Joonas Utriainen

Software-Defined Access
Cisco DNA Center

Opinnäytetyö
Tieto- ja viestintätekniikka

2019

Kaakkois-Suomen
ammattikorkeakoulu

Tekijä/Tekijät	Tutkinto	Aika
Joonas Utriainen	Insinööri (AMK)	Joulukuu 2019
Opinnäytetyön nimi Software-Defined Access Cisco DNA Center		65 sivua 7 liitesivua
Toimeksiantaja Kymijoen ICT		
Ohjaaja Vesa Kankare		

Tiivistelmä

Tämän opinnäytetyön tavoitteena on tutustua SDA-tekniikkaan, sen tuomiin mahdollisiin hyötyihin ja haittoihin, sekä suunnitella ja toteuttaa tilaajayritykselle soveltuvuusselvitys SDA-verkkoratkaisun käyttöönotosta Cisco DNA Center -alustaa käyttäen.

SDA-verkkoratkaisun tarkoituksena on yksinkertaistaa verkkoinfrastruktuuria, sen suunnittelua, helpottaa käyttöönottoa ja parantaa verkon tietoturvallisuutta. Cisco DNA Center on SDA-ratkaisun komponentti, joka mahdollistaa keskitetyn verkon hallinnan, käyttöoikeuspolitiikkojen suunnittelun ja verkon suunnittelun ja toteutuksen. Työn yhtenä tavoitteena oli toteuttaa ohjelmallisesti toteutettu, Active Directory -käyttäjäryhmien mukaan segmentoitu verkko ja luoda segmenteille käyttöoikeuspolitiikat. Verkon suunnittelussa otettiin myös huomioon laitteet ja käyttäjät, jotka eivät kuulu mihinkään käyttäjäryhmään, mutta vaativat verkkoon pääsyn.

Työ aloitettiin tutkimalla SDA-tekniikkaan liittyviä komponentteja, protokollia ja keskeisiä tekniikoita, kuten Cisco ISE, Cisco DNA Center, VXLAN, LISP ja 802.1X. Teoriaosuudessa tutustuttiin SDA-verkkoratkaisun komponentteihin, protokolliin ja tekniikoihin. Työn käytännön osuus koostui SDA-verkon suunnittelusta ja toteutuksesta. Toteutusvaiheessa kokeiltiin myös käyttäjäsegmentointia ja käyttöpolitiikkoja. Käyttäjät ohjattiin käyttäjäryhmän mukaan omiin lähiverkkoihinsa ja niiden sisällä sovellettiin suunniteltuja käyttöoikeuspolitiikkoja.

Työn lopputuloksena oli onnistunut prototyyppi SDA-ratkaisun suunnittelusta ja käyttöönotosta.

Asiasanat

SDA, tietoverkot, pääsynhallinta

South-Eastern Finland University of Applied Sciences

Author (authors)	Degree	Time
Joonas Utriainen	Bachelor of Engineering	December 2019

Thesis title

Software-Defined Access
Cisco DNA Center

65 pages
7 pages of appendices

Commissioned by

Kymijoen ICT

Supervisor

Vesa Kankare

Abstract

The purpose of this thesis was to become familiar with SDA technology, its potential benefits and drawbacks, and to design and implement a feasibility study for the subscriber company to deploy an SDA network solution using the Cisco DNA Center platform.

The purpose of the SDA network solution is to simplify the network infrastructure, design it, facilitate deployment and improve network information security. Cisco DNA Center is a component of the SDA solution that enables centralized network management, design of access policies, and network design and implementation. One of the goals of the work was to implement a programmatically implemented network segmented by Active Directory user groups and create access policies for the segments. The design of the network also considered devices and users who fell outside any user group but require network access.

The work began by exploring SDA-related components, protocols, and key technologies such as Cisco ISE, Cisco DNA Center, VXLAN, LISP, and 802.1X. The theoretical part introduced the components, protocols and techniques of the SDA network solution. The practical part of the work consisted of the design and implementation of the SDA network. In the implementation phase, user segmentation and usage policies were also tested. According to their designated user group, users were directed to their local area networks and within them, planned access policies were applied.

The work resulted in a successful prototype for the design and deployment of the SDA solution.

Keywords

SDA, data networks, access control

SISÄLLYS

KÄSITTEET JA LYHENTEET

802.1X	IEE 802.1X-standardinumero porttikohtaiselle todennukselle.
AAA	Käyttäjän tunnistamiseen käytettävä kehysmalli, joka tulee sanoista Authentication, Accounting ja Authorization
Cisco DNA Center	Cisco Digital Network Architecture Center. SDA-ratkaisun keskeinen komponentti verkon suunnitteluun ja toteutukseen.
Cisco ISE	Cisco Identity Services Engine. Ohjelmisto keskitettyyn verkon pääsynhallintaan.
CTS	Cisco TrustSec. Tekniikka käyttöoikeuspolitiikkojen määrittämiseen SGT-tagein VLAN:in sijaan.
DNS	Domain Name System. Verkon nimipalvelu.
EAP	Extensible Authentication Protocol. Käyttäjien tunnistamiseen liittyvän tiedon siirtoprotokolla.
EAPoL	EAP over LAN. 802.1X-standardin käyttämä tiedonsiirtoprotokolla.
EID	Endpoint Identifier. Päätelaitteen osoite LISP-protokollaa käytettäessä.
ETR	Egress Tunnel Router. LISP-reitityksen komponentti.
GPO	Group Policy Option.
HSRP	Hot Standby Router Protocol. Protokolla verkon yhdyskäytävän konvergenssiin.
HTDB	Host tracking database. Tietokanta, johon EID-osoitteet rekisteröidään.
IETF	Internet Engineerin Task Force. Organisaatio, joka kehittää Internet-standardeja.
IGP	Interior Gateway Protocol. Valittu reititysprotokolla, jota verkko käyttää reittien mainostamiseen.
IP	Internet Protocol. TCP/IP-viitemallin protokolla tiedon siirtoon.
IPAM	IP Address Management. IP-osoitteiden ylläpitoon tarkoitettu ohjelmisto.

IPv4	Internet Protocol version 4. Neljäs malli IP-protokollasta.
IPv6	Internet Protocol version 6. Kuudes malli IP-protokollasta.
IS-IS	Intermediate System to Intermediate System. Reititysprotokolla.
IT	Information Techonology. Informaatioteknologia.
ITR	Ingress Tunnel Router. LISP-reitityksen komponentti.
ITSM	IT Service Management. IT-palveluihin liittyvä suunnittelu, tuki ja ylläpitojärjestelmä.
LAN	Local Area Network. Lähiverkko.
LISP	Locator/ID Separator Protocol. Reititysprotokolla.
MAC	Media Access Control. TCP/IP-viitemallin siirtoyhteyskerroksen alikerros fyysisten linkkien osoitteistukseen.
MR	Map-Resolver. LISP-reitityksen komponentti.
MS	Map-Server. LISP-reitityksen komponentti.
MTU	Maximum Transmission Unit. Suurin kehyskoko, joka voidaan kuljettaa verkkokerroksella.
NVE	Network Virtualization Edge. Verkon virtuaalinen reunalaite.
NTP	Network Time Protocol. Protokolla kellonajan synkronointiin.
OSPF	Open Shortest Path First. Reititysprotokolla.
PETR	Proxy Egress Tunnel Router. LISP-reitityksen komponentti.
PITR	Proxy Ingress Tunnel Router. LISP-reitityksen komponentti.
RADIUS	Remote Authentication Dial-In User Service. Todennusprotokolla, jonka avulla siirretään kirjautumistietoja palvelimen ja RADIUS-asiakkaan välillä.
RLOC	Routing Locator. Reitittimen osoite LISP-protokollaa käytettäessä liikenteeseen SDA-verkosta ulos.
SDA	Software-Defined Access. Ohjelmistopohjainen verkon pääsynhallinta.

SGT	Scalable Group Tag. Käyttäjän tunnistamiseen käytettävä merkintä.
STP	Spanning Tree Protocol. Verkkoprotokolla, joka mahdollistaa silmukkavapaan topologian.
SXP	SGT Exchange Protocol. IETF-protokollaluonnos SGT-tagien siirtoon.
UDP	User Datagram Protocol. Tiedonsiirtoprotokolla.
VLAN	Virtual Local Area Network. Virtuaalinen lähiverkko.
VM	Virtual Machine. Virtualisoitu tietokone.
VN	Virtual Network.
VNI	VXLAN Network Identifier. VXLAN-segmentin tunnisteesta käytetty termi.
VNID	Virtual Network Identifier. VXLAN-segmentin tunniste.
VRF	Virtual Routing and Forwarding. Virtuaalinen reititystaulu.
VRRP	Virtual Router Redundancy Protocol. Protokolla verkon yhdyskäytävän konvergenssiin.
VTEP	VXLAN Tunnel End Point. VXLAN tunnelin päätepiste.
VTP	VLAN Trunking Protocol. VLAN-tiedonsiirtoprotokolla.
VXLAN	Virtual Extensible LAN. Verkon virtualisointitekniikka.
VXLAN-GBP	VXLAN Group Based Policy. Tiedonsiirrossa käytettävä otsikko.
VXLAN-GPO	VXLAN Group Policy Option. VXLAN-laajennus.
WAN	Wide Area Network. Useita lähiverkkoja yhdistävä laaja verkko, kuten Internet.
WLAN	Wireless Local Area Network. Langaton lähiverkko.

1 JOHDANTO

Digitaalisten tietoverkkojen jatkuva laajentuminen, niihin liitettyjen erilaisten laitteiden määrien kasvaminen ja käyttäjämäärien lisääntyminen luovat nykymaailmassa tarpeen uusille tavoille toteuttaa tietoverkkoja. Perinteisin menetelmin nykyisin toteutetut tietoverkot ovat usein manuaalisesti konfiguroitavia ja voivat sisältää useita komponentteja eri valmistajilta. Näin ollen tietoverkkojen ylläpito on hidasta ja virheille altistuvaa.

SDA (*Software-Defined Access*) on kehitys perinteisten kampusverkkojen lähiverkkosuunnittelusta, joka mahdollistaa keskitetyn verkon hallinnan ja virhevapaan ylläpidon. SDA tuo mukanaan myös mahdollisuuden automatisoituun verkon päästä-päähän segmentointiin, verkon analytiikkaan ja loppukäyttäjän liikkuvuuteen identiteetin ja paikan erottamisen avulla.

Tämän opinnäytetyön toimeksiantajana toimi Kymijoen ICT, joka toimii Kymenlaakson kuntien maakunnallisena palveluntuottajana ja tarjoaa monipuolisia ICT-ratkaisuja yritysten, maakuntien, kaupunkikonsernien ja niiden sidosorganisaatioiden tarpeisiin.

1.1 Opinnäytetyön tavoitteet

Opinnäytetyön tavoitteena oli luoda uutta tietoa SDA-ratkaisusta toimeksiantajalle, sillä heillä ei ollut aiempaa kokemusta SDA-verkkoratkaisusta tai tietoa sen soveltuvuudesta yrityksen käyttöön. Ciscon DNA Center SDA-ratkaisua oli aiemmin markkinoitu yritykselle, mutta toimeksiantajaa kiinnosti, millä tavalla SDA-ratkaisun avulla toteutettu kampusverkko eroaisi tavallisin tavoin toteutetusta kampusverkosta ja voisiko SDA-kampusverkkoratkaisusta olla hyötyä yritykselle tulevaisuudessa.

Työn varsinaisena tutkimusongelmana oli selvittää, millä tavoin SDA-ratkaisu toimii, mitä komponentteja sen implementointi vaatii ja mistä vaiheista sen käyttöönotto koostuu. Tutkimusongelma voidaan laajemmin määritellä seuraavin tutkimuskysymyksin:

1. Mikä on SDA-ratkaisu?

2. Miten Cisco DNA Center SDA-ratkaisu toimii?
3. Mitä vaiheita SDA-ratkaisun käyttöönotto vaatii?
4. Mitä hyötyjä tai haittoja SDA-ratkaisusta on?
5. Millä tavoin SDA-ratkaisua voitaisiin hyödyntää tulevaisuudessa?

1.2 Tutkimusmenetelmän valinta

Opinnäytetyö toteutettiin tutkimuksellisena kehittämishankkeena, sillä se vastasi menetelmänä parhaiten työn sisältöä ja tarkoitusperää. Kanasen mukaan (2015, 54–61) tutkimuksellisessa kehittämishankkeessa yhdistyvät kvalitatiiviset sekä kvantitatiiviset tutkimusmenetelmät ja Salosen mukaan (2013, 25) ”kehittämishankkeen tuloksena syntyy tuotos, joka sisältää uuden tiedon lisäksi palvelun, tuotteen, oppaan, mallin, toimintatavan tai minkä tahansa sellaisen innovaation, joka on aikaisempaa parempi tai kokonaan uusi”.

Kehittämishankkeen tavoitteena oli luoda yritykselle parempi käsitys SDA-teknologian tuomista hyödyistä ja haitoista ja sitä varten perustettiin laboratorioympäristö, jossa SDA-teknologiaan voitiin tutustua tarkemmin käytännön kokeilla.

2 CISCO SOFTWARE-DEFINED ACCESS -VERKKOARKKITEHTUURI

Perinteistä kampusverkkoa suunnitellessa tulee ottaa huomioon suuri määrä komponentteja ja tekniikoita. Suunnittelun lisäksi perinteinen kampusverkko on hankala ylläpitää ja aikaa vievä laajentaa. Voidaan ajatella esimerkiksi, että yrityksen olemassa oleva, viidenkymmenen kiinteistön laajuinen verkko tulisi laajentaa uuteen kiinteistöön, johon sijoitettaisiin 10 kytkintä. Tämä tarkoittaisi, että kytkinten konfigurointiin, testaukseen ja paikan päällä asennettaessa vielä mahdollisesti vianselvitykseen kuluisi suuri määrä aikaa. Näin ollen yrityksessä työskentelevät henkilöt eivät voisi keskittyä tärkeimpään tehtäväänsä: verkon ylläpitoon. (Hill ym. s.a., 30.) Verkon suunnittelun ja toteuttamisen helpottamiseksi sekä ylläpidon yksinkertaistamiseksi Cisco Systems kehitti SDA-ratkaisun.

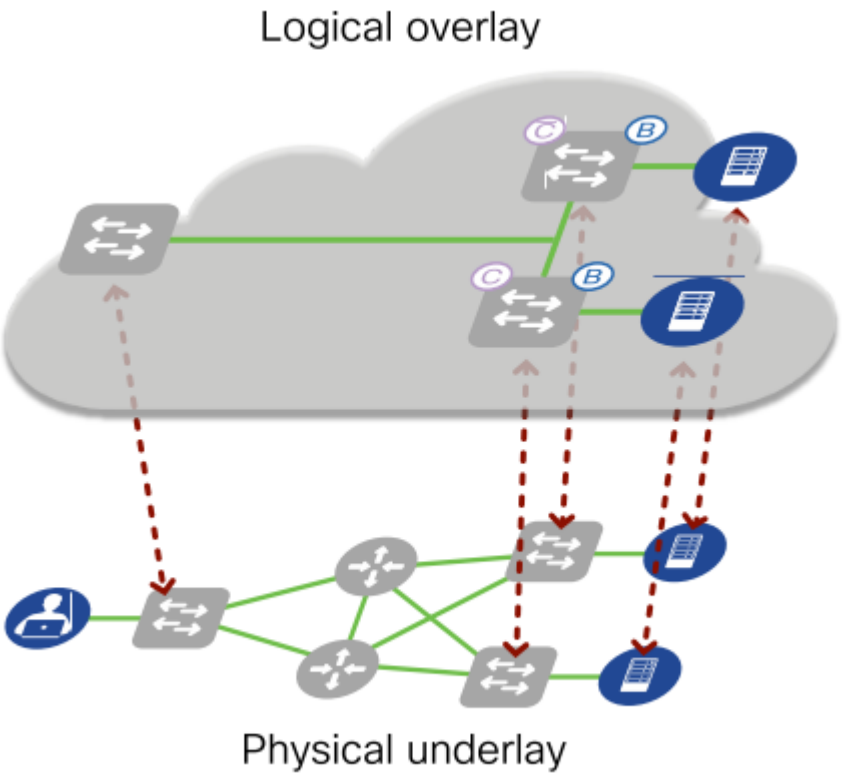
Ciscon SDA-ratkaisu on ohjelmoitava verkkoarkkitehtuuri, joka tarjoaa ohjelmistopohjaisen politiikan ja segmentoinnin verkon reunalta sovelluksiin. SDA on Ciscon ratkaisussa toteutettu käyttäen Cisco Digital Network Architecture

Centeriä (*Cisco DNA Center, DNA*), joka tarjoaa verkon osille suunnitteluasetukset, politiikkamääritykset ja automaattisen provisioinnin, sekä analytiikan älykkäille langallisille ja langattomille verkoille. (Hill ym. s.a., 30.)

Yritysten verkkoarkkitehtuurissa verkko voi levittyä useiden toimialueiden, sijaintien ja toimipisteiden, kuten kampusten ja etätoimipisteiden välille sisältäen useita laitteita, palveluita ja politiikkoja. Ciscon SDA-ratkaisu tarjoaa päästä päähän -arkkitehtuurin, joka varmistaa yhdenmukaisuuden eri sijaintien välillä niin yhdistettävyyden, segmentoinnin kuin politiikkojen osalta. Ciscon SDA-ratkaisu voidaan kuvailla kahdella tasolla: SDA-verkko ja
Cisco DNA Center. (Hill ym. s.a., 30.)

2.1 SDA-verkko

SDA-verkko on verkkotopologia, joka koostuu kahdesta verkkokerroksesta, alusverkko (underlay) ja päällysverkko (overlay) (kuva 1), joilla molemmilla on SDA-verkossa oma tehtävänsä. Alusverkon tehtävänä on tarjota fyysinen verkkoyhteyskerros ja toimittaa päällysverkon paketteja verkon aktiivilaitteiden välillä. Päällysverkon tehtävänä on vastaavasti tarjota looginen tunneloitu verkko, joka virtuaalisesti yhdistää kaikki verkkolaitteet keskenään muodostaen SDA-verkon. (Cisco, 2018c.) SDA-verkolla tarkoitetaan useasti yhdistetytsi alus- ja päällysverkkokerroksia (Hill ym. s.a., 31).



Kuva 1. Fyysinen alusverkko ja looginen päällysverkko kuvattuna (Hill ym. s.a., 31)

2.1.1 Alusverkko

Alusverkko on SDA-verkon kerros, joka koostuu Layer 3-reitityksestä (L3), fyysisistä verkkolaitteista, kuten reitittimistä, kytkimistä ja langattoman WLAN-verkon (*Wireless Local Area Network*) kontrollereista (Microchip Technology, Inc. 2019). Alusverkon tarkoitus on toteuttaa yksinkertainen ja luotettava, helposti skaalautuva perusta verkon laitteiden väliseen kommunikointiin. Näin ollen alusverkkoa ei käytetä päätelaitteiden välisessä kommunikoinnissa, vaan niiden kommunikointi tapahtuu päällysverkon avulla.

Kaikkien alusverkon laitteiden tulee pystyä luomaan IPv4-yhteys (*Internet Protocol version 4*) toistensa välille, joka tarkoittaa sitä, että pystytään hyödyntämään jo mahdollisesti olemassa olevaa IPv4-verkkoa alusverkkona. Vaikkakin alusverkossa voidaan käyttää mitä tahansa topologiaa ja reititysprotokollaa, niin silti suositellaan käytettäväksi hyvin suunniteltua L3-yhteystopologiaa, joka takaa johdonmukaisen suorituskyvyn, skaalautuvuuden ja korkean saavutettavuuden.

Käytettäessä reititettyä L3-yhteystopologiaa voidaan alusverkossa eliminoida tarve seuraaville verkon protokollille: STP (*Spanning Tree Protocol*), VTP (*VLAN Trunking Protocol*), HSRP (*Hot Standby Router Protocol*), VRRP (*Virtual Router Redundancy Protocol*) ja muille vastaaville. Näin ollen verkko yksinkertaistuu huomattavasti samalla kuitenkin lisäten verkon joustavuutta ja parantaen vikasietoisuutta.

Ciscon SDA-ratkaisussa alusverkon reititysprotokollina voidaan käyttää joko IS-IS- tai OSPF-protokollia (*Intermediate System to Intermediate System, Open Shortest Path First*), niiden ollessa linkkitilaprotokollia (Oran 1990; Mou 1998). Linkkitilaprotokollan käyttöön alusverkossa on päädytty siksi, että IS-IS- ja OSPF-protokollat ovat yritysverkoissa käytetyimmät reititysprotokollat ja ne tuovat mukanaan muun muassa seuraavat hyödyt:

1. Linkkitilaprotokollat eivät mainosta kokonaisia reititystauluja, vaan sen sijaan mainostavat tietoa verkon topologiasta, jotta kaikilla reitittimillä on sama topologiatietokanta alueen sisällä.
2. Linkkitilaprotokollat tuottavat monitasoisen hierarkian, jotta tietyn alueen sisällä määriteltyjä reittejä ei mainosteta alueen ulkopuolelle.
3. Linkkitilaprotokollat käyttävät algoritmia etsiessään lyhyintä tietä jokaiseen topologian solmukohtaan.
4. Linkkitilaprotokollat konvergoituvat huomattavasti nopeammin, kuin pituusvektorireititysprotokollat (*Distance Vector Routing Protocol*). (Hill ym. s.a., 32–33).

2.1.2 Päällysverkko

SDA-verkon päällysverkkokerros on looginen, fyysisen alusverkon päälle virtualisoitu verkkotopologia, joka tarjoaa infrastruktuurin politiikkoihin perustuviin segmentointirakenteisiin, mobiliteetin mahdollistaviin dynaamisiin verkkopalveluihin ja parannettuun tietoturvaan. Päällysverkkokerros on täysin automatisoitava ratkaisu riippumatta siitä, minkälaisen alusverkon päälle verkkokerros on rakennettu. SDA-verkkojen lisäksi voidaan rakentaa ja ylläpitää vastaavanlaisia verkkoja, mutta niissä tapauksissa verkkoratkaisu on SDA:n sijaan enemmänkin kampusverkko, joka käyttää alus- ja päällysverkkomallia ilman automatisointia ja keskitettyä hallintaa.

Ciscon SDA-ratkaisussa päällysverkko perustuu LISP- ja VXLAN -protokolliin (*Locator ID Separator Protocol*, *Virtual eXtensible Local Area Network*) ja vaatii toimiakseen seuraavat komponentit:

1. SDA-verkon datataso
2. SDA-verkon kontrollitaso
3. SDA-verkon politiikkataso. (Cisco 2018c.)

2.1.3 SDA-verkon kontrollitaso

SDA-verkon kontrollitasolla käytetään LISP-reititysprotokollaa, joka perustuu yksinkertaiseen EID-osoitteen ja RLOC-osoitteen yhdistävään malliin, jossa erotellaan identiteetti (osoite) sijainnista (lähin reititin). Ciscon SDA-ratkaisumallissa tavalliseen LISP-määrittelyyn on lisätty joitain parannuksia, kuten anycast gateway- ja Virtual Network -mahdollisuus (*VN*). (Cisco 2018c.)

LISP-reititysprotokolla yksinkertaistaa huomattavasti tavallisia reititysverkkoja poistamalla tarpeen, jossa jokainen reititin käsittelee kaikki mahdolliset IP-kohdeosoitteet ja reitit. LISP-reitityksessä tämä toteutetaan siirtämällä kohdeosoitteen tieto keskitettyyn tietokantaan, joka mahdollistaa sen, että jokainen reititin hallitsee paikallisia reititystietojaan ja kyselee ulkopuolisten osoitteiden reitit keskustietokannasta. Tämä tekniikka mahdollistaa kustannustehokkaan ratkaisumallin prosessorin laskentatehon tarpeen vähentyessä, kun reitittimen ei tarvitse ylläpitää suuria reititystauluja. (Cisco 2018c.)

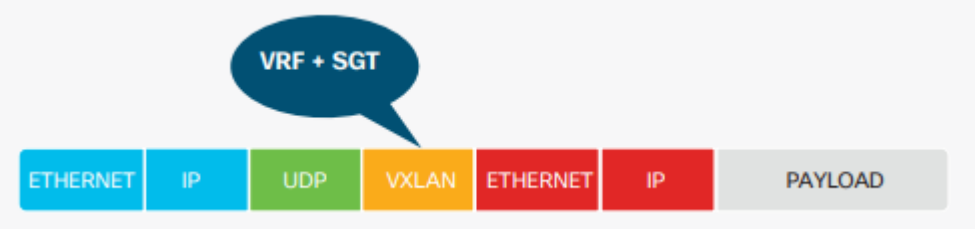
2.1.4 SDA-verkon datataso

SDA-verkon datatasolla käytetään usein IETF-standardiin (*Internet Engineering Task Force*) RFC-7348 (*Request for Comments*) perustuvaa VXLAN-enkapsulointia (Mahalingam ym. 2014). VXLAN on IP/UDP-protokollapohjainen (*Internet Protocol, User Datagram Protocol*), joka mahdollistaa pakettien kuljettamisen missä tahansa IP-pohjaisessa verkossa. Ciscon SDA-ratkaisussa datatasolla käytetään muunneltua VXLAN-enkapsulointia LISP-enkapsuloinnin sijaan. Muunneltu VXLAN-enkapsulointimalli mahdollistaa L2 Ethernet -kehyksen, VRF-tunnisteen (*Virtual Routing and Forwarding*) ja muun lisätiedon, kuten SGT-merkinnän kuljettamisen paketin sisällä (kuva 2). Muunneltu VXLAN-formaatti on kirjoitushetkellä IETF-luonnos, joka tunnetaan nimellä Group Policy Option (*GPO)* tai VXLAN-GPO. (Cisco 2018c.)

2.1.5 SDA-verkon politiikkataso

SDA-verkon politiikkataso perustuu CTS-tekniikkaan (*Cisco TrustSec*), joka mahdollistaa ohjelmistopohjaisen segmentoinnin, jossa politiikat ovat määritelty SGT-merkintöjen perusteella VLAN:ien sijaan. Cisco TrustSec käyttää SXP-protokollaa (*SGT Exchange Protocol*), joka on opinnäytetyön kirjoitushetkellä IETF-protokollaluonnos (Smith ym. 2019).

