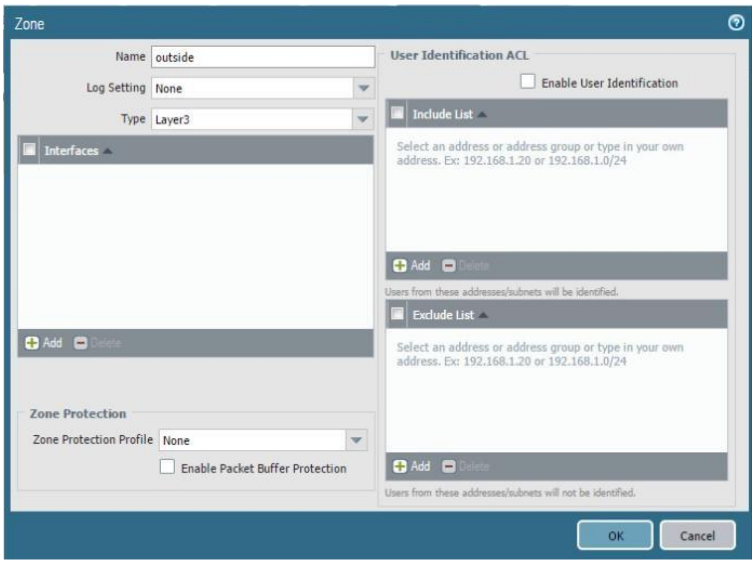
ABHIJIT GUPTA
ESSENTIAL PROJECT 1

Module 1A (LAB 1): Creating a Zero Trust Environment

Summary: In this module, we created a zero day trust environment by creating zones, applying security policies and after that we tested it. We created 3 zones inside, outside & DMZ for different purposes. We also created NAT Policies for packets. After committing all the changes, we tested it by visiting a webpage and we can see all the traffic in Firewall Logs.



Zone

Name: outside

Log Setting: None

Type: Layer3

User Identification ACL

□ Enable User Identification

Include List

Select an address or address group or type in your own address. Ex: 192.168.1.20 or 192.168.1.0/24

Users from these addresses/subnets will be identified.

Exclude List

Select an address or address group or type in your own address. Ex: 192.168.1.20 or 192.168.1.0/24

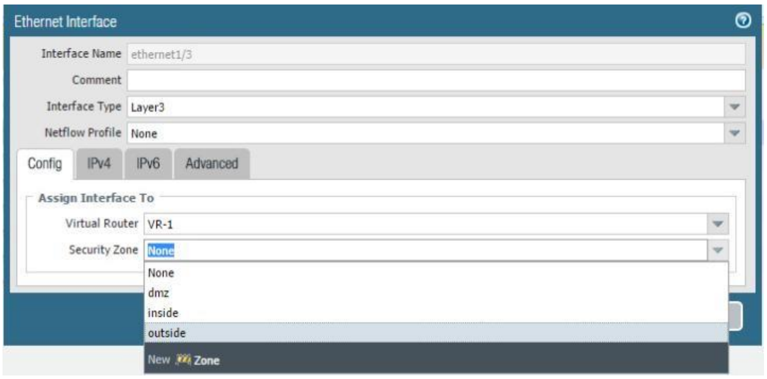
Users from these addresses/subnets will not be identified.

Zone Protection

Zone Protection Profile: None

□ Enable Packet Buffer Protection

- Creating Zones



Ethernet Interface

Interface Name: ethernet1/3

Comment:

Interface Type: Layer3

Netflow Profile: None

Config | IPv4 | IPv6 | Advanced

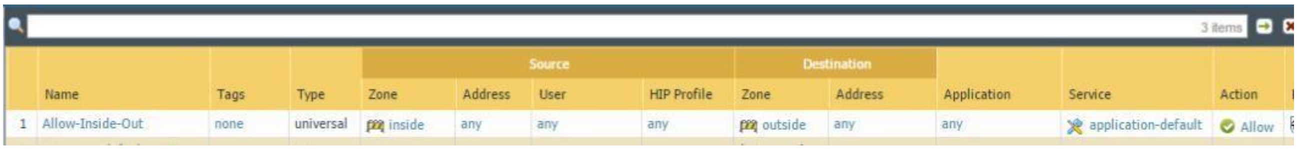
Assign Interface To

Virtual Router: VR-1

Security Zone: None

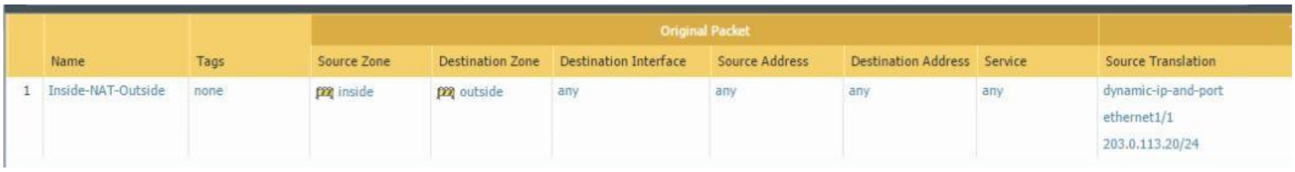
None
dmz
inside
outside

- Configuring Ethernet Interfaces



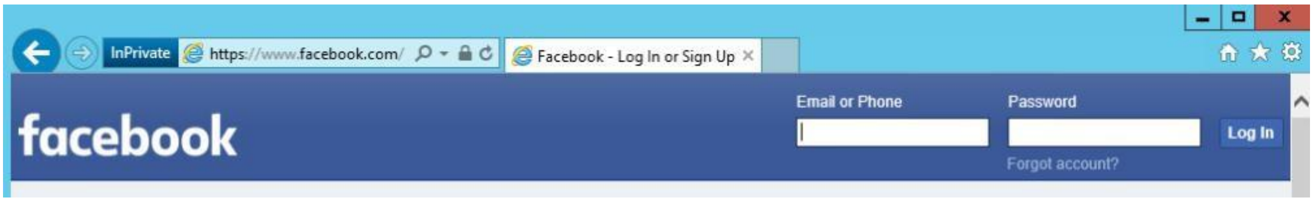
	Name	Tags	Type	Source				Destination		Application	Service	Action
				Zone	Address	User	HIP Profile	Zone	Address			
1	Allow-Inside-Out	none	universal	inside	any	any	any	outside	any	any	application-default	Allow

