Name: Enid Shilwatso

Student ID: 671055

Lab4Instructor: Professor Dennis Kaburu

Assignment: Lab 21.4.7 - Certificates Authority Stores

Title: Advanced Information System Security

Course: ISC6120

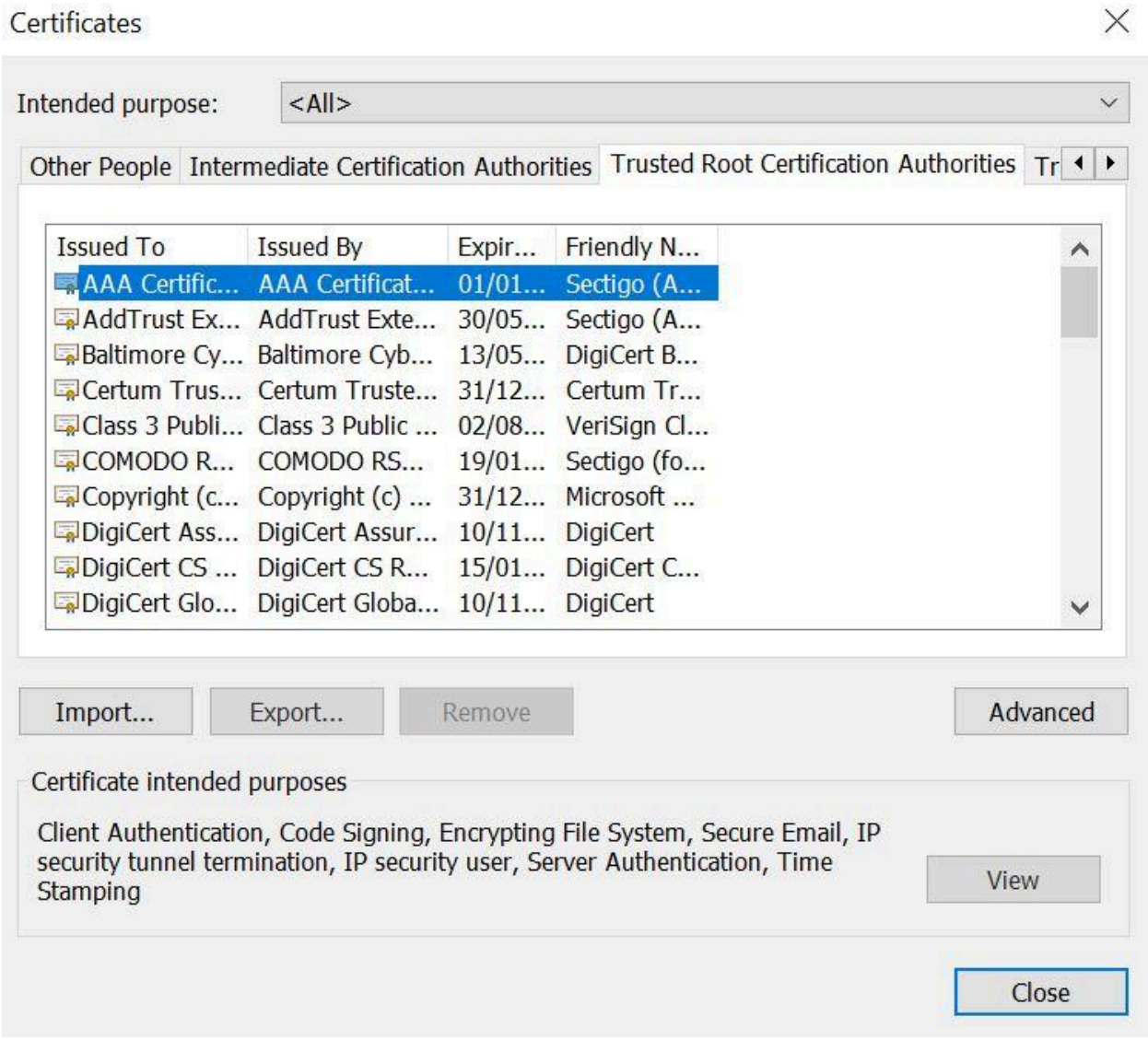
Due: Summer 2025

Part 1: Certificates Trusted by Your Browser

HTTPS relies on a third-party entity for validation. Known as Certification Authority (CA), this third-party entity verifies if a domain name really belongs to the organization claiming its ownership. If the verification checks, the CA creates a digitally signed certificate containing an information about the organization, including its public key. The entire system is based on the fact that web browsers and operating systems ship with a list of CAs they trust. Any certificates signed by any of the CAs in the list will be seen by the browser as legitimate and be automatically trusted. To make the system more secure and more scalable, CAs often spread the task of creating and signing certificates among many child CAs. The parent CA is known as the Root CA. If a browser trusts a Root CA, it also trusts all of its children CAs