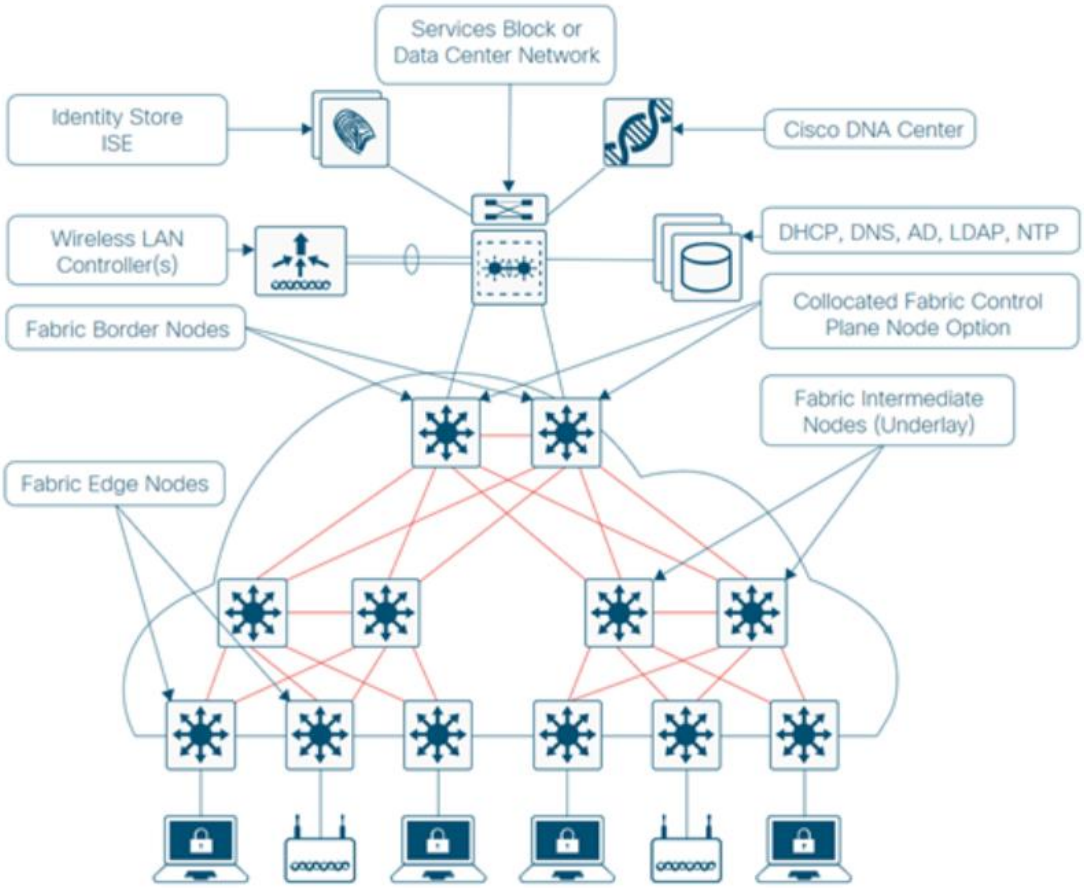
SGT on uniikki 16-bittinen ID-merkintä, joka on erillään verkko-osoitteesta. SGT-merkinnät mahdollistavat käyttäjän verkkopolitiikkojen, kuten tietoturvan, määrittelyn ainoastaan SGT-merkinnän perusteella välittämättä käyttäjän sijainnista. SGT-merkintä ja VN yhdistämällä mahdollistetaan kaksitasoisen käyttöoikeuspolitiikan luominen: käyttäjät voidaan sijoittaa tiettyyn VN:iin ja sen sisällä voidaan suorittaa SGT-merkintöihin perustuvaa segmentointia. (Cisco 2018c.)



Kuva 2. Muunneltu VXLAN-kehys LISP-paketissa (Cisco 2018c)

3 CISCO SDA-RATKAISUN KOMPONENTIT

Cisco SDA-ratkaisu koostuu useasta verkon ja infrastruktuurin komponentista (kuva 3), sekä keskeisistä teknologioista, joita käsitellään tässä luvussa.



Kuva 3. Cisco SDA-ratkaisun komponentit Cisco (2019d)

3.1 Verkon komponentit

3.1.1 Control plane node

SDA-verkossa kontrollitason tehtävänä on seurata kaikkia päätelaitteita SDA-verkon sisällä ja yhdistää päätelaitteet SDA-verkon solmupisteisiin samalla erottaen päätelaitteen IP- tai MAC-osoitteen verkon sijainnista (lähin reititin). Kontrollitaso perustuu Ciscon SDA-ratkaisussa LISP-protokollan MS- ja MR-komponentteihin (*Map-Server*, *Map-Resolver*), jotka on toteutettu saman solmupisteen sisällä. Kontrollitaso voidaan toteuttaa SDA-verkossa Border node -laitteessa tai se voidaan erottaa erilliseksi komponentikseen. Kontrollitason laitteita voi olla lankaverkon toteutuksessa SDA-verkon sisällä 2–6 kappaletta yhden sijaan verkon vikasietoisuuden lisäämiseksi. Jos verkkoon sijoitetaan useampi kuin yksi kontrollitaso, tulisi kontrollitasojen laitteiden olla toisiaan vastaavia tasaisen suorituskyvyn saavuttamisen kannalta, sillä Edge node -laitteet käyttävät kaikkia verkkoon määritettyjä kontrollitasoja ja rekisteröivät itsensä niihin.

Kontrollitasossa toteutetaan SDA-verkon seuraavat toiminnot:

1. Host tracking database (*HTDB*). HTDB:n tehtävä on ylläpitää keskitetysti SDA-verkon päätelaitteiden osoitteen ja Edge-laitteen yhdistävää tietokantaa.
2. Map-server. Map-serverin tehtävä on lisätä HTDB:hen SDA-verkon reunalta tulevien laitteiden rekisteröintiviestit.
3. Map-resolver. Map-resolverin tehtävä on vastata map query -viesteihin SDA-verkon reunalaitteilta ja yhdistää SDA-verkossa käytettävä RLOC-osoite (*Routing Locator*) päätelaitteen EID-osoitteeseen (*Endpoint Identifier*).

3.1.2 Border node

SDA-verkossa Border node -laitteen tarkoituksena on toimia yhdyskäytävänä SDA-verkon ja sen ulkopuolisen verkon välillä. Border node on vastuussa verkon virtualisoinnista ja SGT-merkintöjen (*Scalable Group Tag*) levittämisestä SDA-verkosta muualle tietoverkkoon.

Border node -laitteissa toteutetaan SDA-verkon seuraavat toiminnot:

1. EID-aliverkkojen (*Endpoint Identifier*) mainostus
2. SDA-verkon päätepiste
3. LISP-reititysinstanssin yhdistäminen VRF-reititystauluun (*Virtual Routing and Forwarding*)
4. Politiikkojen yhdistys. (Cisco 2019d.)

3.1.3 Intermediate node

Intermediate node on SDA-verkossa laite, joka toimii Border- ja Edge-laitteiden välillä ottamatta kuitenkaan osaa SDA-verkon toiminnallisuuksiin (VXLAN, LISP-kontrollitason viestit, SGT-tietoisuus). Intermediate node -laitteen tarkoituksena on siis vain välittää SDA-verkon L3-verkon IP-paketteja, joka asettaa vaatimuksen, että laitteen MTU (*Maximum Transfer Unit*) tulee asettaa vastaavaksi, kuin muissakin SDA-verkon verkon kytkimissä. (Cisco 2019d.) Intermediate node -laitetta voidaan verrata tavallisessa kampusverkossa toteutettuun jakelukerroskytkimeen (*Distribution Layer Switch*).

3.1.4 Edge node

SDA-verkon Edge node -laitteet ovat verrattavissa tavallisen kampusverkon pääsytason kytkimiin (*Access Layer Switch*) ja niihin kytketään SDA-verkkoon liittyvät päätelaitteet. Edge node -laitteissa toteutetaan tavallisen L3-tason pääsyn lisäksi SDA-verkon kannalta olennaiset seuraavat toiminnot:

1. Päätelaitteen rekisteröinti. Jokainen SDA-verkon Edge node -laite ylläpitää LISP-kontrollitason yhteyttä kaikkiin kontrollitason laitteisiin ja kun päätelaite tunnistetaan SDA-verkon reunalla, lisätään se paikalliseen EID-tietokantaan. Edge node -laite lähettää kontrollitason laitteelle map-register viestin, jonka avulla EID voidaan lisätä HTDB:en.
2. Käyttäjän yhdistäminen virtuaaliseen verkkoon. Päätelaitteet määritellään Edge node -laitteella verkkoon liittyessä VLAN:iin ja sille määritellään SGT-tunniste, jota voidaan käyttää segmentoinnissa ja politiikkojen määrittelyssä.
3. Anycast L3 -yhdyskäytävä. Mobiliteetin mahdollistamiseksi Edge node -laitteisiin määritellään yleinen anycast yhdyskäytävä, jolla on sama IP- ja MAC-osoite riippumatta Edge node -laitteesta.
4. LISP-edelleenlähetys. Tavallisesta reitityksestä poiketen Edge node -laite suorittaa kontrollitason laitteelle map-server-kyselyn, jonka vastauksena määritellään mikä RLOC-osoite yhdistetään päätelaitteen EID-osoitteeseen ja tätä tietoa käytetään liikenteen reitityksessä.
5. VXLAN-enkapsulointi ja dekapsulointi. SDA-verkon Edge node -laitteet käyttävät pääteosoitteeseen liitettyä RLOC-osoitetta enkapsuloidakseen liikenteen VXLAN-kehykseen. Vastaavasti VXLAN-enkapsuloitu liikenne vastaan ottavassa päässä dekapsuloidaan. Tämä mahdollistaa päätelaitteen liikkuvuuden, sillä sen ei tarvitse vaihtaa IP-osoitettaan, vaan reitityksessä käytettävä RLOC-osoite vaihtuu. (Cisco 2019d.)

3.2 Muut komponentit

3.2.1 Cisco Identity Services Engine

Cisco ISE (*Identity Services Engine*) on ohjelmisto, jonka tarkoituksena on toteuttaa keskitetysti verkon pääsynhallinta turvallisesti käyttäjille ja verkon laitteille. ISE mahdollistaa pääsynhallinnan lisäksi mm. verkon käytön seurannan ja sen avulla on mahdollista seurata mitä laitteita ja ohjelmia verkossa käytetään ja kuka niitä käyttää. ISE:ssä on myös sisään rakennettu AAA-palvelut (*Authentication, Authorization and Accounting*), jotka mahdollistavat RADIUS-protokollan (*Remote Authentication Dial-In User Service*) avulla verkon pääsynhallinnan ja käyttäjän tunnistuksen. Ciscon SDA-ratkaisussa ISE on tärkeä komponentti, sillä sen tarkoituksena on mm. rakentaa SDA-politiikat ja todentaa käyttäjä. Ilman ISE-komponenttia ei Ciscon SDA-ratkaisu olisi mahdollinen, sillä se on vahvasti sidottu Cisco DNA Centeriin, jolla toteutetaan SDA-ratkaisun konfigurointi. (Cisco 2019b.)

3.2.2 Cisco Digital Network Architecture Center

Cisco DNA Center on keskitetty operointialusta, joka tarjoaa yrityksen LAN-, WLAN- ja WAN-ympäristöjen (*Wide Area Network*) päästä päähän -automatisoinnin ja työkalut ympäristön ulkopuolisten ratkaisujen ja toimialueiden orkestrointiin. Cisco DNA Center tarjoaa verkon ylläpitäjille ja IT-operaattoreille (*Information Technology*) yhden käyttöliittymän, joka mahdollistaa verkon automaation ja hallinnan. Tämän lisäksi Cisco DNA Center tarjoaa intuitiivisen työnkulun, joka tekee helpoksi verkon ja käyttäjäpolitiikkojen suunnittelun ja proaktiivisen palveluiden seurannan. (Hill ym. s.a., 108).

Cisco DNA Center on suunniteltu skaalautumaan keskisuurten ja suurten yritysten verkkojen käyttöönottoihin. Cisco DNA Center koostuu verkon kontrollerista, johon on sisäänrakennettu automaatiovalmiudet ja tiedon analysointiin ja monitorointiin vaadittavat toiminnallisuudet, näin ollen tarjoten yhtenäisen alustan verkon toteutukseen, hallintaan ja vian etsintään. Cisco DNA Center on toteutettu käyttäen skaalautuvaa mikropalveluarkkitehtuuria. (Hill ym. s.a., 108).

3.3 Keskeiset teknologiat

3.3.1 VXLAN

Palvelinvirtualisoinnin lisääntyessä ja fyysisten palvelinten sisältäessä useita virtuaalisia koneita (*VM, Virtual Machine*) ovat vaatimukset fyysiselle verkkoinfrastruktuurille kasvaneet valtavasti. Fyysisessä palvelimessa jokainen VM voi sisältää yhden tai useita virtuaalisia verkkokortteja, joilla jokaisella on oma MAC-osoite (*Media Access Control*). Näin ollen satoja, tai jopa tuhansia virtuaalikoneita sisältävissä palvelinkeskuksissa Ethernet-verkon MAC-osoitetaulujen koot kasvavat valtaviksi. (Mahalingam ym. 2014.)

Tietoverkkoja voidaan jakaa useampiin L2-lähetystoimialueisiin (*Broadcast Domain*) VLAN-tekniikan (*Virtual Local Area Network*) avulla. Näin ollen suuren verkon segmentointi voi tarvita tuhansia VLAN-verkkoja. VLAN-verkkojen määrän ollessa rajoitettu 4 094 kappaleeseen, voi se joissain tapauksissa olla riittämätön vastaamaan verkon tarpeita. (Mahalingam ym. 2014.)

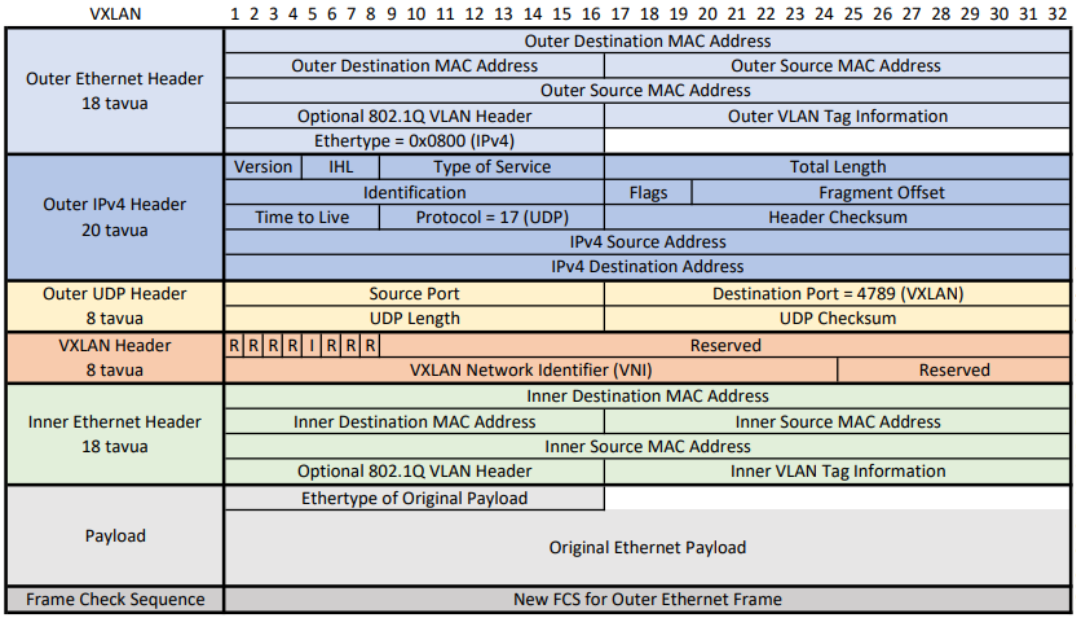
Perinteisesti suunnitellun, VLAN-virtualisoidun kampusverkon tarkoituksena on levittää L2-lähetystoimialuetta, jotta monessa toimipisteessä sijaitsevat päätelaitteet pystyisivät keskustelemaan keskenään. Jotta verkosta saataisiin vikasietoinen, asettaa se verkolle vaatimuksen pakettien kierron estämiseksi. Näin ollen verkkoon joudutaan implementoimaan tekniikoita kuten STP (*Spanning-Tree Protocol*). STP-protokolla estää toiminnallaan kierrot verkossa jättäen samalla kuitenkin useita verkkolinkkejä ”toimimattomaksi” kiertojen estämiseksi. Tämä aiheuttaa STP-protokollaa käyttävissä verkoissa lisäkustannuksia, koska osa porteista ei kuljeta liikennettä. (Mahalingam ym. 2014.)

Toisena lisäkustannuksena perinteisessä L2-verkossa voidaan pitää riittävän suorituskyvyn omaavien kytkinten hankintahintaa. Ennen virtualisointia, palvelinkeskuksissa esim. 48-porttisen kytkimen tuli pystyä oppimaan vain yksi MAC-osoite per liitetty palvelin. Nykyisin yhden palvelinkaapin sisältäessä esim. 40 tehokasta palvelinta, joista jokaisella virtualisoidaan 100 virtuaalikonetta, tuloksena on huomattavasti suurempi MAC-osoitetaulukko. Jos oletetaan, että palvelinkeskuksessa on esim. 10 kpl vastaavanlaisia palvelinkaappeja, niin kasvavat MAC-osoitetaulukot valtaviksi kytkinten ylläpitäessä myös muualta fyysisestä verkosta tulevan liikenteen MAC-osoitteita. Jos MAC-osoitetaulussa tapahtuu ylivuoto liiallisen MAC-osoitemäärän takia, voi kytkin lopettaa kokonaan uusien MAC-osoitteiden oppimisen, kunnes vanhat merkinnät vanhentuvat. Tämä taas tarkoittaa sitä, että verkossa syntyy sitä kuormittava liikennetulva verkkolaitteiden toimittaessa paketteja, jotka eivät löydä määränpäätään. (Mahalingam ym. 2014.)

Muun muassa edellä mainittujen ongelmien ehkäisyä varten kehitettiin vuonna 2014 VXLAN-enkapsulointi RFC-7348. VXLAN-enkapsuloinnin tarkoituksena on ”venyttää” L2-verkkoa reititetyn L3-verkon avulla, niin kutsuttuina päällysverkkoina. Päällysverkkoja käytetään niin ikään kuin loogisena ”tunneleina”, jonka sisällä kuljetetaan MAC-liikennettä päätelaitteilta toisille. VXLAN-enkapsuloinnissa jokaista päällysverkkoa kutsutaan segmentiksi, joissa vain segmenttiin kuuluvat päätelaitteet voivat keskustella keskenään. (Mahalingam ym. 2014.)

Verkossa VXLAN-segmentit tunnistetaan 24-bittisen segment ID:n mukaan, joita kutsutaan nimellä VNI (*VXLAN Network Identifier*). VNI:in ollessa 24-bittinen, mahdollistaa se jopa noin 16 miljoonaa (2^{24}) samanaikaista VXLAN-segmenttiä saman toimialueen sisällä. VNI on siis ulompi otsikko, johon enkapsuloidaan MAC-kehys päätelaitteelta. Juuri tämän enkapsuloinnin vuoksi VXLAN:ia kutsutaan tunneloinniksi. VXLAN-tunnelit ovat tilattomia, joten jokainen kehys enkapsuloidaan erikseen tiettyjen sääntöjen mukaan. Tunneleiden päätepisteitä kutsutaan nimellä VTEP (*VXLAN Tunnel End Point*) ja niiden tehtävä on kuljettaa, enkapsuloida ja purkaa VXLAN-liikenne päätelaitteelta toiselle UDP-protokollan avulla, päätelaitteen kuitenkaan olematta tietoinen VXLAN-tunnelista. Näin ollen VTEP-päätepisteinä voivat toimia vain fyysiset palvelimet tai verkon kytkimet. (Mahalingam ym. 2014.)

VXLAN-enkapsulointia käytettäessä tulee ottaa huomioon se, että VTEP:it eivät missään nimessä saa fragmentoida paketteja. Näin ollen VXLAN-kehyksen (kuva 4) pakettiin tuoma kehystyskuorma tulee ottaa huomioon, joka asettaa verkolle vaatimuksia MTU-arvon osalta. VXLAN-kehysmallia (kuva 4) tutkiessa huomataan, että se lisää pakettiin 8 tavun kokoisen kehyksen, jossa ensimmäinen tavu sisältää asetukset, seuraavat 3 tavua ovat varattu tulevaisuuden käyttökohteille, tämän jälkeen tulee 3 tavun kokoinen VNI ja viimeinen tavu on varattu tulevaisuuden käyttökohteille. Kun VXLAN-kehys lisätään alkuperäisen Ethernet-kehyksen alkuun ja sen lisäksi 8 tavun kokoinen UDP-kehys, 20 tavun kokoinen ulompi IPv4-kehys ja viimeiseksi ulompi 18 tavun kokoinen Ethernet-kehys, saadaan VXLAN-enkapsuloinnin tuomaksi kehystyskuormaksi 54 tavua. Jos käytössä on IPv6-protokolla IPv4:n sijaan, enkapsuloinnin tuoma kehystyskuorma kasvaa 74 tavuun, IPv6-kehyksen ollessa 40 tavun kokoinen. (Mahalingam ym. 2014.)



Kuva 4. VXLAN-kehysmalli (Mahalingam ym. 2014)

3.3.2 VXLAN-GPO

VXLAN-GPO (*VXLAN Group Policy Option*) on VXLAN-enkapsulointiprotokollan laajennus, joka mahdollistaa TSI-ryhmän (*Tenant System Interface*) tunnisteen lisäämisen VXLAN-otsikkoon politiikkojen käytön mahdollistamiseksi (Smith & Kreeger 2017, 2). Tämän työn osalta TSI-tunniste tarkoittaa samaa kuin SGT-tunniste.

Ryhmiin perustuva politiikkamalli määrittelee sovelluskeskeisen politiikkamallin, jossa sovelluksen yhdistämiseen liittyvät vaatimukset ovat irrallaan verkkotopologiasta. Kyseisessä mallissa TSI:t ovat liitettynä TSI-ryhmiin, jossa jokainen TSI-ryhmään kuuluva TSI jakaa samat verkkopolitiikat ja vaatimukset. Verkkopolitiikat määritellään lähde- ja kohde-TSI-ryhmien välille ja ne otetaan käyttöön, kun TSI liittyy verkkoon. (Smith & Kreeger 2017, 2.)

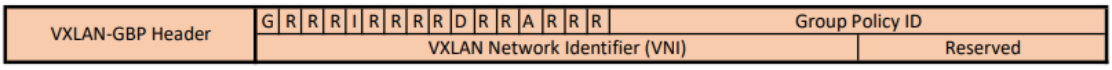
Useassa tilanteessa TSI-TSI-ryhmien määritykset ovat vain verkon virtuaalisen reunalaitteen (NVE, *Network Virtualization Edge*) tiedossa, johon kyseinen TSI on liitettynä. Tämä tarkoittaa sitä, että paketin kohteen TSI-ryhmä ei välttämättä ole tiedossa paketin saapuessa NVE:lle, jossa kohteen määränpää liitetään pakettiin. Näin ollen on tärkeää, että lähteen TSI-ryhmän tieto säilytetään paketissa, jotta siihen voidaan paketin lähtiessä NVE:ltä määritellä politiikat. (Smith & Kreeger 2017, 2.)

VXLAN-GPO-laajennuksessa käytetään VXLAN-GBP-otsikkoa (*VXLAN Group Based Policy*) (kuva 5), jossa on määritelty seuraavat bitit perinteisen VXLAN-otsikon lisäksi:

1. G-bitti. Otsikon ensimmäinen bitti (bitti numero 0) määrittelee laajennuksen käytön.
   Bitin arvon ollessa 1, määritellään, että TSI-ryhmän tieto, joka sijaitsee otsikon Group Policy ID -kentässä, kuljetetaan paketissa. Jos bitin arvoksi on määritelty 0, niin Group Policy ID -tietoa ei kuljeteta paketissa.
2. D-bitti. Otsikon yhdeksäs bitti on määritelty ”Don’t learn” -bitiksi. Bitin arvon ollessa 1 ei paketin lähettävä VTEP opettele enkapsuloidun kehyksen lähdeosoitetta.
3. A-bitti. Otsikon 12. bitti on määritelty ”Policy Applied”-bitiksi ja se määritellään A-bitiksi vain, jos G-bitin arvo on 1.

   A-bitin arvon ollessa 1 se merkitsee, että pakettiin on jo määritelty ryhmäpolitiikka. Politiikkaa ei tule missään nimessä lisätä, jos A bitin arvo on 1.

   A-bitin arvon ollessa 0 se merkitsee, että pakettiin ei ole lisätty ryhmäpolitiikkaa. Jos A-bitin arvo on 0 ja ryhmä on määritelty, tulee pakettiin lisätä ryhmäpolitiikat ja asettaa A-bitin arvoksi 1.
4. Group Policy ID. Otsikossa oleva 16-bittinen Group Policy ID määrittelee lähdeosoitteen TSI-ryhmän. (Smith & Kreeger 2017, 2.)



Kuva 5. VXLAN-GBP-kehys (Smith & Kreeger 2017, 2)

3.3.3 LISP

LISP (*Locator/ID Separator Protocol*) on verkkojen skaalautuvuutta parantava reititysprotokolla, jonka konseptina on erottaa paikka ja identiteetti toisistaan käyttäen L3-reitityksessä kahta nimiavaruutta: EID *(Endpoint Identifier*) ja RLOC (*Routing Locator*). LISP-reititysprotokollan tehtävänä on määrittää yhteys nimiavaruuksien välille enkapsuloimalla EID-osoitteilta tuleva liikenne niin, että paketti kuljetetaan verkkoinfrastruktuurissa käyttäen RLOC-osoitetta. (Farinacci ym. 2013.)

