TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

EETU PIKI
HIP-PROTOKOLLAN KÄYTTÖMAHDOLLISUUDET ESINEIDEN INTERNETISSÄ

Kandidaatintyö

Tarkastaja: Yliopisto-opettaja Mikko Salmenperä

TIIVISTELMÄ

EETU PIKI: HIP-protokollan käyttömahdollisuudet esineiden internetissä
Tampereen teknillinen yliopisto
Kandidaatintyö, 29 sivua
Marraskuu 2018
Automaatiotekniikan koulutusohjelma
Pääaine: Automaatiotekniikka
Tarkastaja: Yliopisto-opettaja Mikko Salmenperä
Avainsanat: Host Identity Protocol, esineiden internet

Host Identity Protocol (HIP) on verkottamisarkkitehtuuri, jolla voidaan toteuttaa esineiden internetille hyödyllisiä toiminnallisia ja tietoturvaa parantavia ominaisuuksia. HIP-protokollan keskeinen periaate on luopua IP-osoitteen päätelaitteen identifioivasta roolista, ja siirtyä turvallisten kryptografisten identiteettien käyttöön. Yhteyden sitominen luotettaviin identiteetteihin IP-osoitteiden sijasta parantaa sijaintianonymiteettiä, ja mahdollistaa helposti toteutettavan saumattoman yhteyden laitteen IP-osoitteen vaihtuessa.

HIP:n laajennuksilla voidaan toteuttaa verkon mobiliteetin hallinta, laitteiden moniverkotus ja monilähetys. Kirjallisuuskatsauksen perusteella HIP-pohjaiselle mobiliteetille ja monilähetykselle voidaan löytää toteutusmalleja ja testituloksia, jotka osoittavat protokollan arkkitehtuurisen toimivuuden. HIP:n normaalin yhteydenmuodostuksen korkea laskentatehovaatimus aiheuttaa haasteita esineiden internetissä, jossa laitteiden laskentakapasiteetit voivat olla erittäin rajoittuneita. CHIP ja CD-HIP ovat laskentakuormitusta hajauttavia HIP-malleja, jotka on kehitetty vähäresurssisille IoT-laitteille.

HIP:n käyttö on edelleen vähäistä sen teknisestä edistyksellisyydestä huolimatta, eikä sen nopeasta yleistymisestä tai kaupallisesta käytöstä ole viitteitä. Saatavilla olevia avoimen lähdekoodin HIP-ohjelmistoja ovat OpenHIP ja HIP for Linux.

SISÄLLYS

LYHENTEET JA MERKINNÄT

6LoWPAN	IPv6 over Low-Power Wireless Personal Area Networks, matalan tehon langaton likiverkko
AMIKEY	Adapted Multimedia Internet KEYing, avaintenhallintaprotokolla
BEET	Bound End-To-End Tunnel, ESP-formaatti
CD-HIP	Compressed and Distributed Host Identity Protocol
CHIP	Collaborative Host Identity Protocol
DDoS	Distributed Denial of Service, hajautettu palvelunestohyökkäys
DNS	Domain Name System, internetin nimipalvelujärjestelmä
DoS	Denial of Service, palvelunestohyökkäys
DTLS	Datagram Transport Layer Security
ESP	Encapsulated Security Payload, pakettivirtojen salaamiseen käytetty protokolla
HI	Host Identity, päätelaitteen identiteetti
HIP	Host Identity Protocol, päätelaitteen identifiointiprotokolla
HIP BEX	HIP Base Exchange, HIP:n yhteydenmuodostusproseduuri
HIP DEX	HIP Diet EXchange, HIP:n kevennetty yhteydenmuodostusproseduuri
HIPL	HIP for Linux, HIP-toteutus Linuxille
HIT	Host Identity Tag, identiteettitunnistetagi
IEEE	Institute of Electrical and Electronics Engineers, kansainvälinen tekniikan alan järjestö
IETF	Internet Engineering Taskforce, kansainvälinen internetstandardeja kehittävä organisaatio
IoT	Internet of Things, esineiden internet
IP	Internet Protocol, verkkokerroksen protokolla
IPsec	Internet protocol security architecture, protokollajoukko internetyhteyden turvaamiseen
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
LISP	Locator/Identifier Separation Protocol, IP-osoitteen kaksoisroolin eriyttävä protokolla
LRVS	Local Rendezvous server, paikallinen rendezvous-palvelin
LSI	Local Scope Identifier, paikallinen identiteettitunniste
MIKEY	Multimedia Internet KEYing, avaintenhallintaprotokolla
MITM	Man-In-The-Middle Attack, mies välissä -hyökkäys

NAPT	Network Address and Port Translation, IP-osoitteita ja porttinumeroita muuntava osoitteenmuunnostekniikka
NAT	Network Address Translation, IP-osoitteita piilottava tai säästävä osoitteenmuunnostekniikka
ORCHID	Overlay Routable Cryptographic Hash Identifier
RSA	Rivest-Shamir-Adleman –salausalgoritmi
RVS	Rendezvous Server, rendezvous-palvelin
SHA	Secure Hash Algorithm, kryptograafinen tiivistefunktio
SPI	Security Parameter Index, IPsec-otsikon identifiointitunniste
SSL	Secure Sockets Layer, TLS-salausprotokollan edeltäjä
TCP	Transmission Control Protocol, kuljetuskerroksen tiedonsiirtoprotokolla
TLS	Transport Layer Security, tietoliikenteen salausprotokolla
UDP	User Datagram Protocol, kuljetuskerroksen protokolla
WiFi	Wireless Fidelity, IEEE 802.11 standardin langaton lähiverkkoteknologia
WLAN	Wireless Local Area Network, langaton lähiverkko

1. JOHDANTO

Transmission Control Protocol (TCP) ja Internet Protocol (IP) ovat internetin kaksi tärkeintä protokollaa, joiden yhdistelmä tunnetaan TCP/IP:nä. TCP huolehtii tietoliikennepakettien kuljetuksesta ja IP määrittää tavan, jolla paketit lähetetään ja vastaanotetaan. Nelikerroksisen internetin protokollapinon muodostavat sovellus-, kuljetus-, verkko- ja linkkikerros. Kuljetus- ja verkkokerrokset ovat tiukasti sidottu toisiinsa, joten niillä toimivien prokollien toisistaan riippumaton kehittäminen ei ole mahdollista (Gurtov 2008, s. 8).

Host Identity Protocol (HIP) sai alkunsa Robert Moskowitzin (1999) luonnoksesta, joka esitteli tavan siirtää julkisen avaimen kryptografiaan perustuva identiteetti. Julkisen avaimen salauksella tarkoitetaan epäsymmetristä salausta, jossa tiedon salaukseen ja purkamiseen käytetään eri avaimia. HIP:n ensimmäinen vakaa versio julkistettiin vasta vuonna 2007 (Nikander et al. 2010).

HIP kehitettiin paikkamaan yleisesti käytössä olevan TCP/IP-mallin turvallisuuspuutteita sekä helpottamaan moniverkotusta (multihoming) ja laitteiden mobiliteettia (mobility). Internet on rakennettu toimimaan TCP/IP-mallilla, jossa kaikilla verkossa olevilla laitteilla on IP-osoite. IP-osoitteella on kaksi roolia: identifioiva ja paikantava. HIP:n keskeisenä periaatteena on eriyttää nämä roolit toisistaan siten, että IP-osoitetta käytetään vain laitteen paikannukseen (Moskowitz & Nikander 2006).

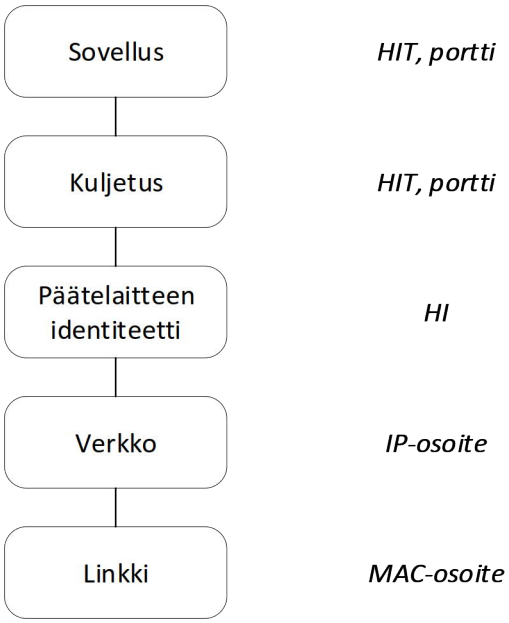
Tässä työssä tutkitaan HIP-protokollan käyttökelpoisuutta esineiden internetissä sekä sen käyttöön liittyviä etuja ja rajoitteita. Lisäksi tavoitteena on arvioida esineiden internetin kannalta oleellisia sovelluskohteita, joissa HIP:tä on tutkittu tai testattu. Tutkimus toteutetaan kirjallisuusselvityksenä.

Työssä esitellään aluksi HIP:n tärkeimmät toimintaperiaatteet. Protokollan laajuudesta ja monimutkaisuudesta johtuen siitä käsitellään vain esineiden internetin toiminnan ja tietoturvallisuuden kannalta oleellisimmat kohdat. Protokollan toiminnan esittely auttaa lukijaa ymmärtämään HIP:n verkon toiminnalle hyödylliset ja tietoturvaa parantavat ominaisuudet.

Työ painottuu HIP:n toiminnan yksityiskohtaisen kuvauksen sijasta enemmän sen tuomiin tietoturvaparannuksiin ja mahdollisiin sovelluskohteisiin. Luku 3 käsittelee HIP:n mahdollisuuksia moniverkotuksen ja mobiliteetin parantamisen suhteen, jotka ovat tärkeitä ominaisuuksia esineiden internetin (IoT) kannalta. Lisäksi käsitellään HIP:n mahdollisuutta toimia IPv4- ja IPv6–verkkojen yhdistäjänä. Luku 4 sisältää HIP:n tietoturvaa parantavat ominaisuudet. Tärkeimpiä käsiteltäviä aiheita ovat yksityisyysparannukset sekä kahdelta yleiseltä hyökkäystyypiltä suojautuminen. Luvussa 5 käsitellään HIPSim++ –simulointialustaa sekä kahta avoimen lähdekoodin HIP-ohjelmistoa. Lopuksi tiivistetään HIP:n hyödyllisyys ja mahdollisuudet IoT:ssa ja automaatiossa.

2. HIP:N RAKENNE JA TOIMINTA

HIP on verkottamisarkkitehtuuri (networking architecture) ja ryhmä siihen kuuluvia protokollia. Protokollapinon keskellä toimimisen vuoksi HIP toimii tekemättä muutoksia jo käytössä oleviin sovelluksiin tai reitittimiin. (Nikander et al. 2010) HIP:n sijainti protokollapinossa ja kullakin tasolla viestintään käytettävät tunnisteet on havainnollistettu kuvassa 2.1. HIP-alikerros yhdistää identiteettitunnistetagit (HIT) IP-osoitteisiin, minkä jälkeen paketit liikkuvat verkkokerroksella ja sen alapuolella, kuten normaalissa IP-protokollapinossa (Gurtov 2008, s. 7). Päätelaitteen (host) sijainti ja identiteetti on siten erotettu toisistaan, sillä IP-osoitetta käytetään vain paikantimena. Identiteettitunnisteen (HI) ja identiteettitunnistetagin muodostaminen esitellään tarkemmin aliluvussa 2.1.



