WWW.JAMK.FI

# GROUP ENCRYPTED TRANSPORT VIRTUAL PRIVATE NETWORK JA CISCO DYNAMIC MULTIPOINT VIRTUAL PRIVATE NETWORK -TOTEUTUS

Pasi Tuovinen

Opinnäytetyö
Kesäkuu 2012

Tietotekniikan koulutusohjelma
Tekniikan ja liikenteen ala

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

OPINNÄYTETYÖN
KUVAILULEHTI

| | | |
|---|---|---|
| Tekijä(t)<br>TUOVINEN, Pasi | Julkaisun laji<br>Opinnäytetyö | Päivämäärä<br>03.06.2012 |
| | Sivumäärä<br>161 | Julkaisun kieli<br>Suomi |
| | Luottamuksellisuus<br>( ) saakka | Verkkojulkaisulupa myönnetty<br>( X ) |

Työn nimi
GROUP ENCRYPTED TRANSPORT VIRTUAL PRIVATE NETWORK JA CISCO DMVPN TOTEUTUKSET

Koulutusohjelma
Tietoverkkotekniikka

Työn ohjaaja(t)
NARIKKA, Jorma

Toimeksiantaja(t)
Jyväskylän ammattikorkeakoulu Oy
VATANEN, Marko

Tiivistelmä

SpiderNet on Jyväskylän ammattikorkeakoulun teknologiayksikön laboratorioympäristö. Sen ensisijainen käyttötarkoitus on tietoverkkotekniikan koulutusohjelman opetuksessa, mutta sitä käytetään myös tutkimus – ja kehityshankkeissa sekä pohjana useissa opinnäytetöissä. Vaikka SpiderNet on osa opetusympäristö, se on kuitenkin täysin erillään Jyväskylän ammattikorkeakoulun muusta tuotantoverkosta.

Opinnäytetyön tarkoitus oli tutkia ja testata Cisco Systemsin sekä Juniper Networksin valmistamilla laitteilla Group Encrypted Transport ja Cisco DMVPN ratkaisuja ja mahdollisuuksia. Työ aloitettiin tutustumalla niin teoriassa tarkasteltaviin tekniikoihin kuin myös Juniper Networksin laitteisiin.

Työtä varten muodostettiin SpiderNet ympäristöön neljä mahdollisimman samankaltaista topologiaa Missä erilaisia yhdistelmiä tekniikoiden ja laitevalmistajien välillä testattiin. Kaikkien topologioiden lähtökohta oli simuloida Internetiä ja sitten liittää siihen yrityksen toimipisteitä kuvaavat laitteet, joissa itse teknologia testattiin. Työssä päädyttiin lopulta testaamaan neljä erilaista kokonaisuutta; Cisco DMVPN, Juniper Group VPN Co-location, Cisco GET VPN reduntattisuus sekä Ciscon ja Juniperin yhteensopivuus.

Työn pohjalta voitiin todentaa ja tutkia kyseisten teknologioiden toiminta. Lisäksi tehtiin kaksi laboratorio harjoitusta, joita voidaan käyttää niin koulutuskäytössä kuin lyhyenä pikaoppaana teknologioiden toimintaan.

Avainsanat (asiasanat)
VPN, DVMPN, GET VPN, GROUP VPN, Cisco, Junos, SpiderNet

Muut tiedot

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

DESCRIPTION

| Author(s) | Type of publication | Date |
|---|---|---|
| TUOVINEN, Pasi | Bachelor´s / Master's Thesis | 03.06.2012 |
| | Pages<br>161 | Language<br>Finnish |
| | Confidential<br><br>( ) Until | Permission for web publication<br>( X ) |

Title
IMPLEMENTATIONS OF GROUP ENCRYPTED TRANSPORT VIRTUAL PRIVATE NETWORK AND CISCO DMVPN

Degree Programme
Data Network Technology

Tutor(s)
NARIKKA, Jorma

Assigned by
JAMK University of Applied Sciences
VATANEN, Marko

Abstract

SpiderNet is a laboratory environment at JAMK University of Applied Sciences. SpiderNet´s main use is for data network technology courses and studie; it is, however also used in research and development projects and it provide a rich research base for many Bachelor´s Theses. While SpiderNet is part of the education network it is totally separated from other production networks in JAMK.

The main goal of this thesis was to study and implement different types of group encrypted transport technologies on Cisco Systems and Juniper Networks products. The project started with studying the technologies in theory as well as getting to know how Juniper Networks products operate.

For the thesis four different topologies were built in SpiderNet, and the idea was to keep the topologies similar in order to prove the differences in technologies. The basic idea on each topology was to simulate the Internet and then add two or more nodes to it as if they were offices in a company network. The thesis focuses on testing four different scenarios with the technologies; Cisco DMVPN, Juniper Group VPN Co-location, Cisco GET VPN redundancy and Cisco and Juniper commonality.

The result of this thesis were successfully tested topologies which could be used to prove and study the functionally of the technologies. In addition, based on the testing two laboratory exercises were made, which can be used a part of future education or as "quick guides" to the aforementioned technologies.

Keywords
VPN, DMVPN, GET VPN, GROUP VPN, Cisco, Junos, SpiderNET

Miscellaneous

SISÄLTÖ

KUVIOT

# LYHENTEET

| | |
|---|---|
| AES | Advanced Encryption Standard |
| AH | Authentication Header |
| CEF | Cisco Express Forwarding |
| DES | Data Encryption Standard |
| DH | Diffie-Hellman |
| DMVPN | Dynamic Multipoint Virtual Private Network |
| DNS | Domain Name Server |
| ESP | Encapsulating Security Payload |
| FIB | Forwarding Information Base |
| GDOI | Group Domain of Interpretation |
| GRE | Generic Routing Encapsulation |
| HMAC | Hash-based Message Authentication Code |
| IKE | Internet Key Exchange |
| ISAKMP | Internet Security Association and Key Management Protocol |
| IP | Internet Protocol |
| IPSec | Internet Protocol Security |
| MD5 | Message-Digest Algorithm |
| NHRP | Next-Hop Resolution Protocol |
| NHS | Next-Hop Server |
| OSI | Open Systems Interconnection |
| OSPF | Open Shortest Path First |
| SA | Security Association |
| SHA | Secure Hash Algorithm |
| SSH | Secure Shell |
| TCP | Transmission Control Protocol |
| UDP | User Datagram Protocol |
| VLAN | Virtual Local Area Network |
| VPN | Virtual Private Network |

# 1 TYÖN KUVAUS

## 1.1 Toimeksiantaja

Opinnäytetyön toimeksiantajana toimii Jyväskylän ammattikorkeakoulun (Jamk) tietotekniikan koulutusala. Jyväskylän ammattikorkeakoulun toimipisteet sijaitsevat Jyväskylässä sekä Saarijärven Tarvaalassa. Jyväskylän ammattikorkeakoulu tarjoaa korkeakoulututkintoon johtavaa koulutusta usealla alalla, ammatillista opettajakoulutusta, avoimia korkeakouluopintoja sekä erilaisia täydennyskoulutus mahdollisuuksia niin aikuisille kuin nuorille. Opiskelijoita Jyväskylän ammattikorkeakoulussa on jo yli 8000. (Jyväskylän ammattikorkeakoulu 2011a.)

Jyväskylän ammattikorkeakoululla on vahva asema Keski-Suomessa ja sen luomat yhteydet sekä kehitysyhteistyö alueen elinkeinoelämän kanssa näkyvät myös valmistuneiden korkeana työllistymisprosenttina. Tällä hetkellä valmistuneiden työllistymisprosentti on noin 74 prosenttia vuosi valmistumisen jälkeen. (Jyväskylän ammattikorkeakoulu 2011a.)

Tietotekniikan koulutusohjelma keskittyy nimensä mukaisesti tietoverkkotekniikan eri osa-aluille. Pääpaino opinnoissa keskittyy niin langallisten kuin langattomien operaattoriverkkojen opetukseen sekä kasvavassa määrin palvelin ja verkkojen ylläpitokoulutukseen. Opintojen käytännön osaa tukee vahvasti Jyväskylän ammattikorkeakoulun oma laboratorioympäristö SpiderNet. (Jyväskylän ammattikorkeakoulu 2011b.)

## 1.2 Tavoitteet

Työn tavoitteena oli tutkia Cisco Systems -reitittimille suunniteltua Dynamic Multipoint Virtual Private Network (DMVPN) konfiguraatiota sekä valmistajariippumatonta Multi Point Virtual Private Network(Multi Point VPN) konfiguraatiota ja tehdä niiden toiminnasta käytännön harjoitteita Jamkin tietotekniikan koulutuskäyttöön.

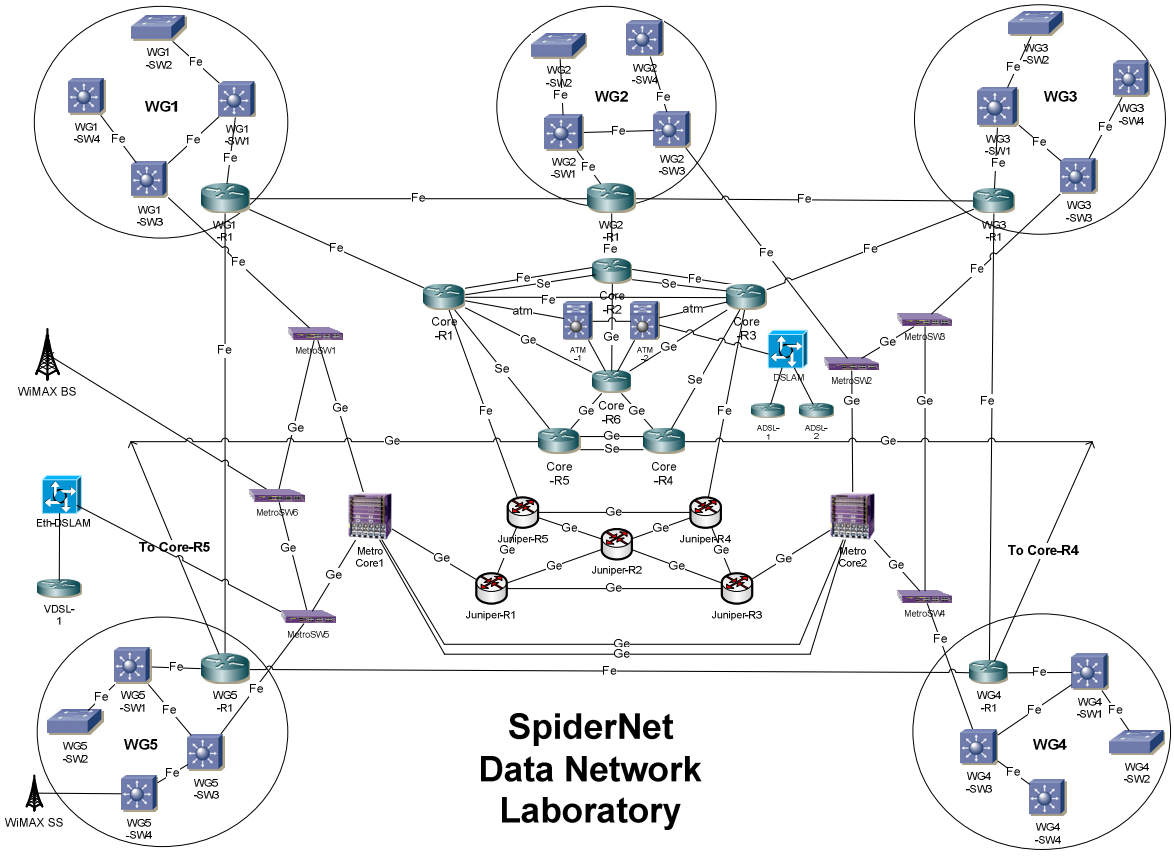
Tavoitteena oli muodostaa SpiderNet laboratorion laitteita käyttäen ympäristö, joka vastaisi yrityksen pääkonttoria sekä kahta tai useampaa toimipistettä joidenka välissä olisi julkinen operaattoriverkko, jota simuloimaan käytettiin kolmea Cisco Systems -reititintä. Tarkoituksena oli muodostaa tämän verkon yli toimiva DMVP tai Multi Point VPN -yhteys. DMVPN -konfiguraatio toteutettiin käyttäen kokonaan Cisco Systems reitittimiä. Multi Point VPN -konfiguraatiossa käytettiin Juniper Networks -reitittimiä sekä Cisco Systems -reitittimiä. Työn edetessä aihetta laajennettiin sitten, että tarkasteltiin vielä erikseen kokonaan Cisco -tai Juniper-laitteista muodostettuja topologioita ja niiden tarjoamia erikoisominaisuuksia.

Konfiguraatioiden testauksen lisäksi tavoitteena oli tehdä kaksi laboratorioharjoitusta. Harjoitusten tarkoitus oli käydä läpi tarvittavat konfiguraatiot DMVP ja Multi Point VPN -yhteyksien muodostamiseksi.

# 2 SPIDERNET

## 2.1 Yleistä

Jyväskylän ammattikorkeakoulun tietoverkkotekniikan koulutusohjelman käyttämä laboratorioympäristö on nimeltään SpiderNet. SpiderNet on käytännönharjoitteita varten rakennettu ympäristö, SpiderNetiä on kehitetty jo yli kymmenen vuotta ja sen kehittämistä jatketaan jatkuvasta uusien teknologioiden tarpeiden mukaan. SpiderNetin pääasiallinen käyttö on tietoverkkotekniikan koulutusohjelman opetuksessa, mutta sitä käytetään myös tutkimus - ja kehityshankkeissa. SpiderNet tarjoaa myös laajat mahdollisuudet opinnäytetöiden tekemiselle. SpiderNetin koko tämän hetkinen topologia on nähtävissä kuviosta 1.(SpiderNet 2011)




KUVIO 1. SpiderNet topologia (Spidernet 2011)

## 2.2 Laitteisto ja topologia

Työn tekohetkellä SpiderNet ympäristössä oli käytössä seuraavien valmistajien laitteita: Airspan Networks, Cisco Systems, Extreme Networks, Juniper Networks ja Zhones. SpiderNet rakentuu tällä hetkellä neljästä isommasta kokonaisuudesta: Cisco Core, Metro Core, Juniper Core sekä viisi työryhmää (SpiderNet 2011)

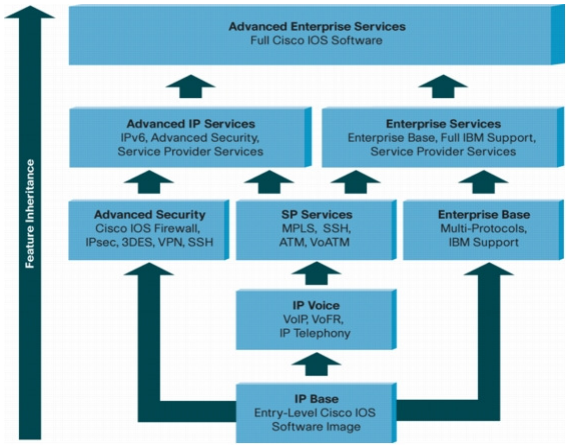
# 3 CISCO SYSTEMS

## 3.1 Cisco Systems yrityksen taustaa

Cisco Systems on monikansallinen verkkolaitteiden valmistaja. Yritys on perustettu vuonna 1984 San Franciscon. Cisco Systems työllistää tällä hetkellä yli 70000 henkilöä, ja se on johtava verkkolaitteiden valmistajista. Cisco Systemsin liikevaihto vuonna 2010 oli noin 40 miljardia dollaria. Laitevalmistuksen lisäksi Cisco tunnetaan myös laajalle levinneestä tietoverkko opetuksesta, jonka kursseja on mahdollista opiskella lähes jokaisessa tietoverkko opetusta tarjoavassa koulussa. Ciscon päämarkkinointialueet laitteiden ja palveluiden osalta ovat suurten yritysten ja operaattoreiden piirissä. (Cisco Company profile 2011)

## 3.2 Cisco IOS-käyttöjärjestelmä.

Cisco Systemsin nykyisten laitteiden käyttöjärjestelmänä toimii Cisco IOS (Internetwork Operation System). IOS on useista yhteen integroiduista ohjelmistoista koostuva kokonaisuus. Se on käytössä kaikissa Ciscon valmistamissa laitteissa riippumatta siitä oliko kyseessä reititin, kytkin tai tietoturvalaite. Cisco IOSia käytetään CLI -(command line interface) komentoja käyttäen. Peruskomennot ovat samat laitteesta tai IOS -versiosta riippumatta. Perus -IOSin lisäksi laitteille on saatavissa erilaisia paketteja, jotka lisäävät ominaisuuksia ja mahdollisuuksia laitteille esimerkiksi IPv6 tuen. (Cisco Company profile 2011).




KUVIO 2 Cisco –ominaisuuspaketit (Company profile 2011)

# 4 JUNIPER NETWORKS

## 4.1 Juniper Networks -yrityksen tausta

Juniper Networks on vuonna 1996 perustettu yhdysvaltainen verkkolaitevalmistaja. Juniper työllistää tällä hetkellä yli yhdeksän tuhatta henkeä, ja sen toimipisteitä sijaitsee 46 maassa. Se tarjoaa palveluitaan yli sadalle verkko-operaattorille ja kymmenille tuhansille yrityksille ja julkisensektorin organisaatioille. Juniperin laite ja palveluvalikoima on erittäin kattava ja siitä syystä sen käyttö on mahdollista verkon koosta ja tarpeista riippumatta.( Juniper Company Profile. 2011.)

## 4.2 Junos käyttöjärjestelmä

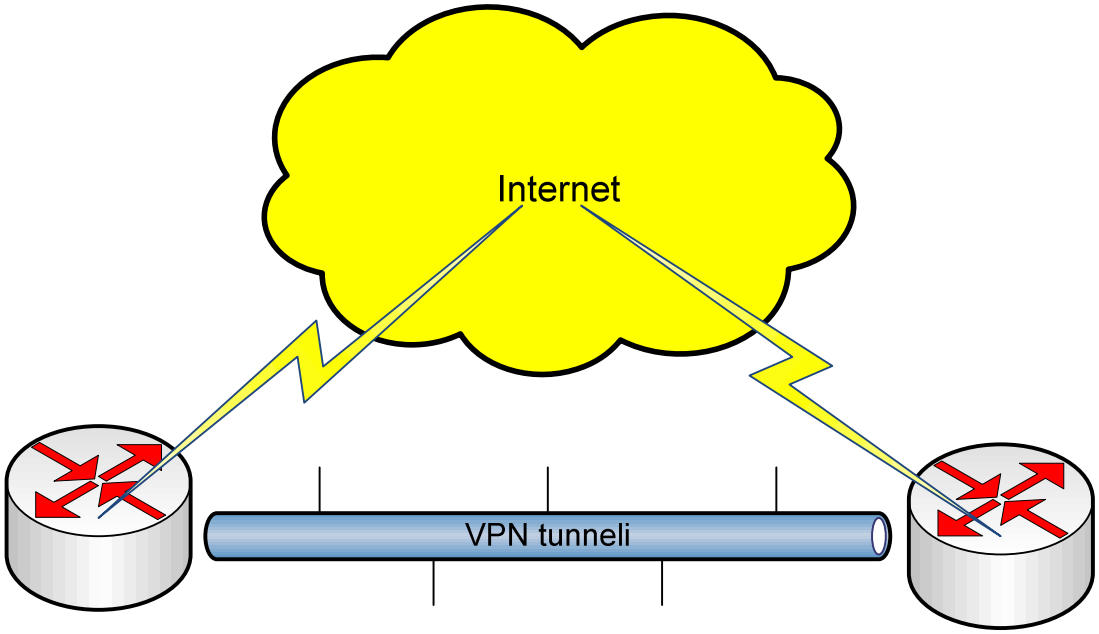
Junos on Juniper Networks -laitteiden käyttöjärjestelmä. Se on käytössä kaikissa Juniperin verkkolaitteissa riippumatta, onko kyseessä reititin vai tietoturvalaite. Tästä syystä laitteiden käyttäminen ja uusien laitteiden käyttöönotto on helppoa eikä jatkuvaa koulutusta ole tarvetta käydä. Junos -käyttöjärjestelmän päivitykset ovat myös yhteensopivia kaikille laitteille.

Junos on arkkitehtuuriltaan modulaarinen käyttöjärjestelmä, eli sen osat jakautuvat itsenäisiin moduuleihin. Tämä ratkaisu tekee Junoksen toiminnasta vakaan ja joustavan: yhden moduulin mahdollinen toimintavirhe ei kaada koko järjestelmää ja muut moduulit voivat jatkaa toimintaa. Modulaarisuus mahdollistaa myös paremman skaalatutuvuuden kun hallintaan ja välitykseen tarvittavat osat ovat eroteltuja. (Juniper Company Profile. 2011.)

# 5 VIRTUAL PRIVATE NETWORK-TEKNOLOGIA (VPN)

## 5.1 Yleisesti

Virtual Private Network (VPN) on nimensä mukaisesti teknologia jonka, avulla on mahdollista luoda virtuaalisia yksityisiä yhteyksiä julkisenverkon (*Internet*) yli. VPN:n käyttö on viimevuosina lisääntynyt verkkoyhteyksien ja etäkäytön mahdollisuuksien parannuttua ja siltä vaadittavat ominaisuudet ovat samalla kasvaneet. VPN:n avulla voidaan esimerkiksi muodostaa suojattu etäyhteys yritysten toimipisteiden välille ilman, että tietoa joudutaan kuljettamaan julkisessa verkossa. Siten että, se olisi kaikkien saatavilla, vaikka sen yli liikutaankin (ks. kuvio 3). Kiinteiden toimipisteiden yhdistämisen lisäksi VPN mahdollistaa etätyöntekijöiden pääsyn yrityksen verkkoon lähes mistä vain. VPN yhteyden muodostuksen apuna käytetään useita protokollia, jotka hoitavat yhteyden eri osa-alueita kapseloinnista tiedon koskemattomuuden ja eheyden takaamiseen.




KUVIO 3. Konteksti

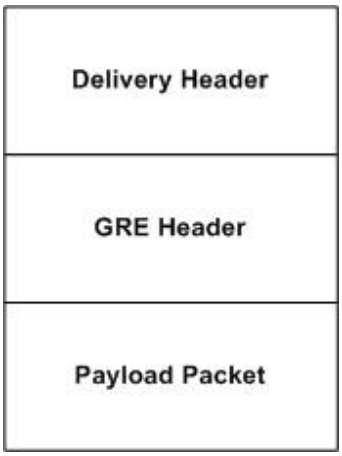
## 5.2 VPN-yhteyden käyttämät protokollat

### 5.2.1 Generic Rounting Encapsulation (GRE)

GRE on Cisco Systemsin kehittämä protokolla, joka on määritelty RFC 2784:ssä. Periaatteeltaan se on yksinkertainen kapselointiprotokolla. Sen avulla IP-paketti kapseloidaan GRE-paketiksi, joka tämän jälkeen kuljetetaan tunnelin läpi. Tunnelin päässä GRE-paketti puretaan alkuperäiseen muotoonsa ja lähetetään loppupäämäärään. Näin paketti voidaan kuljettaa verkossa ilman, että sitä käsitellään IP-pakettina.(RFC 2784)

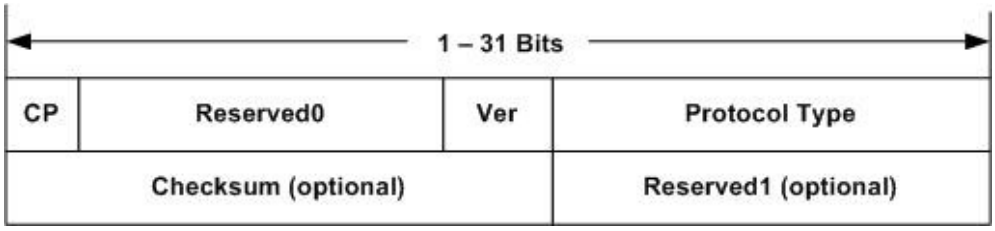
GRE on yleisesti käytetty VPN-tunneloinnin yhteydessä. GRE-tunnelit, joissa paketit kulkevat, ovat täysin tilattomia, eli tunneleiden alku - ja loppupäät eivät ole tietoisia toistensa tilasta.

GRE-paketin perusrakenne on varsin yksinkertainen. Koko alkuperäinen otsikkokenttä (hyötykuorma ja otsikkokenttä) kapseloidaan ensin GRE-paketiksi, joka sen jälkeen kapseloidaan kuljetusprotokollan käyttämään muotoon (ks. kuvio 4). (RFC 2784)




KUVIO 4. GRE-pakettirakenne (Implement IPv4 tunneling and Generic Routing Encapsulation)

GRE-paketin otsikkokenttä on myös pelkistetty (ks. kuvio 5). Se sisältää tiedot sisällä kuljetettavan protokollan tyypistä, esimerkiksi IP-paketin ollessa kyseessä kentän arvoksi merkitään *0x008*. Tarkistussumman avulla taataan hyötykuorman koskemattomuus. (RFC 2784)




KUVIO 5. RFC 2784 GRE –otsikkokenttä (Implement IPv4 tunneling and Generic Routing Encapsulation)

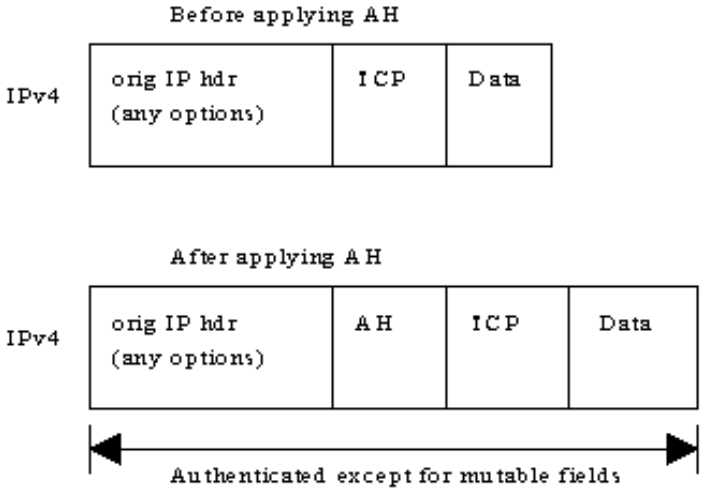
### 5.2.2 IP security protokolla (IPsec)

IPsec on IP/TCP protokollapaketti, jonka tarkoitus on suojata, salata ja hoitaa tunnistautuminen IP yhteyksissä. VPN yhteydessä IPsec tarjoaa kaiken tarvittavan, jotta yhteys pysyy vain haluttujen tahojen saatavilla ja muuttumattomana. IPsecin toiminta perustuu osapuolten välille luotaviin turva-assosiaatioihin (Security association, SA). SA on osapuolten välille muodostettu yksipuoleinen sopimus, jossa sovitaan yhteyden käytössä oleva salaus ja autentikointi. IPsec koostuu kolmesta osasta, joiden avulla edellä mainitut vaatimukset voidaan toteuttaa:

- Encapsulating Security Payload (ESP)
    - Salaa ja/tai autentikoi dataa
- Authentication Header (AH)
    - Paketin autentikointi
- Internet Key Exchange (IKE)
    - Luo avaimet ja määrittelee käytänteet, jotta yllämainitut ESP ja AH pystyvät toimimaan. (RFC 2401)

**Authentication Header**

Authentication Header (AH) on osa IPsec-protokollapakettia, ja se on määritelty RFC 2402:ssa. Sen tarkoitus on taata lähetettävän datan eheys ja koskemattomuus. AH:n avulla voidaan myös estää mahdolliset reply-hyökkäykset käyttäen liukuvaa ikkunaa lähetyksessä. AH toimii suoraan molempien IPv4 ja IPv6 päällä. AH:n toimita perustuu IP paketin tietojen perusteella laskettuun otsikkoon (ks. kuvio 6), jossa käytetään kaikkien IP-paketin kenttien tietoja pois lukien ne kentät, joiden arvo voi muuttua lähetyksen aikana. Vaikka AH takaakin tiedon koskemattomuuden ja eheyden, ei se varsinaisesti salaa sitä. Salausta varten tuleekin käyttää jotain toista protokollaa kuten ESP.(RFC 2402)

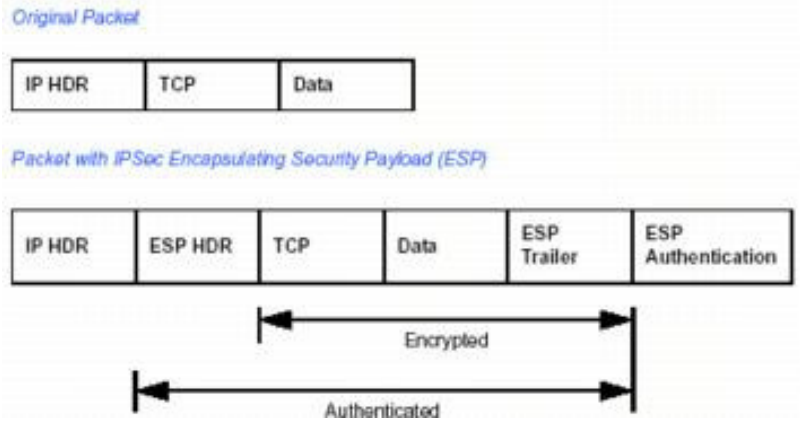



KUVIO 6. AH toimita esimerkki (IPSEC project)

Joissain tilanteissa AH:n tarjoama tiedon koskemattomuuden takaaminen on riittävä, eikä luottamuksellisuuden takaamista salauksen avulla ole tarvetta toteuttaa. Tämä voi esimerkiksi olla, kun liikennöidään tunnelissa kahden luotetun pisteen kesken, eikä tieto ole niin arkaluontoista. (RFC 2402)

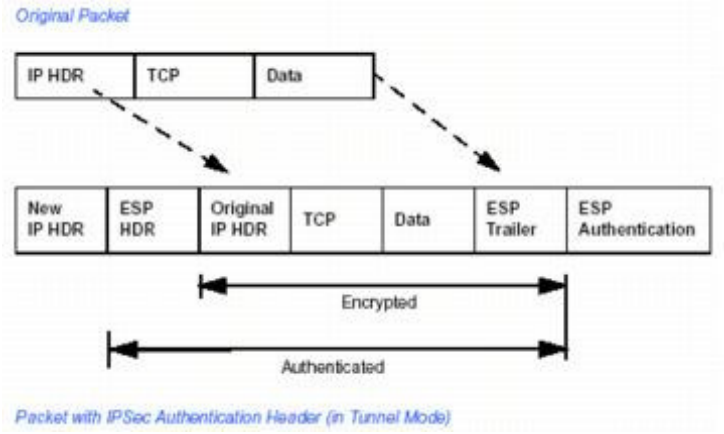
**Encapsulating Security Payload (ESP)**

ESP on IPsecin osa, joka salaa ja autentikoi datapaketit. Se on määritelty RFC 4303, jossa sitä kuvataan turvallisuuspalveluita tarjoavaksi IPv4 -ja IPv6-protokollaksi. ESP-otsikko lisätään IP-otsikon perään ennen kuljetusprotokollaa(ks. kuvio 7). (RFC 4303)




KUVIO 7. ESP otsikkokenttä (VPN Basics: Internet Protocol Security (IPSec).)