LISP-reitityksessä skaalautuvuus on saavutettu korvaamalla IP-osoitteet EID- ja RLOC-osoitteilla. EID- ja RLOC-osoitteet ovat identtisiä IP-osoitteiden kanssa, mutta semantiikka, miten niitä käytetään, eroaa. (Farinacci ym. 2013.)

RLOC-osoitteet ovat reititettäviä, topologisesti verkon liityntäpisteisiin määriteltyjä osoitteita ja niitä käytetään pakettien reitittämiseen verkossa. EID-osoitteet taas puolestaan ovat verkkotopologiasta riippumattomia ei globaalisti reititettäviä osoitteita, joita käytetään päätelaitteiden numeroinnissa (Farinacci ym. 2013).

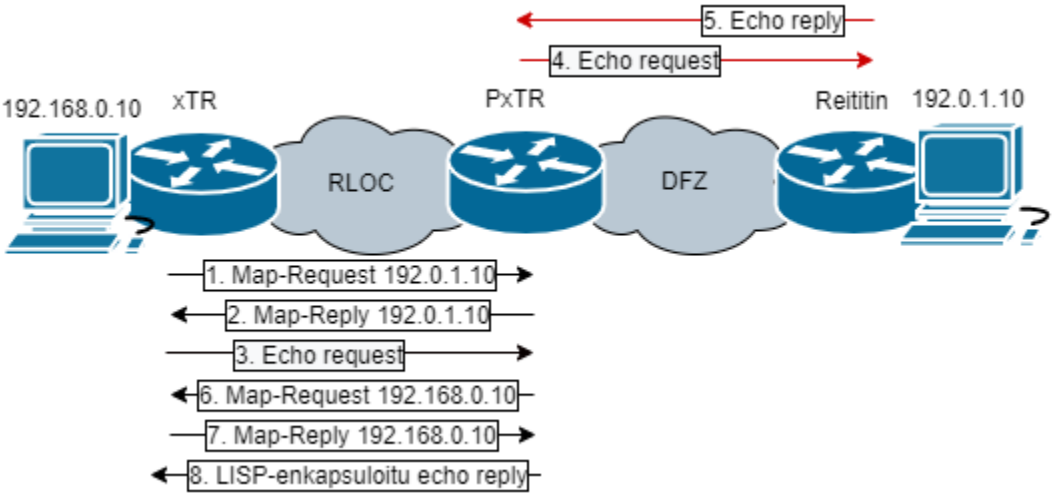
Yksi LISP-reitityksen avainkonsepteista on se, että päätelaitteet toimivat samalla tavalla kuin perinteisessä reitityksessä: IP-osoitteet, joita päätelaitteet käyttävät verkossa kommunikointiin (EID-osoite) eivät muutu ja reitittimet jatkavat pakettien ohjaamista kohteen IP-osoitteen perusteella olematta tietoisia LISP-reitityksestä. Toinen LISP-reitityksen konsepteista on ns. ”tunnelireititin” (*Tunnel Router*), joka lisää päätelaitteelta tulevaan pakettiin LISP-kehyksen ja näin uloimmassa kehyksessä IP-osoitteena toimii RLOC-osoite. Vastaavasti paketin saapuessa tunnelin toisessa päässä olevalle reitittimelle, puretaan ulompi kehys ja reititystä jatketaan normaalisti EID-osoitteen perusteella. Tunnelireitittimiä kutsutaan nimillä ITR (*Ingress Tunnel Router*) ja ETR (*Egress Tunnel Router*). Koska tunnelireititin voi toimittaa sekä ITR:n ja ETR:n virkaa, niin kutsutaan niitä yleisesti nimellä xTR. (Farinacci ym. 2013.) ITR:n ja ETR:n lisäksi tunnelireititin voi olla ns. ”Proxy”-reititin, joiden tehtävänä on välittää paketteja ITR:n ja ETR:n tapaan ei-LISP-toimialueilta LISP-toimialueille (Lewis ym. 2013).

Tunnelireitittimien lisäksi LISP-reitityksessä muita käytettäviä komponentteja ovat myös MS (*Map-Server*) ja MR (*Map-Resolver*), jotka mahdollistavat usean EID-osoitteen käytön RLOC-osoitteen takana. MR on komponentti, joka ottaa vastaan ITR-reitittimeltä tulevan LISP-enkapsuloidun Map-Request-viestin ja yhdistää EID-osoitteen RLOC-osoitteeseen EID-to-RLOC-tietokannan avulla. MS on taas puolestaan komponentti, joka ylläpitää em. tietokantaa ETR-reitittimeltä saatujen EID-to-RLOC-määrittelyjen mukaan ja vastaa Map-Request-viesteihin Map-Reply-viestein. MS-komponenttia voidaankin näin ollen verrata DNS-palvelimeen (*Domain Name System*) ja MR-komponenttia vastaavasti DNS-välimuistiselvittäjään (*DNS Cache Resolver*). (Fuller & Farinacci 2013.)

Alla kaksi esimerkkiä LISP-reitityksen toiminnasta:

Esimerkki 1. Kuvan 7 esimerkkitopologiassa LISP-toimialueella sijaitseva lähdelaite ”A” (EID: 192.168.0.10) haluaa keskustella ei LISP-toimialueella sijaitsevan kohdelaitteen ”B” (EID: 192.168.1.10) kanssa. Jotta yhteys saadaan muodostettua, toimitaan kuvan 6. mukaan seuraavasti:

1. A lähettää paketin kohti kohdeosoitetta B ja sen saapuessa xTR:lle tarkistaa xTR ensin löytyykö kohdeosoitetta paikallisesta tietokannasta, kun osoite ei ole ko. tietokannassa, lähettää xTR PxTR:lle Map-Request viestin.
2. PxTR tarkistaa löytyykö EID-to-RLOC tietokannasta kohdeosoitetta B. Kun sitä ei löydy, lähettää PxTR xTR:lle Map-Reply-viestin, jossa määritellään, että paketti reititetään natiivisti ilman LISP-enkapsulointia.
3. xTR ohjaa paketin PxTR:n ja Reitittimen kautta kohdeosoitteelle B tavallisesti.
4. Kohdeosoitteen B saadessa paketin, se vastaa siihen niin, että lähdeosoitteena on B ja kohdeosoitteena A. Reititin vastaanottaa paketin ja tarkistaa reititystaulustaan, että paketti kohteelle A tulee reitittää PxTR:n kautta.
5. PxTR:n saadessa paketin ja sen toimiessa MS:n roolissa, se tarkistaa EID-to-RLOC-tietokannasta, mihin xTR:n kohdeosoite A on rekisteröinyt itsensä ja Map-Request-viestin xTR:lle.
6. xTR vastaa PxTR:lle, että kohdeosoite A löytyy sen takaa ja määrittelee, että LISP-enkapsuloidun paketin kohdeosoite on yksi sen RLOC-osoitteista (VLAN 100).
7. PxTR enkapsuloi paketin ja toimittaa sen kohti xTR:a.
8. xTR ohjaa paketin kohteelle A. (Langemak 2012.)

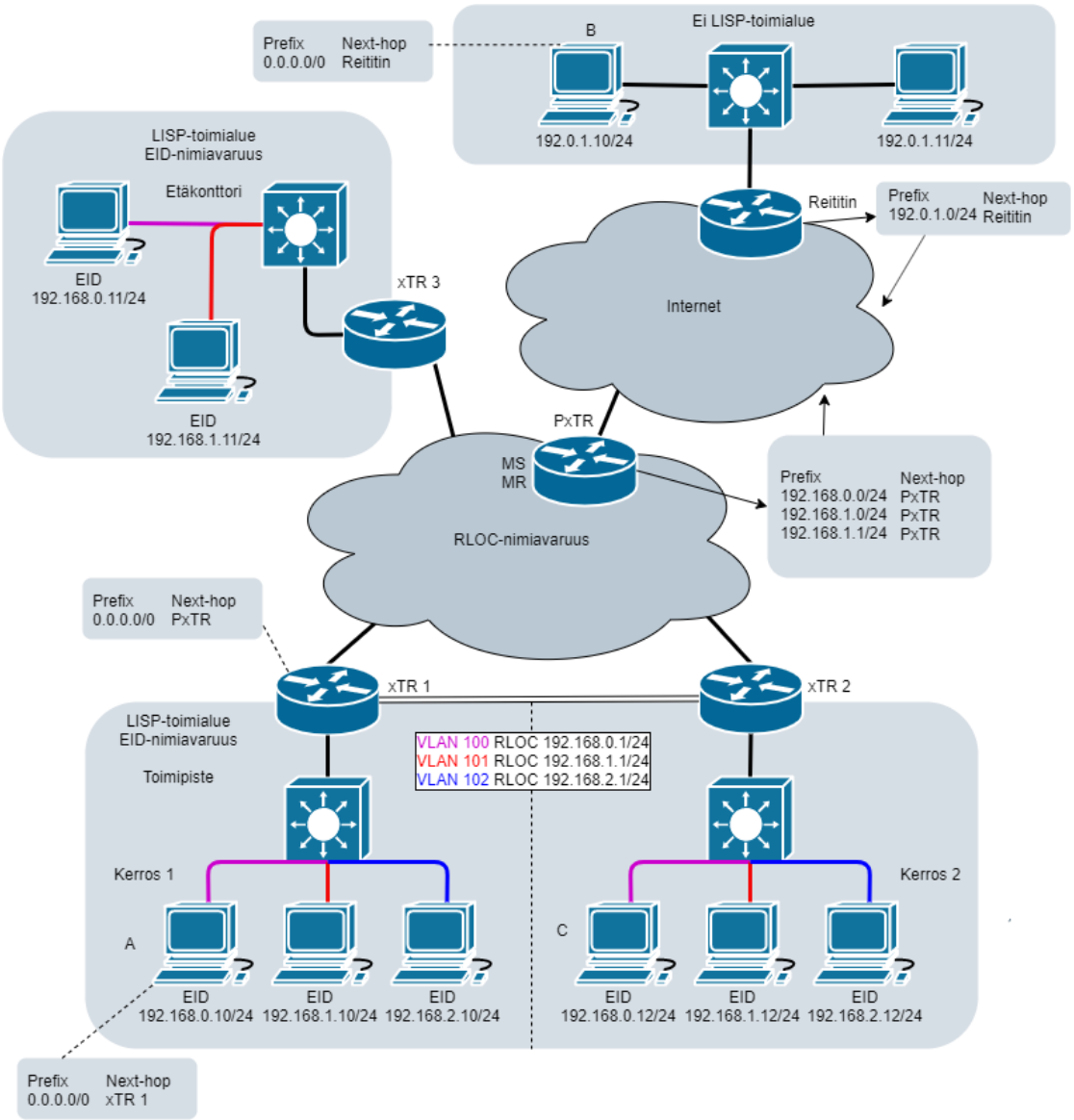


Kuva 6. Liikenne LISP-toimialueelta ei-LISP-toimialueelle ja takaisin

Esimerkki 2. Kuvan 7 esimerkkitopologiassa LISP-toimialueella sijaitseva lähdelaite ”A” (EID: 192.168.0.10) haluaa keskustella saman toimialueen sisällä

sijaitsevan kohdelaitteen ”C” (EID: 192.168.0.12) kanssa. Jotta yhteys saadaan muodostettua, toimitaan seuraavasti:

1. A lähettää paketin kohti kohdeosoitetta C ja sen saapuessa xTR 1:lle tarkistaa xTR 1 ensin löytyykö kohdeosoitetta paikallisesta tietokannasta. Kun osoite ei ole ko. tietokannassa, lähettää xTR 1 PxTR:lle Map-Request-viestin.
2. PxTR tarkistaa löytyykö EID-to-RLOC tietokannasta kohdeosoitetta C.
3. Kohdeosoitteen C löytyessä tietokannasta PxTR vastaa xTR 1:lle Map-Reply-viestillä, jossa kohdeosoitteena on määritelty xTR 2:n osoite.
4. xTR 1 LISP-enkapsuloi paketin ja toimittaa sen xTR 2:lle.
5. xTR 2 purkaa paketin ja toimittaa paketin kohdeosoitteeseen C.
6. Kohdeosoitteen C vastatessa lähteelle A, toimitaan toisin päin, kuin edellä.



Kuva 7. Esimerkkitopologia

3.3.4 IEEE 802.1X

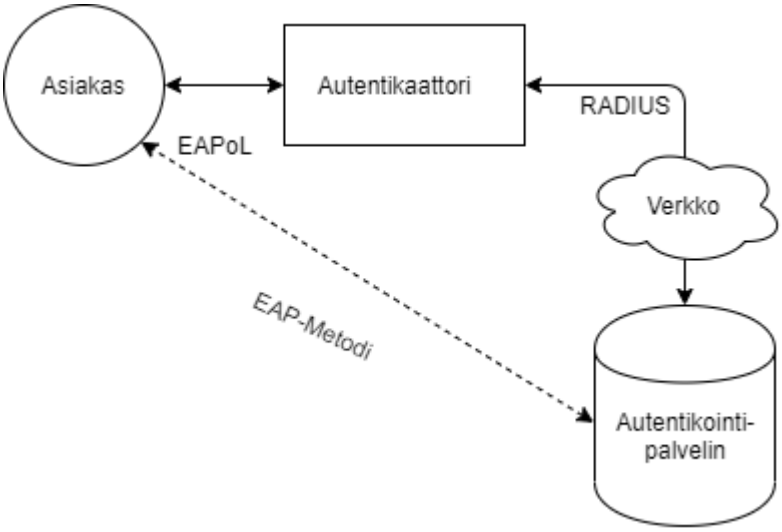
802.1X on IEEE-standardointijärjestön (*Institute of Electrical and Electronics Engineers*) IEEE 802 -työryhmän määrittelemä lähiverkoissa käytettävä standardi asiakkaan ja palvelimen väliseen porttikohtaiseen todennukseen. 802.1X-standardin tarkoitus on parantaa verkon tietoturvaa sallimalla verkkoon liittyminen vain todennetuilta käyttäjiltä tai laitteilta. 802.1X- porttikohtaisessa todennuksessa käytetään protokollia EAP (*Extensible Authentication Protocol*), EAPoL (*EAP over LAN*) ja RADIUS (*Remote Authentication Dial In User Service*) (Geier & Geier 2018, 39.)

Porttikohtaisen todennuksen toiminta perustuu kolmeen komponenttiin (kuva 8): asiakas (*supplicant*), autentikaattori (*authenticator*) ja autentikointipalvelin (*authentication server*).

1. Asiakas on käyttäjä tai laite, joka täytyy autentikoida, jotta sille voidaan sallia pääsy tietoverkkoon. Asiakkaita voidaankin ajatella tuntemattomina, joiden identiteetti ei ole tiedossa, ennen kuin ne voivat vahvistaa identiteettinsä voimassaolevilla käyttäjätunnuksillaan.

2. Autentikaattori on L2-verkkolaite, kuten kytkin, joka toimii kuin tietoturvaporttina asiakkaan ja suojatun verkon välissä. Näin ollen portti, johon asiakas kytkeytyy, pysyy suljettuna, kunnes autentikaattori vahvistaa identiteetin ja sen, onko ko. identiteetillä pääsyoikeus verkkoon. Kun järjestelmä on vahvistanut asiakkaan identiteetin, avataan portti. Tämän lisäksi autentikaattori toimii ns. tulkkina asiakkaan ja autentikointipalvelin välillä, sillä kaikki asiakkaan ja autentikointipalvelimen välinen liikenne kulkee autentikaattorin kautta.

3. Autentikointipalvelin on komponentti, jonka tehtävänä on varmistaa asiakkaan identiteetti. Yleisimmin autentikointipalvelimena toimii RADIUS-palvelin.



Kuva 8. 802.1X- porttikohtaisen todennuksen komponentit ja käytettävät protokollat

Komponentit keskustelevat keskenään käyttäen EAP-viestejä, joita kuljetetaan EAPoL- ja RADIUS-protokollien avulla. Asiakkaan ja autentikaattorin välisessä viestinnässä käytetään EAPoL-protokollaa, kun taas autentikaattori ja autentikointipalvelin viestivät keskenään RADIUS-protokollan avulla.

Porttikohtaisessa todennuksessa EAP-viesteillä määritellään, kuinka autentikointi asiakkaan ja autentikointipalvelimen välillä tapahtuu ja viestit pitävät sisällään mm. asiakkaan identifioivan tunnuksen ja salasanan, käytettävät sertifikaatit ja salausavaimet

Asiakkaan 802.1X-todennusprosessi toimii seuraavasti:

1. Asiakas lähettää autentikaattorille EAPoL-Start-viestin liittyessään verkkoporttiin.
2. Autentikaattorin saadessa viestin lähettää se asiakkaalle takaisin EAP-Request-viestin, jossa pyydetään asiakasta toimittamaan autentikaattorille sen autentikointiin tarvittavat tiedot.
3. Asiakkaan saadessa viestin, vastaa se autentikaattorille EAP-Response-viestillä, joka sisältää sen asiakkaan identiteetin.
4. Autentikaattori lähettää asiakkaalta saamansa identiteettitiedon eteenpäin autentikointipalvelimelle RADIUS Access Request -viestinä.
5. Access Request -viestin saadessaan RADIUS-palvelin lähettää autentikaattorille Access-Challenge-viestin, joka pitää sisällään todennuksessa käytettäviä määrityksiä, kuten EAP-tunnistemenetelmän.
6. Autentikaattorin saadessa viestin, se lähettää asiakkaalle EAP-Request Identity-viestin.
7. Asiakkaan saadessa viestin vastaa se siihen EAP-Response Identity-viestillä.
8. Autentikaattori lähettää saamansa viestin autentikointipalvelimelle RADIUS Acces-Request-viestinä.

9. RADIUS-palvelin vertaa viestissä saamaansa identiteettiä esim. salasanaa AD:ssa olevaan salasanaan ja jos palvelin toteaa identiteetin oikeaksi, lähettää se autentikaattorille RADIUS Access -viestin.
10. Autentikaattorin saadessa viestin, lähettää se asiakkaalle EAP-Success-viestin ja vaihtaa kytkentäportin tilan sallituksi, jotta asiakas voi liittyä verkkoon.

Kun asiakas on saanut käyttöoikeuden verkkoon, lähettää autentikaattori tasaisin väliajoin asiakkaalle kättelyviestejä selvittääkseen, onko asiakas edelleen kytkeytyneenä verkkoon. Jos asiakas vastaa, pitää autentikaattori liityntäportin tilan avoimena. Jos asiakkaalta ei saada vastausta tai se lähettää autentikaattorille EAPoL-Logoff-viestin, suljetaan verkon liityntäportti. (Geier & Geier 2019, 44–49.)

4 KÄYTÄNNÖN TOTEUTUS

4.1 SDA-verkon komponentti- ja IP-suunnittelu

Käytännön toteutus aloitettiin tekemällä suunnitelma SDA-verkkoinfrastruktuurin komponenteista, verkon aktiivilaitteiden kytkennöistä, IP-osoitteista ja VLAN-numeroista (kuva 9), jotta työvaiheessa olisi mahdollisimman helppo edetä tehdyn suunnitelman mukaisesti.

Cisco DNA Centerille valittiin käyttöönottoa varten vaadittavat IP-osoitteet seuraavasti:

1. Enterprise port address: 10.2.1.20/24
2. Cluster port address: 10.3.250.2/28
3. Management port address: 10.2.1.21/24
4. CIMC port address: 10.2.1.22/24
5. Cluster virtual IP address: 10.3.250.1/28
6. Service subnet: 10.4.0.0/21
7. Cluster services subnet: 10.4.8.0/21

Service- ja cluster services -aliverkot tuli valita niin, ettei kyseisiä aliverkkoja ollut käytössä muualla yrityksen lähiverkossa. Edellä mainittujen aliverkkojen suosituskoko oli määritelty /21-verkkoblokkiin, sen sisältäen 2 048 IP-osoitetta. Verkkoja ei kuitenkaan reititetä Cisco DNA Centeristä ulos, vaan niitä tarvitaan virtualisoitujen mikropalveluiden käyttöön Cisco DNA Centerin sisällä.

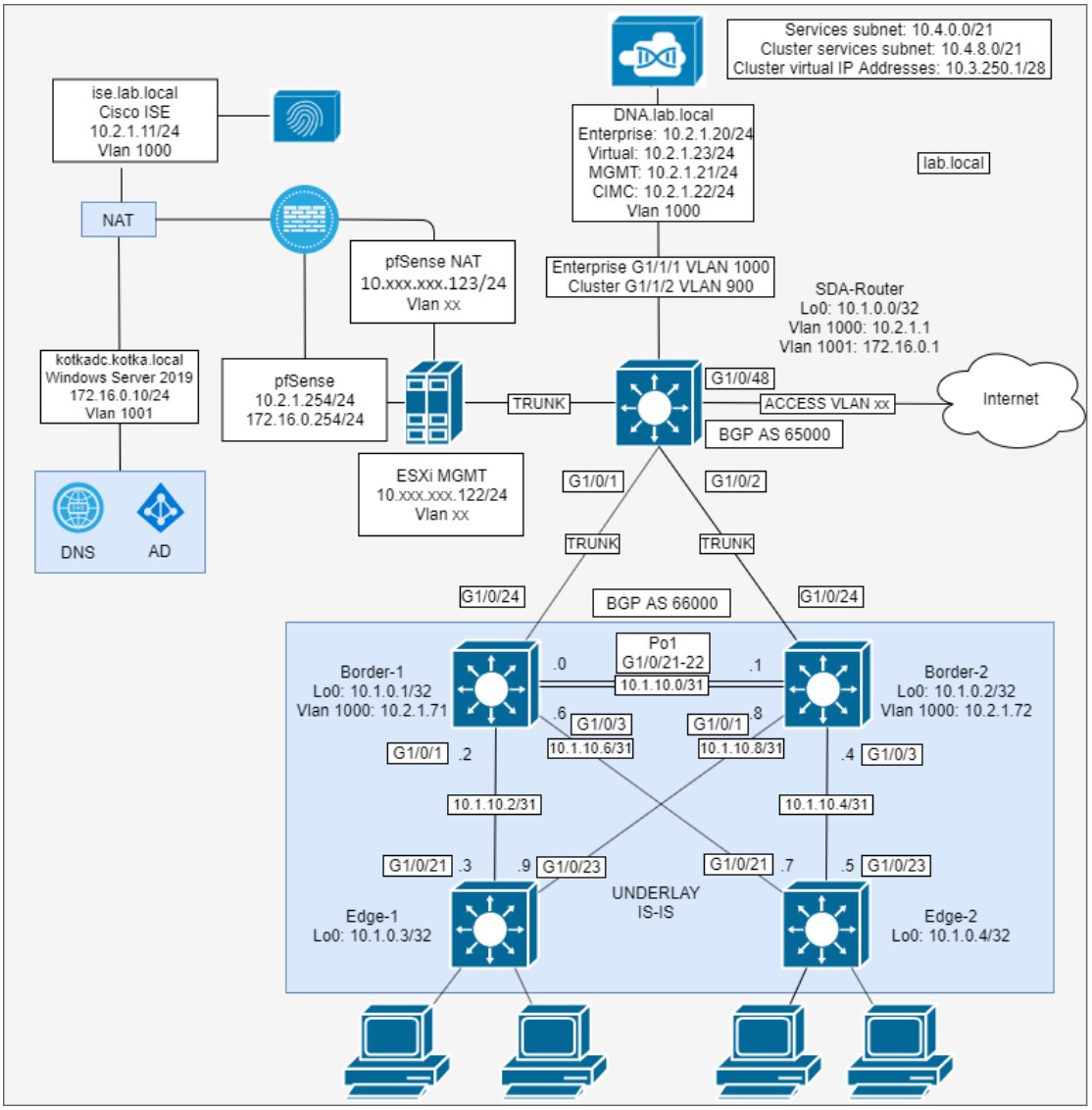
Cisco DNA Centerin lisäksi tuli valita IP-osoitteet ja -osoitealueet ja niiden VLAN-numerot verkon palvelukomponenteille, kuten Cisco ISE, AD (*Active Directory*) ja palomuuri. Osoitteet ja osoitealueet valittiin seuraavasti:

1. Cisco ISE (ise.lab.local): 10.2.1.11/24
2. Palomuuri: 10.2.1.254/24 ja 172.16.0.254/24
3. Palvelu-AD (dc.lab.local): 10.2.2.10/24
4. SDA-AD (kotkadc.kotka.local): 172.16.0.10/24
5. Palveluverkko: 10.2.1.0/24 (VLAN 1000)
6. SDA-verkko: 172.16.0.0/16 (VLAN 1001)
7. Verkon aktiivikomponenttien Loopback-osoitteet: 10.1.0.0/24
8. Cluster VLAN: 900

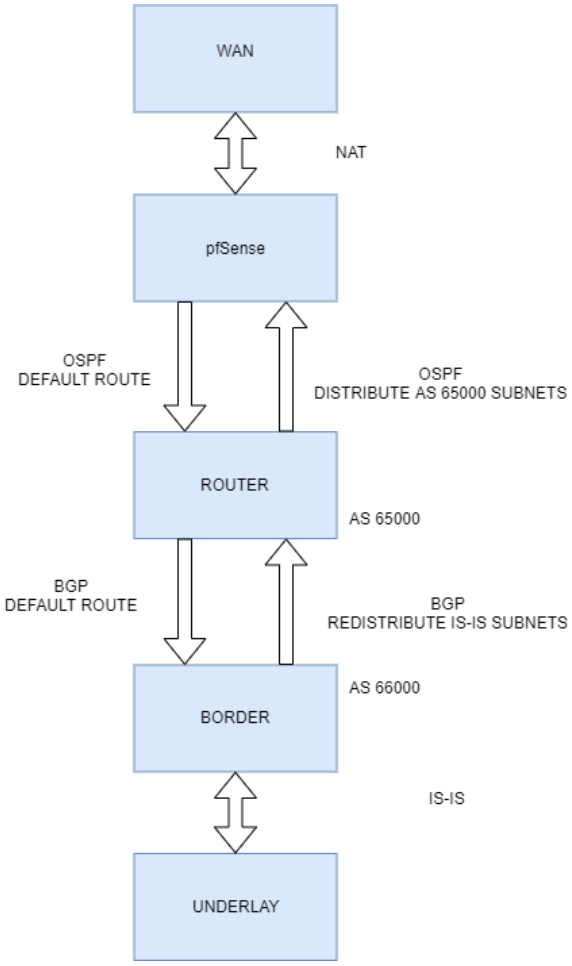
Tämän lisäksi yrityksen sisäverkosta valittiin palomuurille ja virtualisointialustalle osoitteet etä- ja Internetyhteyttä varten, niitä ei tässä opinnäytetyössä voida kuitenkaan julkaista siitä syystä, että ei haluta paljastaa yrityksen osoitteistusta.

