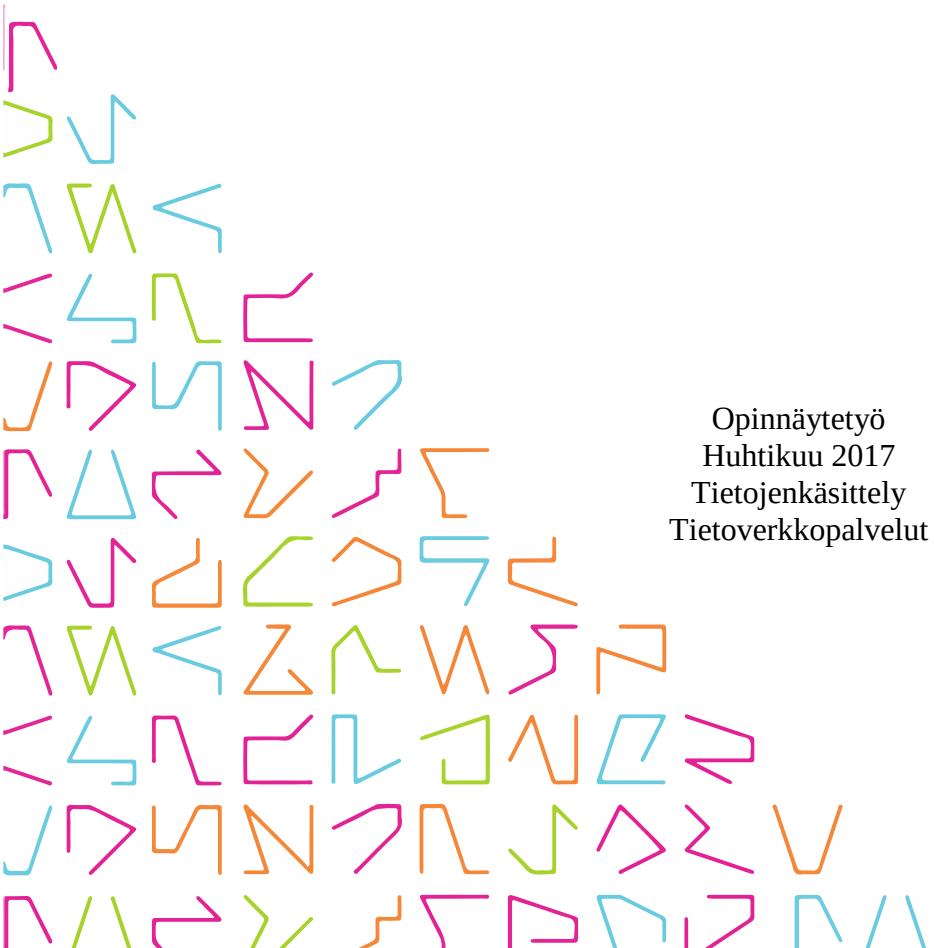
TAMPEREEN
AMMATTIKORKEAKOULU

LABORATORIOVERKON TIETOTURVAN KEHITTÄMINEN

Milko Viirto



Opinnäytetyö
Huhtikuu 2017
Tietojenkäsittely
Tietoverkkopalvelut

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tietojenkäsittely
Tietoverkkopalvelut

VIIRTO, MILKO:
Laboratorioverkon tietoturvan kehittäminen

Opinnäytetyö 63 sivua, joista liitteitä 31 sivua
Huhtikuu 2017

Tämän opinnäytetyön tavoitteena oli kehittää Nokia Networksin laboratorioverkon tietoturvaa kryptologisia salausmenetelmiä hyödyntäen, hallita kytkimien ja reitittimien käyttäjien oikeuksia tunnistuspalveluilla sekä monitoroida verkon laitteiden mahdollisia asetusmuutoksia keskitetyltä lokipalvelimelta. Työn tarkoituksena oli lisätä erilaisia salausmenetelmiä jo käytössä oleviin sekä mahdollisesti tulevaisuudessa käyttöönotettaviin reititys- ja hallintaprotokolliin. Työn yhteydessä laadittiin käyttöohje, jonka tarkoituksena on helpottaa salausprotokollien ja hyvien tietoturvallisten menetelmien käyttöönottoa.

Salausprotokollien ja muiden tietoturvaan liittyvien toimenpiteiden käyttöönotto toteutettiin testaukseen tarkoitetussa verkon osassa yhteen jokaisen käytössä olevan laitevalmistajan laitteeseen. Työssä selvitettiin lyhyesti kryptologian syntyä, yhteiskunnallista merkitystä nykypäivänä sekä pohdittiin kvanttitietokoneiden tuomia mahdollisuuksia.

Tietoturvan rooli on erittäin merkittävä modernissa verkkoympäristössä, joten sen suunnitteluun sekä toteuttamiseen tulee käyttää riittävästi aikaa. Mahdollisista asetusvirheistä aiheutuvaa palveluiden ja verkon häiriöaikaa pystytään lyhentämään, kun verkkoympäristön tietoturva, reaaliaikainen laitteiden ja käyttäjien valvonta sekä hallinta ovat nykyaikaisten vaatimusten mukaisia. Jo prosessin aloitusvaiheessa oli selvää, että tässä työssä asennettavia palveluita ja menetelmiä tullaan hyödyntämään kaikissa laboratorioverkon laitteissa lähitulevaisuudessa.

Avainsanoja: tietoturva, salausmenetelmät, kryptologia, verkkoympäristö

ABSTRACT

Tampere University of Applied Sciences
Degree Programme in Business Information Systems
Network Services

VIIRTO, MILKO:
Enhancing Security in Laboratory Network

Bachelor's thesis 63 pages, appendices 31 pages
April 2017

The objective of this thesis was to enhance Nokia Networks laboratory network environments security by exploiting cryptographic procedures, control user access in switches and routers with user management service and to monitor configuration changes in different network appliances with user credentials that are being sent to a centralized log server. The purpose of this study was to apply different kinds of cryptographic means to routing- and management protocols already in use, and test them to the ones possibly applied in the future. During the work an advisory documentation was created, which purpose is to ease the deployment of cryptographic protocols and security related practices.

The cryptographic protocols and other security related operations were deployed in one of each product from every manufacturer in use in a network environment meant solely for testing purposes. The study briefly examines the birth and meaning of cryptography in modern society and takes a glimpse in the future by scrutinizing the possibilities of quantum cryptography. The study also goes through the most common cryptographic protocols and security appliances in use.

The role of security in modern network environments is highly significant and the time used in designing and deploying it must be sufficient. The downtime caused by possible misconfigurations, can be minimized when the environments security, real time monitoring of appliances and user access control meets the modern requirements. In the beginning of this study it was already clear that the protocols and practices in this work will be applied in every appliance in the network in the near future.

Key words: information security, cryptographic protocols, cryptology, network environment

SISÄLLYS

TERMISTÖ

AAA	Authentication, Authorization and Accounting
ARPANET	The Advanced Research Projects Agency Network
CentOS	Community Enterprise Operating System
CLI	Command Line Interface
DARPA	Defense Advanced Research Projects Agency
DH	Diffie-Hellman
EAP	Extensible Authentication Protocol
FIPS	Federal Information Processing Standards
FreeBSD	Free Berkeley Software Distribution
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IETF	Internet Engineering Task Force
JunOS	Juniper Network Operating System
L3	Layer 3
MD5	Message Digest 5
MIT	Massachusetts Institute of Technology
Modulo, Modulus	Jakojäännös
NASA	National Aeronautics and Space Administration
NMAP	Network Mapper
PAM	A Pluggable Authentication Module
RADIUS	Remote Authentication Dial-In User Service
RHEL	Red Hat Enterprise Linux
Rlogin	Remote login
RSA	Rivest, Shamir & Adleman
SHA	Secure Hash Algorithm
SSH	Secure Shell
TACACS	Terminal Access Controller Access Control System
TCP	Transmission Control Protocol
Telnet	Teletype network
UDP	User Datagram Protocol
USRA	Universities Space Research Association

1 JOHDANTO

Modernissa verkkoympäristössä tietoturvan merkitys on alati kasvavassa roolissa. Hyvin suunnitelluilla ja toteutetuilla tietoturvaan liittyvillä toimenpiteillä on mahdollista säästää järjestelmän ylläpitäjien sekä myös yrityksen aikaa ja resursseja.

Työn toimeksiantajana toimi Nokia Oyj:n liiketoimintayksikkö Networks. Opinnäytetyön tarkoitus oli parantaa liiketoimintayksikön HetRAN-cloud-tiimin hallinnoimaa ja ylläpitämää laboratorioverkkoa, jossa uusien laitteiden ja ohjelmistojen testitoimenpiteet tehdään. Työn tavoite oli mahdollistaa Nokia Networksin käyttäjien ja laitteiden hallinta sekä estää uusien laitteiden lisäys käytössä olevien hallinta- ja reititysprotokollien piiriin.

Opinnäytetyössä käydään lyhyesti läpi työhön valittuja, salaukseen tarkoitettuja algoritmeja ja protokollia sekä muita tietoturvaan liittyviä toimenpiteitä ja palveluita. Salausmenetelmien testausta ja käyttöönottoa varten luotiin erillinen testikäyttöön tarkoitettu verkon osa, jossa oli käytössä yksi jokaisen laboratorioverkossa käytössä olevan laitevalmistajan laite.

Koska kryptologia liittyy olennaisesti tietoturvaan, työ alkaa kryptologian historiakatsauksella ja sen merkityksen hahmottamisella nykypäivänä ja tulevaisuudessa. Tämän jälkeen esitellään käytännön osuuteen valittuja laitteita ja palveluita. Luvut 5 ja 6 koostuvat algoritmien, protokollien sekä palveluiden asennukseen liittyvistä toimenpiteistä luvussa 4 määritellyille laitteille.

Lukijan pohjatietovaatimuksina pidetään TCP/IP-protokollan, yleisimpien reititys- ja hallintaprotokollien, yleisimpien laitevalmistajien laitteiden sekä linux-käyttöjärjestelmän tuntemusta. Opinnäytetyöhön ei kuulu reititys- ja hallintaprotokollien asennusta eikä tietoturva-käsitteen avausta, vaan se keskittyy puhtaasti tietoturvallisiin menetelmiin ja kryptologisiin käsitteisiin. Myöskään lokipalvelimen asennus ei kuulu tähän opinnäytetyöhön.

Liite 1 toimii jatkossa käyttöohjeena, jota tullaan kehittämään edelleen. Sen avulla erilaisten tietoturvakäytäntöjen lisäys jo olemassa oleviin laiteasetuksiin helpottuu.

2 NOKIA NETWORKS

Nokia Oyj on suomalainen maailmanlaajuisesti toimiva tietoliikennealan yhtiö. Se perustettiin vuonna 1865, jolloin Suomeen syntyi kumiteollisuus ja myöhemmin nykyaikainen puu- ja kaapeliteollisuus. Nokia Networks on vuonna 2007 perustettu Nokian liiketoimintayksikkö, alkuperäiseltä nimeltään Nokia Siemens Networks, jonka toimintaan kuuluu tietoliikenneverkoissa käytettävien laitteiden ja ohjelmistojen suunnittelu ja valmistus. Se työllistää 150 maassa noin 55 000 henkilöä, joista Suomessa on 7 000. Liiketoimintayksikön liikevaihto oli vuonna 2014 noin 12 miljardia euroa, ja se on näin ollen tuotantoalueellaan maailman toiseksi tai kolmanneksi suurin valmistaja. Liiketoimintayksikön pääkonttori sijaitsee Espoon Karaportissa. (Nokia Oyj 2016)

Opinnäytetyön taustalla on Nokia Networksin tarve kehittää laboratorioverkkonsa tietoturvaa sekä tämän avulla valvoa ja hallita verkossa työskentelevien henkilöiden verkon laitteiden käyttöoikeuksia. Koska uusimmat kehitteillä olevat tuotteet ovat kehitysvaiheessa ja yhtiön toiminta on maailmanlaajuista, on laboratorioverkon tietoturva, eheys ja muuttumattomuus tuotteiden testiajanjaksojen aikana mahdollisten tietovuotojen vuoksi tärkeää.

Tietoturvaominaisuuksien kehittämisen myötä laboratorioverkossa tehtyjen testien tulokset ovat entistä luotettavampia. Samalla mahdollisten asetusvirheiden ja verkkohyökkäysten riski pienenee huomattavasti. Työ tehtiin erillisessä testiympäristössä. Testiympäristö sisälsi yhden kappaleen Dellin, Juniperin ja Quantan L3 (Layer 3)-kytkintä sekä Nokian reitittimen, joihin työn vaatimat asetusmuutokset tehtiin laboratorioverkossa käytössä olevien laitteiden ja niiden protokollatukien perusteella. Testiverkkoon asennettiin myös AAA-palveluita (Authentication, Authorization & Accounting), joiden tarkoituksena on mahdollistaa käyttäjien tunnistaminen, käyttöoikeuksien hallinta ja sitä kautta laitteiden käytönvalvonta. Työstä syntyvän käyttöohjeen ansiosta lisättävien verkkolaitteiden suojaus ja valvonta voidaan tulevaisuudessa hoitaa entistä nopeammin.

