

Jussi Kankaansivu

AVOIN SIP-PUHELINJÄRJESTELMÄ

Tietotekniikan koulutusohjelma

Tietoliikenteen suuntautumisvaihtoehto

2009



TIIVISTELMÄ

Avoin SIP-puhelinjärjestelmä

Kankaansivu Jussi

Satakunnan ammattikorkeakoulu

Tietotekniikan koulutusohjelma

Kesäkuu 2009

Valvoja: Mustonen Arto

UDK: 004.057.8, 621.395

Sivumäärä: 46

Asiasanat: protokollat, nettipuhelut, atk-ohjelmat

Tässä opinnäytetyössä tutkittiin avoimeen lähdekoodiin perustuvaa OpenSIPS SIP-palvelinta ja sen toimintaa. Toiminnan tutkimiseksi ja testaamiseksi toteutettiin yksinkertainen SIP-palvelin NGN-labradorioon.

Teoriaosuudessa käsiteltiin työhön liittyvän ja keskeisen SIP-protokollan yleinen teoria ja muutama yksinkertainen esimerkkutilanne. Lisäksi käytiin soveltuvin osin lyhyesti läpi läheisesti SIP:n käyttöön liittyvien SDP- ja RTP-protokollien rakenne ja toiminta.

Varsinaisessa työosuudessa käytiin läpi OpenSIPS:in yleinen rakenne, asennus, perustoiminnan kannalta tärkeimmät moduulit ja hallinta. Lopuksi testattiin ja arvioitiin järjestelmän toiminta.

Opinnäytetyön vertailuosuudessa vertailtiin IP-puhelinjärjestelmistä OpenSIPSiä Asteriskiin, joka on moderneista IP-puhelinjärjestelmistä yleisimmin käytetty.

NGN-labradorioon rakennettua, OpenSIPS:ään perustuvaa ympäristöä voitaisiin tulevaisuudessa hyödyntää puhelintekniikan opetuksessa.

ABSTRACT

Opensource SIP-VoIP server
Kankaansivu Jussi
Satakunta University of Applied Sciences
Degree Programme in Information Technology
June 2009
Supervisor: Mustonen Arto
UDC: 004.057.8, 621.395
Number of Pages: 46
Keywords: SIP, VoIP, server

In this thesis the open source OpenSIPS SIP server and its operation was examined. To examine and test it was implemented as a simple SIP server for the NGN-laboratory.

In the theory section, essential general theory of the SIP protocol and some simple examples of the situations are covered. In addition, there is a brief overview of SDP and RTP protocols which are closely related to SIP.

The practical part deals with OpenSIPS's general design, installation, the main modules from view of the basic operation and management. Finally, the system was tested and assessed in action.

In the comparison part of this thesis OpenSIPS is compared to Asterisk, which is used most commonly in a modern IP telephony systems.

The OpenSIPS based environment, built for the NGN-laboratory, can be used in the education of telephone technology in the future.

SISÄLLYS

1 LYHENNELUETTELO.....	5
2 JOHDANTO.....	6
3 SIP.....	7
3.1 SIP-verkkoelementit.....	8
3.1.1 Käyttäjäagentti.....	8
3.1.2 Välityspalvelin.....	9
3.1.3 Registrar.....	9
3.1.4 Uudelleenohjauspalvelin.....	10
3.2.2 Pyyntösanomat.....	14
3.2.3 Vastausanomat.....	17
3.3 Esimerkkitalanteita.....	19
3.3.1 Rekisteröityminen palvelimelle.....	19
3.3.2 Soitto.....	20
3.4 Muita SIP:iin liittyviä protokollia.....	21
3.4.1 SDP.....	21
3.4.2 RTP.....	23
4 OPENSIPS.....	25
4.1 Yleistä OpenSIPS-toteutuksesta.....	25
4.2 Asennus.....	25
4.5 Asetukset.....	27
4.4 Moduulit.....	29
4.4.1 SI.....	29
4.4.2 Tm.....	29
4.4.3 Rr.....	30
4.4.4 Usrloc.....	30
4.4.5 Registrar.....	31
4.4.6 Maxfwd.....	31
4.4.7 Auth ja authdb.....	32
4.4.8 Presence ja presence_XML.....	32
4.4.9 Textops.....	33
4.4.10 Acc.....	34
4.4.11 Muita moduuleita.....	34
4.5 Hallinta.....	34
4.6 Palvelimen testaus.....	35
4.6.1 Käyttäjäagentin rekisteröityminen.....	36
4.6.2 Soitto käyttäjäagenttien välillä.....	38
4.6.3 Olotilan sanomia.....	41
5 VERTAILU OPENSIPS:IN JA ASTERISK:IN VÄLILLÄ.....	44
6 YHTEENVETO.....	45
LÄHTEET.....	46

1 LYHENNELUETTELO

GSM	Global System for Mobile communications
HTTP	Hypertext Transfer Protocol
IAX	Inter-Asterisk eXchange
IETF	Internet Engineering Task Force
IP	Internet Protocol
MGCP	Media Gateway Control Protocol
MMUSIC	Multiparty Multimedia Session Control
RFC	Request for Comments
RTP	Real-time Transport Protocol
RTSP	Real Time Streaming Protocol
SAP	Session Announcement Protocol
SCCP	Skinny Call Control Protocol
SDP	Session Description Protocol
SIP	Session Initiation Protocol
SMS	Short Message Service
SMTP	Simple Mail Transfer Protocol
SQL	Structured Query Language
TLS	Transport Layer Security
UA	User Agent
URI	Uniform Resource Identifier
VoIP	Voice over Internet Protocol
XML	Extensible Markup Language

2 JOHDANTO

Suuntauksena nykyään on, että reaaliaikainen viestintä on siirtymässä yhä enemmän ja vahvemmin jo tällä hetkellä käytössä olevien IP-pohjaisten verkkojen päälle. Reaaliaikaiseen viestintään tarvitaan ensinnäkin protokolla, jonka avulla voidaan muodostaa yhteys käyttäjien välille, siirtää reaaliaikaista tietoa ja päättää yhteys. Yhteyttä pitää myös kyetä seuraamaan läpi koko yhteyden keston. SIP-protokolla on tarkoitettu juuri tähän.

SIP-sanomien kulun ohjaamista ja seuraamista varten pitää olla jokin piste, jossa nämä toiminnot voidaan toteuttaa. Tätä toiminnallisuutta tässä opinnäytetyössä tutkittiin. Työssä toteutettiin OpenSIPS-ohjelmiston avulla SIP-palvelin, jonka toiminta käytännössä testattiin.

Tässä opinnäytetyössä käydään SIP-protokollasta läpi monia eri asioita, joista olennaisimpina ja tärkeimpinä on mainittava SIP-protokollan käyttöön tarvittavia eri verkkokomponentteja, sanomia, sanomien sisältöä esitteleviä osuuksia ja näiden lisäksi mukaan on liitetty muutama esimerkki, miten sanomat kulkevat eri verkkokomponenttien välillä. Näiden lisäksi käydään läpi lyhyesti ja tiiviisti SIP:n käyttöön läheisesti liittyvät protokollat SDP ja RTP.

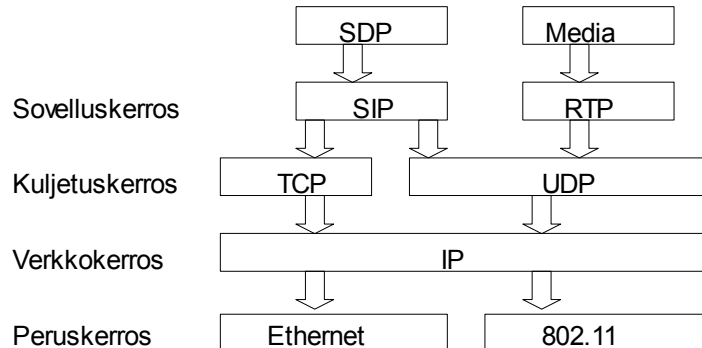
OpenSIPS:stä tarkistellaan läpi sen rakennetta, asennusta, tärkeimpiä asetuksia, tärkeimpiä moduuleita sekä hallintaa. Lopuksi testataan OpenSIPS:n toimintaa soittamalla puheluita järjestelmän avulla ja kaappaamalla sanomaliikennettä eri toiminnoista.

3 SIP

SIP on IETF:n määrittelemä protokolla. Se on ensi kerran määritelty standardissa RFC 2543 vuonna 1999. Myöhemmin sen on korvannut RFC 3261 vuonna 2002. Sen on kehittänyt IETF:n MMUSIC työryhmä, mutta nykyisin sitä varten on perustettu oma standardointiryhmänsä. /4/

SIP on tekstipohjainen protokolla. Siihen on otettu ominaisuuksia HTTP:stä ja SMTP:stä siten, että HTTP:stä on otettu asiakas-palvelimen suunnittelu ja osoitteiden muodostaminen ja SMTP:stä sanomien muotoilu ja otsikointi. /3/

SIP on sovellustason ohjausprotokolla, jonka ensisijainen tehtävä on luoda, muuttaa ja päättää sessioita päätepisteiden välillä. Päätepisteitä kutsutaan käyttäjäagenteiksi. Jo olemassa olevaan sessioon voidaan kutsua lisää osallistujia. Kuvassa 1. esitetään miten SIP:n käyttöön vaadittavat protokollat sijoittuvat IP-protokollapinossa. /2/

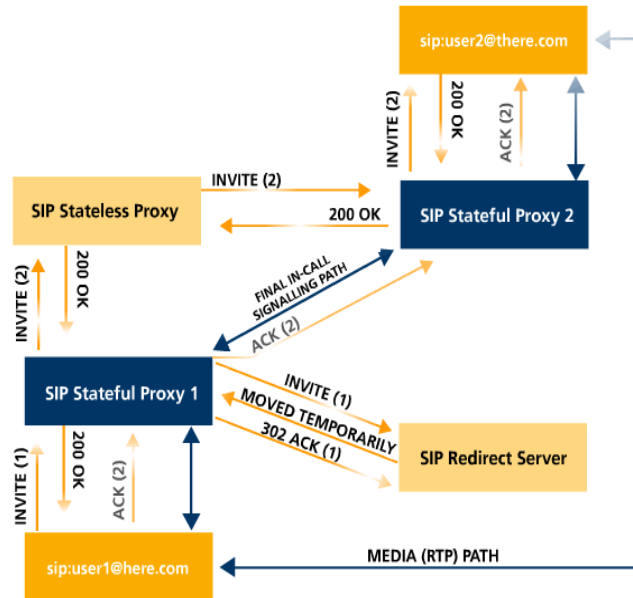


Kuva 1. Protokollapino

SIP-protokollaa voidaan käyttää myös muihin toimintoihin, esimerkiksi lyhytviestien lähettämiseen ja olotilan ilmaisemiseen. /4/

3.1 SIP-verkkoelementit

RFC 3261:ssä määritellään neljä eri SIP-verkon osaa. Määritellyt osat ovat käyttäjäagentti ja kolme erilaista palvelinta, jotka ovat välitys-, rekisteröinti- ja uudelleenohjauspalvelin. Palvelintoiminnot voivat sijaita samassa laitteessa. Kuvassa 2. on esitetty miten VoIP-puhelu kulkee SIP-arkkitehtuurissa. /2/



Kuva 2. VoIP-puhelu SIP-arkkitehtuurissa /1/

3.1.1 Käyttäjäagentti

Käyttäjäagentit edustavat käyttäjille näkyvää osaa SIP-verkosta. Käyttäjäagentti voi olla IP-puhelin, joka käyttää SIP-protokollaa, SIP-asiakasohjelmaa tietokoneella, matkapuhelin tai jokin muu päätelaite. Käyttäjäagentit muodostuvat kahdesta osasta eli käyttäjäagenttiasiakkaasta ja käyttäjäagenttipalvelimesta. Asiakas luo pyyntöjä ja lähettää ne verkkoon käyttäjän valintojen perusteella. Palvelin vastaanottaa pyyntöjä verkosta ja luo niille vastheet. /2/,/3/,/4/

3.1.2 Välityspalvelin

Välityspalvelin ottaa vastaan viestejä käyttäjäagenteilta ja toisilta välityspalvelimilta. Palvelin välittää sanomat eteenpäin tai vastaa niihin. Välittääkseen viestin eteenpäin palvelin hakee tiedon seuraavasta kohteesta tietokannasta tai sijaintipalvelusta. Viestit voivat kulkea usean välityspalvelimen kautta kulkiessaan käyttäjäagentilta toiselle. Jokainen matkan varrella oleva välityspalvelin tutkii sanoman kentät ja niiden perusteella se päättää, minne ne lähettää sanoman tai miten sitä käsitellään. Palvelin lisää ainakin viestin otsikko-osaan tiedon siitä, että sanoman on kulkenut palvelimen kautta. Palatessaan takaisin sanomat kulkevat samaa reittiä käänteisessä järjestyksessä. Välityspalvelimia on kahdenlaisia, tilattomia ja tilallisia.