Kuva 2.1 *HIP:n sijainti protokollapinossa. Käännetty lähteestä (Gurtov 2008, s. 8).*

Ilman HIP-alikerrosta toimivassa internetrakenteessa kaksi päänimiavaruutta ovat IP-osoiteavaruus ja verkkotunnusavaruus (domain namespace). Internetin nimipalvelujärjestelmä DNS (Domain Name System) pitää yllä tietokantaa verkkotunnuksista ja niitä vastaavista IP-osoitteista. HIP lisää TCP/IP-protokollapinon

verkko- ja kuljetuskerrosten väliin uuden nimiavaruuden, joka sisältää käyttäjäidentiteettien kryptografiset julkiset avaimet (Nikander et al. 2010).

2.1 Päätelaitteen identiteettitunniste (HI) ja identiteettitunnistetagi (HIT)

HIP-protokollassa identiteetin tunnisteena käytetään päätelaitteen itse muodostamaa yksityisen ja julkisen avaimen paria, joka muodostetaan oletusarvoisesti RSA-algoritmilla. Avainparin julkisen avaimen pituus voi olla 512, 1 024 tai 2 048 bittiä. (Gurtov 2008, s. 46) RSA-algoritmilla (Rivest-Shamir-Adleman) tarkoitetaan julkisen avaimen salausalgoritmia, jossa turvallinen avainpari muodostetaan modulo-aritmetiikkaa ja alkulukujen ominaisuuksia hyödyntäen (Kurose & Ross 2010, s. 726–727). Asymmetrisen avainparin yksityinen avain säilytetään päätelaitteen muistissa (Moskowitz et al. 2015). Identiteettitunnisteelle keskeistä on, että se ei riipu päätelaitteen topologisesta sijainnista verkossa. Topologisella sijainnilla tarkoitetaan laitteen sijaintia verkossa, joka ei suoraan riipu fyysisestä sijainnista.

Päätelaitteen identiteettitunnistetagilla (HIT) tarkoitetaan julkisen avaimen 128-bittistä tiivistettä. Identiteettitunnistetagia käytetään protokollissa esittämään identiteettiä. Tiivisteet ovat vakiopituisia, joten pakettien otsikkokentien koot pysyvät vakioina. Tiivisteen muoto ja rakenne eivät riipu identiteettitunnisteen muodostustavasta (Moskowitz et al. 2015). Tiivisteestä ei pysty selvittämään alkuperäistä julkista avainta, sillä sen muodostamisessa käytetään yksisuuntaista tiivistefunktiota (one-way hash function). Tiivisteen laskentaan käytettyjä algoritmeja ovat typistetyt SHA-1, SHA-384 ja SHA-256 (Moskowitz et al. 2015). SHA (secure hash algorithm) on kryptograafinen tiivistefunktio, jonka tuottaman tiivisteen pituus riippuu käytetystä versiosta (Wu & Irwin 2013). Typistämisellä tarkoitetaan algoritmien tuottamien tiivisteiden lyhentämistä 128-bittisiksi. Gurtovin (2008, s. 46) mukaan todennäköisyys sille, että kahdelle erilaiselle julkiselle avaimelle tuotetaan sama identiteettitunnistetagi, on merkityksettömän pieni.

HIP:n määritelmän (Moskowitz et al. 2015) mukaan HIT:t kuuluvat RFC7343:ssa määriteltyyn ORCHID-tunnisteluokkaan. ORCHID-tunnisteet (overlay routable cryptographic hash identifier) on tarkoitettu päätepisteiden tunnisteiksi sovelluksissa ja ohjelmointirajapinnoissa. Ne ovat rakenteeltaan kuin IPv6-osoitteita, mutta ne eivät ole reitityskelpoisia verkkokerroksella. ORCHID-tunnisteiden odotetaan olevan reitityskelpoisia jollain verkkokerroksen yläpuolisella tasolla. ORCHID-tunnistetta voi käyttää IPv6-ohjelmointirajapinnoissa ja -sovelluksissa. (Laganier & Dupont 2015)

LSI (local scope identifier) on paikallinen esitystapa identiteettitunnistetagille, jota voidaan käyttää 128-bittisen HIT:n tilalla. LSI:n 32-bittisyys mahdollistaa sen käytön IPv4-protokollassa ja -ohjelmointirajapinnassa. (Moskowitz & Nikander 2006) LSI:t eivät kuitenkaan ole globaalisti uniikkeja, joten niitä voidaan käyttää ainoastaan paikallisesti (Nikander et al. 2010).

2.2 Pakettirakenne

HIP koostuu kontrolliprotokollasta, sen laajennuksista ja dataprotokollista. Kontrolliprotokollan toimintaan kuuluvat muun muassa nelivaiheinen yhteydenmuodostus ja kolmivaiheinen yhteyden katkaisu. Toistaiseksi ainoa HIP:n tukema dataprotokolla on IPsec ESP (Internet Protocol security architecture, encapsulated security payload). HIP:n kontrolliprotollan paketit kulkevat IPv4- ja IPv6-paketeissa. (Nikander et al. 2010)

HIP:n kontrolliprotokollan paketit lähetetään tavallisessa IP-paketissa IP:n käytetyn version mukaisen otsikon (header) jälkeen (Gurtov 2008, s. 52). IPv4- ja IPv6-verkon paketit koostuvat otsikosta ja datasegmentistä. Molempien IP:n versioiden otsikot ovat vakiopituisia. IPv4:n otsikon pituus on 20 tavua ilman mahdollisia optioita, ja se sisältää seuraavat tiedot:

- IP-protokollan versionumero
- otsikon pituus
- palvelutyyppi
- paketin kokonaispituus
- fragmentointitunnus, liput (flags) ja fragmentoinnin paikka
- elinaika
- ylemmän tason protokollan numero
- otsikon tarkistussumma
- 32-bittinen lähde- ja kohdeosoite (Kurose & Ross 2010, s. 368–370).

IPv6-paketin otsikon pituus on 40 tavua, ja se sisältää seuraavat kentät:

- IP-protokollan versionumero

- luokka
- vuon tunnus (flow label)
- datasegmentin pituus
- seuraavan otsikon tyyppi
- elinikä
- 128-bittinen lähde- ja kohdeosoite (Deering & Hinden 2017).

IP-protokollan otsikon jälkeen paketissa lähetetään kuljetuskerroksen protokollan otsikkotiedot ja hyötydata. HIP:n implementointi ei vaikuta kuljetuskerroksen otsikon tietoihin (Nikander et al. 2010).

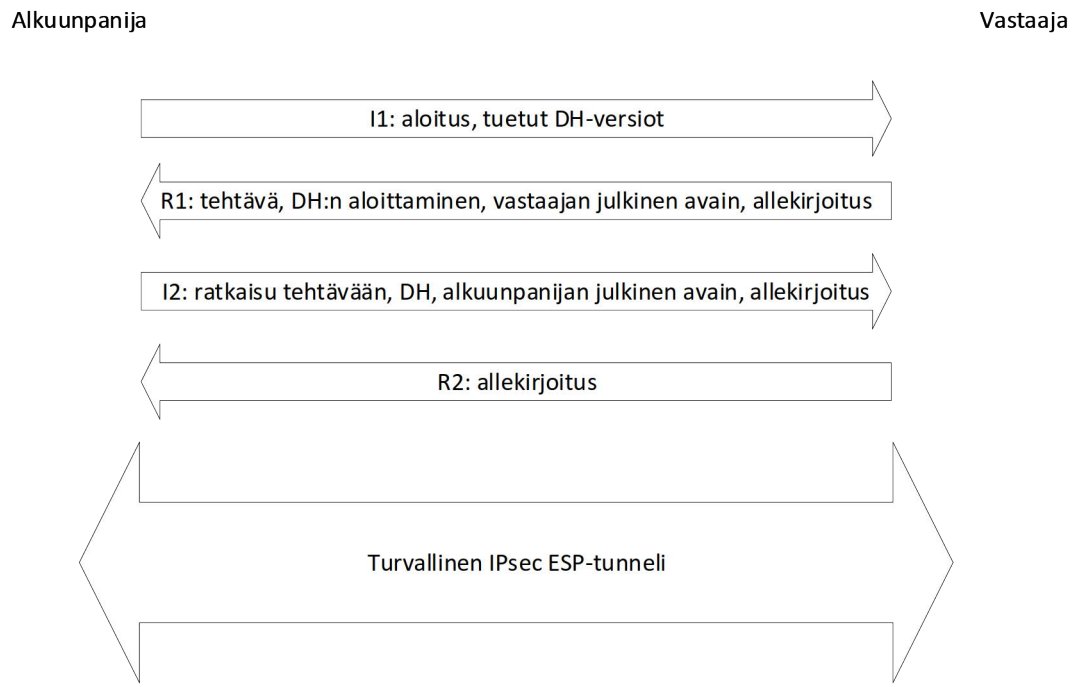
HIP:n kontrollipaketeissa ei voi lähettää käyttäjädataa, sillä HIP:n otsikko on määritelty paketin viimeiseksi otsikoksi (Moskowitz et al. 2015). HIP:n otsikko on muodoltaan kuin IPv6:n laajennusotsikko (Gurtov 2008, s. 52). HIP-paketin vakiopituinen otsikko sisältää seuraavat tiedot:

- seuraavan otsikon tyyppi
- otsikon pituus
- pakettityyppi
- HIP:n versio
- tarkistussumma
- kontrollitiedot
- lähettäjän ja vastaanottajan identiteettitunnistetagit (Moskowitz et al. 2015).

Otsikon jälkeen lähetetään HIP-parametrit, jotka sisältävät yhteyden muodostukseen ja ylläpitoon vaadittavat tiedot. Lähetettävät parametrit riippuvat siitä, käytetäänkö pakettia luomaan uutta yhteyttä vai ylläpitämään olemassa olevaa.

2.3 Nelivaiheinen yhteydenmuodostus

HIP-yhteyden (HIP association) luominen alkaa nelivaiheisella yhteydenmuodostuksella (base exchange). Yhteys luodaan alkuunpanijan (initiator) ja vastaajan (responder) välillä. Nämä roolit ovat käytössä vain yhteyden luontivaiheessa, jonka jälkeen alkuunpanijan ja vastaajan roolit unohdetaan. (Moskowitz et al. 2015) Yhteydenmuodostusprosessi on esitetty kuvassa 2.2.



