**CYBER SECURITY LAB-3 (PKI)**

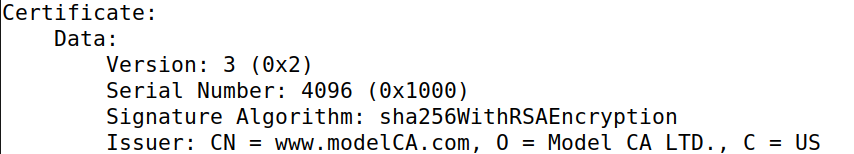
**(CMPG769)**

**ASTHA PATEL**

**Task 1: Becoming a Certificate Authority (CA):**

1. **What part of the certificate indicates this is a CA’s certificate?**

**Ans:**



1. **What part of the certificate indicates this is a self-signed certificate?**

**Ans:**

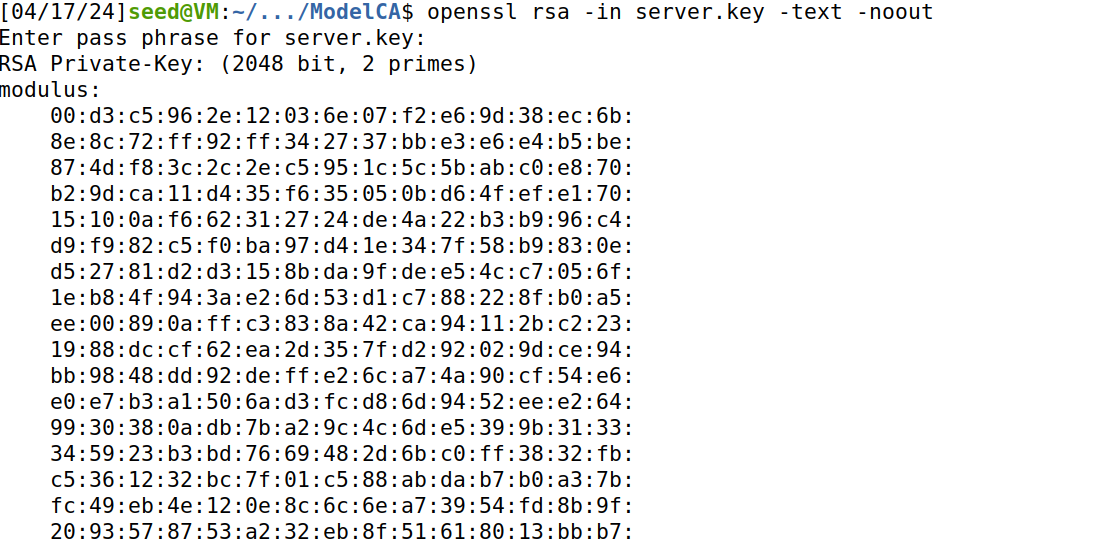
For this lab I used bank32 data to perform this lab which is shared in module.

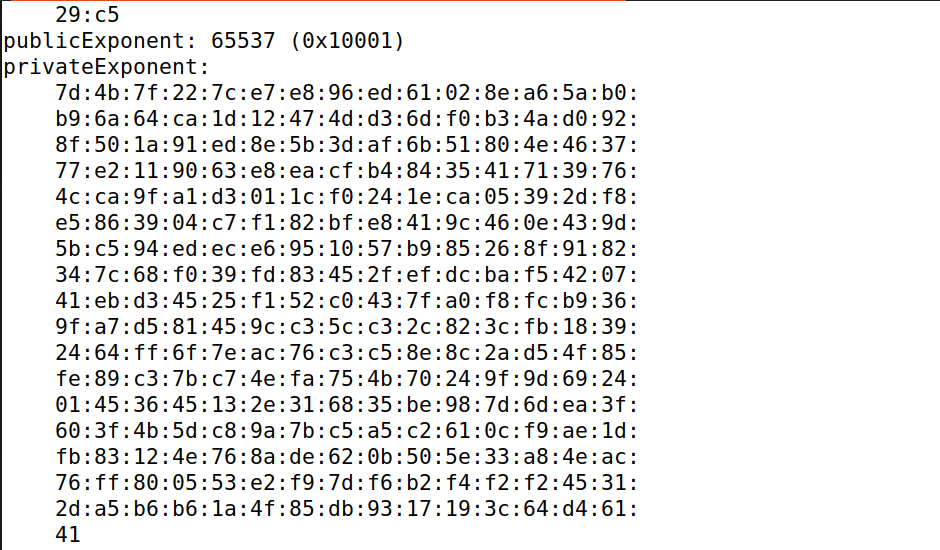


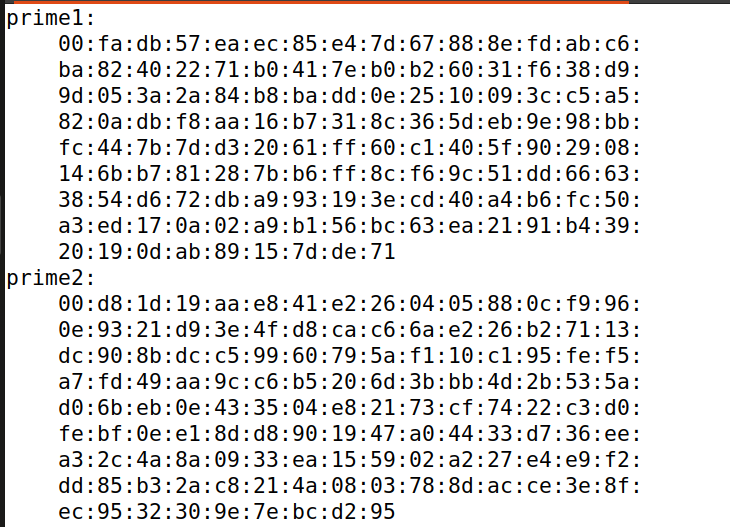
1. **In the RSA algorithm, we have a public exponent e, a private exponent d, a modulus n, and two secret numbers p and q, such that n = pq. Please identify the values for these elements in your certificate and key files.**

