ERIC SOMOGYI

Submit a file, named **<your last name>-<your first name>-HW2.zip** containing the followings:

Part #1 Complete the PKA HW exercise "Site-to-Site VPN in the zipped folder (80 points)

Include a **screenshot** of your PKA configuration below along with answers to the questions below in part#2, also Your PKA file should be submitted along with this document, your lab should have all the tasks completed as indicated in the work area also to get full credit your tests should be successfully verified prior submission.

Note: **You don't need to configure any routing on any routers in this lab (if you do, points will be deducted). All routes are configured for you via static default routes. You only need to focus on completing all the required tasks to build, configure,** and test the IPsec Tunnel.

Part#2 (20 points)

Answer the following questions:

1. Which action do IPsec peers take during the IKE Phase 2 exchange? **(5 points)**
2. If you are using Firewalls at each end of the IPSec VPN tunnel what are the protocols that need to be allowed for the tunnel to be established? **(5 points)**
3. When is a security association (SA) created if an IPsec VPN tunnel is used to connect between two sites? **(5 points)**
4. Which term describes a situation where VPN traffic that is received by an interface is routed back out of that same interface? **(5 points)**

Part#1: Include your screenshots here:

First I provided below, my entire command list I used to program the project. I have provided a brief synopsis of the order of the screenshots before the screenshots were pasted.

Router 0 - Chicago
#enable

```
#license boot module c1900 technology-package securityk9
#y
#reload
#enable
#conf t
#access-list 110 permit ip 192.168.11.0 0.0.0.255 10.10.10.0 0.0.0.255
#access-list 120 permit esp any any
#access-list 120 permit udp any any eq isakmp
#exit
#show access-lists
#conf t
#int s0/1/1
#ip access-group 120 in
#end
#
#crypto isakmp policy 20
#authentication pre-share
#encryption aes
#group 5
#hash md5
#lifetime 86400
#exit
#crypto isakmp enable
#show crypto isakmp policy
#
#conf t
#crypto ipsec transform-set CompanyABCD esp-aes esp-md5-hmac
#exit
#show crypto ipsec transform-set
#conf t
#access-list 110 permit
#crypto map ERICMAP 10 ipsec-isakmp
#match address 110
#set peer 63.100.202.229
#set pfs group1
#set transform-set CompanyABCD
#exit
# int s0/1/1
#crpyto map ERICMAP
#exit
#config t
#crypto isakmp key NET477 address 63.100.202.229
#exit
#show crypto map
#show crypto ipsec sa
#show crypto isakmp policy
```

```
#show crypto ipsec transform-set
#show crypto isakmp sa
#show ip int s0/1/1 (to show access group 120 on int)
```

Router 1 - Washington

```
#enable
#license boot module c1900 technology-package securityk9
#y
#reload
#enable
#conf t
#access-list 110 permit ip 10.10.10.0 0.0.0.255 192.168.11.0 0.0.0.255
#access-list 120 permit esp any any
#access-list 120 permit udp any any eq isakmp
#exit
#
#show access-lists
#conf t
#int s0/1/0
#ip access-group 120 in
#end
#
#crypto isakmp policy 20
#authentication pre-share
#encryption aes
#group 5
#hash md5
#lifetime 86400
#exit
#crypto isakmp enable
#show crypto isakmp policy
#conf t
#crypto ipsec transform-set CompanyABCD esp-aes esp-md5-hmac
#exit
#show crypto ipsec transform-set
#conf t
#access-list 110 permit
#crypto map ERICMAP 10 ipsec-isakmp
#match address 110
#set peer 63.100.202.225
#set pfs group1
#set transform-set CompanyABCD
#exit
#int s0/1/0
#crypto map ERICMAP
```

#exit
#config t
#crypto isakmp key NET477 address 63.100.202.225
#exit
#show crypto map
#show crypto ipsec sa
#show crypto isakmp policy
#show crypto ipsec transform-set
#show crypto isakmp sa
#show ip int s0/1/0 (to show access group 120 on int)

Installing sec license on both router's confirmation of command and success:

Router1

Physical Config CLI Attributes

IOS Command Line Interface

```
Washinton(config)#
Washinton(config)#
Washinton(config)#
Washinton(config)#
Washinton(config)#
Washinton(config)#
Washinton(config)#
Washinton(config)#
Washinton(config)#
Washinton(config)#license boot module c1900 technology-package
securityk9
PLEASE  READ THE  FOLLOWING TERMS  CAREFULLY. INSTALLING THE LICENSE
OR
LICENSE  KEY  PROVIDED FOR  ANY CISCO  PRODUCT  FEATURE  OR  USING
```

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
Chicago(config)#license boot module c1900 technology=package
securityk9
                                                  ^
% Invalid input detected at '^' marker.

Chicago(config)#license boot module c1900 technology-package
securityk9
PLEASE  READ THE  FOLLOWING TERMS  CAREFULLY. INSTALLING THE LICENSE
```

Crypto map configuration

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
Chicago>enable
Chicago#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Chicago(config)#crypto isakmp enable
Chicago(config)#crypto isakmp policy 20
Chicago(config-isakmp)#authentication pre-share
Chicago(config-isakmp)#encryption aes
Chicago(config-isakmp)#group 5
Chicago(config-isakmp)#hash md5
Chicago(config-isakmp)#lifetime 86400
Chicago(config-isakmp)#exit
Chicago(config)#show crypto isakmp policy
                ^
% Invalid input detected at '^' marker.

Chicago(config)#exit
Chicago#
%SYS-5-CONFIG_I: Configured from console by console

Chicago#show crypto isakmp policy

Global IKE policy
Protection suite of priority 20
        encryption algorithm:   AES - Advanced Encryption Standard
(128 bit keys).
        hash algorithm:         Message Digest 5
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #5 (1536 bit)
        lifetime:               86400 seconds, no volume limit
Chicago#
```

Router1

Physical Config CLI Attributes

IOS Command Line Interface

```
Washinton#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Washinton(config)#crypto isakmp enable
Washinton(config)#crypto isakmp policy 20
Washinton(config-isakmp)#authentication pre-share
Washinton(config-isakmp)#encryption aes
Washinton(config-isakmp)#group 5
Washinton(config-isakmp)#hash md5
Washinton(config-isakmp)#lifetime 86400
Washinton(config-isakmp)#exit
Washinton(config)#exit
Washinton#
%SYS-5-CONFIG_I: Configured from console by console

Washinton#show crypto isakmp policy

Global IKE policy
Protection suite of priority 20
        encryption algorithm:   AES - Advanced Encryption Standard
(128 bit keys).
        hash algorithm:         Message Digest 5
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #5 (1536 bit)
        lifetime:               86400 seconds, no volume limit
Washinton#
Washinton#
Washinton#
Washinton#
```

After figuring everything out and getting everything working with the commands I provided, below is the confirmation of the configurations followed by screenshots of the successful pings. The first **4** screenshots are of the Router 0 Chicago of showing the results from the following commands and the next **3** screenshots are of the Router 1 Washington router results. **In the screenshots with my #show crypto isakmp sa command, I have circled the VPN tunnel being IDLE active so it is easier for you to find.** The next **7** screenshots were of the VPN tunnel confirmation messages that started randomly after I pinged, and it automatically started to configure after using my command list. The last **2** screenshots are showing successful pings from each of the devices on each subnet from Chicago to Washington and vice versa.

#show crypto map
#show crypto ipsec sa
#show crypto isakmp policy
#show crypto ipsec transform-set
#show crypto isakmp sa
#show ip int s0/1/1 (to show access group 120 on int)

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
Chicago>enable
Chicago#show crypto map
Crypto Map ERICMAP 10 ipsec-isakmp
        Peer = 63.100.202.229
        Extended IP access list 110
            access-list 110 permit ip 192.168.11.0 0.0.0.255
10.10.10.0 0.0.0.255
        Current peer: 63.100.202.229
        Security association lifetime: 4608000 kilobytes/3600 seconds
        PFS (Y/N): Y
        Transform sets={
                CompanyABCD,
        }
        Interfaces using crypto map ERICMAP:
                Serial0/1/1