ESP voidaan käyttää myös tunneli tilassa, silloin ESP otsikko lisätään alkuperäisen IP-otsikon eteen ja sen eteen lisätään uusi IP-otsikko joka sisältää alkuperäisen lähde – ja kohdeosoitteen (ks. kuvio 8). (RFC 4303)




KUVIO 8. ESP otsikkokenttä tunnelitilassa (VPN Basics: Internet Protocol Security (IPSec).)

**Internet Key Exchange (IKE)**

IKE on osa IPsec-pakettia, sen tehtävä on neuvotella avainten vaihtoon tarvittavat kanavat ja parametrit. IKE koostuu käytännössä kolmesta toiminta vaiheesta;

- Phase 1 IKE
    - määrittelee avainten vaihtokanavan (ISAKMP SA) kahden laitteen välille
- Phase 2 IKE
    - määrittelee käytettävät data kanavat (IPsec SA)
- IPsec
    - Varsinainen IPsec, jossa data lähetetään käyttäen AH ja/tai ESP

IKEä voidaan toteuttaa käyttäen ennalta sovittuja avaimia, eli *manuaalisesti* asettamalla verkkolaitteille molemmissa yhteyden päissä salasana sekä SA:n ominaisuudet. Manuaalinen IKE on yksinkertaisin ratkaisu toteutuksen kannalta, mutta sen kannattavuutta tulee harkita isoissa yrityksissä avainten hallinnan kasvaessa.

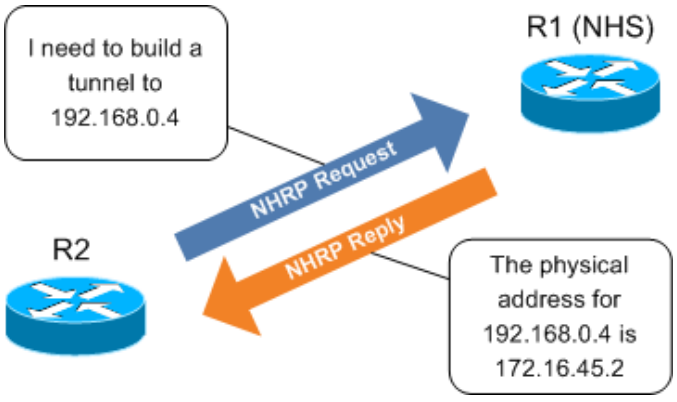
Toinen tapa toteuttaa avainten jakaminen on tehdä se automaattisesti käyttäen *AutoIKE*. AutoIKE:ssä toimitaan aluksi täysin samoin kuin manuaalisessa IKEssä, osapuolten välille määritellään SA sekä avain jolla yhteys voidaan luoda. Seuraavilla kerroilla kun avainten vaihdolle on tarvetta hoitaa IKE sen, käyttäen joko sertifikaattia tai ennalta jaettua avainta. (RFC 4306)

## 5.3 Next-Hop Resolution Protocol

NHRP on jo melko vanha protokolla, sen RFC 2332 määritelmä on julkaistu ensimmäisen kerran vuonna 1998. Sen tarkoitus on optimoida reittejä *non-broadcast multiple-access* (NBMA) verkoissa. Toisinsanoin NHRP tehtävä on optimoida lyhin reitti verkon sisällä kahden pisteen välille. Vaikka NHRP kehitettiin alun perin erilaisia tekniikoita kuten Frame-relay ja ATM ajatellen, pystytään sen toimintaa hyödyntämään myös esimerkiksi GRE -tunneleiden loogistenosoitteiden ”liittämiseen” fyysistenosoitteiden kanssa.(RFC 2332)

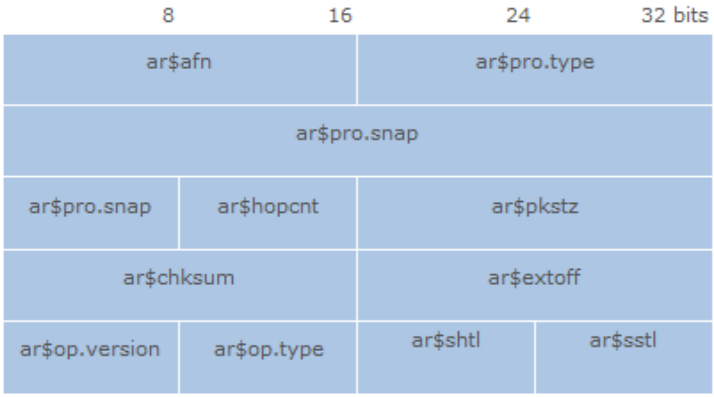
NHRP toiminta perustuu jokaisen laitteen toimimiseen joko *Next Hop Client(*NHC) tai *Next Hop Server*(NHS) tilassa. Jokaisen NHC tulee tietää ainakin yhden NHS:n IP-osoite, johon se pystyy lähettämään omat osoitetietonsa. NHS puolestaan toimii osoitetietokantana ja vastaa NHC:iden sille lähettämiin osoite kyselyihin. (PacketLife)

Esimerkiksi DMVPN yhteydessä HNRP mahdollistaa spoke-to-spoke väliset yhteydet selvittämällä halutun tunneli osoitteen, fyysisen osoitteen lähettämällä kyselyn NHS:lle joka sitten palauttaa halutun osoitteen, ks. kuvio 9 (PacketLife)




KUVIO 9. NHRP toiminta (PacketLife)

NHRP paketit koostuvat useasta kentästä, jotka kertovat tarkemmin viestin tarkoituksesta ja lähettäjästä (ks. kuvio 10);




KUVIO 10. NHRP otsikkokenttä

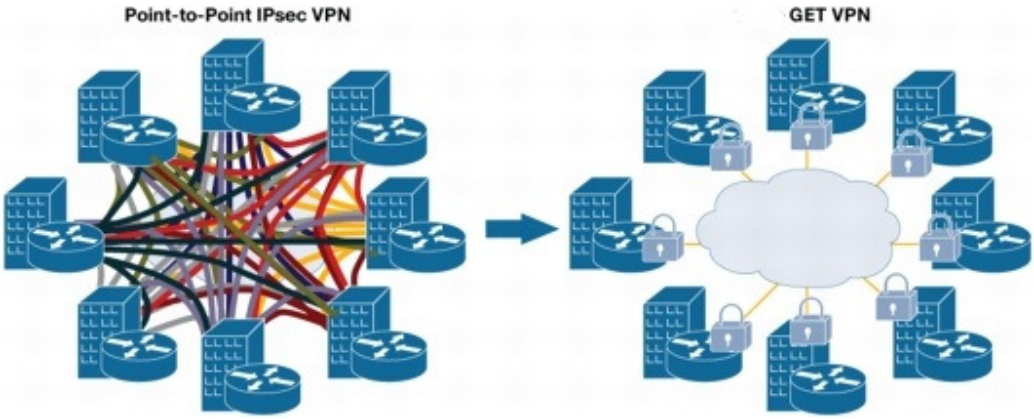
- ar$afn ilmoittaa käytössä olevan yhteyden tason, esim. layer 2
- ar$pro.type
- ar$pro.snap käytetään jos protokollan on koodatussa muodossa
- ar$hopcnt Sallittu hyppyjen määrä. Se määrittelee monenko NHS:n kautta paketti saa kulkea, ennenkö se hylätään
- ar$pkstz NHRP paketin kokonaispituus okteteissa
- ar$chcksum IP tarkastussumma koko NHRP paketille
- ar$extoff kertoo mahdollisista laajennuksista ja niiden sijainnista
- ar$op.version kertoo käytössä olevan kartoitus ja hallintaprotokollan version
- ar$op.type mikäli edellisen kentän arvona on 1, määrittelee kenttä viestin tyypin:
    1 NHRP selvitys kysely
    2 NHRP selvitys vastaus
    3 NHRP rekisteröinti pyyntö
    4 NHRP rekisteröinti vastaus
    5 NHRP puhdistus pyyntö
    6 NHRP puhdistus vastaus
    7 NHRP virhe ilmoitus
- ar$shtl alkuperäisen lähdeosoitteen tyyppi ja pituus selvennys
- ar£$sstl alkuperäisen lähdeosoitteen aliosoitteen tyyppi ja pituus selvennys

(RFC 2332)

# 6 GROUP ENCRYPTED TRANSPORT VPN

## 6.1 Yleistä

Group Encrypted Transport VPN (GET VPN tai Group VPN. *Tässä työssä käytetään yleislyhenteenä GET VPN*) on tunneliton VPN-tekniikka, joka perustuu *Group Domain of Interpretation* (GDOI) protokollaan. GET VPN alkuperäinen tarkoitus on helpottaa, suurten, toimipisteiden välisten, verkkojen konfiguraatiota ja ylläpitoa. Perinteisten kahden pisteen välille muodostettavien IPsec VPN-tunneleiden sijaan luodaan luotettujen pisteiden kesken ryhmä, joidenka välinen liikenne salataan käyttäen yhteisesti sovittuja avaimia ja salausasetuksia ks. kuvio 11 (Junos Security 10.2. 2011.)




KUVIO 11. perinteinen IPsec VPN verrattuna GET VPN (Junos Security 10.2. 2011.)

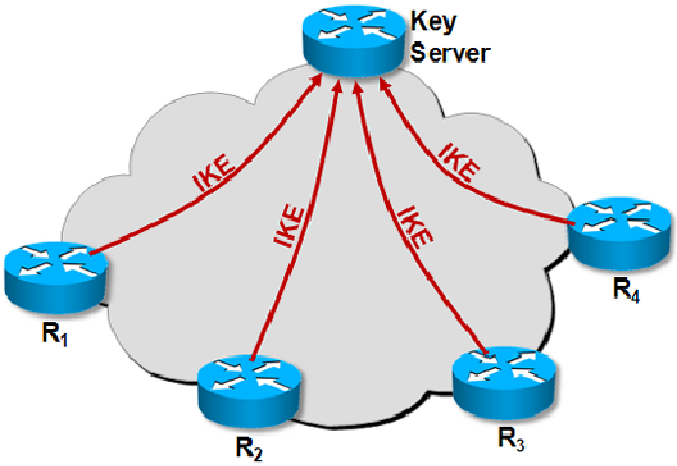
GET VPN hyöty saadaan etenkin suurissa verkoissa, joissa muuten tarvittaisiin moninkertainen määrä yhdyskäytäviä ja laitekapasiteettiä, jotta tiedon salaus olisi mahdollista IPsec VPN avulla. GET VPN mahdollistaa myös jo olemassa olevien reititys rakenteiden käytön, eikä tästä syystä tarvitse omaa reititys protokollaa. Myös mahdollisten palvelunlaatu (*Quality of Service,* QoS) palveluiden toteutus helpottuu.

Molemmat Group ja GET VPN ratkaisut voidaan toteuttaa joko operaattorin toimesta, tai asiakkaantoimesta. Konfiguraatiot tehdään asiakkaan ja operaattorin väliseen rajapintaan, joten molempien on mahdollista hallita laitteita riippuen palvelusopimuksen määritelmistä. (GET VPN design and implementation guide ja Junos Security 10.2. 2011.)

## 6.2 Group Domain of Interpretation (GDOI)

### 6.2.1 Yleistä

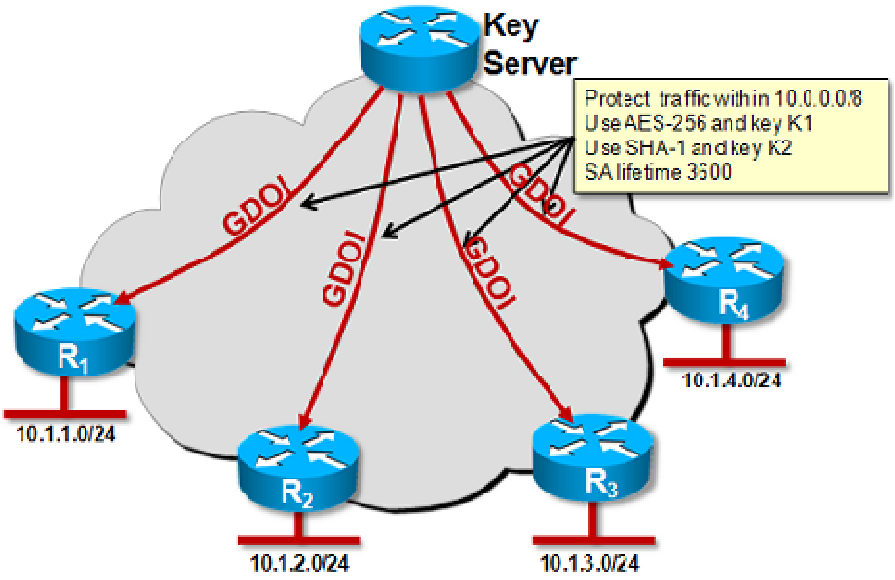
GDOI on RFC 3547:ssa määritelty protokolla, jonka tarkoitus on määritellä ja hallita ryhmäkohtaisia turva-assosiaatioita ja avaimia. GDOI toimii ryhmän jäsenten sekä avainpalvelimen välillä. Koska GDOI on itsessään ”tason 2” protokolla tulee sitä varten muodostaan ensin ISAKMP tason 1 turva-assosiaatio (ks. kuvio 12), jonka jälkeen varsinainen ryhmän muodostus prosessi voidaan suorittaa. (RFC 3547)




KUVIO 12. Avainpalvelimen toiminta tason 1. turva-assosiaatiossa (DESIGNING SITE-TO-SITE IPSEC VPNS)

### 6.2.2 Key Server (KS)

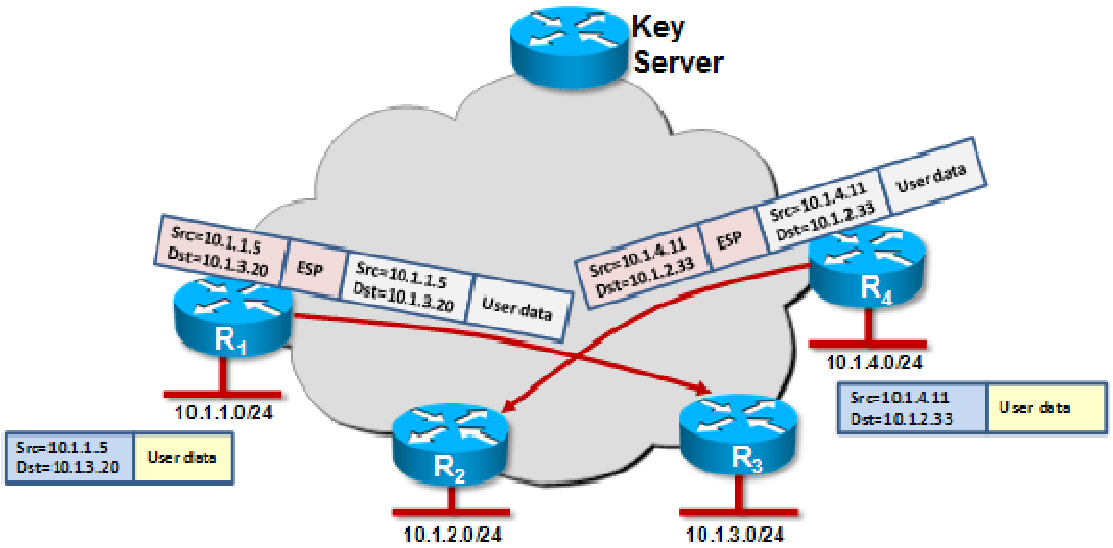
Key Serverin tarkoitus on määritellä ryhmänjäsenet sekä jakaa ryhmän sisäiset asetukset. KS määrittelee käytettävät salausavaimet, salauksen muodon sekä listan osoitteista ja protokollista joiden välinen liikenne ryhmän sisällä salataan (ks kuvio 13). Lisäksi KS huolehtii avainten uudelleen lähetyksestä. Vaikka GET VPN:n yksi päätavoite on vähentää tarvittavaa konfiguraatiomäärää, on jokainen GM ja sille tarkoitettu tason 1 avain määriteltävä erikseen KS:lle. (RFC 3547)




KUVIO 13. Avainpalvelimen toimintaperiaate (DESIGNING SITE-TO-SITE IPSEC VPNS)

### 6.2.3 Group member (GM)

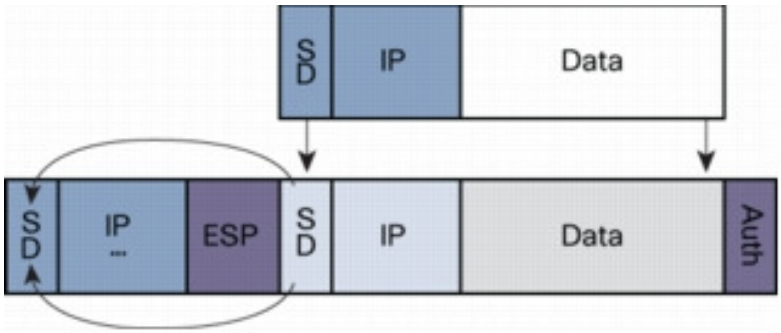
GM ei GDOI:n ollessa käytössä tarvitse tietää muuta kuin tason 1 SA tarvittava avain sekä KS osoite, jotta se voidaan liittää osaksi ryhmää. Kaikki muut tiedot se lataa KS:ltä Kun rekisteröityminen KS kanssa on suoritettu, pystyy GM toimimaan itsenäisesti ryhmän sisällä, käyttäen KS:ltä ladattuja tietoja (ks. kuvio 14). (RFC 3547)




KUVIO 14. Ryhmän jäsenten välisen liikennöinnin periaate (DESIGNING SITE-TO-SITE IPSEC VPNS)

### 6.2.4 Otsikkokenttä

GET VPN käyttää GM:ten välisessä liikenteessä samaa ESP kapselointia kuin perinteinen VPN. ESP käyttämät turvakäytänteet kuten kryptauksen muoto ja avainten elinikä määritellään GDOI avainpalvelimella. GET VPN yhteydessä ESP käytetään tunnelitilassa, jolloin se suojaa kokopaketin sekä IP otsikkokentän. Tunnelitilassa ESP lisää uuden otsikon kapseloinnin jälkeen, tämä otsikko sisältää myös lähde – ja kohdeosoitteet alkuperäisestä IP-otsikosta.




KUVIO 15. GET VPN otsikkokenttä

Koska alkuperäiset kohde ja lähdeosoitteet säilytetään, voidaan paketit lähettää käyttäen jo olemassa olevaa reititys infrastruktuuria. Eikä sitä varten tarvitse luoda omia reititys ratkaisuja. (GET VPN design and implementation guide. 2011. )

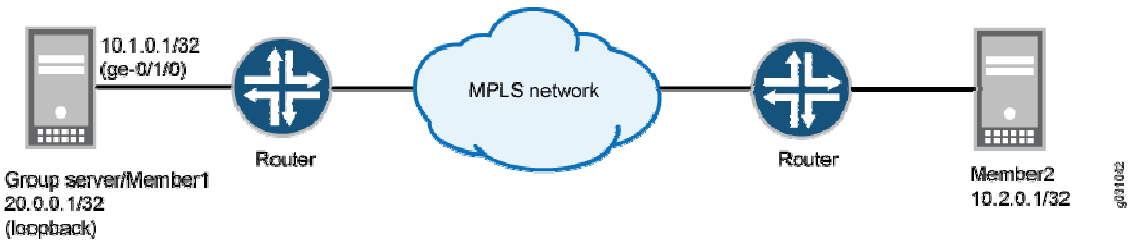
## 6.3 Junos Group Encrypted Transport VPN (Group VPN)

Juniperin Group Encrypted Transport VPN on nimeltään Group VPN. Group VPN perustuu vahvasti GDOI protokollaan ja seuraa vahvasti RFC 3547 määritelmään.

### 6.3.1 Server-Member Colocation

Juniper Group VPN tarjoaa perinteisen GDOI-ryhmä toteutuksen lisäksi Server-Member Colocation mallin. Tässä ratkaisussa erillisen KS:n sijaan, voidaan yhdellä ryhmän laitteella toteuttaa molemmat roolit.

Server-Member Co-location toiminta perustuu mahdollisuuteen käyttää samassa laitteessa sijaitsevaa toista rajapintaa hyväksi, kuvion 16. esimerkin mukaisesti MPLS verkkoon liittyvää rajapintaa voidaan käyttää GM -rajapintana samaan aikaan kun palvelin roolia ajetaan loopback rajapinnalla. Tämä on mahdollista, koska rekisteröityminen ja rekisteröinti tapahtuvat eri rajapinnoissa. (Junos Security 10.2. 2011)




KUVIO 16. Juniper Group VPN (Junos Security 10.2. 2011)

### 6.3.2 Vaatimukset

Tällä hetkellä Juniper Group VNP on tuettu SRX100, SRX200 ja SRX650 sarjan verkkolaitteilla sekä J-sarjan reitittimillä. Junos käyttöjärjestelmä version tulee olla 10.2r2 tai uudempi. (Junos Security 10.2. 2011)

## 6.4 Cisco Group Encrypted Transport VPN (GET VPN)

### 6.4.1 Yleistä

GET VPN on Cisco IOS tarjoama ryhmämalli VPN:lle. Sen perusperiaate on sama kuin Juniper Group VPN:ssä.

GET VPN toimii Key server ja group member roolejen avulla. Avain palvelin jakaa ryhmän jäsenille tarvittavat avaimet ja turva-assosiaatiot käyttäen Group Domain of Interpretation (GDOI) standardia, joka on määritelty RFC 3547:ssä.

### 6.4.2 Cooperative Key servers (COOP)

Cisco IOS tarjoaa mahdollisuuden käyttää useampaa kuin yhtä avainpalvelinta, niin sanotussa COOP mallissa. COOP etuna on sen tuoma reduntanttisuus, sekä mahdollisuus jakaa liikennettä joko maantieteellisesti tai vain tasapainottaa laitteiden rasitusta. Jokainen KS määritellään prioriteetti arvon perusteella, mikäli prioriteetti arvoa ei ole asetettu, käytetään korkeimman IP osoitteen omaavaa KS ensisijaisena. KS:t viestivät keskenään ajastetuin viestein, jotta mahdollisen yhteyden katkeamisen jälkeen uusi ensijainen KS voidaan valita. GM osalta COOP käytöllä ei ole suurta vaikutusta, vaan jokaiselle GM voidaan antaa useamman kuin yhden KS osoite. (GET VPN design and implementation guide. 2011.)

### 6.4.3 Vaatimukset

Tällä hetkellä Cisco suosittelee käytettäväksi IOS versio 12.4(15)T8. Laitteet jotka tukevat GET VPN ovat listattuna alla olevassa taulukossa.

| Product Line | GM | KS |
|---|---|---|
| **Software** | Not recommended | Not recommended |
| **87x (onboard)** | Yes | Not recommended |
| **1800/1841 (onboard, AIM/SSL)** | Yes | Yes for 1841 |
| **2800 (onboard, AIM/SSL)** | Yes | Yes |
| **3800 (onboard, AIM/SSL)** | Yes | Yes |
| **7200 NPEG1/NPEG2, VAM2+** | Yes | Yes |
| **7301 VAM2+** | Yes | Yes |
| **7201 VAM2+** | Yes | Yes |
| **7200 NPEG2, VSA** | Yes[1] | Yes* |
| **ASR** | Yes | No |
| **6500 VPN-SPA** | No | No |

### 6.4.4 Cisco ja Juniper yhteensopivuus

Vaikka molemmat Cisco ja Juniper GETVPN/ Group VPN ratkaisut perustuvat samaa standardisoituun GDOI protokollaan asettaa laitevalmistajien käyttöjärjestelmien eroavuudet tiettyjä rajoituksia. Selkein ero on Junos ja Cisco IOS ero IKE viestien kuittauksessa ja uudelleen lähettämisessä. Tästä johtuen kun Juniper ja Cisco laitteiden halutaan toimivan osana samaa ryhmää, tulee avainpalvelin olla sijoitettuna Ciscon laitteessa. (GET VPN design and implementation guide. 2011. ja Junos Security 10.2. 2011)

| MEMBER | SERVER | REKEY MECHANISM | ANTI-REPLAY PROTECTION | INTEROP |
|---|---|---|---|---|
| SRX Series | CISCO ISR | PULL | NO | YES |
| SRX Series | CISCO ISR | PULL | YES | NO |
| SRX Series | CISCO ISR | PUSH | NO/YES | NO |
| CISCO ISR | SRX Series | PULL/PUSH | NO/YES | NO |

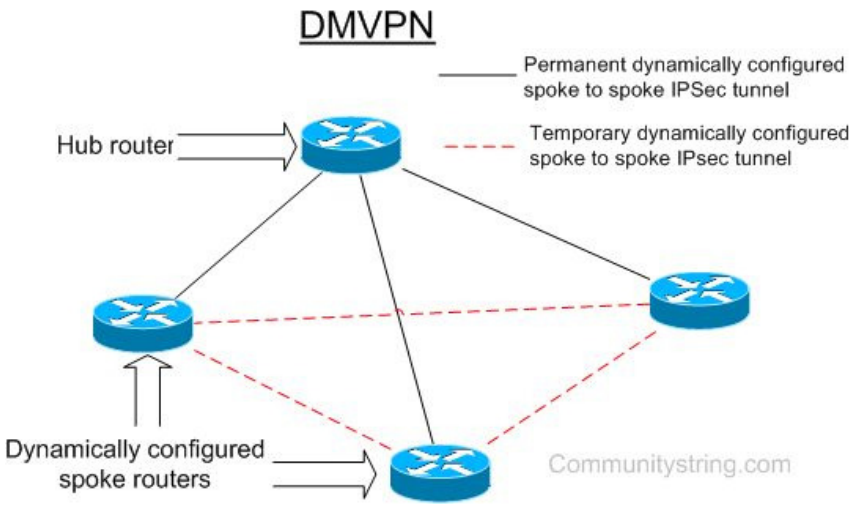
Toinen rajoite on molempien laitevalmistajien lisäominaisuuksien toimivuudessa. Koska Juniper laite ei voi toimia avainpalvelimena Cisco laitteiden ollessa mukana ryhmä ei Juniper Server-Member yhteissijaintia voida käyttää. Cisco COOP toiminto taas on mahdollista ottaa käyttöön vaikka osa ryhmänjäsenistä olisikin Juniper valmistamia laitteita. Tällöin COOP saatu hyöty on kuitenkin huomattavasti vähäisempi, johtuen siitä ettei Juniper laitteille voida asettaa useampaa kuin yhtä käytettävää avainpalvelinta.

(GET VPN design and implementation guide. 2011. ja Junos Security 10.2. 2011)

# 7 DYNAMIC MULTIPOINT VIRTUAL PRIVATE NETWORK (DMVPN)

## 7.1 DMVPN yleisesti

*Dynamic Multipoint Virtual Private Network (DMVPN)* on Cisco IOS-reitittimien tarjoama konfiguraatio, joka mahdollistaa useiden pisteiden välisen yksityisten IPSec-tunneleiden muodostamisen ilman tarvetta konfiguroita jokaista tunnelia staattisesti. DMVPN:n avulla saavutetaan joustava ja huomattavasti skaalatutuvampi järjestely kuin aikaisemmilla IPSec VPN-konfiguraatioilla. DMVPN:n ideana on käyttää yhtä tai useampaa hub-reitintä, joka on esimerkiksi yrityksen pääkonttorissa. Hub reititin on verkon ainoa laite joka tarvitsee staattisen IP-osoitteen. Tähän hub-reitittimeen liitetään spoke-reitittimiä, jotka voivat olla esim. muiden toimipisteiden verkkolaitteita. Spoke-reitittimet konfiguroidaan siten, että niihin määritellään vain hub reitittimen IP-osoite, muiden spoke-reitittimien osoitteiden saamisesta vastaa NHRP. Sen jälkeen kun perus konfiguraatio on tehty, pystyvät spoke-reitittimet muodostamaan tunneliyhteyden myös keskenään. Uuden laitteen lisääminen tämän jälkeen on helppoa, sen IP-osoitetta ei tarvitse konfiguroita jokaiselle verkon laitteelle, vaan riittää kun se on hub-reitittimen tiedossa. (Dynamic Multipoint VPN (DMVPN) Design Guide 2011)




KUVIO 17. DMVPN toimintamalli (comminitystring.com)

DMVPN:n ollessa käytössä tunnelit luodaan tarpeen mukaan ja ne voidaan muodostaa spoke-reitittimen ja hub-reitittimen välille, tai kahden spoke-reitittimen välille. DMVPN koostuu neljästä pää tekniikasta jotka mahdollistavat sen toiminnan

- GRE hoitaa tunneli kapseloinnin
- IPSec takaa yhteyden salassapidon ja tiedon eheyden
- NHRP liittää laitteiden fyysiset osoitteet niitä vastaaviin tunneli osoitteisiin ja mahdollistaa yhteydet *spoke-to-spoke* välillä sekä skaalautuvuuden.
- reititys protokollasta esim. RIP, OSPF, EIGRF tai BGP.