- Creation of Rules

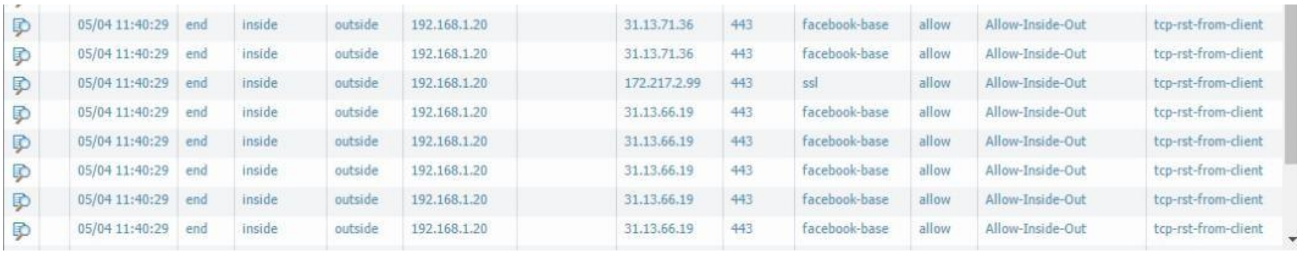


			Original Packet						
	Name	Tags	Source Zone	Destination Zone	Destination Interface	Source Address	Destination Address	Service	Source Translation
1	Inside-NAT-Outside	none	inside	outside	any	any	any	any	dynamic-ip-and-port ethernet1/1 203.0.113.20/24

- NAT Configurations



- Testing Process (Visiting a Webpage)

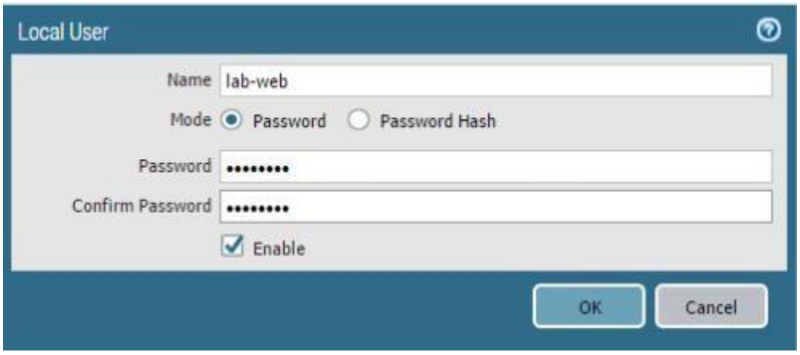


05/04 11:40:29	end	inside	outside	192.168.1.20		31.13.71.36	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20		31.13.71.36	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20		172.217.2.99	443	ssl	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20		31.13.66.19	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20		31.13.66.19	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20		31.13.66.19	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20		31.13.66.19	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20		31.13.66.19	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client

- We can see detailed information about traffic in Logs.

Module 1B (LAB 2): Configuring Authentication

Summary: In this module, we implemented a captive portal gateway for accessing web services in Palo Alto Firewall. We also created local user authentication for security and monitoring. We also analyzed logs of the user.



