



УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У
НОВОМ САДУ



Ана Чадовски

ИМПЛЕМЕНТАЦИЈА ХМРР BOSH ПРОТОКОЛА У ОКВИРУ TR-069 КЛИЈЕНТА

МАСТЕР РАД

Нови Сад, 2023



УНИВЕРЗИТЕТ У НОВОМ САДУ • ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА
21000 НОВИ САД, Трг Доситеја Обрадовића 6

КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, РБР:		
Идентификациони број, ИБР:		
Тип документације, ТД:	Монографска документација	
Тип записа, ТЗ:	Текстуални штампани материјал	
Врста рада, ВР:	Дипломски – мастер рад	
Аутор, АУ:	Ана Теодоровић	
Ментор, МН:	проф. др Илија Башичевић	
Наслов рада, НР:	Имплементација XMPP BOSH протокола у оквиру TR-069 клијента	
Језик публикације, ЈП:	Српски / латиница	
Језик извода, ЈИ:	Српски	
Земља публикавања, ЗП:	Република Србија	
Уже географско подручје, УГП:	Војводина	
Година, ГО:	2023	
Издавач, ИЗ:	Ауторски репринт	
Место и адреса, МА:	Нови Сад; трг Доситеја Обрадовића 6	
Физички опис рада, ФО: (поглавља/страна/ цитата/табела/слика/графика/прилога)		
Научна област, НО:	Електротехника и рачунарство	
Научна дисциплина, НД:	Рачунарска техника	
Предметна одредница/Кључне речи, ПО:	Управљање уређајима у Интернету, TR-069, XMPP, BOSH, CPE, ACS	
УДК		
Чува се, ЧУ:	У библиотеци Факултета техничких наука, Нови Сад	
Важна напомена, ВН:		
Извод, ИЗ:	У овом раду је представљено решење за унапређење комуникације између TR-069 клијента и XMPP послужитеља. У оквиру TR-069 клијента додат је XMPP BOSH клијентски модул који омогућава комуникацију коришћењем XMPP протокола преко HTTP порука	
Датум прихватања теме, ДП:		
Датум одбране, ДО:		
Чланови комисије, КО:	Председник:	
	Члан:	
	Члан, ментор:	
		Потпис ментора



UNIVERSITY OF NOVI SAD • FACULTY OF TECHNICAL SCIENCES
21000 NOVI SAD, Trg Dositeja Obradovića 6

KEY WORDS DOCUMENTATION

Accession number, ANO :	
Identification number, INO :	
Document type, DT :	Monographic publication
Type of record, TR :	Textual printed material
Contents code, CC :	Master Thesis
Author, AU :	Ana Teodorović
Mentor, MN :	Ilija Bašičević, PhD
Title, TI :	Integration of XMPP BOSH protocol in the TR-069 client
Language of text, LT :	Serbian
Language of abstract, LA :	Serbian
Country of publication, CP :	Republic of Serbia
Locality of publication, LP :	Vojvodina
Publication year, PY :	2023
Publisher, PB :	Author's reprint
Publication place, PP :	Novi Sad, Dositeja Obradovica sq. 6
Physical description, PD : (chapters/pages/ref./tables/pictures/graphs/appendixes)	
Scientific field, SF :	Electrical Engineering
Scientific discipline, SD :	Computer Engineering, Engineering of Computer Based Systems
Subject/Key words, S/KW :	Device management in Internet, TR-069, XMPP, BOSH, CPE, ACS
UC	
Holding data, HD :	The Library of Faculty of Technical Sciences, Novi Sad, Serbia
Note, N :	
Abstract, AB :	This paper presents a solution for improving communication between TR-069 client and XMPP server. The XMPP BOSH module has been added to TR-069 client, which enables communication via XMPP protocol using HTTP messages
Accepted by the Scientific Board on, ASB :	
Defended on, DE :	
Defended Board, DB :	President:
	Member:
	Member, Mentor:
	Mentor's sign

Захвалност

Захваљујем се ментору проф. др Илији Башичевићу на стручној помоћи током израде овог рада.

Посебно се захваљујем Душану Живков на стручној помоћи, саветима и утрошеном времену и стрпљењу током израде рада.

САДРЖАЈ

1. Увод.....	1
2. Теоријске основе.....	2
2.1 TR-069	2
2.1.1 CWMP параметри	3
2.2 XMPP.....	4
2.2.1 Адресирање	5
2.3 BOSH	5
2.4 TR-069 и XMPP	6
3. Концепт решења.....	9
3.1 XMPP параметри	9
3.2 Захтев за повезивање	10
4. Програмско решење.....	12
5. Испитивање и верификација.....	17
5.1 Испитивање параметара	17
5.2 Испитивање конекције.....	18
5.3 Gajim.....	18
6. Закључак	20
7. Literatura.....	21

СПИСАК СЛИКА

Слика 2.1 Комуникација између ACS-а и CPE-а без XMPP послужитеља

Слика 2.2 Комуникација између ACS-а и CPE-а са XMPP послужитељем

Слика 2.3 Веза између XMPP послужитеља и TR069 клијента

Слика 3.1 Успостављање конекције са XMPP послужитељем

Слика 3.2 Дијаграм BOSH комуникације

Слика 4.1 Датотека за чување података о параметрима “tr-181-short.xml”