Kun IP-osoitteet ja VLAN-numerot oli valittu, niin seuraavaksi tuli valita haluttu alusta virtualisointiin ja palomuurin toteutukseen. Virtualisointialustaksi valittiin VMware ESXi 6.7 sen kokeiluversion ollessa ilmainen ja palomuuriksi valittiin avoimeen lähdekoodiin perustuva pfSense. Virtualisointialusta päätettiin asentaa kannettavaan tietokoneeseen ja sen avulla virtualisoitiin AD:t ja pfSense-palomuuri.

SDA-verkon aktiivikomponenteiksi valittiin Ciscon SDA-yhteensopivuusmatriisin mukaan (Cisco 2019c) 9200- ja 9300-sarjan kytkimiä, niiden ollessa kykeneväisiä SDA-verkon rakennukseen. Border-kytkimiksi valittiin 9300-sarjan kytkimet, niiden ollessa kykeneväisiä toimimaan SDA-verkon Border-kytkiminä ja LISP-kontrollitasona**.** SDA-verkon Edge-kytkimiksi valittiin 9200L-sarjan kytkimet ja samaa kytkintä käytettiin myös SDA-verkon ulkopuolisena reitittimenä. Tässä suunnitelmassa ulkopuolinen SDA-Router reititin ja pfSense-palomuuri muodostivat ns. fuusioreitittimen (Hill ym. s.a., 86.), jotta liikenne VRF-verkkojen välillä saatiin reititettyä. Liikenne SDA-verkosta WAN-verkkoon päätettiin reitittää kuvan 10 mukaisesti. Tässä opinnäytetyössä ei käsitellä pfSense-palomuurin asennusta eikä konfigurointia.



Kuva 9. Suunnitelma SDA-verkkoinfrastruktuurista

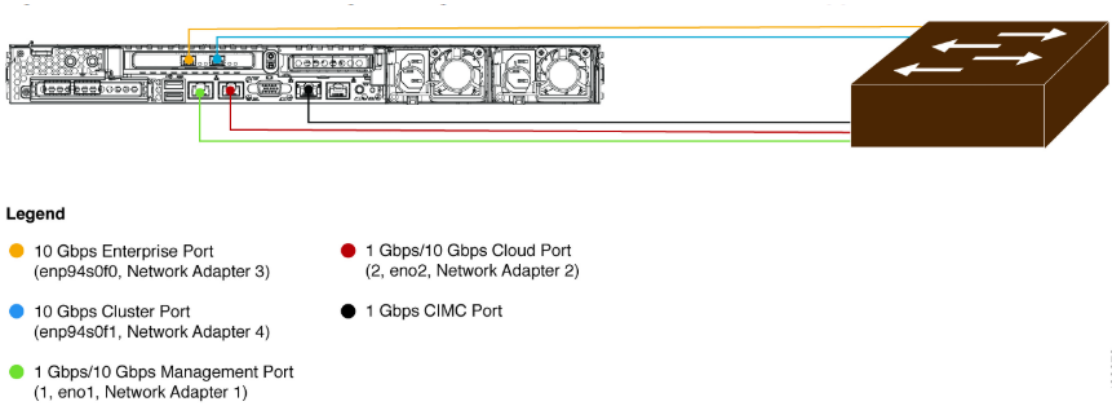


Kuva 10. Verkkoon suunnitellut reititysprotokollat kuvattuna

4.2 SDA-verkon laitteiden kytkentä ja konfigurointi

Seuraavana vaiheena käytännön työssä kytkettiin verkon aktiivikomponentit toisiinsa suunnitelman mukaisesti (kuva 9) ja konfiguroitiin laitteittain Cisco Software-Defined Access Deployment Guide -ohjeen mukaan (Cisco 2018b) (liite 1). Konfiguraation lisäksi verkkolaitteisiin luotiin käyttäjä “dna” korkeimmalla oikeustasolla.

Verkon aktiivilaitteiden kytkennän jälkeen kytkettiin Cisco DNA Center -palvelimen verkkoportit ohjeen mukaisesti (Cisco 2019a) (kuva 11) SDA-Router-kytkimeen. Cloud-portti jätettiin kytkemättä sen ollessa valinnainen. Hallintaliikenteen kulkiessa samassa verkossa kuin laitehallinta, jätettiin myös Management-portti kytkemättä.

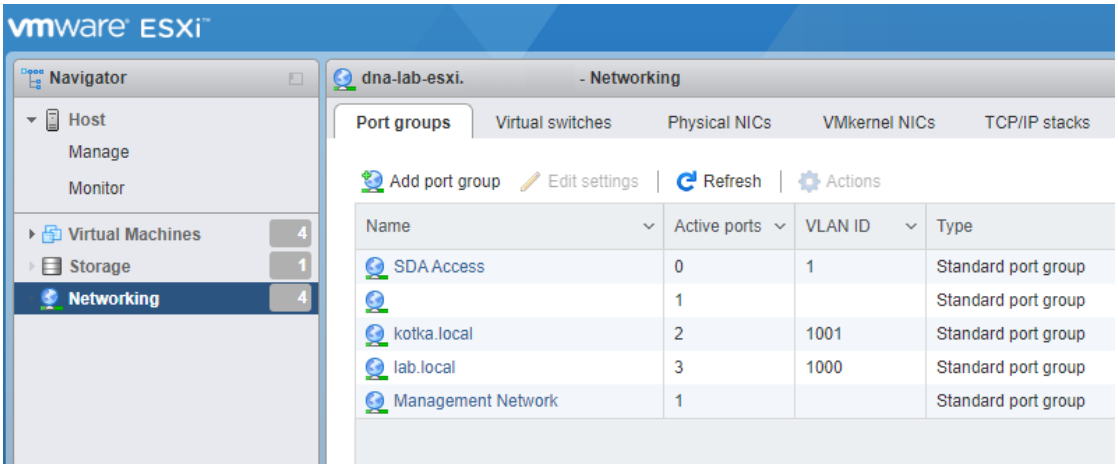


Kuva 11. Cisco DNA Center porttien kytkennät (Cisco 2019a)

4.3 Virtualisointialustan, Cisco ISE:n ja AD:n asennukset ja määritykset

Seuraavana vaiheena työssä ennen Cisco DNA Centerin asennusta oli virtualisointialustan, palvelinten ja palomuurin asennus ja konfigurointi. Jotta Cisco ISE, AD-palvelimet ja palomuuri voitiin virtualisoida, tuli asennus aloittaa virtualisointialustasta.

VMware ESXi 6.7 asennettiin kannettavaan tietokoneeseen (VMware 2018) ja määriteltiin IP-osoitteeksi osoite yrityksen verkosta, jolla virtualisointialustaan pääsi käsiksi. Tämän jälkeen ESXi-alustaan luotiin verkkoliitännät virtuaalikoneille (kuva 12).



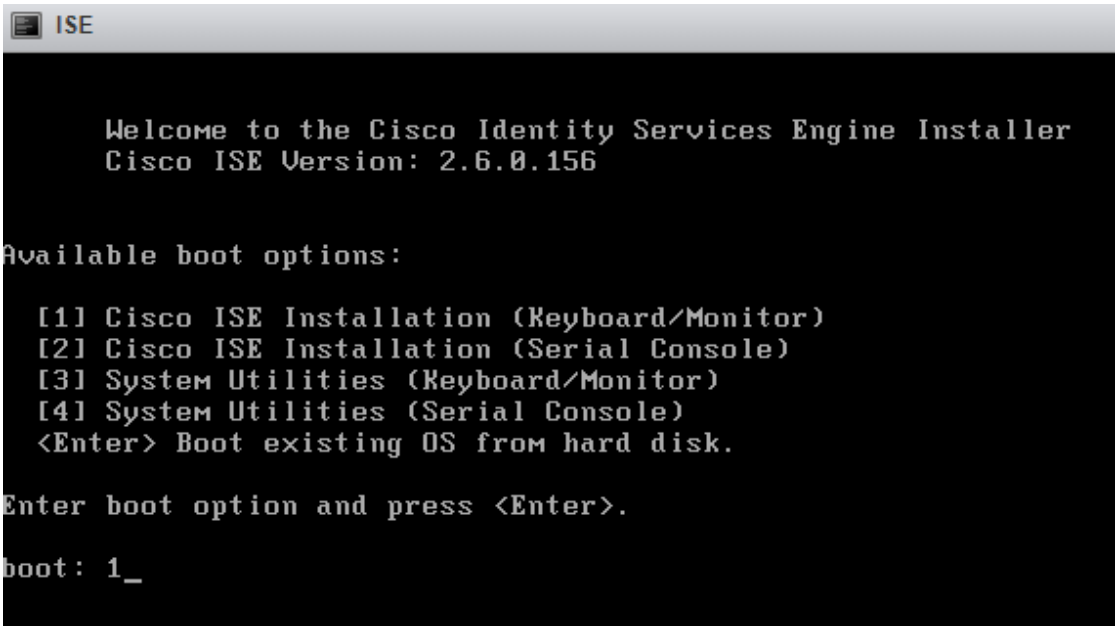
Kuva 12. Verkkoliitännät määriteltynä VMware ESXi-virtualisointialustassa

Virtualisointialustan asennuksen ja verkkojen luonnin jälkeen luotiin Cisco ISE:ä varten uusi virtuaalikone nimeltä ISE. Virtuaalikonetta luodessa määriteltiin sille myös tarvittavat resurssit (kuva 13).

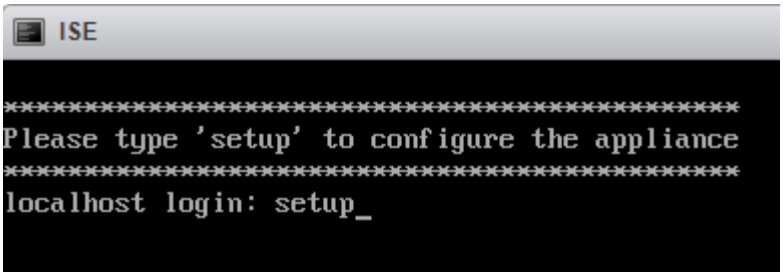
vCPUs	2
Memory	4096 MB
Network adapters	1
Network adapter 1 network	Virtuaalikoneet
Network adapter 1 type	VMXNET 3
IDE controller 0	IDE 0
IDE controller 1	IDE 1
SCSI controller 0	VMware Paravirtual
SATA controller 0	New SATA controller
Hard disk 1	
Capacity	200GB
Datastore	[datastore1] ISE/
Mode	Dependent
Provisioning	Thin provisioned
Controller	SCSI controller 0 : 0
CD/DVD drive 1	
Backing	[datastore1] ise-2.6.0.156.SPA.x86_64.iso
Connected	Yes
USB controller 1	USB 2.0

Kuva 13. ISE-virtuaalikoneen resurssimääritykset

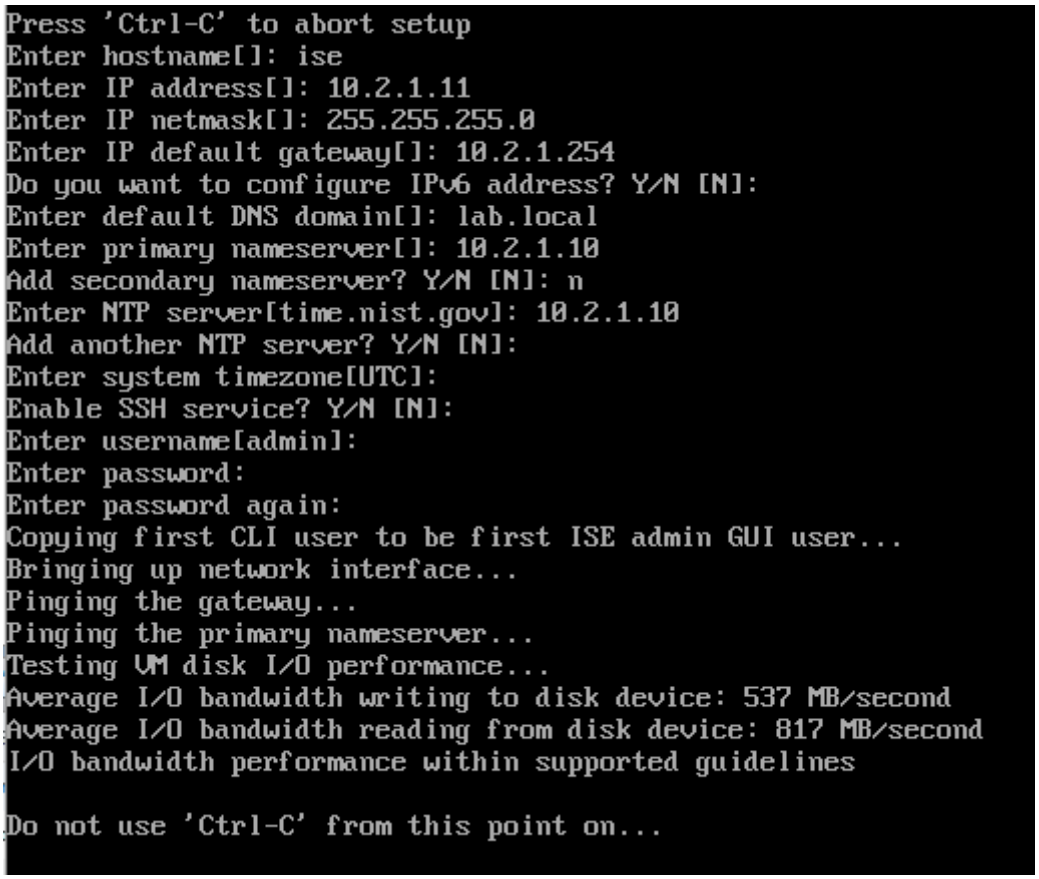
Kun virtuaalikone oli luotu ja käynnistetty, otettiin siihen yhteys ESXi-hallinnasta ja käynnistettiin kone asennusta varten. Cisco ISE:n asennus oli erittäin yksinkertainen ja vaati vain kuvien 14, 15 ja 16 mukaisen prosessin.



Kuva 14. ISE-asennuksen aloitusvalikko

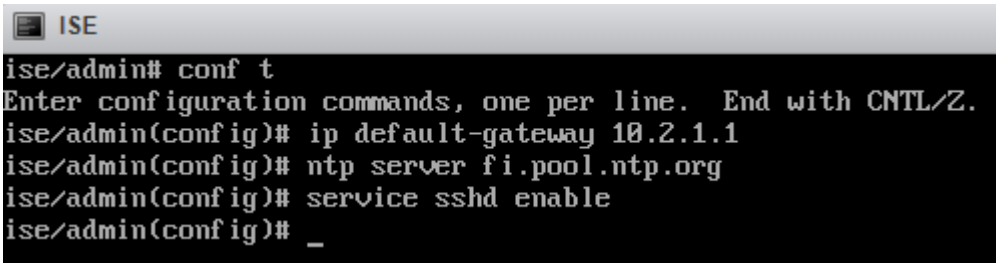


Kuva 15. ISE-asennuksen aloitus

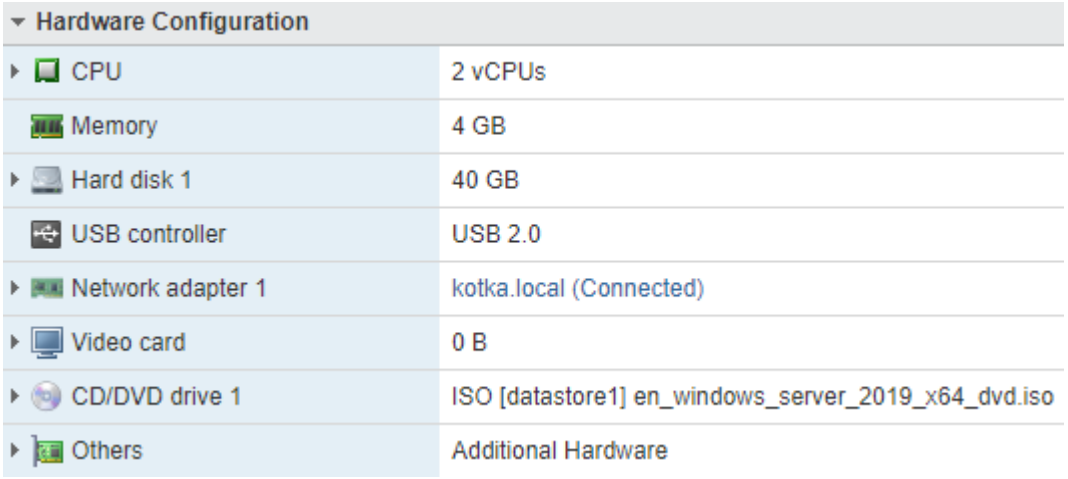


Kuva 16. ISE-asennukseen määritellyt asetukset

Asennuksen jälkeen huomattiin, että SSH-palvelu olisi tullut laittaa käyttöön ja että olisi helpompi käyttää julkista NTP-aikapalvelinta (*Network Time Protocol*), kuin rakentaa omaa vain työtä varten. Myöhemmin myös oletusyhdyskäytävä vaihdettiin osoittamaan IP-osoitetta 10.2.1.1. (kuva 17.) Hieman asennuksen jälkeen myös huomattiin, että virtuaalikoneelle määritelty 4 gigatavun muisti ei ollut riittävä ja muistin määrää nostettiin 6 gigatavuun.

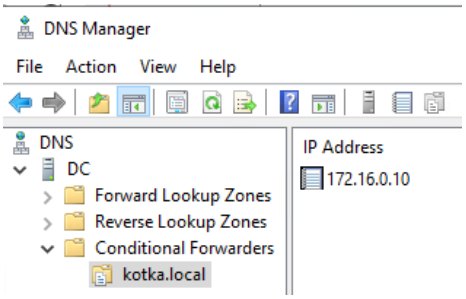


Cisco ISE:n asennuksen jälkeen asennettiin kaksi Windows Server 2019 -virtuaalikonetta: dc.lab.local ja kotkadc.kotka.local tarvittavin resurssein (kuva 17). Asennuksen jälkeen AD-koneisiin määriteltiin palvelinten nimet ja asennettiin Active Directory Domain Services-, DHCP server ja DNS server -komponentit Server Manager -työkalun kautta (Microsoft 2017).



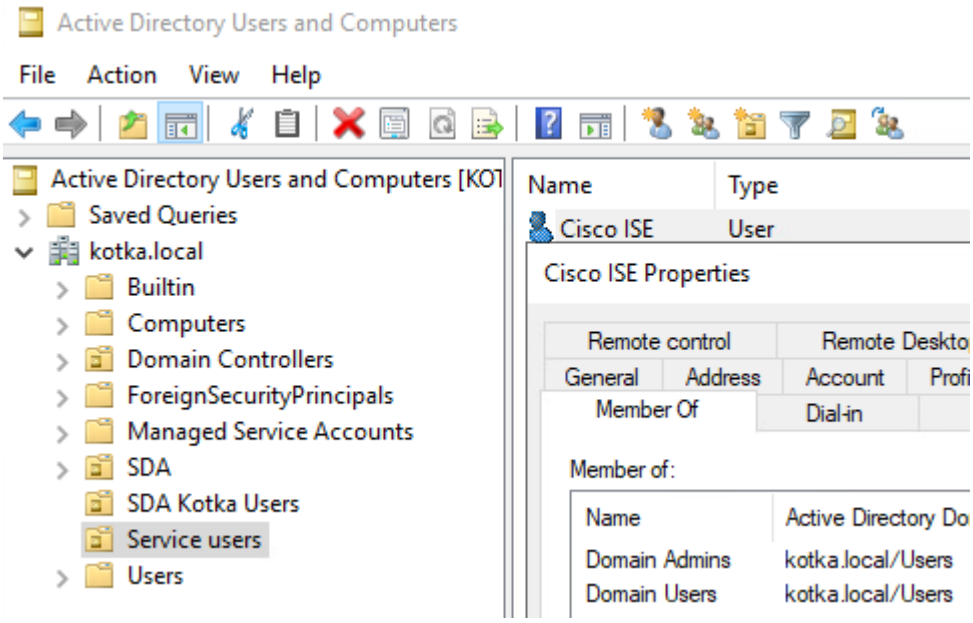
Kuva 17. AD-koneiden resurssit määriteltynä

Toinen AD-koneista toimi ns. palvelu-AD:n roolissa ja siihen asennettuun DNS-palvelimeen määriteltiin kotka.local-verkkotunnukselle Cisco ISE:n yhdistämistä varten ”conditional forwarder” osoittamaan kotkadc.kotka.local-palvelimen IP-osoitteeseen 172.16.0.10 (kuva 18). Näin ollen ISE löytäisi kotka.local-AD:n liitosvaiheessa DNS-palvelun avulla.



Kuva 18. Kotka.local ”conditional forwarder” määriteltynä DNS-palvelimeen

Cisco ISE:n yhdistämistä varten kotkadc-AD:yn tuli lisätä ns. palvelukäyttäjä, jotta ISE voi lisätä yhdistysvaiheessa itsensä koneeksi AD:hen. Tätä varten luotiin Active Directory Users and Computers -ohjelman avulla AD:hen ”Cisco ISE”-käyttäjä Domain Admins -ryhmän oikeuksilla (kuva 19). Palvelukäyttäjän lisäksi luotiin kaksi testikäyttäjää joonas ja testi myöhempää käyttöä varten. Tämän lisäksi AD:yn luotiin kaksi Security Group -ryhmää, SDA_Admins ja SDA_Users ja liitettiin käyttäjät ryhmiin siten, että joonas kuului ryhmään SDA_Admins ja testi ryhmään SDA_Users.



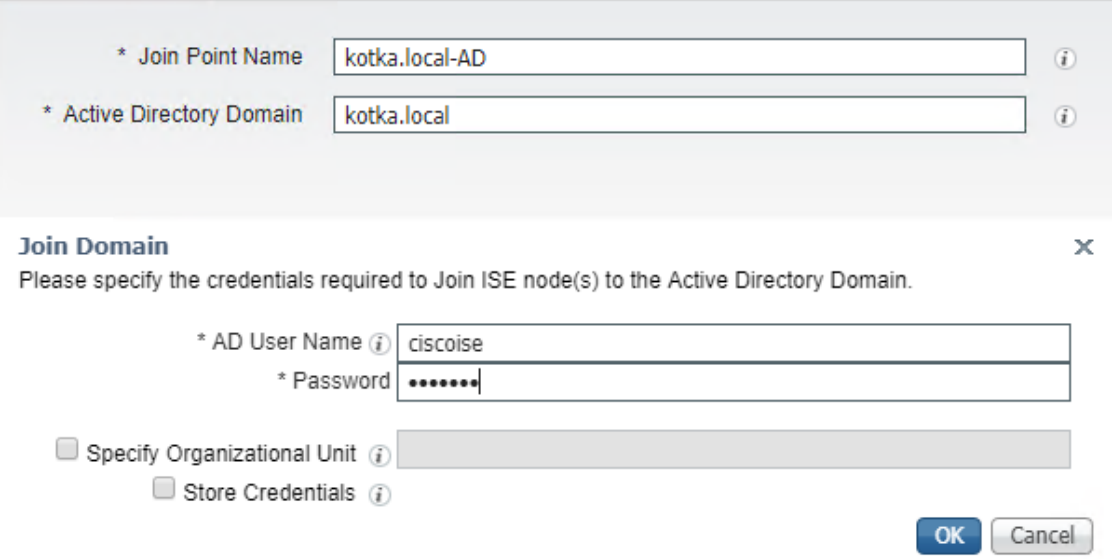
Kuva 19. Cisco ISE-palvelukäyttäjä

Käyttäjien verkkoyhteyksiä varten luotiin kotkadc-AD:yn myös seuraavat DHCP-osoitealueet:

1. VN_Kotka_SDA_Admins
    a. Osoitealue: 172.16.200.0/24
    b. Oletusyhdyskäytävä: 172.16.200.1
2. VN_Kotka_SDA_Users
    a. Osoitealue: 172.16.1.0/24
    b. Oletusyhdyskäytävä: 172.16.1.1
3. VN_Kotka_SDA_Visitors
    a. Osoitealue: 172.16.2.0/24
    b. Oletusyhdyskäytävä: 172.16.2.1

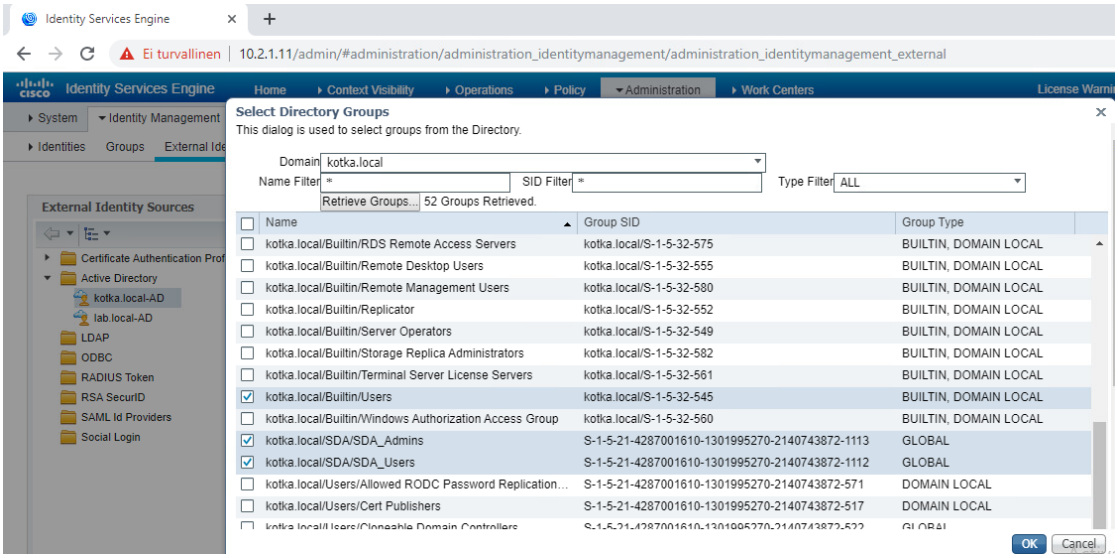
Edellä mainittujen asetusten lisäksi määriteltiin jokaiseen osoitealueeseen DNS-palvelimeksi IP-osoite 172.16.0.10.