Local User

Name: lab-web

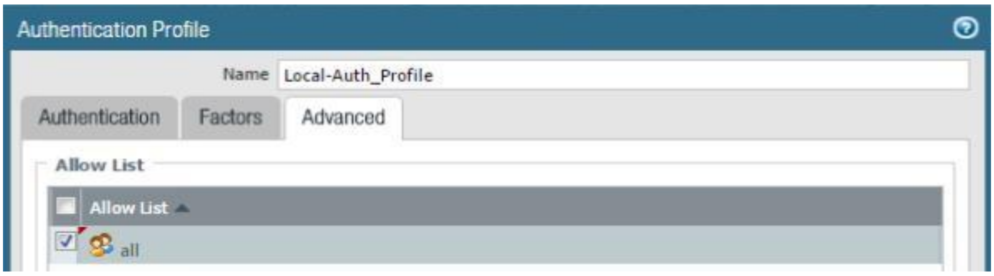
Mode: ● Password ○ Password Hash

Password: ••••••••

Confirm Password: ••••••••

☑ Enable

- ◆ Creating a Local User Account



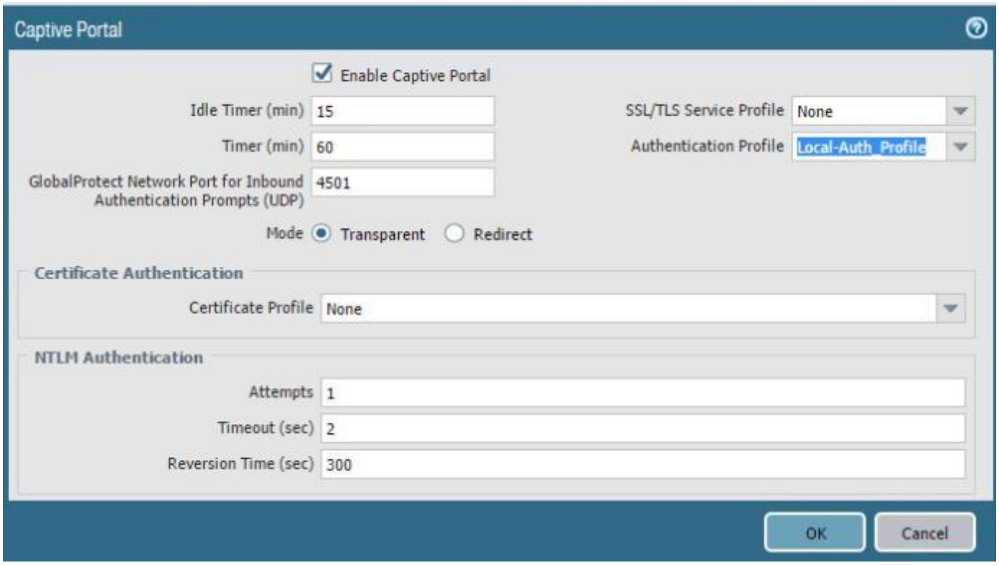
Authentication Profile

Name: Local-Auth_Profile

Allow List

□ Allow List
☑ all

- ◆ Creating an authentication profile



Captive Portal

☑ Enable Captive Portal

Idle Timer (min): 15

Timer (min): 60

GlobalProtect Network Port for Inbound Authentication Prompts (UDP): 4501

Mode: ● Transparent ○ Redirect

SSL/TLS Service Profile: None

Authentication Profile: Local-Auth_Profile

Certificate Authentication

Certificate Profile: None

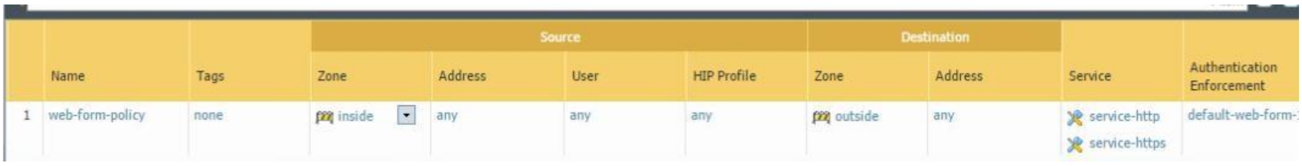
NTLM Authentication

Attempts: 1

Timeout (sec): 2

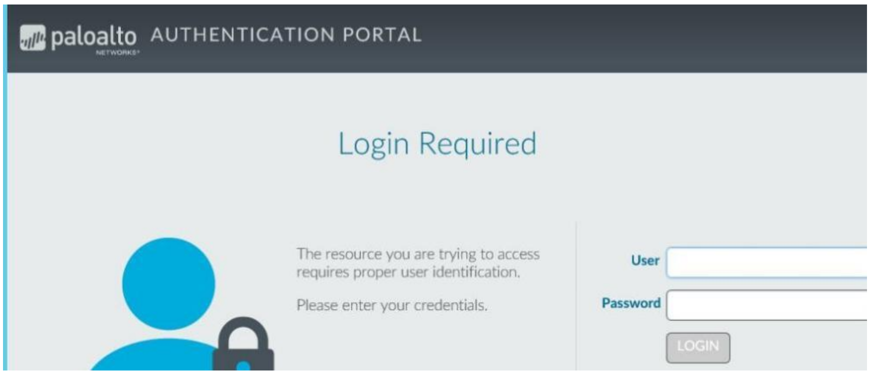
Reversion Time (sec): 300

- ◆ Captive Portal Configurations



			Source				Destination			
	Name	Tags	Zone	Address	User	HIP Profile	Zone	Address	Service	Authentication Enforcement
1	web-form-policy	none	inside	any	any	any	outside	any	service-http service-https	default-web-form-

- Firewall Rules Configurations



paloalto AUTHENTICATION PORTAL

Login Required

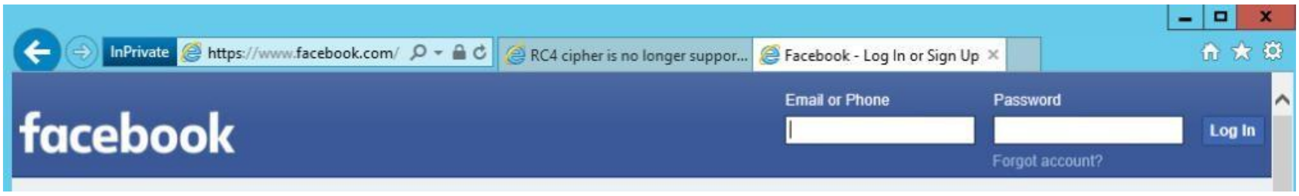
The resource you are trying to access requires proper user identification.

Please enter your credentials.

User

Password

- Our Captive Portal asking for User Information



InPrivate https://www.facebook.com/

RC4 cipher is no longer suppor...