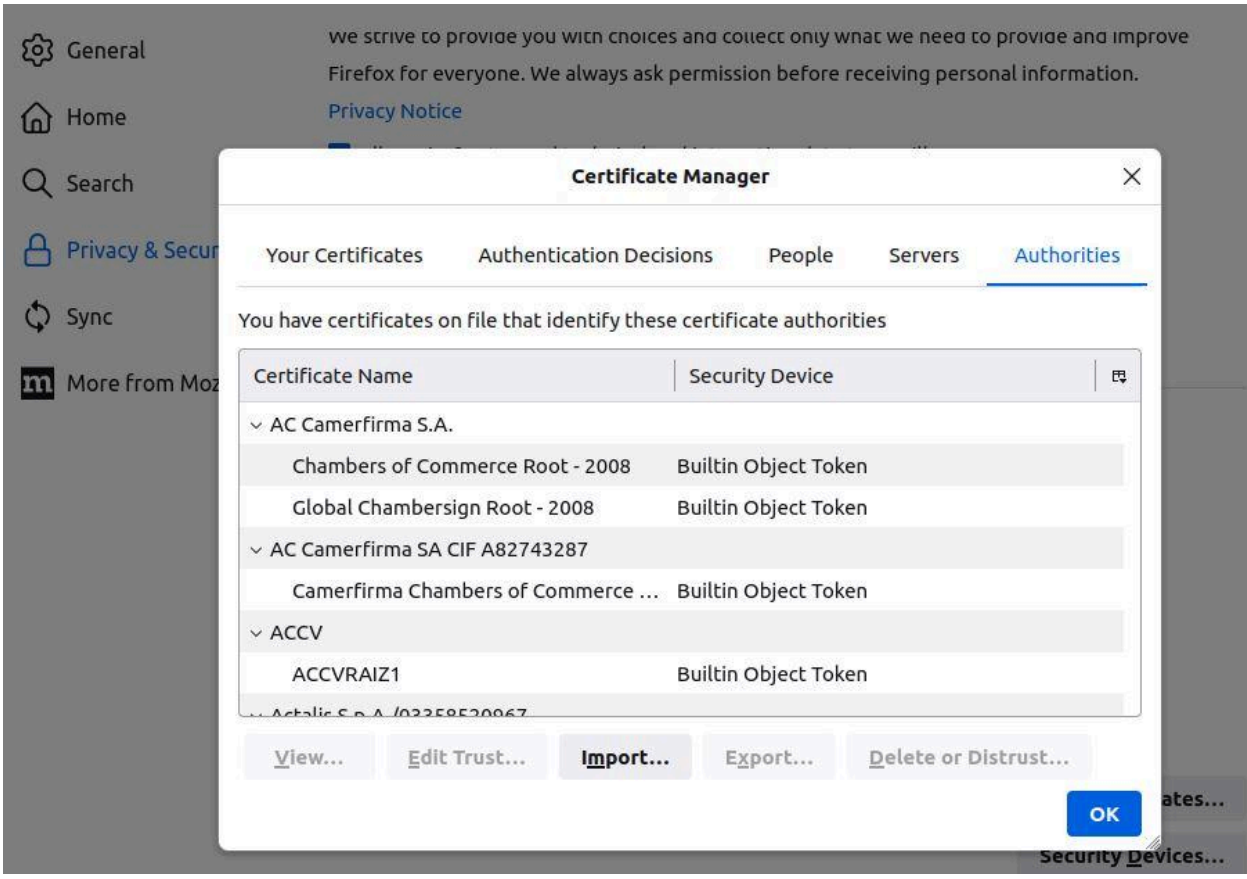
This lab focuses on Chrome 81 and Firefox 75. The menu and graphics may be different for other versions of the web browser.

a. Display the Root Certificates in Chrome.
    - Open the Chrome web browser on your PC
    - Click the three-dot icon on the far right of the address bar to display Chrome's options. Click Settings
    - Scroll down to Privacy and security and click More.
    - Scroll down and select Manage certificates
    - In the Certificate window, select Trusted Root Certification Authorities tab to show all certificates and certificate authorities trusted by Chrome.



b. Display the Certificates in the CA Store in Firefox.

- Open Firefox and click the Menu icon. The Menu icon is located on the far right of the Firefox window, next to the address bar. Click Preferences.
- Click Privacy & Security in the left panel
- Scroll down to the Security section and click View Certificates.
- A window opens that shows the certificates and certification authorities trusted by Firefox.



Part 2: Checking for Man-In-Middle

a. Gathering the correct and unmodified certificate fingerprint.
   - The first step is to gather a few site fingerprints

b. Gather the certificate fingerprint in use by the CyberOps Workstation VM.

   -Use the three piped commands below to fetch the fingerprint for Cisco.com. The line below uses OpenSSL to connect to cisco.com on port 443 (HTTPS), request the certificate and store it on a text file named cisco.pem. The output is also shown for context.

