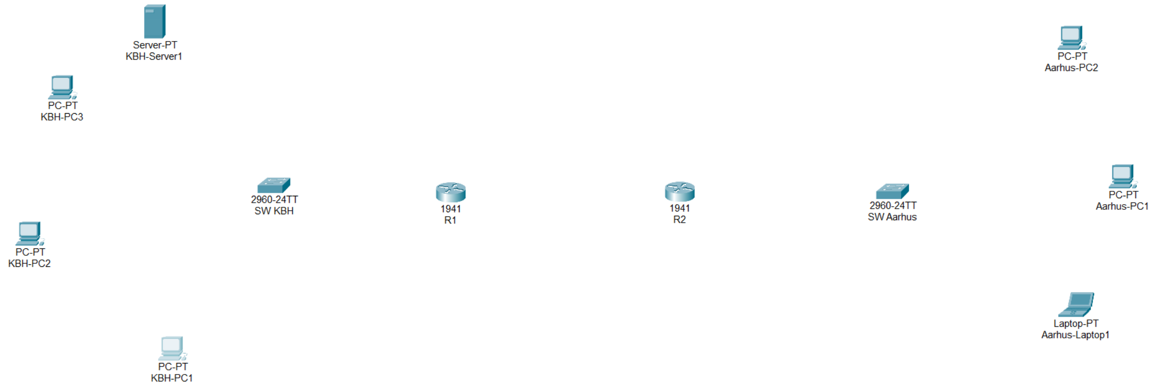
STEP BY STEP PACKET TRACER

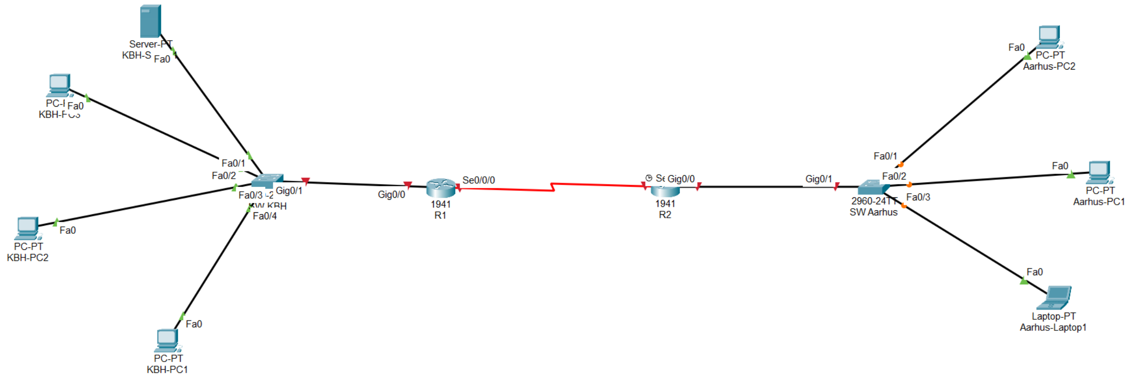
By Sanne N. Nielsen

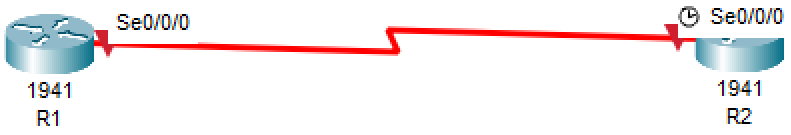
Indhold

Step 1: Topology

Set up the topology as stated beneath the pictures and use the corresponding cables.







DTE/DCE between the routers and Straight-Through copper cables for everything else. Remember to add the serial ports on the routers. ***Physical > Turn off router > Drag-and-drop 'HWIC-2T' to the right open spot > Drag-and-drop 'WIC-Cover' to the left open sot > Turn the router back on***

Step 2: Change device names and add notes.

Change the name in the topology as seen above, but also change hostname on each device.

```
Router(config)#hostname [name after topology]
```

Or be sneaky and change it in the device window under the section called config and change it there.