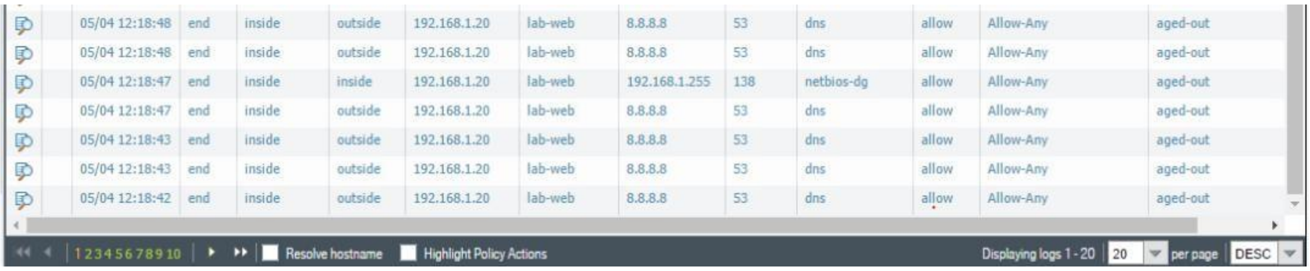
Facebook - Log In or Sign Up

facebook

Email or Phone

Password

- After Successful login user can access websites

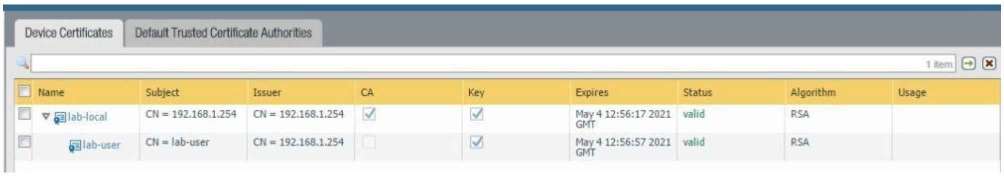


05/04 12:18:48	end	inside	outside	192.168.1.20	lab-web	8.8.8.8	53	dns	allow	Allow-Any	aged-out
05/04 12:18:48	end	inside	outside	192.168.1.20	lab-web	8.8.8.8	53	dns	allow	Allow-Any	aged-out
05/04 12:18:47	end	inside	inside	192.168.1.20	lab-web	192.168.1.255	138	netbios-dg	allow	Allow-Any	aged-out
05/04 12:18:47	end	inside	outside	192.168.1.20	lab-web	8.8.8.8	53	dns	allow	Allow-Any	aged-out
05/04 12:18:43	end	inside	outside	192.168.1.20	lab-web	8.8.8.8	53	dns	allow	Allow-Any	aged-out
05/04 12:18:43	end	inside	outside	192.168.1.20	lab-web	8.8.8.8	53	dns	allow	Allow-Any	aged-out
05/04 12:18:42	end	inside	outside	192.168.1.20	lab-web	8.8.8.8	53	dns	allow	Allow-Any	aged-out

- We can monitor user’ s activity in the logs

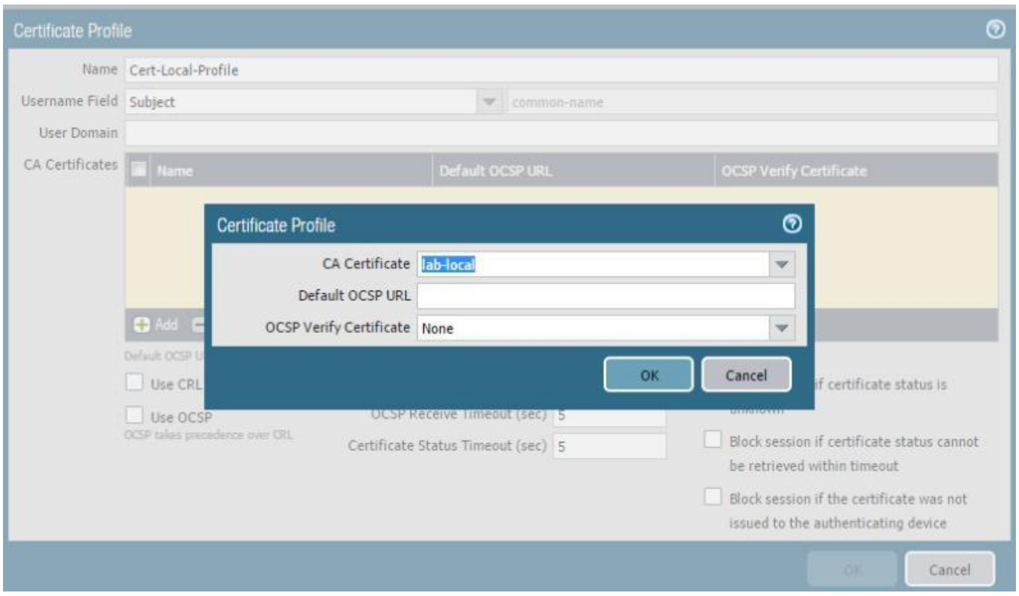
Module 2A (LAB 3): Using 2 Factor Authentication to secure the Firewall

Summary: In this module, we enabled 2 Factor authentication for Palo Alto Firewall using a digital certificate.



□	Name	Subject	Issuer	CA	Key	Expires	Status	Algorithm	Usage
□	▽ lab-local	CN = 192.168.1.254	CN = 192.168.1.254	☑	☑	May 4 12:56:17 2021 GMT	valid	RSA	
□	lab-user	CN = lab-user	CN = 192.168.1.254	□	☑	May 4 12:56:57 2021 GMT	valid	RSA	

- ◆ Creating User account for digital certificate.