Chicago#show crypto ipsec sa

interface: Serial0/1/1
    Crypto map tag: ERICMAP, local addr 63.100.202.225

   protected vrf: (none)
   local  ident (addr/mask/prot/port):
(192.168.11.0/255.255.255.0/0/0)
   remote  ident (addr/mask/prot/port):
(10.10.10.0/255.255.255.0/0/0)
   current_peer 63.100.202.229 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 7, #pkts encrypt: 7, #pkts digest: 0
   #pkts decaps: 6, #pkts decrypt: 6, #pkts verify: 0
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #send errors 1, #recv errors 0
```

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #send errors 1, #recv errors 0

     local crypto endpt.: 63.100.202.225, remote crypto endpt.:
63.100.202.229
     path mtu 1500, ip mtu 1500, ip mtu idb Serial0/1/1
     current outbound spi: 0xF684FE1E(4135910942)

     inbound esp sas:
      spi: 0x8F52BB22(2404563746)
        transform: esp-aes esp-md5-hmac ,
        in use settings ={Tunnel, }
        conn id: 2002, flow_id: FPGA:1, crypto map: ERICMAP
        sa timing: remaining key lifetime (k/sec): (4525504/2523)
        IV size: 16 bytes
        replay detection support: N
        Status: ACTIVE

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0xF684FE1E(4135910942)
        transform: esp-aes esp-md5-hmac ,
        in use settings ={Tunnel, }
        conn id: 2003, flow_id: FPGA:1, crypto map: ERICMAP
        sa timing: remaining key lifetime (k/sec): (4525504/2523)
        IV size: 16 bytes
        replay detection support: N
        Status: ACTIVE

     outbound ah sas:

     outbound pcp sas:
```

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
        replay detection support: N
        Status: ACTIVE

     outbound ah sas:

     outbound pcp sas:

Chicago#
Chicago#
Chicago#show crypto isakmp policy

Global IKE policy
Protection suite of priority 20
        encryption algorithm:   AES - Advanced Encryption Standard
(128 bit keys).
        hash algorithm:         Message Digest 5
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #5 (1536 bit)
        lifetime:               86400 seconds, no volume limit
Chicago#show crypto ipsec transform-set
Transform set CompanyABCD: {    { esp-aes esp-sha-hmac  }
   will negotiate = { Tunnel,  },

Transform set #$!default_transform_set_1: { esp-aes esp-sha-hmac  }
   will negotiate = { Transport,  },
Transform set #$!default_transform_set_0: { esp-3des esp-sha-hmac  }
   will negotiate = { Transport,  },

Chicago#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id slot status
63.100.202.229  63.100.202.225  QM_IDLE           1041    0 ACTIVE


IPv6 Crypto ISAKMP SA
```

Copy Paste

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
  will negotiate = { Transport,  },

Chicago#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id slot status
63.100.202.229  63.100.202.225  QM_IDLE           1041    0 ACTIVE


IPv6 Crypto ISAKMP SA


Chicago#show ip int s0/1/1
Serial0/1/1 is up, line protocol is up (connected)
  Internet address is 63.100.202.225/30
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is not set
  Inbound  access list is 120
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
```