Remember setting a banner on every router and switch!

```
Router(config)#banner motd #Autherized Access Only!#
```

Step 3: IP-addressing with VLSM subnetting.

With our calculations we will address the biggest subnet first, or it won't work. So go on R1 and enter EXEC privileged mode (enable) and then the global configuration mode (config t/configure terminal).

```
R1(config)#int g0/0

R1(config-if)# ip add 192.168.20.30 255.255.255.224

R1(config-if)# ip helper-address 192.168.20.28
```

- A 'mini GPS' command for the routers to use to establish a path to the DHCP server.

```
R1(config-if)# no shut

R1(config-if)# description KBH LAN
```

Then you go to R2 and do the following:

```
R2(config)#int g0/0

R2(config-if)# ip add 192.168.20.46 255.255.255.240

R2(config-if)# ip helper-address 192.168.20.28
```

- A 'mini GPS' command for the routers to use to establish a path to the DHCP server.

```
R2(config-if)# no shut

R2(config-if)# description AARHUS LAN
```

Back to R1:

```
R1(config)#int s0/0/0

R1(config-if)# ip add 192.168.20.49 255.255.255.252

R1(config-if)# ip helper-address 192.168.20.28
```

- A 'mini GPS' command for the routers to use to establish a path to the DHCP server.

```
R1(config-if)# no shut

R1(config-if)# description WAN Port KBH
```

Back to R2:

```
R2(config)#int s0/0/0

R2(config-if)# ip add 192.168.20.50 255.255.255.252

R1(config-if)# ip helper-address 192.168.20.28
```

- A 'mini GPS' command for the routers to use to establish a path to the DHCP server.

```
R2(config-if)# no shut

R2(config-if)# description WAN Port AARHUS
```

Step 4: Router Rip

Then it is time for Router Rip! You can do it on either router first. We just need to set it up for the two routers to be able to talk with each other. Or they won't be able to talk across the different LANs and WAN.

```
R1/R2(config)# router rip

R1/R2(config-router)# version 2

R1/R2(config-router)# network 192.168.20.0

R1/R2(config-router)# network 192.168.20.32

R1/R2(config-router)# network 192.168.20.48

R1/R2(config-router)# no auto-summary
```

Repeat on the second router.

Step 5: Add IP to Switches

Like with the router we need to add IP-addresses to the VLAN’s of each switch to be able to remote to it.

SW-KBH:

```
SW-KBH(config)# int vlan1

SW-KBH(config-if)# ip add 192.168.20.29 255.255.255.224

SW-KBH(config-if)# no shut

SW-KBH(config-if)# exit

SW-KBH(config)# ip default-gateway 192.168.20.30
```

SW-AARHUS:

```
SW-KBH(config)# int vlan1

SW-KBH(config-if)# ip add 192.168.20.45 255.255.255.240

SW-KBH(config-if)# no shut

SW-KBH(config-if)# exit

SW-KBH(config)# ip default-gateway 192.168.20.46
```

Step 6: Security

Now we have to set up the different security measures on each device.

- Line Console 0: Ab12345678
- Line VTY 0 4 (0 15, on switches): **login local**
- Transport input ssh

Go to global configuration mode in either R1 or R2. You will have to do the same steps on them both.

```
R1(config)# ip domain-name CMIS.dk
```

- Give the device a domain to look up.

```
R1(config)# crypto key generate rsa general-keys modulus 1024
```

- Generates a crypto key to ensure a secure connection when using in bands connections.

```
R1(config)# service password-encryption
```

- Encrypt all plain clear passwords and the ones added after.

```
R1(config)# security password min-length 10
```

- Require every password on the device to be at least 12 characters.
- Does not function on switch.

```
R1(config)# username jk.it secret Ab12345678
```

- Creates a local admin user for ssh connections.

```
R1(config)# login block-for 120 attempts 3 within 60
```

- Blocks login attempts for 120 seconds, after 3 wrong attempts within 60 seconds.
- Does not function on a switch.

```
R1(config)# enable secret Ab12345678
```

<u>Line Console 0:</u>

```
R1(config)# line con 0

R1(config-line)# password Ab12345678

R1(config-line)# login

R1(config-line)# logging synchronous
```

- Makes sure the line doesn’t break when typing in any passwords.

```
R1(config-line)# exit
```

<u>VTY Lines:</u>

R1(config)# **line vty 0 4**

R1(config-line)# **transport input ssh**
- Opens up the protocol for ssh connections only.

