Jimi Norman

Yrityksen palvelinsisäverkon VPN-yhteydet

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tietotekniikan Koulutusohjelma

Insinöörityö

27.10.2017

Metropolia

Tekijä(t) Otsikko Sivumäärä Aika	Jimi Norman Yrityksen palvelinsisäverkon VPN-yhteydet 34 sivua + 1 liitettä 27.10.2017
Tutkinto	Insinööri (AMK)
Koulutusohjelma	Tietotekniikka
Suuntautumisvaihtoehto	Tietoverkot ja tietoliikenne
Ohjaaja(t)	Jukka Louhelainen, Vanhempi Lehtori

Insinöörityön tarkoituksena oli tutkia IPSec VPN -tunnelointiteknologiaa ja siihen liittyviä kryptaus- ja todennusalgoritmeja sekä suunnitella ja toteuttaa vikasietoinen VPN-tunneli keskisuuren yrityksen kahden hajautetun palvelinverkon välille.

Taustana yrityksellä on aikaisemmin ollut PPTP-sillalla yhdistetty pää- ja sivukonttorin palvelinverkot. Tämä silta on ollut epäkunnossa. Lisäksi PPTP -protokolla on tunnetusti haavoittuvainen, jonka takia uusi IPSec VPN -tunneli haluttiin pystyttää palvelinverkkojen välille.

Erityisenä painopisteenä työssä oli laitteiden ja tunneleiden vikasietoisuus. Työssä käytettiin MikroTik- ja pfSense-reitittimiä tunneleiden päätepisteinä. Samalla perehdyttiin MikroTik:in skriptausominaisuuksiin.

Työn lopputuloksena luotiin turvallinen ja vikasietoinen IPSec VPN -tunneli yrityksen pääkonttorin ja sivukonttorin palvelinverkkojen välille. Lisäksi ulkomaille pystytettiin kaksi palvelintilaa, joissa sovellettiin korkeaa saatavuutta ja työn tuloksia IPSec VPN -tunneleiden pystyttämisessä.

Avainsanat	IPSec, VPN, Tietoturva, Korkea saatavuus

Author(s) Title Number of Pages Date	Jimi Norman Company's site-to-site VPN connections 34 pages + 1 appendices 27.10.2017
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Communication Networks
Instructor(s)	Jukka Louhelainen, Senior Lecturer

The goal of this thesis was to study IPSec VPN tunneling technology and the related encryption and authentication algorithms, then design and implement a fault-tolerant site-to-site VPN tunnel between two decentralized server networks for a medium sized company.

Previously the company had connected the main and branch office server networks with a PPTP bridge. The bridge had been inoperative, and the PPTP protocol is known to be insecure, so I was tasked with setting up a new IPSec VPN tunnel between the server networks.

Emphasis was placed on the fault-tolerance of the hardware configuration and the tunnels. I was provided with MikroTik and pfSense routers, which were to be used as tunnel end points. This also allowed me to explore MiroTik's scripting features to increase fault-tolerance.

As a result, a secure and fault-tolerant IPSec VPN tunnel was created between the company's main and branch server networks. In addition, two server rooms were set up abroad, where we utilized high availability and the results of my work on IPSec VPN tunnels.

Keywords	IPSec, VPN, Network security, High availability

Sisällys

Lyhenteet

VPN	Virtual Private Network. Tekniikka, jolla kaksi tai useampi verkko voidaan yhdistää julkisen verkon yli muodostaen yksityisen verkon.
IP	Internet Protocol. Huolehtii IP-tietoliikennepakettien toimittamisesta pakettikytkentäisessä verkossa.
DNS	Domain Name System. Internetin nimipalvelujärjestelmä, joka muuntaa verkkotunnuksia IP-osoitteiksi.
DDNS	Dynamic DNS. Sallii DHCP:lla saadun ja vaihtuvan IP-osoitteen liitettävän kiinteään DNS nimeen.
DHCP	Dynamic Host Configuration Protocol. Verkkoprotokolla, jonka tehtävä on jakaa IP-osoitteita kytkettyille laitteille.
ARP	Address Resolution Protocol. Protokolla, jolla selvitetään IP-osoitetta vastaava fyysinen osoite.
IPSec	Internet Protocol Security. Joukko tietoliikenneprotokollia Internet-yhteyksien turvaamiseen.
ESP	Encapsulating Security Payload. IPSec-protokollasarjaan kuuluva protokolla, jota käytetään liikenteen salaamiseen.
AH	Authentication Header. IPSec-protokollasarjaan kuuluva protokolla, jota käytetään todennukseen ja viestien ehyden tarkistamiseen.
SA	Security Association. Kahden verkkokokonaisuuden jaetut turvallisuusparametrit, jotka tukevat turvallista kommunikaatiota.
IKE	Internet Key Exchange. Protokollasarja SA:n muodostamiseen, luo turvallisuusparametrit.
ISAKMP	Internet Security Association and Key Management Protocol. Protokolla SA:n muodostamiseen, jakaa turvallisuusparametrit.

DH	Diffie-Hellman. Salausprotokolla yhteisen salaisuuden sopimiseen turvattoman tietoverkon yli.
PFS	Perfect Forward Secrecy. Salausjärjestelmien ominaisuus, jonka ansiosta salausavaimen murtaminen ei johda aiemmin salattujen viestien vaarantumiseen.
NAT	Network Address Translation. Tekniikka jolla julkisesti liikennöityjä IP-osoitteita piilotetaan tai säästetään.
PPTP	Point-to-Point Tunneling Protocol. Vanhentunut VPN-tunnelointiprotokolla.
IETF	Internet Engineering Task Force. Internet-protokollien standardoinnista vastaava organisaatio.
LACP	Link Aggregation Control Protocol. Standardiprotokolla, jolla voi yhdistää useita fyysisiä yhteyksiä yhdeksi loogiseksi yhteydeksi.
RFC	Request for comments. IETF:n julkaisemia standardeja.
ICV	Integrity Check Value. AH-protokollassa oleva arvo, jota käytetään todennuksessa.
SSH	Secure Shell. Salattuun tietoliikenteeseen tarkoitettu protokolla, sallii etäyhteyden ottamisen SSH sallittuun laitteeseen.
IEEE	Institute of Electrical and Electronics Engineers. Kansainvälinen tekniikka-alan standardijärjestö.
MPLS	Multiprotocol Label Switching. Menetelmä, jolla kuljettaa IP-paketteja ennalta määriteltyjen yhteyksien yli ilman, että tarvitsee tehdä reititystä.
VRRP	Virtual Router Redundancy Protocol. Standardi verkkoprotokolla, jolla voidaan luoda virtuaalireititin useasta fyysisestä reitittimestä.
CARP	Common Address Redundancy Protocol. Vaihtoehtoinen protokolla VRRP:lle, käytetään redundanssin luomiseen.

UPS	Uninterruptible Power Supply. Laite, jonka tehtävä on taata tasainen virransyöttö lyhyissä sähkökatkoksissa.
AES	Advanced Encryption Standard. Lohkosalausalgoritmi, joka on laajassa käytössä.
3DES	Triple Data Encryption Algorithm. Vanhentunut salausalgoritmi, jossa ajetaan DES algoritmi kolme kertaa.
SPI	Security Parameter Index. IPSec-tunnelointitekniikan käyttämä identifikaatio merkki. Auttaa eri tunneleiden erottamisessa.
HMAC	Hash-based Message Authentication Code. Yhdistelmä kryptograafisesta avaimesta ja hash funktiosta.
SHA	Secure Hash Algorithm. Kokoelma hash-funktioita. Hash-funktio luo sisään syötetystä tekstistä sattumanvaraiselta näyttävän merkkijonon.

1 Johdanto

Insinöörityöni tarkoituksena oli tehdä tutkimusta palvelinsalien yhdistämisestä IPsec VPN -tunnelointiteknologian avulla Finnish Net Solutions Oy:lle. Erityisesti työssä keskitytään IPSec-protokollaan, VPN-tunnelin tietoturvaan, palvelinsisäverkkoihin, tunneleiden vikasietoisuuteen ja korkeaan saatavuuteen.

Finnish Net Solutions Oy tai FNS Oy, on vuonna 2001 perustettu ohjelmistoyritys Otaniemessä, jonka liikevaihto vuonna 2016 oli 2,3 miljoonaa euroa ja joka työllistää 43 henkilöä. Yritys myy ja kehittää räätälöityjä ohjelmistotuotteita. Asiakkaat ovat eläinterveydenhuollon ja hoitoalan asiantuntijayrityksiä Suomessa ja ulkomailla.

FNS Oy:n sivuhaarakonttorilla on pieni palvelintila, johon toteutetaan varmuuskopioita ja testipalvelimia, jotka oli yhdistetty PPTP-sillalla pääkonttorin palvelinverkkoihin. Yhteydet konttorin palvelintilasta toiseen olivat epävakaat, ja silta oli usein alhaalla. Sillan tietoturva herätti myös epäilyksiä PPTP-protokollan yleisesti tunnettujen haavoittuvuuksien vuoksi.

Yrityksen toimeksiannon tavoite oli luoda IPSec VPN -tunneli näiden kahden palvelinsalin välille, varmistaa tunnelin vikasietoisuus tapauksissa joissa yhteys tai virta katkeaa, sekä siirtää yrityksen palvelinverkkoja eteenpäin muuttamalla pois sillatuista ratkaisuista reititettyihin ratkaisuihin. Myöhemmin saimme lisätoimeksiannon, jonka tavoitteena oli pystyttää kaksi palvelinsalia ulkomaille ja liittää ne IPSec VPN -tunneleilla yrityksen palvelinverkkoihin. Erityisen tärkeänä tavoitteena oli korkea saatavuus ja vikasietoisuus.

2 IPSec-protokollasarja

Työssä käytetty IPSec-protokollasarja on keskeinen VPN-tunnelin toimintaan ja sen optimoimiseen. Ensimmäisenä käyn läpi protokollasarjaa, siihen liittyviä muita protokollia ja muutamia yleisiä autentikointi- ja salausalgoritmeja.

2.1 Arkkitehtuuri

IPSec on IETF:n kehittämä standardi protokollasarja, joka tarjoaa datan eheyden varmennusta, todennusta ja salausta, kun tietoa siirrettään kommunikaatiopisteiden välillä IP-verkkojen yli. Koska IPSec on määritetty standardi, se on erittäin yhteensopiva eri valmistajien laitteiden kanssa.

IPSec suojaa dataa pakettitasolla. Paketti on datanippu, joka on organisoitu datagrammi -muotoon verkon yli lähettämiseksi ja joka sisältää headerin ja payloadin.

IP-tietoverkkoraudan näkökulmasta kryptattuja paketteja voidaan reitittää kuten mitä tahansa tavallista IP-pakettia. Ainoat laitteet. jotka tarvitsevat IPSec-implementaation ovat IPSec-päätepisteet. [1.]

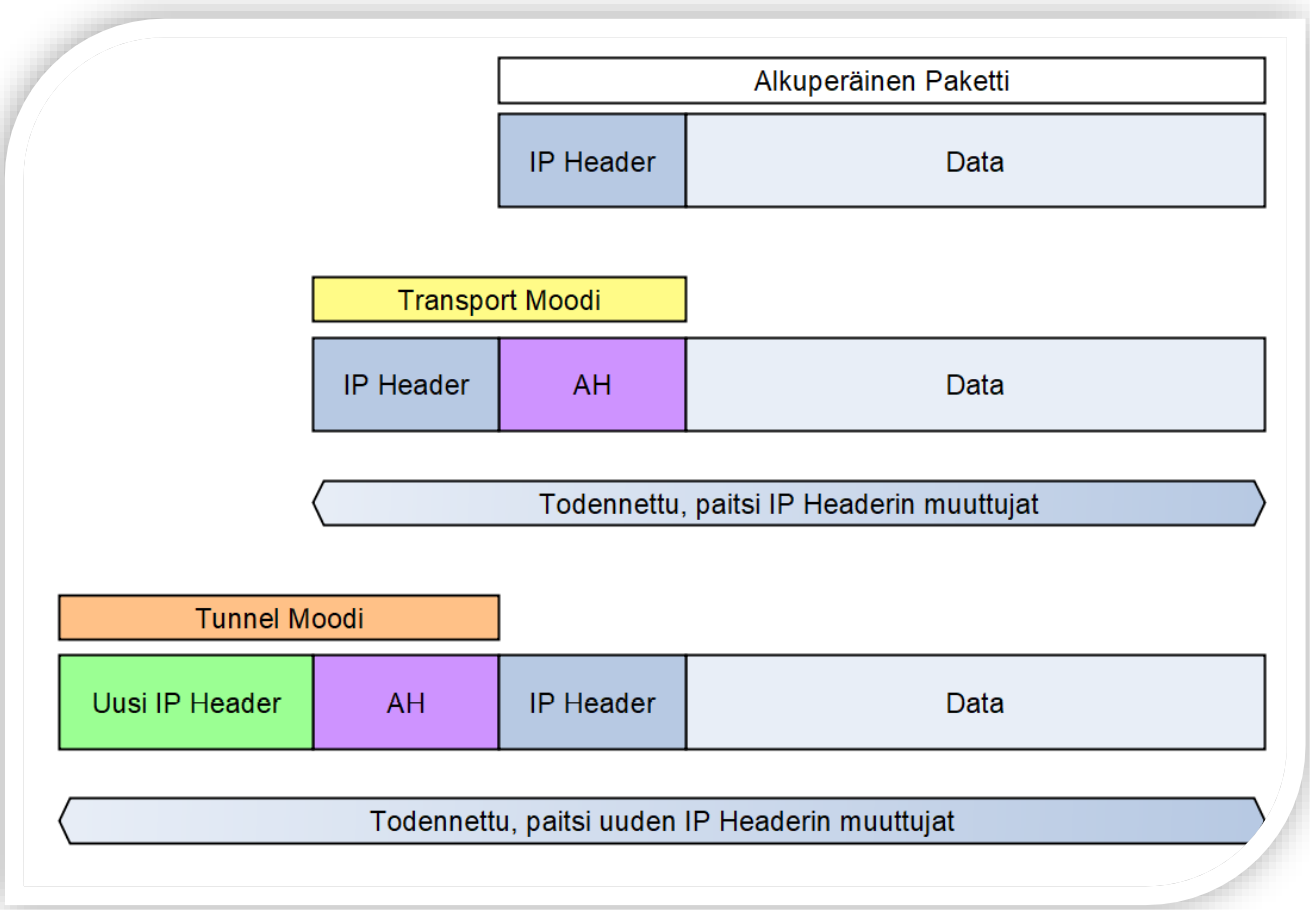
IPSec-arkkitehtuuriin kuuluu kolme pääkomponenttia:

- Authentication Header (AH) -protokolla
- Encapsulating Security Payload (ESP) -protokolla
- Internet Key Exchange (IKE) -protokolla.