Router1

Physical Config CLI Attributes

IOS Command Line Interface

```
Washinton#show crypto map
Crypto Map ERICMAP 10 ipsec-isakmp
        Peer = 63.100.202.225
        Extended IP access list 110
            access-list 110 permit ip 10.10.10.0 0.0.0.255
192.168.11.0 0.0.0.255
        Current peer: 63.100.202.225
        Security association lifetime: 4608000 kilobytes/3600 seconds
        PFS (Y/N): Y
        Transform sets={
                CompanyABCD,
        }
        Interfaces using crypto map ERICMAP:
                Serial0/1/0

Washinton#show crypto ipsec sa

interface: Serial0/1/0
    Crypto map tag: ERICMAP, local addr 63.100.202.229

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (10.10.10.0/255.255.255.0/0/0)
   remote  ident (addr/mask/prot/port):
(192.168.11.0/255.255.255.0/0/0)
   current_peer 63.100.202.225 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 6, #pkts encrypt: 6, #pkts digest: 0
   #pkts decaps: 7, #pkts decrypt: 7, #pkts verify: 0
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #send errors 0, #recv errors 0

     local crypto endpt.: 63.100.202.229, remote crypto endpt.:
63.100.202.225
     path mtu 1500, ip mtu 1500, ip mtu idb Serial0/1/0
     current outbound spi: 0x8F52BB22(2404563746)

     inbound esp sas:
      spi: 0xF684FE1E(4135910942)
        transform: esp-aes esp-md5-hmac ,
        in use settings ={Tunnel, }
```

Router1

Physical Config CLI Attributes

IOS Command Line Interface

```
        IV size: 16 bytes
        replay detection support: N
        Status: ACTIVE

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0x8F52BB22(2404563746)
        transform: esp-aes esp-md5-hmac ,
        in use settings ={Tunnel, }
        conn id: 2003, flow_id: FPGA:1, crypto map: ERICMAP
        sa timing: remaining key lifetime (k/sec): (4525504/2175)
        IV size: 16 bytes
        replay detection support: N
        Status: ACTIVE

     outbound ah sas:

     outbound pcp sas:

Washinton#
Washinton#show crypto isakmp policy

Global IKE policy
Protection suite of priority 20
        encryption algorithm:   AES - Advanced Encryption Standard
(128 bit keys).
        hash algorithm:         Message Digest 5
        authentication method:  Pre-Shared Key
        Diffie-Hellman group:   #5 (1536 bit)
        lifetime:               86400 seconds, no volume limit
Washinton#show crypto ipsec transform-set
Transform set CompanyABCD: {    { esp-aes esp-sha-hmac  }
   will negotiate = { Tunnel,  },

Transform set #$!default_transform_set_1: { esp-aes esp-sha-hmac  }
   will negotiate = { Transport,  },
Transform set #$!default_transform_set_0: { esp-3des esp-sha-hmac  }
   will negotiate = { Transport,  },
```

Copy Paste

Router1

Physical Config CLI Attributes

IOS Command Line Interface

```
Washinton#show crypto ipsec transform-set
Transform set CompanyABCD: {    { esp-aes esp-sha-hmac  }
   will negotiate = { Tunnel,  },

Transform set #$!default_transform_set_1: { esp-aes esp-sha-hmac  }
   will negotiate = { Transport,  },
Transform set #$!default_transform_set_0: { esp-3des esp-sha-hmac  }
   will negotiate = { Transport,  },

Washinton#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id slot status
63.100.202.225  63.100.202.229  QM_IDLE           1092    0 ACTIVE


IPv6 Crypto ISAKMP SA


Washinton#show ip int s0/1/0
Serial0/1/0 is up, line protocol is up (connected)
  Internet address is 63.100.202.229/30
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is not set
  Inbound  access list is 120
  Proxy ARP is enabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachables are always sent
  ICMP mask replies are never sent
  IP fast switching is disabled
  IP fast switching on the same interface is disabled
  IP Flow switching is disabled
  IP Fast switching turbo vector
  IP multicast fast switching is disabled
  IP multicast distributed fast switching is disabled
  Router Discovery is disabled
 --More--
```

Copy Paste

VPN screenshot 1 starting

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
ISAKMP:(1041):Send initial contact

ISAKMP:(1041):SA is doing pre-shared key authentication using id type
ID_IPV4_ADDR

ISAKMP (0:1041): ID payload
next-payload : 8
type         : 1
address      : 63.100.202.225
protocol     : 17
port         : 500
length       : 12


ISAKMP:(1041):Total payload length: 12

ISAKMP:(1041): sending packet to 63.100.202.229 my_port 500 peer_port
500 (I) MM_KEY_EXCH

ISAKMP:(1041):Sending an IKE IPv4 Packet.