R1(config-line)# **logging synchronous**

R1(config-line)# **login local**
- Makes one able to login with a local user.

R1(config)# **exec-timeout 10**
- Logs one out after 10 minutes of inactivity

R1(config-line)# **exit**

Repeat on R2.

Then repeat the steps on the switches, BUT you must add these commands as well.

SW-KBH(config)# **line vty 5 15**

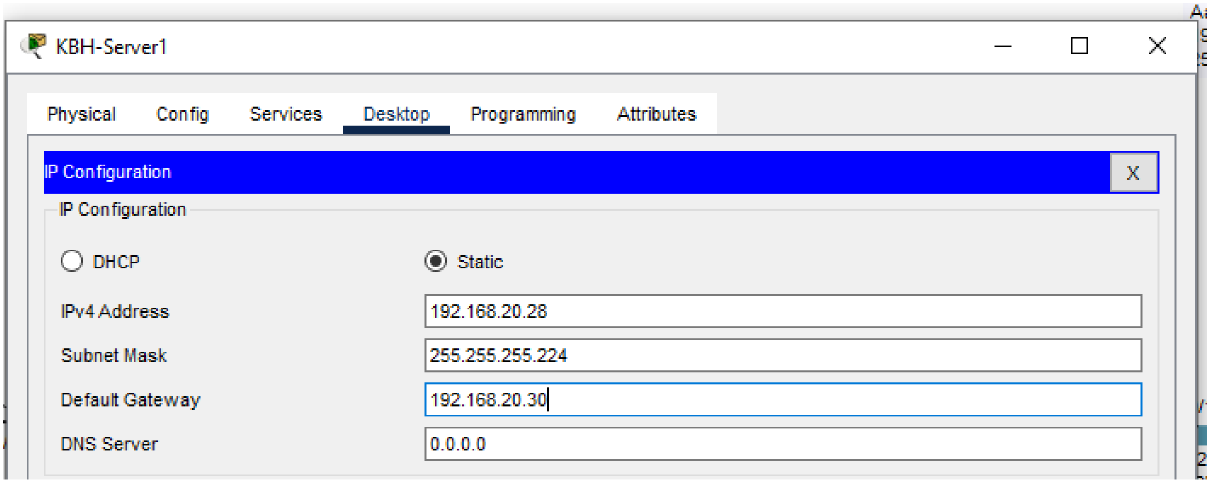
SW-KBH(config-line)# **transport input none**
- Shuts down these lines for any connections, by closing all the protocols.

There may be some of these security commands that gives errors. If so skip them. Some of them only works on routers.

Fun fact!
There is several SVI ports we can use, but since VLAN1 is by default turn on and running, we are using that one for the lesson/assignments. In real life you never use VLAN1 as it can pose as a security risk because it is usually turned 'on' on a cisco device from the moment you boot it up. Basically, it is a way to split up a network so that you can restrict or give access to the right users/devices.
This site explain it really well! Link to site.

Step 7: DHCP server.