2.1.1 Authentication Header (AH)

Authentication Header on toinen IPSec:n päätietoturvaprotokollista. AH-protokolla todentaa joko osan tai kaiken datagrammin sisällöstä lisäämällä pakettiin headerin, joka on laskettu datagrammin arvojen perusteella. Se mistä datagrammin arvoista header lasketaan, riippuu siitä, käytetäänkö IPSec tunnel vai transport -moodia ja onko käytössä IPv4 vai IPv6. Lähettävä laite tekee AH-laskutoimituksen ja syöttää tuloksen omaan headeriinsä. Tätä tulosta kutsutaan ICV:ksi. Vastaanottava laite suorittaa saman AH-laskutoimituksen käyttäen näiden kahden laitteen yhteistä avainta. Tämä antaa sen nähdä, onko yhteenkään datagrammin kenttään tehty muutoksia. AH headerin tehtävä ei siis ole kryptata viestejä, vaan todentaa viestin alkuperä ja eheys. [2.]

AH:ta ei pystytä käyttämään NAT:n ylittävissä verkoissa, sillä AH käyttää ICV-laskelmissaan lähettäjä ja vastaanottaja IP-osoitteita. NAT rikkoo end-to-end -mallin perusideaa, joka on kriittinen AH:n toimintaan. Tämän takia tätä varten luotu NAT-Traversal ei myöskään tue AH:ta.



Kuva 1. Miten Authentication Header vaikuttaa IP pakettiin.

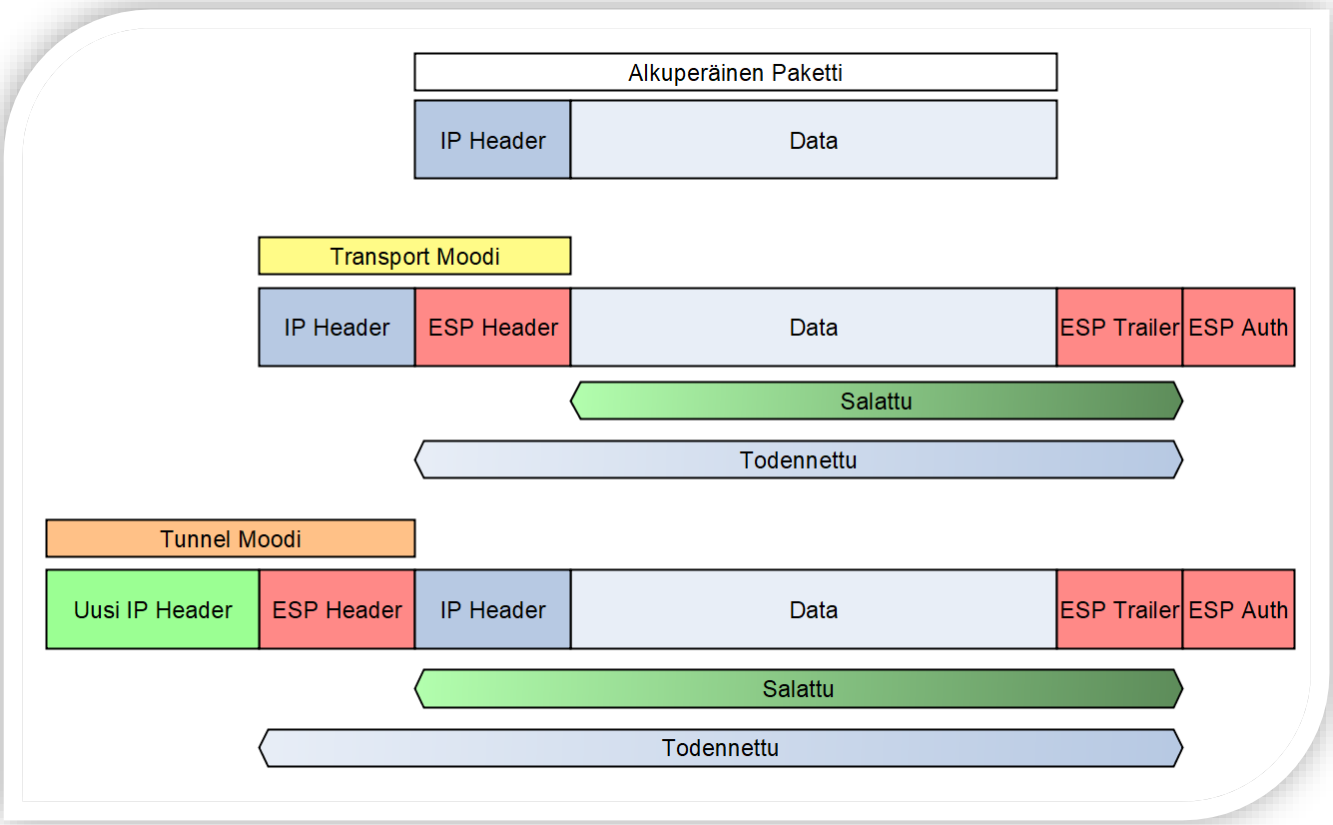
2.1.2 Encapsulating Security Payload (ESP)

Pelkkä eheyden varmistaminen ei ole vielä tarpeeksi, vaan halutaan, ettei lähetettävää dataa pystytä lukemaan ennen kuin se on sille tarkoitetuissa käsissä. Tähän käytetään ESP:tä. Sen päätarkoitus on salata paketin data käyttäen valittua kryptausalgoritmia. ESP sisältää myös oman autentikointikaavan, joka on samankaltainen kuin AH:ssa ja jota pystytään käyttämään AH:n kanssa tai yksinään.

ESP:n toiminta on erilainen riippuen mitä IPSec-moodia käytetään. Transport-moodissa ESP salaa pelkän siirrettävän tiedon osuuden datagrammista, eikä tarjoa autentikointia ja eheyttä koko IP-paketille, kuten AH tekee. ESP-tunnelimoodissa sen sijaan salaa ja

autentikoi koko alkuperäisen IP-paketin. ESP lisää uuden IP headerin paketin eteen, joka jää selkokieliseksi.

ESP:n autentikointi ja eheyden tarkistus toimii NAT:n ylittävissä verkoissa, sillä sen ICV-laskelmat eivät perustu ulommaisiin IP-osoitteisiin. ESP:tä pidetään laajalti parempana protokollana AH:hon verrattuna, sillä se pystyy hoitamaan autentikoinnin ja suojaa datan. NAT:in käyttö on laajaa nykymaailmassa, mikä korostaa ESP:n ylivertaisuutta. [3.]



Kuva 2. Miten ESP vaikuttaa IP pakettiin.

2.2 Security Associations (SA)

Security Association on looginen yhteys kahden dataa lähettävän laitteen välillä. SA suojaa yksisuuntaisen liikenteen dataa käyttäen aiemmin mainittuja ESP- ja AH-protokollia. IPSec-tunneli luodaan käyttäen kahta yksisuuntaista SA:ta, jotka yhdessä suojaavat liikennettä päätepisteiden välillä molempiin suuntiin. [4.]

Jokainen SA sisältää arvoja kuten vastaanottaja- ja lähettäjälaitteiden IP-osoitteet, käytettävät autentikointi- ja kryptausalgoritmit, turvallisuusavaimet, SA:n elinaika, ja Security

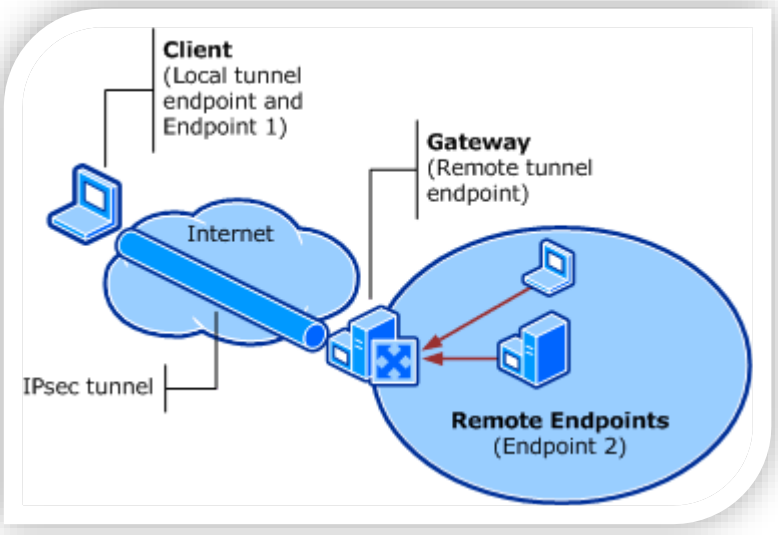
Parameter Index (SPI) -arvo. SPI on identifikaatiomerkki datagrammin headerissä, jonka tarkoituksena on erottaa, mitä SA:ta saapuva liikenne käyttää. [5.]

IPSec SA:n toiminta perustuu sen moodeihin. Moodi kertoo, miten IPSec-protokollaa sovelletaan datapakettiin. Kaksi IPSec-moodia ovat tunnel mode ja transport mode.

2.2.1 Transport mode

Transport moodia käytetään end-to-end -yhteyksien suojaamiseen; kun tunnelin toinen päätepiste on staattinen, mutta toinen ei, esimerkiksi tilanteissa, joissa otetaan VPN etäyhteys kotikoneelta työpaikan verkkoon. Transport moodissa hostin, joka luo paketin pitää pystyä suojaamaan sen ja varmentamaan tulevia paketteja. Hostilla tarkoitetaan laitetta, joka lähettää ja vastaanottaa verkkoliikennettä, kuten esimerkiksi tietokone tai mobiililaite. Käytännössä se tarkoittaa, että laitteelle pitää löytyä toimiva VPN-sovellus.

IPSec transport -moodissa koteloi ainoastaan paketin payloadin, eli kuljetettavan datan. IP headeriin ei kosketa, joten kun IPSec on prosessoinut paketin vanhat lähettäjä/vastaanottaja-IP-osoitteet näkyvät yhä paketissa. Tämän takia mahdollinen hyökkääjä voi tietää tarkasti, mistä paketti lähetetään ja mihin. [6.]

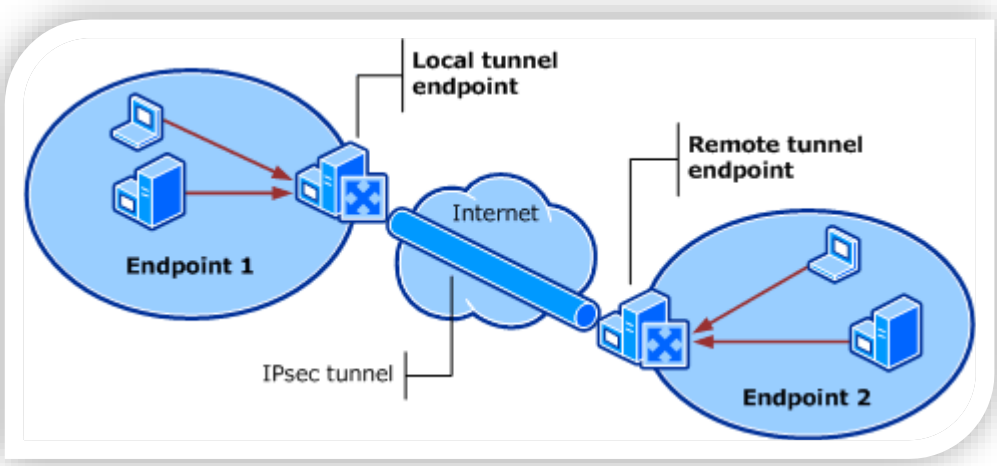


Kuva 3. IPSec transport moodissa. [31.]

2.2.2 Tunnel mode

Tunnel-moodia käytetään yleisesti kaiken muun ja site-to-site-yhteyksien suojaamiseen. Tunnelit muodostetaan usein gateway-reitittimien välille. Gateway on verkkolaite, joka monitoroi ja hallinnoi lähtevää ja tulevaa verkkoliikennettä, sekä reitittää liikenteen sen määränpäähän. Se sijaitsee rajapinnalla julkisen ja yksityisen verkon välillä. Tunnel-moodia käytetään hajautettujen sisäverkkojen yhdistämiseen yrityksissä, kuten pääkonttorin ja sivukonttorin yhdistäminen.

IPSec tunnel -moodissa koteloi koko IP-paketin. Tästä koteloidusta paketista tulee uuden paketin payload. Uusi IP header luodaan, mikä sisältää näiden kahden gatewayn osoitteet. Gatewayt suorittavat koteloinnin ja sen purkamisen päätelaitteiden puolesta. [6.]



Kuva 4. IPSec tunnel moodissa. [31.]

2.2.3 Key Management

Kuten monet verkkoturvallisuuden protokollat, IPSec perustuu ns. ”jaettuun salaisuuteen”. Kaksi laitetta, jotka haluavat lähettää ja vastaanottaa viestejä salaisesti, tekevät niin käyttäen yksityistä tiedonmurua. Ulkopuoliset voivat siepata näiden kahden välillä liikkuvaa dataa, mutta he eivät pysty lukemaan sitä (ESP) tai tehdä muutoksia siihen (AH). Ensin pitää kuitenkin päättää tämä salainen avain. [7.]

IPSec käyttää Internet Key Exchange (IKE) -protokollaa tähän tarkoitukseen. Se helpottaa ja automatisoi SA-prosessia, sekä vastaa avainten jakamisesta lähettäjä-vastaanottaja-parin välillä. Nämä avaimet takaavat, että vain lähettäjä ja vastaanottaja pystyvät avaamaan heidän välillä kulkevia viestejä. [6.]