Tilallinen välityspalvelin pitää kirjaa vastaanottamistaan viesteistä. Se luo jokaiselle viestille oman tapahtuman, jonka avulla se voi seurata sanomien kulkua. Palvelin seuraa tapahtumaa loppuun asti, jonka jälkeen se voi poistaa sanomien seuraamiseen tarvittavat tiedot pois muististaan.

Tilaton välityspalvelin ei pidä kirjaa välittämistään viesteistä. Se ottaa viestit vastaan suoraan kuljetuskerrokselta ja välittää ne eteenpäin viestin sisältämän tiedon perusteella. Ohjaamisen jälkeen palvelin ei säilytä tietoja viestistä vaan poistaa ne. /2/,/3/

3.1.3 Registrar

Rekisteröintipalvelin vastaa käyttäjäagenttien rekisteröinnistä verkkoon. Se vastaanottaa vain SIP REGISTER -pyyntöjä. Muihin pyyntöihin sen pitää lähettää *501 Not Implemented* -vastaussanoma. Onnistuneen rekisteröitymisen jälkeen rekisteröintipalvelin tallentaa tilaajan tiedot sijaintipalveluun, josta muut palvelimet voivat hakea tietoja tilaajasta. Rekisteröintipalvelin voi vaatia tilaajan autentikointia. Ilman autentikointia luvattomat käyttäjät voivat kaapata saapuvat puhelut. Tietoturvan vuoksi rekisteröintipalvelin saattaa vaatia käytettäväksi TLS-salausta rekisteröinnin

aikana. /3/

3.1.4 Uudelleenohjauspalvelin

Uudelleenohjauspalvelin ei ohjaa saamiaan pyyntöjä eteenpäin. Se vastaa pyyntöihin lähettämällä takaisin vastauksena ilmoituksen siitä, missä tavoiteltu palvelu tai käyttäjäagentti sijaitsee. Tiedot se hakee samalla tavalla kuin välityspalvelin. Tämän jälkeen pyynnön tehnyt käyttäjäagentti voi lähettää pyynnön samaansa osoitteeseen ja lähettää kuittaussanomana uudelleenohjauspalvelimelle. Tämä mahdollistaa sen, että SIP-verkko voi siirtää vastuun reitityksestä käyttäjäagentille, kun on tarvetta vähentää esimerkiksi merkinantoverkon kuormitusta. Tilanteessa, jossa se vastaanottaa CANCEL-pyyntöä, pitää sen vastata siihen *200 OK* -vastauksella. /2/,/3/

3.2 Sanomat

3.2.1 Sanomien rakenne

Pyyntörivi

```
INVITE sip:1111@192.168.10.75:11420 SIP/2.0
```

Pyyntörivillä kerrotaan, mikä pyyntö on kyseessä. Seuraavana on tavoiteltu osoite ja viimeisenä on käytetyn SIP-protokollan versionumero. RFC 2543 ja RFC 3261 ovat versiota 2.0

Vastausrivi

```
SIP/2.0 200 OK
```

Vastausrivillä ensimmäisenä on käytetyn SIP-standardin versionumero ja sen jälkeen ilmoitetaan, mikä vastaussanoma on kyseessä.

Molemmissa esiintyviä otsikkokenttiä

To

To-kenttää käytetään ilmaisemaan sanoman vastaanottaja. Lainausmerkkien sisällä on tavoitellun käyttäjän nimi ja kulmasulkujen sisällä käyttäjän SIP URI. SIP URIa ei käytetä koskaan reititykseen. Esimerkki:

```
To: "2222 (Softphone)"<sip:2222@192.168.10.59>
```

From

Tässä on samat tiedot kuin To-kentässä, mutta päinvastoin. Tag-parametri on sattumanvarainen tunniste, jonka avulla tietty puhelu voidaan tunnistaa. Esimerkki:

```
From: "1111"<sip:1111@192.168.10.59>;tag=ba6a4715
```

Contact

Contact-kenttä sisältää URI:n, josta voi myös tavoittaa pyyntösanoman lähettäjän tai pyydetyn resurssin. Kentässä voi olla useita URI-osoitteita ja ne erotellaan pilkuin. Esimerkki:

```
Contact: <sip:1111@193.166.152.166:32122>
```

CSeq

Cseq kertoo, mikä on lähetetyn sanoman järjestysnumero. Molemmilla sessioon osallistuvilla osapuolilla on omat järjestysnumeronsa. Tämän avulla voidaan erottaa uudet pyynnot uudelleenlähetyksistä tai määritellä, mihin sanomaan CANCEL- ja ACK-sanomat viittaavat. Esimerkki:

```
CSeq: 1 INVITE
```

Via

Via-kenttää käytetään tallentamaan sanoman reitti. Ensimmäinen sen sisällössä on SIP:n versio ja käytetty yhteysprotokolla. Sitten tulee osoite, jonka kautta sanoma on

kulkenut. Jokainen välipiste matkanvarrella lisää oman osoitteensa aina ylimmäksi. Branch-parametri on tiiviste pyyntö-URI:sta ja To-, From-, Call-ID- ja Csekkenttiensisällöistä. Esimerkki:

```
Via: SIP/2.0/UDP 193.166.152.166:32122;branch=z9hG4bK-d8754z-6c3e485e9604f942-1---d8754z-
```

User-Agent

Tällä kentällä voidaan välittää tietoa sanoman lähettäneestä käyttäjäagentista. Tätä tietoa voidaan käyttää sanomien muotoiluun, kun lähettäjälle lähetetään sanomia. Esimerkiksi, jos tiedetään, että kyseinen käyttäjäagentti ei ymmärrä jollain tavalla muotoiltuja sanomia oikein. Esimerkki:

```
User-Agent: X-Lite release 11001 stamp 47546
```

Call-ID

Call-ID sisältää satunnaisen uniikin merkkijonon, jonka avulla tapahtuma käyttäjäagenttien välillä voidaan tunnistaa. Merkkijonon pitää olla sama kaikissa samaan sessioon liittyvissä sanomissa. Esimerkki:

```
Call-ID: Mz1hMjdkY2UyYjlkNGI3NmUyOTE2NTA5M2RjYzc1NTc.
```

Record-Route

Tämä kenttä kertoo, että kaikkien tätä seuraavien viestien on kuljettava osoitteen ilmoittaman välityspalvelimen kautta. Esimerkki:

```
Record-Route: <sip:192.168.10.59>
```

Expires

Tällä kentällä kerrotaan, kuinka kauan sanoman sisältö on voimassa. Ajan voi ilmoittaa sekunteina kuinka kauan sanoma on voimassa tai päivämäärä milloin sanoma vanhenee. Esimerkki Expires-kentästä:

Expires: 3600

Vain pyyntösanomissa esiintyviä kenttiä

Max-Forwards

Tämä kenttä kertoo, kuinka monta hyppyä sanoma saa tehdä matkallaan käyttäjäagentilta toiselle. Jokainen matkanvarrella oleva välityspalvelin vähentää arvosta yhden. Jos välityspalvelin saa sanoman, jossa arvo on nolla, se lähettää alkuperäiselle lähettäjälle kuittaussanoman *483 Too Many Hops*. Esimerkki:

```
Max-Forwards: 70
```

Allow

Allow-kenttä kertoo, mitä metodeja sanoman lähettänyt käyttäjäagentti tukee. Esimerkki:

```
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, NOTIFY, MESSAGE,
SUBSCRIBE, INFO
```

Authorization

Tämän kentän sisältönä ovat autentikointiin liittyvät tiedot. Esimerkki:

```
Authorization: Digest username="2222",realm="192.168.10.59",
nonce="49fe9f5b000000166c373d8aab6beae4304e8da685ad522",
uri="sip:192.168.10.59",
response="013625f5a4afdb8c477eb45e730c56d7",
algorithm=MD5
```

Reason

Tällä kentällä on mahdollista ilmaista CANCEL- ja BYE -sanomien yhteydessä, miksi yhteys purettiin. Esimerkki:

```
Reason: SIP;description="User Hung Up"
```

Vain vastaussanomissa esiintyviä otsikkokenttiä

Server

Tällä kentällä voidaan kertoa tietoa palvelimesta, joka on luonut vastaussanoman.

Esimerkki:

```
Server: OpenSIPS (1.5.1-notls (i386/linux))
```

WWW-Authenticate

Tätä kenttää käytetään *401 Unauthorized* -vastauksessa ilmaisemaan, että pyyntöä varten täytyy tunnistautua ennen pyynnön toteuttamista. Samassa siirretään autentikoinnin vaatimat tiedot. Esimerkki:

```
WWW-Authenticate: Digest realm="192.168.10.59",  
nonce="49fe9f5b0000000166c373d8aab6beae4304e8da685ad522"
```

Viestivartaloon liittyvät otsikkokentät

Content-Type

Tämä kertoo mahdollisen sanoman rungon sisältämän sisällön Mime-tyypin. Esimerkki:

```
Content-Type: application/pidf+xml
```

Content-Length

Content-Length ilmoittaa sanoman rungon sisältämän oktettien määrän. Esimerkki:

```
Content-Length: 470
```

3.2.2 Pyyntösanomat

SIP-protokollassa on 14 erilaista pyyntösanomaa. Niitä kutsutaan myös metodeiksi. RFC 3261-standardissa niistä on määritelty kuusi kappaletta. Muut on määriteltyinä erillisissä RFC-dokumenteissa. Niillä käyttäjäagentti pyytää toiselta käyttäjäagentilta tai palvelimelta jotain toimintoa. Välityspalvelimen ei tarvitse ymmärtää pyynnön sisältöä,

mutta sen pitää osata ohjata viesti oikeaan paikkaan. Käyttäjäagentin tulee vastata *501 Not Implemented*, jos se saa pyynnön, jota käyttäjäagentti ei tue. /2/,/3/

INVITE-metodia käytetään muodostamaan sessio kahden käyttäjäagentin välillä. Useimmiten INVITE sisältää myös viestin rungossa soittajan tukemat mediamuodot, jos niitä ei lähetetä INVITE:n mukana, niin ne lähetetään ACK:n rungossa. Se voi myös sisältää muita tietoja. Sanoman pakolliset otsikkotiedot ovat: Call-ID, CSeq, From, To, Via, Contact ja Max-Forwards.