3 KRYPTOLOGIA

Kryptologia tai kryptografia, vapaasti käännettynä salaoppi tai salakirjoitus. Termit ovat kreikkaa ja ne muodostuvat kahdesta sanasta: *kryptós* (piilossa, salainen) ja *logia tai grafia* (oppi, kirjoitus). (Singh 1999: 20–34.)

3.1 Kryptologian synty

Kryptologiasta puhuttaessa on syytä mainita sen niin sanottu esiaste, steganografia. Steganografia on kreikkaa, vapaasti käännettynä piilokirjoitus ja se muodostuu kahdesta sanasta: *steganos* (piilottaa, kätkeä) ja *graphein* (kirjoittaa). Steganografia syntyi sodan ja politiikan ansiosta ja sen tarkoituksena oli mahdollistaa turvallinen kommunikointi yhtäältä liittolaisten ja toisaalta diplomaattien välillä. Steganografisia menetelmiä ovat muun muassa viestien kätkeminen maalausten kääntöpuolelle ja näkymättömän musteen käyttö, joka oikein käsiteltynä saatiin taas näkyväksi. Myös päänahan tatuointi oli tavallista. Hiusten takaisin kasvaessa viesti oli luonnollisesti piilossa. (Singh 1999: 20-34.)

Moderneihin steganografisiin menetelmiin kuuluvat muun muassa mikrofilmit sekä viestien piilottaminen ääni- ja kuvatiedostoihin. Piilotusmenetelmien paljastuessa steganografia alkoi kuitenkin jo hyvin varhaisessa vaiheessa olla turvatonta, joten kommunikaatiolle tarvittiin uusia, luotettavampia keinoja. Tämän johdosta syntyi kryptologia. (Kahn 1973.)

Kryptologiaa hyödynnettiin yleensä yhdessä steganografian kanssa, mutta paljastuneiden piilotusmenetelmien vuoksi viestien piilotus ei enää ollut itseisarvo. Salakirjoituksen perimmäinen tarkoitus on turvata viestin sisältö kolmansilta osapuolilta salaamalla viesti osapuolten ennalta sopimaa menetelmää hyväksikäyttäen. Vihollisen saadessa käsiinsä liittolaisten välisiä viestejä, on viestin sisällön tulkitseminen ilman salauksen purkuun tarvittavaa avainta ja menetelmän tuntemusta erittäin vaikeaa. (Kahn 1973.)

Tämä synnytti kryptologian rinnalle myös kryptoanalyysin, jonka tarkoituksena on kehittää erilaisia menetelmiä viestin salaukseen käytetyn menetelmän ja mahdollisen salausavaimen selvittämiseksi. Kryptologia on syntymästään lähtien ollut eräänlaista kilpajuoksua kryptologisten menetelmien kehittäjien (kryptograafikoiden) ja kryptologisten menetelmien murtajien (kryptoanalyytikoiden) välillä, jonka ansiosta salaus- ja murtamismenetelmät ovat kehittyneet niiden nykymuotoonsa.

3.2 Kryptologia nykypäivänä

Nykyisin erilaisia salausmenetelmiä käytetään säännöllisesti ja ne ovat vakiinnuttaneet asemaansa HTTP-liikenteessä (Hypertext Transfer Protocol), muodostaen uuden protokollan, HTTPS:n (Hypertext Transfer Protocol Secure), sekä erilaisissa sähköpostiliikenteeseen liittyvissä protokollissa.

Koska tavalliset internetin käyttäjät ovat entistä valveutuneempia ja alkavat ymmärtää yksityisten ja turvallisten viestintämuotojen tärkeyttä, myös suosituimmissa pikaviestisovelluksissa on otettu käyttöön erilaisia salauskeinoja niiden omiin palveluihin. Tietokoneiden laskentatehon kasvaessa paine kehittää uusia, vaikeammin murrettavia algoritmeja kasvaa. Kryptograafikoiden ja kryptoanalyytikoiden välinen kilpajuoksu jatkuu edelleen.

Tässä luvussa esitellään testikäyttöön valittuja algoritmeja ja protokollia. Algoritmien valinta tehtiin laitetukien perusteella. Modernit salausmenetelmät ovat luonteeltaan monimutkaisia ja vaativat paljon matemaattisten funktioiden yksityiskohtaista läpikäymistä. Selkeyden vuoksi yksinkertaistin jokaisen algoritmin selitystä ja jätin niiden yksityiskohtaisemmat tarkastelut tämän työn ulkopuolelle.

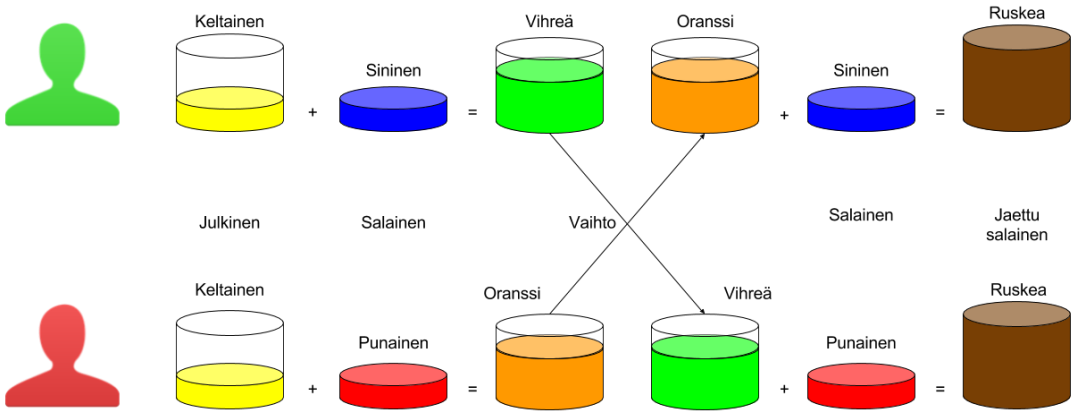
Lisää algoritmien yksityiskohtaisemmista matemaattisista kaavoista ja funktioista voi lukea esimerkiksi Bruce Schneierin kriitikoiden ylistämästä opuksesta Applied Cryptography – Protocols, Algorithms and Source Code in C.

3.2.1 Julkinen avain

Diffie-Hellman on avaintenvaihtomenetelmä (Key exchange), jonka algoritmi julkaistiin vuonna 1976. Tavoitteena oli kehittää turvallinen ja edullinen keino jo pitkään kryptologiaa vaivanneeseen ongelmaan, salausavaimien vaihtamiseen. Algoritmia edeltävän julkisen avaimen vaihtomenetelmän idean kehittämisessä oli mukana Whitfield Diffien ja Martin Hellmanin lisäksi myös Ralf Merkle. (Singh 1999: 339–365.)

Ideana oli luoda kaksi avainta; julkinen (Public key) ja salainen avain (Private key) jotka ovat yhteydessä toisiinsa, mutta toisen avaimen johtaminen toisesta olisi mahdotonta. Tarkoituksena oli käyttää näitä kahta avainta ja jakaa keskustelijoiden välille kolmas avain, jonka avulla turvallinen kommunikaatio olisi mahdollista.

Avaintenvaihtomenetelmä perustui matemaattiseen funktioon jota kukaan kehittäjistä ei onnistunut keksimään. Menetelmä jäi teoriatasolle, eivätkä he koskaan saaneet avaintenvaihtoa toimimaan käytännössä. (Singh 1999: 339–365.) Kuvassa 1 avaintenvaihdon idea on esitetty väreillä.



Kuva 1, DH-avaintenvaihdon perusidea

RSA (Rivest, Shamir & Adleman) on julkisen avaimen salausalgoritmi, joka käyttää hyödykseen DH avaintenvaihtomenetelmän ideaa. Ron Rivest, Adi Shamir ja Leonard Adleman löysivät vuonna 1977 matemaattisen funktion Diffien, Hellmanin ja Merkel kaavailemalle menetelmälle. (Singh 1999: 365–376.)

Tämä matemaattinen funktio hyödyntää suuria alkulukuja ja se käyttää tavanomaisesti avaiminaan 100 - 200 luvun numerosarjoja. RSA toimii luomalla molemmille viestinnän osapuolille omat julkiset sekä yksityiset avaimet ja vaikka se onkin huomattavasti hitaampi kuin monet aikaisemmat algoritmit, sen tuomat hyödyt turvallisuuden suhteen ovat merkittäviä. MIT (Massachusetts Institute of Technology) patentoi algoritmin Yhdysvalloissa vuonna 1983 ja sen patentti raukesi vasta vuonna 2000. (Schneier 1996: 466–474.)

Seuraavassa esimerkissä kuvataan RSA:n käyttämää matemaattista funktiota, joka hyödyntää muutamia käsitteitä jotka vaativat selvennystä. Näistä ensimmäinen on suhteellinen alkuluku, jolla viitataan alkulukuihin. Alkuluvut ovat keskenään jaottomia. Toisena käsitteenä Eukleideen algoritmi, jonka tarkoituksena on löytää kahden kokonaisluvun suurin yhteinen tekijä. Kolmas käsite löytyy matemaattisista kaavoista ja on lyhennetty muotoon mod, jolla viitataan sanaan *modulo* tai *modulus*, joka tarkoittaa jakojäännöstä. RSA toimii seuraavan yksinkertaistetun esimerkin kuvaamalla tavalla.

Luodaksemme kaksi avainta valitsemme satunnaisesti kaksi alkulukua, ***p*** ja ***q*** jotka yhdessä muodostavat luvun ***n***.

$$p = 47$$
$$q = 71$$
$$n = pq = 3337$$

Seuraavaksi valitsemme satunnaisesti lukua 1 suuremman ja lukua ***n*** pienemmän salausavaimen ***e*** jolle ***(p-1)(q-1)*** on suhteellinen alkuluku.

$$(p-1)(q-1) = 46 * 70 = 3220$$
$$e = 79$$

Viimeiseksi käytämme Eukleideen algoritmia laskeaksemme salauksen purkuun tarkoitetun avaimen ***d***, jonka jälkeen tuhoamme luvut ***p*** ja ***q***.

$$d = 79^{-1} \bmod 3220 = 1019$$

Julkinen salausavain muodostuu luvuista ***e*** ja ***n***, yksityinen avain luvusta ***d***.

Salataksemme viestin ***m*** jaamme sen pienempiin lohkoihin, jotka salaamme erikseen käyttämällä lukua ***d*** ja ***n***. Nämä salatut lohkot kuvataan luvulla ***c***.

$m = 6882326879666683$

$m_1 = 688$

$m_2 = 232$

$m_3 = 687$

$m_4 = 966$

$m_5 = 668$

$m_6 = 003$

$c_1 = 688^{79} \bmod 3337 = 1570$

$c_2 = 232^{79} \bmod 3337 = 2756$

$c_3 = 687^{79} \bmod 3337 = 2091$

$c_4 = 966^{79} \bmod 3337 = 2276$

$c_5 = 668^{79} \bmod 3337 = 2423$

$c_6 = 003^{79} \bmod 3337 = 158$

$c = 1570\ 2756\ 2091\ 2276\ 2423\ 158$

Salatun viestin purkaminen tapahtuu käyttämällä yksityistä avainta samalla periaatteella.

$m_1 = 1570^{1019} \bmod 3337 = 688$

$m_2 = 2756^{1019} \bmod 3337 = 232$

$m_3 = 2091^{1019} \bmod 3337 = 687$

$m_4 = 2276^{1019} \bmod 3337 = 966$

$m_5 = 2423^{1019} \bmod 3337 = 668$

$m_6 = 158^{1019} \bmod 3337 = 3$

$m = 6882326879666683$

3.2.2 Tiivistealgoritmit

Tiivistealgoritmien tarkoituksena on luoda datasta uniikki merkkijono, josta alkuperäistä dataa on mahdotonta palauttaa. Kaksi yleisimmin käytössä olevaa tiivistealgoritmia ovat MD5 (Message Digest 5) ja SHA-2 (Secure Hash Algorithm 2).

MD5 on tiedon eheyden tarkistamiseen tarkoitettu yksisuuntainen tiivistealgoritmi, joka hyödyntää Ralph Merklen ja Ivan Damgårdin vuonna 1979 kehittämää Merkle-Damgård viestintiivistysmenetelmää. Algoritmin kehitti RSA-algoritminkin parissa työskennellyt Ronald Rivest vuonna 1995 jatkoksi aikaisemmin kehittämälleen MD4-algoritmille. Algoritmin tarkoituksena on luoda datasta 128-bittinen tiiviste, joka tyypillisesti esitetään käyttäjälle 32-merkkisenä heksakoodina. (RFC 1321.)

Algoritmi prosessoi tiivistettävän viestin tai datan 512-bittisiksi lohkoiksi, jotka jaetaan 16 lohkoon. Lopuksi algoritmi tuottaa viestistä neljä 32-bitin lohkoa, jotka yhdistetään yhdeksi 128-bittiseksi tiivistearvoksi. (Schneier 1996: 436–441.)