Kuva 2.2 *HIP:n nelivaiheinen yhteydenmuodostus*

Merkitään kuvan 2.2 mukaisesti alkuunpanijan lähettämiä paketteja tunnuksella I ja vastaajan paketteja tunnuksella R. Nelivaiheisessa avaintenvaihdossa lähetettävät paketit ovat järjestyksessä I1, R1, I2 ja R2. Paketit R1, I2 ja R2 sisältävät lähettäjän allekirjoituksen. Paketti I1 käynnistää yhteyden luontiprosessin, ja loput kolme pakettia suorittavat autentikoidun Diffie–Hellman-avaintenvaihdon (DH).

Paketti I1:n tehtävänä on aloittaa DH-avaimenvaihdosta sopiminen kertomalla vastaajalle, mitä DH-versioryhmiä se tukee. I1 lähetetään vastaajan IP-osoitteeseen. Paketti I1 sisältää alkuunpanijan identiteettitunnistetagin sekä vastaajan identiteettitunnistetagin, jos se on tunnettu. Vaikka vastaajan HIT ei ole tiedossa, avaintenvaihto voidaan suorittaa opportunistisessa tilassa. HIP:n opportunistinen tila on

altis mies välissä -hyökkäyksille, joten sitä voidaan käyttää vain turvallisessa ympäristössä. (Gurtov 2008, s. 51–52)

I1-paketin vastaanottanut osapuoli ei säilytä paketin sisältämää informaatiota. Paketti R1 sisältää molempien osapuolten identiteettitunnistetagit sekä ongelman (puzzle), joka alkuunpanijan pitkää ratkaista, jotta yhteydenmuodostus voi jatkua. Ongelma on kryptografinen tehtävä, jossa alkuunpanijan pitää muodostaa arvo, jonka SHA-1-tiiviste sisältää tietyn lukumäärän pelkkiä nollia. Ongelman vaikeutta voidaan muuttaa yhdistävän osapuolen luotettavuudesta riippuen vaihtamalla vaadittavien nollien määrää. (Gurtov 2008, s. 54) Kryptografisen tehtävän ratkaisua käytetään palvelunestohyökkäysten torjunnassa, jota käsitellään tarkemmin aliluvussa 4.2. R1 aloittaa autentikoidun DH-avaintenvaihdon. R1 on allekirjoitettu käyttäen vastaajan julkista avainta.

Alkuunpanija varmentaa vastaajan R1-paketin allekirjoituksesta, jonka jälkeen se ratkaisee vastaanotetun ongelman. I2 sisältää ratkaisun ongelmaan, yhdistävän osapuolen julkisen Diffie-Hellman–avaimen ja identiteettitunnisteen julkisen avaimen ja varmennuksen, joka näyttää että I2 on alkuperäisen alkuunpanijan muodostama. Identiteettitunnisteen julkinen avain voi olla salattu DH-avaimella. Vastaaja jatkaa DH–avaimenvaihtoa, kun se on varmentanut vastaanotetun ratkaisun ongelmaan. Kun ratkaisu on varmennettu, molemmat osapuolet tietävät saman DH-avaimen. Lisäksi vastaaja tietää, että yhdistävällä osapuolella on identiteettitunnisteen julkista avainta vastaava yksityinen avain. (Nikander et al. 2010)

Onnistunut yhteydenmuodostus päättyy R2-pakettiin, joka ilmoittaa I2-paketin yhdistävälle osapuolelle. Vastaaja lähettää R2-paketin vasta, kun se on vastaanottanut ja varmistanut oikean ratkaisun lähetettyyn ongelmaan. Vastaaja hylkää väärän vastauksen sisältävät paketit. (Moskowitz et al. 2015)

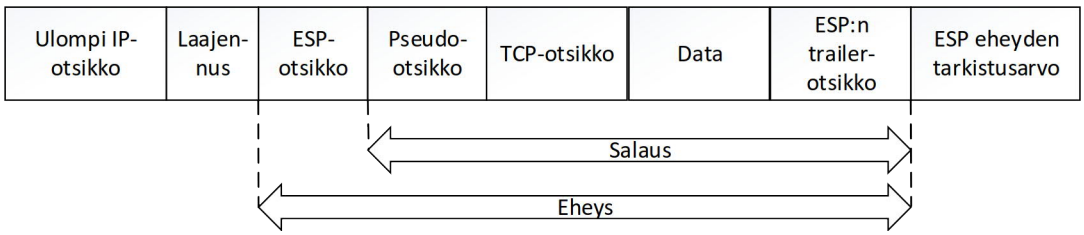
2.4 Turvallinen IPsec ESP –tiedonsiirtoyhteys

HIP-yhteydenmuodostuksen jälkeen osapuolten välille luodaan oletusarvoisesti IPsec ESP –turvayhteys (secure association), jossa liikenteen salaamiseen käytetään yhteydenmuodostuksessa luotua Diffie-Hellman-avainta. HIP-protokolla ei kuitenkaan rajoitu vain IPsec:n käyttöön, vaan se voidaan laajentaa tukemaan myös muita suojausprotokollia, kuten SRTP:tä (secure real-time transport protocol). (Nikander et al. 2010)

IPsec ESP on verkkokerroksen turvallisuusprotokolla, jossa ESP (encapsulated security payload) on käytetty salausmuoto. IPsecin kuljetustilaa (transport mode) käy-

tetään turvaamaan kahden päätelaitteen välistä liikennettä, jolloin paketista salataan vain hyötydata. Verkkojen välillä käytetään yleensä ESP:n tunnelitilaa (tunnel mode), jossa koko paketin sisältö salataan. (Gurtov 2008, s. 31) Paketin rakenne tunnelitilassa eroaa kuljetustilasta siten, että IPsec luo paketille uuden IP-otsikon ja alkuperäinen IP-otsikko lähetetään salattuna IPsec-otsikon jälkeen (Wu & Irwin 2013, s. 1055).

Suositeltu ESP-formaatti HIP:lle on BEET-tila (bound end-to-end tunnel), joka on edellä mainittujen kuljetus- ja tunnelitilojen yhdistelmä. BEET-tilassa IPsec ESP-paketin ensimmäisessä lähetettävässä otsikossa käytetään IP-osoitteita ja ESP-otsikossa identiteettitunnistetageja. Turvayhteyden elinajan vakiona pysyvät identiteettitunnistetagit voidaan jättää paketin otsikoista kokonaan pois. (Gurtov 2008, s. 64-65) Kuvassa 2.3 on esitetty BEET-tilan IPsec ESP-paketin rakenne.



Kuva 2.3 *ESP-pakettirakenne IPv6-paketille BEET-tilassa. Käännetty lähteestä (Jokela et al. 2015)*

.

ESP ympäröi alkuperäisen TCP/IP-paketin uusilla otsikko- ja autentikointikentillä, ja alkuperäinen paketti lähetetään salattuna uuden ESP-paketin sisällä (Wu & Irwin 2013, s. 1056). ESP:llä voidaan varmentaa paketin lähettäjä sekä datan eheys (integrity) ja luottamuksellisuus (confidentiality). Eheyden tarkistamiseen käytetään paketissa viimeisenä lähetettävää autentikointikenttää, joka sisältää tarkistusarvon. (Kurose & Ross 2010, s. 762)

ESP-otsikko sisältää SPI:n (security parameter index), jolla paketin vastaanottaja identifioi ja varmentaa turvayhteyden. SPI-arvo on 32-bittinen numero, ja sen käyttö on määritelty pakolliseksi. (Kent 2005)

2.5 Mobiliteetin hallinta

Päätelaitteen mobiliteetilla tarkoitetaan sitä, että se voi liikkua verkosta toiseen ilman yhteyden katkeamista. IoT-verkossa tämä voi tarkoittaa tilannetta, jossa liikkuvassa kohteessa oleva langaton anturi siirtyy WLAN-tukiasemasta toiseen. Laitteen

siirtyessä verkosta toiseen pakettien pakettien kulkureitti muuttuu.

HIP-protokollan tapa parantaa mobiliteettia perustuu IP-osoitteen kaksoisroolin eriyttämiseen. HIP-protokollan arkkitehtuurissa kuljetuskerroksen rajapinnat (socket) ja ESP-turvayhteydet ovat sidottuja IP-osoitteiden sijasta päätelaitteen identiteetteihin. IP-osoitteen vaihtuminen ei siten aiheuta yhteyden katkeamista. IP-osoitteita käytetään kuitenkin pakettien reititykseen, mutta ne ovat epäolennaisia paketin saavuttua vastaanottavaan verkkorajapintaan (Nikander et al. 2010).

HIP-turvayhteyksien on oltava muodostettuina laitteiden välille ennen IP-osoitteen vaihtumista, jolloin molemmilla laitteilla on yksi turvayhteys sisään ja ulos. HIP-arkkitehtuurissa IP-osoitetta vaihtava laite ilmoittaa uusista osoitteistaan UPDATE-paketilla, jonka LOCATOR_SET-parametri sisältää laitteen sijainnit, joista sen voi tavoittaa. Sijainneistaan ilmoittava laite voi asettaa jotkut sijaintinsa ensisijaiseksi. Oletusarvoisesti ensisijainen IP-osoite on sama kuin HIP-yhteydenmuodostuksessa. (Henderson et al. 2017a)

Toisen laitteen vaihtaessa IP-osoitetta molempien osapuolten täytyy päivittää paikalliset identiteettitunnistetagien ja IP-osoitteiden yhteydet HIP-alikerroksella. Yksi HIP:n UPDATE-paketti voi muuttaa laitteen IP-osoitteen sekä IPsec-yhteyden identifioivan SPI:n. IP-osoitteen vaihtuessa voidaan suorittaa uusi Diffie-Hellman-avaintenvaihto, jolloin uusi IP-osoite otetaan käyttöön vasta uuden salausavaimen luomisen jälkeen. (Henderson et al. 2017a)