IKE perustaa jaetut turvallisuuskäytännöt ja todennetut avaimet. ISAKMP vastaa näiden avainten jakamisesta. Näitä kahta lyhennettä käytetään usein tarkoittamaan samaa asiaa, vaikka ISAKMP on vain osa IKE-protokollaa. [8.]

2.2.4 IKEv2

Alkuperäisessä IKE:ssä oli lukemattomia konfiguraatiovaihtoehtoja, mutta ei mitään automaattista tai yksiselitteistä tapaa käyttää sitä esimerkkitapauksissa. Tästä johtuen IKE:n molempien puolien laitteet piti konfiguroida täsmälleen samanlaisiksi parametri kerrallaan, tai yhteyttä ei pystyttäisi luomaan. Tämä hankaloitti tunneleiden konfigurointia sekä ongelmien diagnosointia ja korjaamista. Ensimmäisessä versiossa oli myös suuria ongelmia tunneleiden uudelleenneuvottelussa (Dead Peer Detection), josta aiheutui paljon ylläpitotöitä. DPD-ongelmassa alhaalla olevat tunnelit kuvittelevat olevansa vielä ylhäällä. Tämän takia tunnelit eivät aina automaattisesti yhdistäneet toisiinsa uudelleen, vaan ne piti manuaalisesti resetoida. [9.]

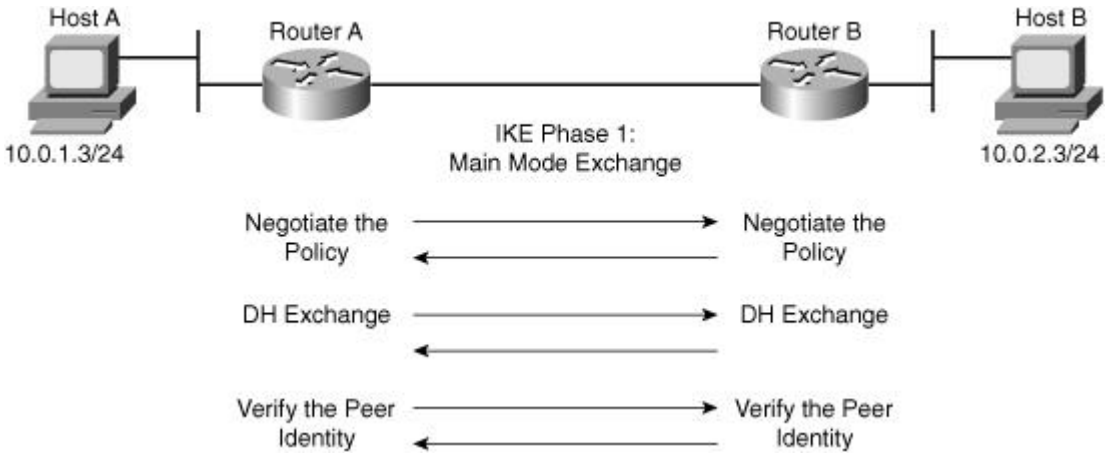
IKEv2 korjasi alkuperäisen version ongelmat. Vuoden 2016 artikkelissa Andrew Crouthamel, Cisco-akatemian opettaja, testasi IKEv1- ja IKEv2-uudelleenneuvotteluun kuluvaa aikaa ja toimivuutta 931 site-to-site VPN-verkon otoksessa. Näistä kaksi kolmasosaa (626) toimi IKEv1:llä ja yksi kolmasosa (305) IKEv2:lla. Failure-tilanteessa tunnelit joudutaan manuaalisesti korjaamaan tunnelit. [10.]

Taulukko 1. IKEv1 vs IKEv2 tulokset - Andrew Crouthamel 2016. [10.]

Version	DPD failure rate	Renegotiation time
IKEv1	55%	5min 30s
IKEv2	5%	15s

2.2.5 IKE Phase 1

IKE:n ensimmäisessä vaiheessa asennetaan turvallinen kanava toisen vaiheen neuvotteluille ja perustetaan ISAKMP SA:t. Tämä kanava luodaan siten, että molempiin VPN-tunnelin päihin konfiguroidaan vertaisparin IP-osoite ja sovitut turvallisuusparametrit. Nämä konfiguraatiot pitää olla yhteneväiset tunnelin molemmissa päissä ja IP-yhteys onnistua, jotta ensimmäinen vaihe onnistuu.



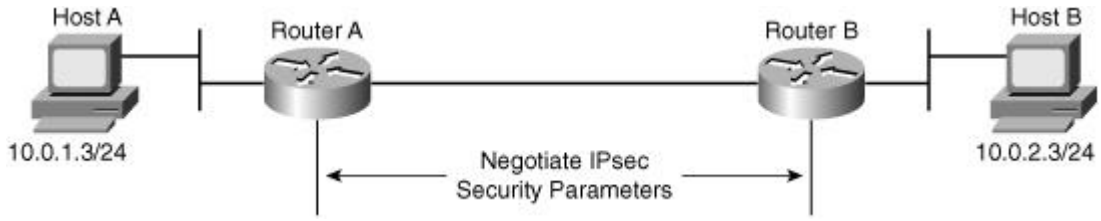
Kuva 5. IKE vaihe 1 neuvottelut. [11.]

Konfiguroitavat parametrit sisältävät:

- autentikointimetodi (sertifikaatti tai yhteinen salasana)
- käytettävä moodi (Main mode, Aggressive mode tai IKEv2)
- kryptausalgoritmi (DES, 3DES, AES)
- hajautusalgortimi (MD5, SHA)
- Diffie-Hellman-ryhmä
- NAT-Traversal.

2.2.6 IKE Phase 2

Toisessa vaiheessa kaksi päätepistettä käyttävät edellisessä vaiheessa luotua turvattua kanavaa neuvottelemaan IPSec SA:t. Toisen vaiheen konfiguraatiot pitää olla samat molemmissa VPN-päätelaitteissa, kuten ensimmäisessä vaiheessakin. Kun toinen vaihe on onnistunut, salattu VPN-tunneli on luotu ja sen yli pystyy siirtämään turvallisesti dataa.



Kuva 6. IKE vaihe 2 neuvottelu. [11.]

Toisen vaiheen konfiguroitavat parametrit:

- IPSec-moodi (Tunnel / Transport)
- IPSec-protokolla (AH / ESP)
- Perfect Forward Secrecy -ryhmä (PFS)
- tunnelin yli kommunikoivat aliverkot ja hostit
- kryptausalgoritmi (AES / 3DES)
- autentikointialgoritmi (SHA / HMAC-MD5)
- SA:n elinikä.

2.3 Autentikointialgoritmit

Autentikointialgoritmien, tai hash-algoritmien, tarkoitus on varmistaa datan eheys. Eheydellä tarkoitetaan, ettei data ole korruptoitunut tai ettei mahdollinen hyökkääjä ole muuttanut datassa jotain. Datan korruptoituminen voi merkitä monta eri asiaa riippuen kontekstista, mutta yleisesti se tarkoittaa, että systeemi ei toimi enää niin kuin sen on ollut alun perin tarkoitus.

Kaikki hash:it toimivat samalla perusperiaatteella; sisään syötetään jokin merkkijono, algoritmi suorittaa matemaattisen laskutoimituksen ja ulos tulee satunnaiselta näyttävä merkkijono. Tämä matemaattinen laskutoimitus on erilainen eri algoritmeissa ja sen suorittamiseen viemä aika riippuu hyvin pitkälti sisään syötetyn merkkijonon pituudesta. Ulos tulleen jonopituus on määritetty käytetyssä algoritmissa.

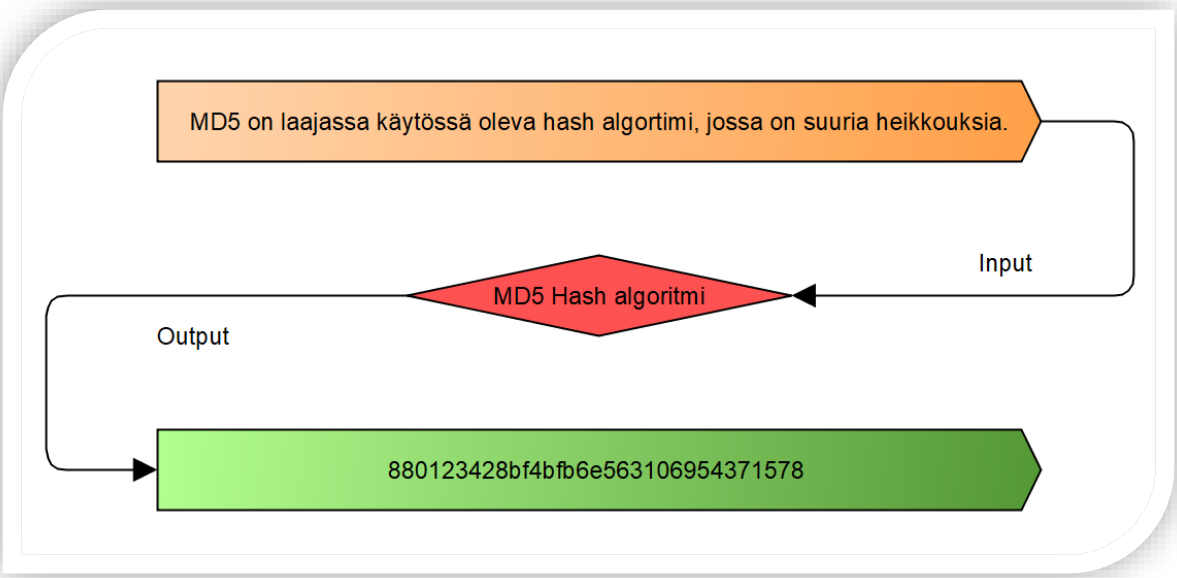
Algoritmit ovat deterministisiä, eli sama input tuottaa aina saman outputin. Juuri tämän takia pystytään huomaamaan, jos data on muuttunut – hash-algoritmi ei tuotakaan enää samaa ulostuloa. Collision-hyökkäykset ovat yleinen tapa yrittää murtaa algoritmeja. Niissä pyritään löytämään saman ulostulon tuottavia syöttöjä. Näitä löytyy sitä enemmän, mitä yksinkertaisempi algoritmi on.

2.3.1 HMAC-MD5

MD5 on laajassa käytössä oleva 128-bittinen hash-algoritmi. Se luotiin alun perin kryptograafiseen digitaaliseen allekirjoittamiseen, mutta siinä on havaittu suuria, vakavia heikkouksia. Sitä pidetään vanhentuneena eikä sitä tulisi käyttää muussa kuin salaamattomassa checksum-käytössä. Tästä huolimatta sitä edelleen käytetään joissakin laitoksissa.

MD5:n käyttäminen salasanojen kryptaamiseen on erittäin riskialtista, sillä MD5-kryptattujen salasanojen rikkominen on nopeata. NVIDIA GeForce 8800 Ultra, joka on kymmenen vuotta vanha näytönohjain ja pystyy laskemaan yli 200 miljoonaa MD5-hashia sekunnissa. Riippuen salasanan vahvuudesta ja siitä, ovatko salasanat ”suolattu”, kymmeniä miljoonia MD5-suojattuja salasananoja voidaan rikkoa jopa parissa sekunnissa.

Vuonna 2016 yli miljardi Yahoon käyttäjää joutuivat hakkeroiduksi. Käyttäjien tiedot varastettiin vuosina 2013 ja 2014. Tiedot sisälsivät nimiä, puhelinnumeroita, syntymäpäiviä, MD5:lla kryptattuja salasanoja ja salaamattomia turvallisuuskysymyksiä, joilla resetoida salasana. [12; 13; 14.]



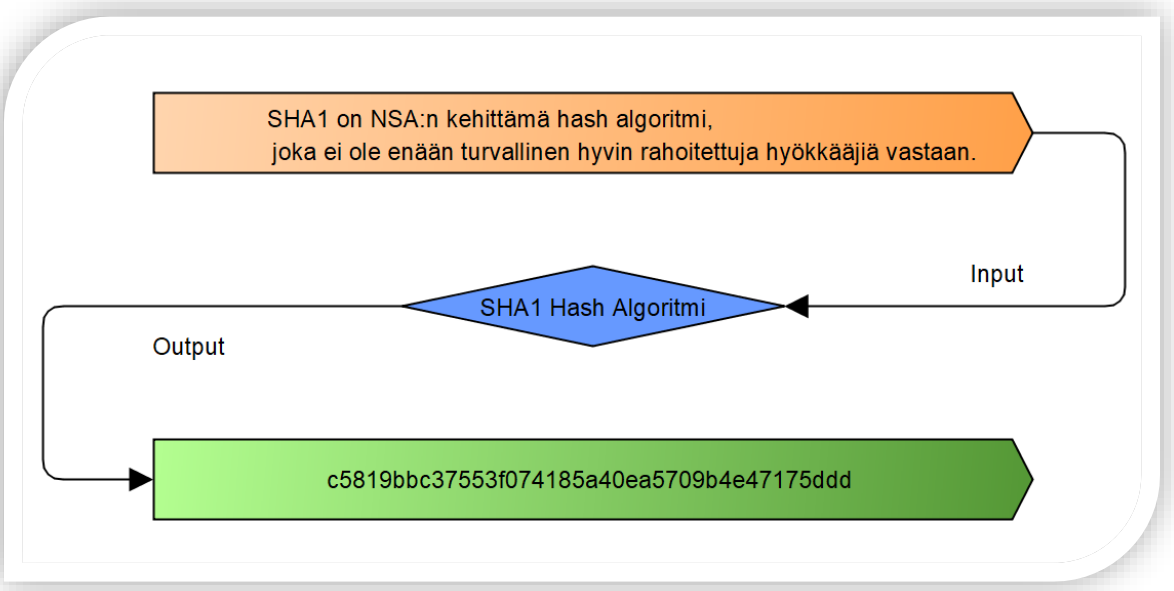
Kuva 7. MD5 hash algoritmi input/output.

