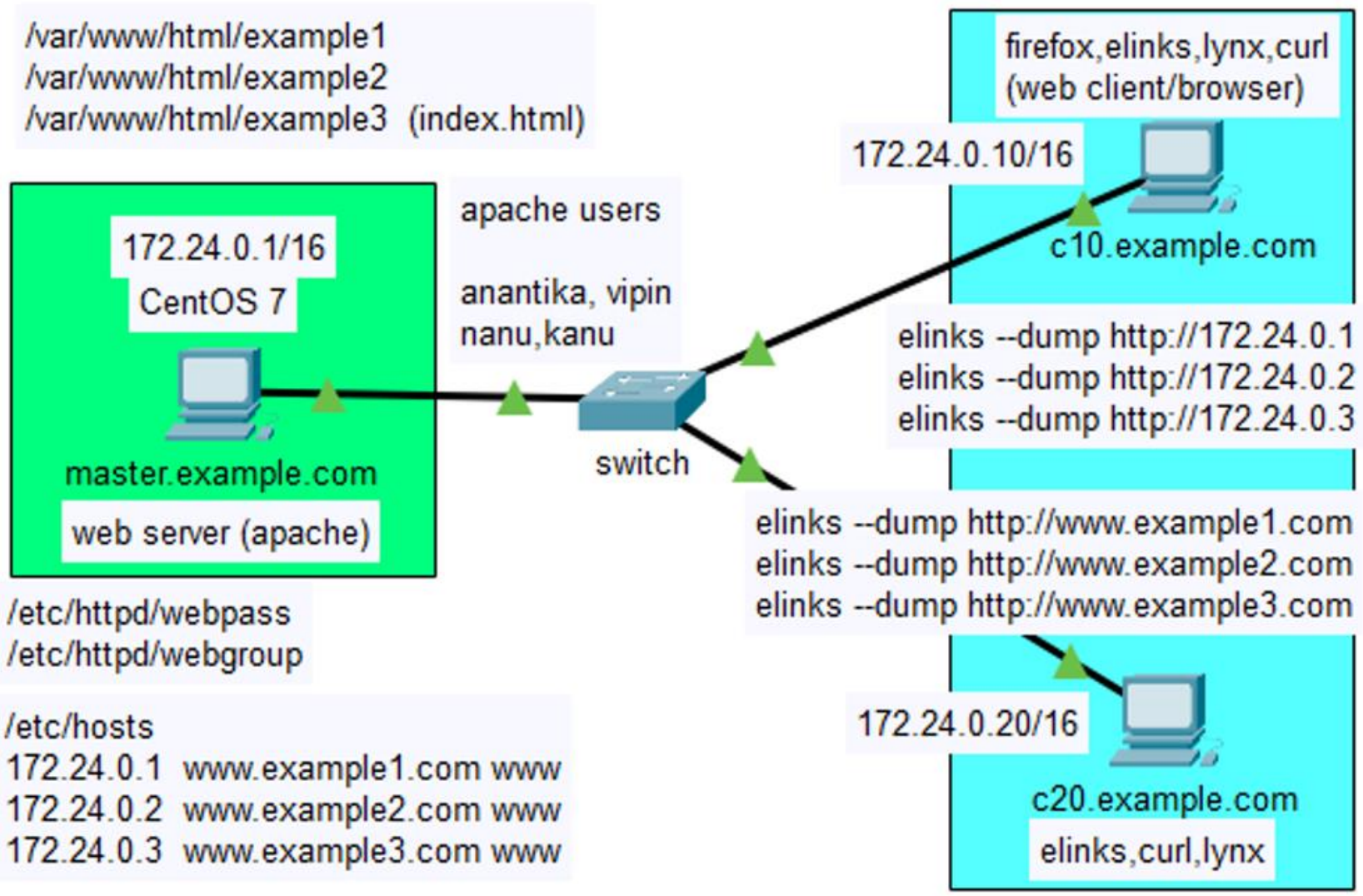
Apache User Authentication

(Authenticated Users Should Be Able To View Web Pages)



Vipin Gupta
BE,RHCE,CEH,CCNA,MCSE,MCSA
vipin2411@gmail.com
Mobile: 93563-10379
www.linuxexpert.in

Apache User Authentication



master c10 c20

Create Apache Users

We want only authenticated users should be able to view web pages. Use "htpasswd" command to create username/password combinations. Use "-c" option to create new file. It creates file, if already not there. It overwrite, if file exists, "-m" means use MD5 encryption)

```
[root@master ~]# htpasswd -c -m /etc/httpd/webpass anantika
New password:
Re-type new password:
Adding password for user anantika
[root@master ~]#
[root@master ~]# htpasswd -m /etc/httpd/webpass vipin
New password:
Re-type new password:
Adding password for user vipin
[root@master ~]#
[root@master ~]# htpasswd -m /etc/httpd/webpass kanu
New password:
Re-type new password:
Adding password for user kanu
[root@master ~]#
[root@master ~]# htpasswd -m /etc/httpd/webpass nanu
New password:
Re-type new password:
Adding password for user nanu
```

