УНИВЕРЗИТЕТ У НОВОМ САДУ

**ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У НОВОМ САДУ**

Милан Туцић

# Протокол за размену порука у системима паметних кућа

МАСТЕР РАД

Нови Сад, 2016

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**: | | **Милан Туцић** | |
| Ментор, **МН**: | | **Доц. др Иштван Пап** | |
| Наслов рада, **НР**: | | **Протокол за размену порука у системима паметних кућа** | |
| Језик публикације, **ЈП**: | | Српски / латиница | |
| Језик извода, **ЈИ**: | | Српски | |
| Земља публиковања, **ЗП**: | | Република Србија | |
| Уже географско подручје, **УГП**: | | Војводина | |
| Година, **ГО**: | | **2016** | |
| 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) | | **8/49/13/20/19/0/0** | |
| Научна област, **НО**: | | Електротехника и рачунарство | |
| Научна дисциплина, **НД**: | | Рачунарска техника | |
| Предметна одредница/Кqучне речи, **ПО**: | | **ИоТ, M2M, МКТТ, паметне куће, протокол, брокер, клијент** | |
| **УДК** | | | |
| Чува се, **ЧУ**: | | У библиотеци Факултета техничких наука, Нови Сад | |
| Важна напомена, **ВН**: | | | |
| Извод, **ИЗ**: | | **Један од јавно доступних протокола за комуникацију између уређаја (енг. МКТТ) је изабран као основа за дефиницију протокола за размену порука између клијената у системима паметних кућа. На асинхрону природу одабраног протокола је додата могућност за размену порука по типу захтев-одговор. Такође, на постојећу реализацију МКТТ послужиоца су додата проширења за аутентификацију, регистрацију нових корисника, ауторизацију и формирање мреже. Тиме је добијено универзално решење за размену порука у посматраним системима. Особине протокола и реализације су приказане кроз неколико различитих мерења, у складу са очекиваном употребом.** | |
| Датум прихватања теме, **ДП**: | | **16.06.2016.** | |
| Датум одбране, **ДО**: | | 04.07.2016. | |
| Чланови комисије, **КО**: | Председник: | **Доц. др Милан Бјелица** | |
| | Члан: | **Доц. др Иван Мезеи** | Потпис ментора |
| | Члан, ментор: | **Доц. др Иштван Пап** | |

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**: | | **Milan Tucic** | |
| Mentor, **MN**: | | **PhD Istvan Pap** | |
| Title, **TI**: | | **Messaging protocol for smart homes** | |
| 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**: | | **2016** | |
| Publisher, **PB**: | | Author's reprint | |
| Publication place, **PP**: | | Novi Sad, Dositeja Obradovica sq. 6 | |
| Physical description, **PD**: (chapters/pages/ref./tables/pictures/graphs/appendixes) | | **8/49/13/20/19/0/0** | |
| Scientific field, **SF**: | | Electrical Engineering | |
| Scientific discipline, **SD**: | | Computer Engineering, Engineering of Computer Based Systems | |
| Subject/Key words, **S/KW**: | | **IoT, M2M, MQTT, smart home, protocol, broker, users** | |
| **UC** | | | |
| Holding data, **HD**: | | The Library of Faculty of Technical Sciences, Novi Sad, Serbia | |
| Note, **N**: | | | |
| Abstract, **AB**: | | **Open source protocol for communication between devices (M2M) is chosen as a foundation for the definition of a messaging protocol for users in the smart home systems. MQTT's asynchronous communication is extended with support for request-response messaging model. Also, an existing broker implementation is upgraded with features like authentication, registration of new users, authorization and network configuration. All this resulted with a universal and generic solution for the message exchange in the smart home systems. Protocol performances are shown through a number of different measurements, in accordance with the expected usage.** | |
| Accepted by the Scientific Board on, **ASB**: | | **16.06.2016** | |
| Defended on, **DE**: | | 04.07.2016 | |
| Defended Board, **DB**: | President: | **PhD Milan Bjelica** | |
| | Member: | **PhD Ivan Mezei** | Menthor's sign |
| | Member, Mentor: | **PhD Istvan Pap** | |

# Zahvalnost

Hvala svima od kojih sam naučio lekcije, jer mi puno znače i njima se vodim. Hvala porodici, učiteljima, nastavnicima, asistentima i profesorima, za sve pruženo u toku školovanja. Hvala Romanu i Ištvanu na podršci i zalaganju tokom rada u timu.