Слика 1.4 Креирање конфигурације

Слика 1.5 Креирање ConnectionListener-а

Слика 5.1 GenieACS

Слика 5.2 Испис са уређаја

Слика 5.3 Gajim

СПИСАК ТАБЕЛА

Табела 5.1 Списак тестова за испитивање параметара

Табела 5.2 Списак тестова за испитивање конекције

Табела 5.3 Списак тестова за испитивање размене порука

СКРАЋЕНИЦЕ

TR-069	- Technical Report 069, Техничка спецификација 069
CWMP	- CPE WAN Management Protocol
CPE	- Customer Premises Equipment, крајњи корисник
ACS	- AutoConfiguration Server, ауто-конфигурацијски послужитељ
XMPP	- Extensible Messaging and Presence Protocol, комуникацијски протокол
XML	- Extensible Markup Language, скуп правила за кодирање докумената
SASL	- Simple Authentication and Security Layer, оквир за аутентификацију и сигурност података у интернет протоколима
TLS	- Transport Layer Security, криптографски протокол који обезбеђује комплетну комуникацију сигурности мреже
TCP	- Transmission Control Protocol, комуникацијски протокол
BOSH	- Bidirectional-streams Over Synchronous HTTP, технологија за комуникацију
HTTP	- Hypertext Transfer Protocol, мрежни протокол за пренос информација

1. Увод

Послужитељи (eng. Servers) уређаја потрошачке електронике су системи засновани на рачунару, чији је основни задатак комуникација са уређајима са једне стране и манипулација подацима са друге. У овом раду описан је проблем, концепт и реализација XMPP комуникационог протокола и једне од његових најзначајнијих технологија – BOSH. Обухваћене су теме које су повезане са темом рада, као и основни појмови везани за TR-069 протокол, који обезбеђује постојање TR-069 послужитеља и TR-069 клијента који се налази на крајњем корисничком уређају.

2. Теоријске основе

2.1 TR-069

TR-069 (Technical Report 069) је техничка спецификација која дефинише комуникациони механизам за даљинско управљање крајњим корисницима[1]. Описује CPE WAN Management Protocol (CWMP), који је намењен за комуникацију између крајњег корисника (CPE, eng. Customer Premises Equipment) и аутоматско-конфигурацијског послужитеља (eng. ACS – AutoConfiguration Server). Гледано са стране TR-069, ACS је мрежни уређај и послужитељ који аутоматизује извршавање основних метода управљања над корисничким уређајима. Широка примена TR-069 протокола омогућила је да CPE уређаји могу бити различитог типа, при томе не постоји разлика у протоколу по коме се преносе подаци до ACS-а, него се разликују само модели података за сваки тип CPE-а.

Протокол подржава разне функционалности, од који су најважније:

- аутоматско конфигурисање и динамичко реконфигурација услуга,
- управљање софтверским модулом,
- праћење статуса и перформанси CPE и њихова дијагностика.

Најзначајнија функционалност је аутоматско конфигурисање и динамичка реконфигурација услуга. Спецификација обезбеђује могућност примарне и накнадне конфигурације у случају нових захтева. Управљање софтверским модулом омогућава инсталацију, деинсталацију и ажурирање софтверских модула, као и праћење њиховог стања.

TR-069 такође дефинише скуп механизма који омогућавају да CPE самостално обавештава ACS о променама у његовом стању.

Протокол има за циљ да обезбеди следеће:

- Дозволи успостављање иницијалне конекције између ACS и CPE, избегавајући потребу за одржавањем сталне везе између сваког CPE и ACS
- Функционалну комуникацију између ACS и CPE без обзира на то која страна је покренула успостављање конекције
- Дозволи једном или више ACS-ова да опслужују групу CPE-ова који могу бити повезани на једог или више провајдера

Протокол има за циљ да минимизира сложеност имплементације, истовремено пружајући флексибилност. Такође, дизајниран је тако да буде проширив, тачније да има подршку будућим проширењима стандарда.

2.1.1 CWMP параметри

CWMP параметри су претстављени у форми модела података (eng. data model). Улога параметара је да обезбеде ACS-у информације о карактеристикама и статусу CPE, као и да управљају њиховом конфигурацијом. Параметри могу бити специфицирани само за читање или за читање и упис.

Параметри имају хијерархијску структуру. Структура је претстављена као објекат. Сваки објекат садржи један или више параметара. Сваки CPE садржи један главни параметар који у зависности од типа уређаја добија вредност Device или Internet Gateway Device.

Уз помоћ параметара се могу променити подешавања са стране ACS и то на следећи начин:

- Могућност уписа вредности параметара
- Ограничење вредности параметара, дефинисање енумерација или граничних вредности параметара
- Обавеза слања вредности параметара при иницијализацији TR-069 сесије
- Надгледање вредности параметара у реалном времену
- Постављање почетне подразумеване вредности параметара
- Ограничавање броја вишеструких објеката
- Дефинисање модела података

Типови података дефинисани стандардом су:

- Низ карактера (eng. String)

- Целобројни (eng. Integer)
- Проширени целобројни (eng. Long)
- Неозначени целобројни (eng. Unsigned Integer)
- Логички (eng. Boolean)
- Временски (eng. dateTime)
- Бинарни (eng. Base64)
- Хексадецимални (eng. hexBinary)

2.2 XMPP

XMPP (Extensible Messaging and Presence Protocol) је протокол за комуникацију који се користи за потребе размене тренутних порука, звучне и видео позиве, информације о доступности других корисника, одржавање листе контаката, базиран на XML (eng. Extensible Markup Language) стандарду [2]. Првобитно се називао "Jabber", креиран од стране Jabber open-source заједнице, намењен за размену тренутних порука. Предност овог протокола у односу на остале овог типа су то што је :