MD5-algoritmia ei enää pidetä turvallisena vaihtoehtona esimerkiksi salasanojen salaamiseen mahdollisten tiivisteiden päällekkäisyyksien vuoksi. Tämä teoreettinen mahdollisuus perustuu vanhaan matemaattiseen arvoitukseen, niin kutsuttuun syntymäpäiväongelmaan.

Syntymäpäiväongelman tarkoituksena on selvittää seuraava asia: kuinka monta henkilöä pitää olla samassa tilaisuudessa, että todennäköisyys sille, että kahdella eri henkilöllä on sama syntymäpäivä, on yli 50%. Vastaus tähän on yllättävän pieni, 23 henkilöä. Tämä todennäköisyys on helposti todettavissa seuraavan laskukaavan avulla.

$$\mathbf{1 - \frac{365 \times 364 \times 363 \times \ldots \times 343}{365^{23}} \approx 0,5073 > 0,5}$$

Teoreettisesta ongelmastaan huolimatta se on vielä laajasti käytössä pääasiassa tiedostojen, esimerkiksi linux-jakeluiden levykuvien eheyden tarkistamisessa ja salasanojen säilyttämisessä.

MD5-algoritmin tarkemmasta toiminnasta voi lukea lisää IETF:n (Internet Engineering Task Force) standardista 1312 tai Bruce Schneierin kirjasta Applied Cryptgraphy – Protocols, Algorithms and Source Code in C. Linkki standardiin sekä lisätietoja kyseisestä kirjasta löytyy lähteet-osiosta.

Koska MD5-algoritmi on turvaton, NSA (National Security Agency) kehitti vuonna 1995 sen tilalle uuden tiivistealgoritmin. SHA-1 tuottaa datasta 160-bittisen tiivisteen, joka tyypillisesti esitetään käyttäjälle 40-merkkisenä heksakoodina. Se käyttää tiivistämiseen samaa viestintiivistämismenetelmää kuin MD5, erona käsiteltävien lohkojen ja toimenpiteiden suurempi määrä. (FIPS 180-4.)

Vuonna 2005 kryptoanalyytikot törmäsivät kuitenkin samaan ongelmaan kuin MD5:n kanssa ja vaikka yhtäkään päällekkäistä tiivistettä ei olla havaittu käytännössä, sen käyttö pyritään lopettamaan vuoteen 2017 mennessä.

SHA-algoritmin tarkemmasta toiminnasta voi lukea lisää Yhdysvaltain hallituksen julkaisemasta standardista FIPS (Federal Information Processing Standards) 180-4 tai Bruce Schneierin kirjasta Applied Cryptgraphy – Protocols, Algorithms and Source Code in C. Linkki standardiin sekä lisätietoja kyseisestä kirjasta löytyy lähteet-osiosta.

3.2.3 SSH

SSH (Secure Shell) on suomalaisen tekniikan lisensiaatin, Tatu Ylösen vuonna 1995 kehittämä salattuihin etäyhteyksiin tarkoitettu protokolla, joka käyttää hyväkseen RSA-avaintenvaihtoa. Sen tarkoituksena oli korvata heikompien suojaustason omaavia etäyhteys-protokollia, kuten Rlogin (Remote login) ja Telnet (Teletype Network). (RFC 4251.)

Windows-ympäristössä SSH:n käyttö tapahtuu erillisen asiakasohjelman avulla. Kuvassa 2 on suosituin SSH-yhteyksiin soveltuva windows-asiakasohjelma, PuTTY. Ohjelma toimii syöttämällä IP-osoitteen sille määriteltyyn kenttään. Yhteyteen käytettävä portti valitaan protokollan mukaan automaattisesti käyttämään protokollan oletusporttia, joka on SSH-protokollan tapauksessa 22.

Linux- ja OSX-käyttöjärjestelmissä asiakasohjelma on yleensä valmiiksi asennettuna. Protokollaa käytetään muun muassa reitittimien, kytkimien ja palvelimien etäkäytössä.

Kuva 2, PuTTY

PuTTYsta poiketen komentoriviltä käytetty SSH-yhteys vaatii käyttäjätunnuksen määrittelyn jo itse komennossa. Protokollan käyttö toimii komentoriviltä kuvan 3 osoittamalla tavalla. Yhteyteen käytettävä porttinumero on mahdollista määritellä manuaalisesti käyttämällä -p optiota. Mikäli porttia ei määritellä, komennon oletusporttina toimii protokollan oletusportti, 22.

```
[root@centoselk ~]# ssh root@10.10.10.10
```

Kuva 3, SSH:n käyttö komentoriviltä

3.3 Kryptologia tulevaisuudessa

Tässä luvussa käsitellään lyhyesti kahta, vielä teoreettisella tasolla olevaa, kvanttikryptologian kuuluisinta algoritmia. Vaikka kvanttitietokone ei vielä ole osa joka päiväistä elämäämme, lukuisia sen toimintaan perustuvia algoritmeja on jo kehitetty. Ennen kvanttikryptologiaan perehtymistä tulee kuitenkin ymmärtää muutama kvanttifysiikan tuoma erikoispiirre, joita käsitellään lyhyesti ja mahdollisimman selkeästi.

Tanskalaisen fyysikon Niels Henrik Bohrin mukaan ihminen, joka ei hämmästy kvanttiteoriasta puhuttaessa, ei ole sitä ymmärtänyt (Barad 2007: 254). Kvanttifysiikassa partikkeleita tarkastellessa tutkijan tarkastelu vaikuttaa tutkimustuloksiin, joten ne ovat siis niin sanotussa superpositiossa. Werner Heisenbergin vuonna 1927 esittämän Epätarkkuusperiaatteen perusideana on seuraava: jos tarkastelemme partikkeleiden sijaintia, emme voi tarkastella partikkelin liikettä ja päinvastoin. (Schneier 1996: 554–557.)

Kvanttitietokoneiden käsitteen esitteli ensimmäisenä englantilainen fyysikko David Deutch. Vuonna 1985 julkaistussa tutkielmassaan hän kuvaili näkemystään kvanttitietokoneesta ja selvensi sen eroja perinteiseen tietokoneeseen nähden. (Singh 1999: 425–467.)

Käsite kvanttitietokone saattaa ensikuulemalta vaikuttaa melko korkealentoiselta ja kaukaiselta, mutta kun sellaiset jättiyritykset ja -yhdistykset kuin NASA (National Aeronautics and Space Administration), Google ja USRA (Universities Space Research Association) tutkivat ja kehittävät kvanttitietokoneita, voimme olettaa asian olevan päivä päivältä ajankohtaisempaa. Kvanttitietokoneiden tarkoituksena ei ole korvata perinteistä tietokonetta, vaan nopeuttaa optimointiin liittyvien algoritmien suorittamista, joita esimerkiksi tekoälyn kehittämiseen tarvitaan. (D-wave Systems 2017.)

Kvanttikryptologia nojautuu vahvasti kvanttitietokoneisiin, joiden mahdollistama laskentateho ja toimintaperiaate voisi tuhota kaikkien nykyaikaisten salakirjoitusten turvallisuuden. Käsitteen kvanttikryptologiasta esitteli ensimmäisenä 1970-luvun alkupuolella yhdysvaltalainen Columbia Universityn opiskelija Stephen Wiesner. Hänen tutkielmansa aiheesta, joka julkaistiin vuonna 1983, käyttää hyväkseen epätarkkuusperiaatetta.

Wiesnerin tutkimus liittyi niin kutsuttujen kvanttisetelien kehittämiseen, joiden uudenlainen aitouden tunnistaminen perustui valon fotonien värähtelysuuntaan, polarisaatioon. Seteli sisälsi fotonivirran, jonka tarkistamiseen pitäisi tietää jokaisen fotonin polarisaatio ja tarkastella virtaa niin kutsuttujen suodattimien läpi. (Singh 1999: 425–467.)

Wiesnerin tutkimusta ei kuitenkaan voitu soveltaa käytännössä sen aikaisilla laitteilla järkevään hintaan, eikä hänen tutkielmaansa otettu vakavasti tai julkaistu yhdessäkään tieteellisessä aikakausilehdessä. Stephen Wiesnerin kehittämää periaatetta kuitenkin hyödynnettiin myöhemmin tunnetuimmassa ja kehitetyimmässä kvanttisalakirjoitusmenetelmässä, kvanttiavaintenvaihdossa (Quantum key distribution). Käsitteen kvanttiavaintenvaihdosta loivat yhdysvaltalaiset Charles Bennett ja Gilles Brassard. (Singh 1999: 425–467.)

Avaintenvaihdon perusideana on lähettää toiselle osapuolelle salausavain samankaltaisena fotonivirtana. Vastaanottaja asettaa satunnaisesti polarisoiduille vastaanottamilleen fotoneille satunnaisessa asennossa olevia suodattimia, jonka jälkeen osapuolet varmistavat jaetun avaimen eheyden erilaisilla menetelmillä. Kvanttifysiikan luonteen vuoksi kyseistä virtaa ei voida kaapata muuttamatta lähetettyjen fotonien polarisaatiota. (Singh 1999: 425–467.)

Kvanttiavaintenvaihtoprotokollan tarkemmasta toiminnasta voi lukea lisää Bruce Schneierin kirjasta Applied Cryptgraphy – Protocols, Algorithms and Source Code in C tai Simon Singhin kirjasta Koodikirja – Salakirjoituksen historia muinaisesta Egyptistä kvanttikryptografiaan. Lisätietoja kyseisistä kirjoista löytyy lähteet-osiosta.

Puhuttaessa nykyisten salausmenetelmien murtumisesta kvanttitietokoneiden avulla on vielä lyhyesti mainittava erään matematiikan professorin kehittämä algoritmi. Shorin algoritmi on yhdysvaltalaisen Peter Shorin vuonna 1994 kehittämä, perinteisten salausmenetelmien murtamiseen tarkoitettu algoritmi. Algoritmin avulla olisi mahdollista simuloida kaikkia mahdollisia tapoja salausavaimen selvittämiseksi. Kvanttitietokoneen tuomien ominaisuuksien mukaisesti, samanaikaisesti. (Monz, Nigg, Martinez, Brandl, Schindler, Rines, Wang, Chuang & Blatt 2015.)

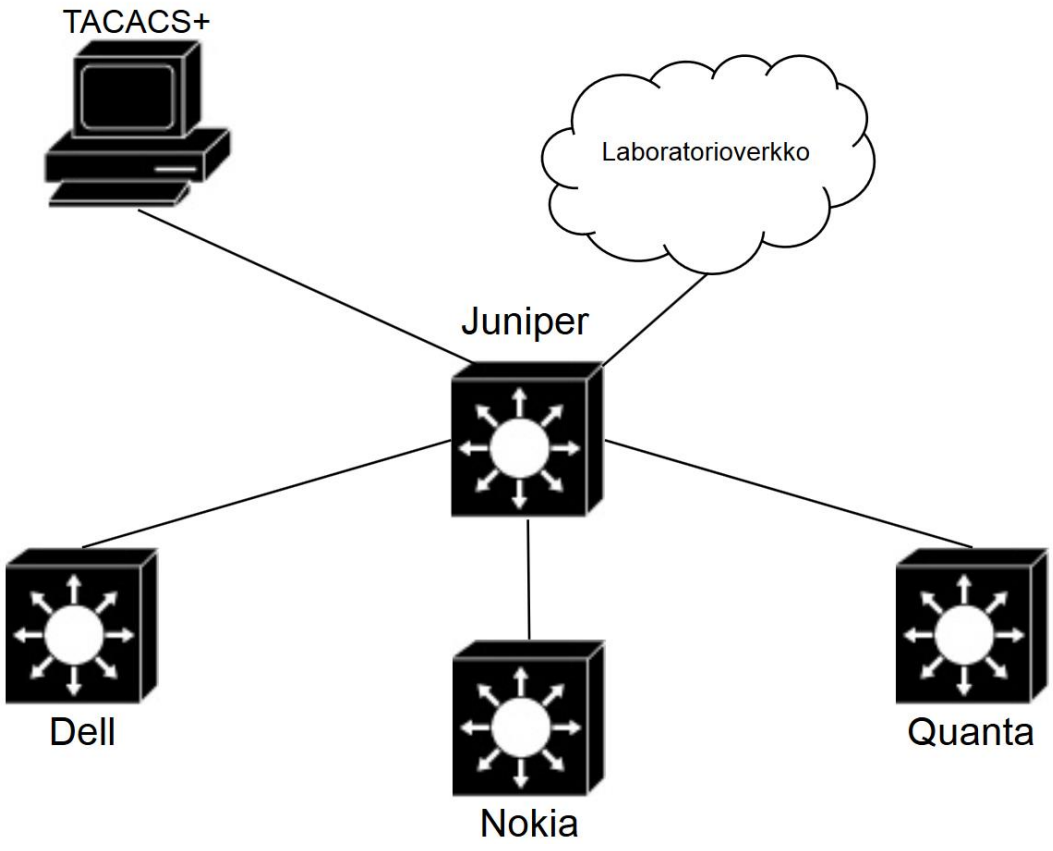
Shorin algoritmin yksityiskohtaisemmasta toiminnasta voi lukea lisää Thomas Monzin, Daniel Niggin, Esteban A. Martinezin, Matthias F. Brandln, Philipp Schindlerin, Richard Rinesin, Shannon X. Wangin, Isaac L. Chuangin ja Rainer Blattin tekemästä tutkimuksesta Realization of a scalable Shor Algorithm. Linkki kyseiseen tutkimukseen löytyy lähteet-osiosta.