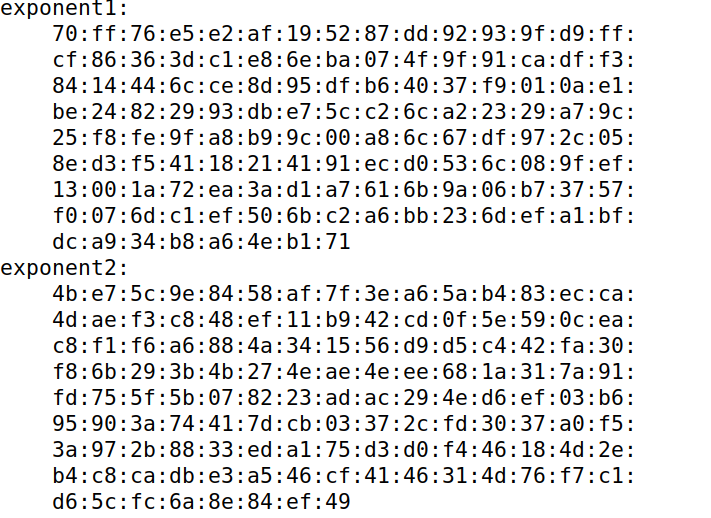
**Ans:**

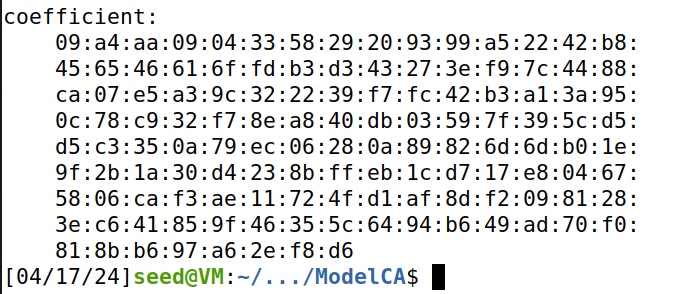
Here are some screenshot that shows the result that asked in this question:



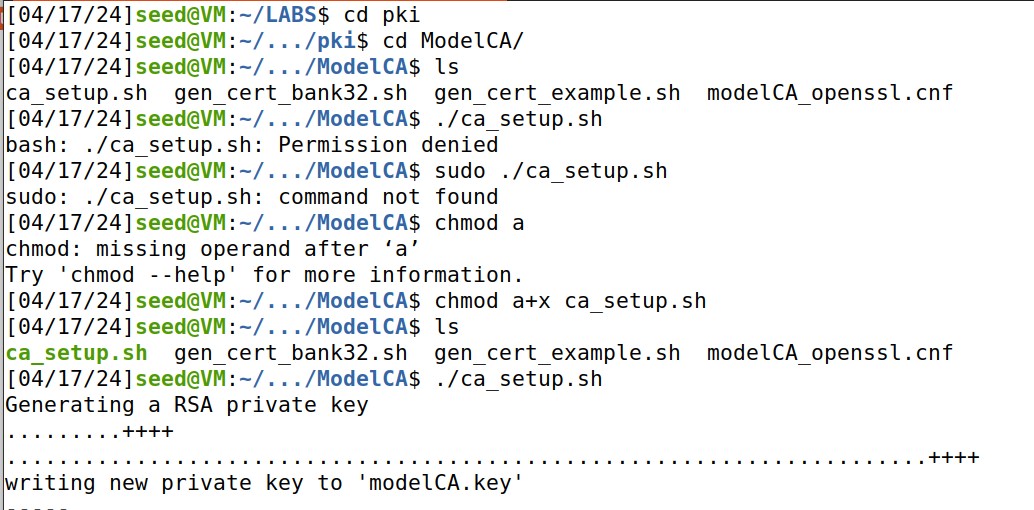




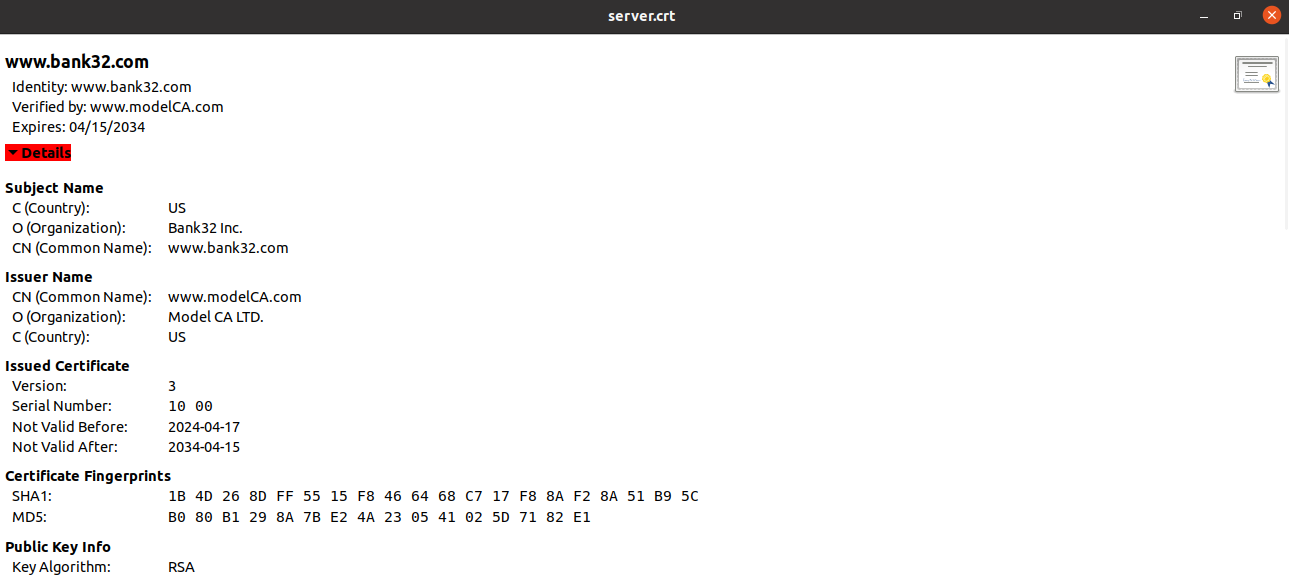


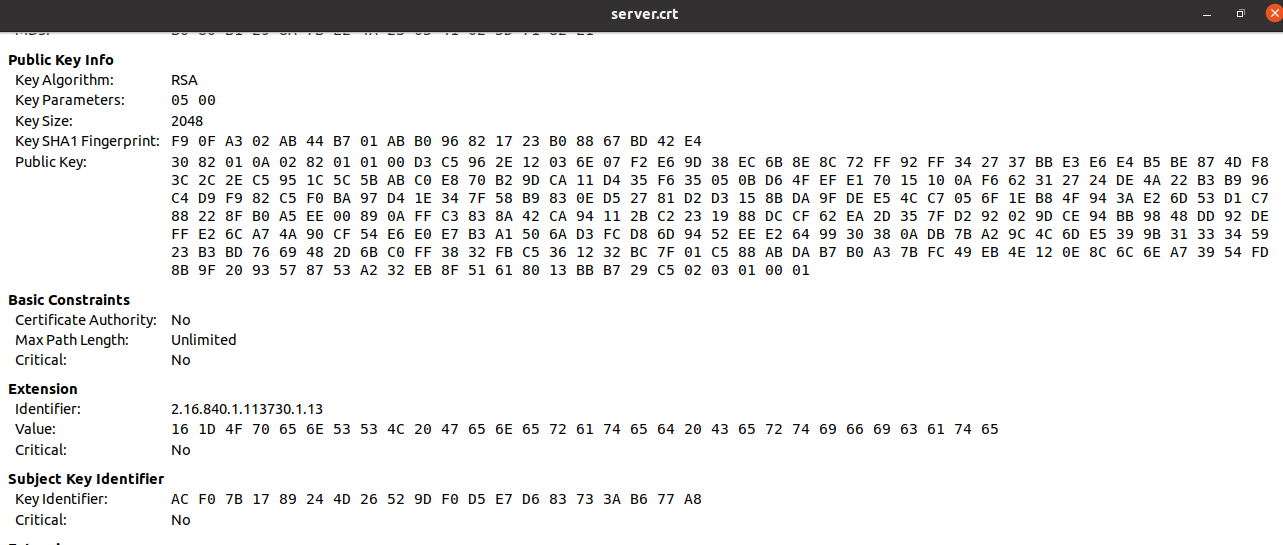


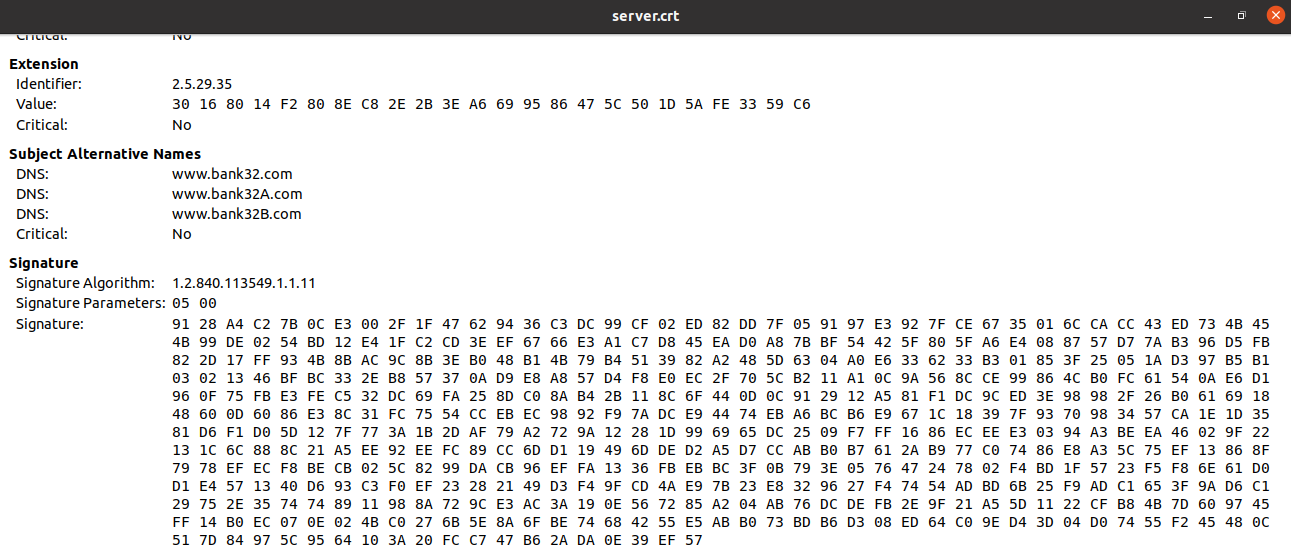
First, we have to run program: server.crt to generate a certificate request for our web server.



Following is the generated certificate:



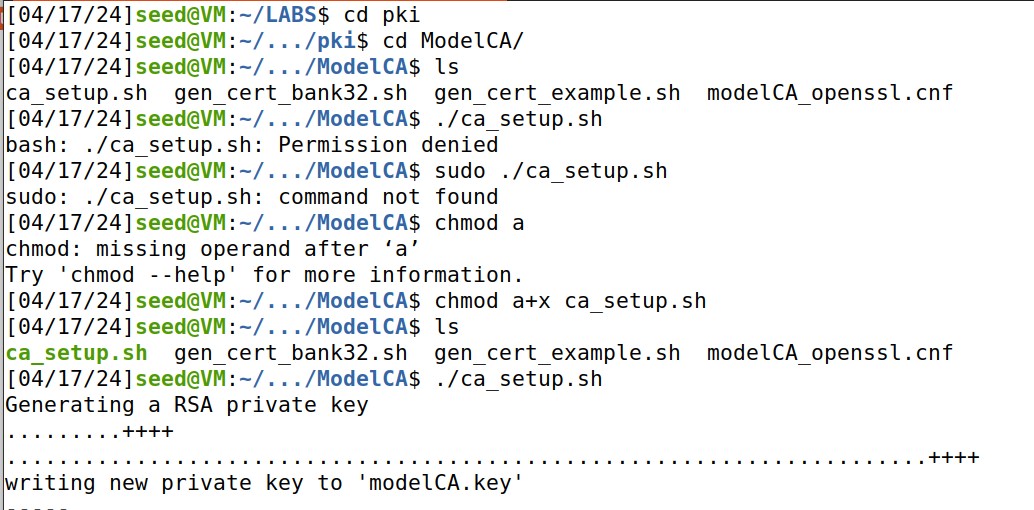


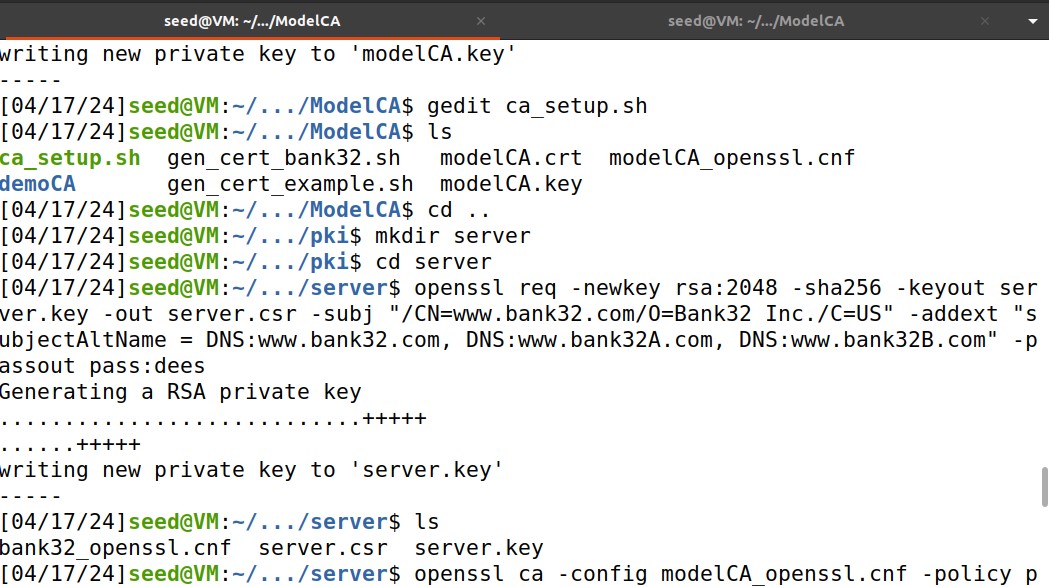


**Task 2: Generating a Certificate Request for Your Web Server:**

The following command generates a CSR for [www.bank32.com](http://www.bank32.com) and we will add this following command in ca\_setup.sh and run it to generate a certificate request for my web server using:

openssl req -newkey rsa:2048 -sha256 \-keyout server.key -out server.csr \ -subj"/CN=www.bank32.com/O=Bank32 Inc./C=US" \-passout pass:dees



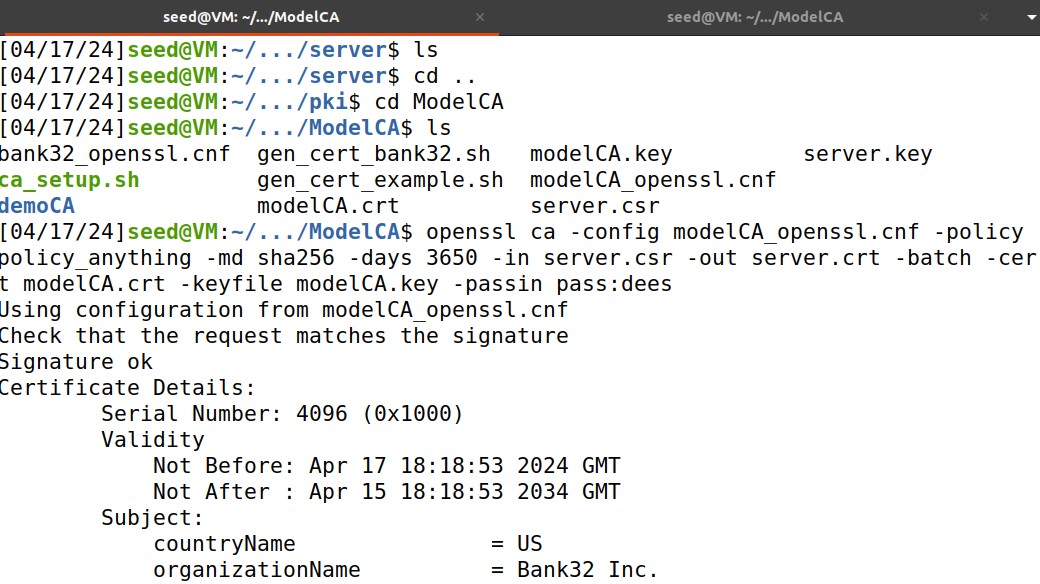


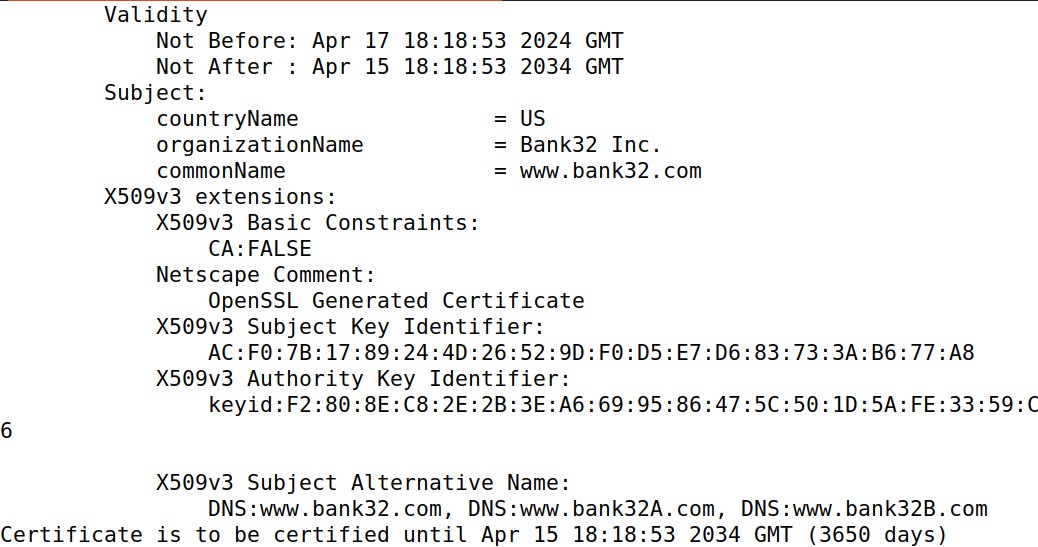
Two alternative names to your certificate signing request are: [www.bank32A.com](http://www.bank32A.com) and [www.bank32B.com](http://www.bank32B.com).

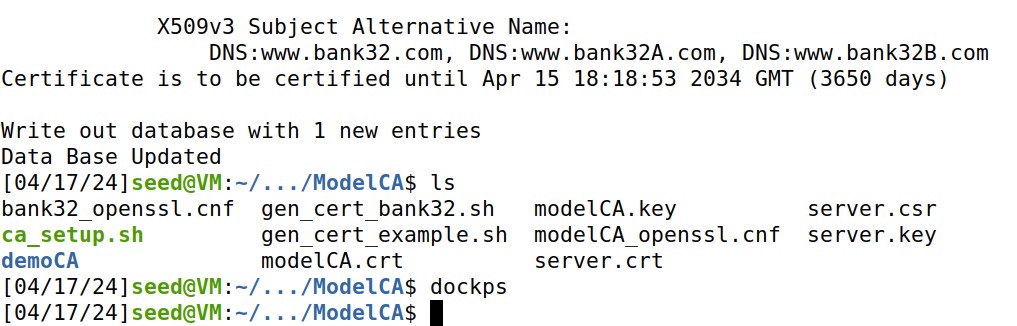
**Task 3: Generating a Certificate for your server:**

By using following command we will generate certificate for our bank server:

openssl ca -config modelCA\_openssl.cnf -policy policy\_anything -md sha256 -days 3650 -in server.csr -out server.crt -batch -cert modelCA.crt -keyfile modelCA.key -passin pass:dees