ISAKMP:(1041):Input = IKE_MESG_INTERNAL, IKE_PROCESS_COMPLETE

ISAKMP:(1041):Old State = IKE_I_MM4  New State = IKE_I_MM5

ISAKMP (0:1041): received packet from 63.100.202.229 dport 500 sport
500 Global (I) MM_KEY_EXCH

ISAKMP:(1041): processing ID payload. message ID = 0

ISAKMP (0:1041): ID payload
next-payload : 8
type         : 1
address      : 63.100.202.229
protocol     : 17
```

Copy Paste

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
port          : 500
length        : 12


ISAKMP:(0):: peer matches *none* of the profiles

ISAKMP:(1041): processing HASH payload. message ID = 0

ISAKMP:(1041):SA authentication status:

authenticated

ISAKMP:(1041):SA has been authenticated with 63.100.202.229

ISAKMP: Trying to insert a peer 63.100.202.225/63.100.202.229/500/,
and inserted successfully 47CA9F80.

ISAKMP:(1041):Input = IKE_MESG_FROM_PEER, IKE_MM_EXCH

ISAKMP:(1041):Old State = IKE_I_MM5  New State = IKE_I_MM6

ISAKMP:(1041):Input = IKE_MESG_INTERNAL, IKE_PROCESS_MAIN_MODE

ISAKMP:(1041):Old State = IKE_I_MM6  New State = IKE_I_MM6

ISAKMP:(1041):Input = IKE_MESG_INTERNAL, IKE_PROCESS_COMPLETE

ISAKMP:(1041):Old State = IKE_I_MM6  New State = IKE_P1_COMPLETE

IPSEC(sa_request): ,
  (key eng. msg.) OUTBOUND local= 63.100.202.225, remote=
63.100.202.229,
    local_proxy= 192.168.11.0/255.255.255.0/0/0 (type=4),
    remote_proxy= 10.10.10.0/255.255.255.0/0/0 (type=4),
    protocol= ESP, transform= esp-aes esp-md5-hmac(Tunnel),
    lifedur= 3600s and 4608000kb,
```

Copy Paste

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
    protocol= ESP, transform= esp-aes esp-md5-hmac(Tunnel),
    lifedur= 3600s and 4608000kb,
    spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x0


ISAKMP:(1041):beginning Quick Mode exchange, M-ID of 69859174

ISAKMP:(1041):QM Initiator gets spi

ISAKMP:(1041): sending packet to 63.100.202.229 my_port 500 peer_port
500 (I) QM_IDLE

ISAKMP:(1041):Sending an IKE IPv4 Packet.

ISAKMP:(1041):Node 69859174, Input = IKE_MESG_INTERNAL, IKE_INIT_QM

ISAKMP:(1041):Old State = IKE_QM_READY  New State = IKE_QM_I_QM1

ISAKMP:(1041):Input = IKE_MESG_INTERNAL, IKE_PHASE1_COMPLETE

ISAKMP:(1041):Old State = IKE_P1_COMPLETE  New State =
IKE_P1_COMPLETE

ISAKMP (0:1041): received packet from 63.100.202.229 dport 500 sport
500 Global (I) QM_IDLE

ISAKMP:(1041): processing HASH payload. message ID = 69859174

ISAKMP:(1041): processing SA payload. message ID = 69859174

ISAKMP:(1041):Checking IPSec proposal 1
IPSEC(validate_proposal_request): proposal part #2


IPSEC(validate_proposal_request): proposal part #2,
```

Copy Paste

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
IPSEC(validate_proposal_request): proposal part #2


IPSEC(validate_proposal_request): proposal part #2,
  (key eng. msg.) INBOUND local= 63.100.202.225, remote=
63.100.202.229,
    local_proxy= 192.168.11.0/255.255.255.0/0/0 (type=4),
    remote_proxy= 10.10.10.0/255.255.255.0/0/0 (type=4),
    protocol= ESP, transform= NONE  (Tunnel), /n    lifedur= 0s and
0kb,
    spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x0


 Crypto mapdb : proxy_matchsrc addr     : 192.168.11.0
dst addr     : 10.10.10.0
protocol     : 0
 + src port     : 0
 + dst port     : 0


ISAKMP: transform 1, ESP_AES

ISAKMP:   attributes in transform:

ISAKMP:      encaps is 1 (Tunnel)

ISAKMP:      SA life type in seconds

ISAKMP:      SA life duration (basic) of 3600

ISAKMP:      SA life type in kilobytes

ISAKMP:      SA life duration (VPI) of  0x0 0x46 0x50 0x0

ISAKMP:      group is 5
```