- Отворен – протокол је јаван, бесплатан и лако разумљив
- Стандардизован – Internet Engineering Task Force (IETF) је одобрио основу XMPP протокола за размену тренутних порука
- Проверен – прве XMPP технологије развијене су још 1998. године и до сада су прилично стабилне
- Децентрализован – архитектура је слична архитектури електронске поште, тачније свако може да покрене сопствени XMPP сервер и да омогући појединцима и организацијама да преузму контролу над својим комуникацијским искуством
- Безбедан – може бити изолован од јавне мреже и има уграђене одређене заштитне протоколе у језгро протокола
- Прилагодљив – свако може да изгради потребну функционалност на основу језгра протокола
- Флексибилан – постоје разне функционалности које се могу надоградити

Најважније XMPP технологије су:

- Core

- Jingle
- Multi-User Chat
- PubSub
- BOSH

Core или основа протокола се користи за пренос (eng. streaming) XML-а. Основе укључују: основни слој XML преноса, енкрипцију канала коришћењем Transport Layer Security (TLS), препознавање преко Simple Authentication and Security Layer (SASL), коришћење UTF-8, уграђене информације о присутности (eng. presence), листа контаката са омогућеном информацијом о присутности.

Jingle обезбеђује клијетима могућност подешавања, управљања и прекида мултимедијалне сесије. Има широк спектар примене: гласовно ћаскање (eng. voice chat), видео ћаскање (eng. video chat), пренос фајлова.

Multi-User-Chat (MUC) је екстензија XMPP протокола која омогућава комуникацију више корисника у оквиру собе или канала. Поред стандардних функционалности као што су теме разговора или позив за приступ соби, постоје и опције избацивања и забране корисника, именовање администратора и модератора, захтев за чланство...

PubSub (publish-subscribe) је проширење које се користи за разне апликације које захтевају технологију “објаве и претплате”, тачније, особа или апликације објављују информације, а обавештење о томе се шаље претплатницима.

BOSH (Bidirectional-streams Over Synchronous HTTP) је технологија за двосмерну комуникацију преко Hypertext Transfer Protocol (HTTP). Углавном се користи за размену између Jabber/XMPP клијената и сервера (нпр. веза са web клијентима, клијентима у покрету који имају мрежу са прекидима).

2.2.1 Адресирање

Сви корисници у мрежи имају јединствену XMPP адресу која се назива JID. Структура JID подсећа на адресу електронске поште. Садржи корисничко име (eng. username), назив домена или IP адресу послужитеља који су раздвојени знаком “@”. Такође, један корисник се може пријавити са више локација (кућа, посао..), тако што ће навести ресурс (eng. resource), који ће бити одвојен косом цртом. На пример, пун назив JID налога би био: “username@domain/resource”.

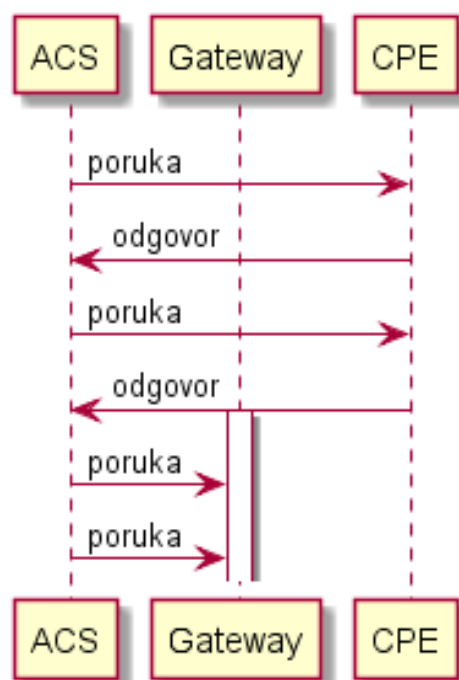
2.3 BOSH

Најчешће коришћен начин преноса за XMPP је дугорочна TCP (Transmission Control Protocol) веза, која у неким случајевима није могућа због природе крајњих уређаја или мреже. Алтернатива овоме је BOSH (eng. Bidirectional-streams Over Synchronous HTTP), који размењује захтеве и одговоре преко краткотрајних веза. BOSH је технологија за

двосмерну комуникацију преко Hypertext Transfer Protocol (HTTP). Дизајниран је тако да преноси податке са минималним кашњењем у оба смера. Техника коју користи BOSH назива се “HTTP дуго анкетирање” (eng. HTTP long polling) која смањује кашњење и потрошњу пропусног опсега у односу на друге технике HTTP анкетирања [3]. Када клијент пошаље захтев, управљач не одговори одмах, него држи захтев отвореним све док не буде имао податке које треба да пошаље или истекне време неактивности. Клијент тада шаље нови захтев и настаља комуникацију. Уколико клијент има нови захтев, док је претходни још увек отворен, он шаље нови захтев, управљач на претходни одговара углавном празном поруком, којом затвара претходну петљу и отвара се нова петља која служи за комуникацију, док стара чека нови захтев.