4 TESTIYMPÄRISTÖ

Ympäristön tarkoituksena oli mahdollistaa asetusmuutosten turvallinen testaus laboratorioverkossa esiintyville protokollille sekä mahdollistaa käyttäjätunnistamiseen tarkoitettujen palveluiden toimivuutta. Tässä luvussa käydään läpi testikäyttöön valittuja laitteita ja palveluita. Valintakriteerinä käytettiin käytössä olevien laitteiden ja protokollien yleisyyttä sekä niiden yhteensopivuutta salausmenetelmien ja palveluiden kanssa.

4.1 Ympäristön esittely

Topologiaa luotaessa pidettiin testiympäristön yksinkertaisuutta suuressa arvossa. Testiympäristön keskeiseksi laitteeksi valittiin Juniperin valmistama QFX5100-48 L3-kytkin. Syy valintaan oli laitteen erinomainen protokollatuki, joka mahdollisti jokaisen testiverkossa käytetyn ja suojattavan hallinta- ja reititysprotokollan asetusten määrittelyn kyseiselle laitteelle. Kuvassa 4 työtä varten pystytetyn testiverkon topologia.



Kuva 4, Testiverkon topologia

Testikäyttöön valitut palvelut asennettiin yhdelle virtuaalikoneelle OpenStack-järjestelmään. OpenStack-järjestelmä ei ole tämän opinnäytetyön kannalta olennainen, joten sen toimintaa tai rakennetta ei esitellä.

4.2 Laitteiston esittely

4.2.1 Dell

Dell Inc. on vuonna 1984 perustettu yhdysvaltalainen tietokoneita, palvelimia ja verkkolaitteita valmistava yritys. Testikäyttöön valittu kytkin, S4048-ON käyttää Dell Networking Operating System-käyttöjärjestelmää. (Dell Inc 2017) Kuvassa 5 Dellin valmistama ja työhön valittu L3-kytkin, S4048-ON.

Kuva 5, Dell S4048-ON

4.2.2 Juniper Networks

Juniper Networks on vuonna 1996 perustettu yhdysvaltalainen verkkolaitteita valmistava yritys. Juniper tunnetaan paremmin runkoverkon laitteiden valmistajana, mutta se on vuodesta 2004 valmistanut myös yrityskäyttöön soveltuvia verkkolaitteita. Testikäyttöön valittu kytkin, QFX5100-48 käyttää FreeBSD:iin (Free Berkeley Software Distribution) pohjautuvaa JunOS-käyttöjärjestelmää (Juniper Network Operating System**)**. (Juniper Networks 2017) Kuvassa 6 Juniperin valmistama ja työhön valittu L3-kytkin, QFX5100-48.

Kuva 6, Juniper QFX5100-48

4.2.3 Nokia

Alcatel-Lucent oli ranskalainen, vuonna 2006 Acatelin ja Lucentin yhdistyessä syntynyt, tietoteknisiä laitteita ja ohjelmistoja valmistava yritys. Nokia osti Alcatel-Lucentin vuonna 2016. (Alcatel… 2017) Testikäyttöön valittu reititin, 7750 SR käyttää käyttöjärjestelmänään TimOS:ia. Kuvassa 7 Nokian valmistama ja työhön valittu reititin, 7750 SR.

Kuva 7, Nokia 7750 SR

4.2.4 Quanta Computer

Quanta Computer Inc. on taiwanilainen, vuonna 1988 perustettu elektroniikkaa valmistava yritys. Testikäyttöön valittu L3-kytkin, QuantaMesh T3048-LY8 käyttää käyttöjärjestelmänään Debian Linuxiin pohjautuvaan Cumulus Linuxia. (Quanta Computer Inc 2017) Kuvassa 8 Quantan valmistama ja työhön valittu L3-kytkin, T3048-LY8.

Kuva 8, Quanta Computer QuantaMesh T3048-LY8

4.3 Palveluiden esittely

Opinnäytetyössä keskityttiin palveluiden osalta ainoastaan käyttäjien tunnistamiseen tarkoitettuihin palveluihin. Palvelut asennettiin CentOS-käyttöjärjestelmään (Community Enterprise Operating System). CentOS on kaupalliseen RHEL:iin (Red Hat Enterprise Linux) pohjautuva avoimen lähdekoodin käyttöjärjestelmä. (CentOS 2016)

Palveluiden asennus tehtiin yhdelle virtuaaliselle palvelimelle. Tuotantokäytössä monien käyttäjien tunnistamiseen tarkoitetun palvelun asentaminen yhdelle palvelimelle ei ole järkevää, mutta testiympäristössä tämä toteutustapa valittiin sen yksinkertaisuuden vuoksi sekä käytössä olevien resurssien säästämiseksi.

Tässä opinnäytetyössä AAA-palveluiden on tarkoitus korvata laitteiden paikallisten käyttäjätietokantojen käyttö ja hakea käyttäjätiedot erilliseltä palvelimelta. Käyttäjätietoihin sisältyy myös käyttäjäoikeuksien määrittely.

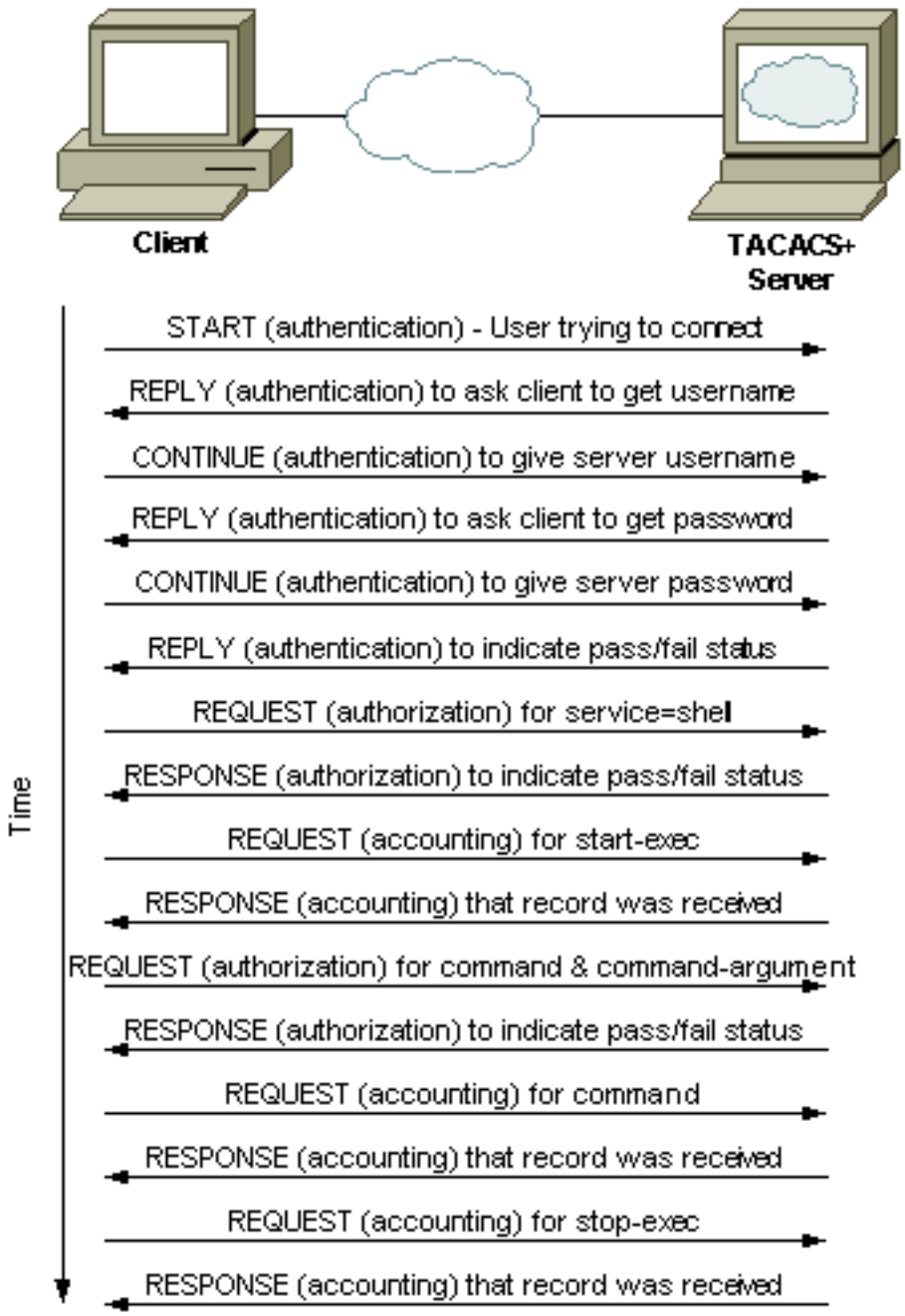
Opinnäytetyöhön valittiin kaksi eri protokollaa hyödyntävää palvelua, tac_plus sekä freeRADIUS. Vaikka kyseiset palvelut kehitettiin alun perin eri tarkoituksiin, jotka käydään jatkossa läpi, molemmilla on mahdollista varmentaa kytkimien käyttäjätietoja sekä rajata käyttäjäoikeuksia. Tässä kappaleessa käydään molempien palveluiden toimintatavat lyhyesti läpi.

4.3.1 TACACS+

TACACS (Terminal Access Controller Access Control System) on BBN Technologies’n vuonna 1984 kehittämä palvelu. TACACS kehitettiin UNIX-päätteiden käyttöä silmällä pitäen DARPAn (Defense Advanced Research Projects Agency) hallinnoiman ARPANET-verkon (The Advanced Research Projects Agency Network) käyttäjien hallintaan. Palvelun tarkoituksena oli mahdollistaa keskitetty käyttäjien tunnistaminen monille laitteille. (RFC 1492.)

Vuonna 1993 Cisco Systems Inc. kehitti uuden, TACACSin idealle pohjautuvan protokollan, TACACS+n. TACACS+ julkaistiin avoimella lähdekoodilla, joka poiki monia erilaisia, protokollaa hyödyntäviä palveluita, joista yksi on tac_plus.

TACACS+ käyttää oletuksena TCP-porttia (Transmission Control Protocol) 49. (RFC 1492.) TACACS+-palvelun toimintaperiaate on kuvattuna kuvassa 9.



Kuva 9, TACACS+

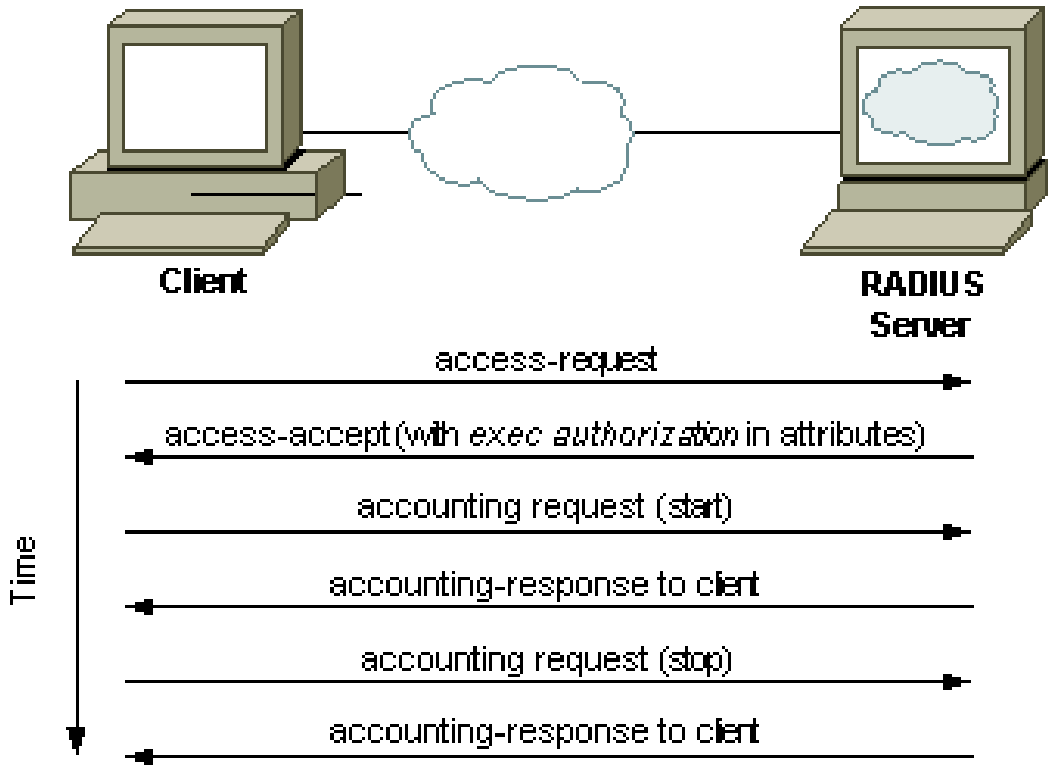
TACACS+ salaa kaiken kirjautumiseen liittyvän liikenteen ja sen toiminta on melko yksinkertaista. Käyttäjän laitteelle syöttämät kirjautumistiedot lähetetään palvelimelle, jonka jälkeen palvelin vertaa tietokannastaan löytyviä tietoja vastaavan käyttäjän tietoihin. Käyttäjän syöttämää salasanaa verrataan tietokannassa esiintyvään salasanaan ja mikäli salasana täsmää, lähetetään varmistus ja käyttöoikeudet takaisin laitteelle. (Woland 2014.)