Double click on the server. Under desktop choose IP-configuration and type in this:



IP Configuration

○ DHCP ● Static

IPv4 Address: 192.168.20.28

Subnet Mask: 255.255.255.224

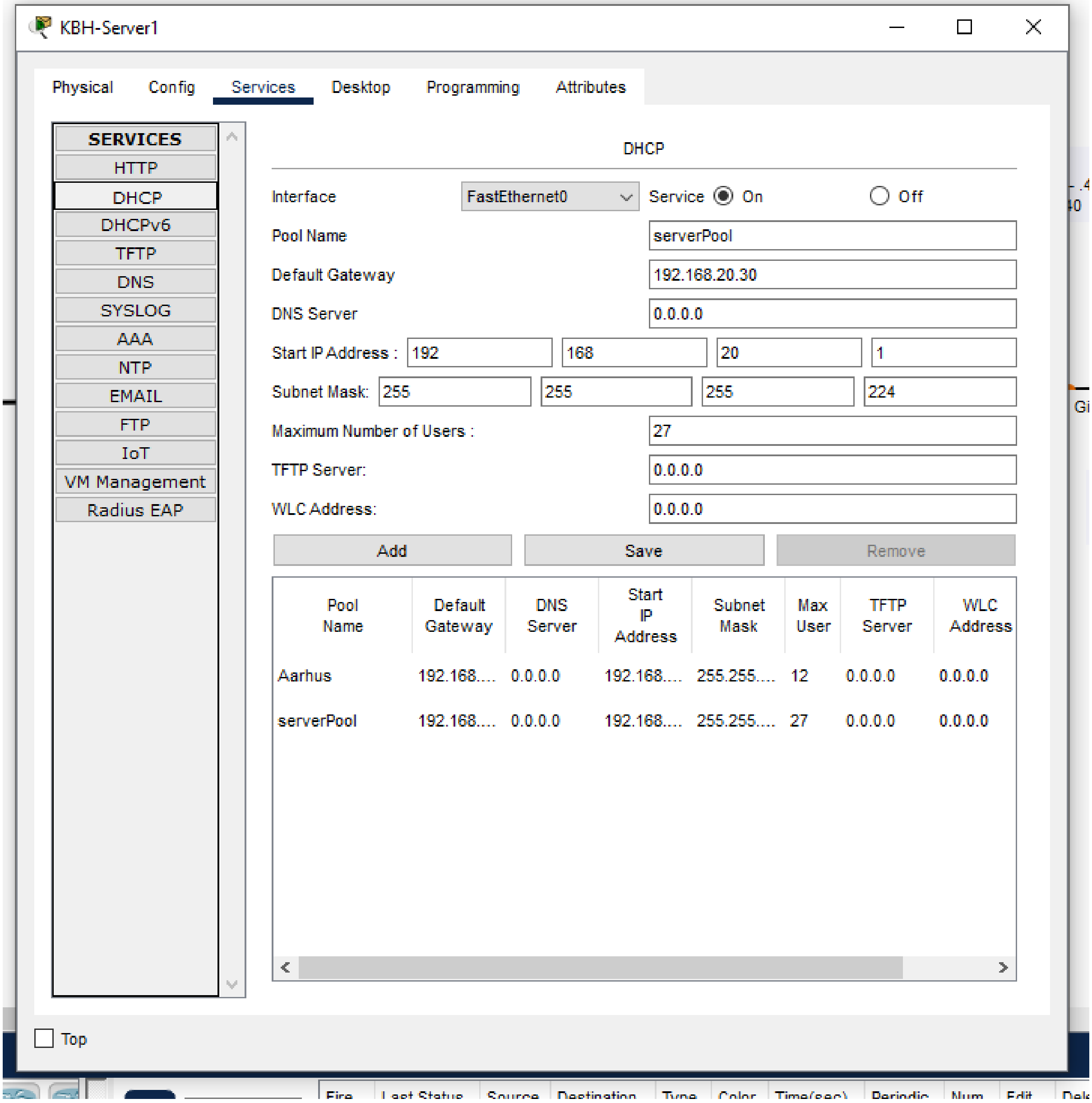
Default Gateway: 192.168.20.30

DNS Server: 0.0.0.0

IPv4 Address: 192.168.20.28
Subnet Mask: 255.255.255.224
Default-Gateway: 192.168.20.30

Go under 'Services' now. You will have to make IP pools. Basically it is the pools of IP addresses the server will use to assign the stuff.