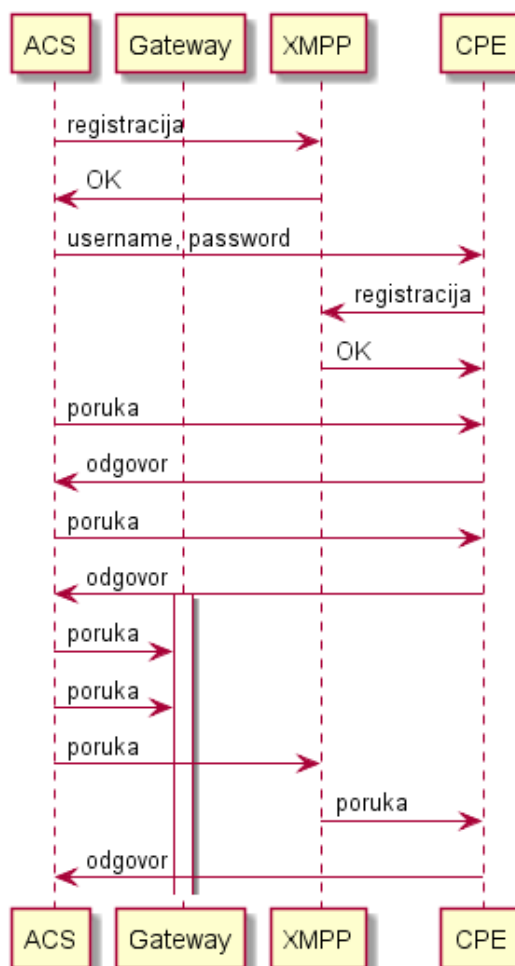
Ова технологија је значајна за комуникацију са web клијентима. Иако је углавном тип поруке који се преноси XML, BOSH технологија може да преноси мешавину елемената који су дефинисани различитим протоколима (XMPP i JSON).

2.4 TR-069 и XMPP



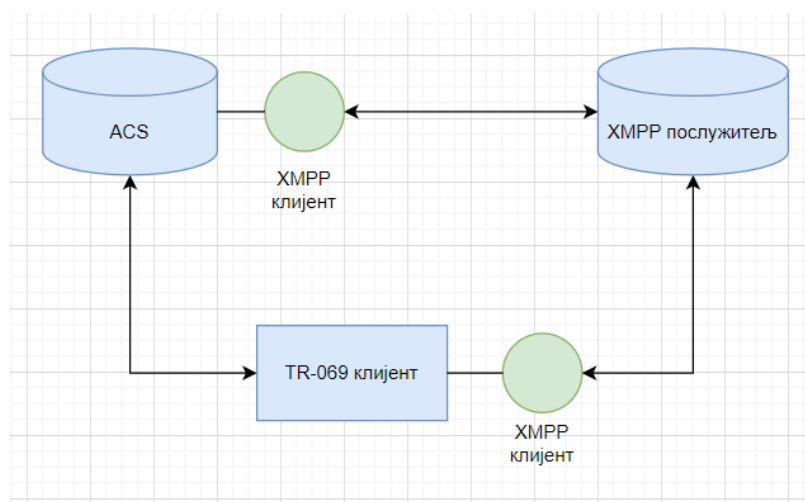
Слика 2.1 Комуникација између ACS-а и CPE-а без XMPP послужитеља

У одређеним случајевима, ACS је спречен да пошаље поруку CPE-у директно због постојања мрежне баријере или конвертора протокола (firewall или gateway).



Слика 2.2 Комуникација између ACS-а и CPE-а са XMPP послужитељем

Тај проблем решава XMPP проткол, као што је приказано на слици. И ACS и CPE се региструју на XMPP послужитељу, а затим га ACS може користити да пренесе поруку до CPE [4]. Када се деси да gateway спречи прослеђивање поруке од ACS-а ка CPE, ACS шаље поруку XMPP послужитељу који на основу наведеног корисничког имена може да пронађе CPE и проследи поруку. Након примања поруке, CPE се јавља ACS-у.



Слика 2.3 Веза између XMPP послужитеља и TR069 клијента

3. Концепт решења

У претходном поглављу описана је намена и значај XMPP протокола. За имплементацију у оквиру TR-069 клијента неопходно је постојање XMPP послужитеља на коме ће се клијент регистровати. За регистрацију је потребно креирати JID, као што је раније описано. На основу њега корисник постаје препознатљив. Такође, CPE мора бити у могућности да одржава сталну везу са XMPP послужитељем.

3.1 XMPP параметри

ACS би требао да омогући коришћење XMPP-а на CPE-у конфигурисањем XMPP.Connection објекта, ManagementServer.ConnReqXMPPConn параметра и опционо ManagementServer.ConnReqAllowedJabberIDs који садржи листу дозвољених JID-ова.

Вредност ManagementServer.ConnReqXMPPConn параметра мора бити постављена на име путање до XMPP.Connection објекта која се користи за везу. Уколико је објекат избрисан, вредност параметра мора бити постављена на празан низ.

XMPP.Connection претставља XMPP везу између корисника и послужитеља [5]. Може постојати највише један унос за одређени Username, Domain и Resource. На основу ових параметара, формира се JID како је раније описано.

Такође, значајан је објекат XMPP.Connection.Server који описује послужитељ који се користи за везу. Неки од параметара у оквиру овог објекта су: адреса послужитеља (ServerAdress), Port, доступност (Enable)...