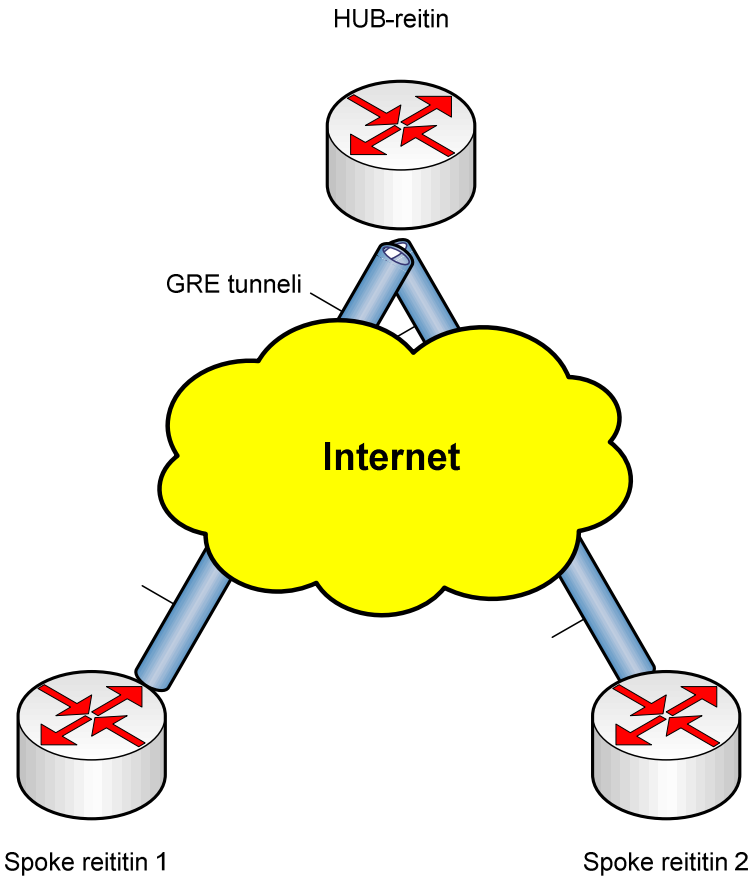
DMVPN-verkon muodostuminen koostuu kolmesta vaiheesta (*phase*) joiden aikana koko verkon kattava tunnelin muodostus saadaan aikaan, mahdollisimman yksinkertaisin konfiguraatioin. (Dynamic Multipoint VPN (DMVPN) Design Guide 2011)

Koska DMVPN reitittää itse alueensa sisällä, sekä mahdollistaa dynaamisten verkko-osoitteiden käyttämisen, tarjoaa se selvästi paremmat mahdollisuudet kuluttaja laitteiden liittämiseen osaksi ryhmää, verrattuna GDOI ratkaisuihin. Rajoittavaksi tekijäksi muodostuu sen vaatimus pelkistä Cisco laitteista. (Dynamic Multipoint VPN (DMVPN) Design Guide 2011)

## 7.2 DMVPN muodostus

### 7.2.1 DMVPN muodostus vaihe yksi

Ensimmäisessä vaiheessa DMVPN-yhteyden muodostuksessa luodaan *point-to-point* GRE yhteydet spoke-reitittimien ja hub-reitittimen välille. Hub-reitittimessä käytetään *multi-point GRE* (mGRE), jotta kaikki spoke-reitittimiin saapuvat tunneli-yhteydet saadaan samaan rajapintaan(*interface*). Spoke-reitittimet lähettävät NHRP-tietonsa hub -reitimille joka myös toimii NHS:nä. Ensimmäisen vaiheen tarkoitus on yksinkertaistaa tarvittavien konfiguraatioiden määrää hub-reittimellä verrattuna muihin VPN-ratkaisuihin. (Pepelnjak I, DMVPN: From Basics to Scalable Networks 2011)




KUVIO 18. DMVPN ensimmäinen vaihe

### 7.2.2 DMVPN muodostus vaihe kaksi

Toisessa vaiheessa DMVPN verkko muodostaa mGRE tunneloinnin kaikkien mukana olevien laitteiden kesken, jolloin *spoke-to-spoke* tunnelin muodostaminen on mahdollista. Kaikki spoke reittimet rekisteröivät tunneli osoitteensa NHS kanssa. Kun spoke haluaa saada yhteyden toiseen spokeen mGRE välityksellä, lähettää se NHRP kyselyn hub/NHS:lle joka vastaa kyselyyn ja lähettää halutun spoke-reitittimen tunneli osoitteen. (Pepelnjak I, DMVPN: From Basics to Scalable Networks 2011)

### 7.2.3 DMVPN muodostus vaihe kolme

Kolmannen vaiheen selkein päämäärä on lisätä verkon skaalautuvuutta ja keventää Hub-reitittimelle muodostuvaa taakkaa verkon kasvaessa. Kolmannessa vaiheessa hub-reititin ei ole enää ainut NHRP informaation lähde, vaan kaikki *spoke*-reitittimet ovat mukana tiedon jakamisessa.

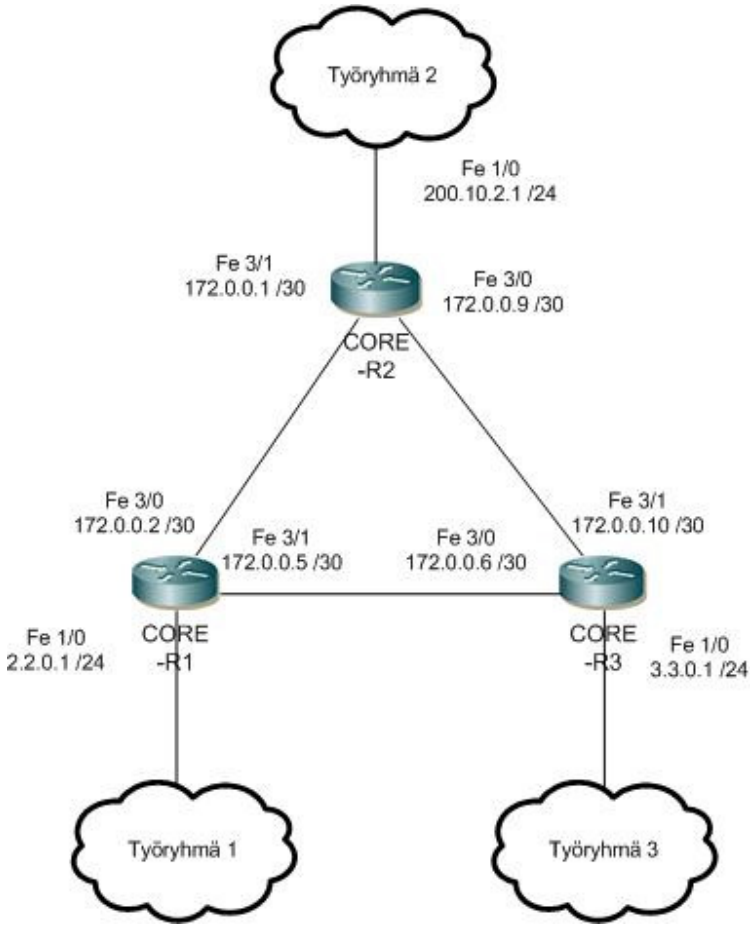
Tämä on mahdollista hyödyntäen *Cisco Express Forwarding* (CEF) tekniikkaa. CEF on Cisco reittitimien OSI-mallin tasolla 3 toimiva teknologia joka parantaa verkon suoritus kykyä. CEFin toiminta perustuu *Forwarding Information Base* (FIB). FIB tietokanta on periaatteeltaan monien reititysprotokollien reititystaulua vastaava, se pitää yllä *next-hop* osoitteiden IP-taulukkoa. (Pepelnjak I, DMVPN: From Basics to Scalable Networks 2011)

# 8 KÄYTÄNNÖN TOTEUTUS

## 8.1 DMVPN laitteisto ja topologia

### 8.1.1 Internet

DMVPN:n testausta varten rakennettiin yksinkertainen Internetiä simuloivan ympäristö. Internet toteutettiin käyttäen kolmea SpiderNetistä löytyvää Cisco Core-reititintä. Kuvi-osta 19. nähdään Internetin topologia sekä työryhmien sijainti.




KUVIO 19. DMVPN-topologia

### 8.1.2 Työryhmät

Koska työryhmissä ei käytetty kuin niiden reuna reitittimiä (WGx-R1), jotka olivat suoraan kytkettyinä ”Internetiin” ei työryhmien topologioiden tarkemmalle kuvaamiselle ole tarvetta.

### 8.1.3 Internetin konfigurointi

Internet laitteiden konfigurointi pidettiin mahdollisimman yksinkertaisena. Koska tarkoitus oli vain simuloida Internetiä, pidettiin reitittimien määrä myös pienenä ja toteutus tehtiin kolmella reitittimellä. Reititys protokollaksi valittiin *Open Shortest Path First* (OSPF) – protokolla. OSPF tarvitsee toimiakseen reitittimen ID:n (Router ID) joka tässä tapauksessa saatiin *loopback 0* rajapinnalta. Yhteys ”Internetin” ja työryhmien WG1 ja WG3 toteutettiin käyttäen *Dynamic Host Configuration Protocol (DHCP).* Internetin ja WG2 välinen yhteys toteutettiin staattisella osoitteella.

Esimerkkinä Core-R1 konfiguraatiot:

```
ip dhcp excluded-address 2.2.0.1 2.2.0.10
!
ip dhcp pool WG1
  network 2.2.0.0 255.255.255.0
  default-router 2.2.0.1
!
interface Loopback0
 ip address 130.0.1.2 255.255.255.252
!
interface FastEthernet1/0
 description Link to WG1-R1
 ip address 2.2.0.1 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet3/0
 description Link To CORE-R2
 no switchport
 ip address 172.0.0.2 255.255.255.252
```

```
!
interface FastEthernet3/1
 description Link to CORE-R3
 no switchport
 ip address 172.0.0.5 255.255.255.252
!
router ospf 1
 log-adjacency-changes
 redistribute static metric-type 1
 network 2.2.0.0 0.0.0.255 area 0
 network 130.0.1.0 0.0.0.3 area 0
 network 172.0.0.0 0.0.0.3 area 0
 network 172.0.0.4 0.0.0.3 area 0
 network 172.0.0.8 0.0.0.3 area 0
```

Yläpuolella olevassa konfiguraatiossa määriteltiin ensin IP-osoitteet joita DHCP:n ei haluttu antavan, tämän jälkeen määriteltiin osoite DHCP osoiteavaruus WG1 ja sen käyttämät IP-osoitteet ja aliverkkopeite. Lisäksi annettiin oletusyhdyskäytävän osoite. Seuraavaksi asetettiin käytetyille rajapinnoille IP-osoitteet sekä aliverkko peitteet, lopuksi konfiguroitiin OSPF reititys.

## 8.2 Työryhmä reitittimien konfigurointi

### 8.2.1 Työryhmä 2 reitittimen konfigurointi

Työryhmä 2 reititin toimi konfiguraatiossa hub laitteena, ja siitä syystä sen konfiguraatio erosi muiden työryhmäreitittimen konfiguraatiosta. DMVPN tarvitsee toimiakseen jonkin reititysprotokollan, työssä päätettiin käyttää myös DMVPN sisäiseen reititykseen OSPF protokollaa sen yksinkertaisen konfiguraation takia.

```
interface GigabitEthernet0/0
 description link to CORE-R2
 ip address 200.10.2.2 255.255.255.0
 duplex auto
 speed auto
!
interface Loopback0
 ip address 130.0.10.1 255.255.255.252
!
router ospf 1
 log-adjacency-changes
 network 200.10.2.0 0.0.0.255 area 0
```

Yläpuolella olevassa konfiguraatiossa määriteltiin reitittimen perus asetukset. Aluksi määriteltiin IP-osoite rajapinnalle, joka on yhteydessä ”Internettiin” sekä Loopback0 rajapinnalle. Seuraavaksi määriteltiin OSPF reititys, jotta yhteys Internetin yli saatiin tehtyä.

Alapuolella on varsinainen DMVPN konfiguraatio, aluksi luotiin ISAKMP käytänne tasoa 1 varten ja valittiin käytettäväksi ennalta sovittu salausavain komennolla *authentication pre-share*. Komento *crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0* määritteli tason 1 salasanaksi cisco123 ja sen olevan käytössä kaikkien osoitteiden kansa. Lopuksi annettu *crypto isakmp invalid-spi-recovery* komento aktivoi moduulin, jossa IKE ilmoitta vastaanottajalle mahdollisesta virheestä SA luonnissa.

```
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0
crypto isakmp invalid-spi-recovery
!
```

Tämän jälkeen luotiin tason *2* käytänteet datan salausta varten ja IPSec profiili jota käytetään GRE tunneleiden kanssa. Aluksi määriteltiin *transform-set* nimellä dmvpn123 ja sille käytettäväksi *esp-3des* salaus-algoritmi sekä *md5* tiiviste-algoritmi. Tämän avulla datan kuljetuksessa käytetyt asetukset voidaan sitoa itse ipsec profiiliin.

```
!
crypto ipsec transform-set dmvpn123 esp-3des esp-md5-hmac
 mode transport
!
crypto ipsec profile dmvpn
 set transform-set dmvpn123
```

Lopuksi luotiin IPsec profiili nimellä *dmvpn* ja liitettiin siihen, sitä ennen luotu *transform-set dmvpn123*.

Seuraavaksi luotiin asetukset dynaamisesti muodostettavia GRE tunneleita varten. Tunneleita varten määriteltiin *Next Hop Resolving Protokollan* tarvitsemat asetukset sekä tunneleissa käytettävä reititys protokolla. Lisäksi määriteltiin fyysinen rajapinta tunneleita varten ja tunneleiden käyttämä tila eli *gre multipoint.* Lopuksi liitettiin aikaisemmin luotu ipsec profiili tunneleihin ja konfiguroitiin ospf reititys tunneleiden väliseen liikennöintiin.

```
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.1 255.255.255.0
 no ip redirects
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map multicast dynamic
 ip nhrp network-id 100000
 ip nhrp holdtime 600
 ip ospf network broadcast
 ip ospf priority 2
 delay 1000
 keepalive 5 4
 tunnel source GigabitEthernet0/0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile dmvpn
!
router ospf 2
 log-adjacency-changes
 network 10.0.0.0 0.0.0.255 area 0
!
```

### 8.2.2 Työryhmä 1 ja 3 reitittimien konfiguraatio

Työryhmä reitittimet 1 ja 3 toimivat hub tilassa, ja niiden konfiguraatiot vastaavat täysin toisiaan pois lukien rajapintojen ip-osoitteita. Alapuolella työryhmä 1 reitittimen konfiguraatiot. Aluksi määriteltiin Cisco core yhteydessä oleva rajapinta, joka otti ip osoitteensa dhcp:ltä

```
!
interface GigabitEthernet0/0
 description Link to CORE-R1
 ip address dhcp
 duplex auto
 speed auto
!
```

Seuraavaksi määriteltiin DMVPN käyttämät IKE ja IPSec parametri, jotka toteutettiin hub-reitittimellä tehtyjen konfiguraatioiden mukaisesti.

```
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0
!
crypto ipsec transform-set dmvpn123 esp-3des esp-md5-hmac
 mode transport
!
crypto ipsec profile dmvpn
 set transform-set dmvpn123
!
```

Seuraavaksi konfiguroitiin tunneli asetukset, nhrp asetukset sekä sidottiin ipsec profiili käytettäväksi tunnelin kanssa.

```
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.2 255.255.255.0
 no ip redirects
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map 10.0.0.1 200.10.2.2
 ip nhrp map multicast 200.10.2.2
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 ip ospf network broadcast
 ip ospf priority 0
 delay 1000
 keepalive 5 4
 tunnel source GigabitEthernet0/0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile dmvpn
!
```

Lopuksi määriteltiin DMVPN sisällä käytettävän ospf reitityksen asetukset.

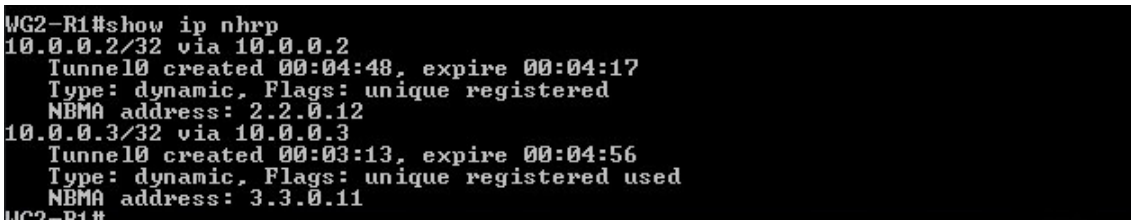
```
!
router ospf 2
 log-adjacency-changes
 network 10.0.0.0 0.0.0.255 area 0
!
```

Työryhmä 3:n konfiguraatiot löytyvät liitteestä 6.

### 8.2.3 DMVPN todennus

DMVPN konfiguraatio todennus suoritettiin käyttämällä Cisco IOS *show-komentoja* käyttämällä, sekä ping ja traceroute komennoilla.

Aluksi katsottiin hub-reitittimen tiedot DMVPN osalta, tämä tehtiin käyttäen *show ip nhrp, show dmvpn* sekä *show crypto isakmp sa* komentoja. Aluksi katsottiin että nhrp on toiminnassa ja löytää molemmat spoke reitittimet (ks. kuvio 20)

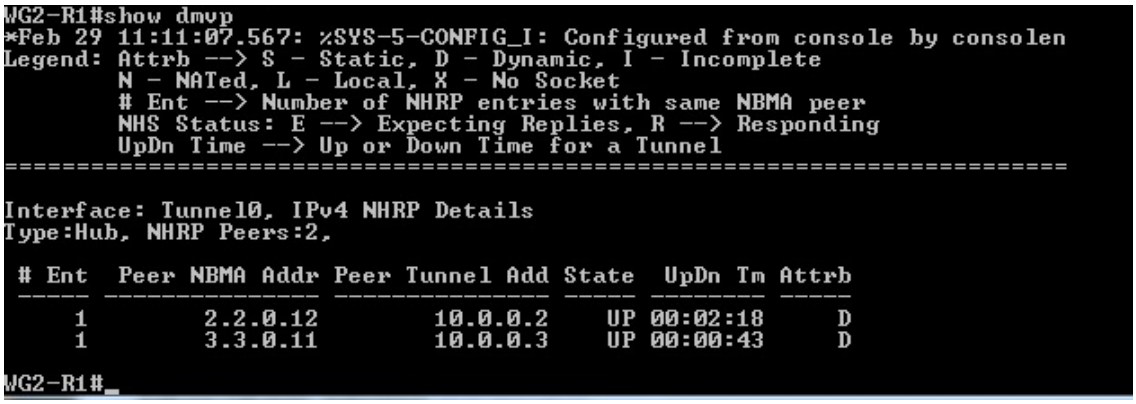


KUVIO 20. Hub-reitittimen nhrp taulu

Tämän jälkeen katsottiin muiden show komentojen antamat tulokset. Kuvion 21. *Show crypto isakmp sa* komennossa nähdään kuinka molemmat spoke reitittimet ovat muodostaneet turva-assosiaation hub:in kanssa, joka kuvassa on kohdeosoitteella (lyhenne dst) 200.10.2.2

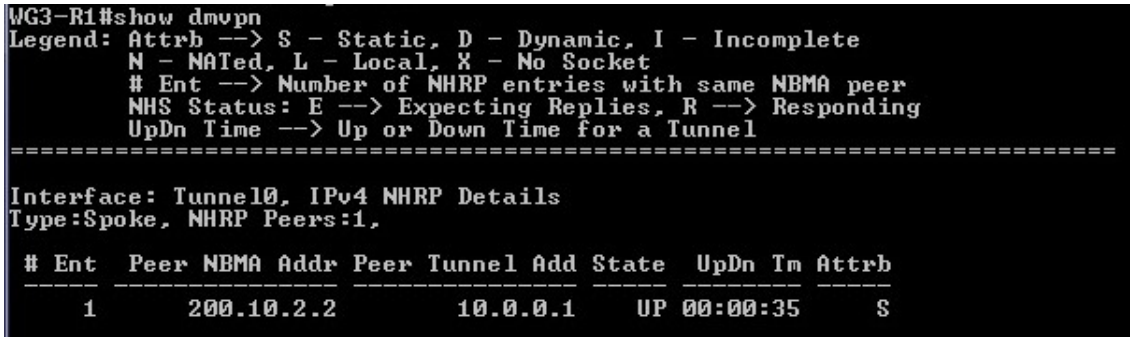
KUVIO 21. Hub-reitittimen isakmp turva-assosiaatio taulu

Myös *show dmvpn* komento antoi halutun tuloksen, kuviosta 22. nähdään molempien spoke reitittimien tunneliosoite, tunnelin olevan käytettävissä sekä hub kanssa muodostetun tunneli tyypin.




KUVIO 22. Hub-reitittimen DMVPN taulu

Kun toimivuus hub reitittimellä oli varmistettu, testattiin tunnelin muodostus spoke reitittimien kesken. Aluksi työryhmä 3 reitittimelle annettiin *show dmvpn* ja *show ip nhrp* komennot jotta nähtiin alkutilanne ennen nhrp:n tekemää osoite kyselyä. Kuvioissa 23. ja 24. nähdään kuinka reititin tietää vain hub reitittimen tunneli -ja fyysisenosoitteen alkutilassa.

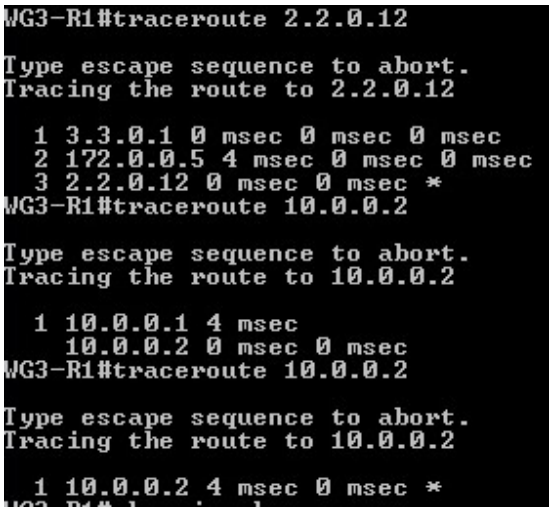



KUVIO 23. Työryhmä 3 reitittimen lähtötilanne, DMVPN


KUVIO 24. Työryhmä 3 reitittimen lähtötilanne, nhrp

Tämän jälkeen suoritettiin *traceroute* komento lähtien työnryhmä 3 reitittimeltä, työ ryhmä 1 reitittimelle. Aluksi testattiin paketin kulkema reitti kun kysely tehtiin käyttäen fyysistä osoitetta.

Kuviossa 25. nähdään kuinka fyysistä osoitetta käyttäen paketti kulkee ensin kahden Cisco core reitittimen kautta (kuvassa osoitteet 3.3.0.1 ja 172.0.0.5) ja saapuu sitten kohdelaitteelle. Seuraava kysely tehdään työryhmä 1. reitittimen tunneliosoitteeseen 10.0.0.2. Tällä kertaa tehdään ensin kysely Hub reitittimelle osoitteessa 10.0.0.1, josta saadaan osoitetiedot ja päästään työryhmä 1 reitittimelle. Kolmannessa kyselyssä laite on jo saanut NHRP tiedot ja pystyy muodostamaan tunnelin suoraan työryhmä 1 reitittimen kanssa ilman tarvetta kulkea hub reitittimen läpi.




KUVIO 25. Työryhmä 3 reitittimen traceroute kyselyt

Tämän jälkeen tarkasteltiin vielä uudelleen työryhmä 3 reitittimen NHRP ja DMVPN tauluja show komentoja käyttäen. Kuvioista 26. ja 27. nähdään kuinka työryhmä 1 reitittimen tiedot on lisätty työryhmä 3 laitteelle ja DMVPN toiminta dynaamisten tunneleiden osalta on mahdollista.

```
WG3-R1#show ip nhrp
10.0.0.1/32 via 10.0.0.1
   Tunnel0 created 00:15:24, never expire
   Type: static, Flags: used
   NBMA address: 200.10.2.2
10.0.0.2/32 via 10.0.0.2
   Tunnel0 created 00:00:58, expire 00:04:01
   Type: dynamic, Flags: router used
   NBMA address: 2.2.0.12
```

KUVIO 26. Työryhmä 3 reitittimen nhrp taulu traceroute kyselyiden jälkeen

```
WG3-R1#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================

Interface: Tunnel0, IPv4 NHRP Details
Type:Spoke, NHRP Peers:2,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
     1       200.10.2.2          10.0.0.1    UP 00:15:18     S
     1         2.2.0.12          10.0.0.2    UP 00:01:02     D
```

KUVIO 27. Työryhmä 3 reitittimen DMVPN taulu traceroute kyselyiden jälkeen

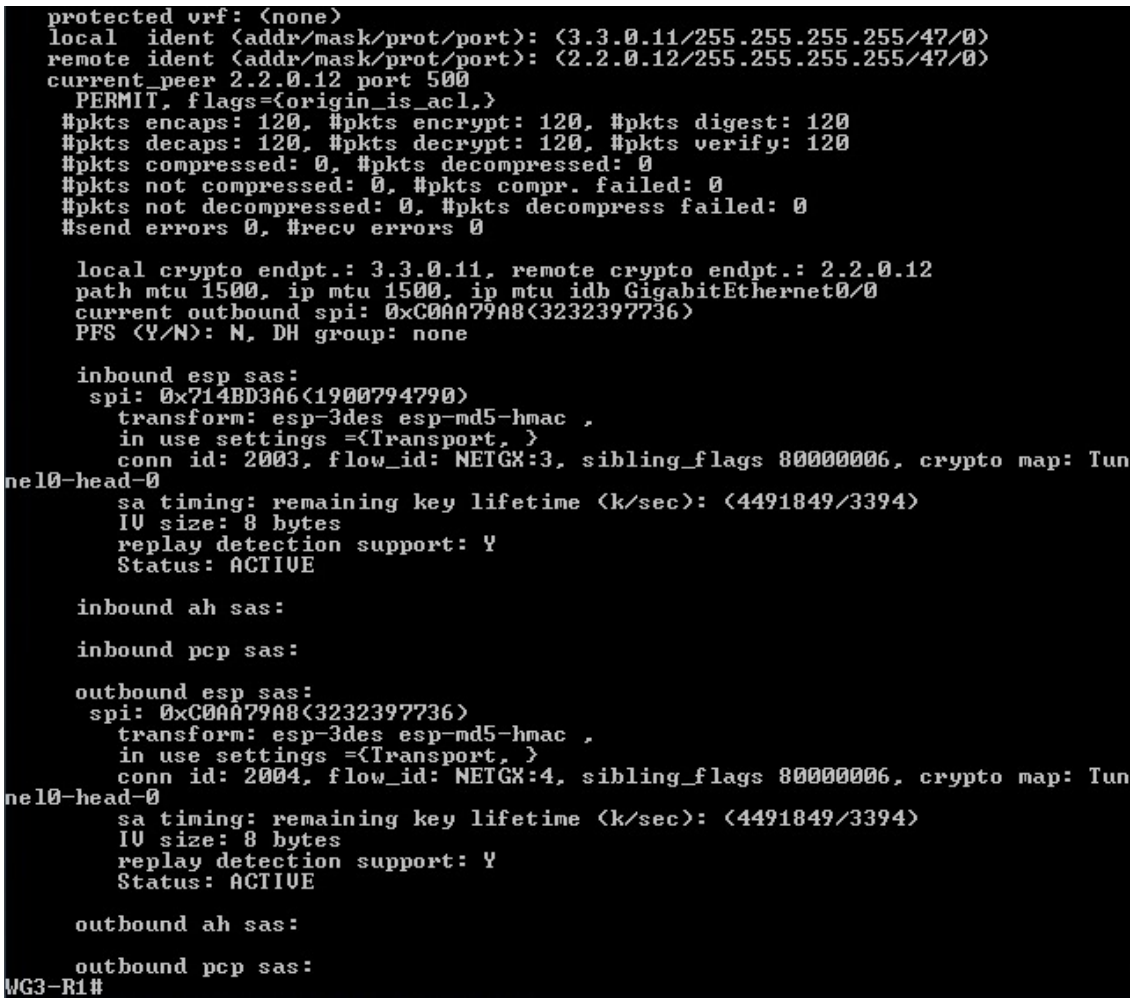
Jotta DMVPN toiminta voitiin vielä varmistaa, tarkasteltiin salauksen toimivuutta, käyttäen *show crypto engine* ja *show crypto ipsec* komentoja. Alla olevassa kuviossa nähdään työryhmä 3 reitittimen ja hub reitittimen välisen ipsec turva-assosiaation parametrit sekä tilastot.

Kuviosta 28. nähdään käytetty tunneli, osapuolten fyysiset osoitteet, salatun liikenteen määrä molempiin suuntiin, lähetettävän ja vastaanotettavan liikenteen ESP asetukset sekä tila.

```
WG3-R1#show crypto ipsec sa

interface: Tunnel0
    Crypto map tag: Tunnel0-head-0, local addr 3.3.0.11

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (3.3.0.11/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (200.10.2.2/255.255.255.255/47/0)
   current_peer 200.10.2.2 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 159, #pkts encrypt: 159, #pkts digest: 159
    #pkts decaps: 133, #pkts decrypt: 133, #pkts verify: 133
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 4, #recv errors 0

     local crypto endpt.: 3.3.0.11, remote crypto endpt.: 200.10.2.2
     path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/0
     current outbound spi: 0xBB65D732(3144013618)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
      spi: 0x634AA5E9(1665836521)
        transform: esp-3des esp-md5-hmac ,
        in use settings ={Transport, }
        conn id: 2001, flow_id: NETGX:1, sibling_flags 80000006, crypto map: Tun
nel0-head-0
        sa timing: remaining key lifetime (k/sec): (4582350/2537)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0xBB65D732(3144013618)
        transform: esp-3des esp-md5-hmac ,
        in use settings ={Transport, }
        conn id: 2002, flow_id: NETGX:2, sibling_flags 80000006, crypto map: Tun
nel0-head-0
        sa timing: remaining key lifetime (k/sec): (4582347/2537)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE
```

KUVIO 28. Työryhmä 3 ja Hub reitittimen välinen turva-assosiaatio

Kuviossa 29. voidaan nähdä työryhmä 3 ja työryhmä 1 reitittimen välille luotu ipsec turva-assosiaatio.




KUVIO 29. Työryhmä 3 ja työryhmä 1 välinen turva-assosiaatio

Lopuksi testattiin *show crypto engine connections active* komentoa jolla nähtiin aktiiviset käytössä olevat salaukset. Komento annettiin työryhmä 3 reitittimellä.