ACK

ACK-metodilla kuitataan viimeinen sanoma INVITE-pyyntöissä. Se voi sisältää SDP-protokollasanoman, jos alkuperäinen INVITE-sanoma ei sitä sisältänyt. /3/

BYE

BYE-metodilla päätetään käynnissä oleva mediasessio. Sen voivat lähettää vain sessioon osallistuvat käyttäjäagentit. /3/

CANCEL

CANCEL-metodilla keskeytetään kesken oleva sessionmuodostus. /3/

OPTIONS

Tällä metodilla käyttäjäagentti kysyy toiselta käyttäjäagentilta tai välityspalvelimelta sen ominaisuuksia. Siihen vastataan vastaussanomalla, joka sisältää Allow, Accept, Accept-Encoding, Accept-Language ja Supported -otsikoissa listat käyttäjäagentin tai palvelimen tukemista ominaisuuksista. /3/

REGISTER

Tällä metodilla käyttäjäagentti ilmoittaa rekisteröintipalvelimelle, mikä on sen nykyinen osoite, josta sen voi tavoittaa. Rekisteröityminen ei ole välttämätöntä puheluisen soittamiseen, mutta puheluiden vastaanottamiseen se on./3/

PRACK

PRACK-metodia käytetään kuittaamaan lähetettyjen väliaikaisten vastaussanomien (1xx) vastaanotto. Muiden vastaussanomien kuittaamiseen käytetään ACK-metodia. Tämä sanoma lähetetään, kun vastaanotettu vastaussanoma sisältää RACK-otsikon./3/

SUBSCRIBE

SUBSCRIBE-metodilla käyttäjäagentti pyytää, että sille lähetetään sanomia tietyistä tapahtumista, esimerkiksi olotilasta, kuten esimerkiksi, milloin käynnissä oleva puhelu päättyy. Ilmoitukset eri tapahtumista lähetetään NOTIFY-metodilla./3/

NOTIFY

NOTIFY-metodilla ilmoitetaan tapahtumista, joita on pyydetty ilmoitettavaksi SUBSCRIBE-metodilla aiemmin./3/

PUBLISH

Tällä metodilla lähetetään olotilan tiedot olotila-palvelimelle, josta muut käyttäjät voivat hakea tiedon. Sanoma voidaan myös lähettää suoraan toiselle käyttäjäagentille. Tiedot olotilasta ovat sanoman vartalossa XML-muodossa./3/

INFO

Tällä metodilla voidaan lähettää tietoja käynnissä olevasta sessiosta ilman, että sen ominaisuuksia muutetaan mitenkään. /3/

REFER

REFER-metodilla pyydetään toista käyttäjäagenttia avaamaan jokin URI tai URL. Sen avulla voidaan esimerkiksi siirtää puhelu toiselle käyttäjälle kesken puhelun./3/

MESSAGE

Tätä metodia käytetään käyttäjäagenttien välisissä pikaviesteissä. Lähetettävä teksti kuljetetaan sanoman runko-osassa. /3/

UPDATE

UPDATE-metodilla voidaan muuttaa session parametreja ennen kuin mediasessio on alkanut. /3/

3.2.3 Vastaussanomien luokat

Vastaussanomia käytetään vastaamaan käyttäjäagenttiasiakkaan tekemiin pyyntöihin. Ne jakaantuvat kuuteen erilaiseen luokkaan. Luokat erotellaan vastauksen ensimmäisen numeron perusteella. Osa niistä on lainattu suoraan HTTP:stä, mutta joidenkin merkitys voi olla erilainen käytettäessä SIP:n yhteydessä. Osa niistä on tarkoitettu vain SIP:n yhteydessä käytettäväksi. Useimmiten ne ovat muotoa ”x8x”, jotteivät ne menisi ristiin HTTP:n kanssa. Suurin osa vastauksista on määritelty RFC 3261-dokumentissa. /2/,/3/

1XX

Nämä vastaukset ilmaisevat, missä vaiheessa puheluiden muodostus on menossa. Esimerkiksi *180 Ringing*-vastaus kertoo, että vastaanottajan päässä hälytetään. 1XX-vastauksen luokat on esitetty taulukossa 1. /3/

Taulukko 1. Tilapäisten vastaukset /8/

100	Trying
180	Ringing
181	Call Is Being Forwarded
182	Queued
183	Session Progress

2XX

Tämän luokan vastaukset ilmoittavat pyynnön onnistuneen tai että se on hyväksytty. Esimerkiksi *200 OK*:lla voidaan ilmoittaa, että INVITE-sanoma on vastaanotettu onnistuneesti. Taulukossa 2. on esitetty 2XX-luokan vastaukset. /3/

Taulukko 2. Onnistumista ilmaisevat vastaukset /8/

200	OK
202	Accepted

3XX

3XX-vastaukset antavat tietoa tavoitellun käyttäjän tai palvelun vaihtoehtoisesta

sijainnista. Taulukossa 3. on esitetty 3XX-luokan vastaukset. /3/

Taulukko 3. Uudelleenohjaavat vastaukset /8/

300	Multiple Choices
301	Moved Permanently
302	Moved Temporarily
305	Use Proxy
380	Alternative Service

4XX

Näillä palvelin tai käyttäjäagenttipalvelin ilmoittaa, ettei sanoman käsittely onnistu siinä muodossa kuin se on lähetetty. Virheen numero ilmaisee käyttäjäagenttiasiakkaalle, miksi sitä ei voitu toteuttaa. Virheen numeron perusteella käyttäjäagentti osaa mahdollisesti lähettää pyynnön uudelleen oikeanlaisena. Taulukossa 4. esitellään 4XX-luokan vastaukset. /3/

Taulukko 4. Epäonnistumista ilmaisevat vastaukset. /8/

400	Bad Request	429	Provide Referrer Identity
401	Unauthorized	433	Anonymity Disallowed
402	Payment Required	436	Bad Identity-Info
403	Forbidden	437	Unsupported Certificate
404	Not Found	438	Invalid Identity Header
405	Method Not Allowed	440	Max-Breadth Exceeded
406	Not Acceptable	470	Consent Needed
407	Proxy Authentication Required	480	Temporarily Unavailable
408	Request Timeout	481	Call/Transaction Does Not Exist
410	Gone	482	Loop Detected
412	Conditional Request Failed	483	Too Many Hops
413	Request Entity Too Large	484	Address Incomplete
414	Request-URI Too Long	485	Ambiguous
415	Unsupported Media Type	486	Busy Here
416	Unsupported URI Scheme	487	Request Terminated
417	Unknown Resource-Priority	488	Not Acceptable Here
420	Bad Extension	489	Bad Event
421	Extension Required	491	Request Pending
422	Session Interval Too Small	493	Undecipherable
423	Interval Too Brief	494	Security Agreement Required
428	Use Identity Header		

5XX

Nämä vastaukset ilmoittavat, että jostain syystä palvelin ei pysty käsittelemään saatua pyyntöä. Sanoma voi sisältää tiedon siitä, milloin kannattaa yrittää uudelleen. Taulukossa 5. on esitetty palvelimen erilaisia häiriötä ilmaisevat viestit. /3/

Taulukko 5. Palvelimen häiriötä ilmaisevat viestit. /8/

500	Server Internal Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Server Time-out
505	Version Not Supported
513	Message Too Large
580	Precondition Failure

6XX

Tämä vastausluokka ilmoittaa, että pyyntö on epäonnistunut ja ettei sitä pitäisi yrittää lähettää uudelleen, edes muille palvelimille. Taulukossa 6. on esitetty 6XX-luokan vastaussanomiat. /3/

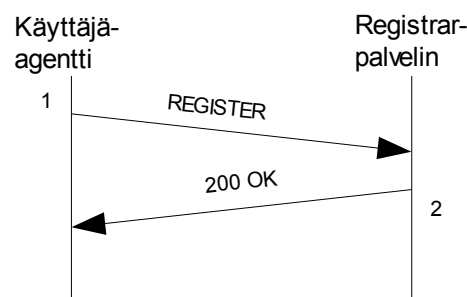
Taulukko 6. Globaalit häiriöt /8/

600	Busy Everywhere
603	Decline
604	Does Not Exist Anywhere
606	Not Acceptable

3.3 Esimerkkitilanteita

3.3.1 Rekisteröityminen palvelimelle

Kuvassa 3. esitetään esimerkki sanomaliikenteestä, kun rekisteröidytään käyttäen autentikointia.

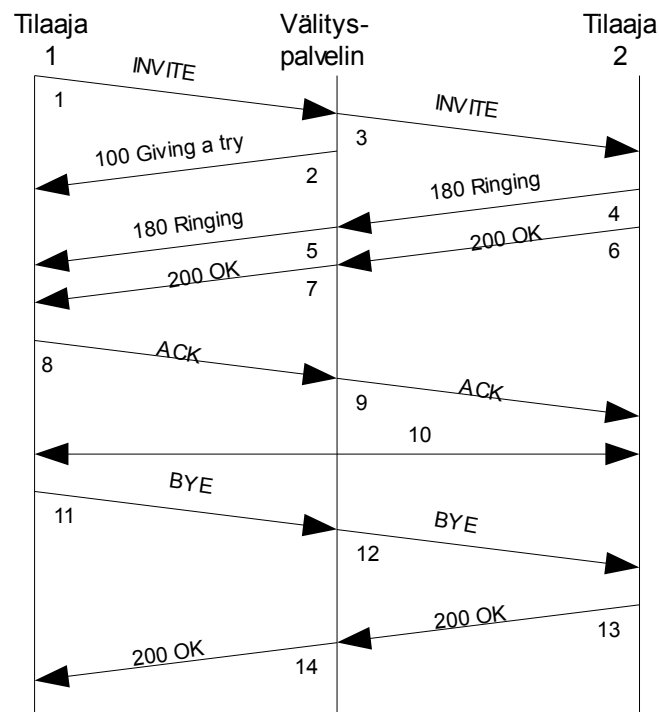


Kuva 3. Rekisteröinnin sanomaliikenne

1. Käyttäjäagentti lähettää palvelimelle rekisteröintipyynnön, jossa ovat käyttäjän tiedot.
2. Palvelin hyväksyy rekisteröintipyynnön ja lähettää takaisin ilmoituksen, että tapahtuma on onnistunut.

3.3.2 Soitto

Kuvassa 4. on esitetty esimerkki sanomaliikenteestä, kun tilaaja 1 soittaa tilaajalle 2 välityspalvelimen avulla.



Kuva 4. Puhelun sanomaliikenne

1. Tilaajalta 1 lähtee INVITE-sanoma kohti välityspalvelinta.
2. Palvelin vastaanottaa sanoman ja lähettää tilaajalle 1 ilmoituksen, että sanoma on matkalla eteenpäin.
3. Palvelin lähettää INVITE-sanoman tilaajalle 2 haettuaan sijaintitietopalvelusta tiedon sijainnista. 4. ja 5. Tilaaja 2 on vastaanottanut INVITE-sanoman ja lähettää ilmoituksen, että käyttäjää hälytetään. Palvelin välittää sanoman tilaajalle 1.
6. ja 7. Tilaaja 2 lähettää 200 OK-vastauksen merkiksi siitä, että INVITE-sanoma on käsitelty ja hyväksytty. Palvelin välittää sanoman tilaajalle 1.
8. ja 9. Tilaaja kuittaa saaneensa tarvittavat tiedot puhelun alkamiseksi ja lähettää ACK-

sanoman, jonka palvelin välittää tilaajalle 2 ja puhelu alkaa.

10. Kun tarvittavat tiedot puhelun aloittamista varten on saatu, tilaajien välille muodostetaan RTP-yhteys.

11. ja 12. Tilaaaja 1 päättää puhelun lähettämällä sanoman BYE. Palvelin välittää sanoman tilaajalle 2.

13. ja 14. Tilaaaja 2 kuittaa BYE-sanoman vastaanotetuksi lähettämällä *200 OK*-sanoman. Puhelu on päättynyt.