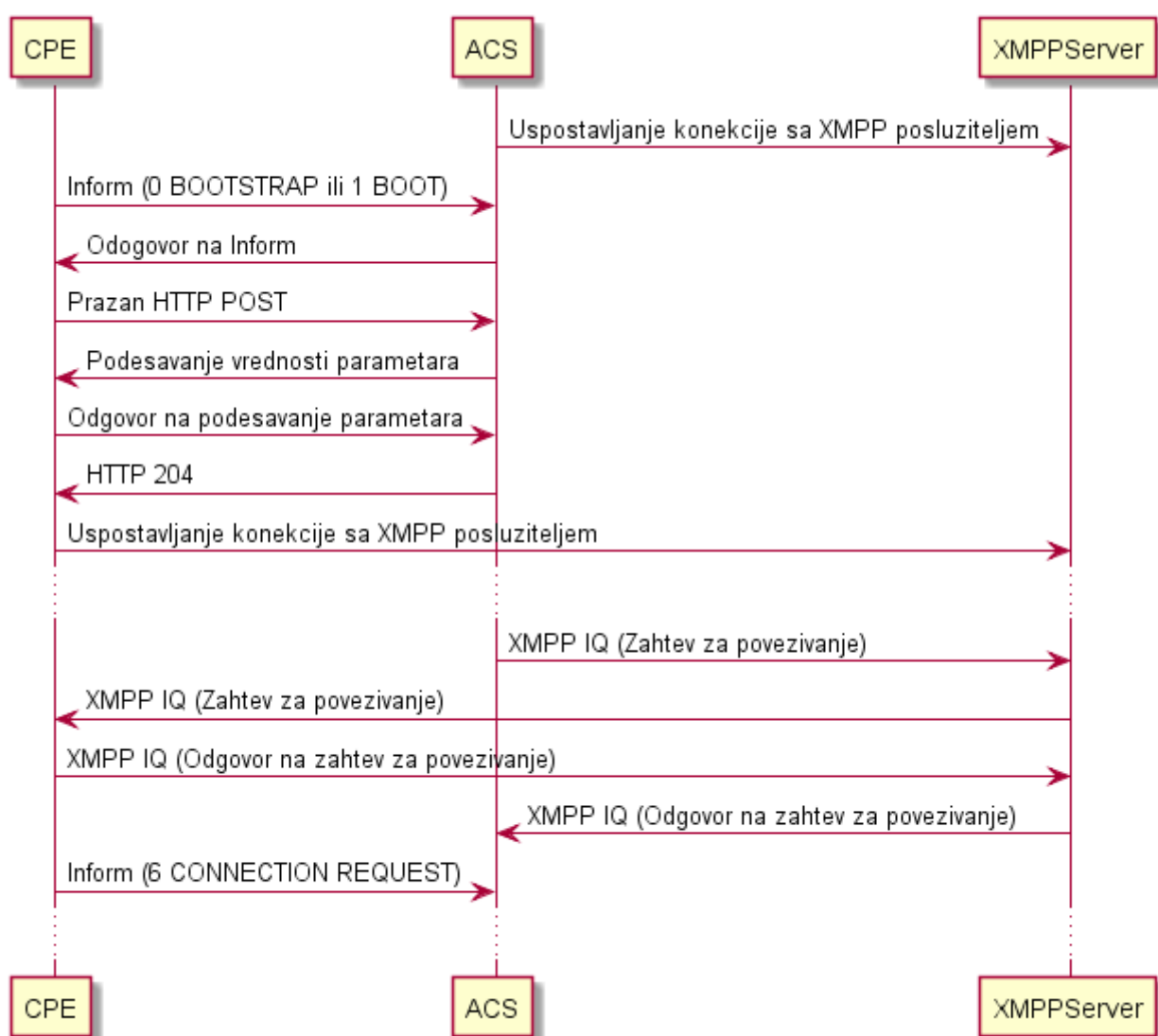
Објекат XMPP.Connection.Stats пружа статистичке информације о вези (број примљених порука, број неуспешно примљених порука, број пренесених порука, број неуспешно пренесених порука).

Корисничко име и лозинка који ће бити коришћени за аутентификацију се добијају из параметара `ManagementServer.ConnectionRequestUsername` и `ManagementServer.ConnectionRequestPassword`.

3.2 Захтев за повезивање

Захтев за повезивање са CPE се обавља на следећи начин:

- ACS успоставља везу са XMPP послужитељем
- ACS омогућава коришћење XMPP-а постављањем параметара
- Уређај успоставља везу са наведеним XMPP послужитељем
- Када ACS пожели да комуницира са уређајем може то урадити преко установљених XMPP порука



Слика 3.1 Успостављање конекције са XMPP послужитељем

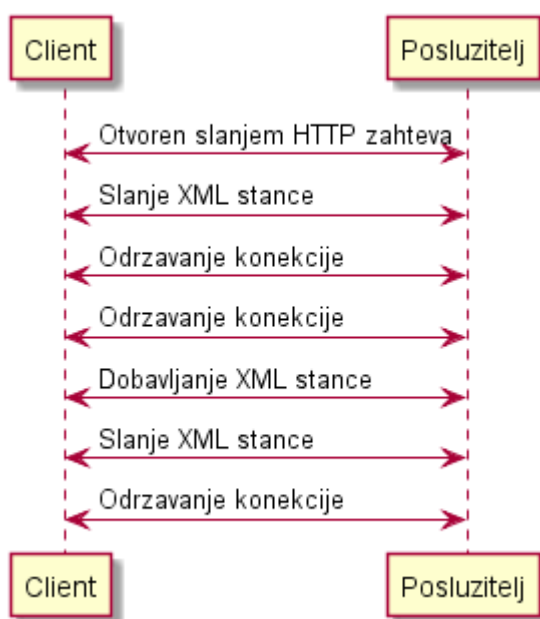
На слици 3.1 је приказано успостављање везе са XMPP послужитељем. ACS на основу постављених параметара шаље захтев за везу XMPP послужитељу. Након

успешног повезивања, ACS преко XMPP послужитеља шаље захтев за повезивање (eng. ConnectionRequest) у виду XMPP IQ станце. У оквиру овог захтева налазе се корисничко име и шифра који су наведени у параметрима

Device.ManagementServer.ConnectionRequestUsername и Device.ManagementServer.ConnectionRequestPassword, а који се односе на корисника. Уколико су наведени исправни подаци, корисник шаље позитиван одговор на захтев. Позитиван одговор од стране CPE је празна IQ станца. У случају неуспешног захтева, CPE шаље IQ станцу која садржи service-unavailable грешку.