HMAC, eli Hash-based Message Authentication Code, on yhdistelmä kryptograafisesta hash-funktiosta ja salaisesta kryptograafisesta avaimesta. Tätä voidaan käyttää minkä tahansa hash algoritmin kanssa. Kun tätä käytetään MD5:n kanssa, tulosta kutsutaan HMAC-MD5:ksi. HMAC:n hyöty on siinä, että ne ovat huomattavasti vahvempia collision-hyökkäyksiä vastaan kuin niiden kanssa käytettävät hash-funktiot itsessään. IPSec-standardi käyttää HMAC-versioita autentikointialgoritmeista. HMAC-MD5:ta pidetään edelleenkin turvallisena oikein käytettynä. Tämä ei kuitenkaan välttämättä ole totta tulevaisuudessa, yleisesti HMAC-SHA1 on suositellumpi vaihtoehto.

2.3.2 HMAC-SHA1 ja SHA2

SHA on perhe kryptografisia hash-funktioita. Näistä tunnetuimpia ja käytetyimpiä ovat SHA-1 ja SHA-2. SHA1 on 160-bittinen algoritmi, joka muistuttaa MD5-algoritmiä. SHA2 on setti hash-funktioita, johon kuuluu kuusi funktiota. Näistä SHA-256 ja SHA-512 ovat tärkeimmät. Niiden rakenteet ovat virtuaalisesti samat. Pitkälti vain bittien ja kierroksien määrä on eri.

NSA julkaisi SHA1:n vuonna 1995, se on pitkään ollut turvallinen, mutta vuodesta 2005 lähtien on myös ollut tiedossa, että se on haavoittuvainen teoreettisille hyökkäyksille. Digitaaliset sertifikointi auktoriteetit eivät ole saaneet jakaa SHA1-allekirjotettuja sertifikaatteja vuoden 2016 tammikuusta alkaen. Nyt vaikuttaisi siltä, että oltaisiin siirtymässä kokonaan SHA2- ja SHA3- funktioihin. [15.]



Kuva 8. SHA1 hash algoritmi input/output.

HMAC-SHA1 on IPSec:n toinen HMAC autentikointialgoritmivaihtoehto. HMAC paikkaa SHA1:n esiintyvät heikkoudet samalla tavalla kuin MD5:ssä. HMAC-SHA1-yhdistelmä on vahvasti turvallinen, jos käytettävä avain on yksityinen.

SHA2 ei ole saatavilla kaikkien valmistajien IPSec-implementaatioissa. SHA2:ssa on samoja heikkouksia kuten SHA1:ssä, mutta sen murtaminen on vaikeampaa ja siihen menee enemmän aikaa. Tämä kuitenkin on ongelma vain joissakin käyttötarkoituksissa. IPSec-tunnelit voidaan asettaa siten, että ne ovat turvallisia SHA-256:kin, asettamalla niiden eliniät lyhyiksi – esimerkiksi muutamaksi tunniksi. Täten SHA-256-avain vaihtuu useasti eikä hyökkääjillä riitä aika sen murtamiseen.

2.4 Salausalgoritmit

Salausalgoritmeja käytetään luottamuksellisuuden saamiseen IPSec-suojattuihin paketteihin. Nämä algoritmit voivat olla tehollisesti vaativia ja voivat vaikuttaa huomattavasti

kryptaavan laitteen suorituskykyyn. Juuri sen takia on tärkeää valita oikeat algoritmit ja mitä dataa halutaan suojata. Mitä vahvempaa salausta käytetään, sitä enemmän kryptaus vaatii laitteelta – hidastaen laitteen muita toimintoja. Reitittimissä tämä toimii porttikohtaisesti, jos yksi portti suorittaa kryptausta kaikki muukin portista kulkeva liikenne hidastuu, mutta muut portit toimivat normaalisti. Usein yrityksen reitittimestä yksi portti on yhteydessä internettiin - on turvallisuusriski olla ylimääräisiä linkkejä internettiin samasta verkosta. Tämän takia pitää olla valikoiva suojattavasta datasta tai dedikoida palvelin kryptaustarkoitukseen. Kryptausalgoritmin valintaan vaikuttaa suuresti sen turvallisuus, vaikutus suorituskykyyn ja se, kuinka nopeasti salaus onnistuu algoritmilta.

2.4.1 3DES

3DES-salaus on vanhaan 56-bittiseen DES-standardiin perustuva algoritmi, joka suorittaa DES-algoritmin kolme kertaa eri avaimilla. DES luotiin 1970-luvulla ja myöhemmin sen päätettiin olevan heikko brute-force-hyökkäyksiä vastaan, jonka seurauksena vuonna 1998 julkaistiin 3DES-standardi. 3DES:lla on 168-bittinen avain, kolme kertaa DES:in 56-bittisen avaimen verran. Tästä johtuen 3DES on myös hidas verrattuna sen tarjoamaan suojaan. OpenSSL ei enää sisällytä 3DES:aa, se on pitkälti korvattu vahvemmalla AES-salauksella, mutta vielä nykyäänkin on mahdollista salata IPSec-tunneleita 3DES:lla.

2.4.2 AES

Advanced Encryption Standard on DES:n jälkeläiseksi kehitetty salausstandardi. 1990 -luvun lopussa NIST, amerikkalainen teknologiastandardien kehittäjä, järjesti julkisen valintaprosessin, jonka tarkoituksena oli löytää symmetriseen avaimeen perustuva salausalgoritmi, joka pystyisi suojamaan herkkää dataa pitkälle 2000-luvulle. Kilpailun ensimmäiselle kierrokselle lähetettiin 15 salausalgoritmia 12 eri maasta. Rijndael, kahden belgialaisen kehittämä algoritmi, voitti kilpailun ja sen 128-, 192- ja 256-bittisestä versiosta tehtiin AES-standardi. Rijndaelin valintaan vaikutti sen yksinkertaisuus, toiminta eri alustoilla ja sen halpa implementaatio. [16.]

AES:aa vastaan on ollut pitkään erilaisia key-recovery-hyökkäysmetodeja, mutta oikein implementoituna jopa 128-bittistä AES:a vastaan ei löydy tehokasta keinoa murtaa sen

avainta. Brute-force-hyökkäyksellä kaikkien 128-bittisen avainten läpikäymiseen voi kulua miljardi miljardia vuotta. Tämän takia toistaiseksi AES on turvallinen algoritmi tiedon suojaamiseen.

3 Laitteisto

3.1 MikroTik

MikroTik on vuonna 1996 perustettu latvialainen tietotekniikka-alan yritys, joka erikoistuu verkkoratkaisuihin sekä hardware- että software-puolella. Pääasiassa yritys myy reitittimiä ja langattomia tuotteita. MikroTik käyttää RouterOS-käyttöjärjestelmää, joka perustuu Linuxin kerneliin. Se tulee yrityksen oman RouterBOARD-sarjan mukana, tai sen voi asentaa x86-pohjaisiin tietokoneisiin, mikä muuttaa nämä reitittäviksi laitteiksi. MikroTik tunnetaan hyvästä hinta-laatu-suhteestaan, joustavuudestaan ja sen halvoista VPN-reitittimistä. Tämän takia yrityksen laitteita käytetään erityisesti kehittyvissä IT-markkinoissa, pienyrityksissä tai kotona. Projektissa käytin Finnish Net Solutions Oy:lta saamaani RB951Ui-2HnD-reititintä käyttöjärjestelmään tutustumiseen ja VPN:n testaamiseen, RB2011UiAS-RM-reititintä käytettiin työn lopullisessa toimeenpanossa.

Kuva 9. RB2011UiAS-RM reititin. [17.]

Taulukko 2. RB2011UiAS-RM tuotetiedot. [17.]

10/100 Ethernet portit	5 kpl
Gigabit Ethernet portit	5 kpl
PoE portit	1 kpl (Eth10)
PoE sisään	Kyllä
PoE ulos	Kyllä
CPU	AR9344

CPU ytimet	1 kpl
CPU taajuus	600 MHz
USB portit	1 kpl microUSB tyyppi AB
Käyttöjärjestelmä	RouterOS
PCB lämpötilan monitorointi	Kyllä
Testattu normaalilämpötila	-35 ºC – +65 ºC
Virtapaikkoja	1 kpl
SFP portit	1 kpl
SFP diagnostiikka monitorointi	Kyllä
Serial portit	RJ45
RAM	128 MB
Tallennustila	128 MB
Tallennustilan tyyppi	NAND
Tuetut sisääntulo jännitteet	8-30 V
Jännitteen monitorointi	Kyllä
Hinta	$119 (~105€)

3.2 Skriptaus

RouterOS käyttää omaa sisäänrakennettua skriptauskieltään. Sen avulla käyttäjä voi automatisoida joitain huoltotöitä suorittamalla itsemääritettyjä skriptejä sidotuissa tapahtumissa, tai tiettyyn kellonaikaan. Tapahtumia, jotka käynnistävät skriptin, voidaan luoda käyttöjärjestelmän mukana tulevilla työkaluilla, kuten System Scheduler, Traffic Monitoring Tool tai Netwatch Tool. Näitä skriptejä on moniin tarkoituksiin ja niitä voi luoda itse tai käyttää jotakin yhteisön valmiiksi tekemää skriptiä. Tässä työssä skriptausta käytettiin IPsec VPN -tunnelin automaattiseen muodostumiseen IP-osoitteen muuttuessa.

3.3 pfSense

pfSense on avoimeen lähdekoodiin perustuva palomuuri ja reititin distribuutio, jonka pohjana toimii FreeBSD. Se on saanut huomiota sen luotettavuudesta ja tarjotuista toiminnoista. Sitä konfiguroidaan ja päivitetään suoraan web-käyttöliittymästä, eikä käyttäjä

tarvitse kokemusta FreeBSD:stä sen hallinnoimiseen. pfSense yleensä sijoitetaan verkon reunalle palomuuriksi, reitittimeksi tai VPN-pisteeksi. Projektissa käyttämämme SG-8860 1U on tarkoitettu keski- ja isokokoisten yritysten sekä palvelinkeskusten käyttöön.

Kuva 10. SG-8860 1U palomuuri. [18.]

Taulukko 3. SG-8860-1U tuotetiedot. [18.]

Gigabit Ethernet portit	6 kpl
CPU	Intel ”Rangeley” Atom C2758
CPU ytimet	8 kpl
CPU taajuus	2,4 GHz
RAM	8GB DDR3L Non ECC
USB portit	2 kpl USB 2.0
Console portti	Mini USB
Tallennustila	64GB eMMC Flash 128GB mSATA SSD 2x 150GB mSATA SSD (RAID1)
Käyttölämpötila	0 ºC – +60 ºC
Virtapaikkoja	1 kpl
Jännite	100-240 V
Jäähdytys	Integroitu jäähdytyslevy ja tuuletin
Max aktiiviset yhteydet	8 miljoonaa
Hinta	$1049 (~925€)

4 Vikasietoisuus

Vikasietoisuus on tärkeää korkeassa käytössä olevissa palvelinverkoissa. Vikasietoisuutta saadaan eri metodeilla, mutta usein se vaatii jonkinlaista investointia laitteisiin tai

palveluihin. Yleisin keino saada suojaa vikatilanteita varten on hankkia UPS-järjestelmä, joka torjuu sähkökatkoksista aiheutuvia ongelmia. Se kuuluu myös palvelinsalien perusvarusteluun. Toinen verkkoihin spesifimpi keino on kahdennus, jolla tarkoitetaan yksinkertaisesti toisen, samanlaisen laitteen hankkimista, joka asennetaan toimimaan varalle toisen rinnalla.

4.1 IP-osoitteen muutos

Työssäni yksi huolenaihe oli, että sivukonttorin palvelintilan julkinen IP-osoite saatiin DHCP:n kautta. Tämä on hankalaa sen takia, että IPSec VPN -tunnelia ei voida luoda kahden laitteen välille, jos toisen laitteen IP-osoite muuttuu kesken kaiken. Yksi vaihtoehto tämän korjaamiseksi on yksinkertaisesti hankkia kiinteä IP-osoite, mutta se maksaa ja on muita keinoja, kuten DDNS, joita voi käyttää tätä varten.

Yrityksellä oli jo valmiiksi DynDNS-tunnukset ja pääsy sen palveluihin, joten päätin käyttää tätä korjaamaan IP-osoitteen muutoksesta aiheutuvia ongelmia. DynDNS on suosittu yritys, joka tarjoaa DDNS-palveluita. DDNS liittää julkiseen IP-osoitteeseen DNS-nimen, jonka avulla laitteeseen voi ottaa yhteyden. Laite, tässä tapauksessa MikroTik-reititin, asetetaan kirjautumaan DynDNS-palveluun ja päivittämään DNS-nimeen liitetyn IP-osoitteensa.

MikroTik:ssa ei suoraan ole asetusta tämän tekemiseksi, mutta sen laajojen skriptausominaisuuksien avulla tämä voidaan asentaa. MikroTik:n omilta foorumeilta löytyy valmiiksi tehty DynDNS-skripti, jota käytän tässä työssä. Skripti asetetaan pyörimään minuutin väliajoin, jolloin se päivittää IP-osoitteen DDNS-nimeen, jos se on muuttunut.

DDNS:n avulla IPSec-tunnelin toinen päätelaite, pfSense:n reititin, voi asettaa MikroTik:n tunnelin pääksi sen DDNS-nimen. Täten kierretään muuttuva IP-osoitteesta johtuva ongelma.

4.2 Käyttäytyminen sähkökatkoksissa

Kriittistä sivukonttorin palvelinverkossa on, miten reititin toimii uudelleen käynnistymisessä ja sähkökatkoksen aikana. Palvelintilan virransyöttö on yhteydessä yhteen UPS-

laitteeseen, josta riittää varavirta noin 15 minuutiksi. UPS-systeemi lähtee toimintaan nopeasti, miltei samanaikaisesti, kun sähkökatkos tapahtuu. IPSec VPN -tunneli on väliaikaisesti alhaalla, jos virta katkeaa kokonaan, mutta reitittimen käynnistyessä uudelleen tunneli pystytetään uudelleen spesifioimalla MikroTik tunnelin pystyttäjäksi. Pitempään kestävässä sähkökatkoksessa tunneli on pakostakin alhaalla, mutta reitittimen konfiguraation takia tunnelin pitäisi pystyttää itsensä, kun virta palaa takaisin.