4.3.2 RADIUS

RADIUS (Remote Authentication Dial-In User Service) on Carl Rigneyn vuonna 2000 kehittämä käyttäjien tunnistamiseen ja käyttöoikeuksien hallintaan tarkoitettu palvelu. (RFC 2865.)

Tosin kuin TACACS+, RADIUS on IETF:n standardisoima AAA-palvelu. RADIUS soveltuu moneen käyttötarkoitukseen ja sen alkuperäinen tarkoitus oli varmentaa käyttäjätietoja verkkoyhteyksissä. Protokolla toimii siirtoprotokollana monille varmennusprotokollille, kuten EAP:lle (Extensible Authentication Protocol).

TACACS+sta poiketen RADIUS hoitaa käyttäjätietojen varmennuksen ja käyttöoikeuksien hallinnan samassa viestissä ja salaa liikenteestä ainoastaan salasanan. RADIUS käyttää oletuksena UDP-portteja (User Datagram Protocol) 1812 & 1813 tai 1645 & 1646. (Woland 2014.) RADIUS-palvelun toimintaperiaate on kuvattuna kuvassa 10.



Kuva 10, RADIUS

5 PALVELUT

Tämän luvun tarkoituksena on käydä läpi tarvittavia asetusmuutoksia erilaisten salausmenetelmien sekä protokollien käyttöönottoon liittyen, sekä palveluiden asennukseen liittyviä toimenpiteitä. Asetuksiin ja asennuksiin tarvittavat komennot löytyvät liitteestä 1.

Kuten jo aiemmin mainittiin, palvelut asennetaan CentOS-käyttöjärjestelmään, versioon 7. Käyttöjärjestelmä on niin sanottu ”out-of-the-box”, eli mitään alustavia asetuksia ei ole tehty. Kaikista liitteistä löytyvistä asetustiedostoista on selvyyden vuoksi poistettu kaikki kommentoidut rivit. Palveluiden asennus vaatii toimivan internetyhteyden.

Molempien palveluiden asennukset suoritettiin kokonaisuudessaan järjestelmänvalvojan oikeuksilla, joten järjestelmään kirjauduttiin root-tunnuksilla. Asennukset alkoivat uusien pakettivarastoiden (Repository) luomisella, joista palveluihin tarvittavat asennustiedostot löytyvät. Pakettivarastot määriteltiin luomalla vim-tekstieditorilla uudet pakettivarastoiden määrittelyyn vaaditut tiedostot. Määrittelyiden jälkeen uudet pakettivarastot otettiin järjestelmän käyttöön, jonka jälkeen palvelut asennettiin järjestelmän omaa pakettienhallinta-työkalua käyttäen.

5.1 Tac_plus

Palvelun asennukseen sovellettiin Rene Molenaarin networklessons.com –verkkosivuille kirjoittamaa asennusohjetta, joka löytyy lähdeluettelosta.

Palveluun tarvittavien pakettien asennuksen jälkeen palvelun asetukset määriteltiin vim-tekstieditorin avulla palvelun asetustiedostoon. Asetuksissa määriteltiin kaksi käyttäjää eri oikeuksin, joista käyttäjä ”view” käyttää salasananaan selkokielistä salasanaa ”view” ja käyttäjä ”testi” käyttöjärjestelmän omaa käyttäjätietokantaa.

Lisää tac_plus-palvelun asetuksista voi lukea palvelun oletusasetukset-tiedostosta. Testiverkossa käytetty palvelun asetustiedosto löytyy kokonaisuudessaan liitteestä 2.

Asetusten määrittelyn jälkeen palvelu lisättiin käynnistyksen yhteydessä käynnistyvien ohjelmien listaan, jonka jälkeen palvelu käynnistettiin.

5.2 FreeRADIUS

Palvelun asennukseen sovellettiin Kiplangat Mutain computingforgeeks.com –verkkosivuille kirjoittamaa asennusohjetta, joka löytyy työn lähdeluettelosta.

FreeRADIUS käyttää asetuksiensa ja käyttäjätietojensa arkistoimiseen tietokantaa, joten sen asennus alkoi mariaDB-tietokannan asennuksella ja sen asetusten määrittelyllä. Tietokannan perusasetusten määrittelyn jälkeen tietokannalle määriteltiin käyttäjäoikeuksia ja tietokantaan luotiin tyhjä tietokanta, joka täytettiin palvelun mukana tulevalla aihio-tiedostolla.

Tietokannan asennuksen ja asetusten määrittelyn jälkeen asennettiin palveluun tarvittavat paketit. Toisin kuin tac_plus-palvelussa, freeRADIUS-palvelun asetusten muokkaaminen vaatii muutoksien tekemistä useaan eri tiedostoon. Lisää freeRADIUS-palvelun asetuksista voi lukea palvelun oletusasetukset-tiedostoista.

Testiverkossa käytetyt palvelun asetustiedostot löytyvät kokonaisuudessaan liitteistä 3-8. Asetusten määrittelyn jälkeen palvelu lisättiin käynnistyksen yhteydessä käynnistyvien ohjelmien listaan, jonka jälkeen palvelu käynnistettiin. Palveluun asennettiin myös RADIUS-yhteisön tekemä lisämoduuli PAM-autentikoinnin (A Pluggable Authentication Module) mahdollistamiseksi.

6 LAITEASETUKSET

Tässä luvussa käydään läpi laitteille syötetyt asetusmuutokset. Komennot tulee syöttää laitteelle yksi kerrallaan, sillä jotkut niistä vaativat käyttäjän syötettä komennon jälkeen. Laitteisiin viitataan tässä luvussa jokaisen laitteen laitevalmistajalla. Asetuksiin tarvittavat komennot löytyvät liitteestä 1.

AAA-palveluihin liittyviä laiteasetuksia määriteltäessä pidettiin käyttäjätunnuksia koskevat tavoitteet mielessä; tarkoituksena oli luoda jokaiselle käyttäjälle omat tunnukset joko järjestelmänvalvojan tai niin sanotuilla tarkastelijan oikeuksilla. Salasanojen tiivistealgoritmiksi valittiin MD5, koska Dellin, Nokian ja Quantan valmistamissa kytkimissä ei ole tukea SHA-1 –algoritmilla tiivistetyille salasanoille.

6.1 Dell

Testiympäristössä olevalle DELLin S4048-ON kytkimelle määriteltiin TACACS+ - ja RADIUS-palveluiden vaatimat asetukset. AAA-palveluiden tarjoamien käyttäjäoikeuksien hallinnan mahdollistamiseksi laitteelle tehtiin myös muutoksia paikallisiin käyttöoikeustasoihin.

Kytkin tuki salausta jokaisessa testikäyttöön valitussa reititys- ja hallintaprotokollassa. Laite määriteltiin myös käyttämään etäyhteysprotokollanaan SSH-protokollaa ja laitteen lokitiedostot uudelleen ohjattiin erilliselle lokipalvelimelle.

6.2 Juniper

Kuten DELLinkin tapauksessa, myös testiverkossa olevalle Juniperin valmistamalle QFX5100-48 -kytkimelle määriteltiin TACACS+ - ja RADIUS-palveluiden vaatimat asetukset. DELListä poiketen Juniperille määriteltiin uusia käyttäjiä ja käyttöoikeusluokkia.

Myös Juniperin kytkin tuki salausta jokaisessa testikäyttöön valitussa reititys- ja hallintaprotokollassa, mutta muista laitteista poiketen niiden salaus on mahdollista toteuttaa myös SHA1-tiivistealgoritmia käyttäen. Juniperikin määriteltiin käyttämään etäyhteyksissään SSH-protokollaa ja laitteen lokitiedot lähetettiin erilliselle lokipalvelimelle.

6.3 Nokia

Testiympäristössä olevalle Nokian 7750SR-reitittimelle määriteltiin ainoastaan TACACS-palvelun vaatimat asetukset, koska käyttöoikeuksien määrittely ei onnistunut toivotulla tavalla RADIUS-palvelua määriteltäessä. AAA-palvelun tarjoaman käyttöoikeuksien hallinnan mahdollistamiseksi laitteelle tehtiin myös muutoksia paikallisiin käyttöoikeustasoihin.

Myös Nokian reititin tuki salausta jokaisessa testikäyttöön valitussa reititys- ja hallintaprotokollassa MD5-viestintiivistysalgoritmia hyödyntämällä. Laite määriteltiin myös käyttämään etäyhteysprotokollanaan SSH-protokollaa ja laitteen lokitiedot lähetettiin erilliselle lokipalvelimelle.

6.4 Quanta

Testiympäristössä olevan Quanta Computerin valmistaman QuantaMesh T3048-LY8 -kytkimen käyttöjärjestelmä oli hyvin samankaltainen DELLin valmistaman S40348-ON-kytkimen käyttöjärjestelmän kanssa, joten DELLiin tehtyjä asetusmuutoksia voitiin hyödyntää Quantan asetusmuutoksia tehtäessä. Kytkimelle määriteltiin TACACS+ - ja RADIUS-palveluiden vaatimat asetukset. AAA-palveluiden tarjoamien käyttöoikeuksien hallinnan mahdollistamiseksi laitteelle tehtiin myös muutoksia paikallisiin käyttöoikeustasoihin.

Muista laitteista poiketen Quanta ei tukenut salausta IS-IS -reititysprotokollassa, eikä NTP-hallintaprotokollassa. Myös BGP-reititysprotokollaa salatessa ainoaksi vaihtoehdoksi paljastui tiivistämättömän salasanan käyttö. Myös Quanta määriteltiin käyttämään etäyhteysprotokollanaan SSH-protokollaa ja laitteen lokitiedot lähetettiin erilliselle lokipalvelimelle.

7 POHDINTA

Opinnäytetyön päätyttyä minulle oli selvää, että työ onnistui toimeksiantajan antamien kriteerien mukaisesti. Työn tuloksia kohtaan osoitettiin mielenkiintoa myös muissa organisaation sidosryhmissä.

Aloittaminen viivästyi muista työtehtävistä johtuen alkuperäisestä suunnitelmasta, mikä puolestaan vaikutti koko työn aikatauluun. Viivästymisen johdosta alun perin laajemmiksi suunnitellut testaukset jätettiin minimiin ja opinnäytetyön ulkopuolelle, mikä joudutti työn valmistumista huomattavasti.

Protokollien ja palveluiden toimivuuden testaus suoritettiin pelkistetysti. Reititysprotokollien toimivuus todettiin naapuruussuhteiden syntymisellä. NTP-protokollan toimivuus todettiin lokitiedoista vertailemalla laitteiden paikallisten lokitietojen aikaleimoja NTP-protokollan käyttämän palvelimen lokitietojen aikaleimoihin.

Etäyhteysprotokollien toimivuus testattiin yksinkertaisesti etäyhteyksiä eri protokollilla kokeilemalla ja varmistamalla laitteiden avonaiset portit portinskannausohjelman, NMAPin (Network Mapper) avulla. Lokitietojen lähetyksen toimivuus todennettiin lokipalvelimelta, AAA-palveluiden toimivuus palveluiden lokitiedoista.

Lähteiksi etsin ensisijaisesti IETF:n julkaisemia, internetiä koskevia RFC-standardeja tai Yhdysvaltain hallituksen julkaisemia FIPS-standardeja, jotta työssä käytettäviä lähteitä voitaisiin pitää mahdollisimman luotettavina. Myös erilaisia akateemisia tutkimustöitä pidin suuressa arvossa. Tapauksissa joissa vaihtoehtoina oli useita eri kirjalähteitä, pyrin valitsemaan lähteet kriitikoiden arviointien mukaan.