Copy Paste

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
ISAKMP:      authenticator is HMAC-SHA

ISAKMP:(1041):atts are acceptable.

Crypto mapdb : proxy_match
src addr     : 192.168.11.0
dst addr     : 10.10.10.0
dst protocol     : 0
src port     : 0
dst port     : 0


 IPSEC(crypto_ipsec_sa_find_ident_head): reconnecting with the same
proxies and peer 63.100.202.229


 IPSEC(policy_db_add_ident): src 192.168.11.0, dest 10.10.10.0,
dest_port 0


 IPSEC(create_sa): sa created,
  (sa) sa_dest= 63.100.202.229, sa_proto= 50,
    sa_spi= 8F52BB22(2404563746),
    sa_trans= ESP_AES HMAC-SHA , sa_conn_id= 1041


 IPSEC(key_engine): got a queue event with 1 KMI message(s)

 IPSEC(key_engine_enable_outbound): rec'd enable notify from ISAKMP

IPSEC(key_engine_enable_outbound): enable SA with spi 4135910942/50

IPSEC(update_current_outbound_sa): updated peer 63.100.202.229current
outbound sa to SPI F684FE1E
```

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
IPSEC(key_engine_enable_outbound): enable SA with spi 415591094270

IPSEC(update_current_outbound_sa): updated peer 63.100.202.229current
outbound sa to SPI F684FE1E

ISAKMP:(1041): processing NONCE payload. message ID = 69859174

ISAKMP:(1041): processing KE payload. message ID = 69859174

ISAKMP:(1041): processing ID payload. message ID = 69859174

ISAKMP:(1041): processing ID payload. message ID = 69859174

ISAKMP:(1041): Creating IPSec SAs


         inbound SA from 63.100.202.229 to 63.100.202.225 (f/i)  0/ 0
        (proxy 10.10.10.0 to 192.168.11.0)


         has spi ox8F52BB22 and conn_id 0

         lifetime of 3600 seconds

         lifetime of 4608000 kilobytes

         outbound SA from 63.100.202.225 to 63.100.202.229 (f/i) 0/0
        (proxy 192.168.11.0 to 10.10.10.0)


         has spi  0xF684FE1E and conn_id 0

         lifetime of 3600 seconds

         lifetime of 4608000 kilobytes
```

Copy Paste

Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
        has spi  0xF684FE1E and conn_id 0

        lifetime of 3600 seconds

        lifetime of 4608000 kilobytes

ISAKMP:(1041): sending packet to 63.100.202.229  my_port 500
peer_port 500 (I) QM_IDLE

ISAKMP:(1041):Sending an IKE IPv4 Packet.

ISAKMP:(1041):deleting node 69859174 error FALSE reason "No Error"

ISAKMP:(1041):Node 69859174, Input = IKE_MESG_FROM_PEER, IKE_QM_EXCH

ISAKMP:(1041):Old State = IKE_QM_I_QM1  New State =
IKE_QM_PHASE2_COMPLETE

Chicago(config)#

Chicago con0 is now available

Press RETURN to get started.
```

Copy Paste

PC0

Physical Config Desktop Programming Attributes

Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.10.10.2

Pinging 10.10.10.2 with 32 bytes of data:

Request timed out.
Request timed out.
Reply from 10.10.10.2: bytes=32 time=40ms TTL=126
Reply from 10.10.10.2: bytes=32 time=20ms TTL=126

Ping statistics for 10.10.10.2:
    Packets: Sent = 4, Received = 2, Lost = 2 (50% loss),
Approximate round trip times in milli-seconds:
    Minimum = 20ms, Maximum = 40ms, Average = 30ms

C:\>
```