The KBH one has to be the serverPool one because of a Packet Tracer glitch. So it should get filled out like this:



DHCP

Interface: FastEthernet0 | Service ● On ○ Off

Pool Name: serverPool

Default Gateway: 192.168.20.30

DNS Server: 0.0.0.0

Start IP Address : 192 . 168 . 20 . 1

Subnet Mask: 255 . 255 . 255 . 224

Maximum Number of Users : 27

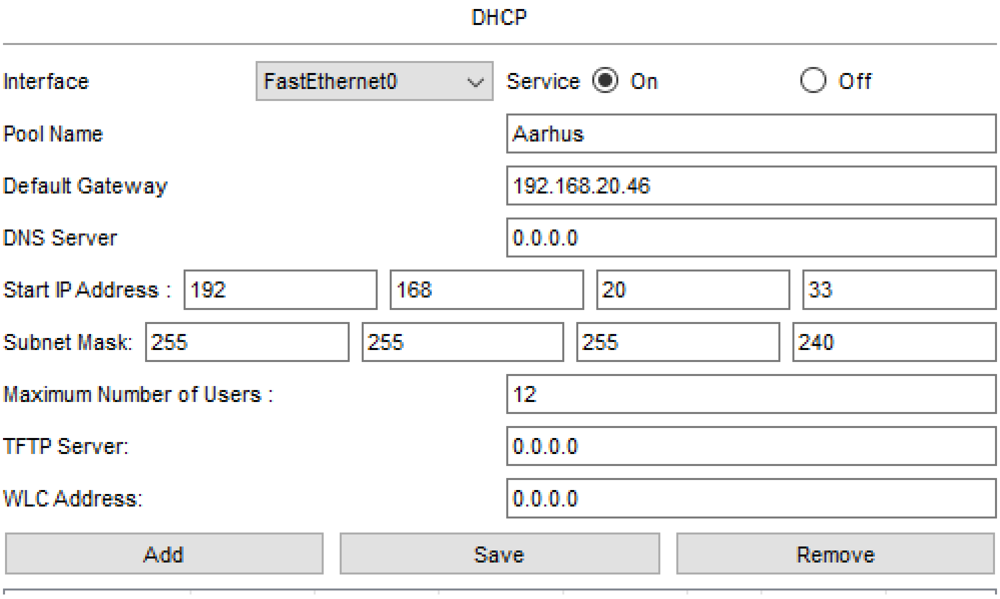
TFTP Server: 0.0.0.0

WLC Address: 0.0.0.0

Pool Name	Default Gateway	DNS Server	Start IP Address	Subnet Mask	Max User	TFTP Server	WLC Address
Aarhus	192.168....	0.0.0.0	192.168....	255.255....	12	0.0.0.0	0.0.0.0
serverPool	192.168....	0.0.0.0	192.168....	255.255....	27	0.0.0.0	0.0.0.0

Remember to press save when everything has been filled out.

While the Aarhus one has to be filled out like this:



DHCP

Interface	FastEthernet0	Service ● On	○ Off
Pool Name	Aarhus		
Default Gateway	192.168.20.46		
DNS Server	0.0.0.0		
Start IP Address : 192	168	20	33
Subnet Mask: 255	255	255	240
Maximum Number of Users :	12		
TFTP Server:	0.0.0.0		
WLC Address:	0.0.0.0		

Remember to turn on the 'on' circle check box at the top. And press add to add Aarhus.

You should now be able to go to a PC device on either device and go in IP-configuration and check the DHCP instead of static.

Step 8: Shutting down unused Ports.

We must close the different ports we don’t use. What do is we are choosing to edit a whole range of ports, instead of just one at a time.