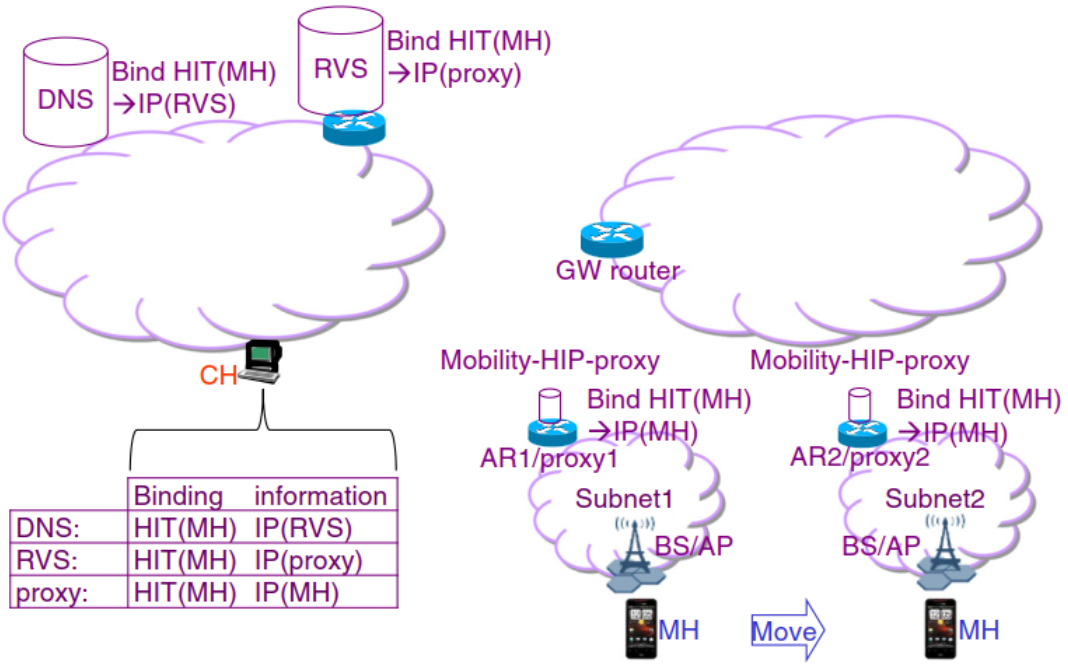
HIP:ta käyttävien verkkolaitteiden identiteettitunnistetageja ja IP-osoitteita voidaan säilyttää myös rendezvous-servereillä (RVS). RFC8004:ssä (Laganier & Eggert 2016) määritellyn rendezvous-laajennuksen tarkoituksena on parantaa mobiilien ja moniverkotettujen laitteiden yhdistettävyyttä ja toimintaa. Rendezvous-serveriä käytettäessä HIP-yhteydenmuodostuksen aloittava laite lähettää ensimmäisen paketin RVS:lle, joka välittää sen HIT:n perusteella oikealle vastaanottajalle. IP-osoitteiden vaihtuessa laitteet ilmoittavat uusista sijanneistaan RVS:lle. IP-osoite- ja HIT-tietokannan säilytys erillisellä serverillä on hyödyllinen tilanteissa, joissa laitteiden IP-osoitteet vaihtuvat, kun turvayhteyksiä ei ole muodostettu.

HIP:n DNS-laajennuksessa (Laganier 2016) määritellän tapa, jolla HIP-laite voi säilyttää identiteettiään ja sitä vastaavan rendezvous-palvelimen verkkotunnusta DNS-palvelimella. Näin saavutetaan hyvä yhdistettävyys, sillä yhteydenmuodostuksen voi aloittaa tekemällä normaalin DNS-kyselyn identiteettitunnistetagilla, jolloin DNS vastaa sen RVS:n IP-osoitteella, johon vastaava HIT on rekisteröity.

Muslamin et al. (2012) ehdotuksessa mobiliteetin hallinta voidaan toteuttaa kahdel-

la eri tason rendezvous-serverillä ja HIP-välityspalvelimella, jolloin saavutetaan vähäisempi signalointipakettien määrä ja pienemmät viiveet mobiililaitteen vaihtaessa tukiasemaa. Ehdotuksen mukaan mobiililaitteisiin ei tarvitse tehdä muutoksia, sillä verkon pääsyreititin (access router) toimii HIP-välityspalvelimena, joka hoitaa HIP-signalointiliikenteen mobiililaitteiden puolesta. Mallissa käytetään myös DNS-palvelinta helpottamaan laitteiden tavoitettavuutta. Mobiililaitteen suorittaessa verkkoon rekisteröitymisen HIP-välityspalvelimelle tallentuu tieto mobiililaitteen IP-osoitteesta ja identiteettitunnistetagista. Tietoa siirretään välityspalvelimesta ylemmille tasoille järjestyksessä LRVS-RVS-DNS, jolloin DNS-palvelimella on tiedot mobiililaitteiden identiteettitunnistetageista ja niitä vastaavien RVS-servereiden IP-osoitteista. HIP-yhteydenmuodostuksen alussa RVS välittää paketin identiteettitunnistetagin mukaan oikealle LRVS:lle, joka ohjaa sen oikealle mobiililaitteelle.

Muslamin et al. (2017) uudemmassa ehdotuksessa mobiliteetin hallinta toteutetaan hajautetusti, mikä on erityisen tarpeellista epähierarkkisissa verkoissa (flat network). Ehdotetun mallin arkkitehtuuri on esitetty kuvassa 2.4, jossa mobiililaite (MH) siirtyy aliverkosta toiseen säilyttäen saumattoman yhteyden verkkoon. Mobiililaite säilyttää IP-osoitteensa siirtyessään verkkojen välillä riippumatta siitä, onko laitteessa HIP-tuki asennettuna. Hajautetussa mallissa ei käytetä erillistä paikallista rendezvous-palvelinta, vaan RVS sisältää laitteiden identiteettitunnistetageja vastaavat HIP-välityspalvelinten IP-osoitteet.



Kuva 2.4 *Hajautetun verkkopohjaisen mobiliteetin hallinnan malli HIP-välityspalvelinta käyttäen (Muslam et al. 2017)*

Paikallisesta palvelimista luopuminen parantaa mobiliteetin hallinnan virheensietokykyä ja järjestelmän skaalautuvuutta. Vaadittavan signaloinnin määrä on pienempi kuin paikallisia palvelimia käyttävässä mallissa, ja suorituskykytestit osoittavat hajautetun mallin toimivan aikaisempaa keskitettyä välityspalvelinmallia tehokkaammin epähierarkkisissa verkoissa. Hajautettu HIP-välityspalvelinta käyttävä mobiliteetin hallintamalli soveltuu suurille laitemäärille ja pienitehoisille laitteille vähäisen signaloinnin takia. Välityspalvelimen käyttö poistaa tarpeen asentaa HIP-ohjelmistoja päätelaitteisiin.

2.6 Yhteensopivuus palomuurien ja osoitteenmuunnoksen kanssa

HIP:n yhteensopivuus osoitteenmuunnoksen ja palomuurien kanssa on oleellista etenkin laitteiden liikkuessa verkosta toiseen, sillä silloin yhteyden päätepisteiden välillä oleva polku muuttuu. Päätepisteiden välillä on joukko verkkolaitteita, jotka voivat muuntaa, tutkia tai suodattaa liikennettä. Päätepisteiden väliset verkkolaitteet (middlebox) voivat sisältää esimerkiksi palomuureja tai osoitteenmuuntimia (NAT). Pakettien reitityksen kannalta HIP:n käyttöönotolla ei ole merkitystä, sillä verkkokerroksen reitittimet ohjaavat paketteja normaalien IP-osoitteisiin perustuvien reitityskäytäntöjen mukaisesti.

Kurosen ja Rossin (2010, s. 773) mukaan ”palomuuri on yhdistelmä laitteita ja ohjelmistoja, jotka eristävät organisaation sisäverkon internetistä päästäen osan paketeista läpi ja torjumalla muut.” Palomuuri sijaitsee ylläpidetyn yksityisen verkon ja julkisen internetin välissä, ja sen läpi päästetään vain verkon ylläpitäjän sallituksi määrittelemä liikenne. Palomuurilla voidaan tarkoittaa myös verkkolaitteeseen asennettua ohjelmistoa, joka valvoo ja kontrolloi laitteen pakettiliikennettä.

NAT (network address translation) on osoitteenmuunnostekniikka, jota käytetään IPv4-osoitteiden vähyyden takia. Kaikilla verkon laitteilla on oltava jokin julkinen IP-osoite, mutta IPv4-osoiteavaruuden pienen koon takia niitä ei riitä kaikille laitteille. Osoiteavaruuden riittämättömyyden takia on käytettävä osoitteenmuunnosta, jonka seurauksena samaa julkista IP-osoitetta voi käyttää useampi sisäverkon laite. Perustason NAT muuttaa vain IP-osoitteen, mutta NAPT (network address and port translation) asettaa yksittäisille yhteyksille uuden ulkoverkon portin, joten yhtä julkista IP-osoitetta voidaan käyttää monessa yhteydessä samanaikaisesti.

Verkon välilaitteet voivat vaikeuttaa HIP:n kontrollipakettiliikennettä tai IPsec ESP-dataliikennettä. Kontrollipakettiliikenteellä tarkoitetaan muun muassa yhteydenmuodostukseen ja uuden IP-osoitteen ilmoittamiseen käytettyjä paketteja.

Yhteydenmuodostuksen paketit eivät ole salattuja, joten IPv4-protokollaa käytettäessä kontrollipaketit kulkevat hyvin vain IP-otsikon IP-osoitteita muuttavan NAT:n läpi. Pelkkää IP-osoitteen muunnosta huomattavasti yleisemmässä NAPT:ssä IPv4-kontrollipakettien kuljetuskerroksen porttien muuntaminen ei onnistu, sillä yhteydenmuodostukseen käytetty IP-hyötykuorma ei sisällä porttinumeroita. NAT:n takana oleva HIP-päätelaite ei ole tavoitettavissa ilman, että se itse luo yhteyden ulospäin. IPv6-protokollalla toteutetussa yhteydenmuodostuksessa HIP:n kontrollidata lähetetään HIP-laajennusotsikossa eikä IP-hyötykuormana, mikä voi aiheuttaa ongelmia IP-osoitteita muuntavissa välilaitteissa. IPv6-verkoissa NAT:t ovat harvinaisia, mutta niiden toimintatapa on sama kuin IPv4-verkoissa. (Stiemerling et al. 2008)

HIP:n oletusarvoisesti muodostama IPsec ESP –tiedonsiirtoyhteys voi aiheuttaa ongelmia palomuureissa, sillä IP-osoitteen alkuperäiset lähde- ja kohdeosoitteet lähetetään salattuna. Tunnelin päätepisteisiin sidotussa IPsec-pakettiliikenteessä IP-otsikon lähde- ja kohdeosoitteena ovat tunnelin päätepisteiden reitittimien rajapintojen IP-osoitteet (Kurose & Ross 2010, s. 764). HIP:n IPsec-dataliikenteellä on samat vaikeudet osoitteenmuuntimien kanssa kuin millä tahansa muullakin IPsec-liikenteellä. NAT ei näe ylempien protokollakerrosten otsikoita, sillä ne lähetetään salattuna. NAT:n tekemä muutos IP-otsikon osoitekentässä voi aiheuttaa sen, että ylemmän tasojen otsikoiden tarkistussummat muuttuvat virheellisiksi, ja paketti todetaan virheelliseksi. HIP-protokollassa tarkistussummien laskeminen voidaan korjata oikeaksi korvaamalla IP-osoitteet laskentavaiheessa identiteettitunnistetageilla. (Stiemerling et al. 2008)

Tschofenigin et al. (2005) julkaisussa ehdotetaan toimintoja, joita HIP:tä tukevan palomuurin tai osoitteenmuuntimen tulisi sisältää. HIP-liikennettä oikein käsittelevien palomuurien ja osoitteenmuuntimien pitää pystyä autentikoimaan HIP-päätelaitteet ennen osoitteenmuunnosta tai uuden palomuurisäännön luomista.