Keskeisimpinä työssä käytettävinä kvalitatiivisina tutkimusmenetelminä olivat pääasiallisesti havainnointiin ja tekstianalyysiin liittyvät tutkimusmetodit. Havainnointi tapahtui niin osallistuvana kuin piilohavaintonakin, mutta työn luonteen vuoksi liittyi pääasiallisesti eri laitteiden laitekonfiguraatioihin, eikä niinkään suoraan eri henkilöiden käyttäytymiseen. Tein tutkimusta tekstianalyysin osalta lukemalla ja tutustumalla yrityksen tarjoamaan materiaaliin, sekä alan kirjallisuutta ja muita julkaisuita hyväksi käyttäen.

Opinnäytetyön aikana otin käyttöön suurimmassa osassa laboratorioverkon kytkimiä työssä käsiteltävän tunnistuspalvelun, tac_plus:n. Keskitetyn käyttäjätietokannan edut paljastuivat minulle hyvin nopeasti ja jo ensimmäisen virhemuokkauksen sattuessa vikatilanteesta aiheutuneet vahingot saatiin korjattua hyvin nopeasti.

Opinnäytetyön alkaessa olin opiskeluideni aikana jo hieman tutustunut erilaisiin salausmenetelmiin ja protokollien suojaukseen liittyviin toimenpiteisiin. Tunnistuspalveluiden asennukseen ja asetusten muokkaukseen liittyvistä asioista sekä Juniperin ja Nokian laitteiden käyttöliittymien käytöstä minulla ei ollut kokemusta. Näihin sain tarvittavan määrän tietoa kollegoiltani, laitteiden ja palveluiden manuaaleista sekä ennen kaikkea laitteita kokeilemalla. Työ kasvatti entisestään omaa mielenkiintoani kryptologiaa, kvanttitietokoneita ja keskitettyä käyttäjienhallintaa kohtaan.

Hyödynsin työssä asennettujen palveluiden ominaisuuksia melko pintapuolisesti, mutta tulevaisuudessa aion selvittää niiden laajempia käyttötarkoituksia. Pääsen myös tutustumaan yhtenäisten käyttäjätietokantojen hallintaan, koska tulevaisuudessa palvelut asennetaan erillisille palvelimille. Palveluita tullaan mahdollisuuksien puitteissa laajentamaan graafisilla, mahdollisesti selaimissa toimivilla käyttöliittymillä.

Solmin opinnäytetyön aikana vakituisen työsuhteen Nokia Networksin kanssa, jonka myötä pääsen kehittämään työtäni edelleen, muiden töiden ohessa. Henkilökohtaisesti olen työhöni tyytyväinen, vaikka oman kiinnostuksen vuoksi useiden kryptologisten menetelmien rajaaminen työn ulkopuolelle tuottikin useaan otteeseen niin sanotusti harmaita partakarvoja.

8 LÄHTEET

- Alcatel-Lucent's History. Luettu 02.03.2017. https://networks.nokia.com/about/history
- Barad, K., 2007. Meeting the Universe halfway: Quantum Physics and the Entanglement of Matter and Meaning. Duke University Press. ISBN: 9780822339175
- CentOS. About > About CentOS. Luettu 20.01.2017. https://www.centos.org/about/
- D-wave Systems. NEWS > PRESS RELEASES > Sep.28. 2015. Luettu 16.02.2017. https://www.dwavesys.com/press-releases/d-wave-systems-announces-multi-year-agreement-provide-its-technology-google-nasa-and
- Dell Inc. About Dell. Luettu 31.01.2017. http://www.dell.com/learn/us/en/uscorp1/corp-comm
- Dell S4048-ON. Luettu 06.04.2017. http://www.dell.com/support/home/us/en/04/product-support/product/force10-s4048-on/research
- FIPS 180-4, 2015. Secure Hash Standard (SHS). http://csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf
- Juniper Networks. About Juniper. Luettu 31.01.2017. http://www.juniper.net/us/en/company/
- Juniper QFX5100-48. Luettu 06.04.2017. https://www.juniper.net/us/en/products-services/switching/qfx-series/qfx5100/
- Kahn, D., 1973. The Codebreakers – The Story of Secret Writing. http://mindguruindia.com/wp-content/uploads/2014/06/MP069_The-CodeBreakers.pdf

- Molenaar, R., 2013. How to install TACACS+ on Linux CentOS. Luettu 05.12.2016. https://networklessons.com/uncategorized/how-to-install-tacacs-on-linux-centos/
- Monz, T., Nigg, D., Martinez, E., Brandl, M., Schindler, P., Rines, R., Wang, S., Chuang, I. & Blatt, R., 2015. Realization of a scalable Shor algorithm. https://arxiv.org/pdf/1507.08852.pdf
- Mutai, K., 2016. Install FreeRADIUS and Daloradius on CentOS 7 and RHEL 7. Luettu 05.12.2016. http://computingforgeeks.com/installing-freeradius-and-daloradius-centos-7/
- Nokia 7750 SR. Luettu 06.04.2017. https://networks.nokia.com/products/7750-service-router-mobile-gateway
- Nokia Oyj. Tietoa meistä. Luettu 15.12.2016. http://www.nokia.com/fi_fi/tie-toa-meista/keita-olemme
- Quanta Computer Inc. About Quanta. Luettu 31.01.2017. http://www.quan-tatw.com/quanta/english/about/company.aspx
- Quanta Computer QuantaMesh T3048-LY8. Luettu 06.04.2017. https://www.unixplus.com/collections/quanta/switch
- RADIUS. Luettu 06.04.2017. http://www.cisco.com/c/en/us/support/docs/secu-rity-vpn/remote-authentication-dial-user-service-radius/13838-10.html
- RFC 1321, 1992. The MD5 Message-Digest Algorithm. https://tools.ietf.org/html/rfc1321
- RFC 1492, 1993. An Access Control Protocol, Sometimes Called TACACS. https://tools.ietf.org/html/rfc1492
- RFC 2865, 2000. Remote Authentication Dial In User Service (RADIUS). http://www.ietf.org/rfc/rfc2865.txt?number=2865

- RFC 4251, 2006. The Secure Shell (SSH) Protocol Architecture. https://tools.ietf.org/html/rfc4251
- Schneier, B., 1996. Applied Cryptography – Protocols, Algorithms and Source Code in C. John Wiley & Sons, Inc., New York. ISBN: 978-0-471-11709-4
- Singh, S., 1999. Koodikirja: salakirjoituksen historia muinaisesta Egyptistä kvanttikryptografiaan. Kustannusosakeyhtiö Tammi, Helsinki. ISBN: 9513115445
- TACACS+. Luettu 06.04.2017. http://www.cisco.com/c/en/us/support/docs/security-vpn/remote-authentication-dial-user-service-radius/13838-10.html
- Woland, A, 2014. RADIUS versus TACACS+. Luettu 01.03.2017. http://www.networkworld.com/article/2838882/radius-versus-tacacs.html

9 LIITTEET

Liite 1. Käyttöohje 1(20)

Tac_plus

Asennus alkaa uuden pakettivaraston (Repository) luomisella, josta palveluun tarvittavat asennustiedostot löytyvät. Testiverkossa käytetty palvelun asetustiedosto löytyy liitteestä 2.

```
1. sudo su root
2. vim /etc/yum.repos.d/nux-misc.repo
```

1. Kirjaudutaan järjestelmänvalvojan oikeuksilla

2. Avataan (tässä tapauksessa myös luodaan) tiedosto vim-tekstieditorilla.

```
[nux-misc]
name=Nux Misc
baseurl=http://li.nux.ro/download/nux/misc/el6/x86_64/
enabled=0
gpgcheck=1
gpgkey=http://li.nux.ro/download/nux/RPM-GPG-KEY-nux.ro
```

Seuraavaksi syötetään ylläolevan tekstilaatikon tiedot tiedostoon. Tämä tiedosto määrittää uuden pakettivaraston tiedot.

```
1. yum –enablerepo=nux-misc install tac_plus
2. vim /etc/tac_plus.conf
```

1. Sallitaan uuden pakettivaraston käyttö paketinhallintaohjelmassa ja asennetaan paketti tac_plus

2. Avataan palvelun oletusasetukset-tiedosto vim-tekstieditorilla.

```
1. adduser testi
2. passwd testi
3. echo “user = testi { login = PAM member = admin }” >> /etc/tac_plus.conf
```

1. Luodaan käyttäjä ”testi”

2. Asetetaan salasana käyttäjälle ”testi”

3. Lisätään rivi ”user = testi { login = PAM member = admin }” tiedostoon /etc/tac_plus.conf.

Asetuksissa määritellään kaksi käyttäjää eri oikeuksin, joista käyttäjä ”view” käyttää salasananaan selkokielistä salasanaa ”view” ja käyttäjä ”testi” käyttöjärjestelmän omaa käyttäjätietokantaa. Lisää tac_plus:n asetuksista voi lukea palvelun oletusasetukset-tiedostosta.

```
1. chkconfig --add tac_plus
2. chkconfig tac_plus on
3. service tac_plus start
```

1. Lisätään tac_plus –palvelu järjestelmän käynnistyksen yhteydessä käynnistyviin ohjelmiin

2. Sallitaan palvelun käynnistyminen järjestelmän käynnistyksen yhteydessä

3. Käynnistetään palvelu.

FreeRADIUS

FreeRADIUS käyttää asetuksiensa ja käyttäjätietojensa arkistoimiseen tietokantaa, joten sen asennus alkaa mariaDB-tietokannan asennuksella. Palvelun asetusten muokkaaminen vaatii muutoksien tekemistä moneen eri tiedostoon. Testiverkossa käytetyt asetustiedostot löytyvät liitteistä.

```
1. sudo su root
2. vim /etc/yum.repos.d/nux-misc.repo
```

1. Kirjaudutaan järjestelmänvalvojan oikeuksilla

2. Avataan (tässä tapauksessa myös luodaan) tiedosto vim-tekstieditorilla.

```
[mariadb]
name=MariaDB
baseurl=http://yum.mariadb.org/10.1/centos7-amd64
gpgkey=https://yum.mariadb.org/RPM-GPG-KEY-MariaDB
gpgcheck=1
```

Seuraavaksi syötetään ylläolevan tekstilaatikon tiedot tiedostoon. Tämä tiedosto määrittää uuden pakettivaraston tiedot.

```
1. yum –enablerepo=mariadb install mariadb-server mariadb
2. systemctl start mariadb
3. systemctl enable mariadb
4. mysql_secure_installation
5. vim /etc/my.cnf
```

1. Sallitaan uuden pakettivaraston käyttö paketinhallintaohjelmassa ja asennetaan paketit mariadb-server ja mariadb

2. Käynnistetään mariadb-tietokanta

3. Sallitaan mariadb-tietokannan käynnistys järjestelmän käynnistyksen yhteydessä

4. Käynnistetään tietokannan automaattinen asetustenmuokkaus. Tietokanta alustetaan oletusasetuksilla

5. Avataan tietokannan asetustiedosto vim-tekstieditorilla.

```
[mysqld]
bind-address=127.0.0.1
```

Seuraavaksi syötetään ylläolevan tekstilaatikon tiedot tiedostoon. Tämä tiedosto määrittää tietokannan muokkaamiseen sallitun lähdeosoitteen, joka tietoturvallisista syistä rajataan palvelimen paikalliseen osoitteeseen. Ennen RADIUS-palvelun asennusta tietokantaan pitää vielä luoda tyhjä kanta.

```
1. mysql -u root -p -e "CREATE DATABASE radius"
2. mysql -u root -p -e "show databases"
3. mysql -u root -p
4. GRANT ALL ON radius.* TO radius@localhost IDENTIFIED BY "[password]";
5. FLUSH PRIVILEGES;
6. \q
```

1. Luodaan tietokantaan ”radius”-niminen tietokanta

2. Varmistetaan taulun luonti

3. Siirrytään tietokannan muokkaustilaan

4. Annetaan paikalliselle käyttäjälle ”radius” täydet oikeudet ”radius”-nimiseen tietokantaan salasanaa vastaan

5. Päivitetään tietokannan oikeudet

6. Poistutaan tietokannan muokkaustilasta.

```
1. yum -y install freeradius freeradius-utils freeradius-mysql unzip
2. systemctl start radiusd.service
3. systemctl enable radiusd.service
```

1. Asennetaan paketit freeradius, freeradius-util, freeradius-mysql sekä unzip
2. Käynnistetään RADIUS-palvelu
3. Sallitaan RADIUS-palvelun käynnistys järjestelmän käynnistyksen yhteydessä.

```
1. mysql -u root -p radius < /etc/raddb/mods-config/sql/main/mysql/schema.sql
2. ln -s /etc/raddb/mods-available/sql /etc/raddb/mods-enabled/
3. wget https://github.com/FreeRADIUS/pam_radius/archive/master.zip
4. unzip master.zip –d . && cd master
5. ./configure
6. make
```

1. Täytetään luotu kanta tarvittavilla tauluilla “radius database” –aihion avulla
2. Luodaan linkki kahden kansion välille
3. Ladataan PAM-moduuli
4. Puretaan ladattu paketti ja siirrytään kansioon
5. Ajetaan automaattinen moduulin asennus
6. Asennetaan ladattu moduuli.

```
1. vim /etc/raddb/mods-available/sql
2. chgrp -h radiusd /etc/raddb/mods-enabled/sql
3. vim /etc/raddb/mods-available/pam
4. chgrp –h radiusd /etc/raddb/mods-enabled/pam
5. vim /etc/raddb/clients.conf
6. vim /etc/raddb/pam_radius_auth.conf
7. vim /etc/raddb/radius.conf
8. vim /etc/raddb/users
9. systemctl restart radius.service
```

1. Avataan SQL-moduulin asetustiedosto vim-tekstieditorilla (liite 3)
2. Muutetaan asetustiedoston ryhmäoikeuksia
3. Avataan PAM-moduulin asetustiedosto vim-tekstieditorilla (liite 4)
4. Muutetaan asetustiedoston ryhmäoikeuksia
5. Avataan RADIUS-asiakkaiden asetustiedosto (liite 5)
6. Avataan PAM-asetusten asetustiedosto (liite 6)
7. Avataan RADIUS-asetusten asetustiedosto (liite 7)
8. Avataan RADIUS-käyttäjien asetustiedosto (liite 8)
9. Käynnistetään RADIUS-palvelu uudelleen.