SW KBH:

```
SW-KBH(config)# int range f0/5 – 24
```

-This ‘int range’ indicates we are grabbing a whole range instead of one port at a time.

```
SW-KBH(config-if-range)# shutdown

SW-KBH(config-if-range)# exit



SW-KBH(config-if)# int g0/2

SW-KBH(config-if-range)# shutdown

SW-KBH(config-if-range)# exit
```

SW Aarhus:

```
SW-AARHUS(config)# int range f0/4 – 24

SW-AARHUS(config-if-range)# shutdown

SW-AARHUS(config-if-range)# exit



SW-AARHUS(config)# int  g0/2

SW-AARHUS(config-if-range)# shutdown

SW-AARHUS(config-if-range)# exit
```

R1 & R2:

```
R1/R2(config)# int g0/1

R1/R2(config-if)# shut

R1/R2(config-if)# exit


R1/R2(config)# int se0/0/1

R1/R2(config-if)# shut

R1/R2(config-if)# exit
```

Repeat on both Routers!

Step 9: VPN

Okay then we have to set up the VPN. I can't tell you exactly what every command does, but I can give a rough take. Note that since we did not get anything about in class, we just have to tell censor, we can set it up and make it work. That's the thing we can.

First thing first. Do a traceroute from a PC in KBH to a PC in Aarhus and take a screenshot. Then we have something to compare to when we have set it all up.

R1 Setup: Step 1 Security Technology Package

R1(config)# **license boot module c1900 technology-package securityk9**
-This is a command we have to use to enable the Security Technology Package required to set up the VPN.
- If a user agreement does not pop up, type in the command again until it does.
- Accept the user agreement by typing 'yes'.

R1(config)# **exit**

R1# **copy running-config startup-config**
- Save the running-config to the startup-config after we just agreed to the user agreement.

R1# **reload**
- Reloads the switch after we got the security package on.
- Use 'R1# **show version**' to go check if it is turned on.

R1 Setup: Step 2 Identifying interesting traffic.

R1(config)# **access-list 100 permit ip 192.168.20.0 0.0.0.31 192.168.20.32 0.0.0.15**
- *What this command does is making the two LANs to interesting traffic. Aka traffic that will be encrypted. The reason it is 0.0.0.31 on the first LAN (KBH) is because you have to minus the original subnet mask with 255.255.255.255. The same with Aarhus LAN. That one will get 0.0.0.15 instead.*
- ***KBH LAN**: 255.255.255.224 – 255.255.255.255 = 0.0.0.31*
- ***AARHUS LAN**: 255.255.255.240 – 255.255.255.255 = 0.0.0.15*

R1 Setup: Step 3 Configure the IKE Phase 1 ISAKAMP policy.

R1(config)# **crypto isakmp policy 10**
- Entering the policy

R1(config-isakmp)# **encryption aes 128**
- Choosing the encryption method, I think

R1(config-isakmp)# **authentication pre-share**
- Key exchange method, whatever that means

R1(config-isakmp)# **group 5**
- Choosing the highest DH group? They say it is the highest available in Packet tracer. It is usually DH 14 normally.

R1(config-isakmp)# **exit**

R1(config)# **crypto isakmp key vpn address 192.168.20.50**
- Telling that the VPN needs to go from R1 to R2' serialport with this WAN IP address.

R1 Setup: Step 4 Configure the IKE Phase 2 IPsec policy.

R1(config)# **crypto ipsec transform-set VPN-P2 esp-aes esp-sha-hmac**
- Creates the transform-set VPN-P2 to use **esp-aes** and **esp-sha-hmac**.
- I think it is some kind of protocol in how it will forward the data safely.

R1(config)# **crypto map VPN-MAP 10 ipsec-isakmp**
- Entering the VPN map that will bind everything together.

R1(config-crypto-map)# **description VPN connection to R2**
- Just a description.

R1(config-crypto-map)# **set peer 192.168.20.50**
- Setting the peer to be the second router. So the VPN know where the other router it has the VPN set up with.

R1(config-crypto-map)# **set transform-set VPN-P2**
- Telling it to use this method for the VPN connection.

R1(config-crypto-map)# **match address 100**
- I think this is just to make sure you can map out the same address on both routers?