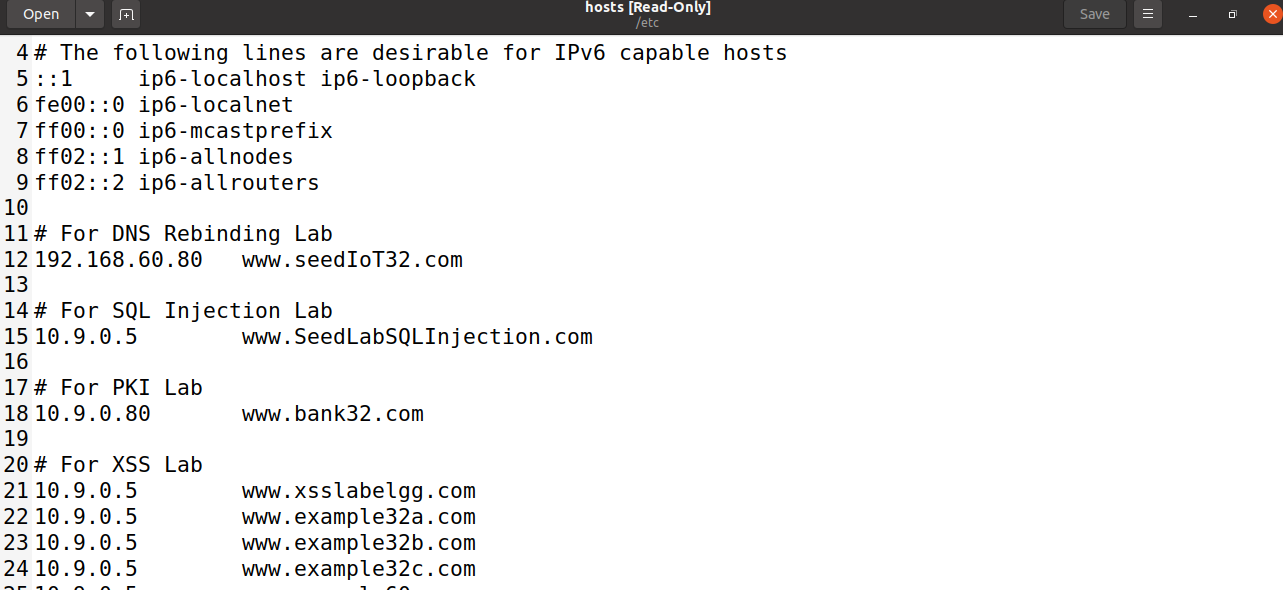


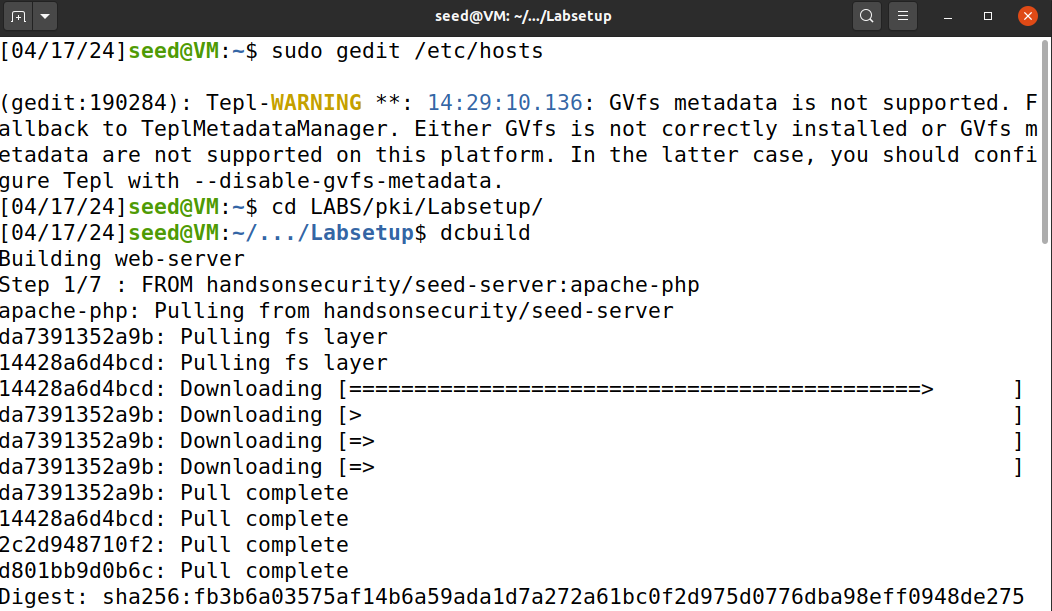
**Task 4: Deploying Certificate in an Apache-Based HTTPSWebsite**

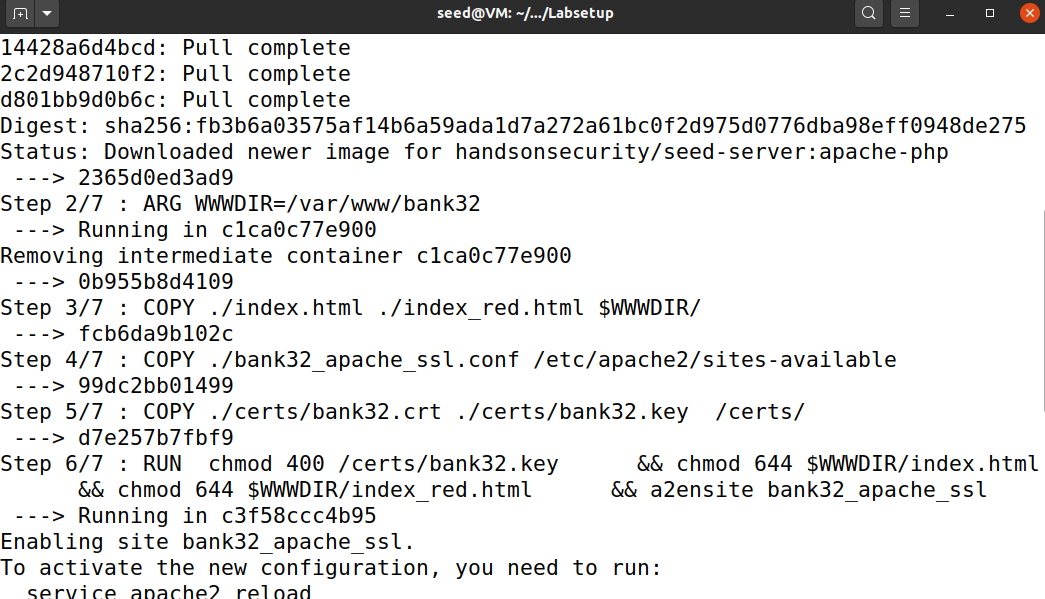
In this task 4 we will set up lab environment and DNS.