RFC-dokumentissa 5770 (Komu et al. 2006) esitetään NAT:n läpäisemisen parantamiseksi UDP-kapselointia ja valinnaista rendezvous-palvelin laajennusta. IP-osoitteiden ja identiteettitunnistetagien yhdistelmiä hallinnoima rendezvous-palvelin voidaan laajentaa välittämään kaikki yhteydenmuodostuksen paketit, jolloin saavutetaan parempi tavoitettavuus myös NAT:n taakse.

3. HIP:N MAHDOLLISUUDET IOT-VERKOISSA

HIP-protokollan käytöllä voidaan ratkaista IoT-sovellusten kannalta tärkeitä verkon toimintaan liittyviä ongelmia ja puutteita. Gladischin et al. (2014) mukaan mobiliteetti ja moniverkotus (multihoming) ovat nykyisen ja tulevaisuuden internetin suurimpia haasteita, jotka johtuvat päätelaitteiden liikkuvuuden ja verkkoympäristöjen epäyhtenäisyyden nopeasta kasvusta. Mobiliteetin ja moniverkotuksen toteuttamisen vaikeudet liittyvät kiinteästi toisiinsa. Yhden laitteen useammalla samanaikaisella verkkoyhteydellä voidaan helpottaa yhteyden säilyttämistä laitteen liikkuessa verkkojen välillä.

Merkittävä osa IoT-laitteista on mobiililaitteita, ja ne vaativat verkolta mobiliteetinhallintaa keskeytyksettömän toiminnan takaamiseksi (Yassein et al. 2017). Mobiliteetin hallinta voidaan toteuttaa erilaisilla protokollajoukoilla. Mobile IP on yleisesti käytössä oleva standardiprotokolla, jonka tavoitteena on ylläpitää yhdistettävyys (connectivity) verkkojen välillä. Mobile IP pyrkii pitämään mobiililaitteiden IP-osoitteet pysyvinä laitteen siirtyessä verkosta toiseen. Mobile IP-teknologia on kehitetty IPv4- ja IPv6-verkoille.

6LoWPAN (IPv6 over Low-Power Wireless Personal Area Networks) on verkkotekniikka, jota voidaan käyttää IEEE 802.15.4-standardin laitteilla. 6LoWPAN on käyttökelpoinen IoT-sovelluksissa ja langattomissa anturiverkoissa, joiden kantama on lyhyt, datanopeudet pieniä ja laitteet matalatehoisia. HIP soveltuu käytettäväksi 6LoWPAN-verkoissa, mutta laitteiden vähäinen laskentakapasiteetti vaatii joko hajautetun laskennan tai HIP:n kevyemmän yhteydenmuodostuksen käyttöä.

LISP (locator/identifier separation protocol) on protokolla, joka HIP:n tavoin erottaa IP-osoitteen kaksoisroolin paikantimena ja tunnisteena. Protokolla on helppo ottaa käyttöön, ja sillä voidaan toteuttaa mobiliteetin hallintaa ja moniverkotusta (Balan et al. 2017). HIP on suosituin IP-osoitteen kaksoisroolin erottamiseen käytetty protokolla (You & Jung 2012). LISP keskitty reititysjärjestelmän skaalattavuuuteen ja HIP turvalliseen päästä päähän mobiliteettiin ja moniverkotukseen. HIP ja LISP nähdään enemmän toisiaan täydentävinä kuin keskenään kilpailevina protokollina (Gurtov et al. 2009).

3.1 Moniverkotus

Moniverkotuksella tarkoitetaan tilannetta, jossa verkkolaitteella on useita yhden hypyn (hop) yhteyksiä verkkoon (Sousa et al. 2011). Hypyllä tarkoitetaan yhteyden lähteen ja kohteen välisen polun yhtä osuutta. Jos yhteyden lähde on reitittimeen yhdistetty tietokone, lähetetyn paketin ensimmäinen hyppy tapahtuu paketin kulkiessa reitittimen läpi.

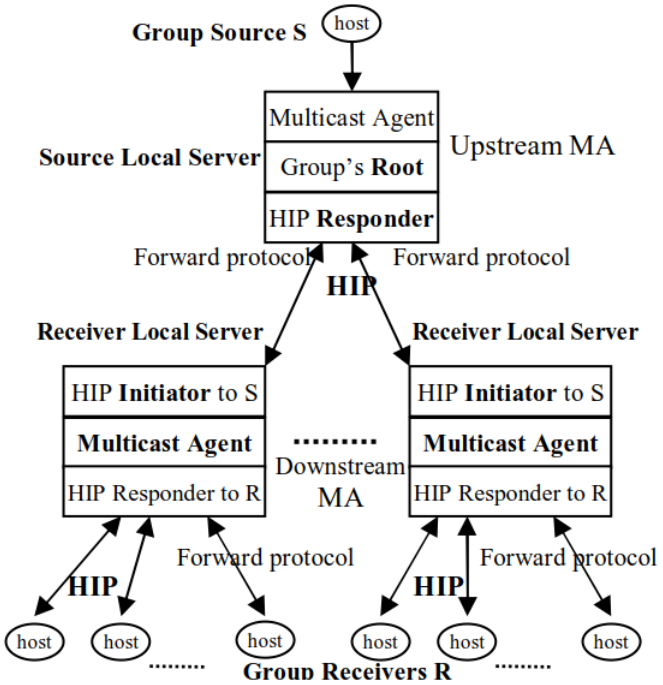
Yksi päätelaite voi sisältää useita verkkorajapintoja, jolloin laite voi olla yhteydessä useaan verkkoon samanaikaisesti. Usean yhteyden käytöllä voidaan saavuttaa parempi toimintavarmuus sekä suurempi kaistanleveys. Monen verkkorajapinnan käyttö on hyödyllistä myös mobiliteetin kannalta, sillä laitteen liikkuessa verkkojen välillä yhtä rajapintaa voidaan käyttää verkonvaihtoon liittyvän neuvotteluprosessin suorittamiseen. (Gladisch et al. 2014)

HIP tukee skenaarioita, joissa yhden laitteen IP-osoitteet voivat olla jaettuna yhdelle tai useammalle verkkorajapinnalle. HIP:n moniverkotuslaajennus tukee sekä IPv4- että IPv6-osoitteita. HIP:n moniverkotusarkkitehtuurissa käytetään samaa LOCATOR_SET-parametria kuin mobiliteetin toteutuksessa. Laite, jolla on useita IP-osoitteita voi ilmoittaa ne jo HIP-yhteydenmuodostuksen aikana R1-, I2- tai R2-paketeissa, jolloin saavutetaan parempi virheensietokyky, sillä turvayhteyden muodostamiseen voidaan käyttää virhetilanteessa jotain muuta IP-osoitetta. Myös moniverkottavan laitteen kuormituksen tasaus (load balancing) voidaan toteuttaa lähettämällä LOCATOR_SET-parametri jo yhteydenmuodostusvaiheessa (Henderson et al. 2017b).

3.2 Monilähetys

Tehokkaan monilähetystekniikan kehittäminen on välttämätöntä suuren mittakaavan ryhmäkommunikaatiosovelluksissa (Zhang et al. 2006). Monilähetysteknologia on hyödyllinen esineiden internetin sovelluksissa sekä langattomissa anturiverkoissa, sillä se mahdollistaa tehokkaamman tavan siirtää dataa yhdestä tai useammasta laitteesta moneen laitteeseen. IP-monilähetyksellä (IP multicast) tarkoitetaan tietoliikennepakettien lähettämistä yhdestä tai useammasta lähteestä usealle eri vastaanottajalle yhdellä lähetyksellä. Monilähetys voidaan toteuttaa sovellus- tai verkkotasolla. Sovellustason monilähetys on helpompi toteuttaa kuin verkkotason IP-monilähetys, mutta se ei ole yhtä tehokas ja luotettava tapa (Garyfalos et al. 2003). Monilähetys on yleisesti käytössä automaatiossa, ja sitä käytetään esimerkiksi Valmet DNA -automaatiojärjestelmässä.

Zhun & Atwoodin (2007) ehdottamassa HIP-protokollaa hyödyntävässä monilähetystekniikassa voidaan sekä todentaa ja valtuuttaa päätelaitteet että varmistaa datavirran luottamuksellisuus, mikä ei ole mahdollista normaalissa IP-monilähetyksessä. Vaikka HIP on tarkoitettu yhdeltä-yhdelle-lähetykseen, laajennusehdotuksen mukaan HIP:tä voidaan käyttää monilähetysryhmän laitteiden hallintaan. Päätelaitteiden identiteetin lisäksi monilähetysryhmällä on oltava identiteetin tunnus, joka on ryhmän julkinen avain. Ryhmän identiteettitunnisteesta (GI) muodostetaan 128-bittinen identiteettitunnistetagi (GIT) yksisuuntaisella tiivistefunktiolla. Uusi HIP:ta tukeva laite voi liittyä monilähetysryhmään aloittamalla HIP-yhteydenmuodostuksen ryhmän jäsenistöä ylläpitävän monilähetysagentin (multicast agent) kanssa. Monilähetysmallin rakenne on esitetty kuvassa 3.1, joka sisältää kaksi vastaanottajaryhmää.



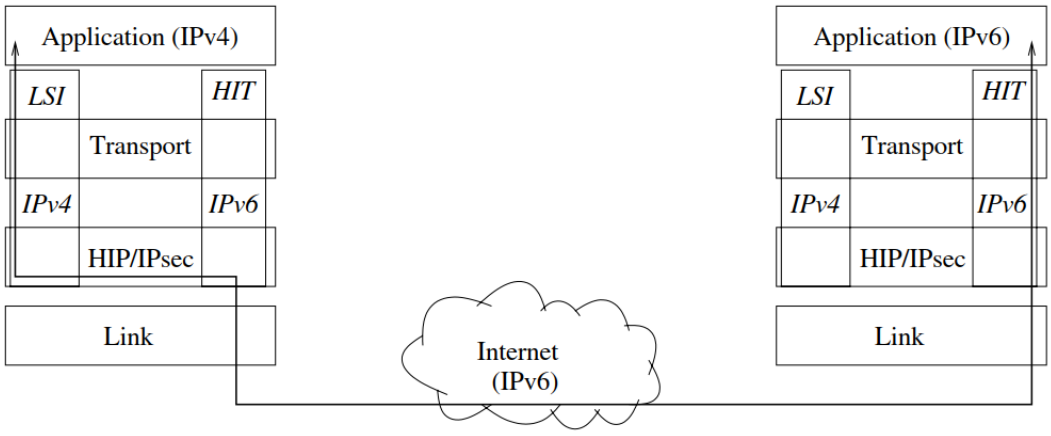
Kuva 3.1 *HIP-monilähetysmallin rakenne (Zhu & Atwood 2007)*

Zhun et al. uudemassa julkaisussa esitetään HIP-monilähetysarkkitehtuurille soveltuva reititysalgoritmi (2011). IP-monilähetysmalliin perustuvat reititysalgoritmit eivät sovellu HIP-monilähetykseen, sillä ne eivät erottele laitteita identiteetin perusteella. Uusi reititysmenetelmä perustuu Minimun Path Cost –algoritmiin ja monilähetyspuiden jakamiseksi pienempiin osiin.

