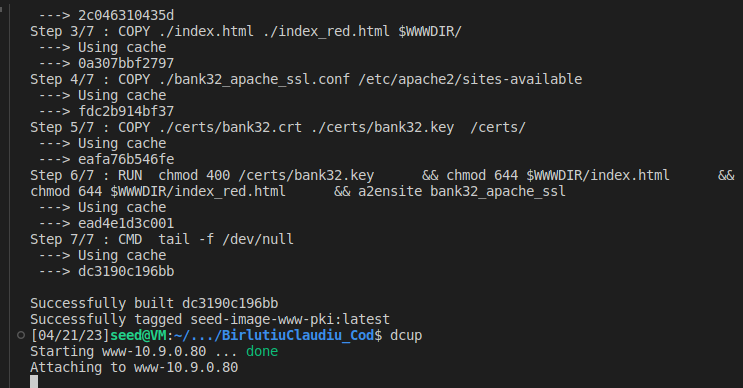
# Raport pentru lucrarea 6: Infrastructura de chei publice (PKI)

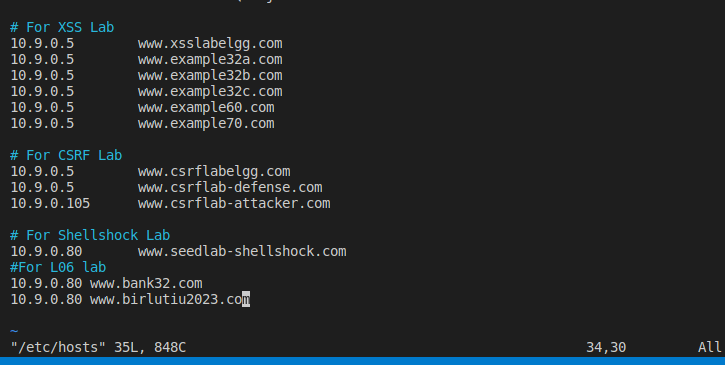
Autor: Birlutiu Claudiu-Andrei

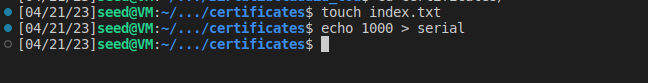
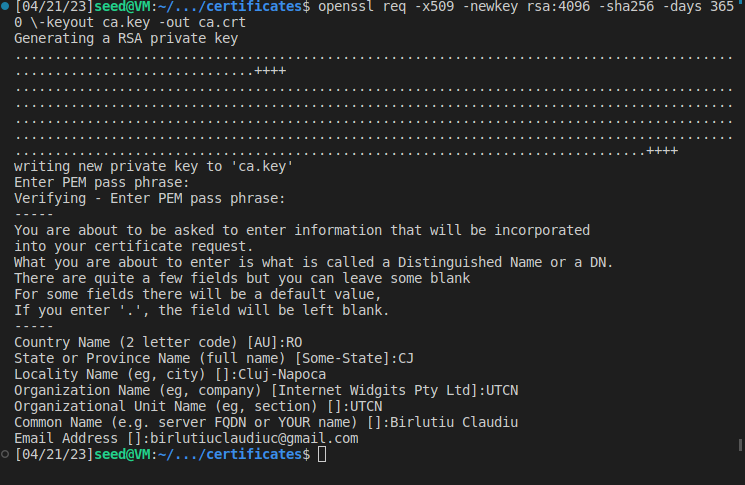
## Sarcina 1: Deveniți o autoritate de certificat (CA)