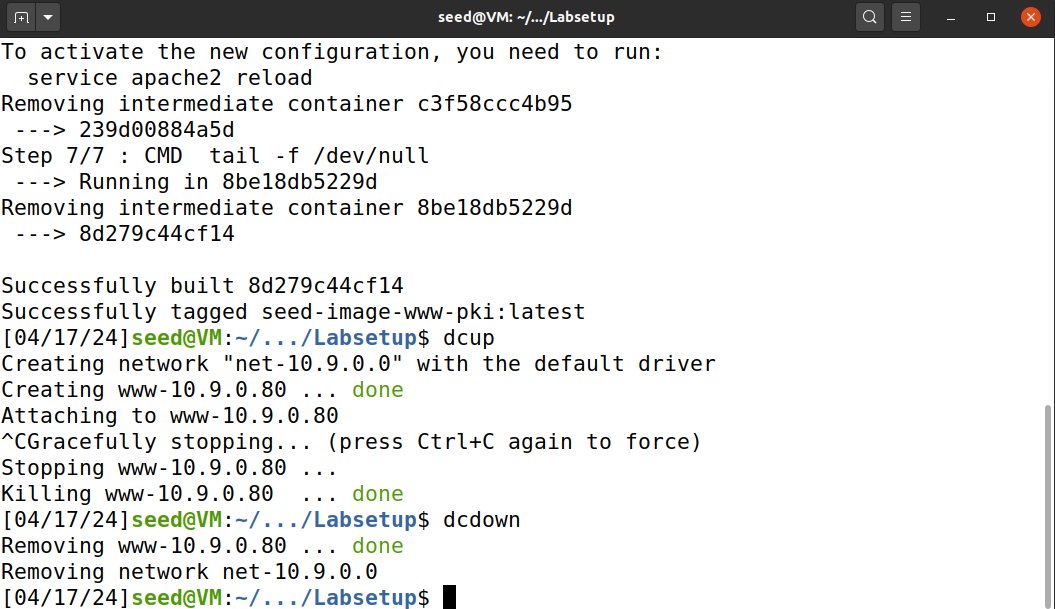
First we will come out of all the directory and run: sudo gedit /etc/hosts command and add line line of code in it which is follows: 10.9.0.80 www.bank32.com and then we will go back to our labsetup file and perform next steps.

For that we will use following command: “dcbuild” in lab setup folder:

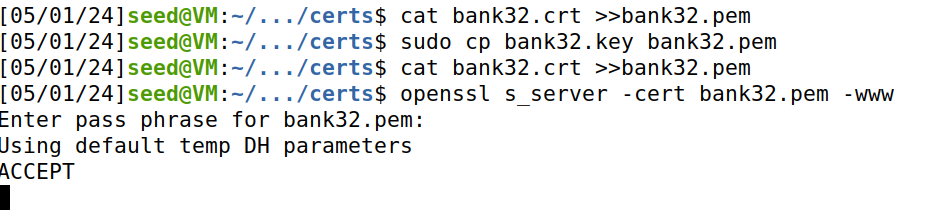




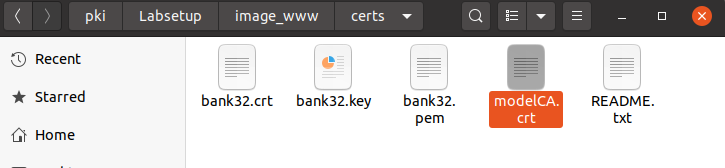


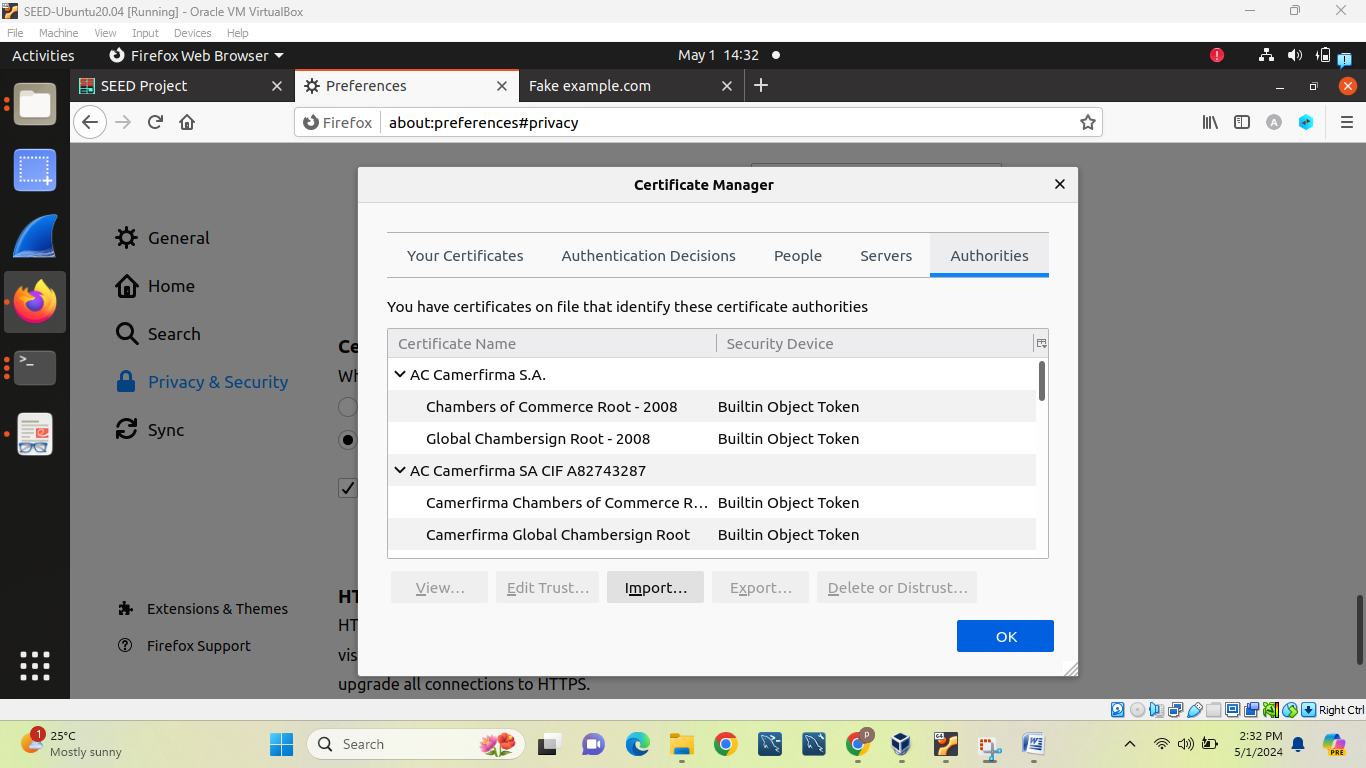


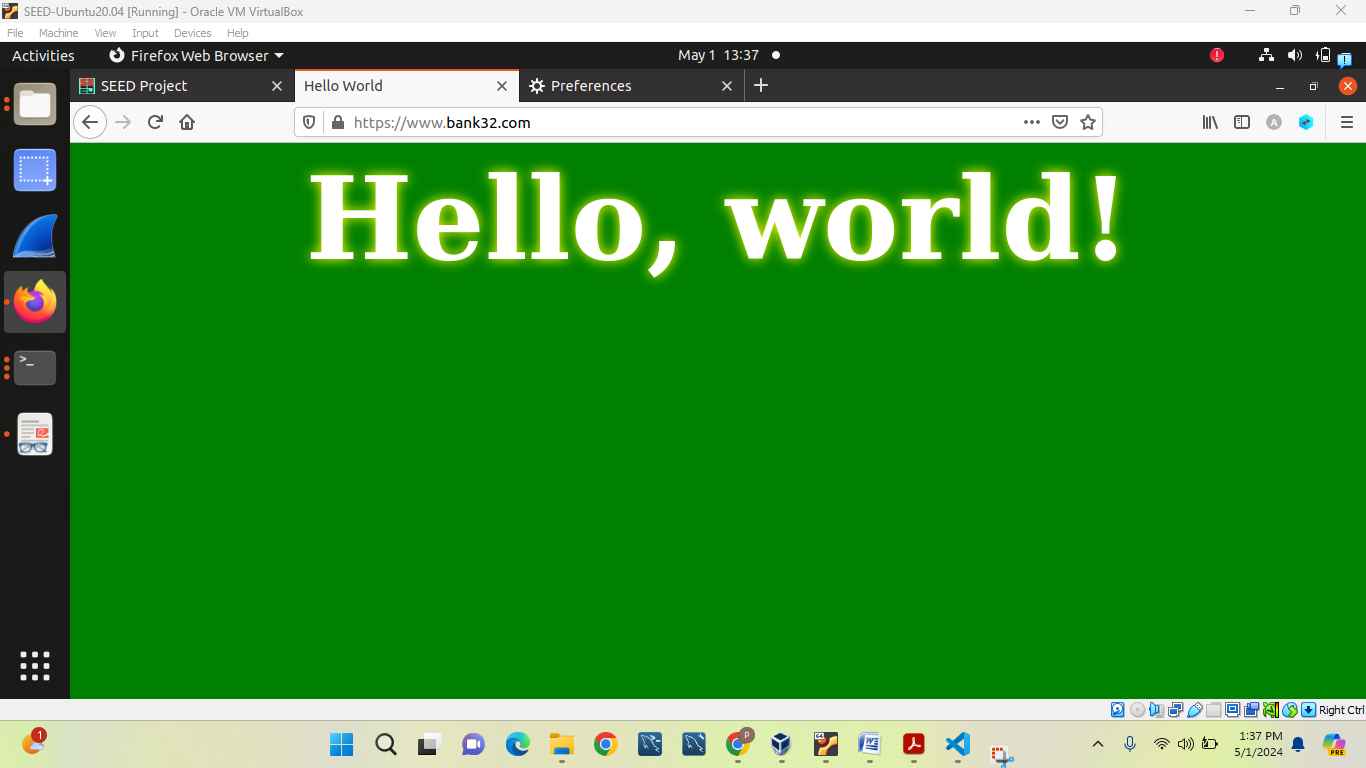
Whenever we want to perform any task we will use dcup and whenever we want to turn off our ubuntu for this task we will use dcdown to avoid any crashes.

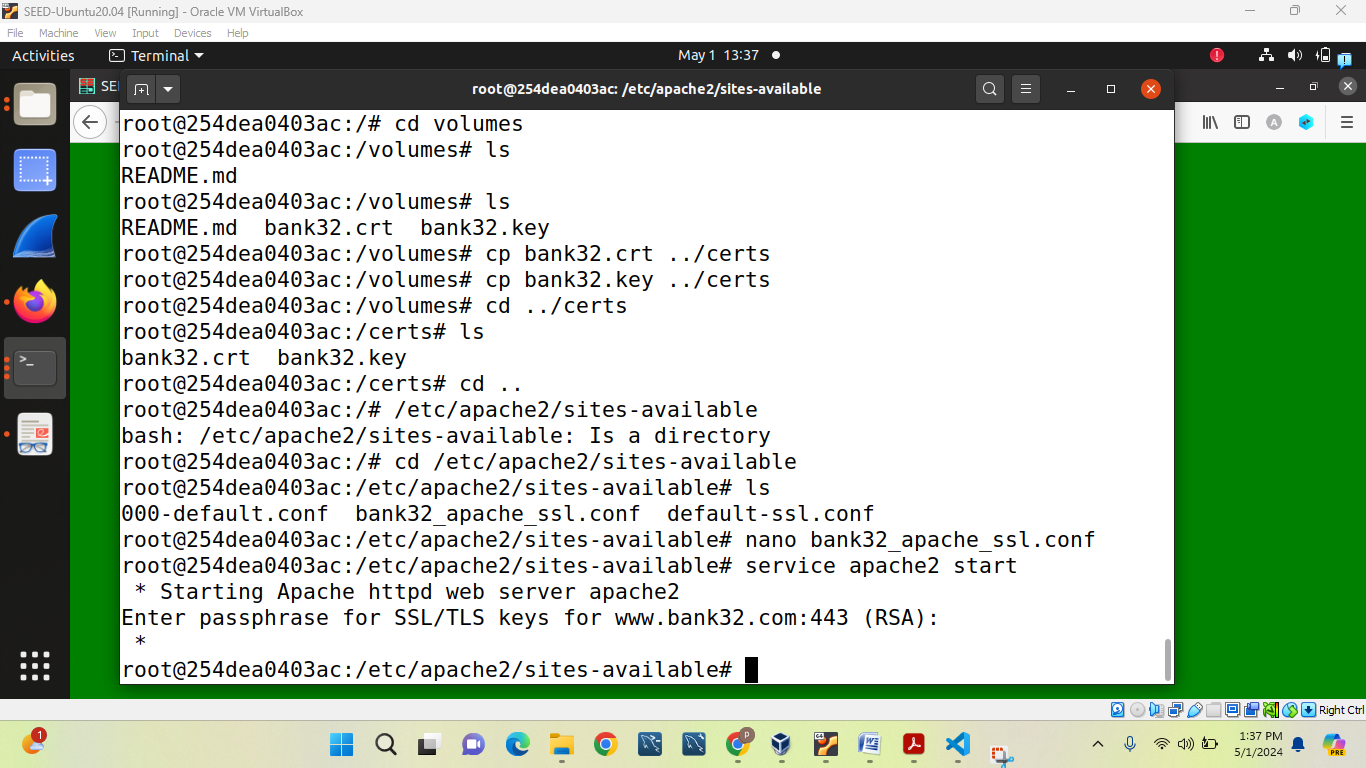


Add certificate in the browser so we can access our certification authority in browser by importing modelCA.crt certificate here from the following folder path.