Name: Cert-Local-Profile

Username Field: Subject — common-name

User Domain:

Certificate Profile

CA Certificate: lab-local

Default OCSP URL:

OCSP Verify Certificate: None

□ Use CRL

□ Use OCSP

OCSP takes precedence over CRL

OCSP Receive Timeout (sec): 5

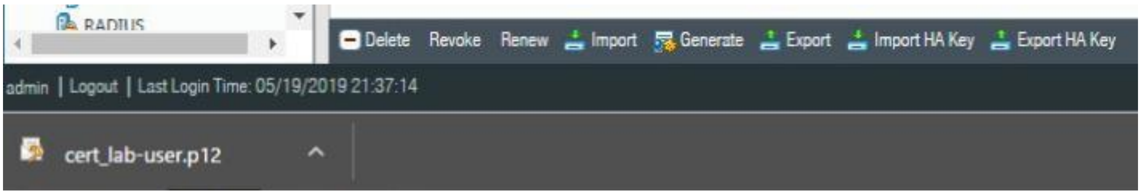
Certificate Status Timeout (sec): 5

if certificate status is unknown

□ Block session if certificate status cannot be retrieved within timeout

□ Block session if the certificate was not issued to the authenticating device

- ◆ Creating Certificate Profile



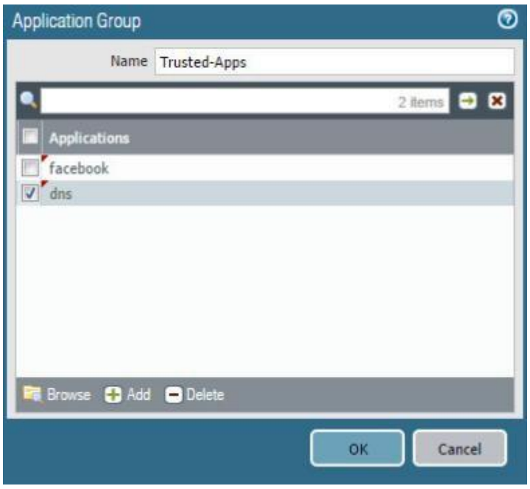
admin | Logout | Last Login Time: 05/19/2019 21:37:14

cert_lab-user.p12

- ◆ Downloading Certificate for the User

Module 2B (LAB 4): Allowing only Trusted Applications

Summary: In this module, we configured our firewall to allow traffic from and to selected apps. First we made an application group and added the websites to it. Then we configured security policy rule for the same. Then we tested it and analyzed the results.

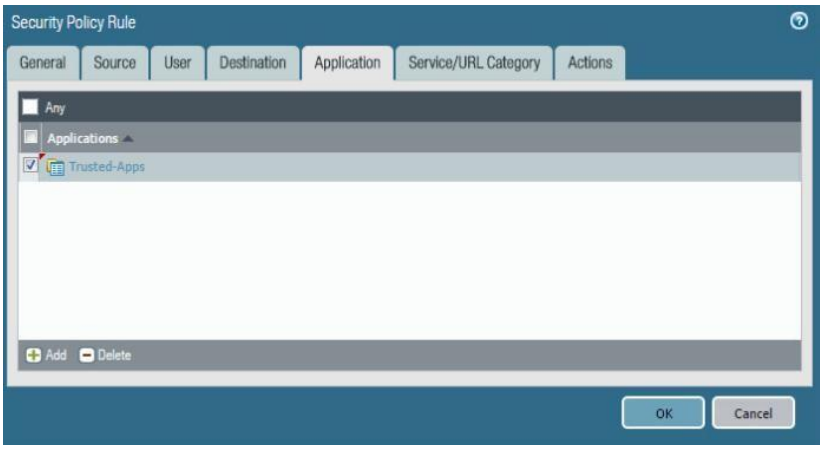


Name: Trusted-Apps

2 items

Applications
□ facebook
☑ dns

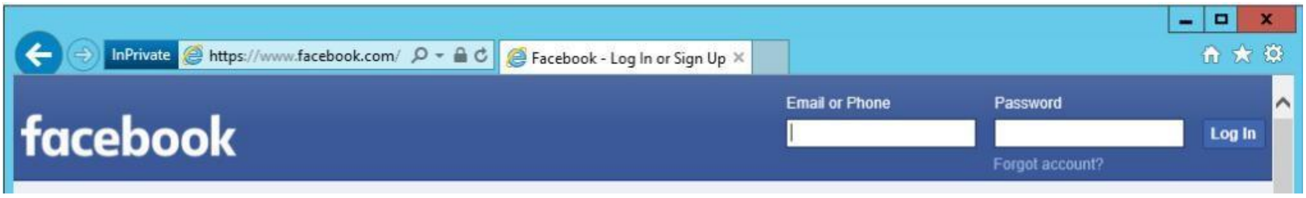
- Application Group Configurations



□ Any

Applications
☑ Trusted-Apps

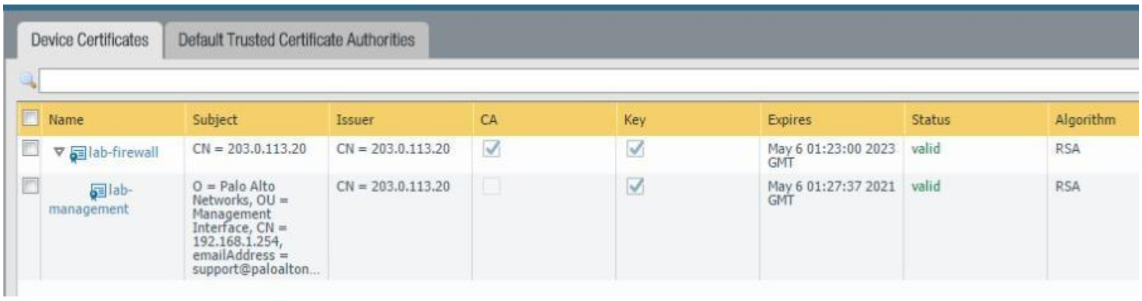
- Security Policy Rule Configurations



- Testing the rules and configurations

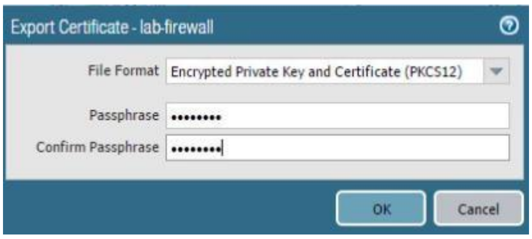
Module 3A (LAB 5): Managing Certificates

Summary: In this module, we generated a digital certificate for inbound management traffic. After that we exported tested and verified it.