Kun DHCP-poolit ja käyttäjä oli luotu, voitiin siirtyä seuraavaan vaiheeseen, jossa ISE yhdistetään kotkadc.kotka.local-AD:yn. AD liitettiin ISE:en hallintapaneelin valikon **Administration -> Identity Management -> External Identity Sources -> Active Directory** -polun alta valitsemalla ”Add” ja määrittelemällä liitokseen tarvittavat kohdat (kuva 20). Liitosvaiheessa ISE kysyi, halutaanko se liittää ko. AD:hen ja siihen vastattiin ”Yes”.



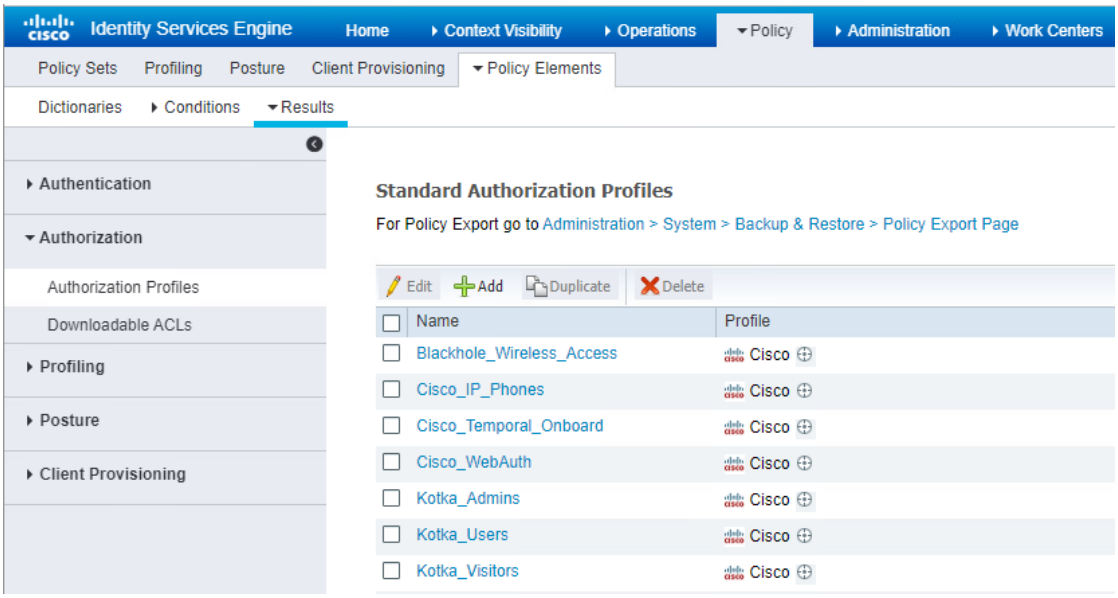
Kuva 20. Cisco ISE:n liittäminen kotkadc.kotka.local AD:yn

Seuraava vaihe oli lisätä aiemmin lisätyt ryhmän ISE:en liitetyn AD:n alle valitsemalla välilehti ”Groups” ja painamalla Add. Tämän jälkeen valikosta valittiin lisättävät ryhmät (kuva 21).

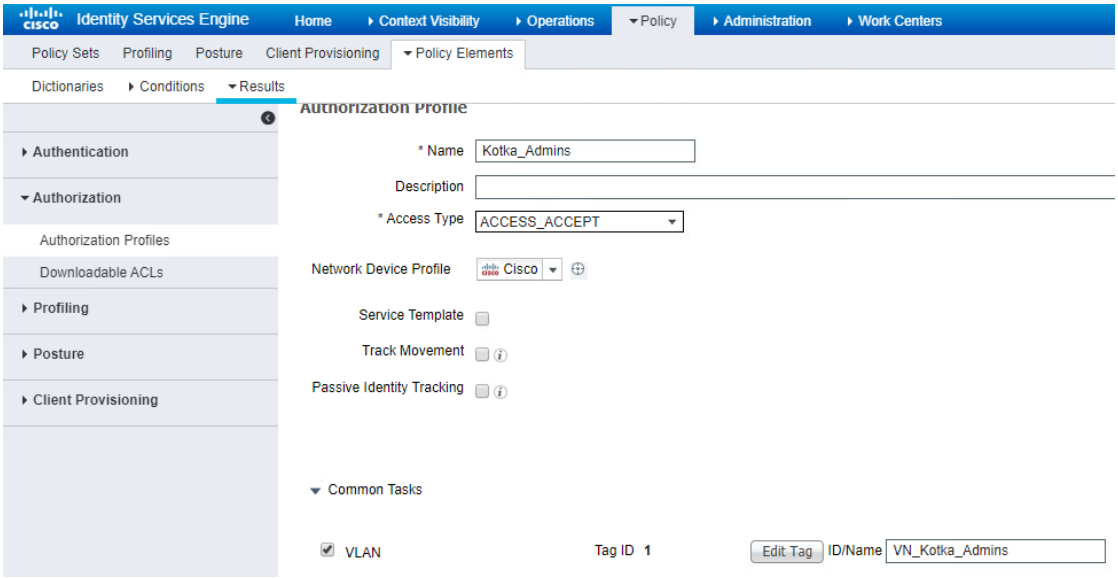


Kuva 21. Ryhmien lisäys AD:yn alle

Seuraavana vaiheena ISE:ssä luotiin Authorization-profiilit ryhmille Kotka_Users, Kotka_Admins ja Kotka_Visitors **Policy -> Policy Elements -> Results -> Authorization -> Authorization Profiles** -polun (kuva 22) alta valitsemalla ”add” ja määrittelemällä asetukset (kuva 23). Asetukset noudattivat kaavaa nimen osalta ”Kotka_Ryhmä” ja VLAN-nimen osalta ”VN_Kotka_Ryhmä”.



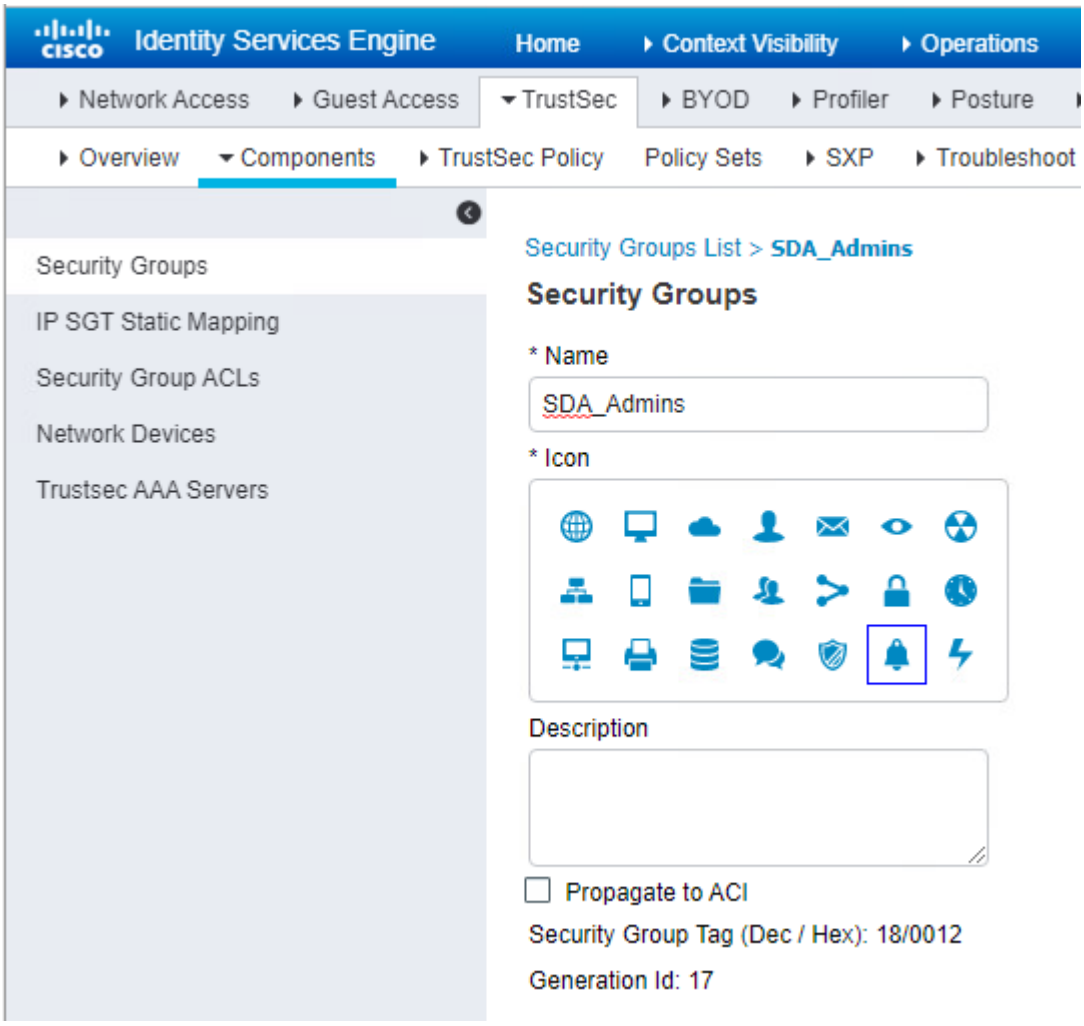
Kuva 22. Authorization-profiilin lisäys



Kuva 23. Authorization-profiilin asetukset

Tämän jälkeen luotiin ISE:en myöhemmin tarvittavat SDA_Admins-, SDA_Users- ja SDA_Visitors- SGT-ryhmät **Work Centers -> TrustSec ->**

Components -> Security Groups -polun alta valitsemalla ”Add” ja syöttämällä avautuvaan valikkoon tiedot (kuva 24). Lisäyksen jälkeen varmistettiin, että ryhmät näkyivät listassa (kuva 25).

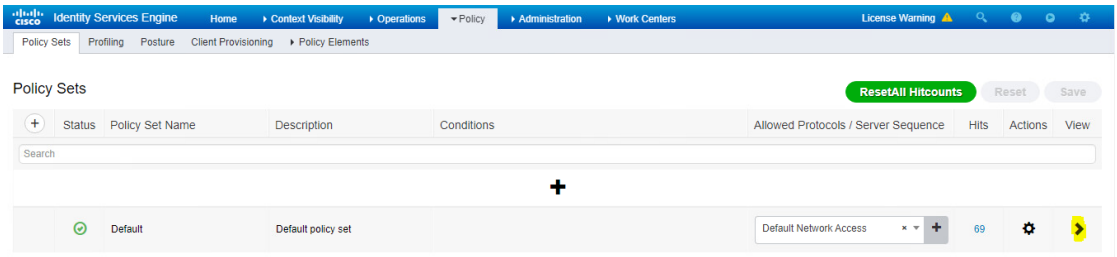


Kuva 24. SGT-ryhmän lisäys ISE:en

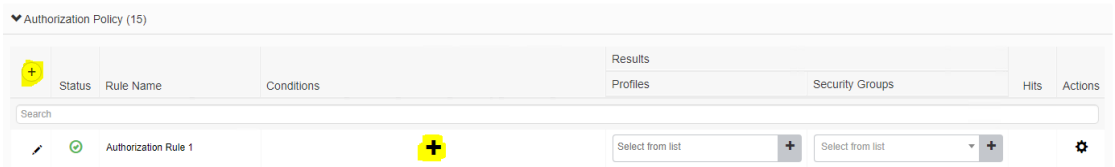
Kuva 25. Lisätyt SGT-ryhmät

Seuraava työvaihe oli lisätä ISE:en käyttöoikeuspolitiikat, jotta käyttäjän liittyessä verkkoporttiin osattaisiin sille valita oikea politiikka. Politiikat luotiin niin, että jos käyttäjää ei pystytä tunnistamaan, asetettaisiin sille ryhmäksi SDA_Visitors ja politiikaksi Kotka_Visitors.

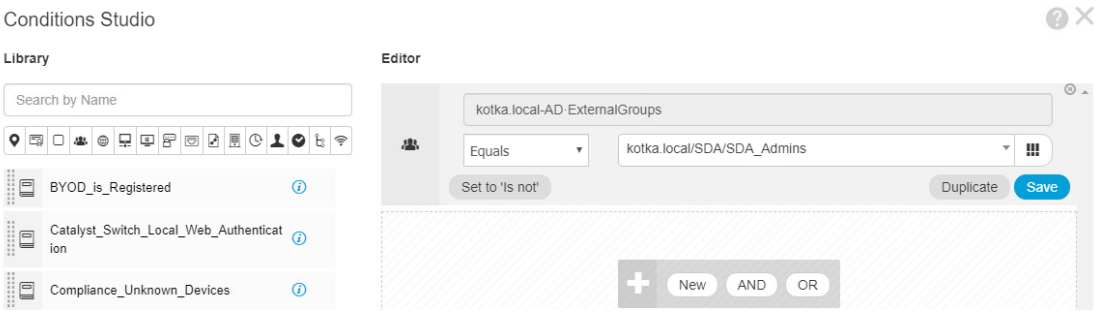
Käyttöoikeuspolitiikkojen luonti tapahtui **Policy -> Policy Sets** -polun alta valitsemalla Default-politiikasta ”View” (kuva 26). Tämän jälkeen avattiin ”Authorization Policy” -listaus ja sen alta valittiin plus-painike politiikan lisäämiseksi ja luotuun politiikkaan määriteltiin nimi. Politiikan ehdot luotiin valitsemalla ”Conditions”-kohdasta plus-painike (kuva 27). Avautuvaan ikkunaan syötettiin ryhmän kohdalle haluttu ryhmä (kuva 28) ja tallennettiin painamalla ”Save”. Tämän jälkeen luodulle politiikalle valittiin profiili ja SGT-ryhmä niille varatuista kohdista. Edellä mainitulla tavalla luotiin politiikat kahdelle ryhmälle (kuva 29) pl. SDA_Visitors, sillä ko. ryhmään kuuluu vain käyttäjät, joilla ei ole AD-identiteettiä. SDA_Visitors-ryhmän käyttöoikeuspolitiikka määriteltiin listassa olevaan ”Basic_Authenticated_Access”-politiikkaan siten, että profiilina käytettiin Kotka_Visitors-profiilia ja SGT-ryhmänä SDA_Visitors-ryhmää (kuva 30). Kun politiikat oli luotu, tallennettiin ne sivun ylälaidasta valitsemalla ”Save”.



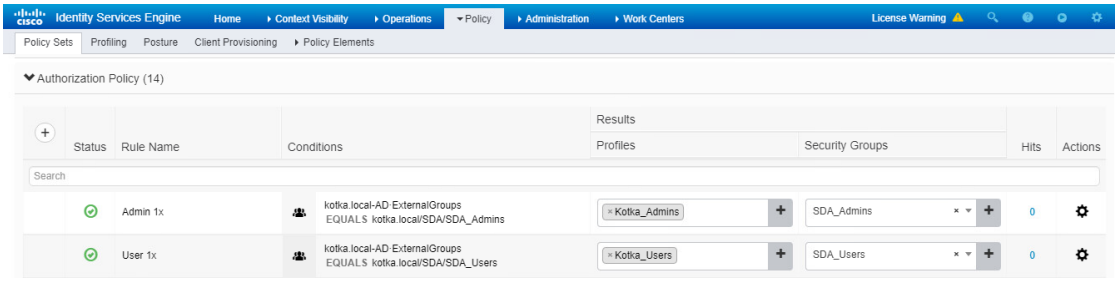
Kuva 26. Käyttöoikeuspolitiikkojen muokkaus



Kuva 27. Uuden politiikan lisäys ja muokkaus



Kuva 28. Politiikan ehtojen määritys

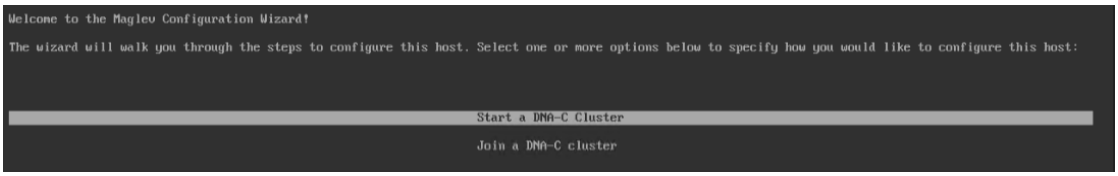


Kuva 29. Luodut politiikat

Kuva 30. Politiikka käyttäjille, joilla ei ole AD-identiteettiä

4.4 Cisco DNA Centerin asennus ja SDA-määritykset

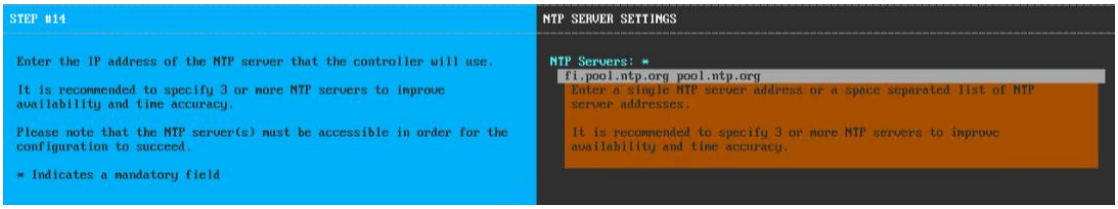
Cisco DNA Centerin asennus aloitettiin ottamalla selaimella yhteys CIMC-osoitteeseen ja valitsemalla konsolinäkymä. Näin ollen asennus voitiin tehdä etänä. Kun konsolinäkymä oli saatu auki, aloitettiin esille tulleesta valikosta asennus valitsemalla ”Start a DNA-C Cluster” (kuva 31). Tämän jälkeen asennusohjelmaan määriteltiin oikeisiin portteihin niihin suunnitellut IP-osoitteet (kuva 32). Porttien nimet eri linkeille löydettiin asennusoppaan kuvasta (kuva 11). Klusterilinkkiä en94s0f1 määritellessä valittiin asennusohjelmasta ”Cluster link” -kohta käyttöön. Seuraavana asennuksessa asetettiin halutut salasanat maglev-käyttäjälle ja admin-käyttäjälle (kuva 33). Asennuksessa kysyttiin seuraavaksi NTP-aikapalvelimen osoitteita (kuva 34) ja kun ne oli syötetty, varmisti asennusohjelma, että aikapalvelimiin saatiin yhteys (kuva 35). Viimeisenä vaiheena asennusohjelmassa ennen itse asennusta määriteltiin palveluiden osoitealueet suunnitelman mukaisesti (kuva 36).



Kuva 31. DNA Centerin asennuksen aloitus

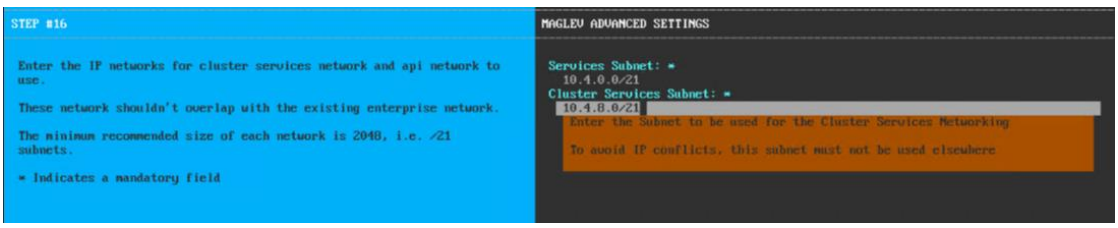
Kuva 32. IP-osoitteidein määritys

Kuva 33. Salasanojen määritys



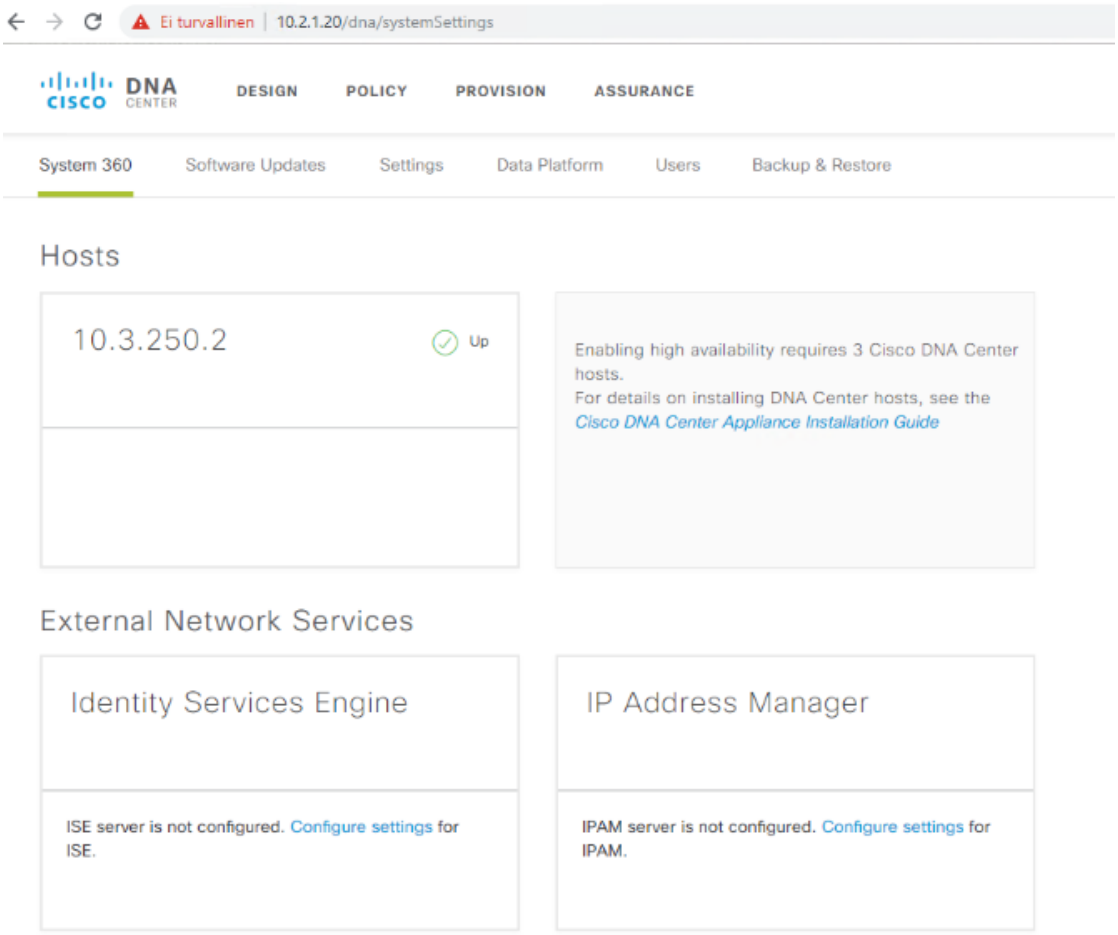
Kuva 34. NTP-aikapalvelinten määritys

Kuva 35. NTP-aikapalvelimen validointi



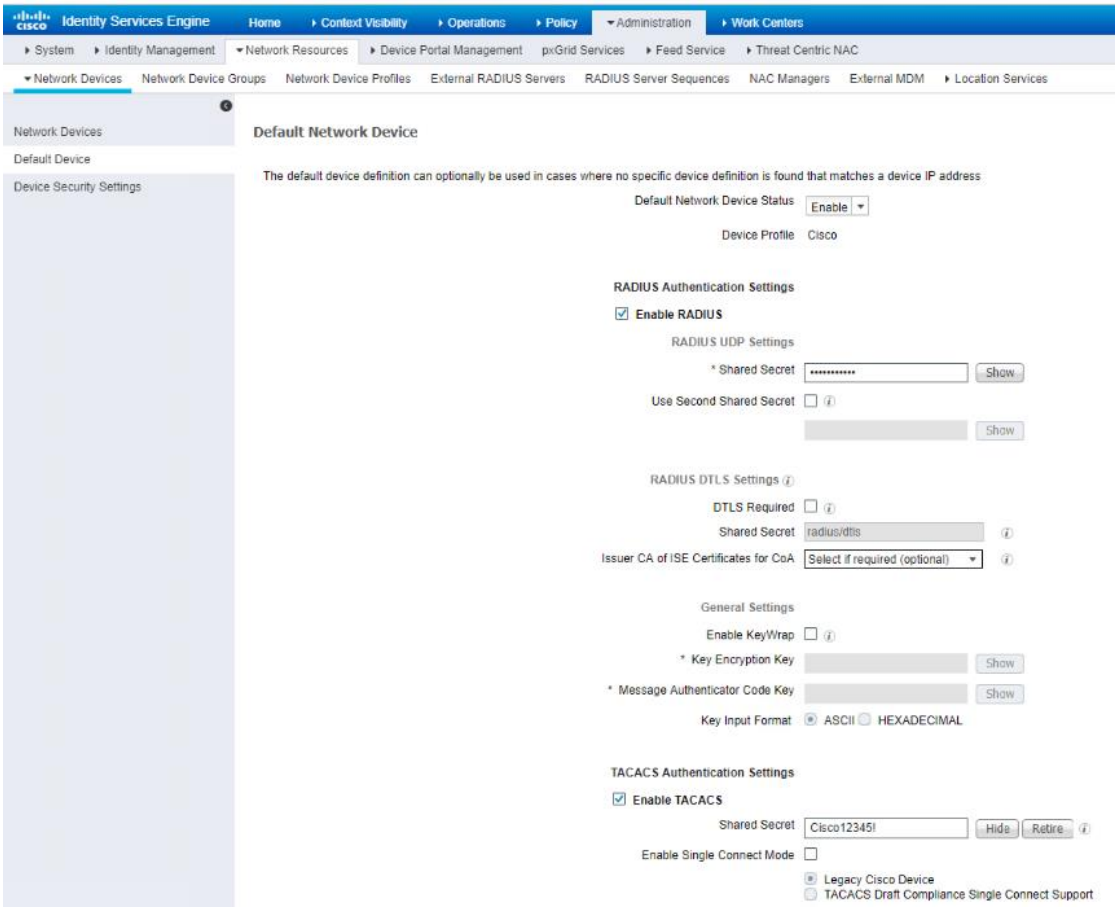
Kuva 36. Palveluiden osoitealueiden määritys

Kun asennusohjelma oli suorittanut asennuksen loppuun, käynnistettiin palvelin uudelleen ja odotettiin noin tunti, että järjestelmä olisi valmiina käytettäväksi. Odotuksen jälkeen kirjauduttiin Cisco DNA Centeriin selaimella ja tehtiin ensimmäisellä kirjautumiskerralla olevat asetukset ja hyväksyttiin käyttöehdot. Tarvittavia asetuksia oli Cisco Login ID. Asetuksissa kysyttiin myös IPAM-järjestelmää ja Proxy-asetuksia, mutta niitä ei syötetty. Tämän jälkeen järjestelmä ohjasi selaimen Cisco DNA Center -hallintasivulle (kuva 37).