Dell

AAA-tunnistuspalvelu

```
1. configure
2. username admin secret [password] privilege 15
3. no enable password
4. privilege exec level 7 show running-config
5. aaa authentication login NO_AUTHENTICATION none
6. aaa authentication login AAA_AUTHENTICATION local tacacs+ radius
7. aaa authorization exec AAA_AUTHENTICATION local tacacs+ radius
8. tacacs-server host [ip] key [password]
9. radius-server host [ip] key [password]
10. line con 0
11. login authentication NO_AUTHENTICATION
12. line vty 0 4
13. login authentication AAA_AUTHENTICATION
14. authorization exec AAA_AUTHENTICATION
15. end
16. copy running-config startup-config
```

1. Siirrytään asetusten muokkaustilaan

2. Luodaan käyttäjä ”admin” salatulla salasanalla ja annetaan tälle oikeustaso 15

3. Poistetaan ”enable”-tilan salasana

4. Lisätään komento “show running-config” käyttäjätasolle 7

5. Luodaan tunnistuslista ilman tunnistusmenetelmiä

6. Luodaan tunnistuslista käyttäen ensisijaisesti paikallista tietokantaa, toissijaisesti TACACS+ -palvelua ja viimeiseksi RADIUS-palvelua

7. Luodaan käyttöoikeuslista käyttäen ensisijaisesti paikallista tietokantaa, toissijaisesti TACACS+ -palvelua ja viimeiseksi RADIUS-palvelua

8. Määritellään TACACS+ -palvelimen osoite ja salasana

9. Määritellään RADIUS-palvelimen osoite ja salasana

10. Siirrytään konsoliportin muokkaustilaan

11. Lisätään tunnistuslista konsoliporttiin

12. Siirrytään etäyhteysporttien muokkaustilaan

13. Lisätään tunnistuslista etäyhteysporteille

14. Lisätään käyttöoikeuslista etäyhteysporteille

15. Siirrytään pois asetusten muokkaustilasta

16. Tallennetaan tehdyt muutokset.

BGP-reititysprotokolla

1. **configure**
2. **router bgp [as-number]**
3. **neighbor [neighbor router-id] password [password]**
4. **end**
5. **copy running-config startup-config**

1. Siirrytään asetusten muokkaustilaan
2. Siirrytään BGP-reititysprotokollan muokkaustilaan
3. Määritellään BGP-naapurin reititystunnus ja salasana
4. Siirrytään pois asetusten muokkaustilasta
5. Tallennetaan tehdyt muutokset.

IS-IS –reititysprotokolla

1. **configure**
2. **router isis [name]**
3. **domain-password hmac-md5 [password]**
4. **area-password hmac-md5 [password]**
5. **end**
6. **copy running-config startup-config**

1. Siirrytään asetusten muokkaustilaan
2. Siirrytään IS-IS –reititysprotokollan muokkaustilaan
3. Määritellään IS-IS –piirisalasana käyttäen MD5-viestintiivistystä
4. Määritellään IS-IS –aluesalasana käyttäen MD5-viestintiivistystä
5. Siirrytään pois asetusten muokkaustilasta
6. Tallennetaan tehdyt muutokset.

NTP-hallintaprotokolla

1. **configure**
2. **ntp authenticate**
3. **ntp authentication-key [key-id] md5 0 [password]**
4. **ntp trusted-key [key-id]**
5. **end**
6. **copy running-config startup-config**

1. Siirrytään asetusten muokkaustilaan
2. Määritellään NTP-protokolla käyttämään tunnistusta
3. Määritellään NTP-protokollalle salasana käyttäen MD5-viestintiivistystä
4. Lisätään luotu salasana luotettujen salasanojen listaan
5. Siirrytään pois asetusten muokkaustilasta
6. Tallennetaan tehdyt muutokset.

OSPF-reititysprotokolla

1. **configure**
2. **interface [interface-id]**
3. **ip ospf authentication-key [password]**
4. **ip ospf message-digest-key 1 md5 [password]**
5. **end**
6. **copy running-config startup-config**

1. Siirrytään asetusten muokkaustilaan
2. Siirrytään portin muokkaustilaan
3. Määritellään OSPF-reititysprotokollalle salasana
4. Määritellään luotu salasana käyttämään MD5-viestintiivystystä
5. Siirrytään pois asetusten muokkaustilasta
6. Tallennetaan tehdyt muutokset.

SYSLOG-järjestelmäloki

```
1. configure
2. service timestamps log datetime msec
3. no logging console
4. no logging monitor
5. logging [hostname/ip] tcp 6514
6. logging trap informational
7. logging source-interface loopback 0
8. logging on
9. end
10. copy running-config startup-config
```

1. Siirrytään asetusten muokkaustilaan
2. Määritellään lokiviestit käyttämään aikaleimaa
3. Poistetaan lokiviestin tulostus konsoliportilta
4. Poistetaan lokiviestin tulostus etäyhteysporteilta
5. Määritellään lokipalvelimen osoite sekä TCP-portti
6. Määritellään lähetettävien lokiviestien lähetystaso
7. Määritellään lokiviestit käyttämään lähdeosoitteenaan ”loopback 0”-portille määritettyä osoitetta
8. Käynnistetään lokiviestien lähetys
9. Siirrytään pois asetusten muokkaustilasta
10. Tallennetaan tehdyt muutokset.

SSH-etähallintaprotokolla

```
1. configure
2. no ip telnet
3. ip ssh server version 2
4. end
5. copy running-config startup-config
```

1. Siirrytään asetusten muokkaustilaan
2. Poistetaan Telnet-palvelu käytöstä
3. Otetaan SSH-palvelun 2. versio käyttöön
4. Siirrytään pois asetusten muokkaustilasta
5. Tallennetaan tehdyt muutokset.

Juniper

AAA-tunnistuspalvelu

1. **configure**
2. **set system tacplus-server [ip] secret [password]**
3. **set system radius-server [ip] secret [password]**
4. **set system authentication-order password authentication-order tacplus authentication-order radius**
5. **set system login user remote class super-user**
6. **set system login class read-only-local permissions view permissions view-configuration**
7. **set system login user view class read-only-local**
8. **set system login user admin authentication plain-text-password**
9. **commit and-quit**

1. Siirrytään asetusten muokkaustilaan

2. Määritellään TACACS+ -palvelimen osoite ja salasana

3. Määritellään RADIUS-palvelimen osoite ja salasana

4. Määritellään tunnistusjärjestys käyttämään ensisijaisesti TACACS+ -palvelinta, toissijaisesti RADIUS-palvelinta

5. Luodaan käyttäjä “remote” super-user -käyttöoikeuksilla

6. Lisätään read-only-local –luokalle oikeus komentoihin “view” ja “view-configuration”

7. Luodaan käyttäjä ”view” read-only -käyttöoikeuksilla

8. Asetetaan “admin” –käyttäjälle salasana

9. Tallennetaan tehdyt muutokset ja poistutaan asetusten muokkaustilasta.

BGP-reititysprotokolla

1. **configure**
2. **set protocols bgp authentication-algorithm md5 authentication-key [password]**
3. **commit and-quit**

1. Siirrytään asetusten muokkaustilaan

2. Määritellään BGP-reititysprotokollalle salasana käyttäen MD5-viestintiivystystä

3. Tallennetaan tehdyt muutokset ja poistutaan asetusten muokkaustilasta.

IS-IS –reititysprotokolla

1. **configure**
2. **set security authentication-key-chains key-chain [name] key [id] algorithm md5 secret [password]**
3. **set protocols isis interface [interface] level [id] hello-authentication-type md5 hello-authentication-key-chain [key-chain]**
4. **commit and-quit**

1. Siirrytään asetusten muokkaustilaan

2. Luodaan avainnippu sekä avain käyttäen MD5-viestintiivistystä

3. Määritellään IS-IS –reititysprotokolla käyttämään luotua avainta MD5-viestintiivistystä hyödyntäen

4. Tallennetaan tehdyt muutokset ja poistutaan asetusten muokkaustilasta.

NTP-hallintaprotokolla

1. **configure**
2. **set system ntp authentication-key [key-id] type md5 value [password]**
3. **set system ntp trusted-key [key-id]**
4. **set system ntp peer [IP] key [key-id]**
5. **commit and-quit**

1. Siirrytään asetusten muokkaustilaan

2. Määritellään NTP-protokolla käyttämään salasanaa MD5-viestintiivistyksellä

3. Lisätään luotu salasana luotettujen salasanojen listaan

4. Määritellään NTP-palvelimen osoite sekä salasana

5. Tallennetaan tehdyt muutokset ja poistutaan asetusten muokkaustilasta.

OSPF-reititysprotokolla

1. **configure**
2. **set protocols ospf area [area-id] interface [interface-id] authentication md5 [key-id] key [password]**
3. **commit and-quit**

1. Siirrytään asetusten muokkaustilaan

2. Määritellään OSPF-reititysprotokollaa käyttävä portti käyttämään MD5-viestintiivistystä hyödyntävää salasanaa

3. Tallennetaan tehdyt muutokset ja poistutaan asetusten muokkaustilasta.

SYSLOG-järjestelmäloki

1. **configure**
2. **set system syslog host [IP] any error**
3. **set system syslog file [filename]**
4. **set system syslog file [filename] structured-data brief**
5. **set system syslog file [filename] archive size 1g**
6. **set system syslog time-format millisecond**
7. **delete system syslog user ***
8. **delete system syslog console**
9. **commit and-quit**

1. Siirrytään asetusten muokkaustilaan

2. Määritellään SYSLOG-palvelimen osoite

3. Luodaan lokiviesteille paikallinen tiedosto

4. Määritellään lokiviesti-tiedoston rakenne

5. Määritellään lokiviesti-tiedoston maksimi koko

6. Määritellään lokiviestien aikaleiman formaatti

7. Poistetaan lokiviestien tulostus käyttäjille

8. Poistetaan lokiviestien tulostus konsoliportille

9. Tallennetaan tehdyt muutokset ja poistutaan asetusten muokkaustilasta.

SSH-etähallintaprotokolla

1. **configure**
2. **set system services ssh root-login allow**
3. **delete system services telnet**
4. **commit and-quit**

1. Siirrytään asetusten muokkaustilaan

2. Määritellään SSH-palvelu ja sallitaan järjestelmänvalvojien pääsy

3. Poistetaan Telnet-palvelu käytöstä

4. Tallennetaan tehdyt muutokset ja poistutaan asetusten muokkaustilasta.

Nokia

AAA-tunnistuspalvelu

```
1.  configure system security
2.  profile “default”
3.  entry 90
4.  match “admin display-config”
5.  action permit
6.  exit
7.  exit
8.  tacplus
9.  authorization use-priv-lvl
10. priv-lvl-map
11. priv-lvl 7 “default”
12. priv-lvl 15 “administrative”
13. exit
14. server 1 address [ip] secret [password]
15. exit all
16. admin save
```

1. Siirrytään turvallisuusasetusten muokkaustilaan
2. Siirrytään profiilinmuokkaustilaan
3. Siirrytään muokkaamaan profiilin oikeuslistan kohtaa 90
4. Luodaan kohdalle komentotunniste
5. Sallitaan tunnisteen mukaisen komennon suorittaminen
6. Poistutaan oikeuslistan muokkaustilasta
7. Poistutaan profiilinmuokkaustilasta
8. Siirrytään TACACS+ -muokkaustilaan
9. Mahdollistetaan käyttäjäoikeuksien määräytyminen oikeustason mukaisesti
10. Siirrytään oikeustasolistan muokkaustilaan
11. Määritellään oikeustaso 7 käyttämään profiilia ”default”
12. Määritellään oikeustaso 15 käyttämään profiilia ”administrative”
13. Poistutaan oikeustasolistan muokkaustilasta
14. Määritellään ensisijaisen TACACS+ -palvelimen IP-osoite ja salasana
15. Poistutaan asetusten muokkaustilasta
16. Tallennetaan tehdyt muutokset.