□	Name	Subject	Issuer	CA	Key	Expires	Status	Algorithm
□	lab-firewall	CN = 203.0.113.20	CN = 203.0.113.20	☑	☑	May 6 01:23:00 2023 GMT	valid	RSA
□	lab-management	O = Palo Alto Networks, OU = Management Interface, CN = 192.168.1.254, emailAddress = support@paloalton...	CN = 203.0.113.20	□	☑	May 6 01:27:37 2021 GMT	valid	RSA

- ◆ Certificate Generation



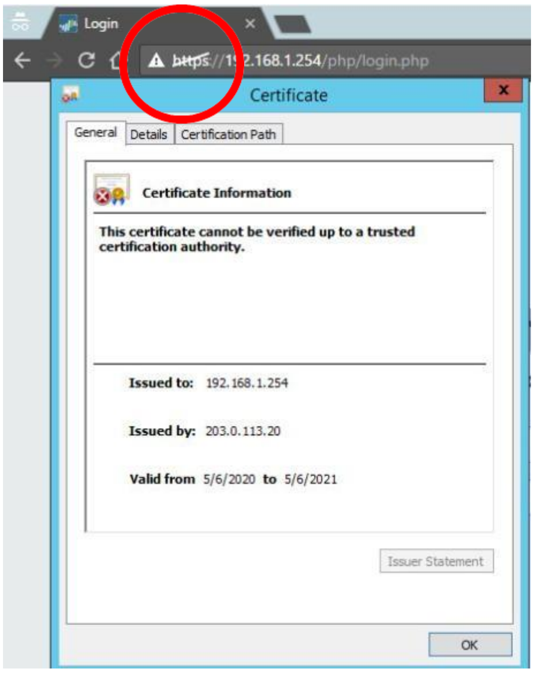
Export Certificate - lab-firewall

File Format: Encrypted Private Key and Certificate (PKCS12)

Passphrase: ••••••••

Confirm Passphrase: ••••••••

- ◆ Exporting Certificate



Certificate Information

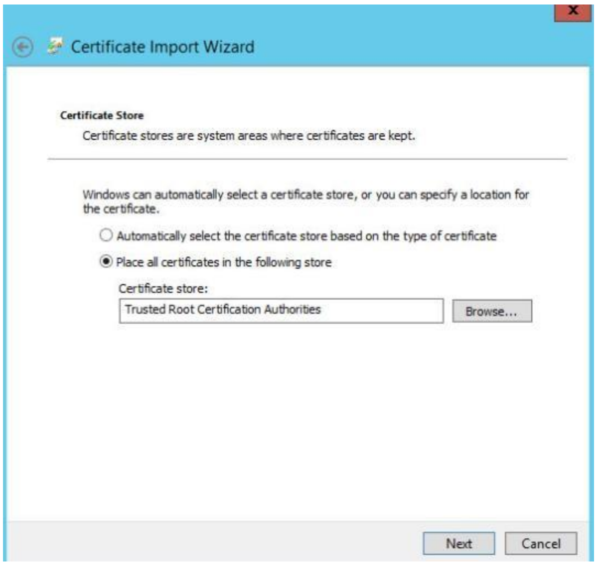
This certificate cannot be verified up to a trusted certification authority.

Issued to: 192.168.1.254

Issued by: 203.0.113.20

Valid from 5/6/2020 **to** 5/6/2021

- ◆ Testing it and we can see that it is not verified.



Certificate Import Wizard

Certificate Store

Certificate stores are system areas where certificates are kept.

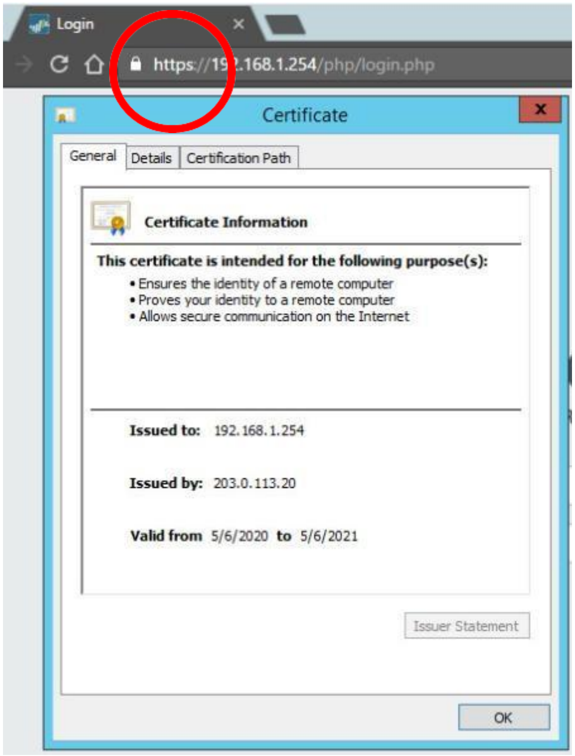
Windows can automatically select a certificate store, or you can specify a location for the certificate.