3.3 IPv4- ja IPv6-verkkojen yhdistäminen

Kahden eri IP-version samanaikainen käyttäminen internetissä aiheuttaa ongelmia, sillä IPv6 ei ole taaksepäin yhteensopiva IPv4:n kanssa. Syy IPv6:een siirtymiseen on IPv4-osoiteavaruuden riittämättömyys nykyiselle verkkolaitteiden lukumäärälle, joka tulee kasvamaan entisestään esineiden internetin yleistymisen myötä. IPv6 on ehdottoman tärkeä esineiden internetille laitteiden yhdistettävyyden kannalta.

Kuva 3.2 esittää IPv4- ja IPv6-sovellusten internetin yli tapahtuvan keskinäisen kommunikoinnin HIP-protokollan välityksellä. IPv6-sovellus käyttää kommunikointiin IPv6-osoitteen muotoista ja pituista identiteettitunnistetagia, joka täytyy IPv4-sovelluksen rajapinnassa muuttaa 32-bittiseksi LSI:ksi.



Kuva 3.2 *IPv4- ja IPv6-sovellusten kommunikointi internetin yli (Gurtov 2008, s.231)*

HIP:n moniverkotuslaajennuksella (Henderson et al. 2017b) HIP-päätelaitteella voi käyttää IPv4- ja IPv6-osoitteita samanaikaisesti yhdellä tai useammalla verkkorajapinnalla. HIP:n mobiliteettilaajennus (Henderson et al. 2017a) mahdollistaa mobiililaitteiden liikkumisen IPv4- ja IPv6-verkkojen välillä yhteyden katkeamatta. (Gurtov 2008, s. 231).

3.4 Suorituskyky IoT-laitteissa

HIP ei ole ainoa mahdollinen tapa toteuttaa moniverkotusta ja helpottaa laitteiden mobiliteettia. Esineiden internetissä protokollien käyttöä rajoittaa laitteiden alhaiset laskentakapasiteetit, sillä raskaiden kryptografisten algoritmien suorittaminen voi olla liian kuormittavaa. HIP Diet EXhange (DEX) on tavallista HIP BEX

–yhteydenmuodostusta kevyempi tapa luoda turvallinen yhteys kahden laitteen välille. Kevennetyssä versiossa luovutaan julkisen avaimen allekirjoituksista ja tiivistefunktioista (Moskowitz & Hummen 2017).

Mecan et al. (2013) esittämässä mallissa HIP:n ja Multimedia Internet KEYingin (MIKEY) yhdistelmällä saavutetaan turvallinen päästä-päähän yhteys ja suoja palvelunestohyökkäyksiä vastaan, joka voidaan saavuttaa kevyempää laskentaa käyttäen. Ehdotuksen sovellusmallina käytetään älykaupunkia, jossa kaupungin infrastruktuuri, autot ja ihmiset vaihtavat informaatiota mahdollistaen uusia palveluita. Ehdotuksen mukainen kevyempi yhteydenmuodostustapa on HIP DEX. Turvalliseen kommunikointiin tarvittava avainmateriaali luodaan MIKEYn päälle rakennetulla Adapted Multimedia KEYingillä (AMIKEY). Prototyypissä mallia testattiin Contiki OS –käyttöjärjestelmällä Redbee Econotag –laitteistolla. Ehdotetun HIP-arkkitehtuurin ROM-muistin kulutus testilaitteistossa oli vain 5,6 %.

Garcia-Morchon et al. (2013) vertaavat HIP:tä ja DTLS:ää (Datagram Transport Layer Security) IP-pohjaisen IoT:n suojauksessa. Molemmissa malleissa käytettiin ennalta jaettuja avaimia (pre-shared key, PSK). HIP:n PSK-kättely perustuu HIP DEX –menetelmään, mutta staattisen Diffie-Hellman-avaimen sijasta istuntoavain määritetään CMAC:llä (Cipher-based Message Authentication Code). Redbee Econotag –laitteistolla ja Contiki OS –käyttöjärjestelmällä tehdyt testit osoittivat, että HIP-pohjainen ratkaisu käytti vähemmän ROM- ja RAM-muistia, ja oli DTLS-pohjaista ratkaisua suorituskykyisempi.

Porambage et al. (2017) ehdottavat vähäresurssisille IoT-laitteille sopivaksi yhteistyöhön perustuvaa Collaborative HIP –ratkaisua (CHIP), jossa vähätehoiset IoT-laitteet voivat delegoida laskennan välityspalvelimelle. Vastaajan roolissa oleva vähäresurssinen laite siirtää suurimman osan kryptografisista toimenpiteistä välityspalvelimen (proxy) hoidettavaksi. Yhteydenmuodostuksen aloittava laite voi siirtää turvallisuussyistä vain osan laskennastaan välityspalvelimelle. CHIP:n testaus osoitti, että sen yhteydenmuodostustapa käyttää vähemmän energiaa kuin HIP BEX tai HIP DEX.

Sahraoui & Bilami (2015) esittävät CD-HIP-mallin (compressed and distributed HIP), jossa yhdistetään HIP-otsikkojen 6LoWPAN-pakkaus ja laskentakuormituksen hajauttaminen. Pakkaamalla HIP-otsikot lähetettävien pakettien koot ovat pienempiä, mikä säästää energia, lyhentää viiveitä ja vähentää pakettien fragmentointitarvetta. HIP-yhteydenmuodostuksen raskasta laskentaa vaativassa Diffie-Hellman-avainten luonnissa käytetään laskentakuormituksen hajautusta, jolloin se soveltuu myös langattomille anturiverkoille. Ehdotuksen mukaan anturi voi

suorittaa laskentaa vaativat toimenpiteet samassa verkossa olevalla tehokkaammalla laitteella.

HIP:n suorituskykyä on mitattu Jemaan et al. (2009) kokeellisessa tutkimuksessa, jossa mitattiin yhteydenmuodostukseen kuluvaa aikaa, paketin kulkuaikaa lähdelaitteesta kohteeseen ja takaisin ja tiedonsiirtonopeutta. HIP-toteutuksena käytettiin HIP for Linuxia. Siirtonopeuden mittaukset tehtiin käyttäen TCP- ja UDP-kuljetusprotokollia, HIP:n kanssa ja ilman. Yhteydenmuodostusajaksi määriteltiin aikaväli yhteydenmuodostuksen aloituksen ja ensimmäisen ESP-paketin lähetyksen välillä. Yhteydenmuodostuksen aikaa mitattiin Ethernet- ja WiFi-yhteyksillä kannettavan tietokoneen ja pöytätietokoneen välillä sekä WiFi-yhteydellä kannettavan tietokoneen ja tablettitietokoneen välillä. Kulkuajan mittaukset tehtiin Ethernetillä kannettavan ja pöytätietokoneen välillä ja WiFillä kannettavan ja tablettitietokoneen välillä.

Jemaan et al. (2009) mittauksissa yhteydenmuodostus kannettavan ja pöytätietokoneen välillä kesti Ethernetillä 188,418 ms ja WiFillä 170,144 ms. Pieni ero voi johtua verkkokorteista tai pakettien kulkureittien eroista testiverkoissa. Kannettavan ja tablettitietokoneen yhdistäminen WiFillä kesti 1409,971 ms. Mittausten mukaan suurin ero ajoissa syntyy aloittavan osapuolen käsitellessä ensimmäistä vastaanottamaansa pakettia R1. Tabletin heikompi laskentakapasiteetti näkyi myös paketin kulkuajan kasvuna. HIP:n käyttö laski siirtonopeuksia sekä TCP- että UDP-yhteyksillä kannettavan tietokoneen ja pöytätietokoneen välillä. Langallisen ja langattoman UDP-yhteyden nopeuden lasku oli noin 5 %. Langallisen TCP-yhteyden nopeuden lasku oli 2,75 % ja langattoman 5,8 %. Siirtonopeuden lasku oli huomattavasti suurempi tablettitietokoneella, jonka siirtonopeus laski noin 50 %.

4. TURVALLISUUSPARANNUKSET

Esineiden internetiin liittyy lukuisia tietoturvauhkia ja -haavoittuvuuksia, jotka johtuvat muun muassa verkkoon liitettyjen laitteiden lukumäärän kasvusta. IoT:ssa laitteet pitää pystyä identifioimaan luotettavasti, jotta tietoliikenteen turvallisuus sekä datan eheys ja luottamuksellisuus voidaan varmistaa. IoT-laitteiden tavoitettavuuden ja verkon toiminnan kannalta on tärkeää, että erilaiset verkkoon kohdistuvat hyökkäykset voidaan torjua. Merkittäviä uhkia IoT:lle ovat epäluotettavat valmisohjelmistot (firmware), käyttöjärjestelmien takaportit, haitalliset TLS- ja SSL-sertifikaatit ja salakuuntelu (Gilchrist 2017, s. 131-132).

Esineiden internetin turvallisuusvaatimukset voidaan jakaa vyörytys- ja toimintavaiheeseen. Vyörytysvaiheeseen (bootstrapping phase) kuuluvat laitteen autentikointi ja valtuutus (authorization). Sen vaatimuksia ovat inkrementaalinen käyttöönotto, identiteetin ja avaimen luominen, yksityinen tunnistautuminen ja ryhmien luonti. Toimintavaiheessa laitteet kommunikoivat verkon ja ohjelmistojen turvamekanismien mukaisesti. Toimintavaiheen vaatimuksia ovat päästä-päähän turvallisuus, tuki mobiliteetille ja ryhmän jäsenhallinta. (Heer et al. 2011)

HIP-yhteydenmuodostus on luotettava tapa luoda turvallinen tiedonsiirtoyhteys kahden laitteen välille. Identiteettiin sidottu yhteys parantaa sijainnin yksityisyyttä ja mahdollistaa turvallisen yhteyden säilymisen myös laitteiden liikkuessa verkkojen välillä. HIP:ta voidaan käyttää parantamaan IoT-verkkojen tietoturvaa ja suojaamaan niitä verkkohyökkäyksiltä.

4.1 MITM-hyökkäysten torjunta

Mies välissä –hyökkäys (man-in-the-middle, MITM) on hyökkäystyyppi, jossa kolmas osapuoli asettuu yhteyden osapuolten A ja B väliin esittäen olevansa A. Kolmas osapuoli C välittää A:n viestin B:lle, jonka jälkeen C suorittaa avaintenvaihdon B:n kanssa ja lähettää oman julkisen avaimensa C:lle. Tämän jälkeen C välittää B:n ensimmäisen viestin A:lle. C saa näin haltuunsa myös A:n julkisen avaimen. Nyt C voi purkaa B:n lähettämän datan salauksen, sillä salaus on tehty C:n julkisella avaimella. C voi salata viestin A:n julkisella avaimella ja lähettää sen A:lle, jolloin A ja B

ovat tietämättömiä välissä olevasta C:stä.