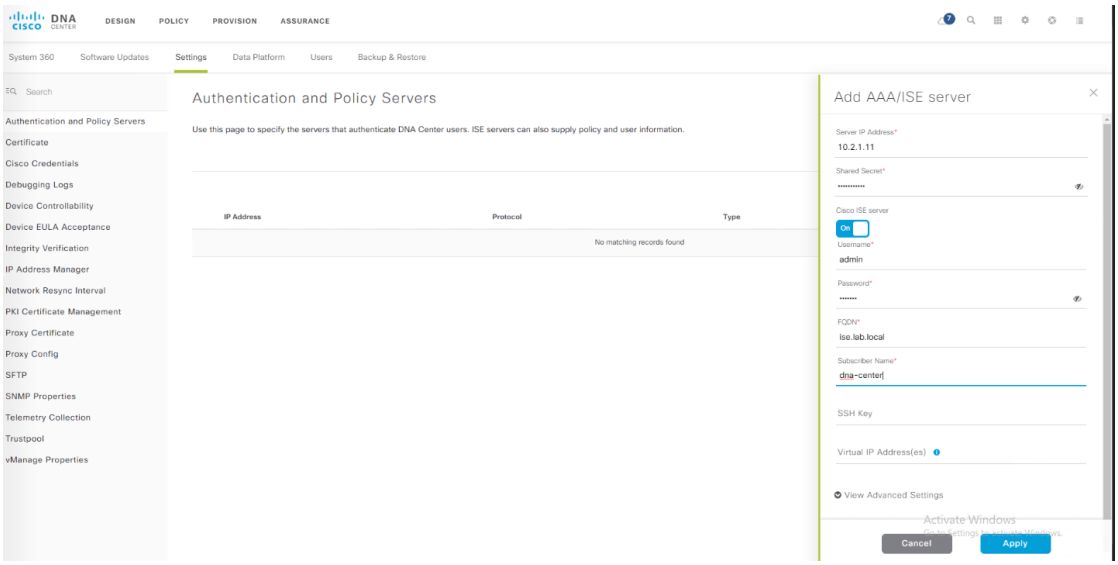
Kuva 37. Cisco DNA Center-hallintasivu

Tässä vaiheessa tehtiin ohjeiden mukaan (Cisco 2019a) määritykset ISE:en, jotta DNA Centerin yhdistys olisi mahdollista. **Administration -> System -> Deployment** -polun alta määriteltiin päälle “pxGrid”-valinta ja **Administration -> System -> Settings -> ERS Settings** -polun alta määriteltiin päälle “Enable ERS for Read/Write” -valinta ja tallennettiin asetus. Tämän lisäksi ISE:en määriteltiin RADIUS-palvelimen osoite **Administration -> Network Resources -> Network Devices -> Default Device** -polun alta (kuva 38).



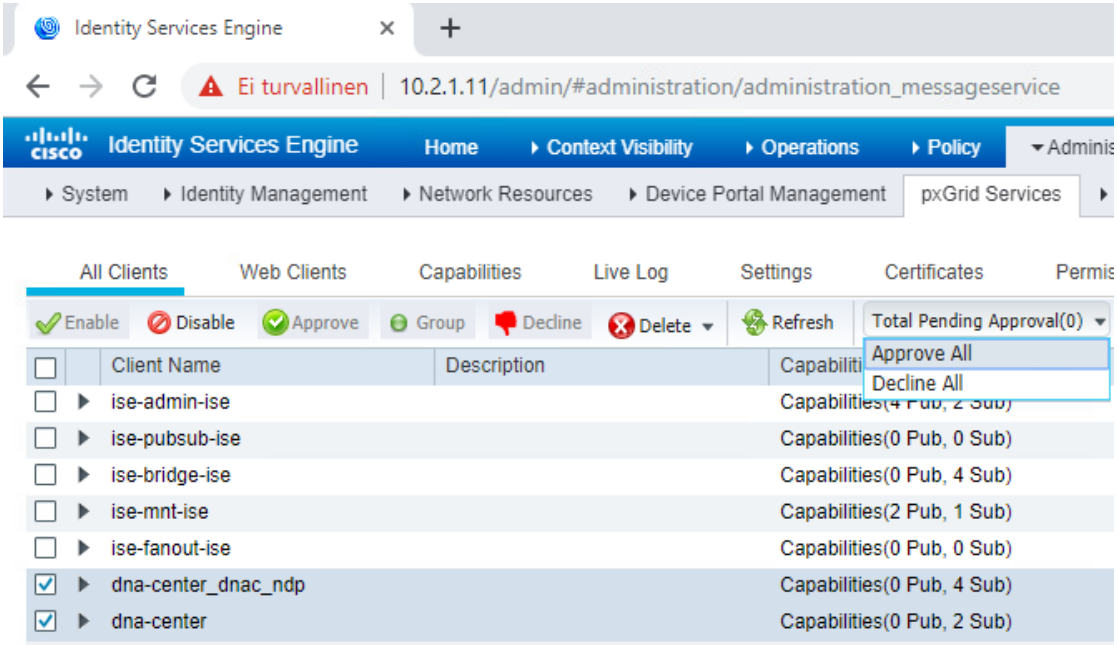
Kuva 38. RADIUS-palvelimen salasanan määritys

ISE:n asetusten jälkeen DNA Center-hallintasivulta valittiin ”Identity Services Engine” -kohdan alta ”Configure settings” ja lisättiin ISE-palvelin DNA-Centeriin (kuva 39).



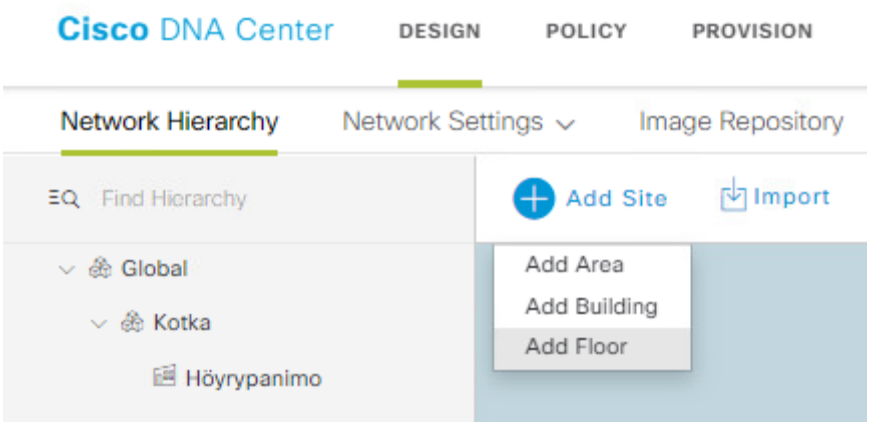
Kuva 39. ISE:n lisäys DNA Centeriin

Seuraavaksi DNA Center-ISE -yhteys vahvistettiin ISE:n hallintapaneelista **Administration -> pxGrid Services** -polun alta valitsemalla ja hyväksymällä DNA Centerin lisäämät kohdat (kuva 39).

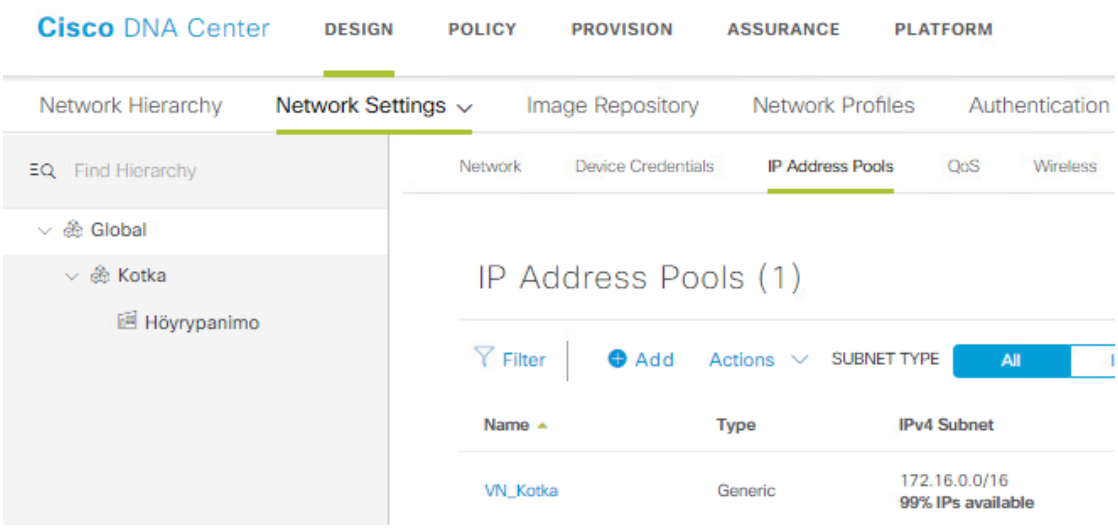


Kuva 40. DNA Center-ISE -yhteyden vahvistaminen

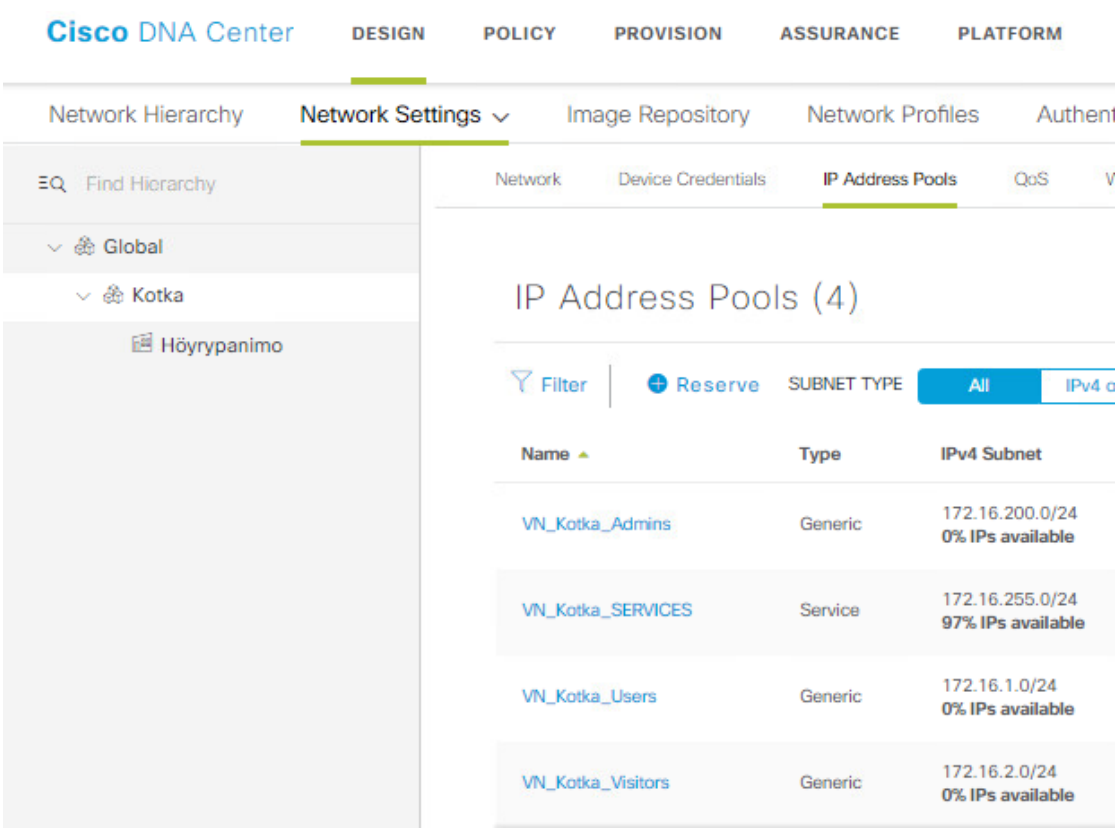
Kun itse asennus ja tarvittavat määritykset järjestelmiin oli tehty, päästiin suunnittelemaan SDA-verkkoa Cisco DNA Centerin sisällä. Ensimmäisenä luotiin verkkohierarkia, johon lisättiin alue (Area) Kotka ja sen alle rakennus (Building) Höyrypanimo (kuva 41). Seuraavaksi verkon laitteille määriteltiin Network Settings -valikon Network-välilehdellä AAA-palvelin ja tunnukset Device Credentials -välilehdellä. IP Address Pools -välilehdellä määriteltiin Global-tasolla käytettävä IP-avaruus 172.16.0.0/16 (kuva 42), joka jaettiin maantieteellisen sijainnin mukaan (Kotka) pienempiin avaruuksiin (kuva 43). Jokaiseen osoitealueeseen määriteltiin oletusyhdyskäytävän, DHCP-palvelimen osoite ja DNS-palvelimen osoitteet. Oletusyhdyskäytävän osoite on se, johon tehtävässä DHCP-osoitealueessa viitataan myöhemmin ja se muodostuu Edge-kytkimille automaattisesti anycast gateway -osoitteena. Oletusyhdyskäytävän osoitteeksi valittiin verkon ensimmäinen osoite. DNS- ja DHCP-palvelimiksi määriteltiin kotkadc.kotka.local-palvelimen IP-osoite 172.16.0.10.



Kuva 41. Verkkohierarkia

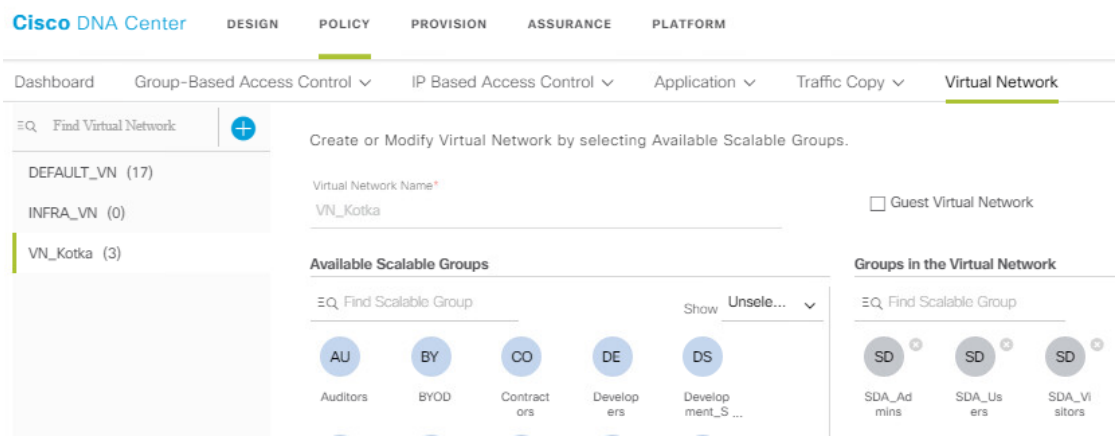


Kuva 42. Globaalilla tasolla määritelty käytettävä IP-osoiteavaruus

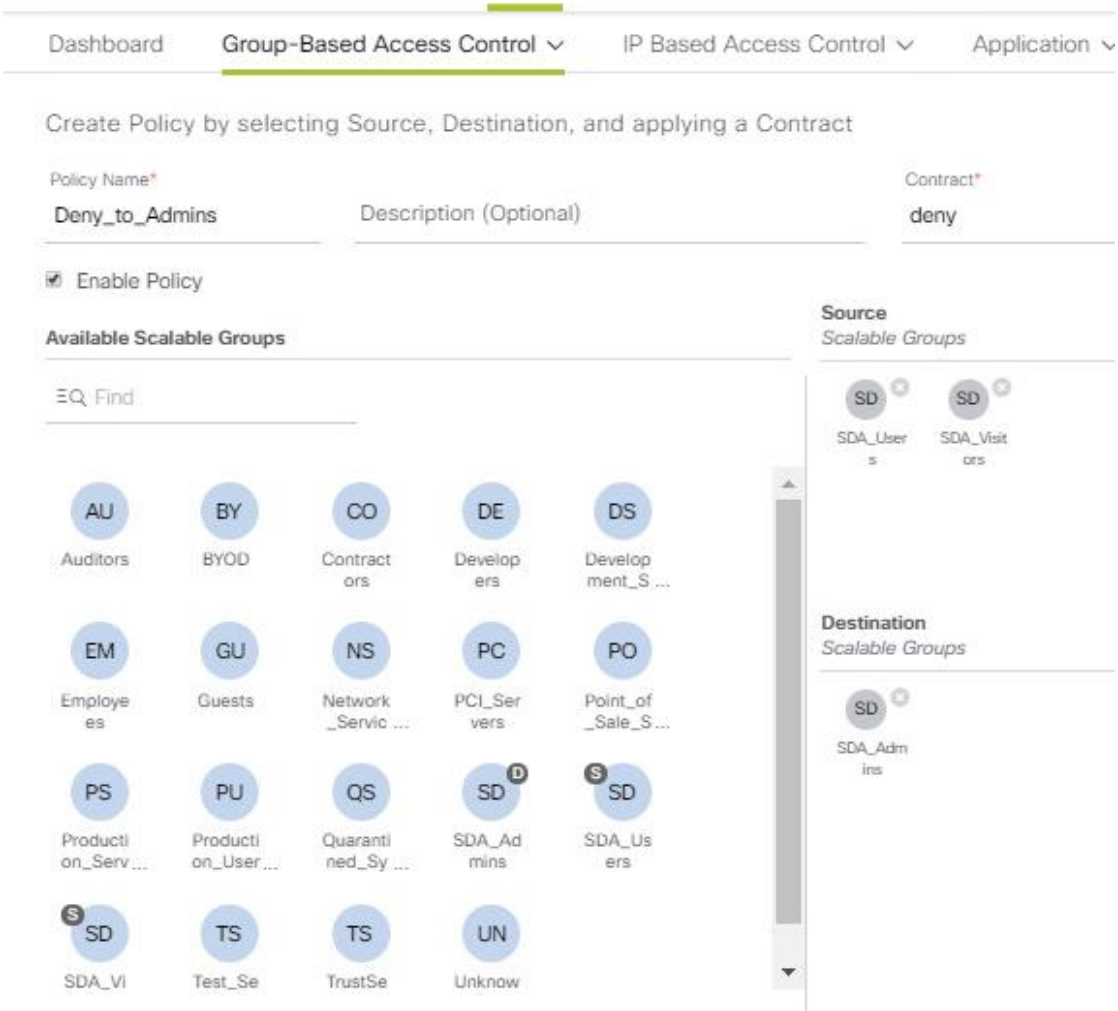


Kuva 43. Maantieteellisen sijainnin mukaan ryhmille jaetut IP-osoiteavaruudet

Verkon suunnittelun jälkeen aloitettiin politiikkojen suunnittelu luomalla Policy-valikon alla olevan Virtual Network -välilehden kautta uusi VN nimeltä VN_Kotka ja liitettiin siihen hiirellä raahaamalla luodut SGT-tunnisteet (kuva 44). VN:n luonnin jälkeen tehtiin käyttöoikeuspolitiikat SGT-ryhmille Group-Based Access Control -välilehden alta (kuva 45). Politiikat luotiin niin, että SDA_Users- ja SDA_Visitors-ryhmistä liikenne SDA_Admins-ryhmään estettiin. Myös SDA_Visitors-ryhmälle luotiin politiikka niin, että laitteet eivät saa keskustella keskenään, eikä muiden ryhmien kanssa. SDA_Users-ryhmälle ei luotu käyttöoikeuspolitiikkoja. Kun politiikat oli luotu, julkaistiin ne Deploy-painikkeella.



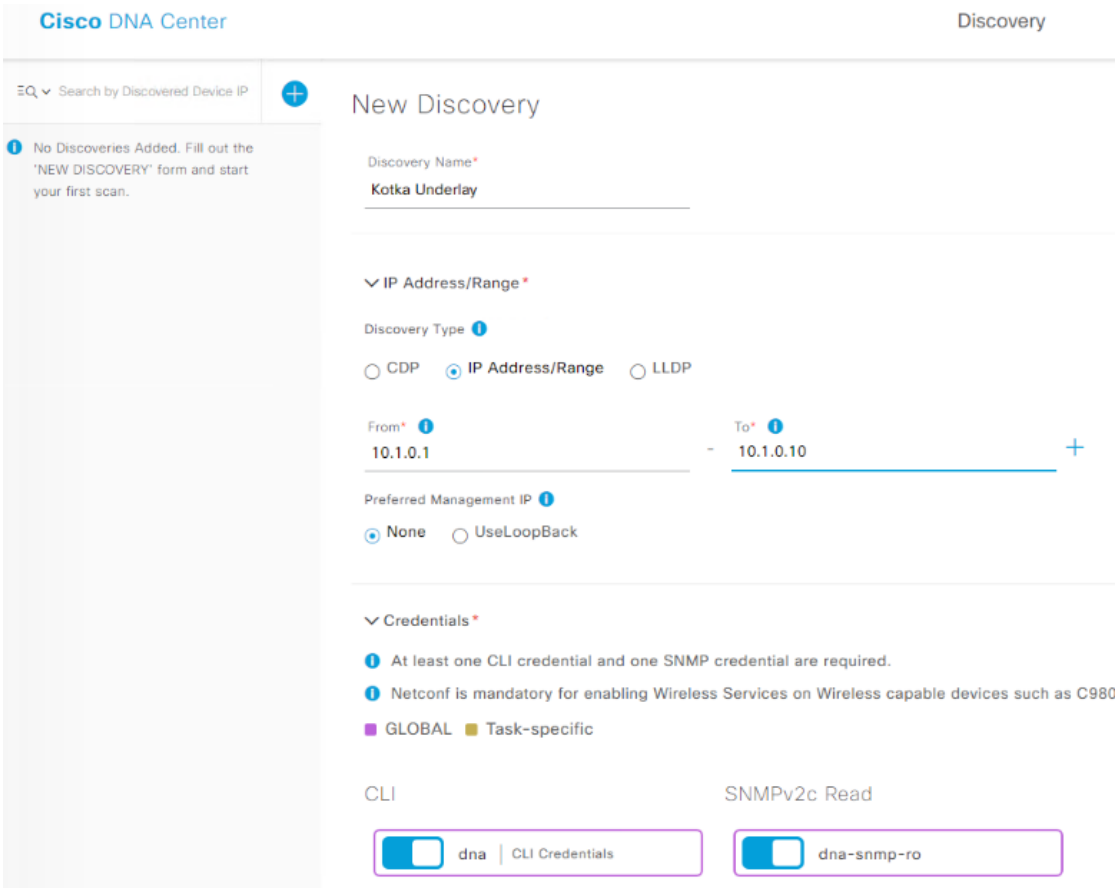
Kuva 44. Virtuaalisen verkon luonti



Kuva 45. Käyttöoikeuspolitiikan luonti

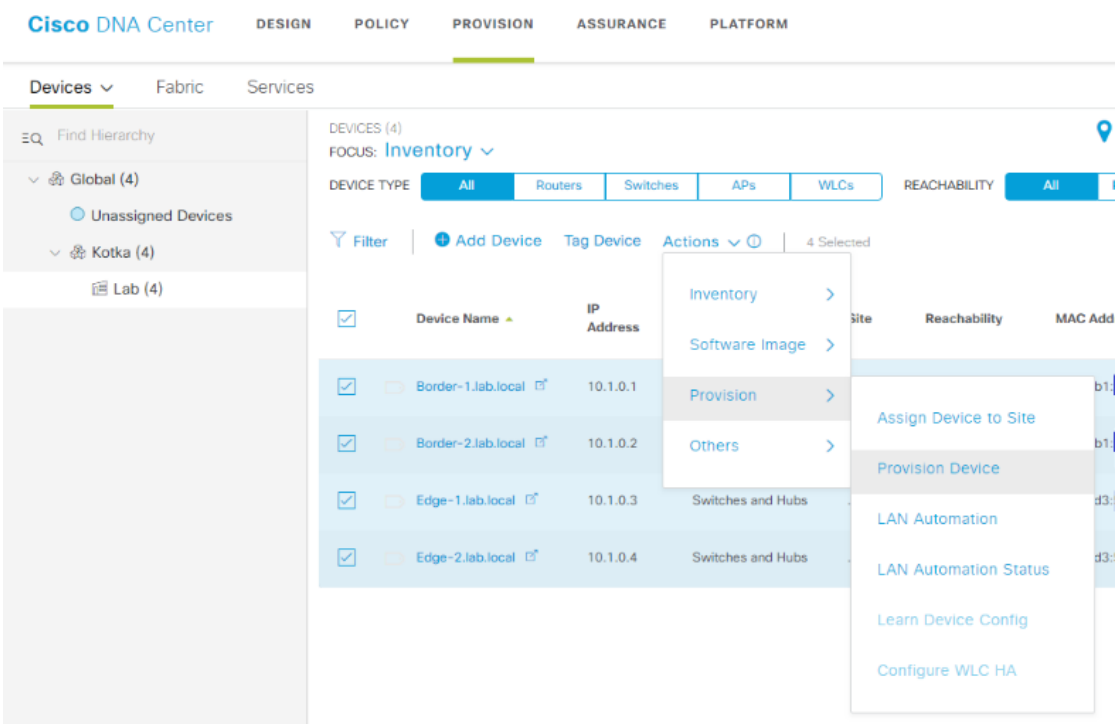
Käyttöoikeuspolitiikkojen jälkeen siirryttiin verkon provisiointivaiheeseen. Provisiointi aloitettiin tekemällä verkon sisällä laitteiden etsintä (device discovery) Discovery Dashboardin kautta valitsemalla ”Add Discovery”. Uuteen avautuneeseen ikkunaan syötettiin osoitealue, jolta laitteita etsittiin, määriteltiin laitteisiin määritellyt tunnukset (kuva 46) ja aloitettiin etsintä. Hetken kuluttua, kun

etsintä oli suoritettu, ilmoitti DNA Center, että on löytynyt 4 laitetta (kuva 47). Seuraavaksi laitteet provisioitiin Devices-välilehden alta valitsemalla kaikki löydetyt laitteet ja valitsemalla Actions-valikosta Provision Device -toiminto (kuva 48). Laitteiden provisioinnissa määriteltiin, että ne sijoitetaan Global/Kotka/Höyrypanimo-toimipisteeseen. Provisioinnin jälkeen laitteet sijoitettiin toimipisteelle valitsemalla kaikki laitteet Devices-välilehden alta ja valitsemalla Actions-valikosta Assign Device to Site -toiminto. Seuraavalla sivulla myös valittiin sijoituspaikaksi Höyrypanimo. Kun laitteet oli provisioitu ja sijoitettu toimipisteeseen, asetettiin niille oikeat roolit (kuva 49).