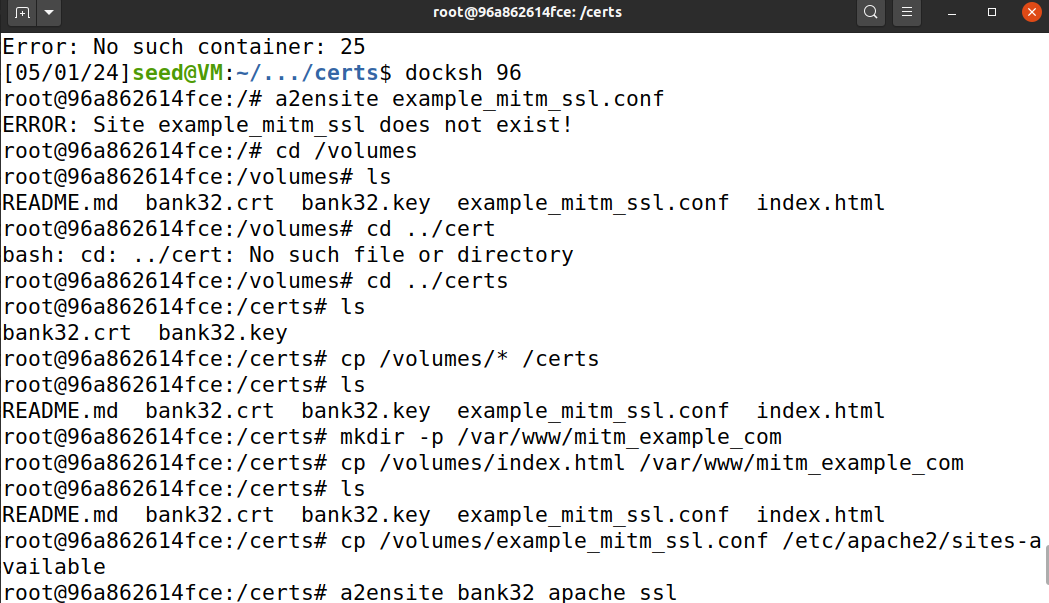


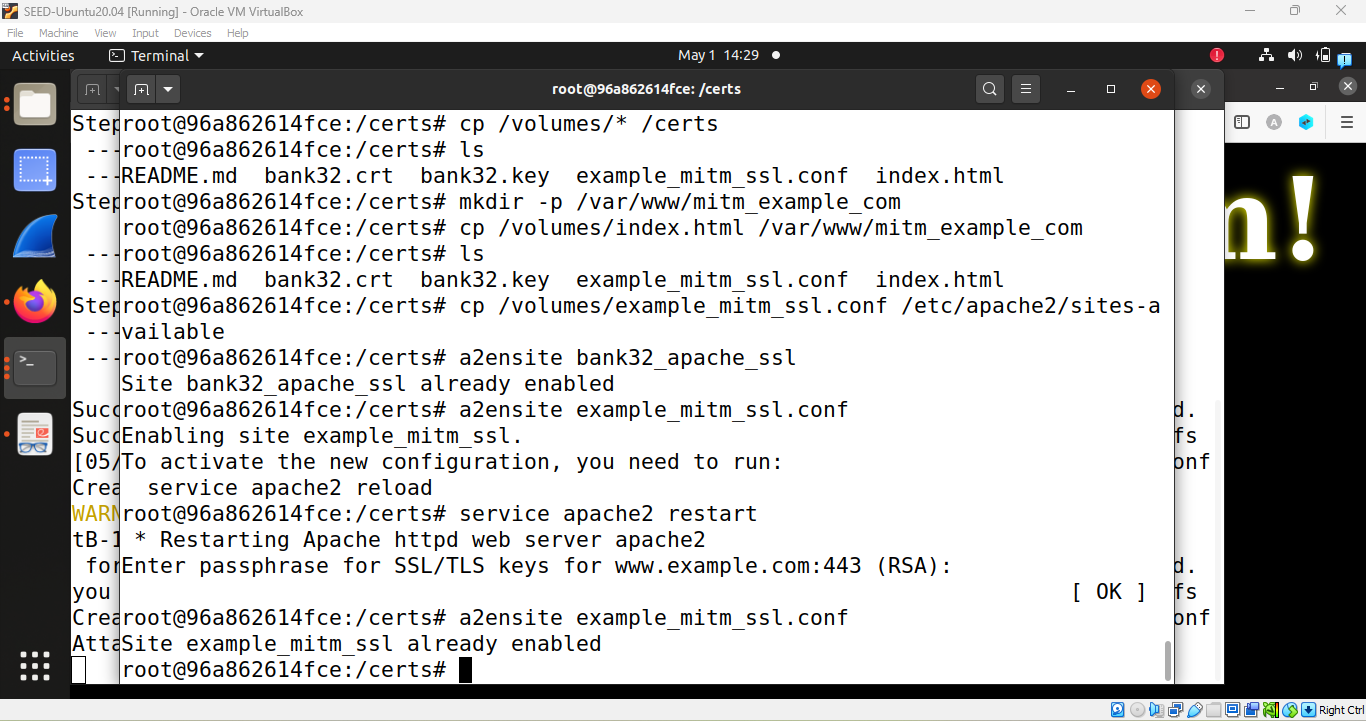




**Task 5: Launching a Man-In-The-Middle Attack**

In my case, I tried multiple times to run this queries but I am not getting error page it shows directly updated page that we want in task 6 instead.





**Task 6: Launching a Man-In-The-Middle Attack with a Compromised CA**

In this task, we perform same operation as task-5 and as a result we will able to launching a Man-In-The-Middle Attack with compromised CA.