Jos IP-osoitteen muutos tapahtuu samanaikaisesti sähkökatkoksen yhteydessä, tunneli ei aukea ennen kuin DynDNS-skripti on ajettu. Parhaassa tapauksessa MikroTik:n Netwatch-työkalu huomaa IP-osoitteen muuttumisen ja ajaa päivityksen 1 minuutin päästä. Huonoimmassa tapauksessa Netwatch ei toimi kuten normaalisti ja system scheduler ajaa IP-osoitteen päivityksen 10 minuutin päästä viime päivityksestä. Koska tarve sivukonttoriin pääsyyn VPN-tunnelin kautta ei ole jatkuva, 10 minuuttia pitäisi olla riittävä aika tunnelin uudelleen muodostamiseen.

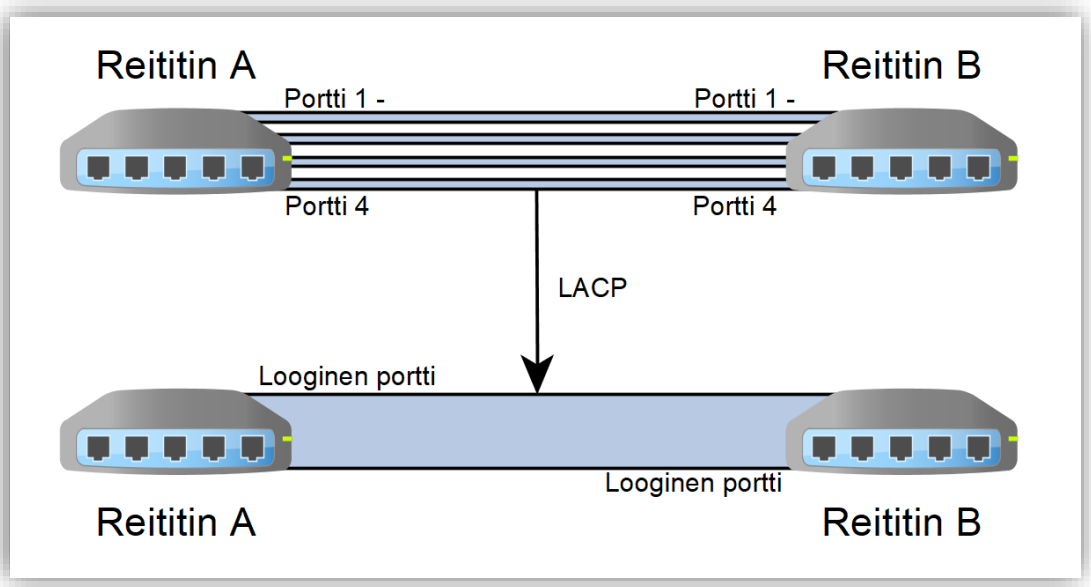
4.3 Korkea saatavuus

Korkea saatavuus, eli HA, on yksi suurista vikasietoisuuden parantajista verkkoympäristöissä. HA:ta luodaan laitteiden redundanssilla, virtualisoinnilla ja protokollien, kuten LACP:n, VRRP:n ja CARP:n käytöllä. Korkeaa saatavuutta kehitetään erityisesti suurissa ja kasvavissa yritysten verkkoympäristöissä. Laitteiden redundanssilla käytännössä tarkoitetaan kahdennusta, eli laitteiden tuplaamista siten, että kun toinen laite kaatuu, toinen ottaa tilanteen haltuun. Nämä kahdennetut laitteet konfiguroidaan siten, että ne ovat suurilta osin identtisiä keskenään ja ne implementoidaan osaksi verkkotopologiaa. Kahdennettujen laitteiden implementointi onnistuu aiemmin mainittujen protokollien avulla.

4.3.1 LACP

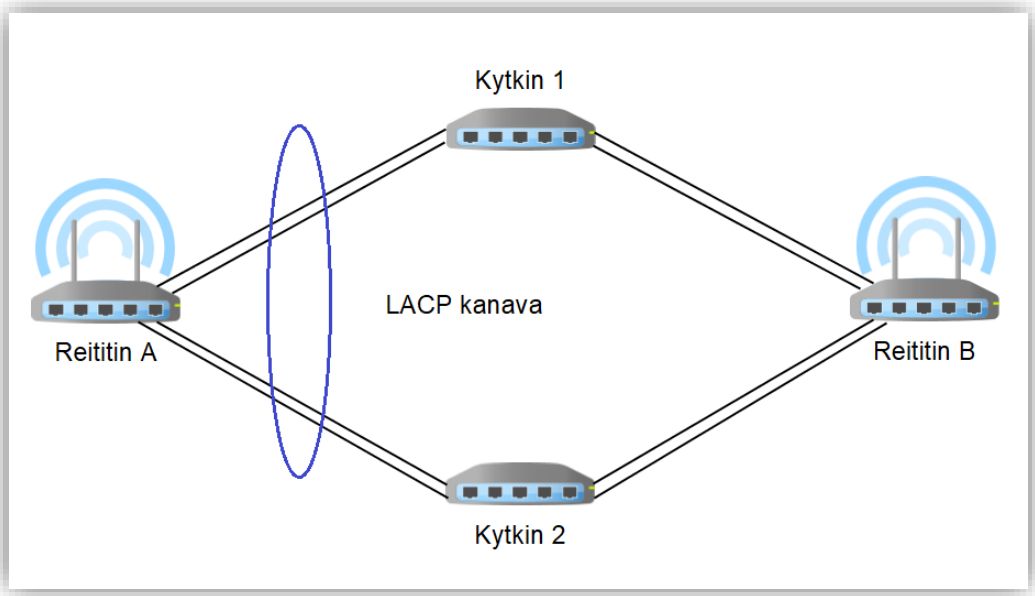
LACP on IEEE:n kehittämä standardiprotokolla, joka sallii kahden tai useamman fyysisen verkkoportin yhdistämisen yhdeksi loogiseksi portiksi. Esimerkiksi kaksi kytkimen porttia voi olla kiinni kahdessa reitittimen portissa ja LACP:lla nämä kaksi linkkiä yhdistetään yhdeksi loogiseksi portiksi. Jos ensimmäinen porteista menee pois päältä, ottaa

LACP heti toisen porteista käyttöön, eikä tästä aiheudu huomattavaa yhteyden katkeamista. Tätä voidaan myös käyttää verkkoliikenteen painon jakamiseen näiden kahden portin välillä, ja suorituskyky paranee.



Kuva 11. Linkkien yhdistäminen LACP:n avulla.

LACP:ta pystytään myös käyttämään yhdistämään kaksi kytkimen fyysistä porttia kahden eri reitittimen yhteen porttiin. Tällä tavalla saadaan yksi looginen yhteys yhdestä



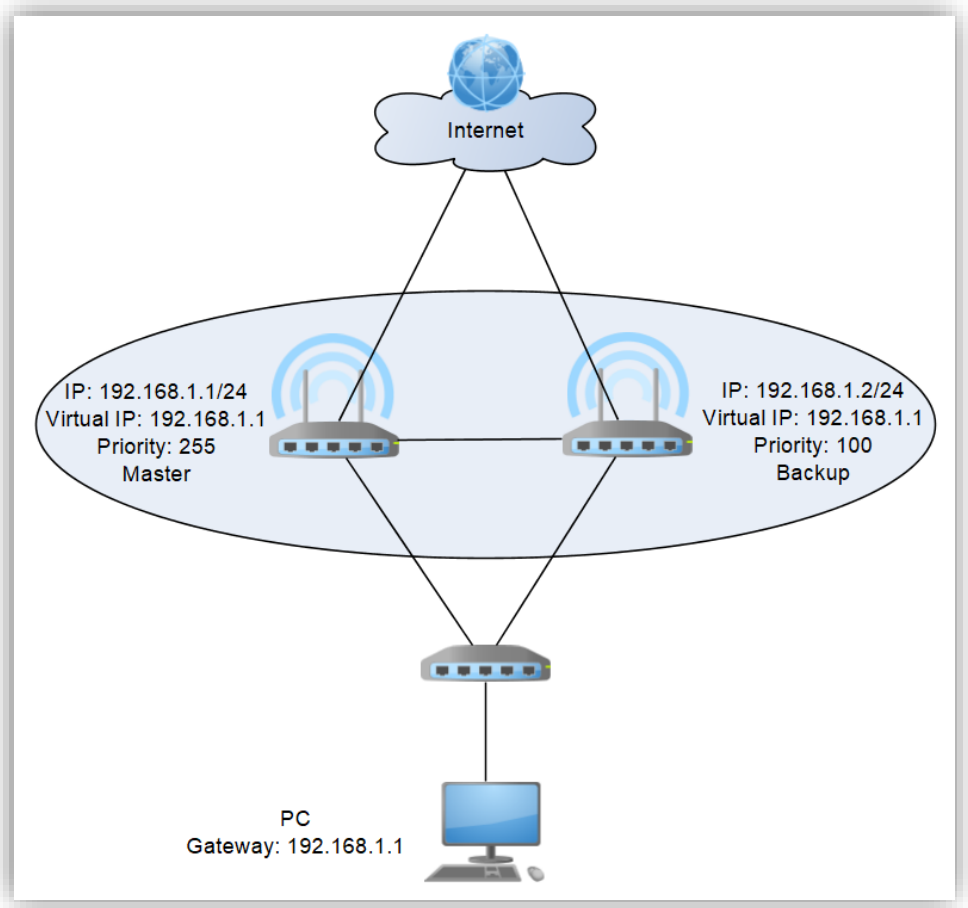
Kuva 12. Looginen linkki kahteen laitteeseen.

kytkimestä kahteen reitittimeen. Vikatilanteessa, jossa yksi reitittimistä kaatuu, looginen yhteys pysyy yhä ylhäällä, koska toisen reitittimen portti on yhteydessä kytkimen porttiin.

4.3.2 VRRP

VRRP on IETF:n RFC 5798:ssa kuvattu avoin standardi protokolla, jolla pystytään usean reitittimen virtualisointiin. VRRP on luotu perustuen Ciscon alkuperäiseen HSRP-virtualisointiprotokollaan. Ne eivät kuitenkaan ole yhteensopivia.

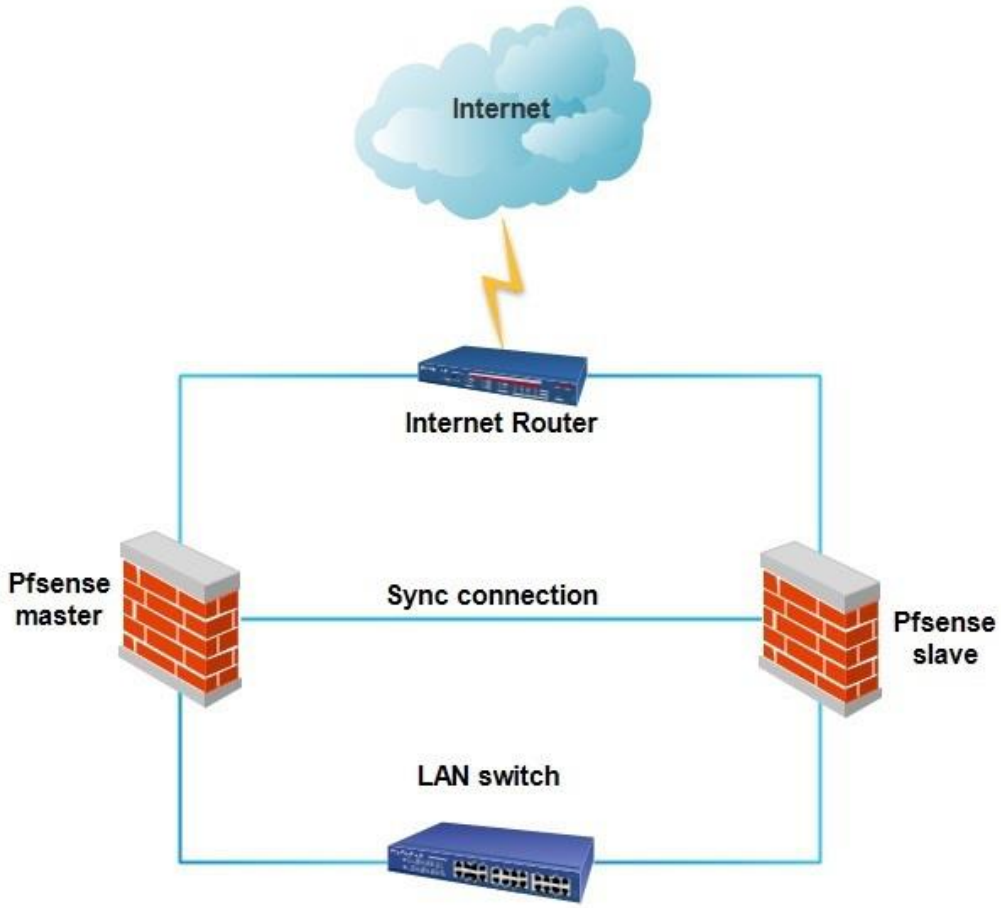
Reitittimen virtualisoinnilla implementoidaan kahdennusta ja parannetaan luotettavuutta ja saatavuutta verkkojen reitityksessä. VRRP luo virtuaalisen reitittimen, joka edustaa useaa fyysistä reititintä. Näistä yksi on master ja loput backup-reitittimiä. Aktiivisen reitittimen kaatuessa yksi backup-reitittimistä ryhtyy aktiiviseksi reitittimeksi, ja verkkoliikenne jatkuu normaalisti. Virtuaalisella reitittimellä on oma IP-osoitteensa ja tätä osoitetta käytetään default gatewayna verkossa.



Kuva 13. VRRP esimerkki topologia.

4.3.3 CARP

CARP on pfSensen oma protokolla, joka luotiin VRRP:n vaihtoehtoisena keinona virtualisointiin. CARP:n perusidea on kuitenkin sama kuin VRRP:ssäkin. Sitä käytetään osana CARP-klusteria, johon kuuluu myös pfsync ja konfiguraatioden synkronointi. Näiden avulla pystytään synkronoimaan kahden pfSense-reitittimen tauluja ja konfiguraatiota, ja luodaan korkean saatavuuden klusteri.



Kuva 14. CARP esimerkki. [19.]