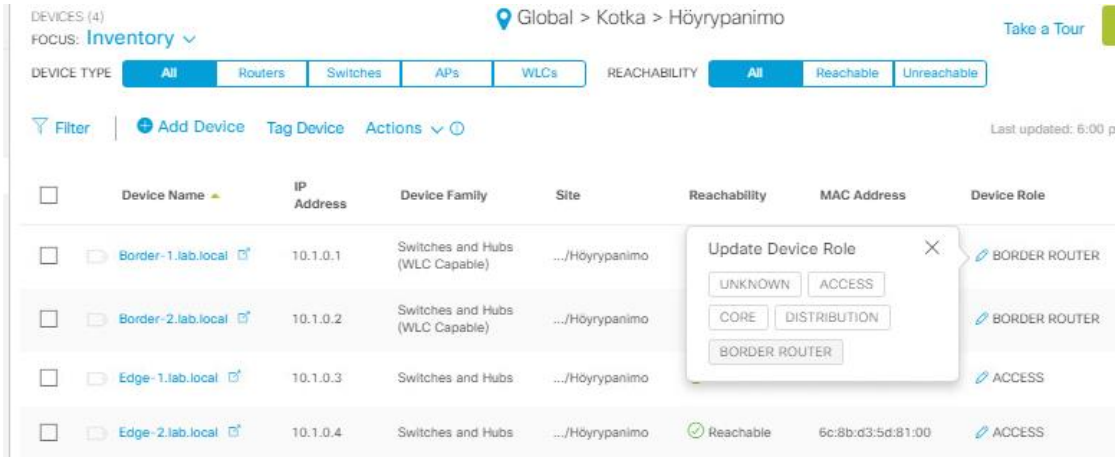


Kuva 46. Verkon laitteiden etsintä

Kuva 47. Löydetyt laitteet



Kuva 48. Laitteiden provisiointi



Kuva 49. Laitteiden roolien määritys

Laitteiden provisioinnin jälkeen aloitettiin SDA-verkon rakennus Provision-valikon Fabric-välilehden alta luomalla uusi SDA-verkko nimeltä Kotka Fabric ja sijoittamalla se Kotkaan. SDA-verkkoon määriteltiin käytettäviksi kaikki VN:t (kuva 50). SDA-verkon lisäksi luotiin Transit/Peer-verkko nimeltä TRANSIT_VN_Kotka, jonka avulla liitetyt päätelaitteet mainostetaan BGP-reititysprotokollaa käyttäen fuusioreitittimelle SDA-Router (kuva 51).

SDA-verkon luonnin jälkeen valittiin Fabric-välilehdeltä luotu Kotka Fabric, jonka jälkeen aukesi uusi sivu, missä näytettiin SDA-verkon topologia (kuva

52). Topologiasta valittiin laitteet yksi kerrallaan ja määriteltiin niihin asetukset seuraavasti:

1. Edge-laitteet määriteltiin Edge-laitteiksi.
2. Border-laitteet määriteltiin Border-laitteiksi ja Control Plane-laitteiksi (kuva 53). Border-laitteiden asetuksiin (kuva 54) lisättiin AS-numero ja verkoksi valittiin VN_Kotka_SERVICES. Verkko määriteltiin oletukseksi kaikille VN-verkoille.

   Tämän jälkeen Border-laitteista katsottiin Control Plane -lisätiedot (kuva 55), jotta verkot voidaan lisätä BGP-reititykseen ja näin ollen saataviksi fuusioreitittimelle.

Border-1-laitteelle oli määritelty transit-verkoksi VLAN 3001 ja Border-2-laitteelle VLAN 3002 seuraavin osoittein:

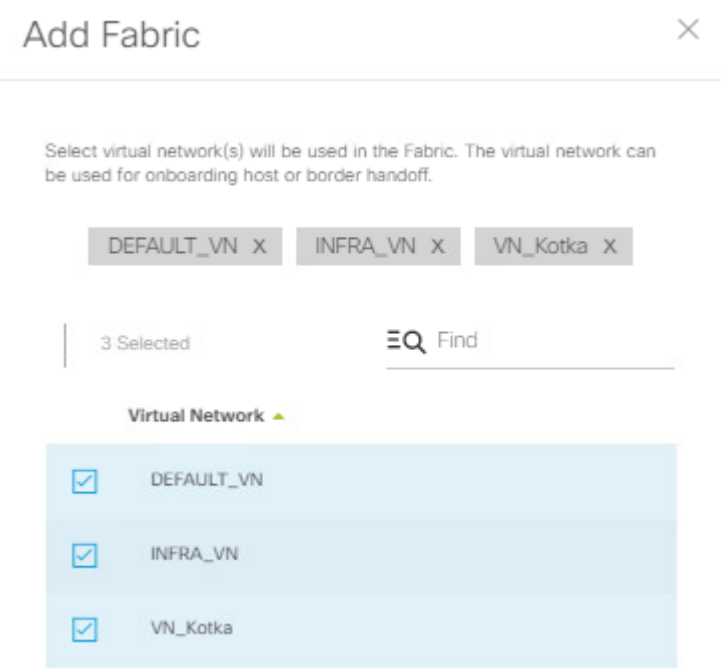
Border-1: 172.16.255.1/30 (VLAN 3001)

Border-2: 172.16.255.5/30 (VLAN 3002)

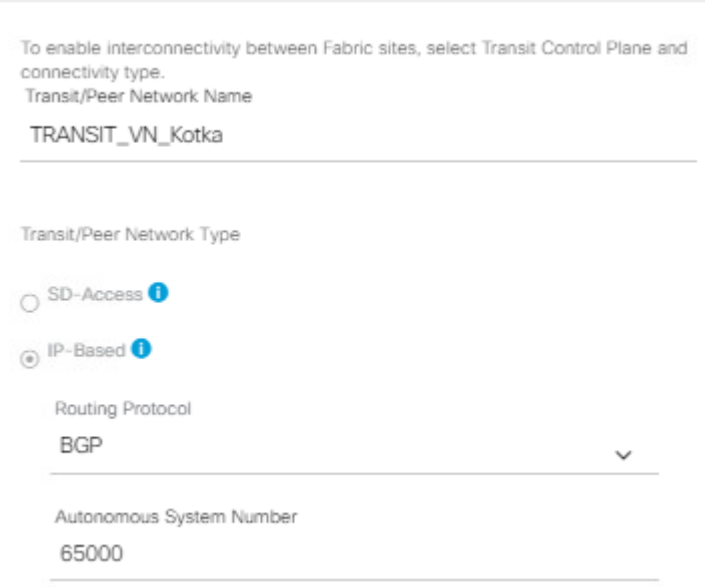
Tämän jälkeen fuusioreitittimelle lisättiin ko. VLAN:t, VRF VN_Kotka ja BGP-reitityksen asetuksiin tehtiin lisäys, joka muodostaa em. IP-osoitteiden kanssa BGP-naapuruuden. Border-laitteisiin muutoksia ei tarvinnut tehdä käsin, sillä SDA-ratkaisun mukaan konfiguroitiin laitteet DNA Centerin toimesta.

```
vrf definition VN_Kotka
 rd 1:4099
 !
 address-family ipv4
  route-target export 1:4099
  route-target import 1:4099
 exit-address-family

interface Vlan3001
 vrf forwarding VN_Kotka
 ip address 172.16.255.2 255.255.255.252
end

interface Vlan3002
 vrf forwarding VN_Kotka
 ip address 172.16.255.6 255.255.255.252
end

address-family ipv4 vrf VN_Kotka
  redistribute static
  neighbor 172.16.255.1 remote-as 66000
  neighbor 172.16.255.1 update-source Vlan3001
  neighbor 172.16.255.1 activate
```

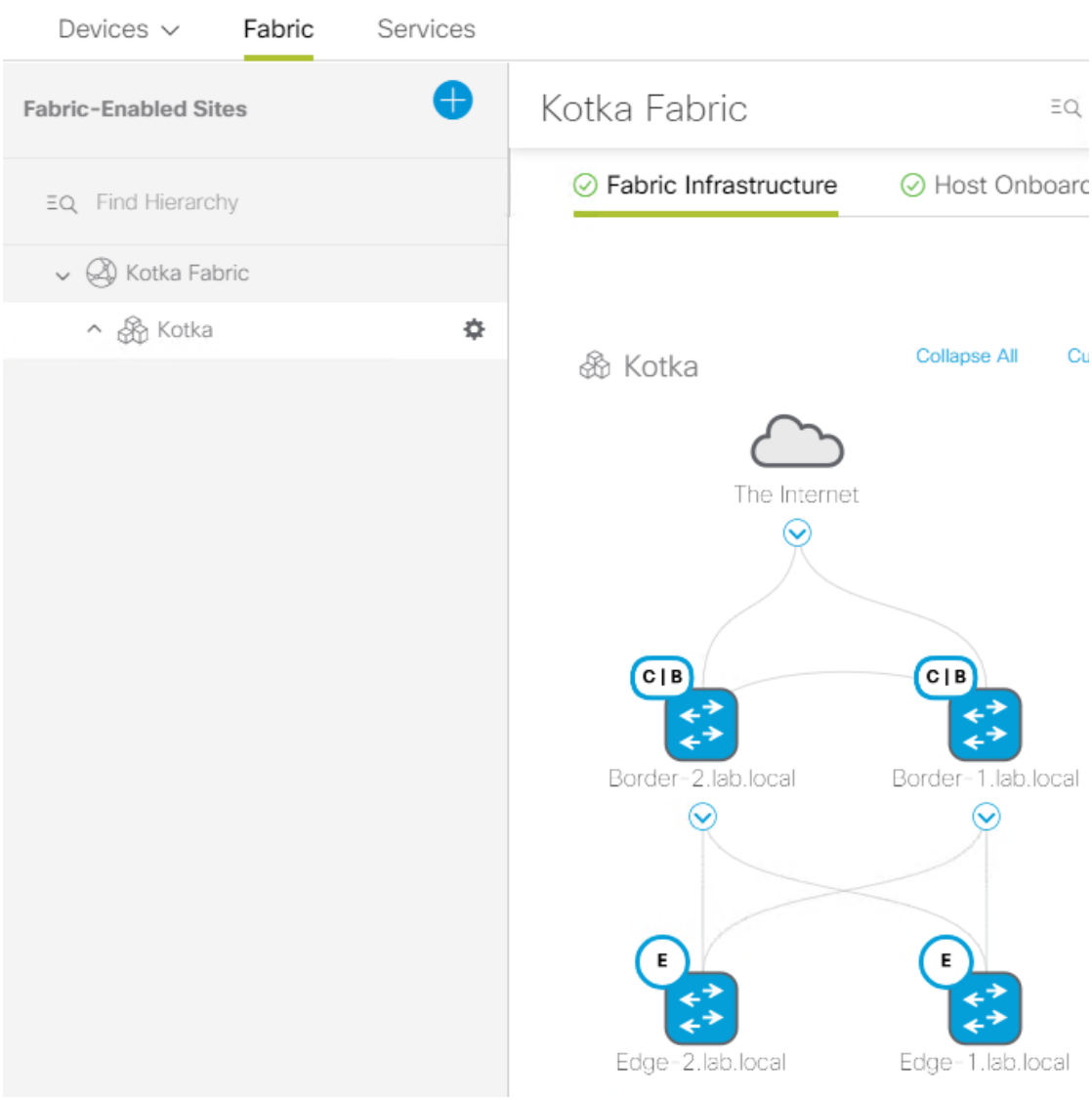
```
 neighbor 172.16.255.5 remote-as 66000
 neighbor 172.16.255.5 update-source Vlan3002
 neighbor 172.16.255.5 activate
 default-information originate
exit-address-family
```



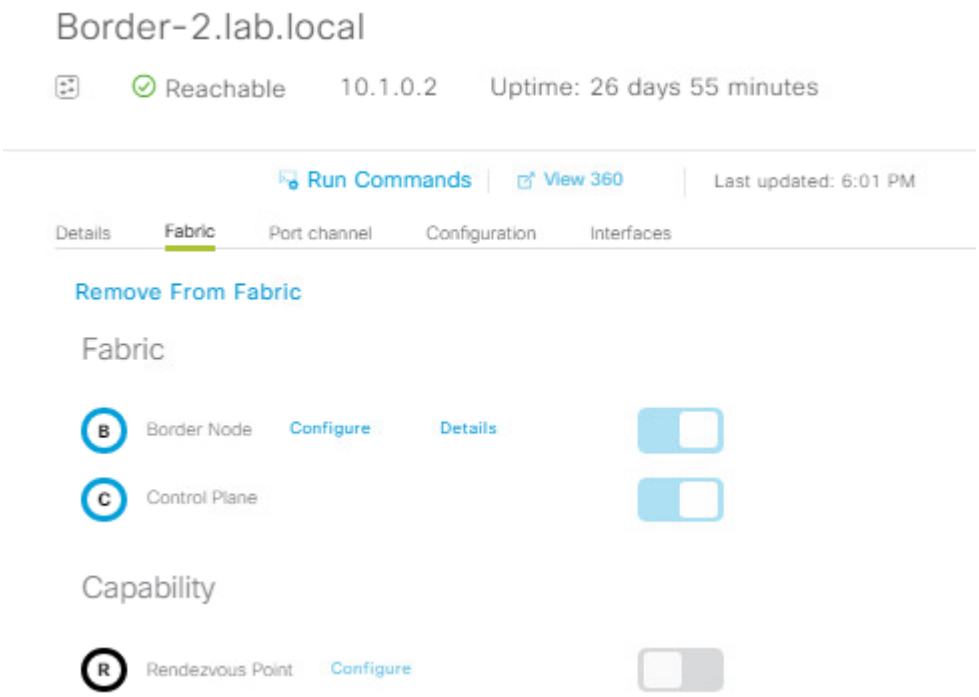
Kuva 50. SDA-verkon luonnissa valitut VN:t



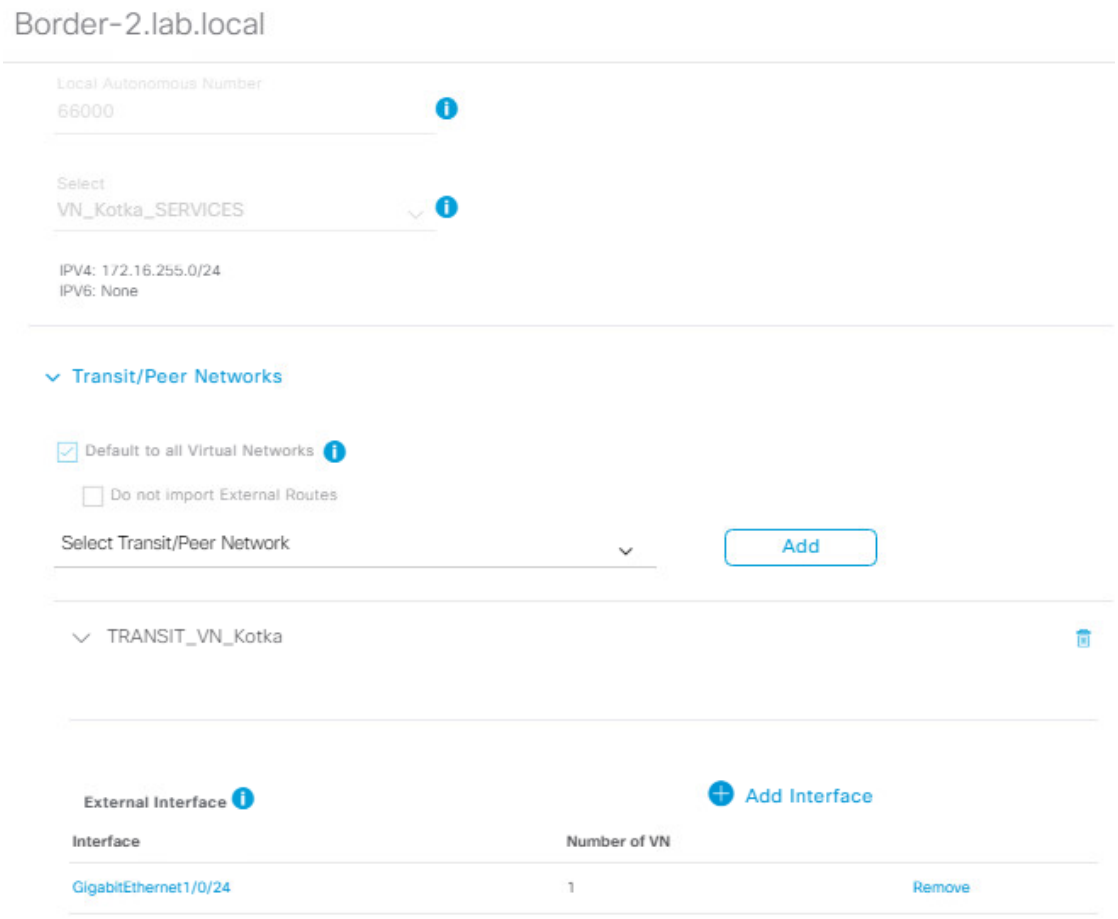
Kuva 51. Transit-verkon määritykset



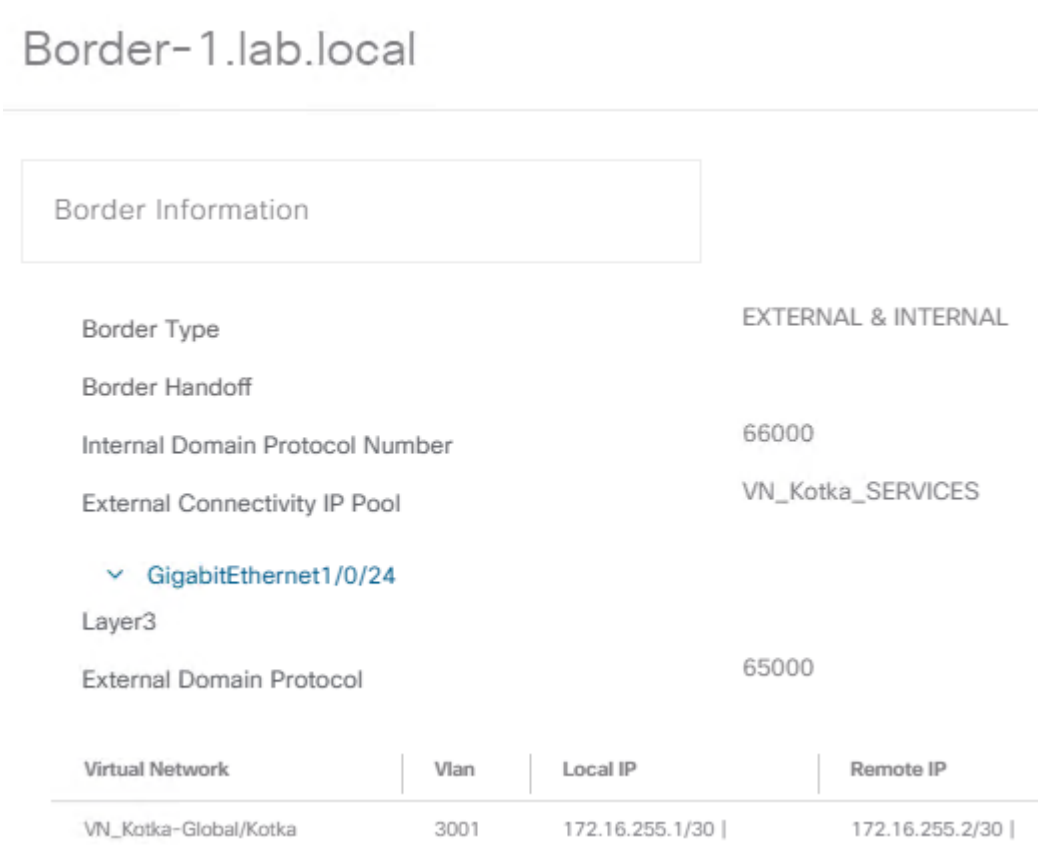
Kuva 52. Kotka Fabric topologia



Kuva 53. Border-laitteen määritys SDA-verkkoon

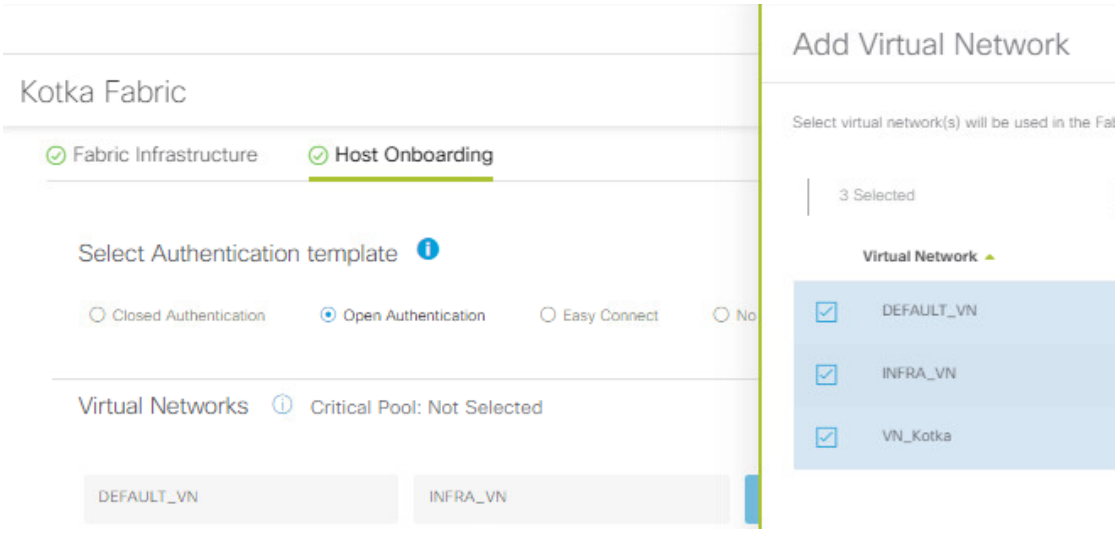


Kuva 54. Border-laitteen asetukset

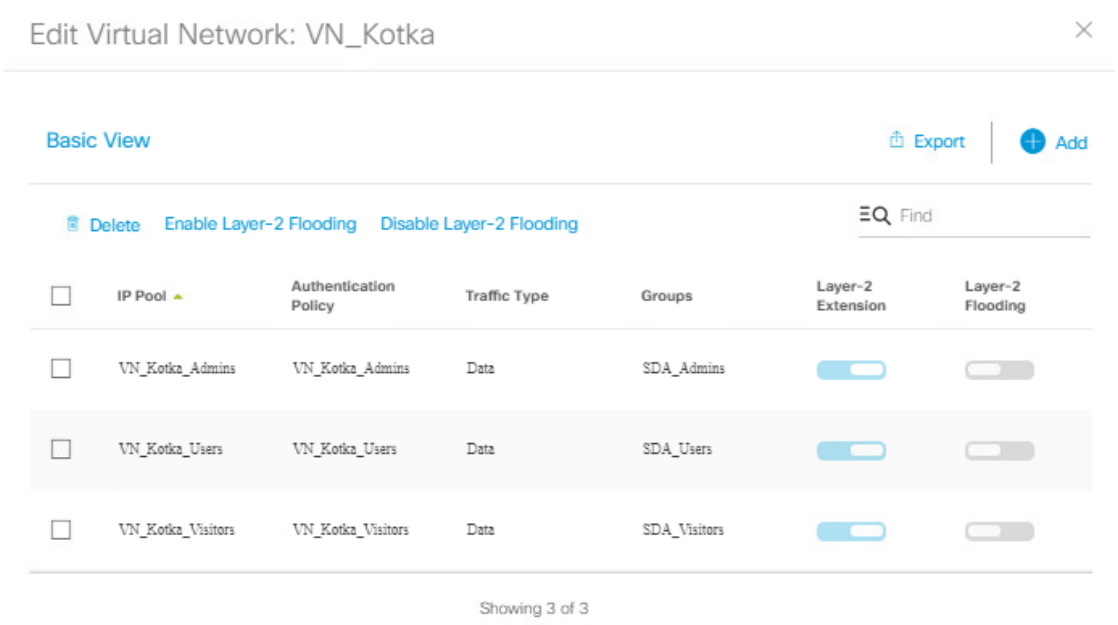


Kuva 55. Border-laitteen Control-Plane-tiedot

SDA-verkon komponenttien määritysten jälkeen asetettiin Kotka Fabric -SDA-verkkoon pääsypolitiikka Host Onboarding -välilehdeltä niin, että mallina toimi ”Open Authentication”, jotta mm. tulostimet ym. voidaan lisätä SDA-verkon edge-kytkimille ja ne saavat verkon ISE:ssä määriteltyjen politiikkojen mukaan (SDA_Visitors), jos eivät kykene tunnistamaan itseään 802.1X-protokollan avulla. Tämän lisäksi lisättiin SDA-verkkoon kaikki VN:t (kuva 56) ja VN_Kotka-verkon asetuksiin luotiin määritykset käytettäville verkoille ja niiden politiikoille (kuva 57). Politiikassa määritelty kenttä vastaa ISE:ssä määriteltyä Authorization-profiiliin määriteltyä ID/Name-kenttää.