Laptop0

Physical Config Desktop Programming Attributes

Command Prompt

```
^C
C:\>
C:\>ping 10.10.10.1

Pinging 10.10.10.1 with 32 bytes of data:

Request timed out.
Reply from 10.10.10.1: bytes=32 time=34ms TTL=126
Reply from 10.10.10.1: bytes=32 time=16ms TTL=126
Reply from 10.10.10.1: bytes=32 time=22ms TTL=126

Ping statistics for 10.10.10.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 16ms, Maximum = 34ms, Average = 24ms

C:\>
```

PC1

Physical Config Desktop Programming Attributes

Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.11.2

Pinging 192.168.11.2 with 32 bytes of data:

Reply from 192.168.11.2: bytes=32 time=38ms TTL=126
Reply from 192.168.11.2: bytes=32 time=29ms TTL=126
Reply from 192.168.11.2: bytes=32 time=32ms TTL=126
Reply from 192.168.11.2: bytes=32 time=34ms TTL=126

Ping statistics for 192.168.11.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 29ms, Maximum = 38ms, Average = 33ms
```

Server0

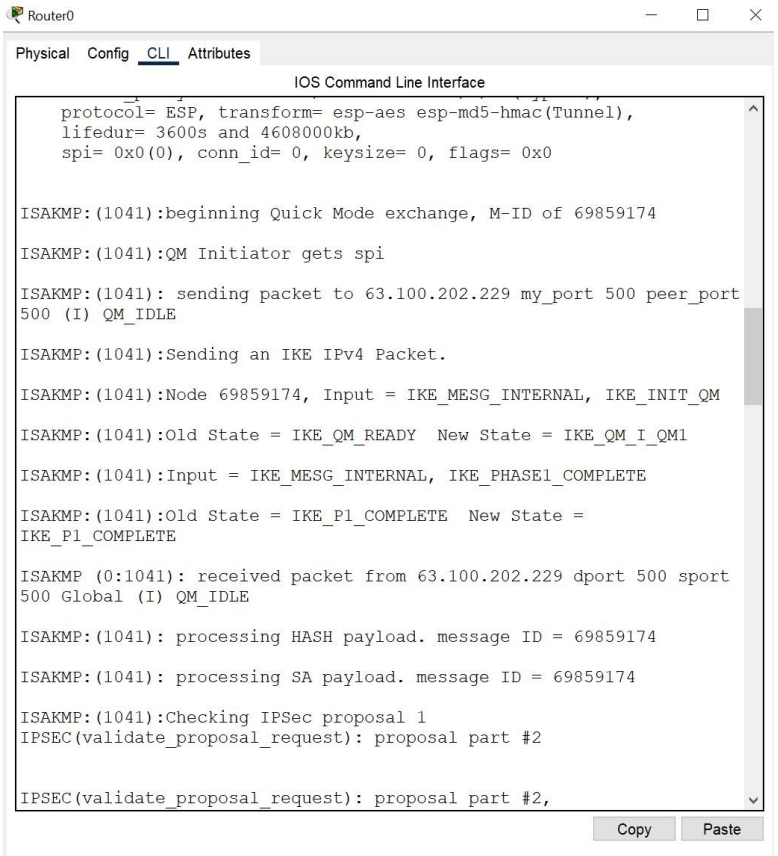
Physical Config Services Desktop Programming Attributes

Command Prompt

```
C:\>
C:\>
C:\>ping 192.168.11.1

Pinging 192.168.11.1 with 32 bytes of data:

Reply from 192.168.11.1: bytes=32 time=32ms TTL=126
Reply from 192.168.11.1: bytes=32 time=23ms TTL=126
Reply from 192.168.11.1: bytes=32 time=34ms TTL=126
Reply from 192.168.11.1: bytes=32 time=28ms TTL=126

Ping statistics for 192.168.11.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 23ms, Maximum = 34ms, Average = 29ms