○ Automatically select the certificate store based on the type of certificate

● Place all certificates in the following store

Certificate store: Trusted Root Certification Authorities

- ◆ After downloading it locally, we imported it in the to certificates.msc utility.



Login

https://192.168.1.254/php/login.php

Certificate

Certificate Information

This certificate is intended for the following purpose(s):

- Ensures the identity of a remote computer
- Proves your identity to a remote computer
- Allows secure communication on the Internet

Issued to: 192.168.1.254

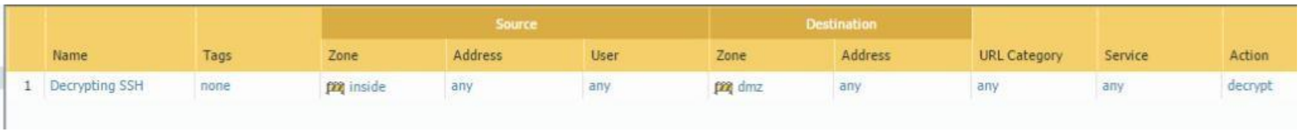
Issued by: 203.0.113.20

Valid from 5/6/2020 **to** 5/6/2021

- ◆ Now if we try to test it, it is verified now.

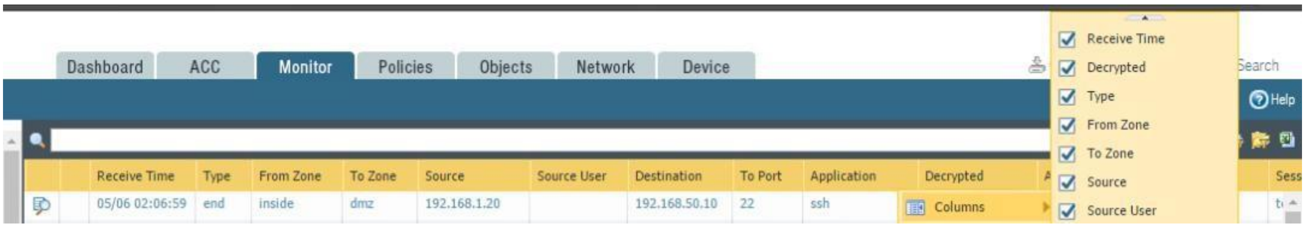
Module 3B (LAB 6 & 7): Decrypting SSH Traffic

Summary: In this module, we decrypted some SSH Packets. First, we created a Policy for decryption from inside zone to DMZ. Then we generated SSH Traffic towards DMZ. After that we decrypted it using the same policy.



			Source			Destination				
	Name	Tags	Zone	Address	User	Zone	Address	URL Category	Service	Action
1	Decrypting SSH	none	inside	any	any	dmz	any	any	any	decrypt

- ◆ Creation of the Policy



	Receive Time	Type	From Zone	To Zone	Source	Source User	Destination	To Port	Application	Decrypted	A	Sess
	05/06 02:06:59	end	inside	dmz	192.168.1.20		192.168.50.10	22	ssh			t

- ◆ Decrypting traffic in logs

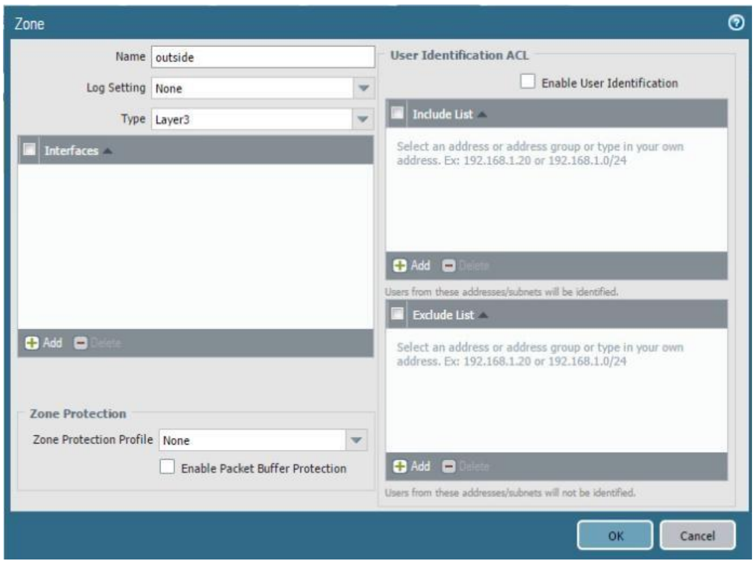
Project Introduction:

In this project, you will configure the firewall for a zero - trust environment.

Objectives:

1. Create zones and associate the zones to interfaces
2. Create a Security Policy Rule
3. Create a NAT Policy.

Screenshot:



Zone

Name: outside

Log Setting: None

Type: Layer3

User Identification ACL

□ Enable User Identification

Include List

Select an address or address group or type in your own address. Ex: 192.168.1.20 or 192.168.1.0/24

Users from these addresses/subnets will be identified.

Exclude List

Select an address or address group or type in your own address. Ex: 192.168.1.20 or 192.168.1.0/24

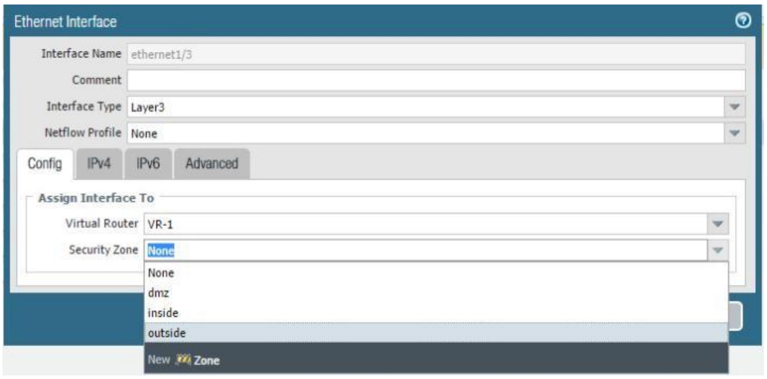
Users from these addresses/subnets will not be identified.