Kuva 56. Host onboarding asetukset ja VN lisääminen



Kuva 57.VN_Kotka VN:n host-onboarding määrittelyt

4.5 Testaus

Kun SDA oli rakennettu, testattiin käyttäjän liittymistä verkkoon kolmen koneen avulla niin, että yhdestä koneesta otettiin 802.1X-protokollapino pois käytöstä. Heti verkkopiuhan kytkettyäni sai kone IP-osoitteen VN_Kotka_SDA_Visitors DHCP-osoitealueesta. Koneella oli yhteys normaalisti Internetiin. Kun verkkoon liitettiin 802.1X-kykenevä toinen tietokone ja kirjauduttiin käyttäen tunnusta, ohjattiin onnistuneen kirjautumisen yhteydessä käyttäjä oikeaan verkon segmenttiin. Tässä tapauksessa käyttäjä ohjattiin SDA_Users-segmenttiin. Seuraavaksi koetettiin muodostaa yhteys koneiden välille ja määriteltyjen politiikkojen mukaan ei yhteys toiminut, joten politiikka

toimi kuten piti. Seuraavaksi lisättiin kolmas 802.1X-kykenevä kone verkkoon ja tunnistettiin SDA_Users tunnuksilla. Yhteyttä testattiin toisen koneen kanssa ja yhteys muodostui politiikkojen mukaisesti niin kuin pitikin. Seuraavaksi ensimmäisestä koneesta laitettiin 802.1X-protokollapino päälle ja tunnistauduttiin SDA_Admins tunnuksilla. Testit laitteiden välillä toimi juuri niin kuin oli politiikoissa määritelty.

Seuraava testivaihe oli mobiliteetti ja käyttäjää vaihdettiin kytkimestä toiseen ja aina IP-osoite seurasi mukana, kuten oletettiin. Mobiliteetin lisäksi testattiin verkon palautuvuussietokykyä ja otettiin linkkikaapeli Border-1- ja Edge-2-laitteen väliltä pois. Tämän jälkeen liikenne kuitenkin kulki normaalisti aiheuttaen vain pienen katkon. Sama toistettiin myös niin, että edellinen linkki kytkettiin takaisin ja Border-1- ja Border-2-välinen linkki katkaistiin. Lopputulos oli sama.

5 TULOKSET JA JOHTOPÄÄTÖKSET

5.1 Tulokset

Käytännön toteutuksen lopputuloksena oli onnistunut verkon rakennus käyttäen SDA-tekniikkaa. Työtä tehdessä muodostui myös sivutuotoksena ikään kuin kuvallinen ohjekirja SDA-ratkaisun implementointiin työn ohjelmistopohjaisen luonteen vuoksi. Verkon pääsynhallintaa testattaessa saavutettiin myös asetetut vaatimukset. 802.1X-porttikohtaiseen autentikointiin kykenemättömät laitteet saivat pääsyn verkkoon, mutta eivät kuitenkaan pystyneet keskustelemaan muiden verkon laitteiden kanssa. Taas puolestaan 802.1X-porttikohtaiseen autentikointiin kykenevät laitteet määriteltiin suunniteltujen politiikkojen mukaan oikeisiin verkon segmentteihin ja niille suunnitellut käyttöoikeuspolitiikat toimivat halutusti identiteetin perusteella IP-osoite-politiikkojen sijaan.

Käyttäjän liikkuvuutta testattaessa todettiin SDA-verkko toimivaksi ratkaisuksi ja näin ollen tulevaisuudessa voitaisiin hyvinkin hyödyntää testattua SDA-tekniikkaa helpottamaan käyttäjän tunnistusta verkossa, kun käyttäjän tunnistetiedot eivät olisi sidottu paikkaan ja paikan IP-osoitteeseen. Sen lisäksi, että IP-osoite liikkui käyttäjän mukana, liikkuivat myös suunnitellut käyttöoikeuspolitiikat. Tekniikka mahdollistaisi myös helpomman IP-suunnittelun, sillä esimerkiksi IP-osoitealueita ei tarvitsisi jakaa sijaintien mukaan, vaan ne voitaisiin jakaa mm. käyttäjäryhmien mukaan.

Verkkolinkkejä testattaessa huomattiin, että verkon konvergenssiaika oli huomattavan nopea, johtuen syystä, että sitä ei toteuteta STP-protokollan avulla, vaan sitä ohjaa kontrollitasolla oleva IGP. Tämä jätti verkosta ylimääräiset tekniikat pois ja mahdollisti myös paremman kaistanleveyden kaikkien porttien ohjatessa liikennettä.

Käytännön toteutusta hieman haittasi liian vähäisillä resursseilla varustettu virtualisointialusta ja myös luultavasti siitä syystä lopulta ISE–DNA Center -välinen yhteys lakkasi toimimasta. Tämä ei onneksi vaikuttanut politiikkoihin työn tekovaiheessa. Jatkokehitystä ajatellen tulisi ISE asentaa järeämpään laitteeseen, jotta vältyttäisiin järjestelmän satunnaiselta kaatuilulta.

Verkkoa suunnitellessa selvittiin suhteellisen vähällä konfiguroinnilla, mutta SDA-ratkaisun käyttöönoton jälkeen kytkinten erittäin pitkiä konfiguraatioita tutkiessa huomattiin, että jos SDA-ratkaisu otettaisiin yrityksessä käyttöön, vaatisi se kustannuksia henkilöstön koulutuksien suhteen. Kytkimeltä valmista SDA-konfiguraatiota katsoessa, sisälsi se useita satoja rivejä ja siitä syystä konfiguraatiota ei ole listattuna tähän työhön.

Toki myöskään SDA-ratkaisun implementointi ei välttämättä ole halpa investointi, mutta jos se toteutettaisiin pala palalta, jäisi loppujen lopuksi työntekijöille aikaa enemmän verkon ylläpitoon suunnittelun, vianhallinnan ym. sijasta.

5.2 Johtopäätökset

Opinnäytetyön tavoitteena oli tutustua SDA-tekniikkaan ja luoda tilaajayritykselle parempi käsitys SDA-tekniikan hyödyistä ja haitoista. Tilaajayritys oli aiemmin saanut esitietoa SDA-tekniikasta Cisco Systemsiltä ja sen pohjalta tehtiin päätös toteuttaa soveltuvuusselvitys SDA-ratkaisusta ns. prototyyppinä.

Opinnäytetyön teoriaosuudessa käsiteltiin Ciscon SDA-ratkaisun kannalta olennaisia protokollia ja verkon komponentteja, joita tekniikka vaatii. Teoriaosuuden tavoitteena oli myös tutustua millä tavalla SDA-ratkaisu eroaa tavallisella tavalla rakennetusta yritysverkosta. Toisessa osiossa suunniteltiin SDA-ratkaisun käyttöönotto ja toteutettiin tehty suunnitelma käytännössä. Toisen

osion tavoitteena oli myös tuoda esille SDA-ratkaisun tuomat mahdolliset hyödyt ja haitat.

Alkuperäisen suunnitelman mukaan työ oli tarkoitus toteuttaa kesän 2019 aikana lainalaitteilla, joka olisi asettanut työlle huomattavasti lyhyemmän aikataulun. Yhteistyön tuloksena tilaajayritykseen päätettiin kuitenkin hankkia oma Cisco DNA Center ja näin ollen kesän 2019 aikana toteutettiin vain työn suunnitteluosio. Toteutuksen aikataulu venyi toiseen otteeseen, kun työtä varten hankitut Cisco 9300 -sarjan kytkimet tarvittiin yrityksen muuhun käyttöön. Näin ollen jouduttiin odottamaan uusien vastaavien laitteiden toimitusta, joka viivästytti työn valmistumista.

Opinnäytetyötä tehdessäni saavutin itselleni asettamat oppimistavoitteet. Teoriaosuutta tehdessäni opin kattavasti SDA-tekniikan protokollista ja komponenteista ja niiden toiminnasta. Käytännön toteutuksen aikana opin monipuolisesti Cisco ISE- ja DNA Center -ympäristöjen käyttöä ja niiden ominaisuuksista. Käytännön toteutuksessa yhdistyi myös suuri osa opinnoissani käytyjä tekniikoita ja ympäristöjä, kuten reititysprotokollat, kytkinverkot ja virtualisointi- sekä palvelinympäristöt. Opinnäytetyön tekeminen oli itselleni mielekästä, sillä itse aihe oli minulle uusi ja erittäin mielenkiintoinen. Työn tekeminen antoi minulle myös mahdollisuuden oppia lisää jo koulussa opiskeltuihin asioihin. Opinnäytetyö onnistui mielestäni hyvin, vaikka alkuperäiseen suunnitelmaan tuli työn aikana muutoksia ja olin hieman aliarvioinut työn laajuutta.

Alkuperäisiä työn tutkimuskysymyksiä tutkittaessa voidaan tulla tulokseen, että työ oli onnistunut ja vastasi kysymyksiin. Työssä käsiteltiin mikä SDA-ratkaisu on, miten se toimii ja mitä vaiheita sen käyttöönotto vaatii. Tämän lisäksi saatiin selville minkälaisia hyötyjä tai haittoja SDA-ratkaisusta voisi olla, miten SDA-ratkaisua voitaisiin hyödyntää tulevaisuudessa ja minkälaisia jatkokehityshankkeita työn pohjalta voitaisiin toteuttaa.

Yhtenä jatkokehityshankkeena näkisin, että voitaisiin tutkia, miten SDA-ratkaisu soveltuu yritysverkkoihin langattomien verkkojen osalta.

Toisena kehityshankkeena olisi hyvä tutustua, mitä mahdollisuuksia Cisco DNA Centerin API-rajapinta (*Application Programming Interface*) mahdollistaisi. Esimerkkinä näkisin, että sen avulla voitaisiin tutkia esimerkiksi verkon kytkinten porttien käyttöä ja jos porteissa ei ole ollut liikennettä tietyn ajan sisällä, voitaisiin portit asettaa ohjelmallisesti alas-tilaan.

Kolmantena kehityshankkeena näkisin erilaiset integraatiot esim. ITSM- ja IPAM-järjestelmiin (*IT Service Management*, *IP Address Management*), kuten ServiceNow ja Infoblox.

LÄHTEET

Cisco (2018a). Campus LAN and Wireless LAN Design Guide. PDF-dokumentti. Saatavissa: https://www.cisco.com/c/dam/en/us/td/docs/solutions/CVD/Campus/CVD-Campus-LAN-WLAN-Design-Guide-2018JAN.pdf [viitattu 25.11.2019].

Cisco (2018b). CISCO VALIDATED DESIGN Software-Defined Access Deployment Guide. PDF-dokumentti. Saatavissa: https://www.cisco.com/c/dam/en/us/td/docs/solutions/CVD/Campus/CVD-Software-Defined-Access-Deployment-Guide-Sol1dot2-2018OCT.pdf [viitattu 11.10.2019].

Cisco (2018c). Software-Defined Access 1.0. PDF-dokumentti. Saatavissa: https://www.cisco.com/c/dam/en/us/solutions/collateral/enterprise-networks/software-defined-access/white-paper-c11-740585.pdf [viitattu 25.11.2019].

Cisco (2019a). Cisco Digital Network Architecture Center Second Generation Appliance Installation Guide, Release 1.3 - Plan the Deployment [Cisco DNA Center]. WWW-dokumentti. Saatavissa: https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/network-automation-and-management/dna-center/1-3/install_guide/M5/b_cisco_dna_center_install_guide_1_3_M5/b_cisco_dna_center_install_guide_1_3_M5_chapter_01.html [viitattu 25.11.2019].

Cisco (2019b). Cisco Identity Services Engine. WWW-dokumentti. Saatavissa: https://www.cisco.com/c/en/us/products/collateral/security/identity-services-engine/data_sheet_c78-656174.html [viitattu 25.11.2019].

Cisco (2019c). Software-Defined Access. WWW-dokumentti. Saatavissa: https://www.cisco.com/c/en/us/solutions/enterprise-networks/software-defined-access/compatibility-matrix.html [viitattu 25.11.2019].

Cisco (2019d). Support - Software-Defined Access - Solution Design Guide. WWW-dokumentti. Saatavissa: https://www.cisco.com/c/en/us/td/docs/solutions/CVD/Campus/sda-sdg-2019oct.html [viitattu 25.11.2019].

Congdon, P., Aboba, B., Smith, A., Roese, J. and Zorn, G. (2003). IEEE 802.1X Remote Authentication Dial-In User Service (RADIUS) Usage Guidelines. WWW-dokumentti. Saatavissa: https://tools.ietf.org/html/rfc3580 [viitattu 25.11.2019].

Farinacci, D., Fuller, V., Meyer, D. and Lewis, D. (2013). The Locator/ID Separation Protocol (LISP). WWW-dokumentti. Saatavissa: https://tools.ietf.org/html/rfc6830 [viitattu 25.11.2019].

Fuller, V. and Farinacci, D. (2013). Locator/ID Separation Protocol (LISP) Map-Server Interface. WWW-dokumentti. Saatavissa: https://tools.ietf.org/html/rfc6833 [viitattu 25.11.2019].

Geier, J, & Geier, JT 2008, Implementing 802.1X Security Solutions for Wired and Wireless Networks, John Wiley & Sons, Incorporated, Hoboken. Saatavissa: https://kaakkuri.finna.fi [viitattu 23.11.2019].

Hill, C., Miller, D., Zacks, D., Suhr, J., Thatikonda, K.K., Karmarkar, K., Hooda, S., Kondalam, S., Prasad, S., Wargo, S., Arena, S., Katkade, V., Pendhakar, V., Rubino, B., Bashir, I., Meek, J., Bhatia, J., Gupta, K., Muralinath, M., DeLong, S. and Ahuja, T. s.a. Cisco Software-Defined Access Enabling intent-based networking 2nd edition. PDF-dokumentti. Saatavissa: https://www.cisco.com/c/dam/en/us/products/se/2018/1/Collateral/nb-06-software-defined-access-ebook-en.pdf [viitattu 25.11.2019].

Kananen, J. 2015. Online research for preparing your thesis: A guide for conducting qualitative and quantitative research online. Jyväskylä: JAMK University of Applied Sciences.

Langemak, J. (2012). LISP – Proxy xTR (PxTR). WWW-dokumentti. Saatavissa: http://www.dasblinkenlichten.com/lisp-proxy-xtr-pxtr/ [viitattu 25.11.2019].

Lewis, D., Meyer, D., Farinacci, D. and Fuller, V. (2013). Interworking between Locator/ID Separation Protocol (LISP) and Non-LISP Sites. WWW-dokumentti. Saatavissa: https://tools.ietf.org/html/rfc6832 [viitattu 25.11.2019].

Mahalingam, M., Duda, K., Agarwal, P., Keeger, L., Sridhar, T., Bursell, M., Wright, C. and Dutt, D. (2014). Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks. WWW-dokumentti. Saatavissa: https://tools.ietf.org/html/rfc7348 [viitattu 11.11.2019].

Mahesh Gupta (2017). 5 Reasons why businesses need Software-Defined Access. WWW-dokumentti. Saatavissa: https://apjc.thecisconetwork.com/site/content/lang/en/id/7942 [viitattu 28.11.2019].

Microchip Technology, Inc. (2019). TCP/IP Five-Layer Software Model Overview - Developer Help. WWW-dokumentti. Saatavissa: https://microchipdeveloper.com/tcpip:tcp-ip-five-layer-model [viitattu 25.11.2019].

Microsoft (2017). Install or Uninstall Roles, Role Services, or Features. WWW-dokumentti. Saatavissa: https://docs.microsoft.com/en-us/windows-server/administration/server-manager/install-or-uninstall-roles-role-services-or-features [viitattu 26.11.2019].

Moy, J. (1998). OSPF Version 2. WWW-dokumentti. Saatavissa: https://tools.ietf.org/html/rfc2328 [viitattu 26.11.2019].

Oran, D.R. (1990). OSI IS-IS Intra-domain Routing Protocol. WWW-dokumentti. Saatavissa: https://tools.ietf.org/html/rfc1142 [viitattu 26.11.2019].

Salonen, K. (2013). NÄKÖKULMIA TUTKIMUKSELLISEEN JA TOIMINNALLISEEN OPINNÄYTETYÖHÖN. PDF-dokumentti. Saatavissa: http://julkaisut.turkuamk.fi/isbn9789522163738.pdf [viitattu 28.11.2019].

Smith, M., Candula, R.R. and Appala, S. (2019). Scalable-Group Tag eXchange Protocol (SXP). WWW-dokumentti. Saatavissa: https://tools.ietf.org/html/draft-smith-kandula-sxp-09 [viitattu 25.11.2019].

Smith, M. and Kreeger, L. (2017). VXLAN Group Policy Option. WWW-dokumentti. Saatavissa: https://tools.ietf.org/html/draft-smith-vxlan-group-policy-03 [viitattu 25.11.2019].

VMware (2018). VMware ESXi Installation and Setup. WWW-dokumentti. Saatavissa: https://docs.vmware.com/en/VMware-vSphere/6.7/vsphere-esxi-67-installation-setup-guide.pdf [viitattu 26.11.2019].

Alusverkon aktiivilaitteiden pohjakonfiguraatiot

SDA-Router:

```
ip routing
system mtu 9100

vlan 900
  name DNA-Cluster

vlan 1000
  name lab.local

vlan 1001
  name kotka.local

interface Loopback 0
  ip address 10.1.0.0 255.255.255.255

interface GigabitEthernet1/0/1
  description To Border-1
  switchport mode trunk

interface GigabitEthernet1/0/2
  description To Border-2
  switchport mode trunk

interface GigabitEthernet1/0/5
  description DNA CIMC
  switchport access vlan 1000
  switchport mode access

interface GigabitEthernet1/0/7
  description DNA MGMT
  switchport access vlan 1000
  switchport mode access

interface GigabitEthernet1/0/47
  description ESXi Host
  switchport mode trunk
  switchport nonegotiate

interface GigabitEthernet1/1/1
  description DNA Enterprise port
  switchport access vlan 1000
  switchport mode access

interface GigabitEthernet1/1/2
  description DNA Cluster port
  switchport access vlan 900
  switchport mode access
```

```
interface Vlan1000
  description lab.local
  ip address 10.2.1.1 255.255.255.0
  ip ospf 1 area 0

interface Vlan1001
  description kotka.local
  ip address 172.16.0.1 255.255.255.0

router ospf 1
  router-id 10.1.0.0
  redistribute bgp 65000 subnets
  passive-interface default
  no-passive interface Vlan1000

router bgp 65000
  bgp router-id 10.1.0.0
  bgp log-neighbor-changes
  neighbor 10.2.1.71 remote-as 65000
  neighbor 10.2.1.71 update-source Vlan1000
  neighbor 10.2.1.71 remote-as 65000
  neighbor 10.2.1.71 update-source Vlan1000
  address-family ipv4
    network 10.1.0.0 mask 255.255.255.255
    network 10.2.1.1 mask 255.255.255.255
    redistribute ospf 1
    neighbor 10.2.1.71 activate
    neighbor 10.2.1.72 activate
    default-information originate
  exit-address-family
```

Border-1:

```
ip routing
system mtu 9100

vlan 1000
  name lab.local

vlan 1001
  name kotka.local

interface Loopback 0
  ip address 10.1.0.1 255.255.255.255
  ip router isis

interface Vlan1000
  ip address 10.2.1.71 255.255.255.0

interface Port-Channel 1
  description To Border-2
  no switchport
```

```
  ip address 10.1.10.0 255.255.255.254
  ip router isis
  isis network point-to-point
  dampening
  logging event link-status
  load-interval 30
  bfd interval 500 min_rx 500 multiplier 3
  no bfd echo

interface GigabitEthernet1/0/1
  description To Edge-1
  no switchport
  dampening
  ip address 10.1.10.2 255.255.255.254
  ip router isis
  logging event link-status
  load-interval 30
  bfd interval 500 min_rx 500 multiplier 3
  no bfd echo
  isis network point-to-point

interface GigabitEthernet1/0/3
  description To Edge-2
  no switchport
  dampening
  ip address 10.1.10.6 255.255.255.254
  ip router isis
  logging event link-status
  load-interval 30
  bfd interval 500 min_rx 500 multiplier 30
  no bfd echo
  isis network point-to-point

interface GigabitEthernet1/0/21
  no switchport
  no ip address
  channel-group 1 mode active

interface GigabitEthernet1/0/22
  no switchport
  no ip address
  channel-group 1 mode active

interface GigabitEthernet1/0/24
  description To Router
  switchport mode trunk

router isis
  net 49.0001.0000.0000.0001.00
  metric-style wide
  log-adjacency-changes
  nsf ietf
  passive-interface GigabitEthernet1/0/24
```

```
  default-information originate
  bfd all-interfaces

router bgp 66000
  bgp router-id 10.1.0.1
  bgp log-neighbor-changes
  neighbor 10.2.1.1 remote-as 65000
  neighbor 10.2.1.1 update-source Vlan1000
  neighbor 10.1.10.1 remote-as 66000
  neighbor 10.1.10.1 update-source Loopback0
  address-family ipv4
    network 10.1.0.1 mask 255.255.255.255
    network 10.2.1.71 mask 255.255.255.255
    aggregate-address 10.1.10.0 255.255.255.0 summary-only
    redistribute isis level-1-2
    neighbor 10.2.1.1 activate
    neighbor 10.1.10.1 activate
    maximum-paths 2
  exit-address-family
```

Border-2:

```
ip routing
system mtu 9100

vlan 1000
  name lab.local

vlan 1001
  name kotka.local

interface Loopback 0
  ip address 10.1.0.2 255.255.255.255
  ip router isis

interface Vlan1000
  ip address 10.2.1.72 255.255.255.0

interface Port-Channel 1
  description To Border-1
  no switchport
  ip address 10.1.10.1 255.255.255.254
  ip router isis
  isis network point-to-point
  dampening
  logging event link-status
  load-interval 30
  bfd interval 500 min_rx 500 multiplier 3
  no bfd echo

interface GigabitEthernet1/0/1
  description To Edge-1
  no switchport
```

```
  dampening
  ip address 10.1.10.8 255.255.255.254
  ip router isis
  logging event link-status
  load-interval 30
  bfd interval 500 min_rx 500 multiplier 3
  no bfd echo
  isis network point-to-point

interface GigabitEthernet1/0/3
  description To Edge-2
  no switchport
  dampening
  ip address 10.1.10.4 255.255.255.254
  ip router isis
  logging event link-status
  load-interval 30
  bfd interval 500 min_rx 500 multiplier 30
  no bfd echo
  isis network point-to-point

interface GigabitEthernet1/0/21
  no switchport
  no ip address
  channel-group 1 mode active

interface GigabitEthernet1/0/22
  no switchport
  no ip address
  channel-group 1 mode active

interface GigabitEthernet1/0/24
  description To Router
  switchport mode trunk

router isis
  net 49.0001.0000.0000.0002.00
  metric-style wide
  log-adjacency-changes
  nsf ietf
  passive-interface GigabitEthernet1/0/24
  default-information originate
  bfd all-interfaces

router bgp 66000
  bgp router-id 10.1.0.2
  bgp log-neighbor-changes
  neighbor 10.2.1.1 remote-as 65000
  neighbor 10.2.1.1 update-source Vlan1000
  neighbor 10.1.10.0 remote-as 66000
  neighbor 10.1.10.0 update-source Loopback0
  address-family ipv4
    network 10.1.0.1 mask 255.255.255.255
```

```
    network 10.2.1.71 mask 255.255.255.255
    aggregate-address 10.1.10.0 255.255.255.0 summary-only
    redistribute isis level-1-2
    neighbor 10.2.1.1 activate
    neighbor 10.1.10.0 activate
    maximum-paths 2
  exit-address-family
```

Edge-1:

```
ip routing
system mtu 9100

interface Loopback0
  ip address 10.1.0.3 255.255.255.255
  ip router isis

interface GigabitEthernet1/0/21
  description To Border-1
  no switchport
  dampening
  ip address 10.1.10.3 255.255.255.254
  ip router isis
  load-interval 30
  bfd interval 500 min_rx 500 multiplier 3
  no bfd echo
  isis network point-to-point

interface GigabitEthernet1/0/23
  description To Border-2
  no switchport
  dampening
  ip address 10.1.10.9 255.255.255.254
  ip router isis
  load-interval 30
  bfd interval 500 min_rx 500 multiplier 3
  no bfd echo
  isis network point-to-point

router isis
  net 49.0001.0000.0000.0003.00
  metric-style wide
  log-adjacency-changes
  nsf ietf
  bfd all-interfaces
```

Edge-2:

```
ip routing
system mtu 9100

interface Loopback0
```

```
  ip address 10.1.0.4 255.255.255.255
  ip router isis

interface GigabitEthernet1/0/21
  description To Border-1
  no switchport
  dampening
  ip address 10.1.10.7 255.255.255.254
  ip router isis
  load-interval 30
  bfd interval 500 min_rx 500 multiplier 3
  no bfd echo
  isis network point-to-point

interface GigabitEthernet1/0/23
  description To Border-2
  no switchport
  dampening
  ip address 10.1.10.5 255.255.255.254
  ip router isis
  load-interval 30
  bfd interval 500 min_rx 500 multiplier 3
  no bfd echo
  isis network point-to-point

router isis
  net 49.0001.0000.0000.0004.00
  metric-style wide
  log-adjacency-changes
  nsf ietf
  bfd all-interfaces
```