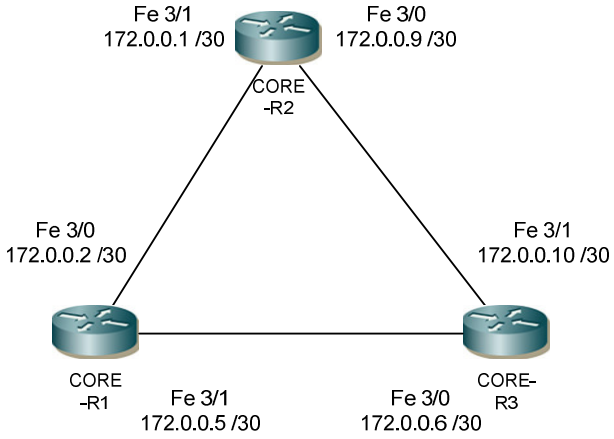
Kuviossa 30. nähdään reitittimellä sillä hetkellä aktiivisena olevat salaukset, niiden tyyppi IKE tai IPsec, käytössä olevat algoritmit sekä salattujen pakettien määrä. Tarkastelemalla salausten ID tunnuksia, voidaan taulukkoa verrata aikaisempiin ipsec sa taulukkoihin (ks kuvio 29. ja 30.) ja nähdään niiden täsmäävän.

```
WG3-R1#show crypto engine connection active
Crypto Engine Connections

   ID  Type    Algorithm           Encrypt  Decrypt IP-Address
 1001  IKE     SHA+DES                   0        0 3.3.0.11
 1002  IKE     SHA+DES                   0        0 3.3.0.11
 2001  IPsec   3DES+MD5                  0      132 3.3.0.11
 2002  IPsec   3DES+MD5                158        0 3.3.0.11
 2003  IPsec   3DES+MD5                  0      120 3.3.0.11
 2004  IPsec   3DES+MD5                120        0 3.3.0.11
```

KUVIO 30. Työryhmä 3 aktiiviset salaukset

## 8.3 Internet

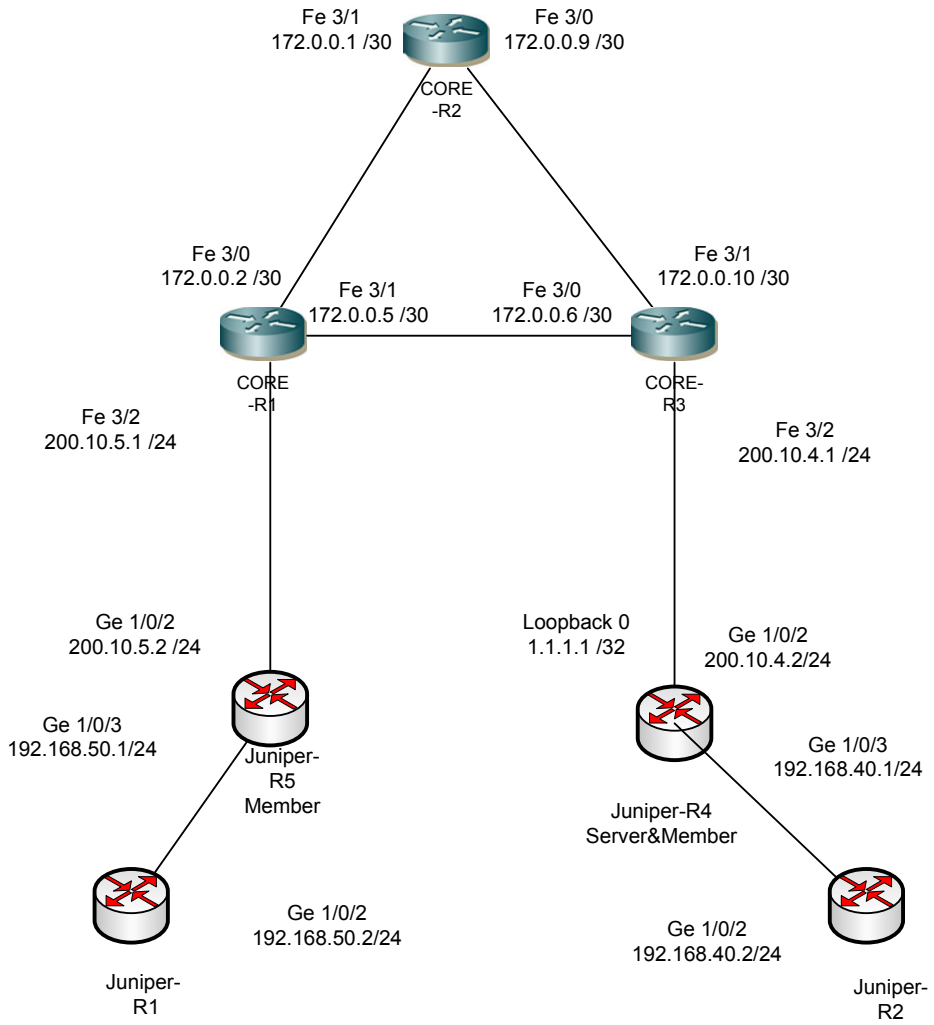
Työn muiden testattujen tekniikoiden kanssa käytettiin samaa Internetin konfiguraatiota kaikille. Internet rakennettiin käyttäen kolmea SpiderNet löytyvää Cisco-core reitintä. Internetin sisällä reititys toteutettiin käyttäen OSPF protokollaa jolla saatiin Internetin konfiguraatiot pidettyä yksinkertaisena. Alla olevassa kuviossa 31. on Internetin topologia, ilman rajapintoja joilla työryhmälaitteet liitettiin siihen.




KUVIO 31. Internetin topologia #2

## 8.4 Juniper Group VPN laitteisto ja topologia

Juniper Group VPN testattiin samalla testaten Server-Member co-location toimintoa, jolla samassa laitteessa suoritteen molemmat key server ja member roolit. Konfiguraatiot tehtiin käyttäen alla olevaa topologiaa (ks kuvio 32.). Jotta testaus saatiin tehtyä, liitettiin Juniper R1 ja Juniper R2 kuvastamaan esimerkiksi yrityksen sisäverkossa olevaa työasemaa.




KUVIO 32. Juniper Group VPN topologia

### 8.4.1 Juniper R4 Konfigorointi

Juniper R4 valittiin laitteeksi jolle määriteltiin kaksois-rooli, ja sen konfigurointi suoritettiin ensimmäisenä. Konfiguraatioiden tekeminen aloitettiin asettamalla käytetyille rajapinnoille ip osoitteet sekä aliverkko peitteet ja luomalla staattinen reititys Cisco Core-R3 reitittimelle jotta yhteys Internetiin saatiin muodostettua.

```
}
   ge-1/0/2 {
      unit 0 {
         family inet {
            address 200.10.4.2/24;
         }
      }
   }
   ge-1/0/3 {
      unit 0 {
         family inet {
            address 192.168.40.1/24;
         }
      }
   }
   lo0 {
      unit 0 {
         family inet {
            address 1.1.1.1/32;
```

Yllä olevassa konfiguraatiossa nähdään käytetyille rajapinnoille asetetut osoitteet sekä aliverkko peitteet bitti muodossa. Seuraavaksi määriteltiin staattinen reititys Core-R3:lle.

```
}
routing-options {
   static {
      route 0.0.0.0/0 next-hop 200.10.4.1;
   }
}
```

Tämän jälkeen testattiin että yhteys Internetin laitteille toimi, ennenkö jatkettiin itse Group-VPN konfiguraatioiden tekemistä. Se aloitettiin määrittelemällä palvelin roolin tarvitsemat asetukset.

```
}
security {
   group-vpn {
      server {
         ike {
            proposal srv-prop {
               authentication-method pre-shared-keys;
               dh-group group2;
               authentication-algorithm sha1;
               encryption-algorithm 3des-cbc;
            }
            policy srv-pol {
               mode aggressive;
               proposals srv-prop;
               pre-shared-key ascii-text "$9$UViqf36A1RSTzRSreXxDik"; ## SE
CRET-DATA
            }
            gateway gw1 {
               ike-policy srv-pol;
               address 200.10.4.2;
            }
            gateway gw2 {
               ike-policy srv-pol;
               address 200.10.5.2;
            }
            gateway srv-gw {
               ike-policy srv-pol;
               address 192.168.40.1;
            }
```

Yllä olevassa konfiguraatiossa luodaan aluksi IKE neuvottelua varten tarvittavat parametrit ehdotelmaan nimeltä *srv-prop*. Autentikointi tavaksi valittiin ennalta jaetut avaimet, Diffie–Hellma ryhmäksi 2 eli 1024 bittinen salaus, autentikointi algoritmiksi sha1 ja salaus algoritmiksi 3des-cbc. Seuraavaksi luotiin käytänne *srv-pol* jolle valittiin aggressiivinen neuvottelu tapa. Tähän käytänteeseen sidottiin aikaisemmin luotu ehdotelma ja asetet-

tiin sala-avain jota palvelimelle kirjautuvat laitteet käyttävät. Tämän jälkeen määriteltiin yhdyskäytävät joiden kanssa näitä käytettäisiin.
Seuraavaksi määriteltiin asetukset ipsec varten. Alla nähdään ipsec asetusten määrittely, ensin luotiin ehdotelma *group-prop* ja sille määriteltiin autentikointi algoritmi, salaus algoritmi sekä elinikä sekunteina.

```
security {
   group-vpn {
      server {
          ipsec {
            proposal group-prop {
               authentication-algorithm hmac-sha1-96;
               encryption-algorithm 3des-cbc;
               lifetime-seconds 3600;
            }
         }
```

Tämän jälkeen määriteltiin ryhmän kesken jaettavat asetukset ja salattavat verkko-osoitteet ja protokollat.

```
group grp1 {
    group-id 1;
    ike-gateway srv-gw;
    ike-gateway gw1;
    ike-gateway gw2;
    anti-replay-time-window 120;
    server-address 200.10.4.2;
    server-member-communication {
        communication-type unicast;
        encryption-algorithm aes-128-cbc;
        sig-hash-algorithm md5;
        certificate srv-cert;
    }
    ipsec-sa group-sa {
        proposal group-prop;
        }
        match-policy pol1 {
            source 192.168.40.0/24;
            destination 192.168.50.0/24;
            source-port 0;
            destination-port 0;
            protocol 0;
        }
        match-policy pol2 {
            source 192.168.50.0/24;
            destination 192.168.40.0/24;
            source-port 0;
            destination-port 0;
            protocol 0;
        }
    }
  }
}
co-location;
```

Aluksi luotiin ryhmä *grp1* jossa määriteltiin ryhmän yhdyskäytävät, uudelleen lähetettävien viestien viive, palvelimen osoite, palvelimen ja jäsenten välisen kommunikoinnin tapa *unicast* ja käytettävät algoritmit. Seuraavaksi määriteltiin salattavat verkko-

osoitteet ja niiden välillä salattavaksi kaikki liikenne. Lopuksi otettiin käyttöön co-location toiminto.

Tämän jälkeen konfiguroitiin member rooli samalle laitteelle käyttäen samoja asetuksia kuin palvelin roolille asetettiin. Aluksi määriteltiin ike asetukset vastaamaan palvelimelle määriteltyjä jotta tunnistautuminen onnistuisi. Sitten ehdotelmat liitettiin käytänteeseen ja tämä taas liitettiin käytettävään yhdyskäytävään. Viimeisenä määriteltiin VPN jota käytetään group ID:llä 1, eli aikaisemmin määritellyn grp1 kanssa ja määriteltiin sen käyttämä rajapinta ge-1/0/2.0

```
}
security {
   group-vpn {
      member {
         ike {
            proposal prop1 {
               authentication-method pre-shared-keys;
               dh-group group2;
               authentication-algorithm sha1;
               encryption-algorithm 3des-cbc;
            }
            policy pol1 {
               mode aggressive;
               proposals prop1;
               pre-shared-key ascii-text "$9$xZr-b2ZUH5Qn4aQn/CB17-V"; ## S
ECRET-DATA
            }
            gateway g1 {
               ike-policy pol1;
               address 200.10.4.2;
               local-address 192.168.40.1;
            }
         }
         ipsec {
            vpn v1 {
               ike-gateway g1;
               group-vpn-external-interface ge-1/0/2.0;
               group 1;
```

Viimeinen osa Juniper R4 konfiguraatiossa oli määritellä turva-alueet ja niiden käytänteet. Koska tarkoitus oli testata vain Group VPN toimintaa, jätettiin tarkempien turvallisuusmääritelmien tekeminen väliin ja sallittiin kaikki liikenne trust ja untrust alueille ja niiltä ulos. Alla on trust alueelta untrust alueen käytänteet, täydelliset konfiguraatiot löytyvät liitteestä 7.

```
policies {
    from-zone trust to-zone untrust {
        policy groupvpn {
            match {
                source-address any;
                destination-address any;
                application any;
            }
            then {
                permit {
                    tunnel {
                        ipsec-group-vpn v1;
                    }
                }
            }
        policy deny-all {
            match {
                source-address any;
                destination-address any;
                application any;
            }
            then { deny;
```

### 8.4.2 Juniper R5 Konfigurointi

Juniper R5 toimi pelkässä member roolissa joten sen konfigurointi jäi huomattavasti yksinkertaisemmaksi. Aluksi laitteelle määriteltiin käytettävät rajapinnat ja reititys Cisco Core-R1:lle. Edellä mainitut konfiguraatiot ovat näkyvillä alla.

```
}
interfaces {
   ge-1/0/0 {
      unit 0 {
         family inet {
            address 192.168.50.1/24;
         }
      }
   }
   ge-1/0/1 {
         unit 0;
   }
   ge-1/0/2 {
      unit 0 {
         family inet {
            address 200.10.5.2/24;
         }
      }

}
routing-options {
   static {
      route 0.0.0.0/0 next-hop 200.10.5.1;
   }
```

Kun yhteys verkon muihin laitteisiin oli testattu, jatkettiin Group VPN asetusten määrittelemisellä. Alla olevat member konfiguraatiot vastaavat täysin Juniper R4 tehtyjä konfiguraatioita niiltä osin. Ainot eroavuudet ovat käytänteiden nimissä sekä paikallisessa osoitteessa.

```
}
security {
   group-vpn {
      member {
         ike {
            proposal prop2 {
               authentication-method pre-shared-keys;
               dh-group group2;
               authentication-algorithm sha1;
               encryption-algorithm 3des-cbc;
            }
            policy pol2 {
               mode aggressive;
               proposals prop2;
               pre-shared-key ascii-text "$9$cYfrK8-VYZUHX7UHqmF3Sre"; ## S
ECRET-DATA
            }
            gateway g2 {
               ike-policy pol2;
               address 200.10.4.2;
               local-address 200.10.5.2;
            }
         }
         ipsec {
            vpn v1 {
               ike-gateway g2;
               group-vpn-external-interface ge-1/0/2.0;
               group 1;
            }
         }
      }
   }
```

Täydelliset Juniper R5 konfiguraatiot löytyvät työn lopusta liitteestä XX.

### 8.4.3 Juniper R1 ja R2 Konfigurointi

Juniper R1 ja R2 reitittimet konfiguroitiin, antaen niille vain ip osoitteet, aliverkkomaskit sekä staattinen reitti niihin yhdistettyihin Juniper R4 ja R5. Alla Juniper R1 konfiguraatiot. Juniper R2 konfiguraatiot ovat työn lopussa liitteessä 10.

```
interfaces {
    ge-0/0/0 {
        unit 0;
    }
    ge-1/0/1 {
        unit 0 {
            family inet {
                address 192.168.50.2/24;
            }
        }
    }
}
routing-options {
    static {
        route 0.0.0.0/0 next-hop 192.168.50.1;
    }
```

### 8.4.4 Group VPN todennus

Juniper Group VPN todennus aloitettiin tarkastelemalla Junos käyttöjärjestelmän tarjoamilla show komennoilla group-vpn asetusten toimivuutta. Aluksi katsottiin Juniper R4 server osan toimivuus. Alla olevasta kuviosta nähdään tiedot jotka palvelin jakaa ryhmän jäsenille näiden tunnistautuessa osaksi ryhmää *grp1* jonka ryhmä ID on 1.

Kuviosta 33. nähdään ryhmälle määritellyn IPSec turva-assosiaation asetukset, kuten protokollana käytettävä ESP ja autentikoinnin sekä salauksen kanssa käytettävät algoritmit.

```
root@Juniper-R4# run show security group-vpn server ipsec sa detail
Group: grp1, Group Id: 1
Total IPsec SAs: 1
  IPsec SA: group-sa
    Protocol: ESP, Authentication: sha1, Encryption: 3des
    SPI: 487908ed
    Lifetime left: 1463
    Policy Name: pol1
      Source: 192.168.40.0/24
      Destination: 192.168.50.0/24
      Source Port: 0
      Destination Port: 0
      Protocol: 0
    Policy Name: pol2
      Source: 192.168.50.0/24
      Destination: 192.168.40.0/24
      Source Port: 0
      Destination Port: 0
      Protocol: 0
```

KUVIO 33. Juniper Group VPN server ipsec

Myös palvelimelle rekisteröityneet laitteet olivat halutut, joten ainakin rekisteröityminen oli onnistunut. (ks. kuvio 34.)

```
[edit]
root@Juniper-R4# run show security group-vpn server registered-members
Group: grp1, Group Id: 1
  Member Gateway                Member IP       Last Update              Vsys
  srv-gw                        192.168.40.1    Mon Feb 13 2012 11:12:53 root
  gw2                           200.10.5.2      Mon Feb 13 2012 11:13:19 root
```

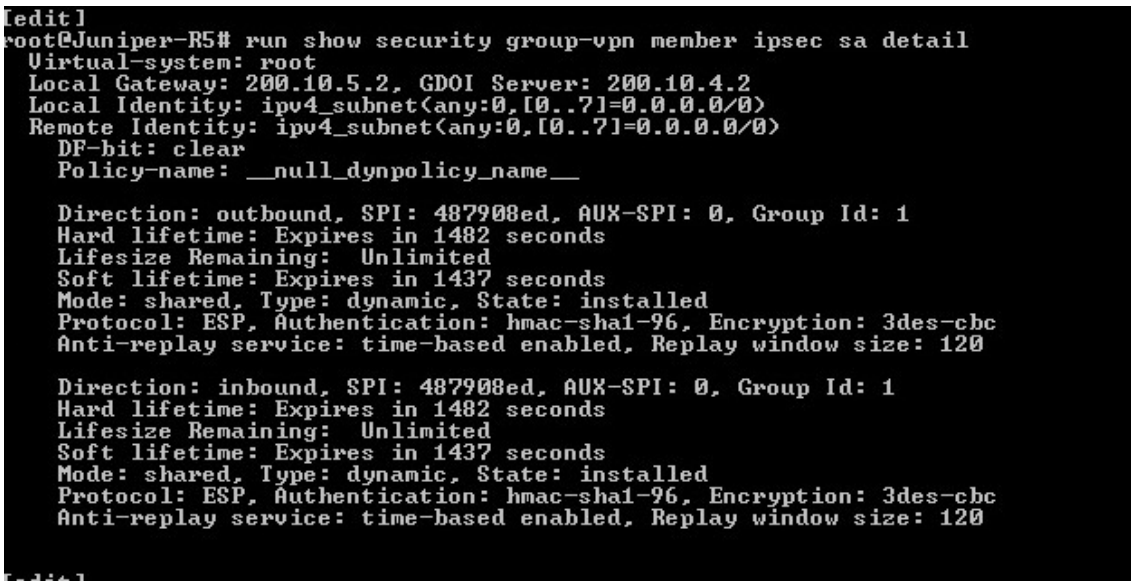
KUVIO 34. Rekisteröityneet laitteet

Seuraavaksi tarkasteltiin IPSec turva-assosiaatiota member näkökulmasta samalla R4-reitittimellä. Alla olevasta kuvasta voidaan nähdä että molemmat *Local Gateway* ja *GDOI server* ovat saman laitteen eri rajapinnat

Kuviossa 35. nähdään myös tarkemmin sisään ja ulospäin olevalle liikenteelle luodut käytänteet ja niiden käyttämät protokolla ja algoritmit.


KUVIO 35. Juniper R4 member ipsec

Myös Juniper R5 reitittimen ipsec sa vastasi toivottua, paitsi sen saama policy-name ei siirtynyt oikein serveriltä. Tähän ongelmaan ei löytynyt vastausta, mutta se ei aiheuttanut ongelmia itse ryhmän toiminnan osalta(ks. Kuvio 36.)

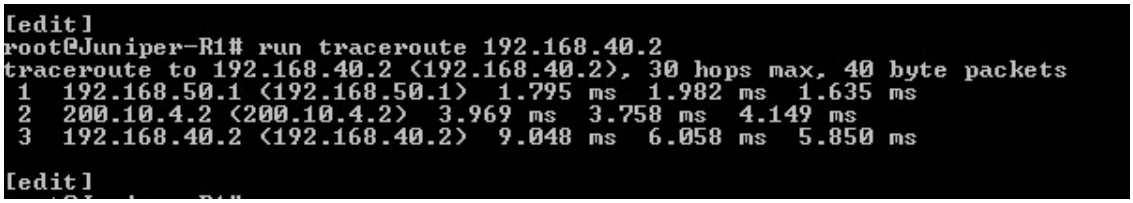


KUVIO 36. Juniper R5 member ipsec

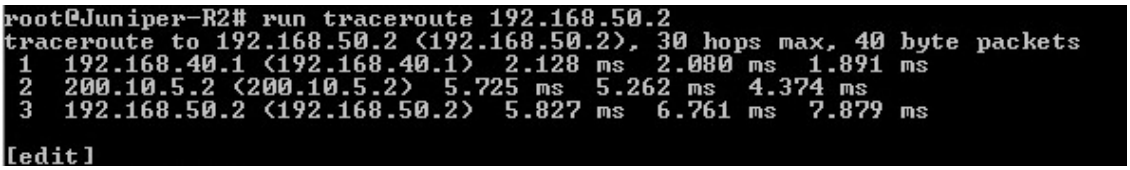
Kun jäsenten rekisteröityminen palvelimelle oli varmistettu, siirryttiin tarkastelemaan verkkojen välisen liikenteen salausta. Sitä varten suoritettiin *ping* sekä *traceroute* komennot Juniper R1 ja R2 välillä ja tarkasteltiin niiden jälkeen *ipsec statistics* komennon antamia tuloksia.

Aluksi tehtiin *traceroute* kysely molemmista R1 ja R2 toisilleen jotta nähtiin, että liikenne kulkee halutulla tavalla tunnelissa.

Kuvioista 37. ja 38. nähdään kuinka liikenne molemmista suunnista kulkee suoraan verkkojen välillä, eikä reitity Internetin kautta.



KUVIO 37. Traceroute verkosta 192.168.50 verkkoon .40



KUVIO 38. Traceroute verkosta 192.168.40 verkkoon .50

Lopuksi ajettiin *ping* komentoa laitteiden välillä jotta saatiin aikaan liikennettä ja voitiin tarkastella salauksen toimivuutta.

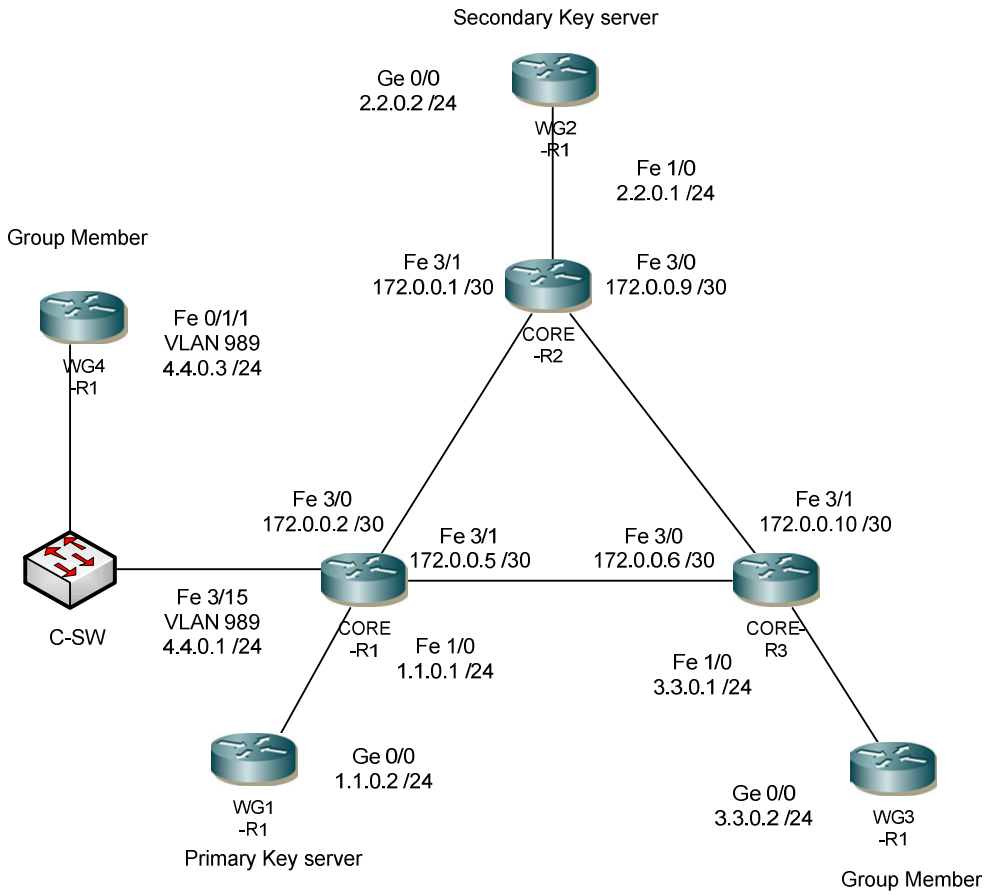
Kuviossa 39. nähdään kuinka salattujen pakettien ja bittien määrä on noussut kun laitteiden välillä on ajettu ping komentoa.

```
root@Juniper-R5# run show security group-vpn member ipsec statistics
ESP Statistics:
  Encrypted bytes:             5328
  Decrypted bytes:             3096
  Encrypted packets:             42
  Decrypted packets:             42
AH Statistics:
  Input bytes:                    0
  Output bytes:                   0
  Input packets:                  0
  Output packets:                 0
Errors:
  AH authentication failures: 0, Replay errors: 0
  ESP authentication failures: 0, ESP decryption failures: 0
  Bad headers: 0, Bad trailers: 0

[edit]
root@Juniper-R5# run show security group-vpn member ipsec statistics
ESP Statistics:
  Encrypted bytes:             6960
  Decrypted bytes:             4104
  Encrypted packets:             54
  Decrypted packets:             54
AH Statistics:
  Input bytes:                    0
  Output bytes:                   0
  Input packets:                  0
  Output packets:                 0
Errors:
  AH authentication failures: 0, Replay errors: 0
  ESP authentication failures: 0, ESP decryption failures: 0
  Bad headers: 0, Bad trailers: 0
```

KUVIO 39. Salaus kahden ping komennon välillä

## 8.5 Cisco GET VPN laitteisto ja topologia

Cisco GET VPN testattiin käyttäen ainoastaan Cisco laitteita, jotta niiden tarjoama usean GM mahdollisuus voitiin myös testata. Testauksessa käytettiin hyväksi aikaisempaa Core-reitittimille tehtyä internet konfiguraatiota. Testaus toteutettiin kaksiosaisena, aluksi muodostettiin vain yksinkertainen GET VPN, jossa oli yksi GM. Tähän ympäristöön sitten liitettiin toinen GM ja testattiin alkuperäisen GM pois kytkemistä. Konfiguraatiossa käytettiin apuna SpiderNetin center switchiä, jolla WG4 saatiin liitettyä helposti internet topologiaan. Vaikka laite on fyysisenä osana, ei se näy varsinaisessa ympäristössä. (ks. kuvio 40.)




KUVIO 40. GETVPN topologia

### 8.5.1 Cisco WG1-R1 Konfigurointi

Cisco WG1-R1 valittiin toimimaan GETVPN topologiassa ensisijaisena KS:nä. Sen konfigurointi aloitettiin tekemällä perus konfiguraatiot, jotta yhteys Internetiin saatiin toimimaan. Alla olevassa konfiguraatiossa nähdään tarvittavat komennot, joilla yhteys internetin reitittimille rakennettiin. Laitteessa käytettiin staattista reititystä suoraan Cisco core-r1 laitteelle, jonka vastaavalle rajapinnalle käytettiin osoitetta 1.1.0.1.

```
!
!
interface GigabitEthernet0/0
 description Link To CORE-R1
 ip address 1.1.0.2 255.255.255.0
 duplex auto
 speed auto
!
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 1.1.0.1
no ip http server
no ip http secure-server
!
!
```

Seuraavaksi aloitettiin itse GETVPN asetusten määrittely, luomalla aluksi ISAKMP käytänteet. Alla olevassa konfiguraatiossa luodaan aluksi ISAKMP käytänne ID:llä 10. Tälle määritellään sitten salausalgoritmiksi aes ja autentikointitavaksi ennalta jaetut avaimet. Komennolla *group 2* asetettiin salaukseen Diffie-Hellman  ryhmä kaksi, eli 1024 bittinen salaus.

```
!
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 2
!
```

Seuraavaksi määriteltiin ryhmän muiden laitteiden ip-osoitteet ja niille ennalta sovittu salasana, jonka avulla laite liitettäisiin osaksi ryhmää. Alla esimerkki WG3-R1 liittämisestä. ISAKMP kanssa käytettäväksi sala-avaimeksi annettiin ”getvpn” ja WG3-R1:den käyttämä ip osoite 3.3.0.2

```
!
crypto isakmp key getvpn address 3.3.0.2
!
```

Seuraavaksi luotiin ipsec asetukset, jotta GDOI:n tiedot pystytään jakamaan laitteille.

```
!
crypto ipsec transform-set getvpn-gdoi esp-aes esp-sha-hmac
!
crypto ipsec profile getvpn
 set security-association lifetime seconds 7200
 set transform-set getvpn-gdoi
!
```

Yllä olevassa konfiguraatiossa luodaan aluksi *transform-set* nimellä *getvpn-gdoi*, jolle määritellään *esp-aes* salausavain ja *esp-sha-hmac* autentikointi. Seuraavaksi luotiin ipsec profiili nimellä *getvpn*. Tälle profiilille määriteltiin SA eliniäksi 7200 sekuntia ja liitettiin sitä ennen luoto *transform-set getvpn-gdoi* siihen.