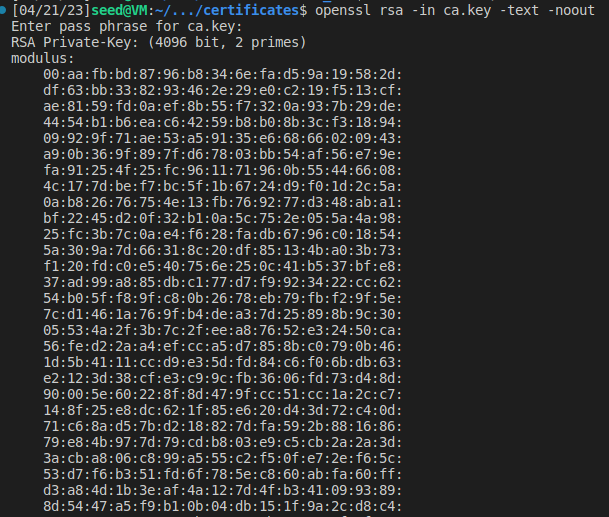
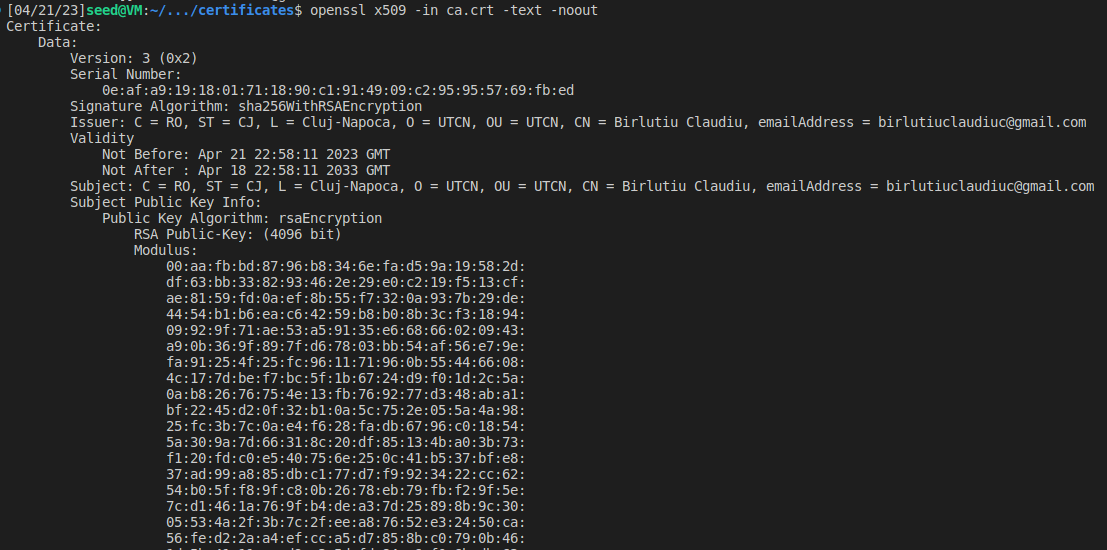
## În prima faza am creat containerul și am pornit acest serviciu



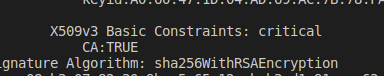
* Am adaugat în /etc/hosts cele 2 intrari



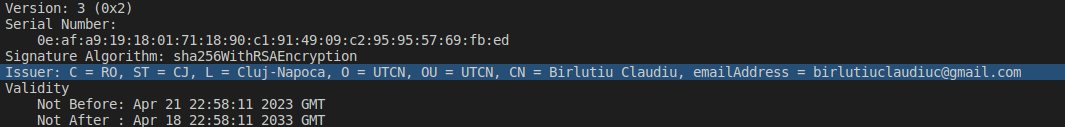
* am creat cele 2 fisiere mentionate
* am generat certificatul X.509 auto-semnat astfel:
* generăm o pereche de chei RSA cu o lungime de 4096 biți prin optiunea -newkey rsa:4096
* -sha256: specifică faptul că dorim să folosim algoritmul de hash SHA-256 pentru semnarea certificatului.
* Certificatul va fi valid 3650 de zile (10 ani aprox)
* am vizualizat continutul decodat al fisierelor



