- VM nutzen (Ubuntu)

- openssl x509 -in ca.crt -text -noout (Inhalt vom Zertifikat anzeigen)

- cat ca.key (Inhalt vom key anzeigen)

- openssl genrsa -out student.key 4096 (OpenSSL generate RSA key, in die Datei student.key mit der Länge 4096) (RSA-Variante)

- (cat student.key)

- geht auch mit elliptischen Kurven

- openssl ecparam -list\_curves (Liste von elliptischen Kurven)

- openssl ecparam -out student\_ec.key -name prime256v1 -genkey (prime256v1 ist eine elliptische Kurve, Schlüssel mit e. Kurve, viel Kürzer, dadurch angenehmer und beliebter)

- openssl req -new -key student.key -out student.csr -nodes -subj "/C=E/ST=RLP/L=Trier/O=Hochschule Trier/CN=942069 (CN = Matrikelnummer, Zertifikat anfordern, deswegen openssl req)

- openssl x509 -req -days 365 -in student.csr -CA ca.crt -CAkey ca.key -set\_serial 1234567 out student.crt (Zertifkat mit private key verheiraten/koppeln um tatsächlich Zertifikat zu erzeugen)

- openssl x509 -in student.crt -text -noout (Zertifikat anschauen und überprüfen | Zertifikat ausgestellt von der CA für einen Studenten)

- Zertifikat + privaten schlüssel zusammenpacken, damit Adobe es annimmt

- openssl pkcs12 -export -in student.crt -inkey student.key -out student.p12 -passout pass:123456 (im Prinzip umformatierung | pass = Passwort (mind. 6 stellen) | mit .p12 kann adobe was anfangen)

- p12 datei exportieren aus vm (seafile.rlp.net in ubuntu hochladen, windows runterladen)

- Adobe

- Einstellungen

- Unterschriften

- Identitäten / Einstellung für digitale IDs und vertrauenswürdige Zertifikate -> digitale IDs um selbst einzubinden

- p12 datei angeben und anschließend passwort angeben

- Zertifikat nun importiert

- CA importieren

- ca.crt importieren damit Adobe auch der CA vertraut

- ITS CA 2022-2023

- Stift

- Dieses Zertifikat als vertrauenswürdigen Stamm anerkennen

- zertifikate von diesem Stamm sind vertrauenwürdig