Seuraavaksi, ennenkö itse KS toiminta voitiin määritellä, luotiin julkista käyttöä varten RSA avain käyttäen *#crypto key generate rsa general-keys label getvpnexport-general modulus 1024 exportable* komentoa. Komennolla luotiin *getvpnexport* niminen avain, jossa 1024 bittininen salaus sekä määriteltiin sen olemaan *exportable*. *Exportable* määritelmä mahdollistaa avaimen jakamisen muille topologiaan liitettäville KS:lle

Seuraavaksi voitiin määritellä itse GDOI ja KS rooli laitteelle (siihen liityttävät konfiguraatiot alla). Se aloitettiin luomalla GDOI ryhmä nimellä *getvpn*. Tälle ryhmälle annettiin ID numero 123. Komento *server local* määrittelee KS roolin toimimaan kyseisellä laitteella. Uudelleen lähetyksen väliksi asetettiin 300 sekuntia, jotta sen toimintaa olisi helpompi valvoa. Oikeassa ympäristössä uuden avaimen lähetyksen välillä voidaan pitää 8-24 tuntia. Rekey avaimeksi asetettiin aikaisemmin luotu *getvpn-export-avain* ja uudelleen lähetyksen muodoksi *unicast*, eli jokaiselle laitteelle yksitellen.

```
!
crypto gdoi group getvpn
 identity number 123
 server local
  rekey lifetime seconds 300
  rekey retransmit 40 number 2
  rekey authentication mypubkey rsa getvpn-export-avain
  rekey transport unicast
  sa ipsec 1
   profile getvpn
   match address ipv4 109
   replay time window-size 5
  address ipv4 1.1.0.2
!
```

Lopuksi sidotaan profiili ipsec profiili ja GDOI ryhmä yhteen sekä asetetaan salattavan liikenteen määrittelevä access-list 109, komennolla *match address ipv4 109.* Lisäksi *anti-reply* viestin ajastimeksi asetettiin 5 sekuntia *replay time window-size 5.* Viimeinen osa konfiguraatiota määrittelee lähdeosoitteen, josta *rekey-paketit* lähetetään. Eli WG1-R1:den Internetiin yhteydessä olevan rajapinnan osoite.

Seuraavaksi luotiin, jo aikaisemmin käytettäväksi määritelty *access-list 109*, koska tarkoitus oli vain todentaa, että haluttujen verkkojen välinen liikenne salaantuu pidettään ACL yksinkertaisena ja siihen määriteltiin vain ne osoitteet joidenka välinen liikenne tulisi salata.

```
!
access-list 109 permit ip 3.3.0.0 0.0.0.255 4.4.0.0 0.0.0.255
access-list 109 permit ip 4.4.0.0 0.0.0.255 3.3.0.0 0.0.0.255
!
```

Yllä olevilla komennoilla ACL 109 määriteltiin koskemaan ip viestejä verkkojen 4.4.0.0 ja 3.3.0.0 välillä, näin ollen liikenne näiden verkkojen välissä tulisi kulkea GDOI suojauksen sisällä. Mikäli verkko olisi selvästi suurempi, vaatisi ACL kokonaan oman osuutensa toteutuksessa, jotta kaikki mahdolliset yhteydet saataisiin ”tunneloitua”.

### Cisco WG3-R1 ja WG4-R1 konfiguraatiot

Cisco WG3-R1 ja WG4-R1 suunniteltiin toimimaan GM tilassa, jotta perinteinen ryhmän sisäinen liikenne voitiin todentaa. Molemmat laitteet konfiguroitiin muuten samoin, paitsi WG4-R1 tuotiin center-switch:iä apuna käyttäen myös kiinni Cisco Core-R3 reitittimeen. Näin Internetin topologiaa ei tarvinnut muuttaa, yhtä rajapintaa luukuun ottamatta. Alapuolella WG4-R1 configuraatiot.

```
!
interface FastEthernet0/1/1
 ip address 4.4.0.3 255.255.255.0
 duplex auto
 speed auto
 vlan-id dot1q 989
  exit-vlan-config
 !
```

Konfiguraatiot aloitettiin muodostamalla yhteys Core-R1 reitittimelle center-switchin kautta. Center switchille menevälle rajapinnalle annettiin ip osoite ja se asettiin tunnistamaan VLAN tagi 989, jota käytettiin liikenteen ohjaamiseen Cisco Core-R1:lle. Center switchille tehdyt konfiguraatiot löytyvät liitteestä xx.

```
!
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 2
 lifetime 1200
crypto isakmp key getvpn address 1.1.0.2
crypto isakmp key getvpn address 2.2.0.2
!
```

Seuraavaksi määriteltiin ISAKMP varten tarvittavat tiedot, jotta yhteys KS kanssa voitiin muodostaa. salaukseksi valittiin aes ja authentikointi tavaksi ennalta sovitut avaimet. Komennolla *group 2* määriteltiin Diffie-Hellman  ryhmä kaksi, eli 1024 bittinen salaus. Lopuksi annettiin molempien käytettävien KS:ien osoitteet ja niiden kanssa käytettävät salasana ”getvpn”, käyttäen komentoa *crypto isakmp key getvpn address 1.1.0.2.*

Seuraavaksi luotiin gdoi ryhmä nimeltä getvpn, annettiin sille identity numero ja siinä toimivat KS:ät vielä uudelleen. Sillä aikaisemmat konfiguraatiot eivät määritelleet varsinaisesti KS:iä vaan vain niiden kanssa keskusteluun tarvittavat asetukset.

```
!
crypto gdoi group getvpn
 identity number 123
 server address ipv4 1.1.0.2
 server address ipv4 2.2.0.2
!
```

Lopuksi edellä luotu ryhmä liitettiin *crypto map* komennolla ensin ”karttaan” ja sitten kartta lisättiin käytettäväksi aikaisemmin määritellyn fastethernet 0/1/1 rajapinnan kanssa.

```
!
crypto map getvpn-map 10 gdoi
 set group getvpn
!

!
interface FastEthernet0/1/1
!
 crypto map getvpn-map
!
```

WG3-R1 konfiguraatiot toteutettiin käyttäen samoja komentoja, ainoastaan rajapinnan konfiguroimiseen ei tarvinnut asettaa VLAN asetuksia.

Kun WG1-R1 sekä molemmat GM:ät WG3-R1 ja WG4-R1 oli konfiguroitu ja saatu toimimaan keskenään, voitiin lisätä WG2-R1 joka toimi *secondary KS* roolilla. Osa sitä pohjustavista konfiguraatioista oli jo tehty ennen laitteen varsinaista lisäämistä. Kuten GM:lle määriteltiin jo alusta molempien KS osoitteet.

### WG2-R1 konfiguraatiot ja secondary KS asetukset

WG2-R1 konfiguraatiot aloitettiin tekemällä sille samat perusasetukset kuin WG1-R1, eli määrittelemällä sen aivan kuten normaalin KS:än. Näiden asetusten osalta konfiguraatiot löytyvät liitteestä 12.

Varsinainen COOP konfiguraatio aloitettiin luomalla WG1-R1:ssä kahden avaimen pari, jonka avulla KS reitittimien välille voitiin luoda sidos. Tämä tehtiin käyttäen komentoa *#crypto key generate rsa general-keys label getvpn-export-general modulus 1024 exportable.* Kyseinen komento luo kaksi avainta, jotka sitten voidaan syöttää toiselle laitteelle käyttäen kopio, liitä periaatetta. Aluksi WG2-R1:llä annettiin komento *#crypto key export rsa getvpn-exportgeneral pem terminal 3des getvpn*. Ja sen jälkeen syötettiin alla oleva kaksiosainen avain sille.

*-----BEGIN PUBLIC KEY-----*
*MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCkE0p+E4l6YeWItdbn8NE4/wor*
*v9Ks3YWKIO+4aUVpWIrcsVsTJlFXAh+SfJPsKNvoO2uQJGQGi61AsetamhHNVh/E*
*sVC0NadZfJZyie5j8X6/smBNTeYIrLTZ3hqQTQdaKYZZDSZXswIqegzqF4a/jO55*
*1iA38YJt7fSvfdzu3wIDAQAB*
*-----END PUBLIC KEY-----*
*-----BEGIN RSA PRIVATE KEY-----*
*Proc-Type: 4,ENCRYPTED*
*DEK-Info: DES-EDE3-CBC,46822EB4665FAC4D*

*CLJ6rBpWSDxGwOzohHg3YJxgcc5ermWeinQT030NfvFGI+LjQApbNEjAtOFbVSoZ*
*XifR9WqwPzdm49u9kEusI6oXARDeJI3D0eWhQKod4YuwXG4HB0DwjZBQpNESkCEB*
*oRrmp6LgCPf3csfA757kZ5kBiUUgyym5DWMFo1DgVkAKAmzH+wQBfxV3LGr7Lbee*
*BaklW0DBoFhH5Ft9o1VGdiEFghAaW08Y7chO5hxAFZb3fV1xyeyG97C8G/AU9lWS*
*6XIRqAwrhBBWMG8397iE0cFBPtZdQF737juCewfnG7Ru+uvQuHIMoiI3llhri8tS*
*oSszHZR0b4XeWbtdL5KrKVaGVdlUXgLBgyEWYIORGNXNhqbOTFaYupc33ZCMhWUZ*
*JP/jetTkGkdejnDs+U0P+d/KvwYclQctX3BU+EGlqQ8rjKFSwhihmvKfifn3Gtm4*
*PBdmM9gjhP4aEh3xBHV5G8ZIqNVZlKXOYdBrS49n5yCbEb+RdvcY0Gin2K9mVFsr*
*YJaxRi3w6sdJQVQtDBldDxXT9F3V0lcGdu6PwGz3MK6utS2q4IvscpmhMIvD4ubg*
*Y7V5RJkj9ucJem51QX5TIIkqLMFb8MS1X9s3KscCoYMn+9kBReL89aAmYZMX1E1G*
*rY+A0D0EyOUB/1DVjiToFavLh8iI0GyObljpITxsNwch8B1kRiKoQUrJWhd1Jie6*
*pdkD32Vm0fJBvVHZkB1tdJAC8BXjAr0hyJqFcy+aEIe1q6Hzg14nMaMhvjlLBUPO*
*2FpEkoZzu9BSbeQpOgdLF4nvouyV7PdBxXvPscxwu7Zwxda8SFH39Q==*
*-----END RSA PRIVATE KEY-----*

Kun avaimet oli saatu jaettua, voitiin aloittaa varsinaiset konfiguraatiot laitteiden saamiseksi COOP tilaan. Aluksi *crypto GDOI group* alle lisättiin komennot joidenka avulla laitteiden ensisijaiset roolit määriteltiin sekä aktivoitiin reduntanttisuus jossa toisen laitteen ollessa estynyt, ottaa toinen sen paikan.

```
redundancy
  local priority 75
  peer address ipv4 1.1.0.2
```

Yllä olevat komennot asetettiin WG2-R2:lle, niissä reduntanttisuus otettiin käyttöön ja sen prioriteetti arvoksi asetettiin 75 sekä määriteltiin toisen KS fyysinen osoite. Alla vastaavat komennot WG1-R1:ltä, niissä voidaan nähdä että sille asetettiin suurempi prioriteetti arvo, jollain se tulee ensisijaisesti valituksi KS:iksi jolle laitteet rekisteröityvät.

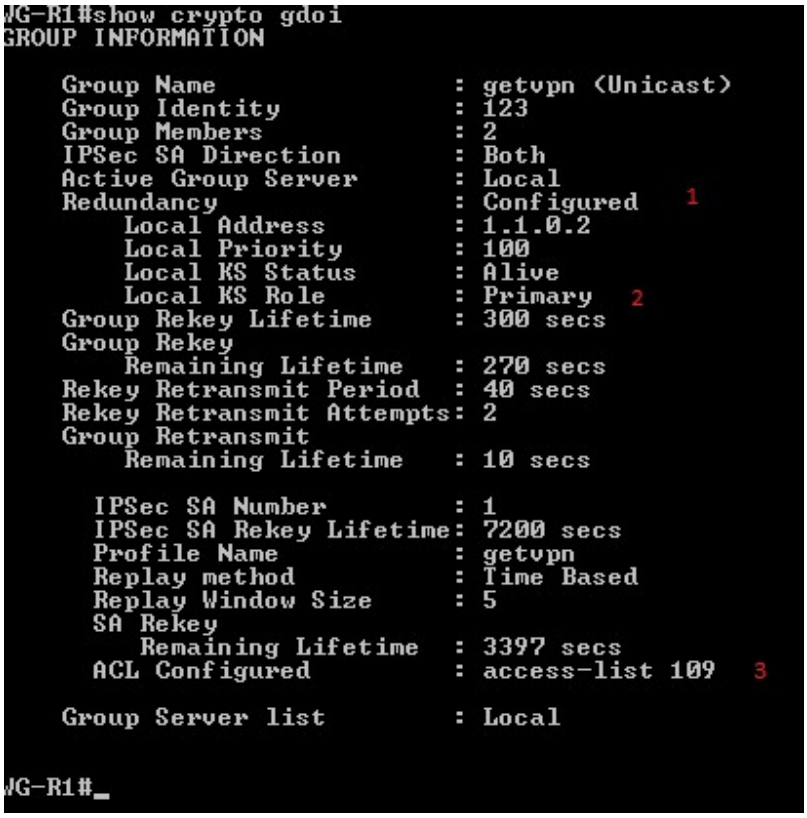
```
redundancy
  local priority 100
  peer address ipv4 2.2.0.2
```

Näiden muutosten lisäksi, muuta ei tarvinnut tehdä sillä kaikille GM:lle oli jo alun perinkin annettu molempien KS osoitteet.

### 8.5.2 GETVPN todennus

GETVPN todennus aloitettiin tarkastelemalla niin KS kuin GM saamia tietoja käyttäen *show crypto gdoi*, *show crypto isakmp sa* ja *show crypto ipsec sa* – komentoja. Aluksi tarkasteltiin KS laitteita, jotta niiden välinen yhteys, sekä niille rekisteröityneet laitteet voitiin todeta.

Kuviossa 41. nähdään aluksi ryhmän nimi, eli getvpn sekä sen ID 123 ja että siihen kuuluu tällä hetkellä kaksi (2) GM tilassa olevaa laitetta. Lisäksi merkityssä kohdassa 1. nähdään että redunttantisuus on käytössä, ja sen alla olevat tiedot paikallisesta osoitteesta ja prioriteetti arvosta. Kohdassa 2. nähdään kuinka kyseinen laite on saanu KS roolin primary, eli se toimii ensijaisena laitteena.




KUVIO 41. WG1-R1 show crypto gdoi

Lisäksi kuviosta 41. nähdään uudelleen lähetettävien viestejen voimassaolo aja. Kohta 3. näyttää että ryhmän kanssa on käytössä access-list 109, joka määrittelee salattavan liikenteen.

Seuraavaksi tarkasteltiin vielä samaa WG1-R1 laitetta tarkemmin COOP osalta, käyttäen *show crypto gdoi ks coop* komentoa.

Kuviosta 42. nähdään aluksi laitteen omat tiedot, sekä ryhmän nimi. Peer Sessions alla nähdään yksi aktiivinen COOP laite. Kohdassa 1. nähdään laitteen fyysinen osoite, eli WG2-R1 osoite, kohdassa 2. WG2 reitittimen prioriteetti arvo. Kohdassa 3. nähdään että kyseinen WG2-R1 on toiminnassa ja se on valittu toissijaiseksi KS-reitittimeksi. Kohdan 4. alla on laitteiden välille muodostuneen IKE käytänteen liikenne, viestien määrä sekä kokonaismäärä bitteinä.

```
WG-R1#show crypto gdoi ks coop
Crypto Gdoi Group Name :getvpn
        Group handle: 2147483650, Local Key Server handle: 2147483650

        Local Address: 1.1.0.2
        Local Priority: 100
        Local KS Role: Primary   , Local KS Status: Alive
        Primary Timers:
                Primary Refresh Policy Time: 20
                Remaining Time: 13
                Antireplay Sequence Number: 64089

        Peer Sessions:
        Session 1:
                Server handle: 2147483651
                Peer Address: 2.2.0.2      1
                Peer Priority: 75     2
                Peer KS Role: Secondary , Peer KS Status: Alive   3
                Antireplay Sequence Number: 1193

                IKE status: Established    4
                Counters:
                    Ann msgs sent: 59697
                    Ann msgs sent with reply request: 4
                    Ann msgs recv: 3686
                    Ann msgs recv with reply request: 3
                    Packet sent drops: 4385
                    Packet Recv drops: 3
                    Total bytes sent: 32526732
                    Total bytes recv: 952086


WG-R1#
```

KUVIO 42. WG1-R1 crypto gdoi ks coop

Seuraavaksi tarkasteltiin WG3-R1 laitteen antamia tietoja, WG3-R1 toimi toteutuksessa toisena GM laitteena ja sen tarkastelu aloitettiin antamalla *show crypto gdoi* komento.

Kuviossa 43 nähdään kuinka GM laitteen antama tuloste samasta komennosta eroaa hieman KS laitteesta. Alusta löytyvät samat tiedot, mutta kohdassa 1. nähdään eroavuus. Laitteella näkyy sillä hetkellä aktiivinen KS, sekä kaikkien muiden sille annettujen KS osoitteet. Kohdassa 2. nähdään uudelleen lähetettyjen viestejen tila, niistä nähdään kuinka kaikki rekisteröinnin jälkeen lähetetyt 39 viestiä on myös saatu kuitattua *Acks sent* ollessa yhtä suuri. Kohdassa 3. nähdään suoraan KS:ltä ladatut ACL tiedot. Kohdassa 4. puolestaan näkyy KEK käytänteiden asetukset. Kohdassa 5 puolestaan nähdään IPsec SA joka on muodostettu liikennöintiin KS kansas rajapinnassa GigabitEthernet 0/0.

```
WG3-R1#show crypto gdoi
GROUP INFORMATION

    Group Name               : getvpn
    Group Identity           : 123
    Rekeys received          : 39
    IPSec SA Direction       : Both
    Active Group Server      : 1.1.0.2
    Group Server list        : 1.1.0.2   1
                               2.2.0.2

    GM Reregisters in        : 2128 secs
    Rekey Received(hh:mm:ss) : 00:02:48


    Rekeys received
         Cumulative          : 39
         After registration  : 39  2
    Rekey Acks sent          : 39

 ACL Downloaded From KS 1.1.0.2:
   access-list  permit ip 3.3.0.0 0.0.0.255 4.4.0.0 0.0.0.255
   access-list  permit ip 4.4.0.0 0.0.0.255 3.3.0.0 0.0.0.255  3

KEK POLICY:
    Rekey Transport Type     : Unicast
    Lifetime (secs)          : 300
    Encrypt Algorithm        : 3DES              4
    Key Size                 : 192
    Sig Hash Algorithm       : HMAC_AUTH_SHA
    Sig Key Length (bits)    : 1024

TEK POLICY for the current KS-Policy ACEs Downloaded:
  GigabitEthernet0/0:
    IPsec SA:
        spi: 0xD7726036(3614597174)
        transform: esp-aes esp-sha-hmac
        sa timing:remaining key lifetime (sec): (2357)   5
        Anti-Replay(Time Based) : 5 sec interval

WG3-R1# _
```

KUVIO 43. WG3-R1 show crypto gdoi (GM)

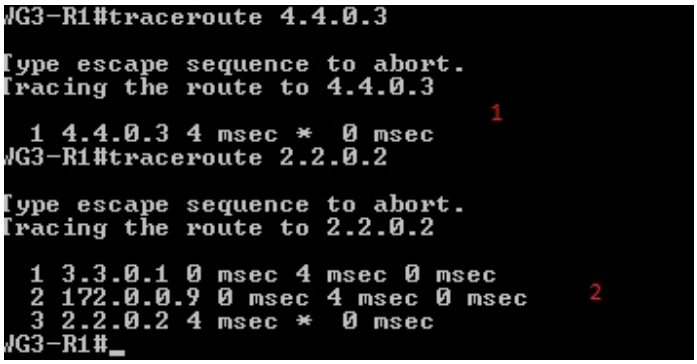
ISAKMP muodostuminen voidaan myös todeta antamalla komento;

*show crypto isakmp sa*, josta nähdään kuinka laitteella on muodostunut ISAKMP sille määritellyn KS kanssa (ks. kuvio 44)


KUVIO 44. WG3-R1 crypto ISAKMP sa

Seuraavaksi haluttiin todentaa, että liikenne todellakin salaantuu verkkojen 3.3.0.0 ja 4.4.0.0 välillä kuten ACL asetusten perusteella kuuluisi. Tätä testattiin aluksi tarkastelemalla *traceroute* komennolla, kuinka liikenne reitittyy.

Kuviosta 45 nähdään kuinka ensin kohdassa 1. ajatetaan *traceroute* WG4-R1 laitteelle johon liikenteen tulisi kulkea tunnelissa, näin tapahtuukin eikä reittiä näy vaan viesti siirtyy suoraan laitteelta toiselle. Jotta tilanne voidaan vielä todentaa paremmin, ajettiin heti perään traceroute laitteelle WG2-R1. Kohdassa 2. nähdään kuinka nyt liikenne reitittyy Internetin kautta, eikä kulje suorassa tunnelissa.




KUVIO 45. WG3-R1 traceroute to WG4 ja WG2

Seuraavaksi tahdottiin vielä varmistaa, että liikenne todellakin oli salautunut. Tätä varten WG3-R1 reitittimellä annettiin komento *show crypto ipsec sa detail*, jolla voitiin nähdä IPSEC pakettejen määrä sekä minkä laitteiden välillä niitä oli liikkunut. Aluksi ajettiin *ping* komentoa laitteiden WG3-R1 ja WG4-R1 jotta niiden välille saatiin kehitettyä liikennettä.


KUVIO 46. WG3-R1 ipsec sa

Kuviossa 46. nähdään laitteen muodostama IPSEC SA, jonka alkuperä ja kohde on määritelty kohdassa 1. Kohdassa 2. nähdään mistä perusteet kyseiselle SA on saatu, eli tässä tapauksessa ACL:stä. Kohdassa 3. nähdään liikenteen määrä ja se kuinka salattujen ja purettujen pakettien määrät ovat samassa suhteessa. Kohdissa 4. ja 5. nähdään molempiin suuntiin luodut esp parametrit, sekä se kuinka ne ovat sidoksia *crypto map: getvpn-map*:iin.

### 8.5.3 GETVPN COOP toiminnon todennus.

Kun itse GETVPN oli todettu toimivan, haluttiin vielä testata kuinka COOP toiminto todellisuudessa käyttäytyisi. Tämä toteutettiin sulkemalla Cisco Core-R1:lla oleva rajapinta WG1-R1 suuntaan ja näin olen estämään muilta laitteilta liikenne sille.

Kuviossa 47. nähdään kuinka WG2-R1 joka alun perin oli secondary KS, siirtää primary roolin itselleen kun yhteys WG1-R1 katkeaa (kohdat 1. ja 2.)

```
WG2-R1#
*May  7 11:34:17.896: %GDOI-5-COOP_KS_TRANS_TO_PRI: KS 2.2.0.2 in group getvpn t
ransitioned to Primary (Previous Primary = 1.1.0.2)     1
WG2-R1#show crypto gdoi ks coop
Crypto Gdoi Group Name :getvpn
        Group handle: 2147483650, Local Key Server handle: 2147483650

        Local Address: 2.2.0.2
        Local Priority: 75
        Local KS Role: Primary    , Local KS Status: Alive
                                                          2
        Primary Timers:
                Primary Refresh Policy Time: 20
                Remaining Time: 2
                Antireplay Sequence Number: 1207

        Peer Sessions:
        Session 1:
                Server handle: 2147483651
                Peer Address: 1.1.0.2
                Peer Priority: 1
                Peer KS Role: Primary    , Peer KS Status: Dead    3
                Antireplay Sequence Number: 64242

                IKE status: Failed
                Counters:
                    Ann msgs sent: 1152
                    Ann msgs sent with reply request: 2
                    Ann msgs recv: 18654
                    Ann msgs recv with reply request: 0
                    Packet sent drops: 52
                    Packet Recv drops: 0
                    Total bytes sent: 311407
                    Total bytes recv: 10384775


WG2-R1#_
```

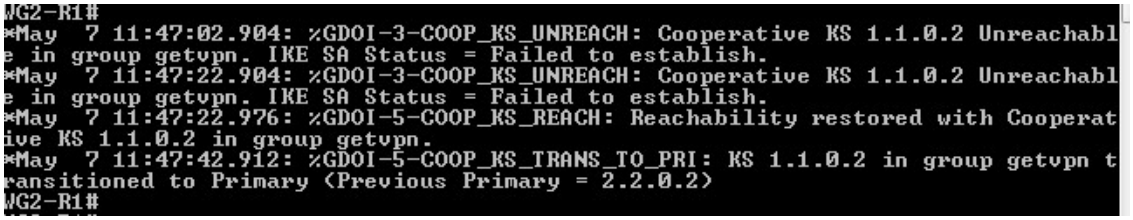
KUVIO 47. WG2-R1 KS Coop toiminta 1.

Tämän jälkeen laite jatkaa toimintaansa aivan normaaliin tapaan, ja aloittaa rekey lähetykset ryhmän GM:ille, sekä lähettää tarkistus viestejä WG1-R1 suuntaan, sen uudelleen käyttöön palautumisen varalta (ks. kuvio 48).

```
*May  7 11:36:17.896: %GDOI-3-COOP_KS_UNREACH: Cooperative KS 1.1.0.2 Unreachabl
e in group getvpn. IKE SA Status = Failed to establish.
*May  7 11:36:23.000: %GDOI-5-KS_SEND_UNICAST_REKEY: Sending Unicast Rekey for g
roup getvpn from address 2.2.0.2 with seq # 11
*May  7 11:36:27.896: %GDOI-3-COOP_KS_UNREACH: Cooperative KS 1.1.0.2 Unreachabl
e in group getvpn. IKE SA Status = Failed to establish.
```

KUVIO 48. KS viestit, rekey ja Coop

Kun yhteys WG1-R1:lle palautetaan, palaa asetelma takaisin normaaliksi ja WG1-R1 ottaa roolinsa primary KS:nä takaisin, sen suuremman priority arvon perusteella. Kyseinen tapahtuma voidaan nähdä alla olevassa kuviossa 49.

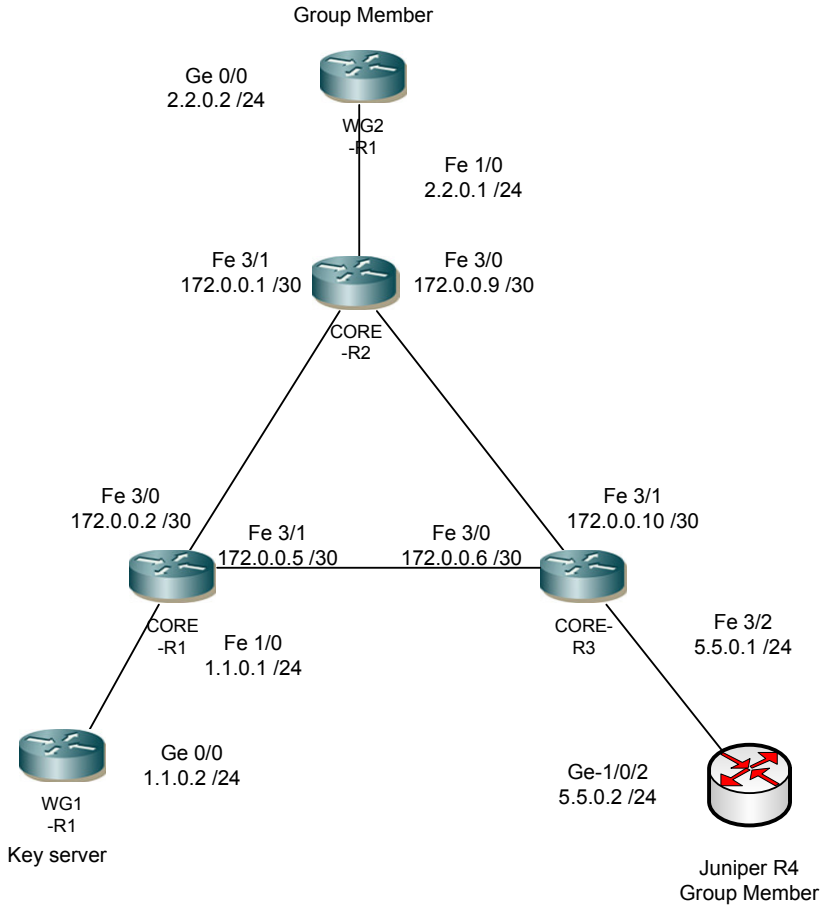


KUVIO 49. KS restore viestit

Tämän jälkeen GETVPN toiminta palautuu aloitustilaan, jossa WG1-R1 on primary KS ja hoitaa kaikki laitteiden rekisteröinnit ja avainten uudistamiset. Mikäli uusi laite yrittäisi liittyä ryhmään, toisen KS ollessa alhaalla, joutuu laite ensin yrittämään sille annetuista KS:istä ensimmäistä ja kun se ei saa vastausta siirtyy se vasta sitten yrittämään toista KS. Tästä syntyy pieni, noin minuutin viive rekisteröintiin jos GM:lle annettu ensisijainen laite lopettaa toimintansa, ennenkö rekisteröinti on suoritettu.

## 8.6 Cisco ja Juniper yhtyeentoimivuus Group Encrypted VPN kanssa

Juniper ja Cisco välist Group VPN ratkaisua varten tehtiin topologia jossa molempien laitevalmistajien laitteet toimivat GM roolissa ja Ciscon laite olisi KS (ks. kuvio50). Tältä osin varsinaista valintaa ei ollut, sillä Juniperin omat dokumentit aiheesta antoivat toimivuudelle ehdoksi nimenomaan Cisco -laitteen käyttämisen KS:nä. Testaus pidettiin mahdollisimman yksinkertaisena, jotta itse VPN toimivuus voidaan todentaa, eikä jouduttaisi keskittymään liiaksi access-list tai trust-zone tarkasteluun.




KUVIO 50. Cisco ja Juniper yhteensopivuus topologia

### 8.6.1 Cisco WG1-R1 konfigurointi

Testaus aloitettiin konfiguroimalla WG1-R1 reitin toimimaan KS roolissa. Aluksi laitteelle määriteltiin käytettävät rajapinnat, niille ip-osoitteet ja aliverkkopeitteet sekä konfiguroitiin staattinen reititys Cisco Core-R1:lle. Kyseiset konfiguraatiot alla.

```
!
interface GigabitEthernet0/0
 description Link-to-CORE1
 ip address 1.1.0.2 255.255.255.0
 duplex auto
 speed auto
!
ip route 0.0.0.0 0.0.0.0 1.1.0.1
!
```

Kun asetukset oli tehty ja yhteyden toimivuus internetin laitteisiin testattu, käyttäen *ping*- komentoa, jatkettiin itse GET VPN asetusten määrittelemiseen. Aluksi luotiin IKE tarvitsemat ISAKMP käytänteet.

```
!
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 2
```

Luotiin ISAKMP käytänne tunnuksella 10, sille asetettiin salausalgoritmiksi aes, autentikointi muodoksi ennalta jaetut avaimet sekä komennolla *group 2* valittiin avainten jakamista varten Diffie-Hellman ryhmä 2.

```
!
crypto isakmp key getvpn address 2.2.0.2
crypto isakmp key getvpn address 5.5.0.2
!
```

Seuraavaksi määriteltiin ryhmänjäsenille ISAKMP avain, joka tässä tapauksessa oli “getvpn”. Avain määriteltiin jokaiselle jäsenellä erikseen, niiden fyysisen osoitteen perusteella.