BGP-reititysprotokolla

1. **configure router bgp**
2. **group [group]**
3. **authentication-key [password]**
4. **exit all**
5. **admin save**

1. Siirrytään BGP-reititysprotokollan muokkaustilaan
2. Siirrytään reititysryhmän muokkaustilaan
3. Määritellään reitityssalasana
4. Poistutaan asetusten muokkaustilasta
5. Tallennetaan tehdyt muutokset.

IS-IS –reititysprotokolla

1. **configure router isis**
2. **authentication-type message-digest**
3. **authentication-key [password]**
4. **exit**
5. **admin save**

1. Siirrytään IS-IS –reititysprotokollan muokkaustilaan
2. Määritellään reitityssalasana käyttämään tiivistealgoritmia
3. Määritellään reitityssalasana
4. Poistutaan IS-IS –reititysprotokollan muokkaustilasta
5. Tallennetaan tehdyt muutokset.

NTP-hallintaprotokolla

1. **configure system time ntp**
2. **authentication-key [key-id] key [password] type message-digest**
3. **exit**
4. **admin save**

1. Siirrytään NTP-protokollan muokkaustilaan
2. Määritellään protokollalle salasana MD5-tiivistealgoritmia käyttäen
3. Poistutaan NTP-protokollan muokkaustilasta
4. Tallennetaan tehdyt muutokset.

OSPF-reititysprotokolla

1. **configure router ospf**
2. **area [area-id]**
3. **interface [interface-id]**
4. **authentication-type message-digest**
5. **authentication-key [password]**
6. **exit all**
7. **admin save**

1. Siirrytään OSPF-reititysprotokollan muokkaustilaan
2. Siirrytään OSPF-alueen muokkaustilaan
3. Siirrytään portin muokkaustilaan
4. Määritellään reitityssalasana käyttämään MD5-tiivistealgoritmia
5. Määritellään reitityssalasana
6. Poistutaan asetusten muokkaustilasta
7. Tallennetaan tehdyt muutokset.

SSH-etähallintaprotokolla

```
1. configure system security
2. ssh version 2
3. ssh no server-shutdown
4. exit
5. admin save
```

1. Siirrytään turvallisuusasetusten muokkaustilaan
2. Määritellään SSH-palvelun versio
3. Käynnistetään SSH-palvelu
4. Poistutaan turvallisuusasetusten muokkaustilasta
5. Tallennetaan tehdyt muutokset.

SYSLOG-järjestelmäloki

```
1. configure log
2. syslog [syslog-id] address [ip]
3. syslog [syslog-id] level [syslog-level]
4. syslog [syslog-id] port [port]
5. exit
6. admin save
```

1. Siirrytään lokitietojen muokkaustilaan
2. Määritellään SYSLOG-palvelimen IP-osoite
3. Määritellään lähetettävin lokiviestien taso
4. Määritellään SYSLOG-palvelimen portti
5. Poistutaan lokitietojen muokkaustilasta
6. Tallennetaan tehdyt muutokset.

Quanta

AAA-tunnistuspalvelu

```
1. configure
2. username admin passwd 0 [password] level 15
3. aaa authentication login NO_AUTHENTICATION none
4. aaa authentication login AAA_AUTHENTICATION local tacacs radius
5. aaa authentication enable AAA_AUTHENTICATION tacacs radius
6. tacacs-server host [ip]
7. key [password]
8. exit
9. radius server host auth [ip]
10. radius server key auth [ip]
11. line console
12. login authentication NO_AUTHENTICATION
13. exit
14. line ssh
15. login authentication AAA_AUTHENTICATION
16. enable authentication AAA_AUTHENTICATION
17. end
18. copy running-config startup-config
```

1. Siirrytään asetusten muokkaustilaan

2. Luodaan käyttäjä ”admin” salasanalla ja annetaan tälle oikeustaso 15

3. Luodaan kirjautumiseen tarkoitettu tunnistuslista ilman tunnistusmenetelmiä

4. Luodaan kirjautumiseen tarkoitettu tunnistuslista käyttäen ensisijaisesti paikallista tietokantaa, toissijaisesti TACACS+ -palvelua ja viimeiseksi RADIUS-palvelua

5. Luodaan ”enable”-tilaan tarkoitettu tunnistuslista käyttäen ensisijaisesti paikallista tietokantaa, toissijaisesti TACACS+ -palvelua ja viimeiseksi RADIUS-palvelua

6. Määritellään TACACS+ -palvelimen osoite ja siirrytään sen muokkaustilaan

7. Määritellään TACACS+ -palvelimelle salasana

8. Poistutaan TACACS+ -palvelimen muokkaustilasta

9. Määritellään RADIUS-palvelimen osoite

10. Määritellään RADIUS-palvelimen salasana

11. Siirrytään konsoliportin muokkaustilaan

12. Lisätään tunnistuslista konsoliporttiin

13. Poistutaan konsoliportin muokkaustilasta

14. Siirrytään etäyhteysporttien muokkaustilaan

15. Lisätään kirjautumiseen tarkoitettu tunnistuslista etäyhteysporteille

16. Lisätään ”enable”-tilaan tarkoitettu tunnistuslista etäyhteysporteille
17. Poistutaan asetusten muokkaustilasta
18. Tallennetaan tehdyt muutokset.

BGP-reititysprotokolla

1. **configure**
2. **router bgp [as-number]**
3. **neighbor [neighbor router-id] password [password]**
4. **end**
5. **copy running-config startup-config**

1. Siirrytään asetusten muokkaustilaan
2. Siirrytään BGP-reititysprotokollan muokkaustilaan
3. Määritellään BGP-naapurin reititystunnus ja salasana
4. Poistutaan asetusten muokkaustilasta
5. Tallennetaan tehdyt muutokset.

OSPF-reititysprotokolla

1. **configure**
2. **interface [interface-id]**
3. **ip ospf authentication encrypt 0 [password] [key-id]**
4. **end**
5. **copy running-config startup-config**

1. Siirrytään asetusten muokkaustilaan
2. Siirrytään portin muokkaustilaan
3. Määritellään OSPF-reititysprotokollalle salasana
4. Poistutaan asetusten muokkaustilasta
5. Tallennetaan tehdyt muutokset.

SYSLOG-järjestelmäloki

```
1. configure
2. no logging console
3. no logging monitor
4. logging host [ip] ipv4 [port] info
5. logging syslog source-interface loopback 0
6. end
7. copy running-config startup-config
```

1. Siirrytään asetusten muokkaustilaan
2. Poistetaan lokiviestin tulostus konsoliportilta
3. Poistetaan lokiviestin tulostus etäyhteysporteilta
4. Määritellään lokipalvelimen osoite, TCP-portti sekä lokiviestien lähetystaso
5. Määritellään lokiviestit käyttämään lähdeosoitteenaan ”loopback 0”-portille määritettyä osoitetta
6. Poistutaan asetusten muokkaustilasta
7. Tallennetaan tehdyt muutokset.

SSH-etäyhteysprotokolla

```
1. configure
2. no telnet sessions
3. ip ssh protocol 2
4. end
5. copy running-config startup-config
```

1. Siirrytään asetusten muokkaustilaan
2. Poistetaan Telnet-palvelu käytöstä
3. Otetaan SSH-palvelun 2. versio käyttöön
4. Poistutaan asetusten muokkaustilasta
5. Tallennetaan tehdyt muutokset.

Liite 2. tac_plus asetukset 1(3)

```
key = testi
accounting file = /var/log/tac.acct
default authentication = file /etc/passwd

acl = default                {

}

group = admin {
        service = exec {
                priv-lvl = 15
        }
        cmd = username {
                permit .*
        }
        cmd = enable {
                permit .*
        }
        cmd = show {
                permit .*
        }
        cmd = exit {
                permit .*
        }
        cmd = configure {
                permit .*
        }
        cmd = interface {
                permit .*
        }
        cmd =  switchport  {
                permit .*
        }
```

2(3)

```
                cmd = description {
                        permit .*
                }
                cmd = no {
                        permit shutdown
                }
                service = junos-exec {
                        local-user-name = remote
                }
}

group = viewer {
                service = exec {
                        priv-lvl = 7
                }
                cmd = username {
                        deny .*
                }
                cmd = enable {
                        deny .*
                }
                cmd = show {
                        permit .*
                }
                cmd = exit {
                        permit .*
                }
                cmd = configure {
                        deny .*
                }
                cmd = interface {
                        deny .*
                }
                cmd =  switchport  {
```

3(3)

```
                deny .*
        }
        cmd = description {
                deny .*
        }
        cmd = no {
                deny shutdown
        }
}

#USERS
user = view { login = cleartext "view" member = viewer }

#ADMINS
user = testi { login = PAM member = admin }
```

Liite 3. SQL-moduulin asetukset 1(1)

```
sql {
        driver = "rlm_sql_mysql"
        dialect = "mysql"
        server = "localhost"
        port = 3306
        login = "root"
        password = "system"
        radius_db = "radius"
        acct_table1 = "radacct"
        acct_table2 = "radacct"
        postauth_table = "radpostauth"
        authcheck_table = "radcheck"
        groupcheck_table = "radgroupcheck"
        authreply_table = "radreply"
        groupreply_table = "radgroupreply"
        usergroup_table = "radusergroup"
        delete_stale_sessions = yes
        pool {
                start = 5
                min = 4
                max = ${thread[pool].max_servers}
                spare = 10
                uses = 0
                lifetime = 0
                idle_timeout = 60
        }
        read_clients = yes
        client_table = "nas"
        $INCLUDE ${modconfdir}/${.:name}/main/${dialect}/queries.conf
}
```

Liite 4. PAM-moduulin asetukset 1(1)

```
pam {
        pam_auth = radiusd
}
```

Liite 5. RADIUS-asiakkaiden asetukset 1(2)

```
client localhost {
        ipaddr = 127.0.0.1
        proto = *
        secret = testing123
        require_message_authenticator = no
        nas_type        = other
        limit {
                max_connections = 16
                lifetime = 0
                idle_timeout = 30
        }
}

client juniper{
        ipaddr = 10.10.10.10/31
        proto = *
        secret = Salasana
        require_message_authenticator = no
        nas_type        = other
        limit {
                max_connections = 16
                lifetime = 0
                idle_timeout = 30
        }
}
client quantadell{
        ipaddr = 10.10.11.0/24
proto = *
        secret = Salasana
        require_message_authenticator = no
        nas_type        = other
        limit {
                max_connections = 16
```

2(2)

```
                lifetime = 0
                idle_timeout = 30
        }
}

client localhost_ipv6 {
        ipv6addr        = ::1
        secret                  = testing123
}
```

Liite 6. PAM asetukset 1(1)

```
127.0.0.1  secret          1
```

Liite 7. RADIUS asetukset 1(2)

```
prefix = /usr
exec_prefix = /usr
sysconfdir = /etc
localstatedir = /var
sbindir = /usr/sbin
logdir = ${localstatedir}/log/radius
raddbdir = ${sysconfdir}/raddb
radacctdir = ${logdir}/radacct
name = radiusd
confdir = ${raddbdir}
modconfdir = ${confdir}/mods-config
certdir = ${confdir}/certs
cadir   = ${confdir}/certs
run_dir = ${localstatedir}/run/${name}
db_dir = ${localstatedir}/lib/radiusd
libdir = /usr/lib64/freeradius
pidfile = ${run_dir}/${name}.pid
max_request_time = 30
cleanup_delay = 5
max_requests = 1024
hostname_lookups = no

log {
        destination = files
        colourise = yes
        file = ${logdir}/radius.log
        syslog_facility = daemon
        stripped_names = no
        auth = yes
        auth_badpass = no
        auth_goodpass = no
        msg_denied = "You are already logged in - access denied"
}
```

```
checkrad = ${sbindir}/checkrad

security {
        user = root
        group = root
        allow_core_dumps = no
        max_attributes = 200
        reject_delay = 1
        status_server = yes
}

proxy_requests  = yes
$INCLUDE proxy.conf
$INCLUDE clients.conf

thread pool {
        start_servers = 5
        max_servers = 32
        min_spare_servers = 3
        max_spare_servers = 10
        max_requests_per_server = 0
        auto_limit_acct = no
}

modules {
        $INCLUDE mods-enabled/
}

policy {
        $INCLUDE policy.d/
        $INCLUDE sites-enabled/
}

$INCLUDE sites-enabled/
```

Liite 8. RADIUS-käyttäjien asetukset 1(1)

```
DEFAULT         Framed-Protocol == PPP
        Framed-Protocol = PPP,
        Framed-Compression = Van-Jacobson-TCP-IP

DEFAULT         Hint == "CSLIP"
        Framed-Protocol = SLIP,
        Framed-Compression = Van-Jacobson-TCP-IP

DEFAULT         Hint == "SLIP"
        Framed-Protocol = SLIP

#Users:
DEFAULT Auth-Type == Accept
view Cleartext-Password := "view"
        Cisco-AVPair = "shell:priv-lvl=7"
```