HIP:ssa suoja MITM-hyökkäyksiä vastaan voidaan toteuttaa hakemalla yhteydenmuodostuksen vastaavan osapuolen identiteetti sertifikaatista tai allekirjoitetusta DNS-alueesta, jolloin vastaanotettu R1-paketti voidaan vahvistaa luotettavaksi. Vastaava osapuoli voi vahvistaa yhdistävän osapuolen luotettavaksi I2-paketin vastaanoton jälkeen. Varmennus tapahtuu hakemalla yhdistävän osapuolen identiteetti turvalliselta DNS-alueelta tai luotetusta sertifikaatista. (Moskowitz et al. 2015)

HIP:n yhteydenmuodostuksen opportunistinen tila on altis MITM-hyökkäyksille, ja sitä tulee käyttää vain luotetussa ympäristössä. Yhteydenmuodostuksen alussa aloittava osapuoli ei tiedä vastaajan HIT:ta, jolloin kolmannen osapuolen on helpompi tunkeutua alkuperäisten osapuolten väliin. (Gurtov 2008, s. 85)

4.2 Palvelunestohyökkäyksiltä suojautuminen

Palvelunestohyökkäys (Denial of Service, DoS) on hyökkäys, jolla pyritään estämään verkon tai sen laitteiden normaali toiminta. Palvelunestohyökkäyksessä palvelimeen kohdistetaan niin kova rasitus, että sen muisti, laskentakapasiteetti tai kaistanleveys kuluu loppuun (Wu & Irwin 2013, s. 29). Hajautetussa palvelunestohyökkäyksessä (Distributed Denial of Service, DDoS) hyökkääjinä toimii joukko laitteita, jolloin yhden laitteen estäminen ei riitä hyökkäykseltä suojautumiseen.

HIP:n yhteydenmuodostuksessa vaadittu kryptografisen tehtävän ratkaisu antaa suojaa palvelunestohyökkäyksiä vastaan. Yhteydenmuodostuksessa vastaava osapuoli ei pidä kirjaa yhteydenmuodostuspyynnöistä, ennen kuin on vastaanottanut oikean ratkaisun lähetettyyn tehtävään. (Gurtov 2008, s. 51)

HIP-yhteydenmuodostus luo mahdollisuuksia uuden tyyppisille palvelunestohyökkäyksille, jos HIP:n käyttöönotossa ei huomioida niiden torjuntaan vaadittavia toimintoja. HIP-laite voi varautua I1-pakettien tulvalla toteutettuun palvelunestohyökkäykseen ennalta määritetyillä R1-paketeilla, jolloin niiden lähettäminen ei vaadi suurta laskentatehoa. Vastaajan täytyy kuitenkin rajoittaa yhteen osoitteeseen lähetettävien pakettien määrää ylikuormituksen estämiseksi. Lähettämällä väärennettyjä I1-paketteja voi aiheuttaa toiselle laitteelle R1-pakettien tulvan, joten HIP-laitteen on jätettävä huomioimatta tuntemattomista laitteista tulevat R1-paketit. I2-pakettien tulvalta suojautumisessa vastaaja alkaa hylkäämään saman väärän vastauksen sisältävät paketit tietyn vastaanotetun pakettimäärän jälkeen. (Moskowitz et al. 2015)

5. SIMULOINTI JA KÄYTTÖÖNOTTO

HIP:n käyttöönotto ei vaadi muutoksia sovelluksiin tai reitittimiin, eikä se vaikuta laitteen ominaisuuksiin kommunikoida HIP:ta tukemattoman laitteen kanssa (Nikander et al. 2010). HIP:ta on testattu IoT:ssa (Meca et al. 2013; Garcia-Morchon et al. 2013), tietokoneissa ja tablettitietokoneessa (Jemaa et al. 2009) ja langattomissa anturiverkoissa (Khurri et al. 2010; Kuptsov et al. 2012).

Ahmadin et al. (2017) mukaan HIP:n teknologinen kehitys on ollut menestyksekästä mobiliteetin ja verkkoturvallisuuden suhteen, mutta sen käyttöönotto on erittäin rajoittunutta. Julkaisussa verrattiin HIP:tä ja laajasti kaupallisesti käytössä olevaa mobile VPN:ää, jossa turvallisuus ja mobiliteetti toteutetaan IPsec:lla tai mobile IP:llä. Tulosten perusteella HIP on parempi ratkaisemaan internetin turvallisuus- ja mobiliteettiongelmia, mutta sen käyttöönotto on vaikeampaa kuin kilpailijoilla.

5.1 Simulointialusta HIPSim++

HIPSim++ on INET/OMNeT++:lle kehitetty simulointiympäristö, jolla voidaan tutkia ja testata HIP:n ja sen laajennuksien toimintaa. OMNeT++ (Objective Modular Network Testbed in C++) on diskreettien tapahtumien simulointiympäristö. INET on OMNeT++:lle asennettava paketti, joka sisältää simulointimalleja yleisille verkkoteknologioille ja protokollille.

HIPSim++ ei sisällä HIP:n käyttämiä kryptografisia menetelmiä eikä IPSsec-mallia, mutta sillä voidaan simuloida HIP-protokollan toimintaa mobiliteettiin ja moniverkotukseen liittyvissä skenaarioissa (Bokor et al. 2009). HIPSim++ on vanhentuneen HIP-spesifikaation mukainen, mutta se soveltuu HIP:n toiminnan havainnollistamiseen ja testaamiseen.

5.2 OpenHIP

OpenHIP on avoimen lähdekoodin HIP-ohjelmisto, joka sisältää tuen Windows-, Linux- ja Mac OS X -käyttöjärjestelmille. Lähdekoodin lisäksi tarjolla on valmiita

rpm-, deb- ja exe-asennuspaketteja. OpenHIP:n ohjelmisto ja dokumentaatio ovat MIT/Expat-lisensoitu. OpenHIP:n uusin versio 0.9 julkaistu maaliskuussa 2012. Uuden HIP-spesifikaation mukainen versio piti ilmestyä kesällä 2015, mutta sitä on lykätty toistaiseksi. (OpenHIP 2018) OpenHIP sisältää mahdollisuuden luoda omia rendezvous-servereitä ja laajentaa BIND 9.X DNS-palvelimet tukemaan HIP:ta (OpenHIP Usage 2018).

OpenHIP on rakennettu HIP:n ja sen laajennusten vanhempien spesifikaatioiden pohjalta, joten se ei ole täysin nykyisen HIP-protokollan mukainen. Kehitteillä oleva uusi Linux-versio tulee sisältämään HIP:n ja sen laajennusten uudempien määritelmien mukaiset ominaisuudet. (OpenHIP – BitBucket 2018)

5.3 HIP for Linux

HIP for Linux (HIPL) on Aalto-yliopiston ja Helsingin yliopiston yhteisen tietotekniikan tutkimuslaitoksen avoimen lähdekoodin ohjelmistoprojekti. HIPL:n asennusohjeet yleisimmille Linux-jakelupaketeille löytyvät projektin verkkosivuilta. HIPL:n viimeisin päivitys on julkaistu 10.9.2017. (HIPL in Launchpad 2018). HIPL tukee Linuxin lisäksi Android-käyttöjärjestelmää (HIP for Linux 2018).

HIPL:n kokeelliset suorituskykytestit (Koskela 2008; Jemaa et al. 2009) osoittavat HIPL:n toimivuuden ja HIP:n vaatiman laskentakapasiteetin vaikutuksen mobiililaitteilla. HIPL tukee HIP:n versioita 1 ja 2, jotka eivät ole uudemman version spesifikaation mukaan keskenään yhteensopivia.

6. YHTEENVETO

HIP on periaatteeltaan ja toiminnaltaan käyttökelpoinen protokolla esineiden internetissä. IP-osoitteen paikannin- ja identifiointiroolien erottaminen mahdollistaa yhteyksien sitomisen luotettaviiin identiteetteihin, jolloin yhteys voidaan säilyttää IP-osoitteiden vaihtuessa. Identiteettien käytöllä saavutetaan myös parempi sijaintianonymiteetti. Lisäksi laitteiden luotettava identifiointi helpottaa verkkohyökkäyksiltä suojautumista.

Laitteiden mobiliteetin parantaminen ja monilähetyksen tehostaminen ovat tärkeitä esineiden internetin kasvaessa. Tehokas monilähetys on tarpeellinen etenkin langattomissa anturiverkoissa, joissa yksittäinen anturi lähettää dataa useisiin eri kohteisiin. Liikkuvien laitteiden lukumäärän nopea kasvu vaatii verkoilta tehokasta mobiliteetin hallintaa. HIP-pohjainen mobiliteetin hallinta voidaan toteuttaa rendezvous-palvelimilla ja DNS-palvelimen laajennuksella. Laajasti käytössä oleva standardiprotokolla mobiliteetin saavuttamiseen on Mobile IP, joten HIP ei ole ainoa vaihtoehto mobiliteetin hallinnan toteutukseen. Myös monilähetykseen on olemassa muita protokollia. Ahmadin et al. (2017) vertailun mukaan HIP:n teknologiset ominaisuudet kilpailijoitaan paremmat ratkaisemaan internetin turvallisuuteen ja mobiliteettiin liittyviä haasteita, mutta olosuhteet sen laajalle käyttöönotolle eivät ole suotuisat.

HIP:n yhteydenmuodostuksen kryptografiset operaatiot vaativat etenkin aloittavan osapuolen laitteistolta laskentatehoa. Yhdistävän osapuolen tehtävänä on laskea arvo, jonka tiiviste sisältää tietyn määrän nollia. Lasketun ratkaisun tarkastaminen on huomattavasti kevyempää, jolloin korkeampi laskentarasitus kohdistuu yhdistävään osapuoleen. Yhdistävän osapuolen korkeampi laskentatehovaatimus on eduksi palvelunestohyökkäyksiltä ja tunkeutujilta suojauduttaessa. IoT:n laitemäärän kasvu lisää etenkin hajautettujen palvelunestohyökkäysten määrää. Verkkoon liitettyjen laitteiden monimuotoistuessa niihin kohdistuvat hyökkäykset voivat aiheuttaa entistäkin suurempaa vahinkoa, joten paremmat suojautumis- ja identifiointiteknologiat ovat tarpeellisia.

IoT-laitteiden kannalta erityisen kiinnostavia ovat HIP:n kevennetty versio ja ha-

jautettuun laskentaan perustuvat toteutukset. Käsiteltyjen testitulosten perusteella HIP-protokollan spesifikaation mukainen yhteydenmuodostus (HIP BEX) on liian raskas vähäresurssisille IoT-laitteille ja antureille. Kirjallisuudesta löytyy useita mahdollisia tapoja käyttää HIP:tä IoT:ssa ja langattomissa anturiverkoissa: kevyempi yhteydenmuodstus (HIP DEX), hajautettu laskenta (CHIP ja CD-HIP) ja HIP DEX yhdistettynä ennalta jaettuun avaimeen (HIP PSK). Hajautetussa mallissa laskentaa suoritetaan HIP-välityspalvelimella tai saman verkon tehokkaammalla laitteella. Kirjallisuuskatsauksen perusteella HIP on teknisesti edistyksellinen, mutta vähäisessä käytössä oleva protokolla. HIP:tä on testattu IoT:ssa, langattomissa anturiverkoissa ja tietokoneverkoissa, mutta viitteitä sen nopeasta yleistymisestä tai laajasta käytöstä kaupallisissa verkoissa tai järjestelmissä ei ole löydettävissä.