Kun IKE asetukset oli määritelty, jatkettiin määrittelemään IPSEC asetuksia. Aluksi luotiin niitä varten *transform-set* nimellä ”*getvpn-trans*” sille asetettiin salaus-algoritmi *esp-aes* ja autentikointi-algoritmiksi esp-sha-hmac. Sen jälkeen luotiin IPSEC profiili ”*getvpn-gdoi*” jonka voimassaolo ajaksi asetettiin 7200 sekunttia ja lopuksi se sidottiin yhteen aikaisemmin luodun transfrom-set kanssa.

```
!
crypto ipsec transform-set getvpn-trans esp-aes esp-sha-hmac
!
crypto ipsec profile getvpn-gdoi
 set security-association lifetime seconds 7200
 set transform-set getvpn-trans
!
```

Kun kysyiset konfiguraatiot oli tehty, aloitettiin ryhmän asetusten määrittely GDOI osalta. Se aloitettiin luomalla RSA avaimet, käyttäen komentoa;

*(config)#crypto key generate rsa general-keys label getvpn-export modulus 1024 exportable*

Kun avain oli luotu, voitiin jatkaa GDOI:n määrittelemiseen. Se aloitettiin luomalla GDOI-ryhmä nimeltä ”getvpn” ja antamalla sille identity numero 123. Sen jälkeen server local komennolla asetettiin kyseinen laite toimimaan KS roolissa.

```
!
crypto gdoi group getvpn
 identity number 123
 server local
```

Seuraavaksi määriteltiin uudelleen lähetettävien avaimien asetukset. Voimassaolo ajaksi asetettiin 300 sekuntia, uudelleen lähetys väliksi 40 sekuntia ja kaksi yritystä. Avaimiksi asetettiin aikaisemmin generoidut RSA avaimet ja lähetystyypiksi unicast, eli jokaiselle laitteelle yksilöllisesti.

```
rekey lifetime seconds 300
 rekey retransmit 40 number 2
 rekey authentication mypubkey rsa getvpn-export
 rekey transport unicast
```

Seuraavaksi liitettiin aikaisemmin luotu IPSEC turva-assosiaatio käytettäväksi kyseisen ryhmän kesken. Valittiin salattavan liikenteen määrittteleväksi pääsylistaksi access-list 109 ja poistettiin vastausviestien tarve. Vastausviestien pois jättäminen, oli yksi vaatimus jotta Cisco ja Juniper pystyvät toimimaan yhdessä. Lopuksi annettiin vielä KS avainten lähetyksessä käyttämän rajapinnan osoite.

```
sa ipsec 1
  profile getvpn-gdoi
  match address ipv4 109
  no replay
 address ipv4 1.1.0.1
```

Viimeinen osa KS konfiguraatiota oli määritellä halutut verkot tai protokolla tyypit salattaviksi, tämä tehtiin käyttäen access-list komentoa. Tässä tapauksessa valittiin liikenne pisteiden 5.5.0.2 ja 2.2.0.2 välillä salattavaksi.

```
access-list 109 permit ip host 2.2.0.2 host 5.5.0.2
access-list 109 permit ip host 5.5.0.2 host 2.2.0.2
```

### 8.6.2 Cisco WG2-R1 konfigurointi

Cisco WG2-R1 konfiguroitiin toimimaan toisena ryhmänjäsenenä. Reitittimen konfigurointi aloitettiin määrittelemällä internetiin yhdeysessä olevalle rajapinnalle ip osoite, aliverkkopeite sekä staattinen reititys reitittimelle Cisco Core-R1.

```
!
interface GigabitEthernet0/0
 description Link-to-CORE2
 ip address 2.2.0.2 255.255.255.0

ip route 0.0.0.0 0.0.0.0 2.2.0.1
!
```

Seuraavaksi luotiin IKE tason 1 käytänteet, niiden osalta kaikki tehtiin vastaamaan KS:lle annettuja asetuksia.

```
!
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 2
 lifetime 1200
crypto isakmp key getvpn address 1.1.0.2
!
```

Aluksi luotiin ISAKMP käytänteet, tunnuksella 10. Salaukseksi aes, autentikointi ennalta jaetulla avaimella, Diffie-Hellman avaintenvaihto ryhmä 2 komennolla *group 2* ja voimassa olo aika 1200 sekunttia. Lopuksi annettiin vielä KS:lle rekisteröitymistä varten sovittu salasana ”getvpn” ja KS:n ip-osoite.

Seuraavaksi luotiin GDOI ryhmä, nimellä *getvpn*. Ryhmä tunnisteeksi annettiin 123 ja sille määriteltiin avainpalvelimen osoite 1.1.0.2. Lisäksi luotiin krypto-kartta *getvpn-map*, joka liitettiin yhteen aiemmin luodun ryhmän *getvpn* kanssa. Lopuksi krypto-kartta liitettiin rajapintaan GigabitEthenet0/0 jossa tunnelit tultaisiin muodostamaan.

```
!
crypto gdoi group getvpn
 identity number 123
 server address ipv4 1.1.0.2
!
crypto map getvpn-map 10 gdoi
 set group getvpn
!
interface GigabitEthernet0/0
 description Link-to-CORE2
 ip address 2.2.0.2 255.255.255.0
 duplex auto
 speed auto
 crypto map getvpn-map
!
```

Näiden konfiguraatioiden jälkeen, testattiin laitteen toimivuus käyttämällä *ping* -komentoa, jotta voitiin nähdä että laitteella oli yhteys kaikkiin verkon laitteisiin.

### 8.6.3 Juniper-R4 konfigurointi

Juniper-R4 oli topologian toinen GM ja sen konfigurointi aloitettiin myös määrittelemällä tarvittavat rajapinnat ja aliverkkopeitteet sekä muodostamalla reitti internetin reitittimille.

```
}
   ge-1/0/2 {
      unit 0 {
         family inet {
            address 5.5.0.2/24;
         }
}
routing-options {
   static {
      route 0.0.0.0/0 next-hop 5.5.0.1;
```

Kun perus konfiguraatiot oli tehty ja yhteys testattu aloitettiin laitteen liittäminen osaksi GDOI ryhmää.

```
}
security {
   group-vpn {
      member {
         ike {
            proposal getvpn-prop {
               authentication-method pre-shared-keys;
               dh-group group2;
               authentication-algorithm sha1;
               encryption-algorithm aes-128-cbc;
            }
```

Aluksi määriteltiin IKE asetukset luomalla ehdotus nimellä *getvpn-prop*. Siihen määriteltiin käytettäväksi ennalta jaettuja avaimia, Diffie-Hellman avaintenvaihto ryhmäksi 2, autentikointi-algoritmiksi sha1 ja salaus-algoritmiksi 128 bittinen aes.

Seuraavaksi tehtiin käytänne, johon sitä ennen luotu ehdotelma voitiin liittää.

```
}
        policy pol2 {
            mode main;
            proposals getvpn-prop;
            pre-shared-key ascii-text "$9$cYfrK8-VYZUHX7UHqmF3Sre"; ## S
ECRET-DATA
        }
```

Yhteydenmuodostus tavaksi valittiin normaali komennolla *mode main*, sen jälkeen liitettiin aiemmin luotu ehdotelma ja annettiin salasana jota käytetään KS kanssa yhteyden luomiseen.

Kun IKE oli määritelty, annettiin sille käytettävä rajapinta sekä paikallinen osoite ja avainpalvelimen osoite.

```
gateway g2 {
            ike-policy pol2;
            address 1.1.0.2;
            local-address 5.5.0.2;
        }
    }
    ipsec {
        vpn v1 {
            ike-gateway g2;
            group-vpn-external-interface ge-1/0/2.0;
            group 123;
        }
```

Yläpuolella konfiguraatiossa näkyy myös kuinka kyseinen gateway liitetään osaksi ipsec vpn:nää ja määritellään tunnelin muodostava rajapinta. Lisäksi sille annetaan sama group ID kuin Cisco laitteille, eli 123.

```
}
   policies {
      from-zone trust to-zone untrust {
         policy getvpn {
            match {
               source-address any;
               destination-address any;
               application any;
            }
            then {
               permit {
                  tunnel {
                     ipsec-group-vpn v1;
                  }
               }
            }
         }
         policy deny-all {
            match {
               source-address any;
               destination-address any;
               application any;
            }
            then {
               deny;
```

Yllä olevassa konfiguraatiossa määritellään luottamusalueet ja ohjataan niiden perusteella liikenne kulkemaan tunneliin. Alla vastaavat toiseen suuntaan olevasta liikenteestä.

```
}
      from-zone untrust to-zone trust {
         policy getvpn {
            match {
               source-address any;
               destination-address any;
               application any;
            }
            then {
               permit {
                  tunnel {
                     ipsec-group-vpn v1;
                  }
```

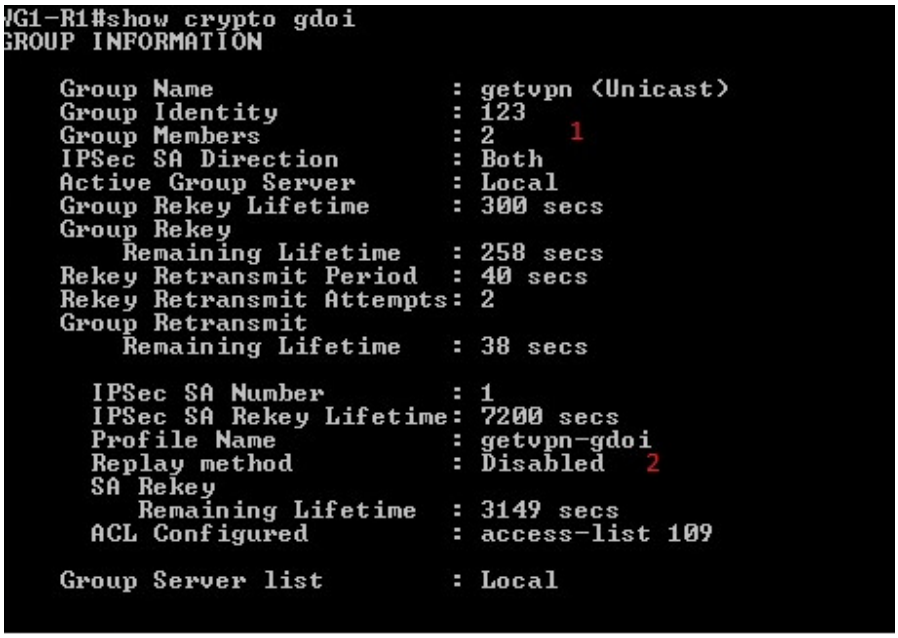
```
            }
        }
    }
    policy deny {
        match {
            source-address any;
            destination-address any;
            application any;
        }
        then {
            deny;
        }
    }
}
default-policy {
    deny-all;
```

Kun konfiguraatiot kaikilla kolmella laitteella oli saatu tehtyä, tarkistettiin vielä, että yhteys reunalaitteisiin oli toiminnassa, ennenkö aloitettiin itse ryhmän testaaminen.

### 8.6.4 GET VPN ja GROUP VPN yhteensopivuuden todennus

Topologian tarkastelu aloitettiin KS laitteelta, eli WG1-R1:ltä. Aluksi katsottiin olivatko molemmat jäsenlaitteet onnistuneet kirjautumaan KS:lle. Tämä tehtiin käyttäen komentoja *show crypto gdoi ks* ja *show cryto gdoi ks members.*

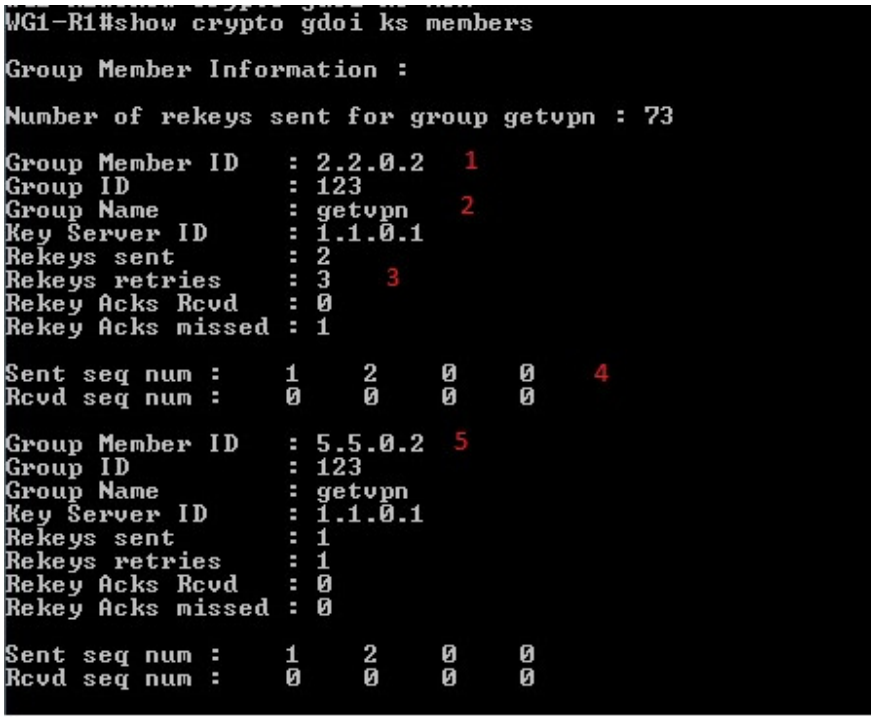
Kuviossa 51. nähdään KS reitittimen GDOI taulu. Kohdassa 1. Näkyy, että laitteelle on kirjautunut kaksi jäsentä ja kohdassa kaksi taas nähdään kuinka Junos vaatima Replay on otettu pois käytöstä.




KUVIO 51. WG1-R1 show crypto gdoi

Show crypto gdoi ks members, komento antaa tarkemman kuvan kaikista ryhmän jäsenistä. Kuvion 52. Kohdassa yksi nähdään WG2-R1, joka on saanu omaksi ID numerokseen oman ip-osoitteen, kohdassa kaksi puolestaan näkyy koko ryhmän ID ja nimi eli ID 123 ja nimi *getvpn*. Kohdassa kolme näkyy WG1-R1 ja WG2-R1 välillä tapahtuneet avainten vaihdot. Kohdassa neljä voidaan nähdä missä vaiheessa uudelleen lähetykset ovat menossa. Kohdassa viisi näkyy Juniper-R5:den tiedot, se on myös saanu jäsen ID:kseen oman ip-osoitteensa 5.5.0.2.

Show crypto gdoi ks members, komento antaa tarkemman kuvan kaikista ryhmän jäsenistä. Kuvion 52. Kohdassa yksi nähdään WG2-R1, joka on saanu omaksi ID numerokseen oman ip-osoitteen, kohdassa kaksi puolestaan näkyy koko ryhmän ID ja nimi eli ID 123 ja nimi *getvpn*. Kohdassa kolme näkyy WG1-R1 ja WG2-R1 välillä tapahtuneet avainten vaihdot. Kohdassa neljä voidaan nähdä missä vaiheessa uudelleen lähetykset ovat menossa. Kohdassa viisi näkyy Juniper-R5:den tiedot, se on myös saanu jäsen ID:kseen oman ip-osoitteensa 5.5.0.2.




KUVIO 52. GETvpn jäsenet

Seuraavaksi katsottiin olivatko molemmat jäsenlaitteet saaneet myös tarvitsemansa tiedot KS:ltä. Ensin katsottiin WG2-R1, koska voitiin olettaa, että Cisco laite tulisi ainakin toimimaan tässä ympäristössä.

Kuviossa 53. Kohdassa yksi nähdään että WG2-R1 on liittynyt samaan getvpn nimiseen ryhmään jonka tunnus on 123. Kohdassa kaksi näkyy sen hetkinen aktiivinen KS eli WG1-R1. Kohdassa kolme on molemmat KS:ltä ladatut access-list tiedot, joidenka perusteella liikenne tullaan salaamaan. Kohdassa neljä nähdään avainten uudelleen lähetystä varten muodostetut turva-assosiaatiot ja kohd assa viisi IPsec SA:t jotka on luotu ryhmää varten.

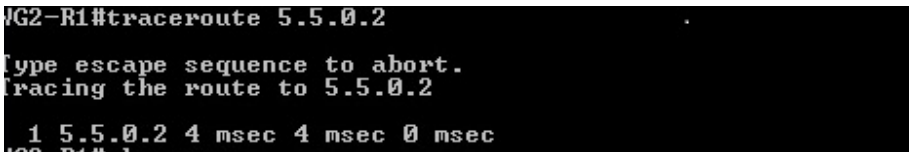
```
WG2-R1#show crypto gdoi
GROUP INFORMATION

    Group Name               : getvpn
    Group Identity           : 123      1
    Rekeys received          : 0
    IPSec SA Direction       : Both
    Active Group Server      : 1.1.0.2    2
    Group Server list        : 1.1.0.2

    GM Reregisters in        : 6910 secs
    Rekey Received(hh:mm:ss) : 01:28:09


    Rekeys received
         Cumulative          : 0
         After registration  : 0
    Rekey Acks sent          : 0

 ACL Downloaded From KS 1.1.0.2:
   access-list  permit ip host 2.2.0.2 host 5.5.0.2
   access-list  permit ip host 5.5.0.2 host 2.2.0.2   3

KEK POLICY:
    Rekey Transport Type     : Unicast
    Lifetime (secs)          : 284
    Encrypt Algorithm        : 3DES
    Key Size                 : 192                    4
    Sig Hash Algorithm       : HMAC_AUTH_SHA
    Sig Key Length (bits)    : 1024

TEK POLICY for the current KS-Policy ACEs Downloaded:
  GigabitEthernet0/0:
    IPsec SA:                                                  5
        spi: 0x44D1AA1D(1154591261)
        transform: esp-aes esp-sha-hmac
        sa timing:remaining key lifetime (sec): (1447)
        Anti-Replay : Disabled

    IPsec SA:
        spi: 0x8A0C480A(2316060682)
        transform: esp-aes esp-sha-hmac
        sa timing:remaining key lifetime (sec): (1530)
        Anti-Replay : Disabled

    IPsec SA:
        spi: 0xCF52E4B8(3478316216)
        transform: esp-aes esp-sha-hmac
        sa timing:remaining key lifetime (sec): (2251)
        Anti-Replay : Disabled

    IPsec SA:
        spi: 0xFD8F2664(4254017124)
        transform: esp-aes esp-sha-hmac
        sa timing:remaining key lifetime (sec): (7175)
        Anti-Replay : Disabled

WG2-R1#_
```

KUVIO 53. WG2-R1 show crypto gdoi

Kuviossa 54. nähdään kaikki eri SA:t, jotka on luotu laitteen liittyessä osaksi ryhmää. Kohdassa yksi nähdään IKE SA, jossa toisena päänä toimii 1.1.0.2 eli WG1-R1. Kuviossa kaksi on puolestaan IPsec SA, kuvassa näkyy selvästi kuinka serveri on jälleen 1.1.0.2 ja kohdassa kolme näkyy GId, eli Group ID joka on myös kyseisen GET VPN ryhmän ID; 123. Kohdassa neljä vielä KEK SA ja joka on myös tehty laitteen 1.1.0.2 kanssa, osana ryhmää 123.


KUVIO 54. Juniper-R4 turva-assosiaatiot

Seuraavaksi testattiin *ping* ja *traceroute* komennoilla kuinka liikenne muodostui pisteiden välille. Alla olevassa kuviossa 55. näkyy kuinka traceroute menee suoraa Juniper-R4:lle, eikä reitity internetin yli.




KUVIO 55. Traceroute WG2-R1 ja Juniper-R4 välillä

Lopuksi tarkasteltiin vielä *ping*-komentoa ajamalla, salautuuko liikenne. Kuviossa 56. näkyy komennon *show ipsec sa detail* antama taulukko. Sen kohdassa yksi nähdään liikenteen alku ja loppupää eli WG2-R1 ja Juniper-R4. Kohdassa kaksi näkyy niiden välillä kulkeneet paketit, jotka on onnistuneesti salattu ja purettu. Kohdassa kolme on kyseisen tunnelin käytössä olevat salaus –ja autentikointi-algoritmit. Kohdassa neljä nähdään vielä, kuinka kyseinen SA on sidoksissa krypto-karttaan *getvpn-map*. Kuviossa 57. vastaavat tiedot Juniper-R4:ltä

```
  protected vrf: (none)
  local  ident (addr/mask/prot/port): (2.2.0.2/255.255.255.255/0/0)  1
  remote ident (addr/mask/prot/port): (5.5.0.2/255.255.255.255/0/0)
  current_peer 0.0.0.0 port 848
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 28, #pkts encrypt: 28, #pkts digest: 28
   #pkts decaps: 28, #pkts decrypt: 28, #pkts verify: 28  2
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #pkts no sa (send) 0, #pkts invalid sa (rcv) 0
   #pkts encaps failed (send) 0, #pkts decaps failed (rcv) 0
   #pkts invalid prot (recv) 0, #pkts verify failed: 0
   #pkts invalid identity (recv) 0, #pkts invalid len (rcv) 0
   #pkts replay rollover (send): 0, #pkts replay rollover (rcv) 0
   ##pkts replay failed (rcv): 0
   #pkts internal err (send): 0, #pkts internal err (recv) 0

    local crypto endpt.: 2.2.0.2, remote crypto endpt.: 0.0.0.0
    path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/0
    current outbound spi: 0x322513DE(841290718)
    PFS (Y/N): N, DH group: none

    inbound esp sas:
     spi: 0x322513DE(841290718)
       transform: esp-aes esp-sha-hmac ,  3
       in use settings ={Tunnel, }
       conn id: 3877, flow_id: NETGX:1877, sibling_flags 80000040, crypto map:
getvpn-map    4
       sa timing: remaining key lifetime (sec): (3353)
       IV size: 16 bytes
       replay detection support: Y  replay window size: 0
       Status: ACTIVE
```

KUVIO 56. Show ipsec sa detail

```
root@Juniper-R4# run show security group-vpn member ipsec statistics
ESP Statistics:
  Encrypted bytes:            22656
  Decrypted bytes:            13296
  Encrypted packets:            162
  Decrypted packets:            162
AH Statistics:
  Input bytes:                    0
  Output bytes:                   0
  Input packets:                  0
  Output packets:                 0
Errors:
  AH authentication failures: 0, Replay errors: 0
  ESP authentication failures: 0, ESP decryption failures: 0
  Bad headers: 0, Bad trailers: 0
```

KUVIO 57. Juniper R4 ipsec statistics

# 9 YHTEENVETO

## 9.1 Opinnäytetyön tekeminen

Opinnäytetyön haastavuus avautui nopeasti varsinaisen tekemisen alettua. Itselläni ei ollut aikaisempaa kokemusta Juniper Networks laitteista, eikä myöskään CCNA-kursseja laajempaa osaamista Cisco Systems laitteista. Tästä syystä työn aikana tuli opeteltua, niin perusasioita uusiksi kuin myös tutustuttua erilaisiin vian paikannus –ja hallinta ongelmiin. Työn toimeksiantaja edustajana, sekä käytännön osuuden pääasiallisena ohjaajana toiminut Marko Vatanen, antoi työn alussa pikaisen ohjeistuksen Juniper Networks laitteille, sekä SpiderNet ympäristöön. Lisäksi sain kattavan määrän muuta materiaalia, jonka avulla pääsin alkuun. Vaikka koulutusohjelman aikana olikin käyty läpi laitteiden ja tekniikoiden perusteita, ilmeni työn aikana suuri määrä uutta opittavaa.

Työn tekeminen aloitettiin perehtymällä saatuun materiaaliin ja sen pohjalta hahmottelemalla mahdollisia testattavia asioita. Työn alussa oli tarkoitus tutkia Cisco Systemsin DMVPN sekä Juniperin Group VPN. Pian kuitenkin työhön lisättiin Cisco GETVPN sekä edellä mainitun ja Juniper Group VPN yhteentoimivuuden selvittäminen. Näiden pohjalta lähdettiin tekemään ensimmäisiä topologioita, jotka vielä muokkaantuivat työn edetessä.

Itse testausympäristön kanssa kului alussa aikaa, vanhojen asioiden mieleen palauttelun sekä yleiseen laitteisiin tutustumiseen. Jälkeenpäin voikin todeta, että pelkkä CCNA-kurssi oli varsin heikko pohja lähteä tekemään, näinkin käytännön painotteista työtä.

Työn tekemisen vaikein, mutta samalla opettavaisin kohta tuli vastaan Juniper Group VPN kanssa, kun annetut konfiguraatiot eivät toimineen useiden yritystenkään jälkeen. Tässä vaiheessa myös työnohjaajat avustivat ongelman etsimisessä, ja lopulta useiden sähköpostien ja palaverien jälkeen syyksi paljastui dokumentoimaton vika käytössä olleen Junos käyttöjärjestelmä version kanssa. Kun tämä oli saatu korjattua päivityksellä, jatkui työn tekeminen siltä osin ilman ongelmia. Myös Juniper ja Cisco laitteiden yhteensopivuuden kanssa ilmeni ongelmia, joidenka ratkaiseminen vaati hieman eri asioiden testaamista. Kummankaan laitevalmistajan omissa dokumenteissa ei varsinaisesti neu-

vottu kuinka kyseiset ongelmat voidaan ratkaista, vaan ilmoitettiin vain että konfiguraatio voi tarvita muutoksia.

Itselle opinnäytetyön tekeminen oli erittäin haastava projekti ja lopulta sen käytännön osuuden määrä yllätti. Se tarjosi kuitenkin paljon uutta ja opinkin todella paljon molempien valmistajien laitteista sekä niiden ongelmien selvittämisestä. Suurimpana asiana työn tekemisessä jäi harmittamaan sen venyminen, joka tosin johtui täysin työntekijästä itsestään ja hetkellisestä uskon loppumisesta omaan tekemiseen.

## 9.2 Tulokset ja tulevaisuus

Työ kasvoi sen edessä käsittelemään muutakin, kuin vain alussa ajatellut DMVPN ja Juniper Group VPN. Kaikki työhön lopulta otetut tekniikat saatiin kuitenkin testattua ja todennettua, vaikka välillä jouduttiinkin pysähtymään hetkeksi miettimään. Työn tuloksena saatiin toimivat ympäristöt, joidenka pohjalta tekniikoiden perusolemus voitiin päätellä. Tehtyjen konfiguraatioiden pohjalta voitiin lisäksi tehdä kaksi laboratorio harjoitus tietoverkko koulutusohjelman ja SpiderNet:in mahdollisia tarpeita varten.

Työssä keskityttiin vahvasti käytännön toteutukseen, teorian ollessa vain sen tukena ja uskonkin, että tehdyistä konfiguraatioista voi olla jatkossa apua, niin opiskelijoille kuin muille vastaavien teknologioiden toimintaan tutustuville.

Jatkoa ajatellen, voisi selvästi suuremman ympäristön toteuttaminen ja etenkin skaalautuvuuden tarkastelu olla vielä hyödyksi. Lisäksi DMVPN osalta reduntanttisuuden toteuttaminen voisi olla yksi tarkastelun kohde.