3.4 Muita SIP:iin liittyviä protokollia

3.4.1 SDP

SDP on RFC 4566 -standardissa esitelty tapa, jolla voidaan kuvailla kahden käyttäjäagentin välisessä multimediasessiossa käytettäviä mediavirtoja ja niiden alustusparametreja. Mediavirta voi olla tyypiltään ääni, video, teksti, sovellus tai viesti. SDP:tä voidaan käyttää muun muassa seuraavien protokollien kanssa: SAP, SIP, RTSP ja HTTP. /3/

SDP-viestin sisältönä ovat tiedot käytettävästä IP-osoitteesta, porttinumeroista, median tyypeistä ja median koodauksesta. Lisäksi sen sisältönä on tietoa session aiheesta, alkamis- ja päättymisaika ja yhteystietoja. SDP:ssä käytetään tekstipohjaista koodausta. Se muodostuu kentistä, joiden nimet on lyhennetty. Kenttien tulee olla tietyssä järjestyksessä käsittelyn yksinkertaistamiseksi. Taulukossa 7. on esitelty eri SDP:n kentät. /3/

Taulukko 7. SDP:n kentät

Kenttä	Käyttötarkoitus	Pakollinen
v=	Protokollan versio numero	X
o=	Session tunniste ja luoja	X
s=	Session nimi	X
i=	Session tiedot	
u=	URI, jossa enemmän tietoa sessiosta	
e=	Sähköpostiosoite	
p=	Puhelinnumero	
c=	Yhteyden tiedot	X
b=	Kaistanleveys	

t=	Session alkamis- ja päättymisaika	X
r=	Toistokerrat	
z=	Aikavyöhykkeen korjaukset	
k=	Salausavain	
m=	Median tiedot	
a=	Median ominaisuudet	

SIP:n yhteydessä soittava osapuoli yleensä lähettää listan mediaominaisuuksistaan INVITE- tai ACK-pyyntösanoman vartalossa. Puhelun vastaanottava osapuoli lähettää omat ominaisuutensa *200 OK*-vastaussanomassa. Niitä voidaan lähettää muidenkin sanomien yhteydessä. Median tietojen ja ominaisuuksien pitää olla yhtenevät molemmilla session osapuolilla. SIP:n yhteydessä ei käytetä kaikkien kenttien tietoja. Kaikki pakolliset kentät kuitenkin lähetetään yhteensopivuuden vuoksi. SIP käyttää SDP:stä yhteyden- ja median tiedot sekä median ominaisuudet. /3/

v=

Ilmoittaa SDP:n versionumeron. Tämän hetkinen versionumero on 0.

o=

Tämä kertoo session perustajan ja tunnistein. Näillä voidaan erottaa yksittäinen sessio muista. Kenttä on muotoa o=[käyttäjänimi] [session tunniste] [session versio] [verkon tyyppi] [osoitteen tyyppi] [osoite]. /3/

s=

Tämä kenttä sisältää tiedon session nimestä. Kentän pitää sisältää vähintään yksi merkki. /3/

c=

Tässä kentässä sisältönä on tieto mediayhteydestä. Kentän muoto on c=[verkon tyyppi] [osoitteen tyyppi] [yhteysosoite]. /3/

t=

Kenttä kertoo session alkamis- ja päätymisajan. SIP:ssä molempien arvoina käytetään nollaa, joka tarkoittaa että sessio on pysyvä. Se on muotoa t=[alkamisaika]

[päättymisaika]. /3/

m=

Tällä kerrotaan median tyyppi, jota käytetään sessiossa. Se on muotoa m=[media]
[porttinumero] [kuljetusprotokolla] [lista median formaateista|käytettävän median
formaatti]. /3/

a=

Näillä kentillä on mahdollista kertoa tarkemmista asetuksista edellisissä kentissä olleille
asetuksille tai jokin muu asetus. Se on muotoa a=[attribuutti]:[arvo] muille asetuksille
a=[asetus]. Esimerkiksi a=sendrecv kertoo että avattavan yhteyden pitää olla
kaksisuuntainen. Mahdollisia attribuutteja ovat rtpmap ja fntp. Esimerkiksi rtpmap:illa
kerrotaan RTP:llä siirrettävien formaattien asetuksia. Se on muotoa rtpmap:
[formaatin_numero] [nimi]/[kello_taajuus] [mahdolliset_pakkauksen_parametrit] /3/

Seuraavassa on esimerkki SDP-sanomasta:

```
v=0
o=- 6 2 IN IP4 192.168.10.90
s=CounterPath X-Lite 3.0
c=IN IP4 192.168.10.90
t=0 0
m=audio 19360 RTP/AVP 107 119 100 106 0 105 98 8 101
a=fntp:101 0-15
a=rtpmap:107 BV32/16000
a=rtpmap:119 BV32-FEC/16000
a=rtpmap:100 SPEEX/16000
a=rtpmap:106 SPEEX-FEC/16000
a=rtpmap:105 SPEEX-FEC/8000
a=rtpmap:98 iLBC/8000
a=rtpmap:101 telephone-event/8000
a=sendrecv
```

3.4.2 RTP

RTP on standardissa RFC 3550 esitelty tapa siirtää IP-verkossa reaaliaikaista dataa,
kuten ääntä ja liikkuvaa kuvaa, mutta siirrettävä voi myös olla jotain muuta tietoa, joka
vaatii reaaliaikaista tiedonsiirtoa. /3/

RTP ei huolehdi tiedonsiirron luotettavuudesta, vaan se jättää luotettavuuden kuljetuskerroksen huolehdittavaksi. RTP mahdollistaa erilaisten tiedonsiirto-ongelmien havaitsemisen. Sen avulla voidaan havaita pakettien katoaminen, vaihteleva siirron viive, pakettien väärässä järjestyksessä saapuminen ja pakettien epäsymmetrinen reititys. /3/

RTP on binäärimuotoinen protokolla. Sen otsikko on vähintään 12 oktettia pitkä, minkä lisäksi siinä voi olla vielä ekstra-kenttiä. Käytettävät kentät on esitelty taulukossa 8. /3/

Taulukko 8. RTP:n kentät

Kenttä	Pituus (bittinä)	Tarkoitus/Merkitys
Version	2	Versionumero. Tämän hetkinen on 2
Padding	1	Kertoo onko loppuun lisätty oktetteja, että kaikki paketit olisivat yhtä pitkiä
Extension	1	Kertoo, onko lisätty laajennuskenttiä
CSRC count	4	Kertoo CSRC-kenttien määrän
Marker	1	Ilmaisee uuden kehyksen alkua siirrettävässä datassa
Payload Type	7	Kertoo käytettävän koodekin. Sama arvo kuin SDP:ssä
Sequence Number	16	Paketin järjestysnumero
Timestamp	32	Aikaleima
SSRCI	32	Paketin lähettäjän yksilöivä tunniste
SCRC	32	Session osallistuvan lähteen tunniste. Voi olla 0-15 kappaletta.

RFC 3550:ssä on myös määritelty RTCP-protokolla, jolla RTP-sessioon osallistuvat voivat lähettää toisilleen tietoja session kulusta. Näitä tietoja ovat lähetettyjen ja vastaanotettujen pakettien määrä, kadonneitten pakettien määrä ja viiveen vaihtelu./3/

4 OPENSIPS

4.1 Yleistä OpenSIPS-toteutuksesta

OpenSIPS on GPL-lisensoitu avoimenlähdekoodin toteutus SIP-palvelimesta. Se koostuu erilaisista moduuleista, joita voidaan ottaa käyttöön tarpeen mukaan. Moduuleja on lähes 100 kappaletta, joista osa on vakaita, osa kehitysversioita. Osa moduuleista ei toimi itsenäisinä, vaan tuovat lisää toiminnallisuutta johonkin toiseen moduuliin. Palvelin noudattaa RFC 3261 -standardissa määriteltyä SIP:iä.

OpenSIPS voi toimia SIP-verkossa useissa eri toiminnoissa. Se voi toimia registrar-, sijainti-, välitys- ja uudelleenohjauspalvelimena. Se sisältää myös sovellustason toimintoja normaalien palvelintoimintojen lisäksi. /5/ ja /6/

Mahdolliset käyttöalustat ovat Linux/i386, Linux/armv4l, FreeBSD/i386, OpenBSD/i386 Solaris/sparc64 sekä NetBSD/sparc64. /7/

4.2 Asennus

OpenSIPS-ohjelmiston asentamiseen on kaksi eri vaihtoehtoa. Ohjelmisto voidaan asentaa joko kääntämällä suoraan lähdekoodista tai asentamalla paketeista, joissa se on jo valmiiksi käännetty. Molemmat vaihtoehdot löytyvät osoitteesta <http://opensips.org/pub/opensips/latest/>. Src-hakemistosta löytyy lähdekoodi ja packages-hakemistosta löytyvät asennuspaketit itse OpenSIP:lle ja käytetyimmille moduuleille.

Palvelinohjelmiston asennusta varten pitää ensin asentaa tarvittavia kirjastoja ja ohjelmistoja. Useimpien Linux-jakeluversioiden pakettienhallintasovelluksesta löytyvät kaikki tarpeelliset kirjastot ja ohjelmat. OpenSIPS:in asennuksessa tarvitaan seuraavat kirjastot: bison tai yacc, flex, sed, tr, GNU tar, GNU install, openssl, libsetp,

libmysqlclient, libz, libpq, libexpat, libxml2, libradius-ng, libxmlrpc-c3, libperl, libsnmp9, libldap ja libconfuse. Ohjelmistoista tarpeellinen on jokin tietokantapalvelinsovellus kuten esimerkiksi MySQL, jonka asennus onnistuu useimmiten paketinhallintasovelluksen kautta.

Valmiiksi käännettyistä paketeista asentaminen on melko helppoa. Lataamisen jälkeen käyttäjä menee hakemistoon, johon paketit latasi. Hakemistossa suoritetaan komento **dpgk -i <paketin_nimi>**. Komento toistetaan jokaiselle paketille, joka halutaan asennettavaksi. Mikäli jokin tarvittava kirjasto puuttuu, niin paketinasennusohjelma ilmoittaa siitä.

Esimerkiksi, kun asennetaan ydinkomponentti, tietokannan käyttöön tarvittavat moduulit ja presence-moduulin annettavat komennot ovat:

- **dpgk -i opensips_1.5.1-1_i386.deb**
- **dpgk -i opensip-mysql-module_1.5.1-1_i386.deb**
- **dpgk -i opensip-precense-module_1.5.1-1_i386.deb.**

Lähdekoodista kääntäminen alkaa sillä, että ladataan lähdekoodit sisältävä paketti. Paketti puretaan komennolla **tar xzcf <paketin_nimi>**. Sen jälkeen muokataan Makefile-tiedostoa jollakin tekstieditoreista ja siirretään *exclude_modules?*-parametrin ne moduulit, jotka halutaan kääntää peruspakettien lisäksi, *include_modules?*-parametrin yhteyteen. Muutosten jälkeen tiedosto tallennetaan ja käyttäjä palaa komentoriville. Sen jälkeen kääntäminen tapahtuu **make prefix=/ all**-komennolla. Kun käännös on mennyt läpi ilman virheitä, asennetaan käännetty ohjelmisto komennolla **make prefix=/ install**.

Kun asennus on suoritettu, jommalla kummalla tavalla, määritellään asetukset. Asetusten määrittelyn jälkeen OpenSIPS-palvelin voidaan käynnistää komennolla **/etc/init.d/opensips start**.