5 Sivukonttorin verkon suunnittelu

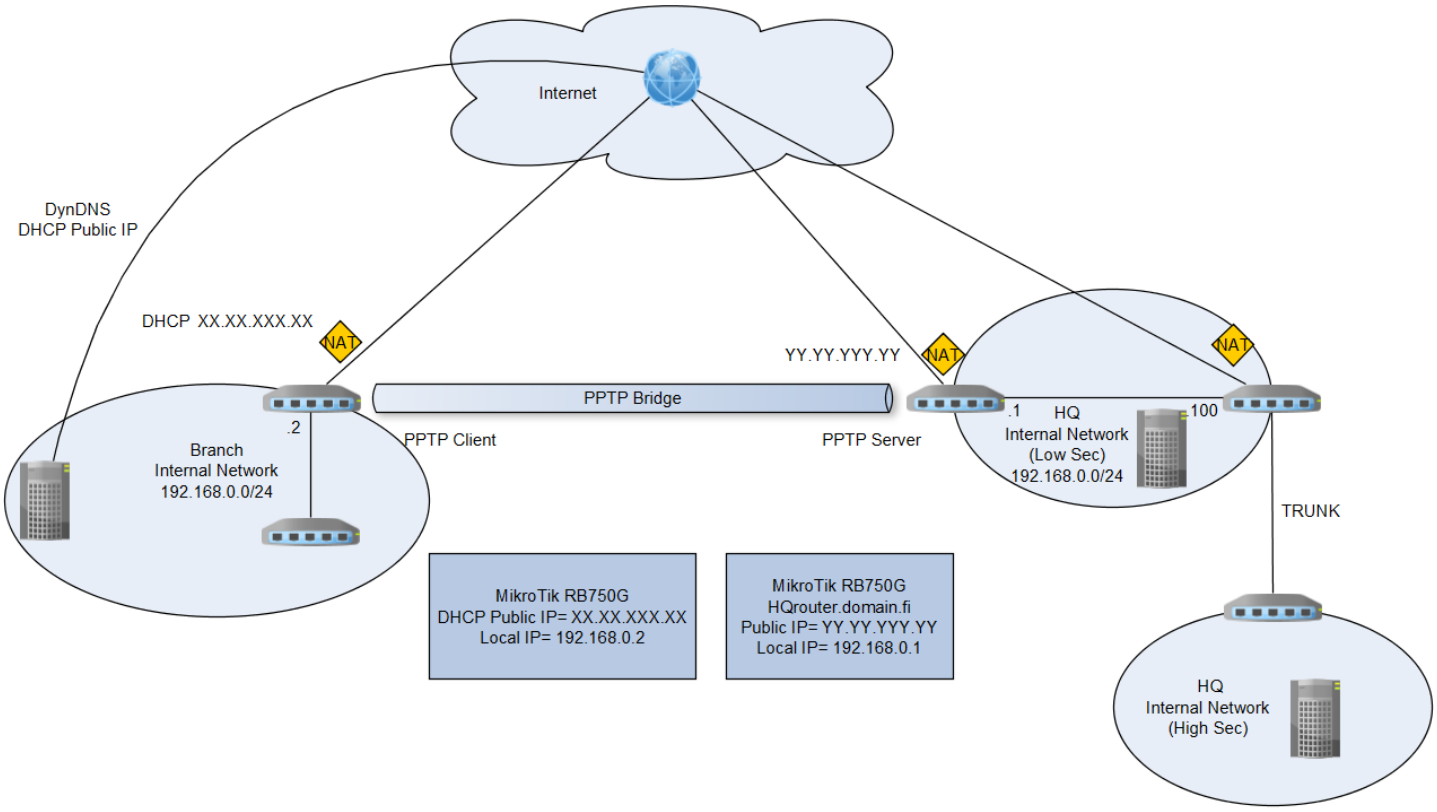
Yrityksen sivukonttorilla on oma pieni palvelintilansa, jossa on palvelimia ja NAS-laitteita. Päivittäistä liikennettä verkkoon ei ole, mutta laitteisiin tulisi päästä helposti käsiksi tarpeen vaatiessa. Aiemmin kahta verkkoa yhdisti PPTP-silta kahden MikroTik-reitittimen välillä, mutta siinä oli esiintynyt ongelmia, ja lopulta sen toiminta oli loppunut kokonaan.

Tehtävänantooni kuului IPSec-tunnelin pystyttäminen näiden kahden palvelinverkon välille ja varmistaa tunnelin vikasietoisuutta.

5.1 Vanha verkko

Sivukonttorin reititin saa julkisen IP-osoitteensa DHCP:llä. IPSec-tunnelia ei pystytä luomaan automaattisesti, jos se ei jostain saa muuttunutta IP-osoitetta. MikroTik tarjoaa uudemmissa RouterOS:ä ilmaista DDNS-palvelua, jonka avulla reitittimen IP-osoitteen saa aina sen DDNS-nimestä. MikroTik-laitteen DDNS-nimi on aina formaatissa MAC-osoite.sn.mynetname.net, eikä sitä pysty muokkaamaan. Tämä on epäkätevää administraatioon, joten päätimme sen sijaan käyttää DynDNS:n maksullista palvelua. Vanhassa topologiassa palvelinverkot oli yhdistetty PPTP-sillalla. PPTP on tunnetusti huono turvallisuudeltaan ja siinä on fundamentaalisia heikkouksia.

Yrityksessä oli samaan aikaan työn alla muutoksia sen verkkoinfrastruktuurissa. Aiemmin kaikki palvelinverkot oli yhdistetty samaan sillattuun IP-aliverkkoon. Päätettiin, että sivukonttoreiden ja muiden hajautetuiden tilojen palvelinverkoissa siirrytään omaan reititettyyn aliverkkoon. Tämä fasilitoi myös IPSec VPN -tunneleita, sillä IPSec toimii OSI-mallin 3-tasolla, ja se tarvitsisi 2-tason L2TP-tunnelin yhdistääkseen molemmat saman aliverkon alle. L2TP-tunnelin ajaminen IPSec-tunnelin sisältä on mahdollista eikä mitenkään epätavallista, mutta se olisi turhaa, koska kun yritys kasvaa, on kannattavampaa siirtyä reititettyyn verkkoon kuin jatkaa sillatussa verkossa.



Kuva 15. Vanha palvelinverkkoja yhdistävä tunneli.

5.2 Uusi verkko

Uudessa verkossa siirrytään IPSec VPN -tunneliin ja reitittimet korvataan uusilla MikroTik- ja pfSense-reitittimillä. IP-aliverkot muutetaan sivukonttorilla. Täten sivukonttorin ARP-taulut rajoitetaan oman aliverkkonsa sisään. Tunnelin liikenne rajoitetaan sivukonttorin palvelinverkon ja pääkonttorin runkoverkon välille. Liikenne rajoitetaan ”High Sec” -alueelta.

Uuden MikroTik-reitittimen DHCP:llä tuleva IP-osoite on liitetty DynDNS-palveluun ja tunneli muodostetaan pfSense reitittimen kiinteän IP-osoitteen ja MikroTik:in DynDNS nimen välille. Tunnelin vikasietoisuus varmistetaan MikroTikin puolella skriptien ja Netwatchin avulla.

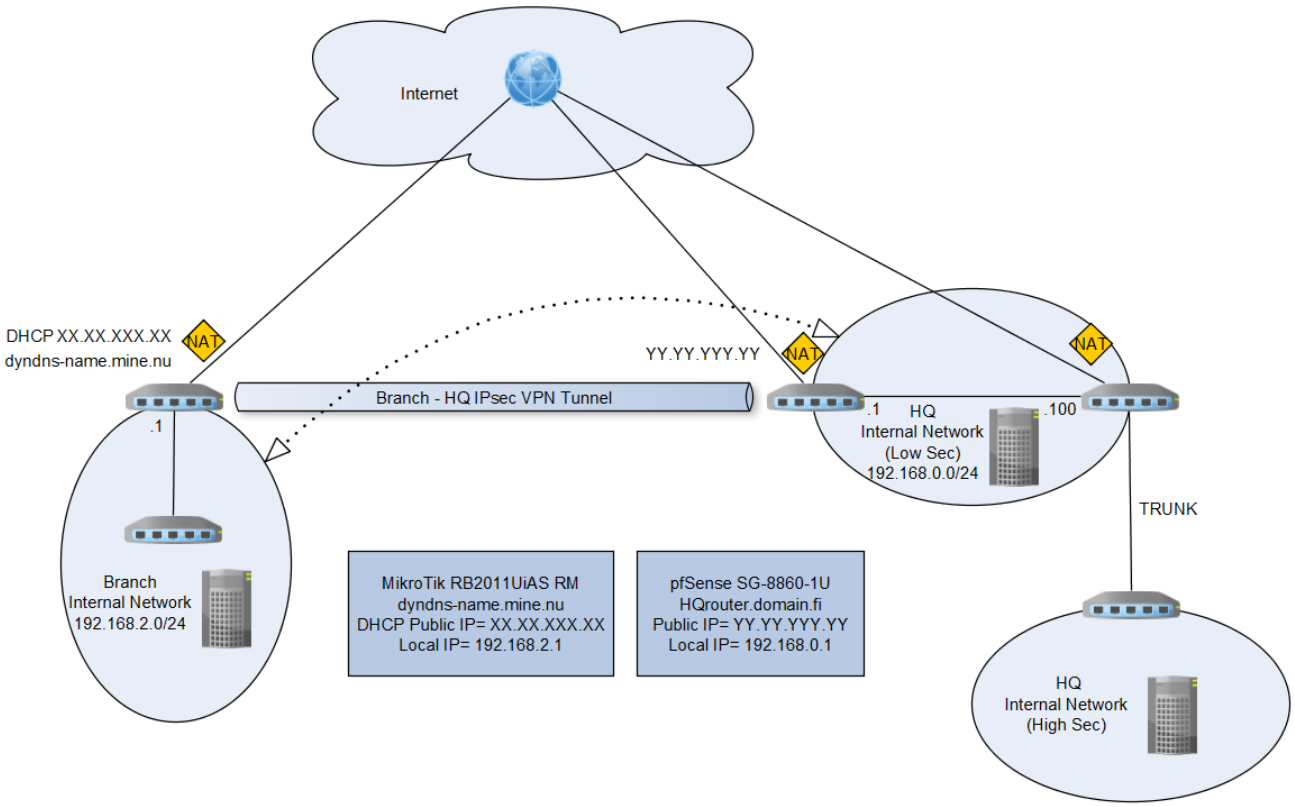
IPSec moodina käytetään tunnel moodia, koska yhdistetään kaksi kiinteää aliverkkoa. Protokollana pidetään ESP:tä, koska se on ylivoimaisesti parempi vaihtoehto. Turvallisuus parametreiksi asetetaan salausalgoritmiksi AES-256 ja todennusalgoritmiksi SHA-256, sillä ne ovat vahvoja algoritmeja, joita ei suurella todennäköisyydellä pystytä murtamaan.

Tämän takaamiseksi asetamme myös avainten eliniäksi 8 tuntia ja käytämme jaettuna salasanana 24-merkkistä, satunnaisesti generoitua PSK:ta, joka sisältää isoja ja pieniä kirjaimia sekä numeroita. Salasanan kompleksisuus on yksi suurimmista turvallisuuden takaajista eikä siitä aiheutu hallinnan osalta ylimääräistä työtä, sillä salasana pitää asettaa molempiin reitittimiin vain kerran.

Exchange-moodi asetetaan IKEv2:ksi, sillä se on helppo ja turvallinen vaihtoehto. Diffie-Hellman- ja PFS -ryhmäksi valittiin ryhmä 2, eli modp1024. Se on VPN-tunneleiden oletusasetus, joka tarjoaa perusturvaa pienellä vaikutuksella suorituskykyyn. MikroTik-päätepiste laitetaan vielä IPSec-keskustelun aloittajaksi, koska sen IP-osoite on sattumanvaraisempi ja täten on varmempi aloittaja.

Sivukonttorin palvelintilan vikasietoisuus IP-osoitteen muutoksen lisäksi riippuu pitkälti mahdollisista sähkökatkoksista. Palvelintilassa on yksi UPS, jossa virta riittää 15 minuutiksi katkoksen aikana. MikroTik asennetaan siten, että jos siitä katkeaa virta, se pystyttää tunnelin heti käynnistyttyään. Koska palvelintilassa käytetään vain yhtä reititintä,

mahdollinen vikapiste on MikroTik. Palvelimiin pääsy VPN-tunnelin kautta ei ole niin kriittistä, että siihen asennettaisiin HA-ratkaisu. Palvelinhuoneessa on kuitenkin varareititin, jonka voi vaihtaa MikroTik:n tilalle, jos reititin menee epäkuntoon.



Kuva 16. Uusi palvelinverkkojen yhdistävä tunneli.

6 IPSec-tunneleiden pystytys

6.1 Tunneli sivukonttorin palvelinsaliin

Tunnelin pystyttäminen alkaa reitittimien konfiguroimisella Tätä varten otan yhteyden reitittimeen SSH:lla. Reitittimissä on valmiiksi konfiguroitu julkiset IP-osoitteet, yhteys internettiin on testattu ja aliverkot on luotu. Ensimmäinen askel tunnelin pystyttämiseen on palomuurin ja NAT:n sääntöjen asettaminen siten, että ne sallivat IPSec-tunnelin pystyttämisen. IPSec käyttää IKE- ja ESP-protokollia, jotka jäävät palomuuriin kiinni, ellei niitä sallita erikseen. Tämä on hyvä tehdä ensimmäisenä, ettei myöhemmin kulu aikaa ongelmien ratkaisuun. [20; 21.]

/ip firewall filter

```
add chain=input action=accept protocol=udp port=500 comment=”IKE”

add chain=input action=accept protocol=ipsec-esp comment=”IPsec ESP protocol”
```

```
3    ;;; IKE
     chain=input action=accept protocol=udp src-port=500 dst-port=500 log=no
     log-prefix=""

4    ;;; IPsec ESP protocol
     chain=input action=accept protocol=ipsec-esp log=no log-prefix=""
```

Kuva 17. IPSec palomuurisäännöt.

Myös NAT-säännöt tulee asettaa, jotta palvelinverkot voivat keskustella, kun tunneli on pystytetty. Normaalisti NAT kääntää yksityisen IP-osoitteen julkiseksi osoitteeksi, kun internetiin kulkee liikennettä. Sääntöihin pitää määrittää, että tunnelia käyttäviä verkkoja ei käännetä NAT:lla. MikroTik:in ”masquerade” -ominaisuus suorittaa NAT-kääntämisen, kun julkinen IP-osoite saadaan DHCP:llä.