Zone Protection

Zone Protection Profile: None

□ Enable Packet Buffer Protection

- Creating Zones



Ethernet Interface

Interface Name: ethernet1/3

Comment:

Interface Type: Layer3

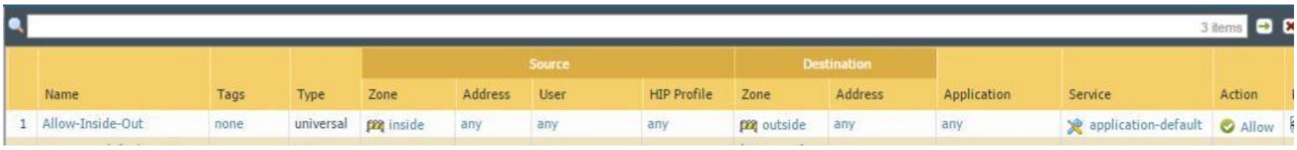
Netflow Profile: None

Assign Interface To

Virtual Router: VR-1

Security Zone: None

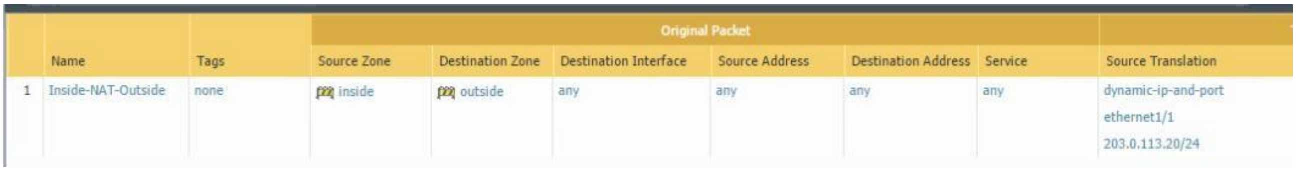
- Configuring Ethernet Interfaces



3 items

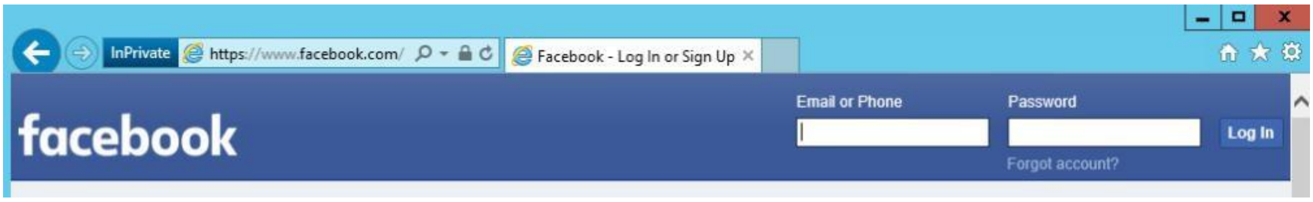
				Source				Destination				
	Name	Tags	Type	Zone	Address	User	HIP Profile	Zone	Address	Application	Service	Action
1	Allow-Inside-Out	none	universal	inside	any	any	any	outside	any	any	application-default	Allow

- ◆ Creation of Rules

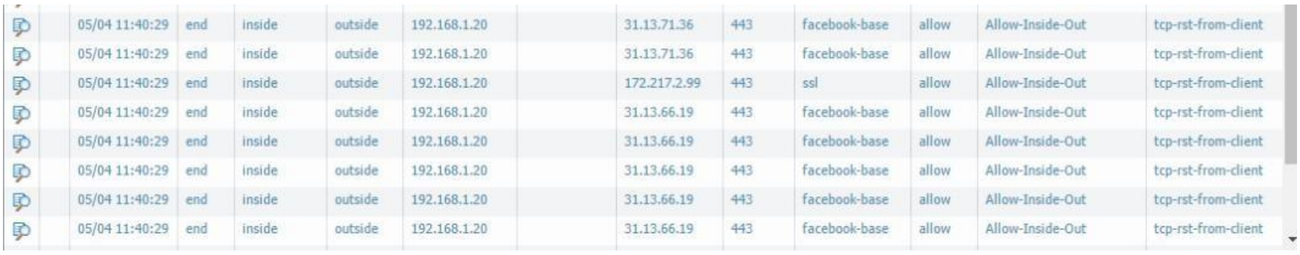


			Original Packet						
	Name	Tags	Source Zone	Destination Zone	Destination Interface	Source Address	Destination Address	Service	Source Translation
1	Inside-NAT-Outside	none	inside	outside	any	any	any	any	dynamic-ip-and-port ethernet1/1 203.0.113.20/24

- ◆ NAT Configurations



- ◆ Testing (Visiting Facebook)



05/04 11:40:29	end	inside	outside	192.168.1.20	31.13.71.36	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20	31.13.71.36	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20	172.217.2.99	443	ssl	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20	31.13.66.19	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20	31.13.66.19	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20	31.13.66.19	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20	31.13.66.19	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client
05/04 11:40:29	end	inside	outside	192.168.1.20	31.13.66.19	443	facebook-base	allow	Allow-Inside-Out	tcp-rst-from-client

- ◆ We can see detailed information about traffic in Logs.

THE END