4.5 Asetukset

OpenSIP:ssä on kolme erillistä asetustiedostoa, `opensips.cfg`, `opensipsctlc` ja `/etc/default-hakemistosta` löytyvä `opensips`, jolla määritellään, suostuuko OpenSIPS käynnistymään. `Opensips.cfg` ja `opensipsctlc` löytyvät molemmat `/etc/opensips-hakemistosta`. `Opensips.cfg`:ssä määritellään, mitä moduuleita ladataan, mitkä ovat niiden asetukset ja OpenSIPS:n reititys. `Opensipsctlc`-tiedostossa määritellään OpenSIP:in hallintaan käytettyjen `opensipsctl-` ja `opensipsdbctl`-sovellusten käyttöön liittyviä asetuksia.

OpenSIPS:n perusasetukset ja sen moduulien asetukset määritellään `opensips.cfg`:ssä. Aluksi tiedosto sisältää globaaleja asetuksia. Tärkeimmät niistä ovat porttinumero ja TLS:n käyttöön liittyvät asetukset. Porttinumeron avulla palvelin vastaanottaa saapuvat sanomat. Samassa perusasetusten osuudessa ovat myös debuggaukseen liittyvät asetukset, joilla voidaan selvittää palvelimen toimintahäiriöitä tarpeen vaatiessa. Seuraavassa esitetään esimerkkinä debuggaukseen liittyvät asetukset:

```
debug=6 /* Määrittää tason kuinka tarkkaan palvelimen toimintaa
tarkkaillaan */
fork=no /* Ei luoda lapsiprosesseja virheen löytämisen
helpottamiseksi */
log_stderr=yes /* Määrittää mihin virheilmoitukset
kirjoitetaan */
```

Seuraavassa osuudessa on määritelty moduulit, jotka ladataan käyttöön palvelinta käynnistettäessä, ja kerrotaan missä ladattavat moduulit sijaitsee. Sijainti määritellään asetuksella, joka on muotoa **`mpath="hakemisto_jossa_moduulit_sijaitsevat"`**. Sen jälkeen jokaiselle käyttöön ladattavalle moduulille on rivinsä, joka on muotoa: **`loadmodule "moduulin_nimi"`**.

Moduulien lataamisen jälkeen niille määritellään asetukset, jotka ovat muotoa: **`modparam("moduuli", "parametri", arvo)`** Jos arvo on aakkosellinen, niin sen pitää olla heittomerkkien sisällä. Numeraaliset arvot ovat ilman heittomerkkejä. Seuraavassa esitetään esimerkkinä `presence`-moduulin asetukset:

```

modparam("presence|presence_xml", "db_url",
         "mysql://opensips:opensipsrw@localhost/opensips")
modparam("presence_xml", "force_active", 1)
modparam("presence", "server_address", "sip:localhost:5060")

```

Viimeisenä määritellään OpenSIP:n reitityslogiikka, jonka perusteella OpenSIP päättää, miten se käsittelee vastaanottamansa sanomat. Reititys tapahtuu erilaisten lauseiden avulla, joissa käytetään moduulien tarjoamia funktiota. Mahdollisia lauseita ovat if-lauseet, switch-lauseet ja while-lauseet.

If-lauseet toteutetaan, mikäli lauseen sisältämä ehto täyttyy. Switch-lauseissa ehtoina käytetään erilaisia pseudomuuttujia. While-lauseet toteutetaan, kun jokin muuttuja saavuttaa tietyn arvon.

OpenSIP:n hallinta työkalujen (opensipsctl ja opensipsdbctl) käyttöön liittyvät asetukset löytyvät opensipsctltc-tiedostosta. Asetukset ovat muotoa [**säädetävä_asetus**]=[**arvo**]. Opensipsctl:n käyttöön asetuksilla voidaan esimerkiksi määrittellä erilaiset tarvittavat salasanat ja käyttäjätunnukset tietokannan käyttöön. Seuraavassa on esitetty joitakin siihen liittyviä asetuksia:

- SIP_DOMAIN - käytössä oleva domain
- DBENGINE - käytössä olevan tietokannan tyyppi
- DBHOST - osoite, jossa tietokanta sijaitsee
- DBNAME - tietokannan nimi
- DBWUSER - tietokannan käyttäjätunnus, jolla on myös kirjoitusoikeudet
- DBRWPW - edellisen salasana
- DBOUSER - käyttäjätunnukset, joilla on vain lukuoikeus
- DBROPW - edellisen salasana.

Opensipsdbctl:iä varten opensipsctltc:ssä voi määrittellä, mitä tauluja se luo tietokantaan.

- INSTALL_EXTRA_TABLES - asennetaanko extra_module-asetuksessa määritellyt taulut
- INSTALL_PRESENCE_TABLES - asennetaanko olotilaan liittyvät taulut
- STANDARD_MODULES - mitä standardi moduulien tauluja asennetaan

- EXTRA_MODULES - mitä extra moduulien tauluja asennetaan.

4.4 Moduulit

OpenSIPS rakentuu erilaisista moduuleista. Moduulit tuovat eri toimintoja käytettäväksi reitityslogiikassa. Toimintoja käytetään moduulien tarjoamien funktioiden kautta. Osa moduuleista voi tarvita toisen moduulin tarjoamia toiminnallisuuksia toimiakseen itse. /5/ ja /6/

4.4.1 Sl

Sl-moduuli mahdollistaa palvelimen tilattoman toiminnan, eli se ei tallenna muistiin tietoja vastaanotetusta sanomasta. Tilaton toiminta aktivoidaan **sl_send_reply(code, reason)-** tai **sl_reply_error()**-funktiolla halutuille pyyntösanomille, pyyntösanomien ominaisuuksien perusteella reitityslogiikassa. Tätä käytetään silloin, kun ei ole tarvetta odottaa varmistusta siitä, että lähetetty vastesanomaa on päässyt perille. Seuraavassa esitetään esimerkkinä if-lause, jonka perusteella SL-moduuli lähettää vastauksena, ettei palvelu ole saatavilla. /10/

```
if (is_method("PUBLISH"))
{
    sl_send_reply("503", "Service Unavailable");
    exit;
}
```

4.4.2 Tm

Tm-moduulin tarjoamat funktiot mahdollistavat palvelimen tilallisen toiminnan, jolloin se tallentaa sessioon liittyvät vastaanottamansa sanomat, kunnes sessio saadaan päätökseen. Esimerkiksi soitto toiselle käyttäjälle käsitellään useimmiten tilallisena ellei jostain syystä sitä haluta käsitellä tilattomasti. Yleisemmin käytettyjä moduulin tarjoamia funktioita ovat: /10/

- **t_newtran()** - Tällä luodaan uusi tapahtuma, jota voidaan seurata.
- **t_relay()** - Välittää sanomat eteenpäin niiden käsittelyn jälkeen.
- **t_check_trans()** - Tarkastaa, liittyykö vastaanotettu sanoma johonkin jo seurattuun tapahtumaan.

4.4.3 Rr

Rr-moduulin funktioilla analysoidaan ja käsitellään sanomien reititykseen liittyviä kenttiä. Nämä kentät ovat Record-Route ja Route. Niiden sisällön perusteella päätetään, minne sanoma lähetetään eteenpäin. /10/

Loose_route()- funktiolla tarkastetaan sanoman Route-kentät. Jos niitä ei löydy, toteutetaan reititys muista kentistä löytyvillä tiedoilla. Kun löytyy vain yksi Route-kenttä, joka sisältää paikallisen välityspalvelimen osoitteen, se poistetaan ja toimitaan kuin sitä ei olisi ollutkaan. /10/

Muissa tilanteissa toimitaan kuten RFC 3261:ssä määritellään. Jos Request-URI löytyy palvelimen käyttämästä sijaintipalvelusta, vaihtaa palvelin sen siihen; muussa tilanteessa kenttää ei muuteta. Seuraavaksi tarkistetaan ylin Route-kenttä, ja jos se on palvelimen oma osoite, se poistetaan. Sitten sanoma lähetetään ylimmäksi jääneeseen osoitteeseen. /10/

Record_route()-funktioilla kirjoitetaan uusi Record-Route-kenttä ylimmäksi.

4.4.4 Usrloc

Usrloc on moduuli, joka ylläpitää taulukkoa käyttäjien sijainneista ja joka toimii sijaintitietopalveluna. Taulukkoa voidaan ylläpitää keskusmuistissa, jolloin tiedot katoavat palvelimen sammutuksen yhteydessä, tai tietokannassa, jolloin tiedot säilyvät palvelimen sammutuksenkin jälkeen. Moduulilla ei ole varsinaisia omia funktioita, vaan muut moduulit käyttävät sitä omien funktioidensa kautta. /10/

4.4.5 Registrar

Registrar-moduuli mahdollistaa OpenSIPS:n toiminnan registrar-palvelimena. Sen avulla voidaan käsitellä ainoastaan REGISTER-pyyntösanomaa. Käyttäjän sijaintitiedot se tallentaa USRLOC-moduulin avulla. /10/

Käyttäjien sijaintitietojen lisäys, poistaminen ja päivittäminen tapahtuu **save()**-funktioilla. Pyynnön onnistuessa siihen vastataan *200 OK*-vastauksella. /10/

Lookup()-funktion avulla voidaan etsiä käyttäjän yhteystietoja sijaintitiedoista. Jos niitä ei löydy, palautetaan virheilmoitus. Jos löytyy, pyyntö-URI ylikirjoitetaan sillä yhteystiedolla, jolla on korkein prioriteetti. /10/

4.4.6 Maxfwd

Tällä moduulilla tarkistetaan ja muokataan Max-Forwards-kenttää SIP-sanomista. **Max_limit(kokonaisluku)**-parametrillä määritellään, mikä on sallittu maksimiarvo eteenpäin lähetettävän sanoman Max-Forwards-kentässä, ja jos arvo ylittää maksimiarvon, muutetaan arvo maksimiarvon mukaiseksi. /10/

Mf_process_maxfwd_header(maksimi_arvo)-funktioilla tarkistetaan Max-Forwards-kentän arvo ja siitä vähennetään yksi. Jos arvo on nolla, lopetetaan sanoman käsittely, ellei jotain ole määritelty jotain tehtäväksi siinä tilanteessa. Jos kenttää ei löydy, se lisätään ja sen arvoksi asetetaan funktiossa ilmoitettu arvo. Seuraavassa esitetään esimerkkinä Max-Forwards-kentän tarkistuskomentosarja: /10/

```
if (!mf_process_maxfwd_header("10"))
```

4.4.7 Auth ja authdb

Auth- ja authdb-moduulit mahdollistavat kaikki autentikointia vaativat toiminnot, esimerkiksi käyttäjän tunnistaminen rekisteröitymisen yhteydessä. Kun käytetään pelkästään auth-moduulia, käytettävien tunnusten ja salasanojen pitää olla asetustiedostossa. Authdb-moduuli lisää mahdollisuuden käyttää tietokantaa autentikoititietojen säilyttämiseen. /10/

Autentikointiin on käytettävissä kaksi eri funktiota **www_authorize(realm, table)** ja **proxy_authorize(realm, table)**. Niiden erona on se, mitä kenttää käytetään pyytämään käyttäjäagenttia suorittamaan autentikointi. /10/

Seuraavassa annetaan esimerkki, jolla autentikointi suoritetaan kun vastaanotetaan register-pyyntö:

```
if (!www_authorize("localhost", "subscriber"))
{
    www_challenge("localhost", "0");
    exit;
}
```

4.4.8 Presence ja presence_XML

Presence- ja presence_XML-moduulit toteuttavat OpenSIPS:n toiminnan olotilapalvelimena. Presence-moduuli käsittelee ja luo olotilaan liittyvät sanomat. Käsiteltävät sanomat ovat PUBLISH, SUBSCRIBE ja NOTIFY. PUBLISH:in sisältönä ovat olotilaan liittyvät tiedot. SUBSCRIBE:lla kirjaudutaan vastaanottamaan olotilapäivityksiä. NOTIFY-sanomalla lähetetään tiedot olotilapäivityksistä niille, jotka ovat päivitystiedot tilanneet. Presence_XML-moduulilla käsitellään sanomien runko-osassa olevat xml-tiedot. /10/