/ip firewall nat

```
add chain=srcnat action=accept src-address=192.168.2.0/24 dst-address=
192.168.0.0/24

add chain=srcnat action=masquerade out-interface=ether1
```

```
[jimi@         -sisaverkko] /ip firewall nat> print
Flags: X - disabled, I - invalid, D - dynamic
 0    ;;;
      chain=srcnat action=accept src-address=192.168.2.0/24
      dst-address=192.168.0.0/24 log=no log-prefix=""

 3    ;;; defconf: masquerade
      chain=srcnat action=masquerade out-interface=ether1 log=no log-prefix=""
[jimi@         -sisaverkko] /ip firewall nat>
```

Kuva 18. MikroTik NAT asetukset.

Kun NAT ja palomuuri säännöt on asetettu, voidaan konfiguroida IPSec VPN. Ensimmäisenä konfiguroidaan tunnelin päät, eli peerit. Peerin osoitteeksi laitan pääkonttorin

pfSensen julkisen IP-osoitteen. pfSense:ssä peerin osoitteeksi voi laittaa DDNS-nimen. Tätä konfiguraatiota käytetään yhdistämään IKE daemonit ensimmäisessä vaiheessa, jossa neuvotellaan avaimet ja algoritmit SA:tä varten.

/ip ipsec peer

```
add address=YY.YY.YYY.YY/32 auth-method=pre-shared-key secret="Vahva PSK"
exchange-mode=ike2 send-initial-contact=yes hash-algorithm=sha256 enc-algo-
rithm=aes-256 comment="IPsecVPN"
```

```
[jimi@         -sisaverkko] /ip ipsec> peer print
Flags: X - disabled, D - dynamic, R - responder
 0      ;;;  IPsecVPN
        address= XX.XX.XXX.XX/32    auth-method=pre-shared-key
        secret="     Vahva PSK          " generate-policy=no
        policy-template-group=default exchange-mode=ike2
        send-initial-contact=yes hash-algorithm=sha256 enc-algorithm=aes-256
        dh-group=modp1024 dpd-interval=2m dpd-maximum-failures=5
```

Kuva 19. MikroTik IPSec peer konfiguraatio.

Kun peer on asetettu molempiin tunnelin päihin, vaihe yksi on suoritettu ja sitä vastaavat SA:t luotu. Ensimmäisen vaiheen onnistumisen voi varmistaa tarkistamalla, että tunnelinpäät ovat lisänneet toisensa vertaistaulukkoonsa.

```
[jimi@         -sisaverkko] /ip ipsec remote-peers> print
 0 local-address= XX.XX.XXX.XX    remote-address= YY.YY.YYY.YY    state=established
   side=responder established=1h39m22s

 1 local-address= XX.XX.XXX.XX    port=4500 remote-address= YY.YY.YYY.YY    port=4500
   state=established side=responder established=1h38m55s
[jimi@         -sisaverkko] /ip ipsec remote-peers>
```

Kuva 20. Reitittimet ovat lisänneet toisensa remote-peereihin.

Nyt siirrytään vaiheeseen kaksi, jossa luodaan itse IPSec-tunneli. Tähän konfiguroidaan turvallisuusparametreja samalla tavalla kuten ensimmäisessä vaiheessa. MikroTik:ssa vaihe kaksi asetetaan proposal- ja policy-asetuksilla. Proposal-asetuksissa valitaan, mitä autentikointi- ja kryptausalgoritmeja käytetään IPSec-tunnelissa, sekä avainten elinikä. Kun avainten elinikä loppuu, ne korvataan uusilla. Perfect Forward Secrecy (PFS) takaa,

ettei uusia avaimia ei derivoida vanhoista avaimista. Jos ulkopuolinen saa käsiinsä yksityisen avaimen hän pääsisi käsiksi vain sillä avaimella suojattuun liikenteeseen, mutta ei mihinkään dataan tulevaisuudessa. Päätimme käyttää SHA-256-autentikoinnissa ja AES-256-kryptauksessa lyhyellä 8 tunnin eliniällä, sekä PFS modp1024-ryhmällä turvallisuuden parantamiseksi pitkällä aikajanalla.

/ip ipsec proposal

```
set default auth-algorithms=sha256 enc-algorithms=aes-256-cbc lifetime=8h pfs-group=modp1024
```

```
[jimi@         -sisaverkko] /ip ipsec proposal> print
Flags: X - disabled, * - default
 0  * name="default" auth-algorithms=sha256 enc-algorithms=aes-256-cbc
      lifetime=8h pfs-group=modp1024
[jimi@         -sisaverkko] /ip ipsec proposal>
```

Kuva 21. IKE vaihe 2 – Proposal.

Policy-asetuksissa määrätään, mitä IPSec-protokollaa käytetään, ESP:tä vai AH:ta, sekä mitä IPSec-moodia käytetään ja mikä liikenne sallitaan tunnelin yli. ESP suoriutuu yleisesti paremmin kuin AH, jonka takia se valittiin. SA-source ja SA-destination merkkaavat kuvan 14 mukaan tunnelin muodostavia kahta reititintä. Kommentti kenttää käytetään MikroTik:n skripteissä, joilla pävitän laitteen julkisen IP-osoitteen, joka tulee DHCP:llä.

/ip ipsec policy

```
add src-address=192.168.2.0/24 dst-address=192.168.0.0/24 action=encrypt level=require ipsec-protocols=esp tunnel=yes sa-src-address=XX.XX.XXX.XX sa-dst-address=YY.YY.YYY.YY proposal=default comment=”IPsecVPN”
```

```
[jimi@         -sisaverkko] /ip ipsec policy> print
Flags: T - template, X - disabled, D - dynamic, I - invalid, A - active,
* - default
 0 TX* group=default src-address=::/0 dst-address=::/0 protocol=all
       proposal=default template=yes

 1  A  ;;; IPsecVPN
       src-address= 192.168.2.0/24 src-port=any dst-address= 192.168.0.0/24
       dst-port=any protocol=all action=encrypt level=require
       ipsec-protocols=esp tunnel=yes sa-src-address= XX.XX.XXX.XX
       sa-dst-address= YY.YY.YYY.YY    proposal=default priority=0 ph2-count=2
```

Kuva 22. IKE vaihe 2 – Policy.

Nyt tunnelit on pystytetty. Tämän voi tarkastaa tulostamalla SA:t. Kun tunnelit on pystytetty, kaikki näiden kahden aliverkon välillä kulkeva liikenne on suojattu.

```
[jimi@         -sisaverkko] /ip ipsec installed-sa> print
Flags: A - AH, E - ESP
 0 E spi=0xDE16002 src-address= YY.YY.YYY.YY    dst-address= XX.XX.XXX.XX
     state=mature auth-algorithm=sha256 enc-algorithm=aes-cbc
     enc-key-size=256 auth-key="fbbce0c87d2b5a6bdcd072662a3b4a8ba75f2b05cf00099
         deee28b58774d0e9c"
     enc-key="8ddb05e50ea60e3e0643393f3b8731577161ba9c030ecfdd3c21c8fb897d8181"

     add-lifetime=6h24m1s/8h2s replay=128

 1 E spi=0xC8722907 src-address= XX.XX.XXX.XX    dst-address= YY.YY.YYY.YY
     state=mature auth-algorithm=sha256 enc-algorithm=aes-cbc
     enc-key-size=256 auth-key="e91a53bc554c775e6217ebb8ba1730b17d0078ac9663426
         bff9b87c195ce4599"
     enc-key="888ad91ac6acf6912919666f4bf49a44e75aa47463aa1e3f25e7ae2af6d0eff4"

     addtime=feb/17/2017 13:01:09 expires-in=7h14m19s
     add-lifetime=6h24m1s/8h2s current-bytes=204 replay=128

 2 E spi=0x804405F src-address= YY.YY.YYY.YY    dst-address= XX.XX.XXX.XX
     state=mature auth-algorithm=sha256 enc-algorithm=aes-cbc
     enc-key-size=256 auth-key="117056c48cdfadc29642e6eb12f1372ec870250de3823a5
         9cb930d4f4259b68c"
     enc-key="da6e175989248a6cb96ff23cda3d1d50528118624399a79b624a52489bfaa513"

     add-lifetime=6h24m16s/8h21s replay=128

 3 E spi=0xC3162D90 src-address= XX.XX.XXX.XX    dst-address= YY.YY.YYY.YY
     state=mature auth-algorithm=sha256 enc-algorithm=aes-cbc
     enc-key-size=256 auth-key="05e8681963ec498aab654c3d278fa0c00a85983d315b029
         7a182e68c04a81bfe"
     enc-key="c94b067faf9d6d6d8dd753c2acb85edc6ea329cfb02fe6f846f6ddf0db8421a7"

     add-lifetime=6h24m16s/8h21s replay=128

[jimi@         -sisaverkko] /ip ipsec installed-sa>
```

Kuva 23. Asennetut SA:t.

MikroTik:ssä lisäksi säädän vielä skriptit, joiden avulla päivitetään DynDNS ja muutetaan IP-osoite -policy konfiguraatioon. Nämä skriptit pyöritetään system schedulerin ja net-watch-työkalun avulla. Skriptit löytyvät liitteestä 1.

System scheduler asetetaan ajamaan DynDNS-päivitys 10 minuutin välein ja pakotettu DynDNS päivitys kerran päivässä.

```
[jimi@         -sisaverkko] > system scheduler
[jimi@         -sisaverkko] /system scheduler> print
Flags: X - disabled
 #    NAME    START-DATE  START-TIME   INTERVAL            ON-EVENT     RUN-COUNT
 0    Dyn...  jan/24/2017 00:00:00     1d                  /system...          25
 1    DynDNS  jan/23/2017 10:00:00     10m                 /system...        2809
[jimi@          sisaverkko] /system scheduler> print detail
Flags: X - disabled
 0   name="DynDNS Force" start-date=jan/24/2017 start-time=00:00:00 interval=1d
     on-event=/system script run DynDNS-Force\r\n/system script run ipsec-
         policy-update
     owner="jimi" policy=read,write,policy,test run-count=25
     next-run=feb/18 00:00:00

 1   name="DynDNS" start-date=jan/23/2017 start-time=10:00:00 interval=10m
     on-event=/system script run DynDNS\r\n/system script run ipsec-policy-
         update
     owner="jimi" policy=read,write,policy,test run-count=2809
     next-run=13:30:00
[jimi@         -sisaverkko] /system scheduler>
```

Kuva 24. System scheduler.

Netwatch työkalua käytetään vielä yhtenä varajärjestelmänä IP-osoitteen muuttuessa. Netwatch pingaa MikroTikin julkista IP osoitetta kerran minuutissa, jos ping ei mene läpi DynDNS-päivitys skripti ajetaan.

```
[jimi@         -sisaverkko] > tool netwatch
[jimi@         -sisaverkko] /tool netwatch> print
Flags: X - disabled
 #    HOST                TIMEOUT             INTERVAL            STATUS
 0    ;;; IPsecVPN
      XX.XX.XXX.XXX       1s                  1m                  up
[jimi@         -sisaverkko] /tool netwatch> print detail
Flags: X - disabled
 0    ;;; IPsecVPN
      host=XX.XX.XXX.XXX   timeout=1s interval=1m since=feb/09/2017 14:33:05
      status=up up-script=""
      down-script=system script run DynDNS\r\nsystem script run ipsec-policy-
          update
[jimi@         -sisaverkko] /tool netwatch>
```

Kuva 25. Netwatch-työkalu.

Nyt reititin oli konfiguroitu VPN-yhteyttä varten, jonka jälkeen kävimme asentamassa reitittimen sivukonttorin palvelinsaliin. Kun kävimme salissa, teimme samalla aliverkon IP-osoitteiden vaihtamisen vastaamaan uutta 192.168.2.0/24 -verkkoa ja testasimme kaikki laitteet ping-komennon avulla sekä lokaalisti, että tunnelin kautta.

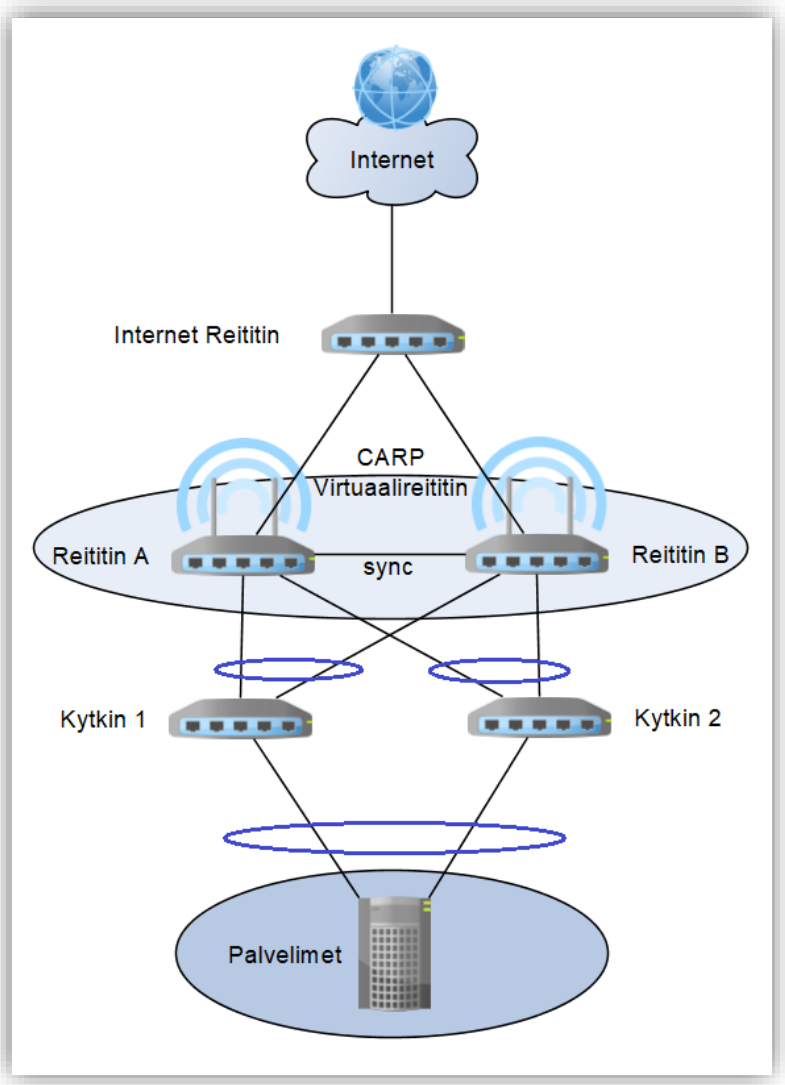
6.2 Tunnelit ulkomailla oleviin palvelinsaleihin

Ruotsiin luotiin kaksi uutta palvelinsalia, jotka tulivat pyörittämään yrityksen Ruotsin asiakkaiden verkkopalvelua. Näiden salien vikasietoisuus oli tärkeätä, jonka takia niissä otettiin käyttöön kahdennus myös VPN-tunneleissa. Tämä onnistui käyttämällä kahta pfSense-reititintä, joista luotiin yksi virtuaalinen reititin, CARP-protokollan avulla. IPSec VPN:n implementoiminen CARP-korkeasaatavuuden kanssa ei eronnut paljoa IPSec:n puolelta. Ainoa suuri ero oli, että IPSec-päätepisteeksi valittiin ulkomaalla olevan pfSensen virtuaalinen julkinen IP-osoite, fyysisen reitittimen julkisen IP-osoitteen sijasta.