C:\>
C:\>
```

Part#2: Include your answers here make sure you refer to the question's number:

1. The peer routers I used for each end router in the tunnel negotiated and established the security associations which encrypted and authenticated the traffic. As show in my 7 vpn tunnel screenshots, in my 1st screenshot which I have also attached again below for reference, after the line IKE_P1_complete, we can see Phase 2 starting on the next line where the Washington router peer is sending back a received packet in the lin "received packet from 63.100.202.229 dport 500 sport 500 Global (I) QM_IDLE and it starts to process the IPSEC security association requirements and these keys are dynamically generated during the negotiation process which are shows in the other VPN screenshots I attached in part 1. In the 2nd screenshot is where the peer of Router 0 Chicago which is 63.100.202.229, there are several confirmations labeled 1041 which is part of the contents in Message ID 69859174 where there are packets being confirmed from a message the peer sent back to the source router through port 500 that it is communicating with. As seen in my VPN screenshots from part 1, in screenshots 4 through 7, the peer router has send back the results from the transform set that it agreed on. Also, during the Phase 2 exchange, the session keys are randomly generated based on the transform set encryption/integrity algorithm rules I set, which were esp-aes and esp-MD5-hmac for the hashing. Another thing important to note, is the peer and the local IP both have proxy ID's, and as long as these proxy IDs match during the negotiation phase 2.



Router0

Physical Config CLI Attributes

IOS Command Line Interface

```
IPSEC(key_engine_enable_outbound): enable SA with spi 415991094(50

IPSEC(update_current_outbound_sa): updated peer 63.100.202.229current
outbound sa to SPI F684FE1E

ISAKMP:(1041): processing NONCE payload. message ID = 69859174

ISAKMP:(1041): processing KE payload. message ID = 69859174

ISAKMP:(1041): processing ID payload. message ID = 69859174

ISAKMP:(1041): processing ID payload. message ID = 69859174

ISAKMP:(1041): Creating IPSec SAs


         inbound SA from 63.100.202.229 to 63.100.202.225 (f/i)  0/ 0
        (proxy 10.10.10.0 to 192.168.11.0)


         has spi ox8F52BB22 and conn_id 0

         lifetime of 3600 seconds

         lifetime of 4608000 kilobytes

         outbound SA from 63.100.202.225 to 63.100.202.229 (f/i) 0/0
        (proxy 192.168.11.0 to 10.10.10.0)


         has spi  0xF684FE1E and conn_id 0

         lifetime of 3600 seconds

         lifetime of 4608000 kilobytes
```

Copy Paste

2. Depending on if it's a NGFW that can perform VPN/IPSec services, so I would be able to program the IPSec VPN tunnel in the actual firewall on each endpoint. But for the purpose of this assignment, what I did was I had to permit the UDP protocol which is used for the internet key exchange negotiations where it exchanges the NET477 key. In the simplest way possible, you would need to permit UDP port 500, which I did on my routers with my UDP permit rule. You would also need to permit ESP protocol on the firewall which I also did in my permit esp rule. If you are using AH protocol, you would need to permit that as well. You would need to set ingress and egress rules for each direction to allow or deny traffic depending on the requirements. To summarize, depending on the setup and requirements, you would want to permit/deny tcp/ip traffic for the from designated source ranges to be accepted and then allow esp/udp traffic from the necessary ports for the tunnels to be established.

3. The SA is initiated and negotiated in the beginning of Phase 2 for IPsec. However, the SAs are created ***after*** the tunnel is created in Phase 1 after negotiating the SA policy and thereafter after the transform-sets and key exchanges are confirmed during Phase 2. "IPsec SA lifetimes are negotiated during IKE phase 2" – quote from class PowerPoint.

4. "A VPN client sending IPsec-protected traffic to another VPN user by allowing that traffic in and out of the same interface. This is also called "hairpinning." – Source (https://www.cisco.com/c/en/us/td/docs/security/asa/asa94/config-guides/cli/vpn/asa-94-vpn-config/vpn-params.pdf) Pg 2.