Izvrsno! Prešli ste na drugi tip zadatka, koji se bavi digitalnim sertifikatima i autoritetima za sertifikaciju (CA).

Evo detaljnog pojašnjenja zadatka, ključnih pojmova i analize skripte koja ga rješava.

📝 Pojašnjenje Zadatka i Ključni Pojmovi

1. Cilj Zadatka

Zadatak 3 (sa **prve slike**) je jednostavan: **Utvrditi koji od datih klijentskih sertifikata (clientcertX.crt) su potpisani od strane datog CA sertifikata (cacert.pem).**

2. Ključni Pojmovi

Pojam	Objašnjenje
Sertifikat (Certificate)	Digitalni dokument koji dokazuje vlasništvo nad javnim ključem. Sadrži javni ključ subjekta (npr. klijenta), identifikacione podatke i **digitalni potpis** izdat od strane CA. U ovom slučaju, to su datoteke clientcertX.crt.
CA Sertifikat (CA Certificate)	Sertifikat koji pripada **Autoritetu za Sertifikaciju** (Certificate Authority). CA je pouzdani entitet koji izdaje i potpisuje digitalne sertifikate. U ovom slučaju, to je datoteka **cacert.pem**.
Potpisivanje	Proces kojim CA koristi svoj **privatni ključ** da digitalno potpiše novi sertifikat (u ovom slučaju, klijentski sertifikat). Ovaj potpis služi kao **dokaz autentičnosti** i garantuje da je klijentski sertifikat izdao pouzdani CA.
Verifikacija (Verify)	Proces provjere digitalnog potpisa na sertifikatu. Da bismo provjerili da li je klijentski sertifikat potpisao određeni CA, koristimo **javni ključ CA** (koji se nalazi u CA sertifikatu, cacert.pem). Ako se potpis poklapa, sertifikat je validan, tj. potpisao ga je taj CA.

Zaključak: Morate provjeriti (verifikovati) svaki klijentski sertifikat pojedinačno, koristeći cacert.pem kao referentni CA.

💻 Pojašnjenje Skripte (skripta.sh)

Skripta (sa **druge slike**) koristi openssl komandu za automatizovanu provjeru svih klijentskih sertifikata.

Analiza Skripte Liniju po Liniju:

1. Definisanje Liste Sertifikata

```Bash
certs="client*"
```

- certs: Promjenljiva.
- "client*": Ovo je tzv. **wildcard** (džoker) koji, kada se koristi u for petlji u Bash-u, automatski ekspanduje na sva imena datoteka u trenutnom direktorijumu koja počinju sa client (npr. clientcert1.crt, clientcert2.crt, ..., clientcert30.crt, kao što se vidi na **trećoj slici**).

2. Petlja za Iteraciju

```Bash
for cert in $certs
do
  # ... komanda
done
```

- Petlja će se pokrenuti za svaki pronađeni fajl koji odgovara obrascu client*.
- U svakoj iteraciji, varijabla cert će sadržati ime datoteke (npr. clientcert1.crt).

3. Ključna Komanda za Verifikaciju

```Bash
  openssl verify -CAfile cacert.pem -verbose $cert 2>error.txt
```

Ovo je centralni dio skripte:

- **openssl verify**: Komanda za verifikaciju digitalnih sertifikata.
- **-CAfile cacert.pem**: Parametar koji govori OpenSSL-u da koristi cacert.pem kao **korenski (root) sertifikat** ili **izdavaoca (issuer)**. Drugim riječima, provjeri da li je sertifikat $cert potpisao entitet čiji je sertifikat cacert.pem.
- **-verbose**: Ispisuje detaljnije informacije o procesu verifikacije, što olakšava potvrdu.
- **$cert**: Varijabla koja sadrži trenutni klijentski sertifikat koji se provjerava.
- **2>error.txt**: Standardni preusmjerivač grešaka.
  - 2: Standardni tok greške (stderr).
  - >: Preusmjeri.
  - error.txt: Ime datoteke.
  - **Svrha:** Preusmjerava sve **poruke o grešci** koje generiše openssl verify u datoteku error.txt.

3. Kako Skripta Rješava Zadatak

Skripta rješava zadatak na sljedeći način:

1. Automatski provjerava svaki klijentski sertifikat protiv cacert.pem.
2. **Uspjeh verifikacije** znači da je sertifikat **potpisan** od strane CA. openssl verify vraća poruku "OK" (npr. clientcert15.crt: OK).
3. **Neuspjeh verifikacije** znači da sertifikat **nije potpisan** od strane tog CA. openssl verify će vratiti poruku o grešci (koja se preusmjerava u error.txt).

Rezultat: Nakon izvršavanja skripte, sve što je potrebno je pogledati **standardni izlaz** (koji nije preusmjeren u error.txt) da bi se vidjelo koji sertifikati su vraćeni sa statusom **OK**.

Komentarisane linije (#clientcert15.crt: OK, #clientcert27.crt: OK, #clientcert29.crt: OK) jasno ukazuju da su **clientcert15.crt**, **clientcert27.crt** i **clientcert29.crt** jedini sertifikati potpisani datim CA sertifikatom.

☑ Sugestije

1. **Ispitajte Izlaz (Output):** Iako skripta preusmjerava greške, i dalje ispisuje uspješne rezultate na standardni izlaz (terminal). Pogledajte terminal nakon pokretanja skripte da vidite koji fajlovi su "OK".
2. **Proučite Greške:** Pregledajte sadržaj datoteke error.txt. U njoj će se nalaziti razlozi zašto preostalih 27 sertifikata nije uspjelo proći verifikaciju (npr. unable to get local issuer certificate).
3. **Bolja Skripta (Opcija):** Da biste dobili čistiji izlaz samo sa sertifikatima koji su "OK", mogli biste modifikovati skriptu da koristi grep:
   Bash

```
#!/bin/bash
certs="client*.crt" # Preciznije definisanje

echo "Sertifikati potpisani od strane CA:"

for cert in $certs
do
   # Privremeni fajl za izlaz, ukljucujuci standardni izlaz i gresku
   output=$(openssl verify -CAfile cacert.pem $cert 2>&1)

   # Proveri da li izlaz sadrzi "OK"
   if echo "$output" | grep -q ": OK"; then
      echo "$cert"
   fi
done
```

Da li želite da nastavite sa nekim drugim zadatkom ili da Vam pružim dodatne detalje o OpenSSL komandi verify?