# LÄHTEET


Company profile 2011. Cisco Systems verkkosivut. Viitattu 25.10.2011.
http://www.cisco.com/web/about/ac49/ac20/ac19/ar2004/profile.html

DESIGNING SITE-TO-SITE IPSEC VPNS, NIL verkkosivut. Viitattu 10.11.2011.
http://stack.nil.com/ipcorner/IPsecVPN5/

Dynamic Multipoint VPN (DMVPN) Design Guide. 2011. Cisco Systems verkkosivut. Viitattu 24.10.2011

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns171/c649/ccmigration_09186a008075ea98.pdf

GET VPN design and implementation guide. 2011. Cisco Systems verkkosivut. Viitattu 10.11.2011
http://www.cisco.com/en/US/prod/collateral/vpndevc/ps6525/ps9370/ps7180/GETVPN_DIG_version_1_0_External.pdf

Implement IPv4 tunneling and Generic Routing Encapsulation (GRE). Viitattu 20.11.2011. Eric Leayh The World of Networking verkkosivu. http://ericleahy.com/?p=768

IPSEC project. Viitauttu 2011. Internet Protocol Security at the Telecommunications and Multimedia Laboratory verkkosivut http://www.tml.tkk.fi/Tutkimus/IPSEC/

Juniper Company Profile. 2011. Juniper Networks verkkosivut. Viitattu 25.10.2011.
http://www.juniper.net/us/en/local/pdf/fact-sheets-backgrounder/3000054-en.pdf

Junos Security 10.2. 2011. Juniper Networks verkkosivut. Viitattu 12.11.2011

http://www.juniper.net/techpubs/software/junos-security/junos-security10.2/junos-security-swconfig-security/topic-45799.html

Jyväskylän Ammattikorkeakoulu 2011 a. Viitattu 6.7.2011. www.jamk.fi/tutustu

Jyväskylän Ammattikorkeakoulu 2011 b. Viitattu 6.7.2011. Tietotekniikan koulutusohjelma

Pepelnjak I, 2011. DMVPN: From Basics to Scalable Networks seminaarinaineisto

RFC 2332. 1998. Viitattu 21.11.2011 http://www.ietf.org/rfc/rfc2332.txt

RFC 2401. 1998. Viitattu 20.11.2011 http://www.ietf.org/rfc/rfc2401.txt

RFC 2402. 1998. Viitattu 20.11.2011 http://www.ietf.org/rfc/rfc2402.txt

RFC 2784. 2000. Viitattu 21.11.2011 http://www.ietf.org/rfc/rfc2784.txt

RFC 3547. 2003. Viitattu 5.11.2011 http://www.ietf.org/rfc/rfc3547.txt

RFC 4303. 2005. Viitattu 3.3.2012 http://www.ietf.org/rfc/rfc4303.txt

RFC 4306. 2005. Viitattu 3.3.2012 http://www.ietf.org/rfc/rfc4306.txt

SpiderNet 2011, viitattu 6.7.2011. http://student.labranet.jamk.fi/SpiderNet/

Strech J, Dynamic Multipoint VPN blog. Viitattu 24.10.2011
http://packetlife.net/blog/2008/jul/23/dynamic-multipoint-vpn-dmvpn/

VPN Basics: Internet Protocol Security (IPSec). Viitattu 25.11.2011. Netgear verkkosivut
http://support.netgear.com/app/answers/detail/a_id/19030/~/vpn-basics%3A-internet-protocol-security-(ipsec)

# LIITTEET

## Liite 1. Cisco Core-R1, DMVPN konfiguraatio DHCP

```
CORE-R1#show run
Building configuration...

Current configuration : 1900 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CORE-R1
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
memory-size iomem 25
!
!
ip cef
no ip dhcp use vrf connected
ip dhcp excluded-address 2.2.0.1 2.2.0.10
!
ip dhcp pool WG1
   network 2.2.0.0 255.255.255.0
   default-router 2.2.0.1
!
interface Loopback0
 ip address 130.0.1.2 255.255.255.252
!
interface ATM0/0
 no ip address
 shutdown
 no atm ilmi-keepalive
!
interface FastEthernet1/0
 description Link to WG1-R1
 ip address 2.2.0.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial1/0
 no ip address
!
interface Serial1/1
 no ip address
```

```
 shutdown
 clock rate 2000000
!
interface FastEthernet2/0
 no ip address
 shutdown
 half-duplex
!
interface FastEthernet3/0
 description Link To CORE-R2
 no switchport
 ip address 172.0.0.2 255.255.255.252
!
interface FastEthernet3/1
 description Link to CORE-R3
 no switchport
 ip address 172.0.0.5 255.255.255.252
!
interface FastEthernet3/2
!
interface FastEthernet3/3
!
interface FastEthernet3/4
!
interface FastEthernet3/5
!
interface FastEthernet3/6
!
interface FastEthernet3/7
!
interface FastEthernet3/8
!
interface FastEthernet3/9
!
interface FastEthernet3/10
!
interface FastEthernet3/11
!
interface FastEthernet3/12
!
interface FastEthernet3/13
!
interface FastEthernet3/14
!
interface FastEthernet3/15
!
interface GigabitEthernet3/0
!
interface Vlan1
 no ip address
!
router ospf 1
 log-adjacency-changes
 redistribute static metric-type 1
 network 2.2.0.0 0.0.0.255 area 0
```

```
 network 130.0.1.0 0.0.0.3 area 0
 network 172.0.0.0 0.0.0.3 area 0
 network 172.0.0.4 0.0.0.3 area 0
 network 172.0.0.8 0.0.0.3 area 0
!
ip http server
no ip http secure-server
!
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
!
end

CORE-R1#
```

## Liite 2. Cisco Core-R2 DMVPN

```
CORE-R2#show run
Building configuration...

Current configuration : 1775 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CORE-R2
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
memory-size iomem 25
!
!
ip cef
no ip domain lookup
!
interface Loopback0
 ip address 130.0.2.1 255.255.255.252
!
interface ATM0/0
 no ip address
 shutdown
 no atm ilmi-keepalive
```

```
!
interface FastEthernet1/0
 description Link to WG2-R1
 ip address 200.10.2.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial1/0
 no ip address
 shutdown
 no fair-queue
!
interface Serial1/1
 no ip address
 shutdown
 clock rate 2000000
!
interface FastEthernet2/0
 no ip address
 shutdown
 half-duplex
!
interface FastEthernet3/0
 description Link to CORE-R3
 no switchport
 ip address 172.0.0.9 255.255.255.252
!
interface FastEthernet3/1
 description Link to CORE-R1
 no switchport
 ip address 172.0.0.1 255.255.255.252
!
interface FastEthernet3/2
!
interface FastEthernet3/3
!
interface FastEthernet3/4
!
interface FastEthernet3/5
!
interface FastEthernet3/6
!
interface FastEthernet3/7
!
interface FastEthernet3/8
!
interface FastEthernet3/9
!
interface FastEthernet3/10
!
interface FastEthernet3/11
!
interface FastEthernet3/12
!
interface FastEthernet3/13
```

```
!
interface FastEthernet3/14
!
interface FastEthernet3/15
!
interface GigabitEthernet3/0
!
interface Vlan1
 no ip address
!
router ospf 1
 log-adjacency-changes
 redistribute static metric-type 1 subnets
 network 172.0.0.0 0.0.0.3 area 0
 network 172.0.0.4 0.0.0.3 area 0
 network 172.0.0.8 0.0.0.3 area 0
 network 200.10.2.0 0.0.0.255 area 0
!
ip http server
no ip http secure-server
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
!
end

CORE-R2#
```

## Liite 3. Cisco Core-R3 DMVPN konfiguraatio DHCP

```
CORE-R3#show run
Building configuration...

Current configuration : 1942 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname CORE-R3
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
memory-size iomem 25
```

```
!
!
ip cef
no ip domain lookup
no ip dhcp use vrf connected
ip dhcp excluded-address 3.3.0.1 3.3.0.10
!
ip dhcp pool WG3
  network 3.3.0.0 255.255.255.0
  default-router 3.3.0.1
!
interface Loopback0
 ip address 130.0.3.2 255.255.255.252
!
interface ATM0/0
 no ip address
 shutdown
 no atm ilmi-keepalive
!
interface FastEthernet1/0
 description Link to WG3-R1
 ip address 3.3.0.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial1/0
 no ip address
 shutdown
 no fair-queue
!
interface Serial1/1
 no ip address
 shutdown
 clock rate 2000000
!
interface FastEthernet2/0
 no ip address
 shutdown
 half-duplex
!
interface FastEthernet3/0
 description Link to CORE-R1
 no switchport
 ip address 172.0.0.6 255.255.255.252
!
interface FastEthernet3/1
 description Link to CORE-R2
 no switchport
 ip address 172.0.0.10 255.255.255.252
!
interface FastEthernet3/2
!
interface FastEthernet3/3
!
interface FastEthernet3/4
```

```
!
interface FastEthernet3/5
!
interface FastEthernet3/6
!
interface FastEthernet3/7
!
interface FastEthernet3/8
!
interface FastEthernet3/9
!
interface FastEthernet3/10
!
interface FastEthernet3/11
!
interface FastEthernet3/12
!
interface FastEthernet3/13
!
interface FastEthernet3/14
!
interface FastEthernet3/15
!
interface GigabitEthernet3/0
 no switchport
 no ip address
!
interface Vlan1
 no ip address
!
router ospf 1
 log-adjacency-changes
 redistribute static metric-type 1
 network 3.3.0.0 0.0.0.255 area 0
 network 172.0.0.0 0.0.0.3 area 0
 network 172.0.0.4 0.0.0.3 area 0
 network 172.0.0.8 0.0.0.3 area 0
!
ip http server
no ip http secure-server
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
!
end

CORE-R3#
```

## Liite 4. Cisco WG1-R1 DMVPN

```
WG1-R1#show run
Building configuration...


Current configuration : 1981 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname WG1-R1
!
boot-start-marker
boot-end-marker
!
logging message-counter syslog
!
no aaa new-model
!
dot11 syslog
ip source-route
!
!
ip cef
!
!
no ipv6 cef
!
multilink bundle-name authenticated
!
voice-card 0
!
archive
 log config
  hidekeys
!
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0
!
!
crypto ipsec transform-set dmvpn123 esp-3des esp-md5-hmac
 mode transport
!
crypto ipsec profile dmvpn
 set transform-set dmvpn123
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.2 255.255.255.0
```

```
 no ip redirects
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map 10.0.0.1 200.10.2.2
 ip nhrp map multicast 200.10.2.2
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 ip ospf network broadcast
 ip ospf priority 0
 delay 1000
 keepalive 5 4
 tunnel source GigabitEthernet0/0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile dmvpn
!
interface GigabitEthernet0/0
 description Link to CORE-R1
 ip address dhcp
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 description Link to wg1-sw1
 ip address 192.168.1.1 255.255.255.0
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0/0
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
!
interface Serial0/0/1
 no ip address
 shutdown
!
interface FastEthernet0/1/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
router ospf 2
 log-adjacency-changes
 network 10.0.0.0 0.0.0.255 area 0
```

```
 network 192.168.1.0 0.0.0.255 area 0
!
ip forward-protocol nd
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
end

WG1-R1#
```

## Liite 5. Cisco WG2-R1 DMVPN

```
WG2-R1#show run
Building configuration...


Current configuration : 2097 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname WG2-R1
!
boot-start-marker
boot-end-marker
!
logging message-counter syslog
!
no aaa new-model
!
dot11 syslog
ip source-route
!
ip cef
!
no ipv6 cef
!
multilink bundle-name authenticated
!
voice-card 0
!
vtp mode transparent
archive
```

```
 log config
  hidekeys
!
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0
crypto isakmp invalid-spi-recovery
!
!
crypto ipsec transform-set dmvpn123 esp-3des esp-md5-hmac
 mode transport
!
crypto ipsec profile dmvpn
 set transform-set dmvpn123
!
interface Loopback0
 ip address 130.0.10.1 255.255.255.252
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.1 255.255.255.0
 no ip redirects
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map multicast dynamic
 ip nhrp network-id 100000
 ip nhrp holdtime 600
 ip ospf network broadcast
 ip ospf priority 2
 delay 1000
 keepalive 5 4
 tunnel source GigabitEthernet0/0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile dmvpn
!
interface GigabitEthernet0/0
 ip address 200.10.2.2 255.255.255.0
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 description link to wg2-sw1
 ip address 192.168.2.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/0/0
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
!
interface Serial0/0/1
```

```
 no ip address
 shutdown
!
interface FastEthernet0/1/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
router ospf 1
 log-adjacency-changes
 network 192.168.0.0 0.0.0.255 area 0
 network 200.10.2.0 0.0.0.255 area 0
!
router ospf 2
 log-adjacency-changes
 network 10.0.0.0 0.0.0.255 area 0
!
ip forward-protocol nd
no ip http server
no ip http secure-server
!
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
end
```

## Liite 6. Cisco WG3-R1 DMVPN

```
WG3-R1#show run
Building configuration...


Current configuration : 1981 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname WG3-R1
```

```
!
boot-start-marker
boot-end-marker
!
logging message-counter syslog
!
no aaa new-model
!
dot11 syslog
ip source-route
!
ip cef
!
!
no ipv6 cef
!
multilink bundle-name authenticated
!
voice-card 0
!
archive
 log config
  hidekeys
!
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0
!
!
crypto ipsec transform-set dmvpn123 esp-3des esp-md5-hmac
 mode transport
!
crypto ipsec profile dmvpn
 set transform-set dmvpn123
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.3 255.255.255.0
 no ip redirects
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map 10.0.0.1 200.10.2.2
 ip nhrp map multicast 200.10.2.2
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 ip ospf network broadcast
 ip ospf priority 0
 delay 1000
 keepalive 5 4
 tunnel source GigabitEthernet0/0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile dmvpn
```

```
!
interface GigabitEthernet0/0
 description Link to CORE-R3
 ip address dhcp
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 description link to wg3-sw1
 ip address 192.168.3.1 255.255.255.0
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0/0
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
!
interface Serial0/0/1
 no ip address
 shutdown
!
interface FastEthernet0/1/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
router ospf 2
 log-adjacency-changes
 network 10.0.0.0 0.0.0.255 area 0
 network 192.168.3.0 0.0.0.255 area 0
!
ip forward-protocol nd
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
end
```

```
WG3-R1#
```

## Liite 7. Group VPN Juniper-R4

```
root# show
## Last changed: 2012-06-03 15:41:45 UTC
version 11.2R5.4;
system {
    host-name Juniper-R4;
    root-authentication {
        encrypted-password "$1$v0YaCGuz$7Dy9/WWQ/DMxxKqtUNQ2W1"; ## SECRET-DATA
    }
    syslog {
        user * {
            any emergency;
        }
        file messages {
            any any;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    ge-1/0/1 {
        unit 0;
    }
    ge-1/0/2 {
        unit 0 {
            family inet {
                address 200.10.4.2/24;
            }
        }
    }
    ge-1/0/3 {
        unit 0 {
            family inet {
                address 192.168.40.1/24;
            }
        }
    }
    ge-1/0/4 {
        unit 0;
    }
    ge-1/0/5 {
        unit 0;
    }
    ge-1/0/6 {
```

```
      unit 0;
    }
    ge-1/0/7 {
      unit 0;
    }
    lo0 {
      unit 0 {
        family inet {
          address 1.1.1.1/32;
        }
      }
    }
}
routing-options {
    static {
      route 0.0.0.0/0 next-hop 200.10.4.1;
    }
}
security {
    group-vpn {
      member {
        ike {
          proposal prop1 {
            authentication-method pre-shared-keys;
            dh-group group2;
            authentication-algorithm sha1;
            encryption-algorithm 3des-cbc;
          }
          policy pol1 {
            mode aggressive;
            proposals prop1;
            pre-shared-key ascii-text "$9$xZr-b2ZUH5Qn4aQn/CB17-V"; ## S
ECRET-DATA
          }
          gateway g1 {
            ike-policy pol1;
            address 200.10.4.2;
            local-address 192.168.40.1;
          }
        }
        ipsec {
          vpn v1 {
            ike-gateway g1;
            group-vpn-external-interface ge-1/0/2.0;
            group 1;
          }
        }
      }
      server {
        ike {
          proposal srv-prop {
            authentication-method pre-shared-keys;
            dh-group group2;
            authentication-algorithm sha1;
            encryption-algorithm 3des-cbc;
```

```
            }
            policy srv-pol {
                mode aggressive;
                proposals srv-prop;
                pre-shared-key ascii-text "$9$UViqf36A1RSTzRSreXxDik"; ## SE
CRET-DATA
            }
            gateway gw1 {
                ike-policy srv-pol;
                address 200.10.4.2;
            }
            gateway gw2 {
                ike-policy srv-pol;
                address 200.10.5.2;
            }
            gateway srv-gw {
                ike-policy srv-pol;
                address 192.168.40.1;
            }
        }
        ipsec {
            proposal group-prop {
                authentication-algorithm hmac-sha1-96;
                encryption-algorithm 3des-cbc;
                lifetime-seconds 3600;
            }
        }
        group grp1 {
            group-id 1;
            ike-gateway srv-gw;
            ike-gateway gw1;
            ike-gateway gw2;
            anti-replay-time-window 120;
            server-address 200.10.4.2;
            server-member-communication {
                communication-type unicast;
                encryption-algorithm aes-128-cbc;
                sig-hash-algorithm md5;
                certificate srv-cert;
            }
            ipsec-sa group-sa {
                }
                match-policy pol1 {
                    source 192.168.40.0/24;
                    destination 192.168.50.0/24;
                    source-port 0;
                    destination-port 0;
                    protocol 0;
                }
                match-policy pol2 {
                    source 192.168.50.0/24;
                    destination 192.168.40.0/24;
                    source-port 0;
                    destination-port 0;
                    protocol 0;
```

```
            }
          }
        }
      }
      co-location;
    }
  policies {
      from-zone trust to-zone untrust {
        policy groupvpn {
          match {
            source-address any;
            destination-address any;
            application any;
          }
          then {
            permit {
              tunnel {
                ipsec-group-vpn v1;
              }
            }
          }
        }
        policy deny-all {
          match {
            source-address any;
            destination-address any;
            application any;
          }
          then {
            deny;
          }
        }
      }
      from-zone untrust to-zone trust {
        policy groupvpn {
          match {
            source-address any;
            destination-address any;
            application any;
          }
          then {
            permit {
              tunnel {
                ipsec-group-vpn v1;
              }
            }
          }
        }
        policy deny {
          match {
            source-address any;
            destination-address any;
            application any;
          }
          then {
```

```
            deny;
          }
        }
      }
      default-policy {
        deny-all;
      }
    }
```

## Liite 8. Group VPN Juniper-R5

```
root@Juniper-R5# show
## Last changed: 2012-06-03 15:39:02 UTC
version 11.2R5.4;
system {
    host-name Juniper-R5;
    root-authentication {
        encrypted-password "$1$v0YaCGuz$7Dy9/WWQ/DMxxKqtUNQ2W1"; ## SECRET-DATA
    }
    syslog {
        user * {
            any emergency;
        }
        file messages {
            any any;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    ge-1/0/0 {
        unit 0 {
            family inet {
                address 192.168.50.1/24;
            }
        }
    }
    ge-1/0/1 {
        unit 0;

    }
    ge-1/0/2 {
        unit 0 {
            family inet {
                address 200.10.5.2/24;
            }
```

```
        }
    }
    ge-1/0/3 {
        unit 0;
    }
    ge-1/0/4 {
        unit 0;
    }
    ge-1/0/5 {
        unit 0;
    }
    ge-1/0/6 {
        unit 0;
    }
    ge-1/0/7 {
        unit 0;
    }
    lo0 {
        unit 0 {
            family inet {
                address 2.2.2.2/24;
            }
        }
    }
}
routing-options {
    static {
        route 0.0.0.0/0 next-hop 200.10.5.1
    }
}
security {
    group-vpn {
        member {
            ike {
                proposal prop2 {
                    authentication-method pre-shared-keys;
                    dh-group group2;
                    authentication-algorithm sha1;
                    encryption-algorithm 3des-cbc;
                }
                policy pol2 {
                    mode aggressive;
                    proposals prop2;
                    pre-shared-key ascii-text "$9$cYfrK8-VYZUHX7UHqmF3Sre"; ## S
ECRET-DATA
                }
                gateway g2 {
                    ike-policy pol2;
                    address 200.10.4.2;
                    local-address 200.10.5.2;
                }
            }
            ipsec {
                vpn v1 {
                    ike-gateway g2;
```

```
            group-vpn-external-interface ge-1/0/2.0;
            group 1;
        }
    }
  }
}
policies {
  from-zone trust to-zone untrust {
    policy groupvpn {
      match {
        source-address any;
        destination-address any;
        application any;
      }
      then {
        permit {
          tunnel {
            ipsec-group-vpn v1;
          }
        }
      }
    }
    policy deny-all {
      match {
        source-address any;
        destination-address any;
        application any;
      }
      then {
        deny;
      }
    }
  }
  from-zone untrust to-zone trust {
    policy groupvpn {
      match {
        source-address any;
        destination-address any;
        application any;
      }
      then {
        permit {
          tunnel {
            ipsec-group-vpn v1;
          }
        }
      }
    }
    policy deny {
      match {
        source-address any;
        destination-address any;
        application any;
      }
      then {
```

```
        deny;
      }
    }
}
default-policy {
   deny-all;
}
```

## Liite 9. Group VPN Juniper-R1

Welcome to SpiderNet, press ENTER to continue

```
[edit]
root@Juniper-R1# show
## Last changed: 2012-05-10 08:05:19 UTC
version 11.2R5.4;
system {
   root-authentication {
      encrypted-password "$1$MScTnDbq$OIhEQnQsrJz7Nk94aPnKt/"; ## SECRET-DATA
   }
}
interfaces {
   ge-0/0/0 {
      unit 0;
   }
   ge-1/0/1 {
      unit 0 {
         family inet {
            address 192.168.50.2/24;
         }
      }
   }
}
routing-options {
   static {
      route 0.0.0.0/0 next-hop 192.168.50.1;
   }
}
security {
   policies {
      from-zone Salli to-zone Salli {
         policy permit {
            match {
               source-address any;
               destination-address any;
               application any;
            }
            then {
               permit;
            }
         }
      }
   }
   zones {
      security-zone Salli {
         host-inbound-traffic {
            system-services {
               all;
            }
```

```
        }
        interfaces {
           ge-1/0/1.0 {
              host-inbound-traffic {
                 protocols {
                    all;
                 }
              }
           }
        }
     }
  }
}

[edit]
root@Juniper-R1#
```

## Liite 10. Group VPN Juniper-R2

```
[edit]
root@Juniper-R2# show
## Last changed: 2012-05-10 07:36:37 UTC
version 11.2R5.4;
system {
   root-authentication {
      encrypted-password "$1$v0YaCGuz$7Dy9/WWQ/DMxxKqtUNQ2W1"; ## SECRET-DATA
   }
   syslog {
      user * {
         any emergency;
      }
      file messages {
         any any;
         authorization info;
      }
      file interactive-commands {
         interactive-commands any;
      }
   }
   license {
      autoupdate {
         url https://ae1.juniper.net/junos/key_retrieval;
      }
   }
}
interfaces {
   ge-1/0/0 {
      unit 0;
   }
   ge-1/0/1 {
      unit 0;
   }
   ge-1/0/2 {
```

```
        unit 0 {
            family inet {
                address 192.168.40.2/24;
            }
        }
    }
    ge-1/0/4 {
        unit 0;
    }
    ge-1/0/5 {
        unit 0;
    }
    ge-1/0/6 {
        unit 0;
    }
    ge-1/0/7 {
        unit 0;
    }
    lo0 {
        unit 0;
    }
}
routing-options {
    static {
        route 0.0.0.0/0 next-hop 192.168.40.1;
    }
}
security {
    policies {
        default-policy {
            permit-all;
        }
    }
    zones {
        security-zone All {
            host-inbound-traffic {
                system-services {
                    all;
                }
                protocols {
                    all;
                }
            }
            interfaces {
                all;
            }
        }
    }
}
```

## Liite 11. GET VPN Cisco WG1-R1

```
WG-R1#show run
```

```
Building configuration...


Current configuration : 2151 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname WG-R1
!
boot-start-marker
boot-end-marker
!
logging message-counter syslog
!
no aaa new-model
memory-size iomem 25
!
dot11 syslog
ip source-route
!
!
ip cef
!
no ip domain lookup
no ipv6 cef
!
multilink bundle-name authenticated
!
voice-card 0
!
archive
 log config
  hidekeys
!
!
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 2
crypto isakmp key getvpn address 1.1.0.2
crypto isakmp key getvpn address 2.2.0.2
crypto isakmp key getvpn address 3.3.0.2
crypto isakmp key getvpn address 4.4.0.3
crypto isakmp keepalive 15 periodic
!
!
crypto ipsec transform-set getvpn-gdoi esp-aes esp-sha-hmac
!
crypto ipsec profile getvpn
 set security-association lifetime seconds 7200
 set transform-set getvpn-gdoi
!
```

```
crypto gdoi group getvpn
 identity number 123
 server local
  rekey lifetime seconds 300
  rekey retransmit 40 number 2
  rekey authentication mypubkey rsa getvpn-export-avain
  rekey transport unicast
  sa ipsec 1
   profile getvpn
   match address ipv4 109
   replay time window-size 5
  address ipv4 1.1.0.2
  redundancy
   local priority 100
   peer address ipv4 2.2.0.2
!
interface GigabitEthernet0/0
 description Link To CORE-R1
 ip address 1.1.0.2 255.255.255.0
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0/0
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
!
interface Serial0/0/1
 no ip address
 shutdown
!
interface FastEthernet0/1/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 1.1.0.1
no ip http server
no ip http secure-server
!
```

```
!
!
access-list 109 permit ip 3.3.0.0 0.0.0.255 4.4.0.0 0.0.0.255
access-list 109 permit ip 4.4.0.0 0.0.0.255 3.3.0.0 0.0.0.255
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
end
```

## Liite 12. GET VPN Cisco WG2-R1

```
WG2-R1#show run
Building configuration...


Current configuration : 2193 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname WG2-R1
!
boot-start-marker
boot-end-marker
!
logging message-counter syslog
!
no aaa new-model
!
dot11 syslog
ip source-route
!
!
ip cef
!
!
no ipv6 cef
!
multilink bundle-name authenticated
!
voice-card 0
!
vtp mode transparent
archive
```

```
 log config
  hidekeys
!
!
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 2
crypto isakmp key getvpn address 1.1.0.2
crypto isakmp key getvpn address 2.2.0.2
crypto isakmp key getvpn address 3.3.0.2
crypto isakmp key getvpn address 4.4.0.3
crypto isakmp keepalive 15 periodic
!
!
crypto ipsec transform-set getvpn-gdoi esp-aes esp-sha-hmac
!
crypto ipsec profile getvpn
 set security-association lifetime seconds 7200
 set transform-set getvpn-gdoi
!
crypto gdoi group getvpn
 identity number 123
 server local
  rekey lifetime seconds 300
  rekey retransmit 40 number 2
  rekey authentication mypubkey rsa getvpn-export-avain
  rekey transport unicast
  sa ipsec 1
   profile getvpn
   match address ipv4 109
   replay time window-size 5
  address ipv4 2.2.0.2
  redundancy
   local priority 75
   peer address ipv4 1.1.0.2
!
interface GigabitEthernet0/0
 description Link To CORE-R2
 ip address 2.2.0.2 255.255.255.0
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0/0
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
!
```

```
interface Serial0/0/1
 no ip address
 shutdown
!
interface FastEthernet0/1/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 2.2.0.1
no ip http server
no ip http secure-server
!
access-list 109 permit ip 1.1.0.0 0.0.0.255 2.2.0.0 0.0.0.255
access-list 109 permit ip 3.3.0.0 0.0.0.255 4.4.0.0 0.0.0.255
access-list 109 permit ip 4.4.0.0 0.0.0.255 3.3.0.0 0.0.0.255
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
end

WG2-R1#
```

## Liite 13. GET VPN Cisco WG3-R1

```
WG3-R1#show run
Building configuration...


Current configuration : 1536 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname WG3-R1
```

```
!
boot-start-marker
boot-end-marker
!
logging message-counter syslog
!
no aaa new-model
!
dot11 syslog
ip source-route
!
!
ip cef
!
no ip domain lookup
no ipv6 cef
!
multilink bundle-name authenticated
!
voice-card 0
!
archive
 log config
  hidekeys
!
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 2
 lifetime 1200
crypto isakmp key getvpn address 1.1.0.2
crypto isakmp key getvpn address 2.2.0.2
!
crypto gdoi group getvpn
 identity number 123
 server address ipv4 1.1.0.2
 server address ipv4 2.2.0.2
!
crypto map getvpn-map 10 gdoi
 set group getvpn
!
interface GigabitEthernet0/0
 description Link to CORE-R3
 ip address 3.3.0.2 255.255.255.0
 duplex auto
 speed auto
 crypto map getvpn-map
!
interface GigabitEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0/0
```

```
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
!
interface Serial0/0/1
 no ip address
 shutdown
!
interface FastEthernet0/1/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 3.3.0.1
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
```

## Liite 14. GET VPN Cisco WG4-R1

```
WG4-R1#show run
Building configuration...

Current configuration : 1548 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname WG4-R1
!
boot-start-marker
boot-end-marker
!
logging message-counter syslog
!
```

```
no aaa new-model
!
dot11 syslog
ip source-route
!
!
ip cef
!
no ip domain lookup
no ipv6 cef
!
multilink bundle-name authenticated
!
voice-card 0
!
archive
 log config
  hidekeys
!
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 2
 lifetime 1200
crypto isakmp key getvpn address 1.1.0.2
crypto isakmp key getvpn address 2.2.0.2
!
crypto gdoi group getvpn
 identity number 123
 server address ipv4 1.1.0.2
 server address ipv4 2.2.0.2
!
crypto map getvpn-map 10 gdoi
 set group getvpn
!
interface GigabitEthernet0/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0/0
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
!
interface Serial0/0/1
 no ip address
```

```
 shutdown
!
interface FastEthernet0/1/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1/1
 ip address 4.4.0.3 255.255.255.0
 duplex auto
 speed auto
 vlan-id dot1q 989
  exit-vlan-config
 !
 crypto map getvpn-map
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 4.4.0.1
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
```

## Liite 14. Juniper ja Cisco. Juniper-R4

```
version 11.2R5.4;
system {
   root-authentication {
      encrypted-password "$1$v0YaCGuz$7Dy9/WWQ/DMxxKqtUNQ2W1"; ## SECRET-DATA
   }
   syslog {
      user * {
         any emergency;
      }
      file messages {
         any any;
         authorization info;
      }
      file interactive-commands {
         interactive-commands any;
      }
   }
   license {
      autoupdate {
         url https://ae1.juniper.net/junos/key_retrieval;
```

```
      }
    }
}
interfaces {
    ge-1/0/0 {
        unit 0;
    }
    ge-1/0/1 {
        unit 0;
    }
    ge-1/0/2 {
        unit 0 {
            family inet {
                address 5.5.0.2/24;
            }
        }
    }
    ge-1/0/3 {
        unit 0;
    }
    ge-1/0/4 {
        unit 0;
    }
    ge-1/0/5 {
        unit 0;
    }
    ge-1/0/6 {
        unit 0;
    }
    ge-1/0/7 {
        unit 0 {
    }
    lo0 {
        unit 0;
    }
}
routing-options {
    static {
        route 0.0.0.0/0 next-hop 5.5.0.1;
    }
}
security {
    ipsec {
        traceoptions {
            flag all;
        }
    }
    group-vpn {
        member {
            ike {
                proposal prop2 {
                    authentication-method pre-shared-keys;
                    dh-group group2;
                    authentication-algorithm sha1;
                    encryption-algorithm aes-128-cbc;
```

```
            }
            policy getvpn {
                mode main;
                proposals prop2;
                pre-shared-key ascii-text "$9$0F7xIEyMWxVs4Ndi.fzCA"; ## SEC
RET-DATA
            }
            gateway g2 {
                ike-policy getvpn;
                address 1.1.0.2;
                local-address 5.5.0.2;
            }
        }
        ipsec {
            vpn v1 {
                ike-gateway g2;
                group-vpn-external-interface ge-1/0/2.0;
                group 123;
            }
        }
    }
  }
  policies {
    from-zone trust to-zone untrust {
        policy getvpn {
            match {
                source-address any;
                destination-address any;
                application any;
            }
            then {
                permit {
                    tunnel {
                        ipsec-group-vpn v1;
                    }
                }
            }
        }
    }
    from-zone untrust to-zone trust {
        policy getvpn {
            match {
                source-address any;
                destination-address any;
                application any;
            }
            then {
                permit {
                    tunnel {
                        ipsec-group-vpn v1;
                    }
                }
            }
        }
    }
```

```
    default-policy {
      permit-all;
    }
  }
  traceoptions {
    flag all;
  }
  zones {
    security-zone All {
      host-inbound-traffic {
        system-services {
          all;
        }
        protocols {
          all;
        }
      }
      interfaces {
        all;
      }
    }
    security-zone trust;
    security-zone untrust;
  }
}

[edit]
root#
```

## Liite 15. Juniper ja Cisco. Cisco WG1-R1

```
Current configuration : 1953 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname WG1-R1
!
boot-start-marker
boot-end-marker
!
logging message-counter syslog
!
no aaa new-model
memory-size iomem 25
!
dot11 syslog
ip source-route
!
ip cef
!
```

```
no ipv6 cef
!
multilink bundle-name authenticated
!
voice-card 0
!
archive
 log config
  hidekeys
!
!
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 2
crypto isakmp key getvpn address 2.2.0.2
crypto isakmp key getvpn address 5.5.0.2
crypto isakmp invalid-spi-recovery
!
!
crypto ipsec transform-set getvpn-trans esp-aes esp-sha-hmac
!
crypto ipsec profile getvpn-gdoi
 set security-association lifetime seconds 7200
 set transform-set getvpn-trans
!
crypto gdoi group getvpn
 identity number 123
 server local
  rekey lifetime seconds 300
  rekey retransmit 40 number 2
  rekey authentication mypubkey rsa getvpn-export
  rekey transport unicast
  sa ipsec 1
   profile getvpn-gdoi
   match address ipv4 109
   no replay
  address ipv4 1.1.0.1
!
interface GigabitEthernet0/0
 description Link-to-CORE1
 ip address 1.1.0.2 255.255.255.0
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0/0
 no ip address
 shutdown
 no fair-queue
```

```
 clock rate 2000000
!
interface Serial0/0/1
 no ip address
 shutdown
!
interface FastEthernet0/1/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 1.1.0.1
no ip http server
no ip http secure-server
!
access-list 109 permit ip host 2.2.0.2 host 5.5.0.2
access-list 109 permit ip host 5.5.0.2 host 2.2.0.2
!
control-plane
!

line con 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
end

WG1-R1#
```

## Liite 16. Juniper ja Cisco. Cisco WG2-R1

```
Current configuration : 1465 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname WG2-R1
!
boot-start-marker
boot-end-marker
!
```

```
logging message-counter syslog
!
no aaa new-model
!
dot11 syslog
ip source-route
!
ip cef
!
no ipv6 cef
!
multilink bundle-name authenticated
!
voice-card 0
!
vtp mode transparent
archive
 log config
  hidekeys
!
!
crypto isakmp policy 10
 encr aes
 authentication pre-share
 group 2
 lifetime 1200
crypto isakmp key getvpn address 1.1.0.2
!
!
crypto gdoi group getvpn
 identity number 123
 server address ipv4 1.1.0.2
!
!
crypto map getvpn-map 10 gdoi
 set group getvpn
!
interface GigabitEthernet0/0
 description Link-to-CORE2
 ip address 2.2.0.2 255.255.255.0
 duplex auto
 speed auto
 crypto map getvpn-map
!
interface GigabitEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial0/0/0
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
```

```
!
interface Serial0/0/1
 no ip address
 shutdown
!
interface FastEthernet0/1/0
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface FastEthernet0/1/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
ip forward-protocol nd
ip route 0.0.0.0 0.0.0.0 2.2.0.1
no ip http server
no ip http secure-server
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
end
```

Jyväskylän Ammattikorkeakoulu
Teknologia / ICT

# *Liite 17. DMVPN-Harjoitus*



## JOHDANTO

### *Yleistä*

Tämän harjoituksen tarkoituksena on toteuttaa ja tarkastella Cisco Systems kehittämää VPN ratkaisua; Dynamic Multipoint VPN:ää (DMVPN). Tavallisen point-to-point VPN sijaa DMVPN mahdollistaa dynaamisesti muodostettavat tunnelit määriteltyjen laitteiden kesken. Koska DMVPN on Cisco Systems patentoima ratkaisu, on sen käyttö mahdollista vain ympäristössä jossa kaikki laitteet ovat Cisco valmistamia.