Olotilasanomien käsittelyyn liittyvät funktiot ovat **handle_publish(char* sender_uri)**, jolla käsitellään PUBLISH-sanomat ja **handle_subscribe()**, jolla käsitellään SUBSCRIBE-sanomat. Kun vastaanotetaan PUBLISH-sanoma, sen sisältämät tiedot

tallennetaan tai sen sisältämien tietojen perusteella päivitetään jo ennestään tietokannassa olevat tiedot. Käsittelyn jälkeen lähetetään NOTIFY-sanoma. /10/

Seuraavassa on esimerkki, jossa käsitellään molemmat sanomat:

```

{
if (!t_newtran())
{
    sl_reply_error();
    exit;
};

if(is_method("PUBLISH"))
{
    handle_publish();
    t_release();
}
else
if( is_method("SUBSCRIBE"))
{
    handle_subscribe();
    t_release();
}

exit;
}

```

4.4.9 Textops

Textops-moduulin funktioilla tarkastellaan sanomia niiden sisältämien tekstien perusteella sekä tarvittaessa korvata on ne jollain toisella tekstillä. Tekstejä tarkastellaan säännöllisten lauseiden avulla. Esimerkiksi **is_method(metodi)**-funktioilla voidaan tarkistaa onko pyyntösanoman metodi jokin tietty. Esimerkkinä lause jossa tarkistetaan onko viestin metodi PUBLISH: /10/

```

if(is_method("PUBLISH"))

```

4.4.10 Acc

Acc-moduulia käytetään keräämään erilaisia tietoja puhelutapahtumista. Tapahtumasta tallennetaan pyyntömetodin nimi, From- ja To-otsikoista tag-parametri, puhelun id, viimeisen vastausviestin syykoodi sekä aika milloin tapahtuma päättyi. Kerättyjä tietoja on mahdollista tallentaa tietokantaan SQL:n avulla tai lähettää eteenpäin käyttäen syslogia tai RADIUS-palvelinta. Kerättyjä tietoja voisi käyttää laskutuksen perusteena, mutta koska OpenSIPS vain välittää sanomia, se ei saa tietoa siitä, jos käyttäjien välinen yhteys katkeaa. /10/

4.4.11 Muita moduuleita

Edellä esiteltyjen lisäksi OpenSIPS:tä löytyy vielä monia muita moduuleita. Niiden avulla voidaan määritellä palvelimen roolia SIP-verkossa tai lisätä ominaisuuksia, jotka laajentavat sen käyttömahdollisuuksia. Joitakin esimerkkejä tarjolla olevista moduuleista:

- SMS - mahdollistaa lyhytsanomien lähetyksen SIP-verkon ja GSM-verkon välillä
- Call Control - tuo prepaid ominaisuuden
- Load Balancer - mahdollistaa toiminnan kuormantasauspalvelimena
- Jabber - mahdollistaa pikaviestinnän ja olotilan välittämisen eri pikaviestiverkkoihin.

4.5 Hallinta

OpenSIP:n hallintaan käytetään opensipsctl- ja opensipsdbctl-sovelluksia. Opensipsctl:llä hallitaan palvelimen eri asetuksia ja opensipsdbctl:llä hallitaan OpenSIP:n käyttämiä tauluja tietokannassa ilman että käytetään tietokannan omia työkaluja siihen.

Opensipsctl:llä haetaan ja muokataan moduulien käyttämiä tietoja, jotka ovat tietokannassa tai keskusmuistissa, jos tietokantaa ei käytetä. Komennot ovat muotoa **opensipsctl toiminto [parametrit]**. Muutamia esimerkkejä komennoista:

- **add oppilas tentti** - Luo käyttäjän nimellä oppilas, jonka salasana on tentti
- **online** - näyttää online-tilassa olevat käyttäjät
- **monitor** - näyttää palvelimen tilan.

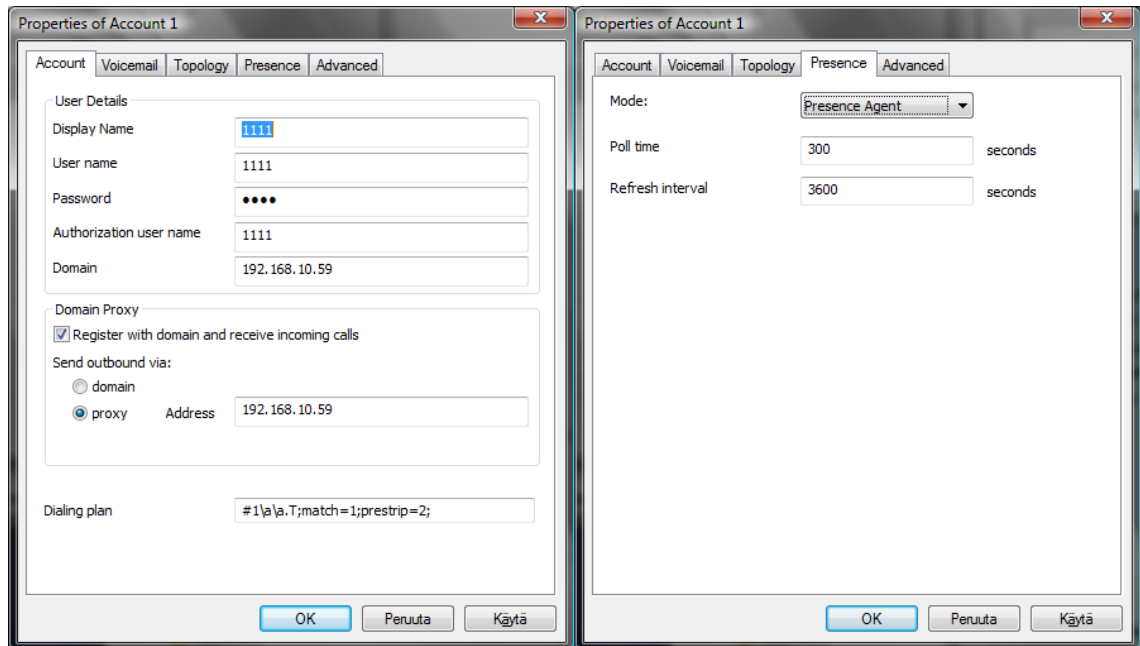
Opensipsdbctl:llä voidaan luoda ja poistaa OpenSIP:n moduulien tarvitsemia tauluja tietokantaan. Komennot ovat muotoa **opensipsdbctl toiminto [parametrit]**. Komennot annetaan suoraan järjestelmän komentoriviltä.

- **create** - luo taulut tietokantaan
- **drop** - tuhoa kaikki taulut
- **presence** - luo olotilaan liittyvät taulut.

4.6 Palvelimen testaus

Palvelimen toimintoja testattiin CounterPath:in X-Lite ohjelmalla. Testaaminen toteutettiin rekisteröitymällä palvelimelle, soittamalla puheluita käyttäjäagenttien välillä ja muuttamalla olotilan statusta. Palvelimen ja käyttäjäagenttien välistä liikennettä tarkkailtiin Wireshark-ohjelman avulla, palvelimelta ja työasemilta, joiden välillä koepuhelut soitettiin.

Seuraavassa kuvassa on esitetty käyttäjäagenttina toimineen X-Lite:n asetukset, joita käytettiin testaamisessa. Kuvassa 5. on esitetty käytetyt käyttäjätunnukset ja olotilaan liittyvät asetukset.



Kuva 5. X-Lite:n asetukset.

Wireshark-ohjelmalla kaapattiin liikennettä palvelimelta. Kaappauksista selviää miten SIP-sanomat ovat kulkeneet palvelimen ja käyttäjäagenttien välillä. Wireshark-ohjelma oli asennettu palvelimena toimineelle koneelle. Testitapaukset, joista kaapattiin liikennettä olivat käyttäjäagentin rekisteröityminen rekisteröintipalvelimelle, olotilaan liittyvät sanomat ja testipuhelu kahden käyttäjäagentin välillä.

4.6.1 Käyttäjäagentin rekisteröityminen

Seuraavassa esitetään rekisteröintitapahtumassa lähetettävät sanomat. Kuvassa 6. esitetään sanomaliikenne kun käyttäjä rekisteröityy palvelimelle ja kirjautuu samalla olotila palvelun käyttäjäksi.

Time	UA	Palvelin	Sanoma
	193.166.152.166	192.168.10.59	Comment
186,957	(32122)	Request: REGISTER s	SIP: Request: REGISTER sip:192.168.10.59
186,957	(32122)	Status: 401 Unautho	SIP: Status: 401 Unauthorized (0 bindings)
187,163	(32122)	Request: REGISTER s	SIP: Request: REGISTER sip:192.168.10.59
187,164	(32122)	Status: 200 OK (SIP: Status: 200 OK (1 bindings)

Kuva 6. Rekisteröitymisen sanomaliikenne

Seuraavassa on esitetty UA:n lähettämä REGISTER-pyyntö:

```
REGISTER sip:192.168.10.59 SIP/2.0
Via: SIP/2.0/UDP 193.166.152.166:32122;branch=z9hG4bK-d8754z-
ba492e6cfb306e72-1---d8754z-;rport
Max-Forwards: 70
Contact: <sip:1111@193.166.152.166:32122;rinstance=6a405babdf9b0270>
To: "1111"<sip:1111@192.168.10.59>
From: "1111"<sip:1111@192.168.10.59>;tag=14608123
Call-ID: NWEyNDQxMTQxYzE4MmMyOTZhMjk3NTZiZDBjMmU0NDg.
CSeq: 1 REGISTER
Expires: 3600
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE,
SUBSCRIBE, INFO
User-Agent: X-Lite release 11001 stamp 47546
Content-Length: 0
```

Palvelin ei hyväksy rekisteröintiä vaan lähettää takaisin vastaussanoman, jossa on tiedot autentikointia varten:

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP 193.166.152.166:32122;branch=z9hG4bK-d8754z-
ba492e6cfb306e72-1---d8754z-;rport=32122
To:
"1111"<sip:1111@192.168.10.59>;tag=c97b4d1cb1f3d0da549e06a8d482ef63.28
5d
From: "1111"<sip:1111@192.168.10.59>;tag=14608123
Call-ID: NWEyNDQxMTQxYzE4MmMyOTZhMjk3NTZiZDBjMmU0NDg.
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="192.168.10.59",
nonce="49fe9f52000000005dc0b853a829d6e02a03dcc97f5448af"
Server: OpenSIPS (1.5.1-notls (i386/linux))
Content-Length: 0
```

UA lähettää uudelleen REGISTER-sanoman, jossa on autentikointiin tarvittavat tiedot:

```
REGISTER sip:192.168.10.59 SIP/2.0
Via: SIP/2.0/UDP 193.166.152.166:32122;branch=z9hG4bK-d8754z-
e312e85b7e747c0e-1---d8754z-;rport
Max-Forwards: 70
Contact: <sip:1111@193.166.152.166:32122;rinstance=6a405babdf9b0270>
To: "1111"<sip:1111@192.168.10.59>
From: "1111"<sip:1111@192.168.10.59>;tag=14608123
Call-ID: NWEyNDQxMTQxYzE4MmMyOTZhMjk3NTZiZDBjMmU0NDg.
CSeq: 2 REGISTER
Expires: 3600
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE,
SUBSCRIBE, INFO
User-Agent: X-Lite release 11001 stamp 47546
Authorization: Digest
username="1111", realm="192.168.10.59", nonce="49fe9f52000000005dc0b853a
829d6e02a03dcc97f5448af", uri="sip:192.168.10.59", response="5b5e2bfc792
71bd1ca5551f877b2bda2", algorithm=MD5
Content-Length: 0
```

Saatuun REGISTER-sanoman uudestaan tarvittavilla tiedoilla palvelin lähettää takaisin 200 OK-vastauksen:

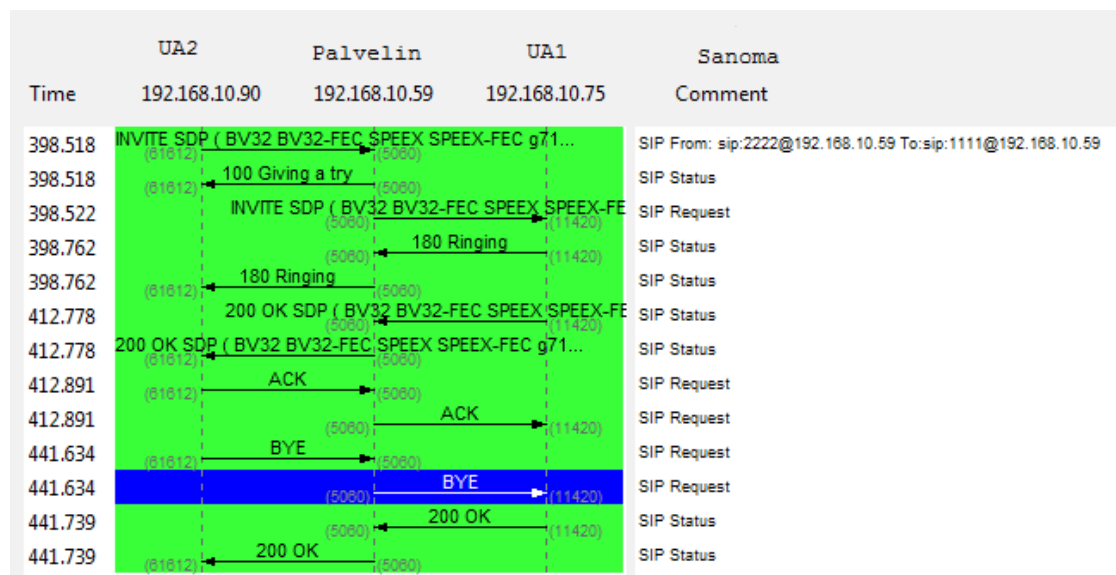
```

SIP/2.0 200 OK
Via: SIP/2.0/UDP 193.166.152.166:32122;branch=z9hG4bK-d8754z-
e312e85b7e747c0e-1---d8754z-;rport=32122
To:
"1111"<sip:1111@192.168.10.59>;tag=c97b4d1cb1f3d0da549e06a8d482ef63.bf
73
From: "1111"<sip:1111@192.168.10.59>;tag=14608123
Call-ID: NWEyNDQxMTQxYzE4MmMyOTZhMjk3NTZiZDBjMmU0NDg.
CSeq: 2 REGISTER
Contact:
<sip:1111@193.166.152.166:32122;rinstance=6a405babdf9b0270>;expires=36
00
Server: OpenSIPS (1.5.1-notls (i386/linux))
Content-Length: 0

```

4.6.2 Soitto käyttäjäagenttien välillä

Seuraavassa esitetään puhelun sanomaliikenne kun käyttäjä 1 soittaa käyttäjälle 2. Kuvassa 7. on esitetty sanomien suunnat graafisessa muodossa.



Kuva 7. SIP-puhelun sanomaliikenne

1. Seuraavassa esitetään UA2:n lähettämä INVITE-sanoma, joka sisältää SDP-protokollalla tiedot UA2:n tukemista mediamuodoista. Palvelin välittää sanoman UA1:lle:

```

INVITE sip:1111@192.168.10.75:11420;rinstance=48a8a5c6eaf6bc5c SIP/2.0
Record-Route: <sip:192.168.10.59;lr=on>
Via: SIP/2.0/UDP 192.168.10.59;branch=z9hG4bK04b3.bdd72593.0
Via: SIP/2.0/UDP
192.168.10.90:61612;received=192.168.10.90;branch=z9hG4bK-d87543-
510ee3531d5c855a-1--d87543-;rport=61612
Max-Forwards: 69
Contact: <sip:2222@192.168.10.90:61612>

```

```
To: "1111"<sip:1111@192.168.10.59>
From: "2222"<sip:2222@192.168.10.59>;tag=01367d2f
Call-ID: YjY2MjY5NDlhY2RhNjI4Mjc4MjU0N2Q2NWVmZWlzMjc.
CSeq: 1 INVITE
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE,
SUBSCRIBE, INFO
Content-Type: application/sdp
User-Agent: X-Lite release 1011s stamp 41150
Content-Length: 423
```

```
v=0
o=- 6 2 IN IP4 192.168.10.90
s=CounterPath X-Lite 3.0
c=IN IP4 192.168.10.90
t=0 0
m=audio 19360 RTP/AVP 107 119 100 106 0 105 98 8 101
a=alt:1 1 : t96kZqKN QUqhZVg6 192.168.10.90 19360
a=fmtp:101 0-15
a=rtpmap:107 BV32/16000
a=rtpmap:119 BV32-FEC/16000
a=rtpmap:100 SPEEX/16000
a=rtpmap:106 SPEEX-FEC/16000
a=rtpmap:105 SPEEX-FEC/8000
a=rtpmap:98 iLBC/8000
a=rtpmap:101 telephone-event/8000
a=sendrecv
```

2. Saatuaan INVITE-sanoman palvelin lähettää UA2:lle 100 Giving a try-sanoman:

```
SIP/2.0 100 Giving a try
Via: SIP/2.0/UDP 192.168.10.90:61612;branch=z9hG4bK-d87543-
510ee3531d5c855a-1--d87543-;rport=61612
To: "1111"<sip:1111@192.168.10.59>
From: "2222"<sip:2222@192.168.10.59>;tag=01367d2f
Call-ID: YjY2MjY5NDlhY2RhNjI4Mjc4MjU0N2Q2NWVmZWlzMjc.
CSeq: 1 INVITE
Server: OpenSIPS (1.4.2-notls (i386/linux))
Content-Length: 0
```

3. UA1:n lähettää 180 Ringing-sanoman UA2:lle välityspalvelimen kautta:

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP
192.168.10.90:61612;received=192.168.10.90;branch=z9hG4bK-d87543-
510ee3531d5c855a-1--d87543-;rport=61612
Record-Route: <sip:192.168.10.59;lr>
Contact: <sip:1111@192.168.10.75:11420;rinstance=48a8a5c6eaf6bc5c>
To: "1111"<sip:1111@192.168.10.59>;tag=5c622b22
From: "2222"<sip:2222@192.168.10.59>;tag=01367d2f
Call-ID: YjY2MjY5NDlhY2RhNjI4Mjc4MjU0N2Q2NWVmZWlzMjc.
CSeq: 1 INVITE
User-Agent: X-Lite release 1011s stamp 41150
Content-Length: 0
```

4. UA1 lähettää 200 OK-sanoman, jossa on mukana tiedot sen tukemista mediamuodoista SDP-protokollalla:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP
192.168.10.90:61612;received=192.168.10.90;branch=z9hG4bK-d87543-
510ee3531d5c855a-1--d87543-;rport=61612
Record-Route: <sip:192.168.10.59;lr>
Contact: <sip:1111@192.168.10.75:11420;rinstance=48a8a5c6eaf6bc5c>
To: "1111"<sip:1111@192.168.10.59>;tag=5c622b22
From: "2222"<sip:2222@192.168.10.59>;tag=01367d2f
Call-ID: YjY2MjY5NDlhY2RhNjI4Mjc4MjU0N2Q2NWVmZWlzMjc.
CSeq: 1 INVITE
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE,
SUBSCRIBE, INFO
Content-Type: application/sdp
User-Agent: X-Lite release 1011s stamp 41150
Content-Length: 524
```

```
v=0
o=- 1 2 IN IP4 192.168.10.75
s=CounterPath X-Lite 3.0
c=IN IP4 192.168.10.75
t=0 0
m=audio 49178 RTP/AVP 107 119 100 106 0 105 98 8 101
a=alt:1 3 : k3zr2uyE vB5U9zzO 192.168.10.75 49178
a=alt:2 2 : domDdRt5 af1PTzbi 192.168.50.1 49178
a=alt:3 1 : kZ3jQwi9 HskzUP3A 192.168.159.1 49178
a=fmtp:101 0-15
a=rtpmap:107 BV32/16000
a=rtpmap:119 BV32-FEC/16000
a=rtpmap:100 SPEEX/16000
a=rtpmap:106 SPEEX-FEC/16000
a=rtpmap:105 SPEEX-FEC/8000
a=rtpmap:98 iLBC/8000
a=rtpmap:101 telephone-event/8000
a=sendrecv
```

5. UA2:n lähettämä ACK-sanoma. Tämän jälkeen puhelu voi alkaa:

```
ACK sip:1111@192.168.10.75:11420;rinstance=48a8a5c6eaf6bc5c SIP/2.0
Via: SIP/2.0/UDP 192.168.10.59;branch=z9hG4bK04b3.bdd72593.2
Via: SIP/2.0/UDP
192.168.10.90:61612;received=192.168.10.90;branch=z9hG4bK-d87543-
0f29506b4129ab09-1--d87543-;rport=61612
Max-Forwards: 69
Contact: <sip:2222@192.168.10.90:61612>
To: "1111"<sip:1111@192.168.10.59>;tag=5c622b22
From: "2222"<sip:2222@192.168.10.59>;tag=01367d2f
Call-ID: YjY2MjY5NDlhY2RhNjI4Mjc4MjU0N2Q2NWVmZWlzMjc.
CSeq: 1 ACK
User-Agent: X-Lite release 1011s stamp 41150
Content-Length: 0
```

6. Puhelun aikainen RTP-virta joka, kulkee suoraan UA2:n ja UA1:n välillä kiertämättä palvelimen kautta.

7. UA2:n lähettämä BYE-sanoma, jolla se päättää puhelun:


```

BYE sip:1111@192.168.10.75:11420;rinstance=48a8a5c6eaf6bc5c SIP/2.0
Via: SIP/2.0/UDP 192.168.10.90:61612;branch=z9hG4bK-d87543-
4470946ce52cd76f-1--d87543-;rport
Max-Forwards: 70
Route: <sip:192.168.10.59;lr>
Contact: <sip:2222@192.168.10.90:61612>
To: "1111"<sip:1111@192.168.10.59>;tag=5c622b22
From: "2222"<sip:2222@192.168.10.59>;tag=01367d2f
Call-ID: YjY2MjY5NDlhY2RhNjI4Mjc4MjU0N2Q2NWVmZWlzMjc.
CSeq: 2 BYE
User-Agent: X-Lite release 1011s stamp 41150
Reason: SIP;description="User Hung Up"
Content-Length: 0

```

8. UA1 kuittaa puhelun päättämisen lähettämällä 200 OK-sanoman:

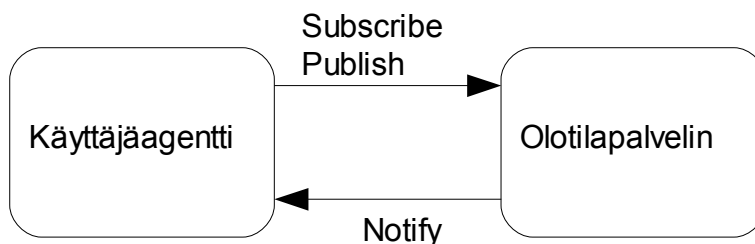
```

SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.168.10.59;branch=z9hG4bKd3b3.2ab05da4.0
Via: SIP/2.0/UDP
192.168.10.90:61612;received=192.168.10.90;branch=z9hG4bK-d87543-
4470946ce52cd76f-1--d87543-;rport=61612
Contact: <sip:1111@192.168.10.75:11420;rinstance=48a8a5c6eaf6bc5c>
To: "1111"<sip:1111@192.168.10.59>;tag=5c622b22
From: "2222"<sip:2222@192.168.10.59>;tag=01367d2f
Call-ID: YjY2MjY5NDlhY2RhNjI4Mjc4MjU0N2Q2NWVmZWlzMjc.
CSeq: 2 BYE
User-Agent: X-Lite release 1011s stamp 41150
Content-Length: 0

```

4.6.3 Olotilan sanomia

Seuraavassa esitetään olotilaan liittyviä sanomia. Kuvassa 8. esitetään olotilaan liittyvät sanomat ja niiden suunta.



Kuva 8. Olotilan sanomaliikenne

Seuraavassa SUBSCRIBE-sanoma, jolla käyttäjä 1111 kirjautuu vastaan ottamaan olotilaviestejä palvelimelta:

```

SUBSCRIBE sip:1111@192.168.10.59 SIP/2.0
Via: SIP/2.0/UDP 193.166.152.166:32122;branch=z9hG4bK-d8754z-

```

d70ed71a7b7d9a4e-1---d8754z-;rport
 Max-Forwards: 70
 Contact: <sip:1111@193.166.152.166:32122>
 To: <sip:1111@192.168.10.59>
 From: "1111"<sip:1111@192.168.10.59>;tag=9d6d657e
 Call-ID: Zjk3ZmY0MTkzYTU3YTNhMmY1OTRmYzZmM3ZmVhZTZiNmQ.
 CSeq: 1 SUBSCRIBE
 Expires: 3600
 Accept: application/watcherinfo+xml
 Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE,
 SUBSCRIBE, INFO
 User-Agent: X-Lite release 11001 stamp 47546
 Event: presence.wininfo
 Content-Length: 0

Seuraavassa käyttäjän 1111 palvelimelle lähetämä PUBLISH-sanoma. Sanomassa ilmoitetaan, että käyttäjän 1111 tila on Busy.

PUBLISH sip:1111@192.168.10.59 SIP/2.0
 Via: SIP/2.0/UDP 193.166.152.166:32122;branch=z9hG4bK-d8754z-72774252d67db87c-1---d8754z-;rport
 Max-Forwards: 70
 Contact: <sip:1111@193.166.152.166:32122>
 To: "1111"<sip:1111@192.168.10.59>
 From: "1111"<sip:1111@192.168.10.59>;tag=ca6d0622
 Call-ID: MjRmNTY4ZjlmYmQwZTBkNGJmYTI3NGI3NjY0MWY5NjI.
 CSeq: 2 PUBLISH
 Expires: 3600
 Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE,
 SUBSCRIBE, INFO
 Content-Type: application/pidf+xml
 User-Agent: X-Lite release 11001 stamp 47546
 Event: presence
 Content-Length: 441

```
<?xml version='1.0' encoding='UTF-8'?>
<presence xmlns='urn:ietf:params:xml:ns:pidf'
  xmlns:dm='urn:ietf:params:xml:ns:pidf:data-model'
  xmlns:rpidd='urn:ietf:params:xml:ns:pidf:rpidd'
  xmlns:c='urn:ietf:params:xml:ns:pidf:cipidd'
  entity='sip:1111@192.168.10.59'>
  <tuple id='tbe67ee07'>
  <status><basic>open</basic></status>
  </tuple>
  <dm:person id='tbe67ee07'>
  <rpidd:activities>
  <rpidd:busy/>
  </rpidd:activities>
  <dm:note>Busy</dm:note>
  </dm:person>
  </presence>
```

Palvelimen lähetämä NOTIFY-sanoma käyttäjälle 2222, jossa kerrotaan että käyttäjän 1111 tila on Busy.

NOTIFY sip:2222@192.168.10.50:41806 SIP/2.0

Via: SIP/2.0/UDP 192.168.10.59;branch=z9hG4bK502d.5cdd21c2.0
To: <sip:2222@192.168.10.59>;tag=2d17eb67
From: <sip:1111@192.168.10.59>;tag=155c340f586c28d0300cf5a6ccf90d99-
b2f1
CSeq: 2 NOTIFY
Call-ID: ZDA4NmUwZmY4YTNiYTAzOWZiOGZlZmQ3NjJjMzc4OWU.
Content-Length: 495
User-Agent: OpenSIPS (1.5.1-notls (i386/linux))
Max-Forwards: 70
Event: presence
Contact: <sip:192.168.10.59:5060>
Subscription-State: active;expires=3524
Content-Type: application/pidf+xml

```
<?xml version="1.0" encoding="UTF-8"?>
<presence xmlns="urn:ietf:params:xml:ns:pidf"
  xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
  xmlns:rpidd="urn:ietf:params:xml:ns:pidf:rpidd"
  xmlns:c="urn:ietf:params:xml:ns:pidf:cipidd"
  entity="sip:1111@192.168.10.59">
  <tuple id="tbe67ee07">
    <status>
      <basic>open</basic>
    </status>
  </tuple>
  <dm:person id="tbe67ee07">
    <rpidd:activities>
      <rpidd:busy/>
    </rpidd:activities>
    <dm:note>Busy</dm:note>
  </dm:person>
</presence>
```

5 VERTAILU OPENSIPS:IN JA ASTERISK:IN VÄLILLÄ

Tässä osiossa vertaillaan Asterisk:in ja OpenSIPS:in muutamia ominaisuuksia kun käytetään SIP-protokollaa. OpenSIPS ja Asterisk ovat tarkoitettu välittämään VoIP-puheluita.

Asterisk on keskittynyt enemmänkin tarjoamaan puhelinvaihteen palveluita ja toimimaan yhdyskäytävänä perinteiseen puhelinverkkoon. Asterisk tukee SIP:in lisäksi myös muita VoIP-protokollia: IAX, H.323, MGCP ja SCCP. /9/

OpenSIPS on keskittynyt olemaan vain SIP-verkon palvelin, mutta sen avulla voidaan ohjata puhelut erilliseen yhdyskäytävään tarpeen vaatiessa. OpenSIP:llä voidaan myös toteuttaa puhelinvaihteen palveluita ohjaamalla puhelut sellaisille palvelimille, joissa niitä tarjotaan. /6/

Molempien asetukset ovat tekstipohjaisissa tiedostoissa. OpenSIPS:issä asetuksilla määritellään pääasiassa miten se käsittelee sanomia, Asterisk:n asetuksilla pääasiassa säädetään soittosuunnitelmia. Molempien asetusten säätöön on myös saatavilla graafinen käyttöliittymä. Kummassakin pääosa asetuksista voidaan asettaa graafisen käyttöliittymän kautta. /9/,

IP-osoitteiden vähyden vuoksi osa käyttäjistä on NAT:in takana. Tällöin käyttäjäagentit tarvitsevat jonkin yhdyskäytävän välittämään medialiikenteen verkon ylitse. Asterisk:issä on ominaisuudet valmiina tätä varten. OpenSIPS:issä ei ole vastaavaa toimintoa itsessään vaan siihen löytyy moduuli, jonka avulla medialiikenne voidaan siirtää ulkopuolisen ohjelman avulla.

Vaikka Asterisk voi toimia käyttäjäagentille välityspalvelimena, siinä ei ole kaikkia välityspalvelimen toimintoja. Se osaa reitittää SIP-sanomat toiselle Asterisk-palvelimelle, mutta muissa tapauksissa sen täytyy käyttää yhdyskäytävää. Esimerkiksi OpenSIPS voi toimia tällaisessa tapauksessa yhdyskäytävänä, sillä OpenSIPS sisältää toiminnot sanomien oikean määränpään etsimiseksi.

6 YHTEENVETO

Tämän opinnäytetyön tarkoituksena oli tutkia avoimenlähdekoodin OpenSIPS -toteutusta, joka toimii SIP-palvelinohjelmistona, joka soveltuu käytettäväksi moneen eri rooliin SIP-verkossa. Työn aluksi tutustuttiin SIP-protokollaan ja sen teoriaan, mikä laajensi käsitystä siitä. Kurssien jälkeen protokollasta muotoutunut käsitys protokollasta syveni. Sen jälkeen aloitettiin perehtyminen OpenSIPS:iin tutustumalla sen dokumentaatioon.

Tutustumisen jälkeen toteutettiin OpenSIPS:llä yksinkertainen SIP-palvelin, joka toimi välitys-, rekisteröinti- ja olotilapalvelimenä. Asentamisen ja asetuksen asettamisen jälkeen todettiin testaamalla OpenSIPS-palvelinohjelmiston toimivan kuten odotettiin. Kaappausten perusteella OpenSIPS toimii kuten SIP-palvelimen kuuluu toimia tulosten luotettavuuden osalta arvioituna. Selkeimmät tulokset tässä tutkielmassa ovat kaappaukset, muita tutkimusmenetelmiä tai tuotoksia ei tässä tutkielmassa käytetty.

OpenSIPS on kokonaisuudessaan toimiva, mutta siihen tutustuminen vei jonkin verran aikaa. Ensimmäisestä tutkimusvaiheesta viimeiseen asti vaiheet olivat kuitenkin selkeitä osuuksia ja itseopiskelun avulla eteneviä tehtäviä. Esimerkiksi OpenSIPS:llä toteutetut toiminnollisuudet (välitys-, rekisteröinti- ja olotilapalvelin) testattiin yhdellä kertaa.

Selkeästi esille tulevat, lähinnä ainoat OpenSIPS:iin rinnastettavissa olevat vaihtoehdot ovat jääneet liiankin läheisiksi perustavan laatukselle OpenSIPS:lle. Ulkopuoliseen toimintaan nähden niissä ei olekaan eroja, vaan erot jäävät sisäisiksi toimintaeroiksi. Vertailua tässä tutkimuksessa tehtiinkin siitä, miten Asterisk toimii samoissa tehtävissä.

OpenSIPS täyttää SIP-palvelimelle standardissa asetetut vaatimukset. Sen avulla voidaan toteuttaa laboratoriossa eri SIP-palvelintoiminnallisuuksia

LÄHTEET

- /1/ http://en.wikipedia.org/wiki/Session_Initiation_Protocol [Viitattu 18.5.2009]
- /2/ RFC 3261. SIP: Session Initiation Protocol. Rosenberg, J. Schulzrinne, H. Camarillo, G. Johnston, A. Peterson, J. Sparks, R. Handley, M. Schooler, E. Kesäkuu 2002
- /3/ Johnston, A. SIP: Understanding the Session Initiation Protocol (Second Edition) Artech House, Incorporated, 2003. 307 p.
- /4/ Henry Sinnreich, Alan B. Johnston, Internet Communications Using SIP: Delivering VoIP and Multimedia Services with Session Initiation Protocol, 2nd Edition. Wiley, 2006. 408 p.
- /5/ OpenSIPS <http://www.opensips.org/Main/HomePage> [Viitattu 11.3.2009]
- /6/ openSIPS | Resources / Features
<http://www.opensips.org/Resources/Features> [Viitattu 11.3.2009]
- /7/ <http://opensips.svn.sourceforge.net/viewvc/opensips/trunk/INSTALL>
[Viitattu 11.3.2009]
- /8/ <http://www.iana.org/assignments/sip-parameters> [Viitattu 28.4.2009]
- /9/ Jim Van Meggelen, Leif Madsen, Jared Smith Asterisk: The Future of Telephony. 2th ed. Sebastopol: O'Reilly Media, Inc Elokuu 2007. 604 p.
- /10/ <http://www.opensips.org/Resources/DocsModules15> [Viitattu 16.5.2009]