LÄHTEET

Ahmad, I., M. Liyanage, M. Ylianttila & A. Gurtov (2017). Analysis of deployment challenges of Host Identity Protocol. EuCNC 2017 - European Conference on Networks and Communications. IEEE, s. 1–6.

Balan, T., D. Robu & F. Sandu (2017). Multihoming for Mobile Internet of Multimedia Things. Mobile Information Systems 2017.

Bokor, L., S. Nováczki, L. T. Zeke & G. Jeney (2009). Design and evaluation of host identity protocol (HIP) simulation framework for INET/OMNeT++. Proceedings of the 12th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems - MSWiM '09. MSWiM '09. ACM, s. 124. Saatavissa: http://portal.acm.org/citation.cfm?doid=1641804.1641827.

Deering, S. & R. Hinden (2017). Internet Protocol, Version 6 (IPv6) Specification. RFC 8200, IETF. Saatavissa: https://tools.ietf.org/html/8200.

Garcia-Morchon, O., S. L. Keoh, S. Kumar, P. Moreno-Sanchez, F. Vidal-Meca & J. H. Ziegeldorf (2013). Securing the IP-based Internet of Things with HIP and DTLS. Proceedings of the Sixth ACM Conference on Security and Privacy in Wireless and Mobile Networks. WiSec '13. Budapest, Hungary: ACM, s. 119–124.

Garyfalos, A., K. Almeroth & J. Finney (2003). A Comparison of Network and Application Layer Multicast for Mobile IPv6 Networks. Proceedings of the 6th ACM International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems. MSWIM '03. San Diego, CA, USA: ACM, s. 58–65. Saatavissa: http://doi.acm.org/10.1145/940991.941003.

Gilchrist, A. (2017). IoT Security Issues. ID: 4810138. Boston: DEG Press. Saatavissa: http://ebookcentral.proquest.com/lib/tut/detail.action?docID=4810138.

Gladisch, A., R. Daher & D. Tavangarian (2014). Survey on Mobility and Multihoming in Future Internet. Wireless Personal Communications 74.1, s. 45–81. Saatavissa: https://doi.org/10.1007/s11277-012-0898-6.

Gurtov, A., M. Komu & R. Moskowitz (2009). Host Identity Protocol. The Internet Protocol Journal Vol 12, No. 1.

Gurtov, A. (2008). Host Identity Protocol (HIP) : towards the secure mobile Internet. Chichester: Wiley, 295 s.

Heer, T., O. Garcia-Morchon, R. Hummen, S. L. Keoh, S. S. Kumar & K. Wehrle (2011). Security Challenges in the IP-based Internet of Things. Wireless Personal Communications 61.3, s. 527–542.

Henderson, T, C. Vogt & J Arkko (2017a). Host Mobility with the Host Identity Protocol. RFC 8046, IETF. Saatavissa: https://tools.ietf.org/html/rfc8046.

Henderson, T, C. Vogt & J Arkko (2017b). Host Multihoming with the Host Identity Protocol. RFC 8047, IETF. Saatavissa: `https://tools.ietf.org/html/rfc8047`.

Jemaa, M. B., N. Abid, M. Laurent-Maknavicius & H. Chaouchi (2009). Experimental Measurements of Host Identity Protocol for Mobile Nodes' Networks. Journal of Computer Systems, Networks and Communications.

Jokela, P., R. Moskowitz & J. Melen (2015). Using the Encapsulating Security Payload (ESP) Transport Format with the Host Identity Protocol (HIP). RFC 7402, IETF. Saatavissa: `https://tools.ietf.org/html/7402`.

Kent, S. (2005). IP Encapsulating Security Payload (ESP). RFC 4303, IETF. Saatavissa: `https://tools.ietf.org/html/4303`.

Khurri, A., D. Kuptsov & A. Gurtov (2010). On application of host identity protocol in wireless sensor networks. 2010 IEEE 7th International Conference on Mobile Adhoc and Sensor Systems, MASS 2010, s. 538–545. Saatavissa: `http://ieeexplore.ieee.org/document/5663902`.

Komu, M., T. Henderson & H. Tschofenig (2006). Basic Host Identity Protocol (HIP) Extensions for Traversal of Network Address Translators. RFC5770, IETF. Saatavissa: `https://tools.ietf.org/html/rfc5770`.

Koskela, J. (2008). A HIP-based peer-to-peer communication system. 2008 International Conference on Telecommunications, s. 1–7.

Kuptsov, D., B. Nechaev & A. Gurtov (2012). Securing Medical Sensor Network with HIP. Wireless Mobile Communication and Healthcare. Toim. K. S. Nikita, J. C. Lin, D. I. Fotiadis & M.-T. Arredondo Waldmeyer. Berlin, Heidelberg: Springer Berlin Heidelberg, s. 150–157.

Kurose, J. F. & K. W. Ross (2010). Computer networking: a top-down approach. 5th, international. New York: Pearson.

Laganier, J. (2016). Host Identity Protocol (HIP) Domain Name System (DNS) Extension. RFC 8005, IETF. Saatavissa: `https://tools.ietf.org/html/rfc8005`.

Laganier, J. & F. Dupont (2015). An IPv6 Prefix for Overlay Routable Cryptographic Hash Identifiers Version 2 (ORCHIDv2). RFC 7343, IETF. Saatavissa: `https://tools.ietf.org/html/rfc7343`.

Laganier, J & L Eggert (2016). Host Identity Protocol (HIP) Rendezvous Extension. RFC 8004, IETF.

Meca, F. V., J. H. Ziegeldorf, P. M. Sanchez, O. G. Morchon, S. S. Kumar & S. L. Keoh (2013). HIP Security architecture for the IP-based internet of things. Proceedings - 27th International Conference on Advanced Information Networking and Applications Workshops, WAINA 2013, s. 1331–1336. Saatavissa: `http://eprints.gla.ac.uk/85514/`.

Moskowitz, R. & R. Hummen (2017). HIP Diet EXchange (DEX) draft-ietf-hip-dex-06. IETF. Saatavissa: `https://tools.ietf.org/html/draft-ietf-hip-dex-06`.

Moskowitz, R, T. Heer, P Jokela & T Henderson (2015). Host Identity Protocol Version 2 (HIPv2). RFC 7401, IETF. Saatavissa: `https://tools.ietf.org/html/rfc7301`.

Moskowitz, R. (1999). The Host Identity Payload. Saatavissa: `https://tools.ietf.org/html/draft-moskowitz-hip-00`.

Moskowitz, R. & P. Nikander (2006). Host Identity (HIP) Architecture. RFC4423, IETF. Saatavissa: `https://tools.ietf.org/html/rfc4423`.

Muslam, M. M., H. A. Chan, L. A. Magagula & N. Ventura (2012). Network-based mobility and Host Identity Protocol. IEEE Wireless Communications and Networking Conference, WCNC, s. 2395–2400.

Muslam, M. M., H. A. Chan & N. Ventura (2017). Distributed mobility management with mobile Host Identity Protocol proxy. EURASIP Journal on Wireless Communications and Networking 2017.1, s. 71. Saatavissa: `https://jwcn-eurasipjournals.springeropen.com/articles/10.1186/s13638-017-0853-z`.

Nikander, P., A. Gurtov & T. R. Henderson (2010). Host Identity Protocol (HIP): Connectivity, IPv4 and IPv6 Networks. IEEE Communications Surveys & Tutorials 12.2, s. 1–19.

Porambage, P., A. Braeken, P. Kumar, A. Gurtov & M. Ylianttila (2017). CHIP: Collaborative Host Identity Protocol with Efficient Key Establishment for Constrained Devices in Internet of Things. Wireless Personal Communications 96.1, s. 421–440.

Sahraoui, S. & A. Bilami (2015). Efficient HIP-based approach to ensure lightweight end-to-end security in the internet of things. Computer Networks 91, s. 26–45. Saatavissa: `https://search.proquest.com/docview/1727423537`.

Sousa, B. M., K. Pentikousis & M. Curado (2011). Multihoming Management for Future Networks. Mobile Networks and Applications 16.4, s. 505–517.

Stiemerling, M, J Quittek & L Eggert (2008). NAT and Firewall Traversal Issues of Host Identity Protocol (HIP) Communication. RFC 5207, IETF. Saatavissa: `https://tools.ietf.org/html/rfc5207`.

HIP for Linux (2018). Saatavissa (viitattu 14.10.2018): `http://mkomu.kapsi.fi/hipl/`.

HIPL in Launchpad (2018). Saatavissa (viitattu 14.10.2018): `https://code.launchpad.net/hipl`.

OpenHIP (2018). Saatavissa (viitattu 14.10.2018): `http://openhip.sourceforge.net/`.

OpenHIP – BitBucket (2018). Saatavissa (viitattu 14.10.2018): `https://bitbucket.org/openhip/openhip`.

OpenHIP Usage (2018). Saatavissa (viitattu 14.10.2018): `http://openhip.sourceforge.net/wiki/index.php/Usage`.

Tschofenig, H., A. Gurtov, J. Ylitalo, A. Nagarajan & M. Shanmugam (2005). Traversing Middleboxes with the Host Identity Protocol. Information Security and Privacy. Toim. C. Boyd & J. M. González Nieto. Berlin, Heidelberg: Springer Berlin Heidelberg, s. 17–28.

Wu, C.-H. J. & J. D. Irwin (2013). Introduction to Computer Networks and Cybersecurity. Bosa Roca: CRC Press, 1373 s.

Yassein, M. B., S Aljawarneh & W Al-Sarayrah (2017). Mobility management of Internet of Things: Protocols, challenges and open issues. 2017 International Conference on Engineering & MIS (ICEMIS), s. 1–8.

You, T. & H. Jung (2012). A qualitative analysis of MOFI, LISP, and HIP. 2012 International Conference on ICT Convergence (ICTC), s. 772–774.

Zhang, B., W. Wang, S. Jamin, D. Massey & L. Zhang (2006). Universal IP multicast delivery. Computer Networks 50.6. Overlay Distribution Structures and their Applications, s. 781 –806.

Zhu, X. & J. W. Atwood (2007). A Secure Multicast Model for Peer-to-Peer and Access Networks Using the Host Identity Protocol. 2007 4th IEEE Consumer Communications and Networking Conference, s. 1098–1102.

Zhu, X., Z. Ding & X. Wang (2011). A Multicast Routing Algorithm Applied to HIP-Multicast Model. 2011 International Conference on Network Computing and Information Security. Vol. 1, s. 169–174.