Create 4 users "anantika", "vipin", "kanu", "nanu" and store information in "/etc/httpd/webpass" file

```
[root@master ~]# cat /etc/httpd/webpass
anantika:$apr1$50DW2bBw$g0..FRiRs/RaW.YeE6n0m0
vipin:$apr1$9GzztQIT$HFbjaoVXFLl4yl0Iz1mkf.
kanu:$apr1$Fz3kLZW0$CqfOtoPvfHpYGVOXma67.1
nanu:$apr1$MdmiM566$zMFj/LE0JMnBpxxkFlwCu/
```

Edit “httpd.conf”

Edit “/etc/httpd/conf/httpd.conf” file and add the “Directory” directive. Since contents for “172.24.0.1” are in “/var/www/html/example1” directory, so we have started the directive as “<Directory /var/www/html/example1>”.

AuthType: Here we are using “Basic” authentication. In this authentication, password will be sent in clear text. Other is “digest” which we are not using.

AuthName: This option will pop up the message “Restricted Access” when you will try to access protected pages. You can specify any message you want.

AuthUserFile: Option is referring to “/etc/httpd/webpass” file where user authentication details are stored. Again it could be any name.

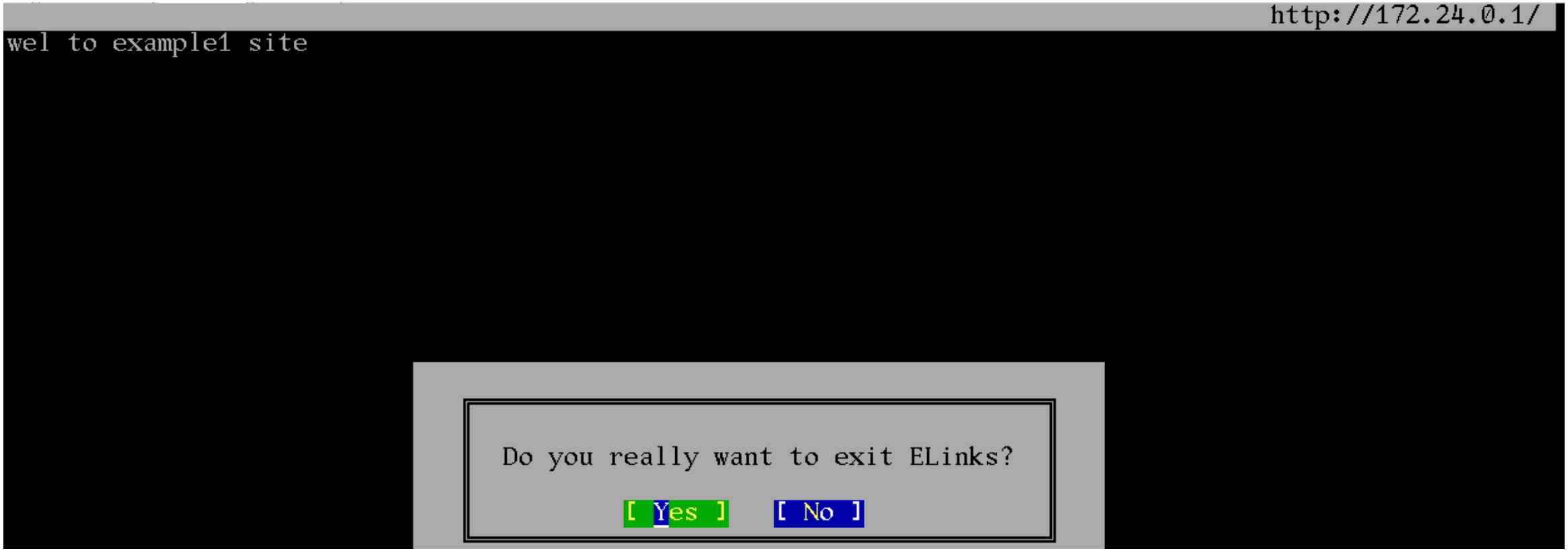
Require user: The “Require user anantika nanu” option will instruct the web server to show web pages only to “anantika” and “nanu” users.

```
[root@master ~]# tail -7 /etc/httpd/conf/httpd.conf

<Directory /var/www/html/example1>
        AuthType Basic
        AuthName "Restricted Access"
        AuthUserFile "/etc/httpd/webpass"
        Require user anantika nanu
</Directory>
[root@master ~]#
[root@master ~]# service httpd configtest
Syntax OK
[root@master ~]#
[root@master ~]# systemctl reload httpd
[root@master ~]#
```