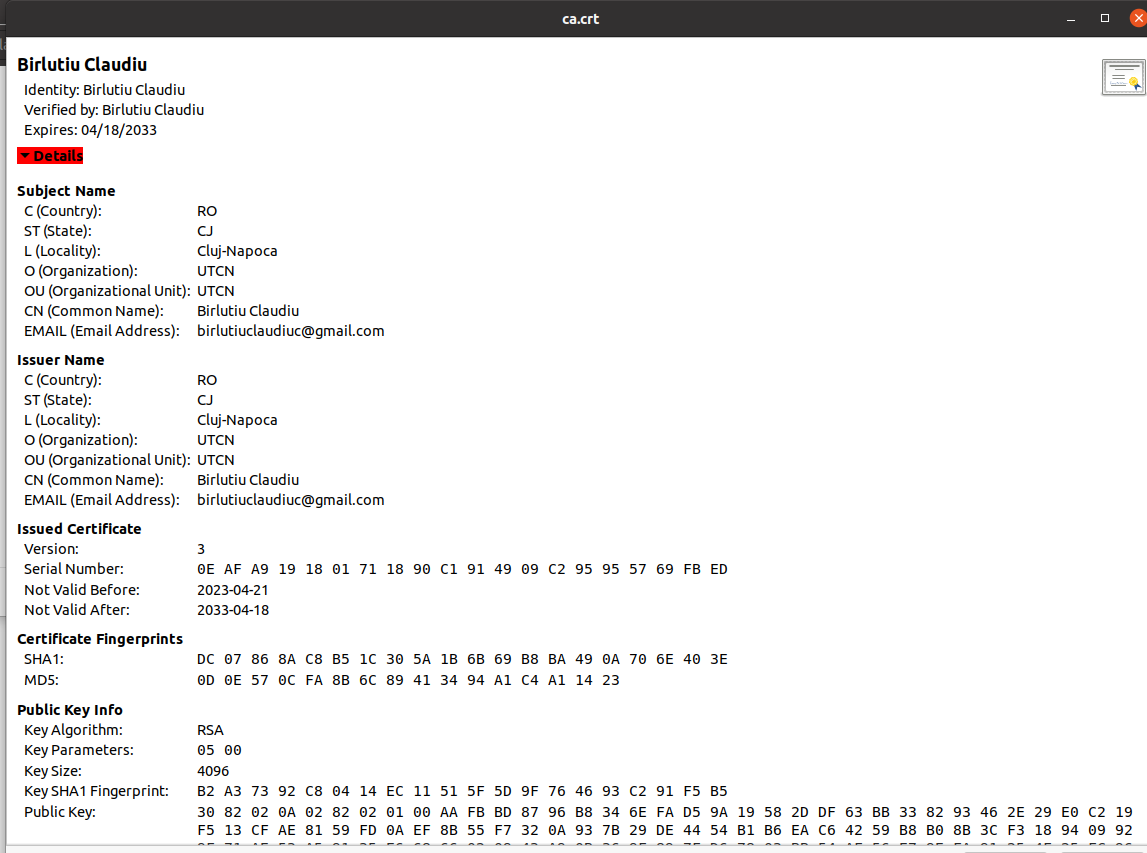
* Ce parte a certificatului indica ca acesta este un certificat CA?
  + există o extensie numită "Basic Constraints" care indică dacă certificatul este un certificat de autoritate de certificare (CA) sau nu.



* Ce parte a certificatului indica ca acesta este un certificat semnat de sine?
  + există un câmp numit "Issuer" care indică entitatea care a emis certificatul.



* In algoritmul RSA, avem un exponent public **e,** un exponent privat **d** , un modul **n** si doua secrete, numerele **p** și **q,** astfel ıncat **n=pq.** Va rugam sa identificati valorile pentru aceste elemente in certificatul dvs. si fisierele cheie
  + Exponent: 65537 (0x10001)
  + publicExponent: 65537 (0x10001)
  + celelalte valori se pot observa din : **openssl rsa -in ca.key -text -noout**
* certificatul



## Sarcina 2: Generarea unei cereri de certificat pentru serverul dvs.

## Sarcina 3: Generarea unui certificat pentru serverul dvs.

## Sarcina 4: Plasarea unui certificat într-un sit de web HTTPS bazat pe Apache

## Sarcina 5: Lansarea unui atac de tipul om-la-mijloc

## Sarcina 7: Lansarea unui atac de tipul om-la-mijloc cu o CA compromisă