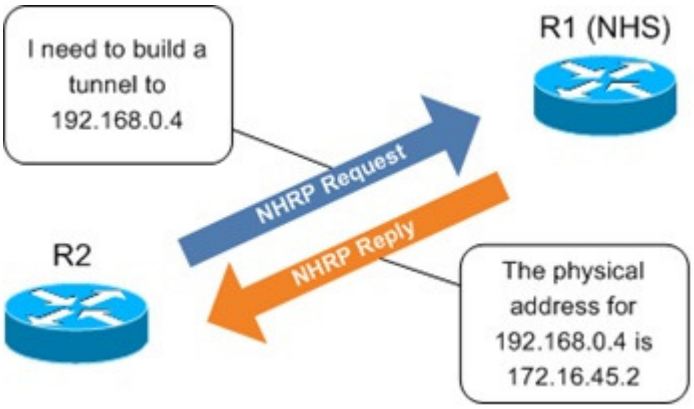
### *DMVPN toiminta*

DMVPN toiminta perustuu useaan jo aikaisemmin käytössä olleeseen protokollaan. Pääasiassa DMVPN toiminnan mahdollistavat IPsec, Next-Hop Resolution Protocol (NHRP), multi Generic Routing Encapsulation (mGRE) sekä käyttötarkoituksen mukaan valittu reititys protokolla (tässä harjoituksessa OSPF). Käytännön tasolla DMVPN toiminta perustuu kahteen perus rooliin, Hub-reititin vastaa siitä että, muut Spoke-reitittimet voivat toimia DMVPN alueen sisällä. Toisin sanoen, Hub-reititin määrittelee ja tarjoaa tiedot, jotta Spoke-laitteet pystyvät liikennöimään.

# *DMVPN-Harjoitus*

Alussa laitteet muodostavat point-to-point yhteyden Hub-reitittimelle ja näin rekisteröivät oman tunneli –ja fyysisenosoitteen sille. Koska DMVPN käyttää tunneloinnissa mGRE, voidaan samaa tunnelia käyttää useammassa yhteydessä ja näin ollen NHRP voi helposti liittää fyysisen -ja loogisenosoitteen toisiinsa.

Kun Spoke-laite haluaa ensimmäisen kerran muodostaa yhteyden toisen Spoke-laitteen kanssa, lähettää se NHRP kyselyn Hub-reitittimille joka sitten palauttaa halutun kohdelaitteen tunneliosoitteen ja näin Spoke pystyy muodostamaan spoke-to-spoke tunnelin. Jatkossa kun laite tahtoo muodostaa yhteyden saman Spoke kanssa, ei sen ole tarvetta kulkea enää Hub-reitittimen kanssa vaan se voi katsoa omasta NHRP-taulustaan halutun kohdelaitteen osoitteen. Hub-reitin toimii siis myös Nex-Hop serverinä (NHS) kuten kuviosta 1. nähdään.



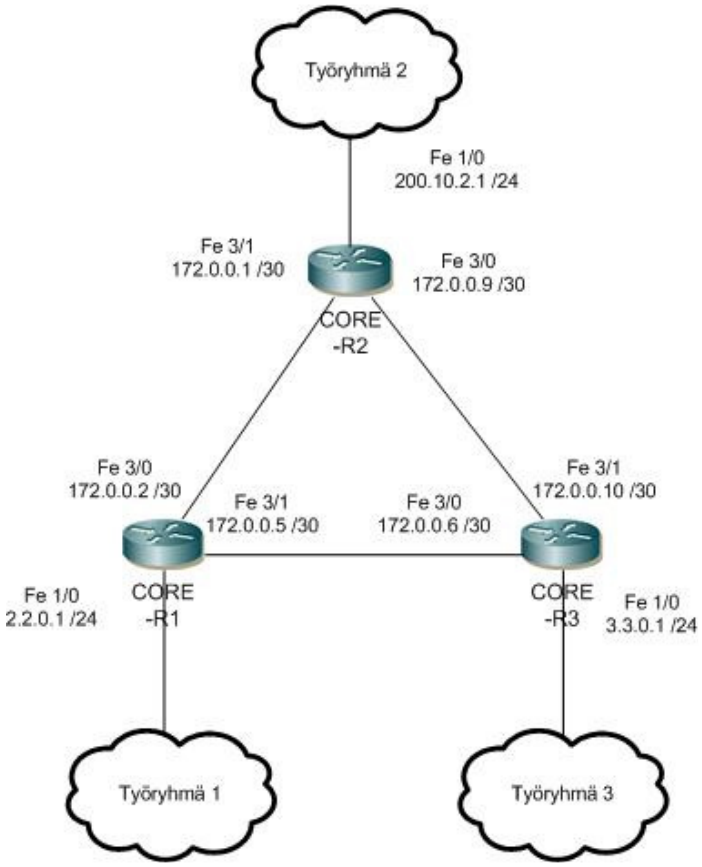

**KUVIO 1. Hub-reittiminen ja Spoke välinen viestin**

# *DMVPN-Harjoitus*

## HARJOITUS

### *Yleistä*

Harjoituksen tarkoituksena on konfiguroida alla olevan topologian (ks. kuvio 2) mukainen ympäristö ja todentaa DMVPN toiminta, käyttäen kahta Spoke reitintä. Internetin pohjakonfiguraatiot ovat valmiina, joten harjoituksessa Cisco Core-reitittimille on tarpeellista määritellä vain työryhmälaitteille menevät rajapinnat.




**KUVIO 2 DMVPN harjoituksen topologia**

# *DMVPN-Harjoitus*

## *Cisco Core-reitittimien konfiguraatiot*

Cisco Core-laitteille on valmiiksi luotu ”Internetiä” simuloiva reititys, joten niiden osalta määritellään vain tarvittavat rajapinnat.

```
ip dhcp excluded-address 2.2.0.1 2.2.0.10
!
ip dhcp pool WG1
network 2.2.0.0 255.255.255.0
default-router 2.2.0.1
!
interface FastEthernet1/0
description Link to WG1-R1
ip address 2.2.0.1 255.255.255.0
duplex auto
 speed auto
!
```

Molemmat Core-R1 ja Core-R3 tulee määritellä käyttämään DHCP:tä osoitteiden jakamiseen ja sen jälkeen määritellä ip-osoite ja aliverkkopeite työryhmä 1 ja 3 meneville rajapinnoille. Yläpuolella nähtävissä Core-R1 konfiguraatiot, Core-R3 toteutetaan vastaavalla kaavalla, käyttäen topologia kuvan mukaisia osoitteita.

Koska WG2-R1 toimii Hub-reitittiminä, tulee sille määritellä staattinen osoite ja näin ollen Core-R2 ei tarvitse asettaa DHCP rooliin.

# *DMVPN-Harjoitus*

## *WG2-R1 Hub-reitittimen konfigurointi*

WG2-R1 asetusten määrittely aloitetaan antamalla Core-R2:lle yhteydessä olevalle rajapinnalle tarvittavat tiedot, sekä lisätään OSPF reitys.

```
interface GigabitEthernet0/0
 description link to CORE-R2
ip address 200.10.2.2 255.255.255.0
 duplex auto
 speed auto

!
Interface Loopback0
ip address 130.0.10.1 255.255.255.252
!
router ospf 1
log-adjacency-changes
network 200.10.2.0 0.0.0.255 area 0
```

Tässä vaiheessa on hyvä myös testata *ping*-komentoa apuna käyttäen, että liikenne toimii Core-laitteille.

Kun perus konfiguraatiot on tehty ja yhteyden toimivuus saatu testattua voidaan siirtyä itse DMVPN asetuksiin. Niiden konfiguroiminen aloitetaan luomalla ISAKMP tason 1 tunnistautumista varten.

```
!
crypto isakmp policy 1
authentication pre-share
crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0
crypto isakmp invalid-spi-recovery
!
```

Seuraava vaihe on luoda IPsec asetukset, joidenka avulla tunneloita liikenne tullaan salaamaan. Salaus muodoksi voidaan valita myös vahvempi salaus, jos sille on perusteltu tarve, eikä laitteiden muisti ole rajoite.

```
!
crypto ipsec transform-set dmvpn123 esp-3des esp-md5-hmac
mode transport
!
crypto ipsec profile dmvpn
set transform-set dmvpn123
```

# *DMVPN-Harjoitus*

Viimeinen vaihe Hub-reitittimen osalta on luoda asetukset itse tunneleille ja ottaa NHRP käyttöön.

```
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.1 255.255.255.0
 no ip redirects
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map multicast dynamic
 ip nhrp network-id 100000
 ip nhrp holdtime 600
 ip ospf network broadcast
 ip ospf priority 2
 delay 1000
 keepalive 5 4
 tunnel source GigabitEthernet0/0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile dmvpn
!
```

Aluksi luodaan tunnelli rajapinta, jolle määritellään sen käyttämä kaista, ip-osoite ja maski normaalisti. Asetetaan NHRP varten tarvittavat asetukset, määritellään tunnellin sisällä tapahtuvaan reititykseen OSPF. (HUOM! ospf priority tulee olla eri kuin Internetin reitityksessä käytetty OSPF ID).

Lopuksi tunnelille määritellään fyysinen rajapinta jonka yhteydessä sitä käytetään, asetetaan se multipoint GRE tilaan sekä liitetään aikaisemmin luotu ipsec profiili käytettäväksi sen kanssa.

Aivan lopuksi määritellään DMVPN sisällä käytettävä OSPF, joka asetettiin jo edellä olleissa konfiguraatioissa.

```
!
router ospf 2
 log-adjacency-changes
 network 10.0.0.0 0.0.0.255 area 0
!
```

On erittäin tärkeä muistaa että OSPF ID on eri kuin muualle tapahtuvassa reitityksessä, muuten tunnelointia ei tapahdu kun DMVPN reitittää liikenteen muun liikenteen sekaan

# *DMVPN-Harjoitus*

## *WG1-R1 ja WG3-R1 Spoke reitittimien konfigurointi*

Spoke-reitittimien, tässä tapauksessa WG1-R1 ja WG3-R1 konfigurointi aloitetaan tuttuun tapaan määrittelemällä tarvittavat rajapinnat. Esimerkeissä käydään läpi vain WG1-R1 konfiguraatiot, mutta WG3-R1 toteutetaan identtisillä asetuksilla, ottaen huomioon eri ip-osoitteet.

Aluksi asetetaan Core-reitittimeen yhteydessä oleva rajapinta saamaan osoitteensa DHCP:ltä.

```
!
interface GigabitEthernet0/0
 description Link to CORE-R1
 ip address dhcp
 duplex auto
 speed auto
!
```

Kun osoite on saatu, ja yhteys muihinkin Core laitteisiin sekä WG2-R1 testattu voidaan jatkaa itse DMVPN asetuksiin. Ensimmäisenä määritellään ISAMKP ja IPsec asetukset jotka tulee vastata Hub-reititimmelle tehtyjä asetuksia.

```
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0
!
crypto ipsec transform-set dmvpn123 esp-3des esp-md5-hmac
 mode transport
!
crypto ipsec profile dmvpn
 set transform-set dmvpn123
!
```

# *DMVPN-Harjoitus*

Seuraavaksi määritellään tunneliasetukset, jotka poikkeavat hieman Hub-reitittimelle tehdyistä. Mutta suurimmalta osalta ne ovat varsin samankaltaiset.

```
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.2 255.255.255.0
 no ip redirects
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map 10.0.0.1 200.10.2.2
 ip nhrp map multicast 200.10.2.2
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 ip ospf network broadcast
 ip ospf priority 0
 delay 1000
 keepalive 5 4
 tunnel source GigabitEthernet0/0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile dmvpn
!
```

Kun yllä oleva konfiguraatio tehdään WG3-R1:lle, on ainut muutos kohdan; *ip address 10.0.0.2 255.255.255.0* ip-osoite joka WG3-R1:llä tulee olla *10.0.0.3 255.255.255.0* . Kun tunneli on saatu määriteltyä, asetetaan myös Spoke-laitteille OSPF reititys.

```
!
router ospf 2
 log-adjacency-changes
 network 10.0.0.0 0.0.0.255 area 0
!
```

Kun edellä olleet asetukset on tehty molemmille Spoke-laitteille, voidaan siirtyä DMVPN toiminnan todentamiseen

# *DMVPN-Harjoitus*

## *DMVPN testaus*

DMVPN testaus tehdään käyttäen Cisco IOS tarjoamilla *show* komennoilla sekä *ping* ja *traceroute* komennoilla. Ennen *ping* komennon ajamista laiteilla, on hyvä tarkastella show komentojen antamia tuloksi kun DMVPN on ns. alkutilassa.

Testattavia komentoja:

*show ip nhrp*
*show crypto isamkp sa*
*show dmvpn*
*show crypto ipsec sa*
*show crypto engine connection active*

Alkutilan tarkastelun jälkeen voidaan ajaa traceroute sekä ping komentoa laitteiden välillä. Jokaisen ajetun komennon jälkeen on hyvä tarkastella, kuinka laitteiden ”tietoisuus” muista laitteista muuttuu.

Jyväskylän Ammattikorkeakoulu
Teknologia / ICT

# *Liite 18 .Juniper Group VPN Co-Location harjoitus*



## JOHDANTO

### *Yleistä*

Tämän harjoituksen tarkoitus on toteuttaa Group VPN ratkaisu käyttäen Juniper Networksin reitittimiä. Harjoituksen erikoisuutena on vain Juniper-laitteilla toimiva Co-Location konfiguraatio. Co-Location mahdollistaa saman laitteen käyttämistä niin Key Server kuin Group Member roolissa.

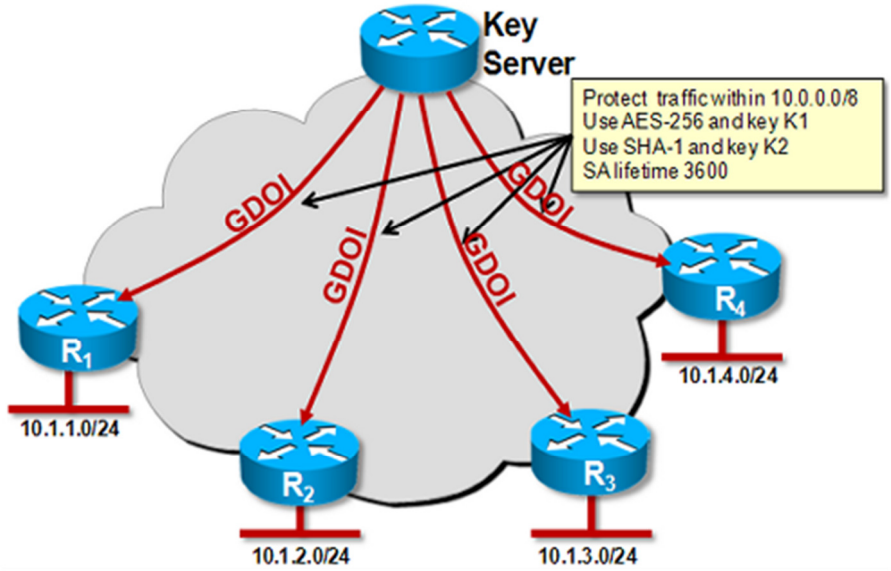
### *Group Encrypted Transport VPN*

Juniper Group VPN on *Group Domain of Interpretation* (GDOI):iin perustuva ratkaisu. Sen perusidea on muodostaa key serverin(KS) hallinnoima ryhmä, johon uudet group memberit (GM) voivat liittyä. Näiden GM välinen liikenne salataan KS asetettujen rajoitusten mukaisesti. Perinteiseen VPN verrattuna Group VPN mahdollistaa paremman skaalautuvuuden, kun jokaista yhteyttä GM:ien kesken ei ole tarvetta määritellä.

Toiminnaltaan GDOI ja Group VPN nojaa jo olemassa olevaan infrastruktuuriin. Näin ollen se vaatii jonkin toimivan verkkoratkaisun, jonka päällä itse ryhmä pystyy toimimaan. Tämä toisaalta mahdollistaa erilaisten Quality of Service (QoS) – palveluiden tarjoamisen salauksen lisäksi.

# *Juniper Group VPN Co-Location harjoitus*

GDOI ympäristössä laitteet rekisteröityvät ensin KS:llä, jolta ne saavat muiden jäsenten kanssa käytettävät turvakäytänteet, sekä salattavien verkkojen ja protokollien tiedot.



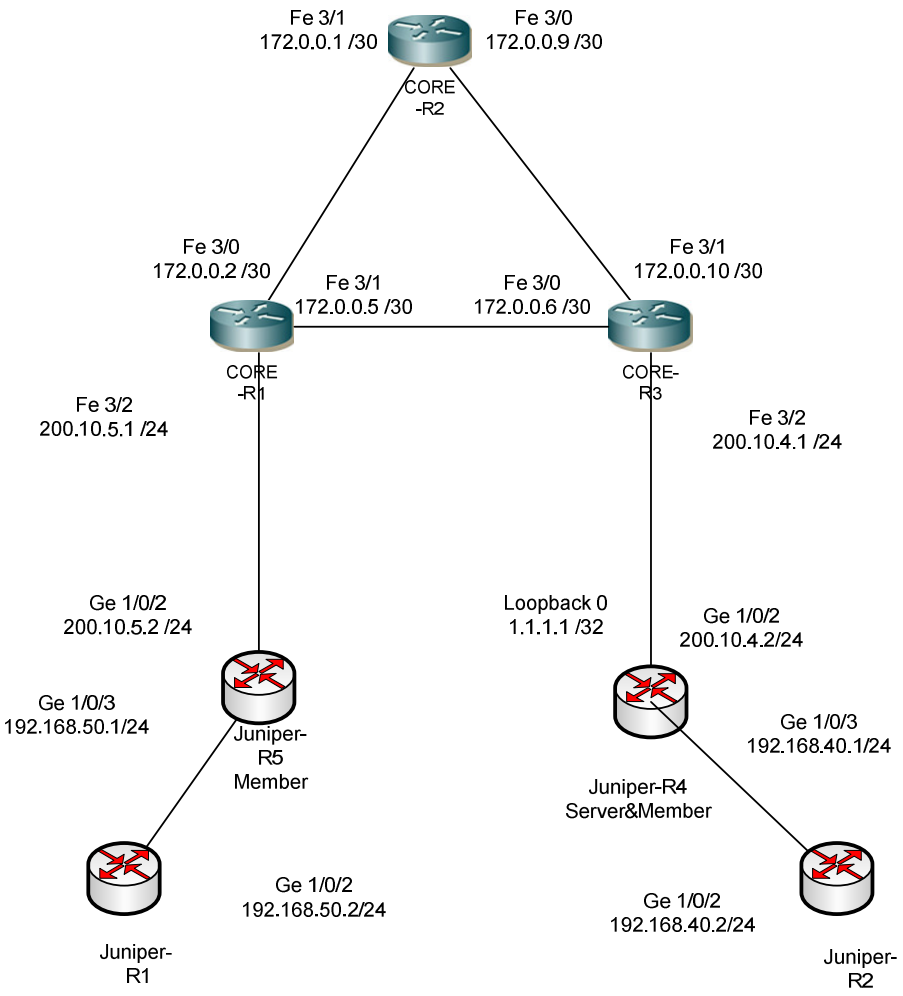

**KUVIO 1. KS toiminta**

# *Juniper Group VPN Co-Location harjoitus*

## HARJOITUS

### *Yleistä*

Harjoituksen tarkoituksena on konfiguroida Co-Location käyttäen kahden laitteen ryhmä, sekä näille ”sisäverkon” laitteet. Tämän jälkeen tarkastellaan Junos-käyttöjärjestelmän mahdollistamia komentoja, joidenka avulla harjoituksen toimivuus voidaan todentaa. Harjoituksen tarvitsema Internetin konfiguraatio on valmiiksi tehty käyttäen SpiderNetin Cisco Core-reitittimiä. Harjoituksen topologia tulee vastaamaan kuvion 2. esimerkkiä.




**KUVIO 2. Harjoituksen topologia**

# *Juniper Group VPN Co-Location harjoitus*

## *Juniper-R4 KS konfiguraatio*

Juniper-R4 toimii harjoituksessa molemmissa rooleissa (GM ja KS), joten harjoitus aloitetaan määrittelemällä sille KS asetukset. Aluksi kuitenkin asetetaan harjoituksessa käytettävät rajapinnat ja reititys Cisco Core-R1:lle.

```
}
   ge-1/0/2 {
      unit 0 {
         family inet {
            address 200.10.4.2/24;
         }
      }
   }
   ge-1/0/3 {
      unit 0 {
         family inet {
            address 192.168.40.1/24;
         }
      }
   }
   lo0 {
      unit 0 {
         family inet {
            address 1.1.1.1/32;

}
routing-options {
   static {
      route 0.0.0.0/0 next-hop 200.10.4.1;
   }
```

Cisco-laitteista poiketen, Junosissa voidaan antaa aliverkko peitteet myös bitti muodossa. Komennot tulee kuitenkin muistaa hyväksyä *commit* –komentoa käyttäen.

Tässä kohtaa on hyvä testata yhteyden toimivuus Cisco Core-laitteille. Kun yhteys ”Internetiin” on varmistettu, voidaan jatkaa konfiguraatiota.

```
}
security {
   group-vpn {
      server {
         ike {
            proposal srv-prop {
               authentication-method pre-shared-keys;
               dh-group group2;
               authentication-algorithm sha1;
               encryption-algorithm 3des-cbc;
            }
```

Aluksi luodaan IKE käytänteet, joidenka avulla GM-laitteet tulevat tunnistautumaan. Valitaan tunnistautumistavaksi ennalta jaetut avaimet ja määritellään haluttu salauksen muoto ja vahvuus.

```
 }
 policy srv-pol {
    mode aggressive;
    proposals srv-prop;
    pre-shared-key ascii-text grouvpn

 }
 gateway gw1 {
    ike-policy srv-pol;
    address 200.10.4.2;
 }
 gateway gw2 {
    ike-policy srv-pol;
    address 200.10.5.2;
 }
 gateway srv-gw {
    ike-policy srv-pol;
    address 192.168.40.1;
 }
```

Seuraavaksi, edelleen IKE alla luodaan käytänne jolle määritellään salasanaksi *”groupvpn”* ja neuvottelu muodoksi *aggressive*. Näiden jälkeen käytänne sidotaan haluttuihin yhdysosoitteisiin.

Seuraavaksi siirrytään määrittelemään IPsec asetukset server roolin alla.

```
security {
   group-vpn {
      server {
           ipsec {
            proposal group-prop {
               authentication-algorithm hmac-sha1-96;
               encryption-algorithm 3des-cbc;
               lifetime-seconds 3600;
            }
          }
```

Seuraavaksi määritellään varsinaiset server asetukset ja ne tiedot mitä KS lähettää rekisteröinnin yhteydessä GM:lle.

```
        group grp1 {
          group-id 1;
          ike-gateway srv-gw;
          ike-gateway gw1;
          ike-gateway gw2;
          anti-replay-time-window 120;
          server-address 200.10.4.2;
          server-member-communication {
             communication-type unicast;
             encryption-algorithm aes-128-cbc;
             sig-hash-algorithm md5;
             certificate srv-cert;
```

Aluksi asetetaan ryhmä ja sille tarvittava tunnus, eli ID. Sitten sidotaan aikaisemmin määritellyt gatewayt osaksi ryhmään. Tämän jälkeen aktivoidaan server rajapinnassa 200.10.4.2. Ja määritellään ryhmän ja KS välisen liikenteen asetukset.

Seuraavaksi asetetaan salattavan liikenteen asetukset, määritellään salattavaksi kaikki liikenne verkkojen 192.168.50.0 ja 192.168.40.0 välillä. Aivan lopuksi vielä aktivoidaan co-location yksinkertaisella komennolla.

```
}
        ipsec-sa group-sa {
           proposal group-prop;
           }
           match-policy pol1 {
              source 192.168.40.0/24;
              destination 192.168.50.0/24;
              source-port 0;
              destination-port 0;
              protocol 0;
           }
           match-policy pol2 {
              source 192.168.50.0/24;
              destination 192.168.40.0/24;
              source-port 0;
              destination-port 0;
              protocol 0;
           }
        }
     }
  }
  co-location;
```

Nyt KS rooli on määritelty laitteelle ja voidaan jatkaa seuraavaan vaiheeseen, eli GM-rooli määrittelyyn.

# *Juniper Group VPN Co-Location harjoitus*

## *Juniper-R4 ja Juniper-R5 GM konfiguraatio*

Molemmat Juniper-R4 ja R5 tullaan konfiguroimaan samalla tavalla GM -roolin osalta. Tässä vaiheessa onkin hyvä määritellä perusasetukset (rajapinnat ja reititys) Juniper-R5:lle. Sen jälkeen molempien laitteiden konfiguraatiot ovat identtiset.

```
}
security {
   group-vpn {
      member {
         ike {
            proposal prop1 {
               authentication-method pre-shared-keys;
               dh-group group2;
               authentication-algorithm sha1;
               encryption-algorithm 3des-cbc;
            }
            policy pol1 {
               mode aggressive;
               proposals prop1;
               pre-shared-key ascii-text groupvpn
            }
            gateway g1 {
               ike-policy pol1;
               address 200.10.4.2;
               local-address 192.168.40.1;
            }
         }
         ipsec {
            vpn v1 {
               ike-gateway g1;
               group-vpn-external-interface ge-1/0/2.0;
               group 1;
```

Aluksi luodaan IKE käytänteet, jotka vastaavat KS-roolille asetettuja. Seuraavaksi valitaan käytettävä gateway, asetetaan sille laitteen oma rajapinta sekä KS:än käyttämä rajapinta. Lopuksi määritellään ipsecin käyttämä vpn versio sekä rajapinta jossa group-vpn:n tulee toimimaan.

Koska GM-laitteet saavat halutut policy asetukset suoraan KS:ltä ei niitä ole tarpeen muokata muulta osin kuin sallia ipsec vpn.

```
policies {
    from-zone trust to-zone untrust {
        policy groupvpn {
            match {
                source-address any;
                destination-address any;
                application any;
            }
            then {
                permit {
                    tunnel {
                        ipsec-group-vpn v1;
                    }
                }
            }
        }
        policy deny-all {
            match {
                source-address any;
                destination-address any;
                application any;
            }
            then {
                deny;
            }
        }
```

untrust to trust zonen osalta konfiguraatio tehdään täysin identtisesti.

Kun molemmat R4 ja R5 on konfiguroitu GM rooliin, voidaan harjoitusta jatkaa ja siirtyä määrittelemään R1 ja R2. Näiden laitteiden tarkoitus on simuloida sisäverkkoa.

# *Juniper Group VPN Co-Location harjoitus*

## *Juniper R1 ja R2 konfiguraatio*

R1 ja R2 osalta konfiguraatio on yksinkertainen, molemmat laitteet konfiguroidaan vain rajapintojen sekä staattisen reitityksen osalta.

```
interfaces {
   ge-0/0/0 {
      unit 0;
   }
   ge-1/0/1 {
      unit 0 {
         family inet {
            address 192.168.50.2/24;
         }
      }
   }
}
routing-options {
   static {
      route 0.0.0.0/0 next-hop 192.168.50.1;
   }
```

## *Group VPN testaus*

Kun konfiguraatiot on tehty ja yhteyksien toimivuus on varmistettu ping –komennon avulla voidaan siirtyä itse GDOI toiminnan tarkasteluun. Aja ping komentoa laitteiden R1 ja R2 välillä, voit myös koittaa miten traceroute komento toimii laitteiden välillä verrattuna kun sen tekee esim. Cisco Core-R1:lle

Testattavia komentoja (huomaa kaikki kyseisten alla olevat lisäkomennot):

*run show security group-vpn server*

*run show security group-vpn member*