ManagmentServer.ConnReqAllowedJabberIDs параметар се поставља опционо, али даје додатну заштиту у виду забране примања порука од корисника чији JID се не налази у листи дозвољених.

Након успоставе везе, комуникација се одвија као на дијаграму са слике 3.2.



Слика 3.2 Дијаграм BOSH комуникације

4. Програмско решење

У овом поглављу ће за почетак бити описан начин додавања параметара у модел подата, као и начин повезивања Native и Java слоја.

Постоји неколико битних фајлова за дефинисање модела података:

- props.txt – садржи све параметре модела података
- all_params.txt – садржи листу параметара са дефинисаним могућностима постављања и добијања параметара (setter, getter and callback)
- good_objects.txt – садржи објекте модела података
- tr-181-2-12-0-cwmp-full.xml – садржи цео стандард модела података са свим могућим стандардним параметрима
- non-standard.xml – садржи нестандартне параметре модела података
- tr-181-short.xml – садржи комбинацију стандардних и нестандартних параметара који се користе

```

<object name="Device.DeviceInfo." access="readOnly" minEntries="1" maxEntries="1" dmr:version="2.0">
  <description>
    This object contains general device information.
  </description>
  <parameter name="Manufacturer" access="readOnly" dmr:version="2.0">
    <description>
      The manufacturer of the CPE (human readable string).
    </description>
    <syntax>
      <string>
        <size maxLength="64" />
      </string>
    </syntax>
  </parameter>
  <parameter name="ManufacturerOUI" access="readOnly" dmr:version="2.0">
    <description>
      Organizationally unique identifier of the device manufacturer. Represented as a six hexadecimal-digit value using
      The value MUST be a valid OUI as defined in {{bibref|OUI}}.
      This value MUST remain fixed over the lifetime of the device, including across firmware updates.
      Any change would indicate that it's a new device and would therefore require a BOOTSTRAP Inform.
    </description>
    <syntax>
      <string>
        <size minLength="6" maxLength="6" />
        <pattern value="[0-9A-F]{6}" />
      </string>
    </syntax>
  </parameter>
  <parameter name="ModelName" access="readOnly" activeNotifv="canDeny" dmr:version="2.0">

```

Слика 4.1 Датотека за чување података о параметрима “tr-181-short.xml”

Када су сви параметри дефинисани, потребно је извршити команду “make”. На тај начин ће се изменити “tr-181-short.xml”. Такође, генерисани модел података ће постојати у креираној излазној датотеци која саджи датотеке са креираним кодовима свих параметара.

Потребно је креирати потребне параметре везане за XMPP комуникацију, који су описани у ранијим поглављима.

Нови параметри ће бити видљиви на ACS-у.

ManagementServer.java

У *init()* функцији се проверава да ли постоји вредност параметра *Device.ManagementServer.ConnReqXMPPConnection*. Уколико постоји, позива се *connectXMPP()*. У оквиру ове функције постављају се вредности за *XMPP.Connection* објекат, као и вредности за корисничко име и лозинку који ће касније бити потребни за аутентификацију. Такође, дефинише се *XMPPClient* и позива његова метода *connect()*.

Уколико је параметар *Device.ManagementServer.ConnReqXMPPConnection* празан, када се унесе његова вредност на ACS-у, биће позвана метода *onSetConnReqXMPPConnection()* у оквиру које се позива *connectXMPP()*. Метода *onSetConnReqXMPPConnection()* је изгенерисана приликом измене параметара тако

што се за параметар *Device.ManagementServer.ConnReqXMPPConnection* у оквиру датотеке “all_params.txt” назначи да је потребан callback.

XMPPClient.java

У оквиру XMPPClient-а налазе се две важне функције - *connect()* и *create()*.

У оквиру функције *connect()* креира се *BOSHConfiguration* конфигурација која је потребна за успостављање комуникације са послужитељем. Конфигурација је неопходна за повезивање, јер се у оквиру ње наводе потребни подаци. У оквиру конфигурације наводи се корисничко име и лозинка, домен послужитеља, ресурс, адреса и порт послужитеља, HTTP заглавље итд.

```
BOSHConfiguration mBoshConfig = BOSHConfiguration.builder()
    .setXmppDomain(mDomain)
    .setUsernameAndPassword(mUsername, mPassword)
    .setHost(mServerAddress)
    .setPort((int)mPort)
    .setFile(mFilePath)
    .setUseHttps(mUseTls)
    .setSecurityMode(SecurityMode.ifpossible)
    .enableDefaultDebugger()
    .addEnabledSaslMechanism("SCRAM-SHA-1")
    .addHttpHeader("User-Agent", TR181Monitor.getInstance().getUserAgent())
    .build();
```

Слика 4.2 Креирање конфигурације

Функција *create()* дефинише *XMPPBOSHConnection* на основу задате конфигурације и додаје му *ConnectionListener* који служи са праћење стања конекције (слика 4.3).