Verify Authentication

Verify Authentication



Verify Authentication

```
[root@c10 ~]# elinks --dump http://anantika:123456@172.24.0.1
   wel to example1 site
[root@c10 ~]#
[root@c10 ~]# elinks --dump http://nanu:123456@172.24.0.1
   wel to example1 site
[root@c10 ~]#
[root@c10 ~]# elinks --dump http://vipin:123456@172.24.0.1
                                    Unauthorized

   This server could not verify that you are authorized to access the
   document requested. Either you supplied the wrong credentials (e.g., bad
   password), or your browser doesn't understand how to supply the
   credentials required.
[root@c10 ~]#
[root@c10 ~]# elinks --dump http://kanu:123456@172.24.0.1
                                    Unauthorized
```

Use .htaccess

Instead of putting authentication directives in "/etc/httpd/conf/httpd.conf" file, we can create ".htaccess" file in "/var/www/html/example1" directory.

By simply creating ".htaccess" file in relevant directory will have no effect. To make this file effective, you have to change the value of option "AllowOverride" from "None" to "AuthConfig" in configuration file "httpd.conf".

```
[root@master ~]# cd /var/www/html/example1/
[root@master example1]#
[root@master example1]# cat >.htaccess
AuthType Basic
AuthName "Restricted Access"
AuthUserFile "/etc/httpd/webpass"
Require user anantika nanu
[root@master example1]#
[root@master example1]# ls -a
.  ..  .htaccess  index.html
[root@master example1]#
[root@master example1]# cat .htaccess
AuthType Basic
AuthName "Restricted Access"
AuthUserFile "/etc/httpd/webpass"
Require user anantika nanu
[root@master example1]#
```

Use .htaccess

Add the line “AllowOverride AuthConfig” in “httpd.conf” in “<Directory /var/www/html/example1>” directive.

```
[root@master example1]# tail -4 /etc/httpd/conf/httpd.conf

<Directory /var/www/html/example1>
        AllowOverride AuthConfig
</Directory>
[root@master example1]#
[root@master example1]# service httpd configtest
Syntax OK
[root@master example1]#
[root@master example1]# systemctl reload httpd
[root@master example1]#
```

Do not change “<Directory />” directive.

```
<Directory />
    AllowOverride none
    Require all denied
</Directory>
```

Verify Authentication

```
[root@c10 ~]# elinks --dump http://anantika:123456@172.24.0.1
   wel to example1 site
[root@c10 ~]#
[root@c10 ~]# elinks --dump http://vipin:123456@172.24.0.1
                                    Unauthorized

   This server could not verify that you are authorized to access the
   document requested. Either you supplied the wrong credentials (e.g., bad
   password), or your browser doesn't understand how to supply the
   credentials required.
[root@c10 ~]#
```

Here “anantika” is username and “123456” is the password.

Use “Require valid-user”

```
[root@master example1]# cat .htaccess
AuthType Basic
AuthName "Restricted Access"
AuthUserFile "/etc/httpd/webpass"
Require valid-user
```

Change “/var/www/html/example1/.htaccess” file. We have changed the parameter to “Require valid-user”. Now all the users in “/etc/httpd/webpass” file will be given access.

Verify Authentication

```
[root@c10 ~]# elinks --dump http://anantika:123456@172.24.0.1
   wel to example1 site
[root@c10 ~]#
[root@c10 ~]# elinks --dump http://vipin:123456@172.24.0.1
   wel to example1 site
[root@c10 ~]#
[root@c10 ~]# elinks --dump http://kanu:123456@172.24.0.1
   wel to example1 site
[root@c10 ~]#
[root@c10 ~]# elinks --dump http://nanu:123456@172.24.0.1
   wel to example1 site
```

Use “Require group”

```
[root@master example1]# cat /etc/httpd/webgroup
webadmin: nanu kanu
operator: vipin anantika
[root@master example1]#
[root@master example1]# cat .htaccess
AuthType Basic
AuthName "By Invitation Only"
AuthUserFile "/etc/httpd/webpass"
AuthGroupFile "/etc/httpd/webgroup"
Require group webadmin
```

Create file “/etc/httpd/webgroup”. In this file create 2 groups “webadmin” having users “nanu”, “kanu” and “operator” group having users “vipin”, ”anantika”. Change “/var/www/html/example1/.htaccess” file. We have changed the parameter to “Require group webadmin”. Now all the users in “webadmin” group will be given access while that of “operator” group will be denied access.

Verify Authentication

```
[root@c10 ~]# elinks --dump http://nanu:123456@172.24.0.1
   wel to example1 site
[root@c10 ~]#
[root@c10 ~]# elinks --dump http://kanu:123456@172.24.0.1
   wel to example1 site
[root@c10 ~]#
[root@c10 ~]# elinks --dump http://anantika:123456@172.24.0.1
                                Unauthorized
```

Remove .htaccess & Related Settings in httpd.conf

```
[root@master example1]# ls -a
.  ..  .htaccess  index.html
[root@master example1]#
[root@master example1]# rm -rf .htaccess
[root@master example1]#
```