R1(config-crypto-map)# **exit**

R1 Setup: Step 5 Configure the crypto map on the outgoing interface.

```
R1(config)# interface s0/0/0
```

```
R1(config-if)# crypto map VPN-MAP
```

- Just making sure that the interface uses the VPN map we just created in the steps above.

R2 Setup: Step 6 Security Technology Package.

```
R2(config)# license boot module c1900 technology-package securityk9
```

-This is a command we have to use to enable the Security Technology Package required to set up the VPN.

- If a user agreement does not pop up, type in the command again until it does.
- Accept the user agreement by typing 'yes'.

```
R2(config)# exit
```

```
R2# copy running-config startup-config
```

- Save the running-config to the startup-config after we just agreed to the user agreement.

```
R2# reload
```

- Reloads the switch after we got the security package on.
- Use 'R2# **show version**' to go check if it is turned on.

R2 Setup: Step 7 Identifying interesting traffic.

```
R2(config)# access-list 100 permit ip 192.168.20.32 0.0.0.15 192.168.20.0 0.0.0.31
```

- *What this command does is making the two LANs to interesting traffic. Aka traffic that will be encrypted. AARHUS LAN first because we are on R2 now.*

- ***KBH LAN****: 255.255.255.224 – 255.255.255.255 = 0.0.0.31*
- ***AARHUS LAN****: 255.255.255.240 – 255.255.255.255 = 0.0.0.15*

R2 Setup: Step 8 Configure the IKE Phase 1 ISAKAMP policy.

```
R2(config)# crypto isakmp policy 10
```

- Entering the policy

```
R2(config-isakmp)# encryption aes 128
```

- Choosing the encryption method, I think

```
R2(config-isakmp)# authentication pre-share
```

- Key exchange method, whatever that means

```
R2(config-isakmp)# group 5
```

- Choosing the highest DH group? They say it is the highest available in Packet tracer. It is usually DH 14 normally.

```
R2(config-isakmp)# exit
```

```
R2(config)# crypto isakmp key vpn address 192.168.20.49
```

- Telling that the VPN needs to go from R2 to R1' serialport with this WAN IP address.

R2 Setup: Step 9 Configure the IKE Phase 2 IPsec policy.

```
R2(config)# crypto ipsec transform-set VPN-P2 esp-aes esp-sha-hmac
```

- Creates the transform-set VPN-P2 to use **esp-aes** and **esp-sha-hmac**.
- I think it is some kind of protocol in how it will forward the data safely.

```
R2(config)# crypto map VPN-MAP 10 ipsec-isakmp
```

- Entering the VPN map that will bind everything together.

```
R2(config-crypto-map)# description VPN connection to R1
```

- Just a description.

```
R2(config-crypto-map)# set peer 192.168.20.49
```

- Setting the peer to be the first router. So the VPN know where the other router it has the VPN set up with.

```
R2(config-crypto-map)# set transform-set VPN-P2
```

- Telling it to use this method for the VPN connection.

```
R2(config-crypto-map)# match address 100
```

- I think this is just to make sure you can map out the same address on both routers?

```
R2(config-crypto-map)# exit
```

R2 Setup: Step 10 Configure the crypto map on the outgoing interface.

R2(config)# **interface s0/0/0**

R2(config-if)# **crypto map VPN-MAP**
- Just making sure that the interface uses the VPN map we just created in the steps above.

R1 & R2 Setup: Step 11 Verify the tunnel before and after interesting traffic.

Go back to R1 and enter this command:

R1# **show crypto ipsec sa**

It should show a lot of different package types that all is 0. Now try Ping between a PC in KBH and one in Aarhus.

Go back to R1 and reenter the same command:

R1# **show crypto ipsec sa**

Now the different package types should have numbers that isn't 0. That means that the VPN tunnel is working.

Now you can try a traceroute again and see the differences in the output.

Step 10: Test SSH Connection.

Go on a PC, enter desktop, then the commando prompt and write the following command:

C:/ **ssh -l jk.it** ***[ip-address for switches or routers]***

If you get a login prompt it is correctly set up. Also remember that '-l' is a lowercase 'L' for local login.