Затим се позива методу *connect()* у оквиру *XMPPBOSHConnection* и на тај начин се повезује са XMPP послужитељем. *XMPPBOSHConnection* се налази у Smack библиотеци.

Уколико је конекција успешна, потребно је послати корисничко име и лозинку како би се извршила регистрација.

```
mBoshConnection.login(mUsername, mPassword);
```

Након извршења, CPE је повезан са XMPP послужитељем и може да прима поруке.

```

XMPPBOSHConnection mBoshConnection = new XMPPBOSHConnection(mBoshConfig);
mBoshConnection.addConnectionListener(new ConnectionListener() {
    @Override
    public void connecting(XMPPConnection connection) {
        Log69.d(TAG, "XMPP reconnecting");
    }

    @Override
    public void connected(XMPPConnection connection) {
        Log69.d(TAG, "XMPP connection Successful");
    }

    @Override
    public void authenticated(XMPPConnection connection, boolean resumed) {
        Log69.d(TAG, "XMPP User authenticated successfully");
    }

    @Override
    public void connectionClosed() {
        Log69.e(TAG, "XMPP connectionClosed value");
    }

    @Override
    public void connectionClosedOnError(Exception e) {
        Log69.e(TAG, "XMPP connectionClosedOnError value"+ e.getMessage());
        create(reference);
    }
}

```

Слика 4.3 Креирање ConnectionListener-a

У оквиру ове функције се такође региструје *IQRequestHandler* који служи за пријем и обраду IQ порука.

```

mBoshConnection.registerIQRequestHandler(new ConnectionRequestHandler(mConnReqUsername,
                                                                    mConnReqPassword, mServerJabberList));

```

ConnectionRequestHandler.java

Обрада захтева за повезивање се обавља у ConnectionRequestHandler.java који имплементира *IQRequestHandler*. У оквиру override-оване методе *handleIQRequest()* се проверава исправност корисничког имена и лозинке унетих у оквиру IQ захтева, као и да ли се JID пошиљаоца налази у листи дозвољених.

У случају неисправних података, шаље се одговор “not_authorized”.

```

StanzaError error = StanzaError.getBuilder()
    .setCondition(StanzaError.Condition.not_authorized)
    .build();
return IQ.createErrorResponse(iqRequest, error);

```

У случају да се JID пошиљаоца не налази у листи дозвољених, шаље се “not_acceptable”.

```
StanzaError error = StanzaError.getBuilder()
    .setCondition(Condition.not_acceptable)
    .build();
return IQ.createErrorResponse(iqRequest, error);
```

Уколико се потврди исправност свих података, креира се одговор на захтев и шаље се.

```
return IQ.createResultIQ(iqRequest);
```

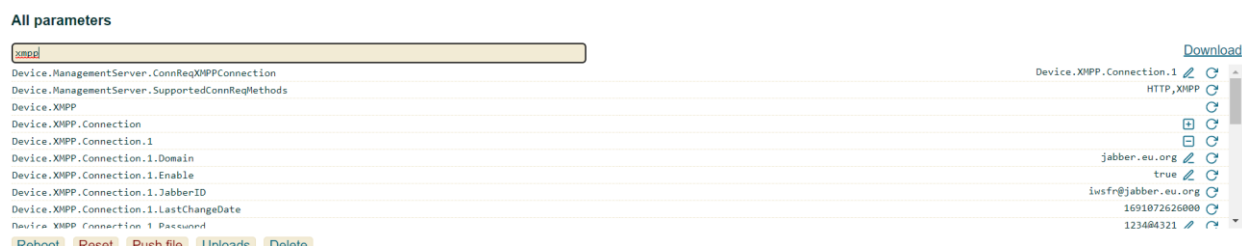
Након тога, ACS и CPE имају успостављену везу и могу да комуницирају.

5. Испитивање и верификација

5.1 Испитивање параметара

Испитивање параметара рађено је ручно упоређивањем параметара из базе уређаја и на послужитељу. Као послужитељ коришћен је GenieACS.

GenieACS је open-source ACS. Окружење GenieACS је приказано на слици 4.



Слика 5.1 GenieACS

Тест	Резултат
Ручна проверка параметара	✓
Провера вредности параметара у бази података	✓
Провера вредности параметара на GenieACS-у	✓

Табела 5.1 Списак тестова за испитивање параметара

5.2 Испитивање конекције

Испитивање успешности конекције на XMPP послужитеља обављено је праћењем исписа са уређаја.

```
09-06 15:02:31.973 16981 17035 D SMACK : RECV (0):
09-06 15:02:31.973 16981 17035 D SMACK : <body xmlns='http://jabber.org/protocol/httpbind'>
09-06 15:02:31.973 16981 17035 D SMACK :   <iq type='result' id='X5Q2D-1' xmlns='jabber:client'>
09-06 15:02:31.973 16981 17035 D SMACK :     <bind xmlns='urn:ietf:params:xml:ns:xmpp-bind'>
09-06 15:02:31.973 16981 17035 D SMACK :       <jid>
09-06 15:02:31.973 16981 17035 D SMACK :         iwsfr@jabber.eu.org/138906072331256835717061922
09-06 15:02:31.973 16981 17035 D SMACK :       </jid>
09-06 15:02:31.973 16981 17035 D SMACK :     </bind>
09-06 15:02:31.973 16981 17035 D SMACK :   </iq>
09-06 15:02:31.973 16981 17035 D SMACK : </body>
09-06 15:02:31.973 16981 17024 D SMACK : User logged (0): iwsfr@jabber.eu.org:-1/138906072331256835717061922
```