Tunneleita luotiin kolme kappaletta: yksi yhdistämään molempia palvelinsaleja, ja yksi molemmista saleista pääkonttorin palvelinsisäverkkoon. Lisäksi nämä kaksi ulkomailla sijaitsevaa palvelintilaa käyttivät kahta eri ISP:tä ja yhdistettiin MPLS-protokollalla. Täten jos toisella ISP:llä on ongelmia yhteyksien kanssa, toinen palvelintiloista on tavoitettavissa. MPLS on ISP:den tarjoama kalliimpi uniikki VPN-ratkaisu. Se on vikasietoinen ja katsottiin, että näiden kahden palvelinsalin välinen yhteys on tärkeä.

Kahdennusta palvelinsaleissa toteutettiin reitittimissä, kytkimissä, palvelimissa, NAS:ssa ja virransyötössä. Virransyöttöön oli käytössä kaksi virtalähdettä, jotka olivat kiinni kahdessa UPS:ssa. Palvelimet ja kytkimet, joissa on kaksi virtapaikkaa, kiinnitettiin molempiin lähteisiin. Kaksi reititintä, joissa on vain yksi virtapaikka, kiinnitettiin vastakkaisiin virransyöttöihin. Ethernet-yhteydet toteutettiin siten, että palvelimista ja NAS:sta oli yksi linkki molempiin kytkimiin ja kytkimet vuorostaan olivat yhteydessä molempiin reitittimiin yhdellä piuhalla. Näiden kahdennettujen linkkien ryhmittämiseen käytettiin LACP:ta.

Tuloksena systeemi kestää yhden reitittimen ja kytkimen hajoamisen, sekä mahdollisen virransyötön katkeamisen. IPSec VPN -tunneli kestää myös toisen reitittimen hajoamisen CARP:in ansiosta.



Kuva 26. HA verkkotopologia.

7 Loppupäätelmät

Insinöörityöni kartoitti yrityksen palvelintilojen yhdistämistä IPSec VPN -tunnelointitekniikalla sekä tunneleiden turvallisuutta ja vikasietoisuutta.

Työn tavoite oli hankia teoriatietoa site-to-site IPSec VPN -tunneloinnista, sen turvallisuudesta ja systeemin palautumisesta sähkökatkoksen tai muun häiriötilanteen aikana, sekä korvata olemassa oleva PPTP-silta IPSec VPN -tunnelilla ja siirtyä sillatusta verkkorakenteesta staattisesti reititettävään verkkorakenteeseen.

Aloittaessani työn yrityksen yhdistävä sillattu verkkomalli puski äärirajojaan, eikä verkon kasvuun ollut jäljellä tarpeeksi tilaa. PPTP-silta sivukonttorin palvelintilaan oli epäkunnossa ja protokolla itsessään haavoittuva.

Kahdesta työssä käytetystä reitittimestä pfSense oli selkeästi parempi vaihtoehto yrityksen nykyisen ja tulevan verkkotopologian implementointiin. MikroTik:ssa on laajoja skriptausmahdollisuuksia, mutta sen RouterOS:sta puuttuu oleellisia ominaisuuksia varsinkin korkean saatavuuden puolelta. pfSense puolestaan osoittautui helposti hallittaviksi ja sen palomuuri sekä HA-ominaisuudet olivat korkealaatuiset.

Työn tuloksena luotiin sivukonttorin ja pääkonttorin palvelinverkkoja yhdistävä vikasietoinen IPSec VPN -tunneli, jonka tulosta sovellettiin ulkomaille pystytetyissä palvelintiloissa ja sen korkeaa saatavuutta luovassa CARP-ratkaisussa.

Insinöörityötä voisi jatkaa tutkimalla dynaamisia reititysprotokollia, kuten OSPF:ää ja MPLS:ää, ja soveltamalla niitä yrityksen kasvavaan verkkoinfrastruktuuriin.

Lähteet

1 IPsec VPN Overview, Brocade IPsec Site-to-Site Reference Guide http://www.brocade.com/content/html/en/vrouter5600/35r6/vrouter-35r6-ip-secvpn/GUID-8BC898A5-9906-485E-BE44-7C8D38E8EB05.html. Luettu 07.06.2017.

2 IPSec Authentication Header (AH), The TCP/IP Guide, Charles Kozierok, 2005 http://www.tcpipguide.com/free/t_IPSecAuthenticationHeaderAH.htm. Luettu 07.06.2017.

3 IPSec VPN documentation, Netgear http://documentation.netgear.com/reference/enu/vpn/VPNBasics-3-05.html. Luettu 07.06.2017.

4 VPN and VPN Technologies, Cisco article, Andrew Mason, 2002 http://www.ciscopress.com/articles/article.asp?p=24833&seqNum=7. Luettu 10.09.2017.

5 MikroTik, Wikipedia https://en.wikipedia.org/wiki/MikroTik. Luettu 25.11.2016.

6 IPsec Parameter Choice Rationales, Servercentral blog, Joselito Tagarao http://blog.servercentral.com/ipsec-parameter-choice-rationales. Luettu 01.03.2017.

7 Bringing Sanity to Routing Over IPsec, Servercentral blog, Joselito Tagarao http://blog.servercentral.com/bringing-sanity-to-routing-over-ipsec. Luettu 01.03.2017.

8 What's the difference between IKE and ISAKMP, Stackexchange questions, 2013 https://networkengineering.stackexchange.com/questions/1/whats-the-difference-between-ike-and-isakmp. Luettu 05.09.2017.

9 Internet Key Exchange, Wikipedia https://en.wikipedia.org/wiki/Internet_Key_Exchange. Luettu 05.09.2017.

10 Improvements to IPSec Negotiations: Why IKEv2 is King, Andrew Crouthamel, Syyskuu 2016 https://www.linkedin.com/pulse/improvements-ipsec-negotiations-why-ikev2-king-andrew-crouthamel. Luettu 05.09.2017.

11 IKE Protocol, Cisco documentation http://ciscodocuments.blogspot.fi/2011/05/chapter-05-site-to-site-vpns-part03.html. Luettu 05.09.2017.

12 Yahoo Says 1 Billion User Accounts Were Hacked, New York Times, Joulukuu 2016 https://www.nytimes.com/2016/12/14/technology/yahoo-hack.html?mcubz=0. Luettu 10.09.2017.

13 Important Security Information for Yahoo Users, Yahoo tumblr post, Bob Lord, Joulukuu 2016 https://yahoo.tumblr.com/post/154479236569/important-security-information-for-yahoo-users. Luettu 10.09.2017.

14 Yahoo MD5 crack, Stackexchange questions, 2016 https://security.stackexchange.com/questions/145369/how-long-will-it-take-to-crack-the-passwords-stolen-in-the-yahoo-hack-announced. Luettu 10.09.2017.

15 MD5, TechTarget article, Margaret Rose, Toukokuu 2017 http://searchsecurity.techtarget.com/definition/MD5. Luettu 10.09.2017.

16 The SHA1 hash function is now completely unsafe, Computerworld article, Lucian Constantin, Tammikuu 2017 https://www.computerworld.com/article/3173616/security/the-sha1-hash-function-is-now-completely-unsafe.html. Luettu 10.09.2017.

17 RB2011UiAS-RM, Product specifications, Routerboard https://routerboard.com/RB2011UiAS-RM. Luettu 07.01.2017.

18 pfSense SG-8860-1U Product Guide, NetGate Documentation https://www.netgate.com/docs/sg-8860-1u/quick-start-guide.html. Luettu 07.01.2017.

19 How to install pfSense Cluster using CARP, Techencyclopedia, 2016 https://techencyclopedia.wordpress.com/2016/04/12/how-to-install-pfsense-cluster-using-carp/. Luettu 01.03.2017.

20 Manual:IP/Firewall, MikroTik Documentation wiki, 2010. http://wiki.mikrotik.com/wiki/Manual:IP/Firewall/. Luettu 05.12.2016.

21 Manual:IP/IPsec, MikroTik Documentation wiki, 2017. http://wiki.mikrotik.com/wiki/Manual:IP/IPsec. Luettu 05.12.2016.

22 Class Video - Mikrotik VPN, Greg Sowell Consulting, 2010. http://gregsowell.com/?p=1290. Luettu 05.12.2016.

23 MikroTik IPsec tunnel with DDNS and NAT, Occursus Arca blog, Pessoft, 2016 http://blog.pessoft.com/2016/05/29/mikrotik-ipsec-tunnel-with-ddns-and-nat/. Luettu 15.12.2016.

24 IP Cloud DDNS, MikroTik Forum. 2014-2016. http://forum.mikrotik.com/viewtopic.php?f=2&t=85906&sid=34f1be3ed3d712d9ec41ef9215f76cee. Luettu 15.12.2016.

25 DynDNS Update Script, MikroTik forums. 2009-2016. http://forum.mikrotik.com/viewtopic.php?p=171805#p171805. Luettu 15.12.2016.

26 Optimizing IPsec Tunnels, Servercentral blog, Joselito Tagarao http://blog.servercentral.com/optimizing-ipsec-tunnels. Luettu 01.03.2017.

27 NAT traversal for IPSec AH protocol, Stackexchange questions, 2013 https://security.stackexchange.com/questions/46461/nat-traversal-for-ipsec-ah-protocol. Luettu 07.06.2017.

28 IPSec & IKE, Checkpoint Admin Guide, 2016 https://sc1.checkpoint.com/documents/R77/CP_R77_VPN_Admin-Guide/13847.htm. Luettu 11.08.2017.

29 SHA1 & MD5 hash generator, danstools https://www.md5hashgenerator.com/md5-hash-generator/sha1-generator.php. Luettu 05.10.2017.

30 AES, TechTarget article, Margaret Rose, Maaliskuu 2017 http://searchsecurity.techtarget.com/definition/Advanced-Encryption-Standard. Luettu 05.10.2017.

31 Connection Security and IPSec, Windows Server Microsoft, 2009 https://technet.microsoft.com/en-us/library/cc771593(v=ws.10).aspx. Luettu 05.10.2017.

MikroTik skriptit

IPSec policy:n päivittäminen, skriptissä käytetään kommenttia ”IPsecVPN” löytämään muutettavat asetukset. Skripti kirjaa laitteen muistiin tunneliin tehdyt muutokset.

```
/system script add name="ipsec-policy-update" policy=read,write,policy,test source="\
    \n:local peerid    \"IPsecVPN\"\
    \n:local peerhost  \"dnshostname.mine.nu\"\
    \n:local peerip    [:resolve \$peerhost]\
    \n:local policyuid\
    \n:local netwatchuid\
    \n:set policyuid   [/ip ipsec policy find comment=\"\$peerid\" and sa-src-address!=\"\$peerip\"]\
    \n:if (\$policyuid != \"\") do={\
    \n  /ip ipsec policy set \$policyuid sa-src-address=\"\$peerip\"\
    \n  :log info \"Script ipsec-peer-update updated policy '\$peerid' with address '\$peerip'\"\
    \n  /tool netwatch set \$netwatchuid host=\"\$peerip\"\
    \n  :log info \"Script ipsec-peer-update updated netwatch '\$peerid' with address '\$peerip'\"\
    \n}"
```

DynDNS skripti yhdistää laitteen DynDNS-verkkopalveluun, johon se päivittää DDNS-nimeen uuden IP-osoitteen muutostilanteessa. Skripti kirjaa reitittimen muistiin IP-osoitteeseen tehdyt toimenpiteet.

```
:global ddnsuser "user"
:global ddnspass "password"
:global theinterface "ether1"
:global ddnshost dnshostname.mine.nu
:global ipddns [:resolve $ddnshost];
:global ipfresh [ /ip address get [/ip address find interface=$theinterface ] address ]
:if ([ :typeof $ipfresh ] = nil ) do={
   :log info ("DynDNS: No ip address on $theinterface .")
} else={
   :for i from=( [:len $ipfresh] - 1) to=0 do={
      :if ( [:pick $ipfresh $i] = "/") do={
    :set ipfresh [:pick $ipfresh 0 $i];
      }
}

:if ($ipddns != $ipfresh) do={
    :log info ("DynDNS: IP-DynDNS = $ipddns")
    :log info ("DynDNS: IP-Fresh = $ipfresh")
   :log info "DynDNS: Update IP needed, Sending UPDATE...!"
```

```
   :global str "/nic/update\?hostname=$ddnshost&myip=$ipfresh&wild-
card=NOCHG&mx=NOCHG&backmx=NOCHG"
   /tool fetch address=members.dyndns.org src-path=$str mode=http user=$ddnsuser \
        password=$ddnspass dst-path=("/DynDNS.".$ddnshost)
    :delay 1
    :global str [/file find name="DynDNS.$ddnshost"];
    /file remove $str
    :global ipddns $ipfresh
  :log info "DynDNS: IP updated to $ipfresh!"
    } else={
     :log info "DynDNS: dont need changes";
    }
}
```

Toinen DynDNS-skripti pakottaa IP-osoitteen päivittämisen.

```
:global dyndnsForce true
/system script run DynDNS
```