   [analyst@secOps ~]$ echo -n | openssl s_client -connect cisco.com:443 | sed -ne '/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p' > ./cisco.pem

```
[analyst@secOps ~]$ echo -n | openssl s_client -connect cisco.com:443 | sed -ne
'/-BEGIN CERTIFICATE-/,/-END CERTIFICATE-/p' > ./cisco.pem
depth=2 C = US, O = IdenTrust, CN = IdenTrust Commercial Root CA 1
verify return:1
depth=1 C = US, O = IdenTrust, OU = HydrantID Trusted Certificate Service, CN =
HydrantID Server CA O1
verify return:1
depth=0 C = US, ST = California, L = San Jose, O = Cisco Systems Inc., CN = www.
cisco.com
verify return:1
DONE
```

c. Optionally, use the cat command to list the contents of the fetched certificate and stored in the cisco.pem text file

[analyst@secOps ~]$ cat cisco.pem

```
[analyst@secOps ~]$  cat cisco.pem
-----BEGIN CERTIFICATE-----
MIIHkDCCBnigAwIBAgIQQAGR0fPH7E6nOzAb4+BqkDANBgkqhkiG9w0BAQsFADBy
MQswCQYDVQQGEwJVUzESMBAGA1UEChMJSWRlblRydXN0MS4wLAYDVQQLEyVIeWRy
YW50SUQgVHJ1c3RlZCBDZXJ0aWZpY2F0ZSBTZXJ2aWNlMR8wHQYDVQQDExZIeWRy
YW50SUQgU2VydmVyIENBIE8xMB4XDTI0MDkwODE0MDUwMFoXDTI1MDkwODE0MDQw
MFowajELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNhbGlmb3JuaWExETAPBgNVBAcT
CFNhbiBKb3NlMRswGQYDVQQKExJDaXNjbyBTeXN0ZW1zIEluYy4xFjAUBgNVBAMT
DXd3dy5jaXNjby5jb20wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC/
wU5OuamFtpNuSRH1xFjisnS+beEVUs/zhJ+cJRFu/iu485ijVJH+yNc0CdoZRRsA
CzhMiwKME0cfkPa3OCBQzY55b4HPn+pIsDzKCBdIe8RyB/ecvhuLhfBKpfjnmy2z
EJ9CqvOGl8m9nN791lvIE/qgPx6VXQnE7aSikH1hRtlSsL3uMxXyju2VvooKt1CK
A7JsBCgFdchLprUJPqYXByuh9koK7K0beoAhjGyFjgPmm9XN1REgJVArVfMZGNUu
37txLGcLOsFAmMoSR44A8VBrMDy+Xz3djxyvH5lUUHzoENTSTztuWeB7THr5nECb
FnufAIvQ2LmLvrGkWS3tAgMBAAGjggQoMIIEJDAOBgNVHQ8BAf8EBAMCBaAwgYUG
CCsGAQUFBwEBBHkwdzAwBggrBgEFBQcwAYYkaHR0cDovL2NvbW1lcmNpYWwub2Nz
cC5pZGVudHJ1c3QuY29tMEMGCCsGAQUFBzAChjdodHRwOi8vdmFsaWRhdGlvbi5p
ZGVudHJ1c3QuY29tL2NlcnRzL2h5ZHJhbnRpZGNhTzEucDdjMB8GA1UdIwQYMBaA
FIm4m7ae7fuwxr0N7GdOPKOSnS35MCEGA1UdIAQaMBgwCAYGZ4EMAQICMAwGCmCG
SAGG+S8ABgMwRgYDVR0fBD8wPTA7oDmgN4Y1aHR0cDovL3ZhbGlYXRpb24uaWRl
bnRydXN0LmNvbS9jcmwvaHlkcmFudGlkY2FvMS5jcmwwggE9BgNVHREEggE0MIIB
MIIJY2lzY28uY29tgg13d3cuY2lzY28uY29tgg53d3cxLmNpc2NvLmNvbYIOd3d3
Mi5jaXNjby5jb22CDnd3dzMuY2lzY28uY29tghB3d3ctMDEuY2lzY28uY29tghB3
d3ctMDIuY2lzY28uY29tghF3d3ctcnRwLmNpc2NvLmNvbYISd3d3MS1zczIuY2lz
```

d. Now that the certificate is saved in the cisco.pem text file, use the command below to extract and display its fingerprint

[analyst@secOps ~]$ openssl x509 -noout -in cisco.pem -fingerprint -sha1

```
[analyst@secOps ~]$ openssl x509 -noout -in cisco.pem -fingerprint -sha1
sha1 Fingerprint=37:0D:3C:46:33:24:CA:54:21:6C:03:0D:27:CD:AA:A8:C8:8E:9F:8C
[analyst@secOps ~]$
```

e. Compare the Fingerprints
    - They do not match this is because the organisation probably renew there certificates.