Слика 5.2 Испис са уређаја

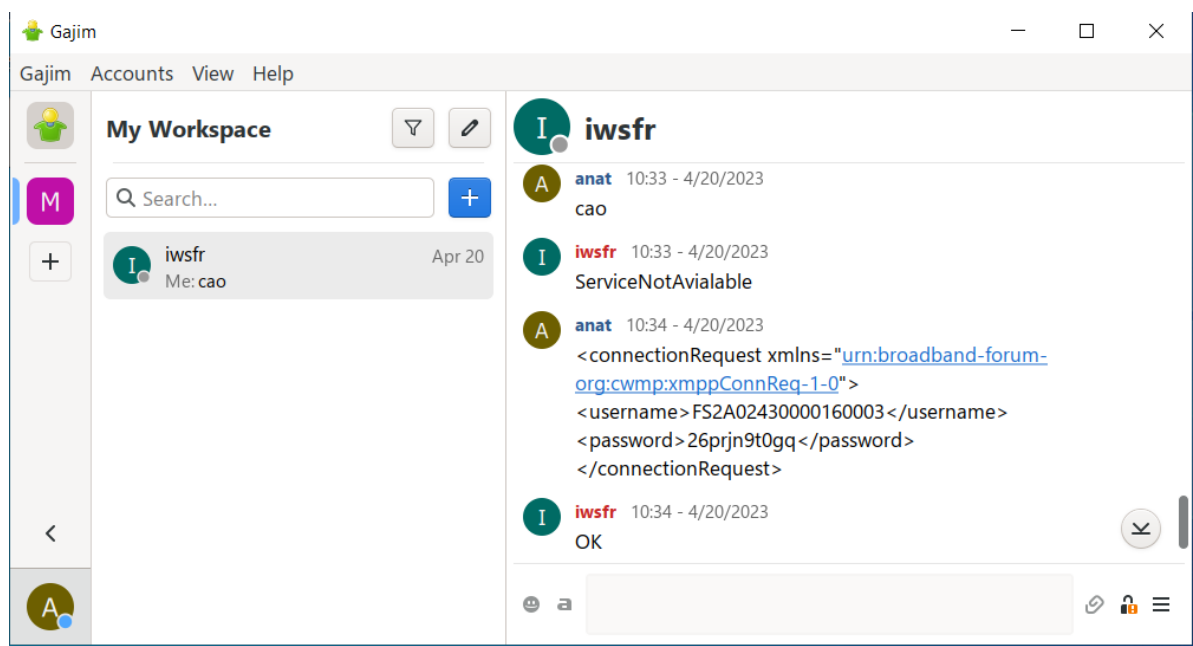
Тест	Резултат
Праћење исписа са уређаја за потврђивање конекције	✓
Праћење исписа са уређаја за потврђивање пријема поруке	✓

Табела 5.2 Списак тестова за испитивање конекције

5.3 Gajim

Gajim је бесплатни, open-source клијент за размену тренутних порука у оквиру XMPP протокола. Тестирање преко Gajim-а је рађено тако што је направљен нови налог са сопственим JID-ом. Налог је додат у листу дозвољених параметара како би био омогућен пријем порука. Послате су поруке са различитим садржајем како би се потврдио пријем и исправан одговор на различит садржај.

На слици 5.3 је приказано Gajim окружење као и комуникација са уређајем. На поруку која не одговара облику захтева за повезивање добија се одговор да послужитељ није доступан, а на поруку са испавним обликом захтева за повезивање, након провере корисничког имена и лозинке добија се одговор да је успешно извршено повезивање.



Слика 5.3 Gajim

Тест	Резултат
Примљена порука са исправним подацима	✓
Успешна аутентификација	✓
Исправан одговор на поруку са исправним подацима	✓
Исправан одговор на поруку са неисправним подацима	✓

Табела 5.3 Списак тестова за испитивање размене порука

6. Закључак

Главни циљ овог рада био је имплементација подршке за XMPP у оквиру TR-069 клијента. Коришћење XMPP BOSH протокола омогућило је комуникацију између ACS-а и CPE-а без сметњи изазваних постојањем мрежних баријера или конвертора протокола. У оквиру CWMP параметара укључени су сви неопходни параметри за реализацију XMPP комуникације. За испитивање решења, као послужитељ, коришћен је Gajim клијент.

Даљи развој решења би био имплементација сопственог послужитеља, што би додатно побољшало тренутно решење.

7. Literatura

- [1] <https://en.wikipedia.org/wiki/TR-069> (приступљено у јулу 2023.)
- [2] An Overview of XMPP, <https://xmpp.org/about/technology-overview/> (приступљено у септембру 2023.)
- [3] XEP-0124: Bidirectional-streams Over Synchronous HTTP (BOSH), XMPP Standards Foundation, 2021, <https://xmpp.org/extensions/xep-0124.html> (приступљено у јулу 2023.)
- [4] I. Basicovic, "An analysis of the TR069 (CWMP) protocol," 2023 46th MIPRO ICT and Electronics Convention (MIPRO), Opatija, Croatia, 2023, pp. 460-465
- [5] TR-069 Device:2.12 Root Object definition, Broadband Forum, 2018, <https://cwmp-data-models.broadband-forum.org/tr-181-2-12-0-cwmp.html> (приступљено у јулу 2023.)