# SADRŽAJ

# SPISAK SLIKA

# SPISAK TABELA

# SKRAĆENICE

**XML** - *Extensible Markup Language*, Jezik za opis podataka

**HTML** - *HyperText Markup Language*, Jezik za opis veb stranice

**CoAP** - *Constrained Application Protocol*, Komunikacioni protokol namenjen upotrebi na malim uređajima

**XMPP** - *Extensible Messaging and Presence Protocol*, Komunikaciona protokol i platforma za razmenu poruka baziran na XML-u

**AMQP** - *Advanced Message Queuing Protocol*, Komunikacioni protokol za razmenu poruka korišćenjem redova i ruta

**MQTT** - *Message Queueing Telemetry Transport*, Komunikacioni protokol za asinhronu razmenu poruka

**TCP** - *Transmission Control Protocol*, Protokol transportnog nivou iz paketa internet protokola, sa garantovanom isporukom paketa

**QoS** - *Quality of service*, Pokazatelji kvaliteta usluge u računarskim mrežama

**JSON** - *JavaScript Object Notation*, Standard za tekstualni opis podataka

**EPL** - *Eclipse Public License*, Licenca za upotrebu otvorenog koda definisana od strane Eclipse fondacije

**EDL** - *Eclipse Distribution License*, Licenca za upotrebu otvorenog koda definisana od strane Eclipse fondacije

**AES** - *Advanced Encryption Standard*, Standard za šifrovanje podataka

**SHA1** - *Secure Hash Algorithm 1*, Algoritam za zaštitu podataka

**ECB** - *Electronic Codebook*, Algoritam za blokovsko šifrovanje podataka

**PKCS** - *Public Key Cryptography Standards*, Standardi za šifrovanje bazirano na javnom ključu

**Base64** - *Base64*, Algoritam za prebacivanje binarnog podatka u tekstualni i obrnuto

**TLS** - *Transport Layer Security*, Protokol za šifrovanje podataka

**SSL** - *Secure Sockets Layer*, Protokol za šifrovanje podataka nastao unapređenjem TLS protokola

**RAM** - *Random-access memory*, Memorija sa nasumičnim pristupom

**PC** - *Personal Compute*, Lični računar

**IPC** - *Inter-process communication*, Međuprocesna komunikacija

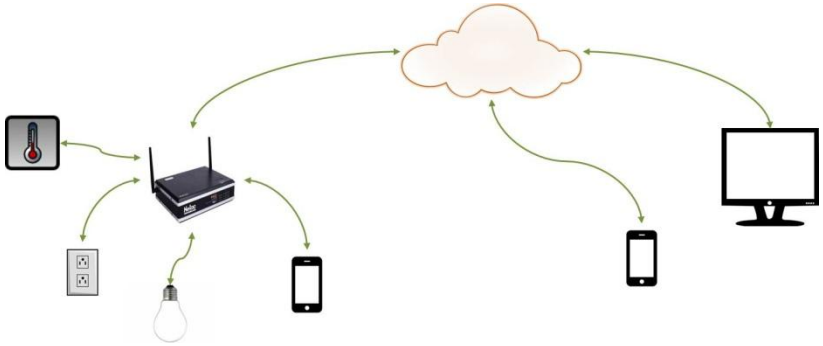
# 1. Uvod

Arhitektura kompleksnih i distribuiranih programskih rešenja zahteva efikasan način komunikacije između delova sistema. Kako takvi sistemi uključuju različite, u osnovnom obliku nepovezane komponente, potrebno je za određeni tip sistema i njegove komponente definisati način sprezanja i protokole razmene podataka.

Sa svojim skupom tehnologija i zahteva, sistemi za pametne kuće donose potrebe za novim realizacijama programskih rešenja u svetu računarstva. Uvode se potrebe za interakcijom koja ranije nije bila prisutna, a sa konstantnim razvojem i primenama koje tek treba da budu definisane, jedna od ključnih osobina sistema za pametne kuće je **laka proširivost**. Trenutno zahtevane funkcionalnosti će, gotovo sigurno, u narednom periodu biti proširene, ako ne i zamenjene nekim potpuno novim.

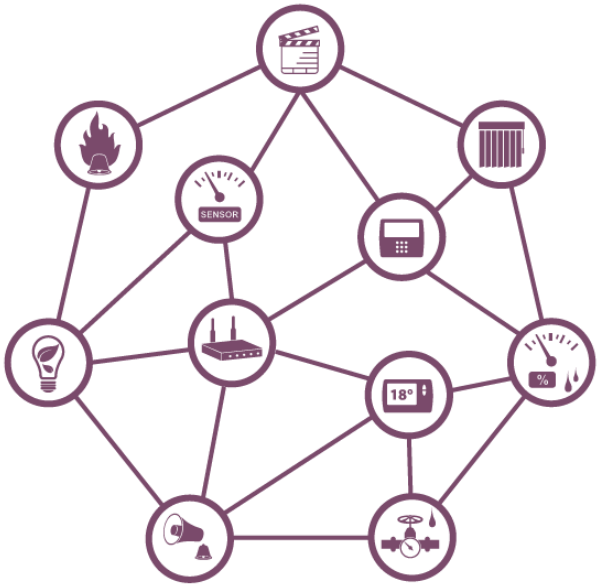
Sa naglaskom na laku proširivost, odnosno fleksibilnost sistema, potrebno je definisati jednostavan, a pritom što univerzalniji način komunikacije između komponenti sistema za pametne kuće. Trenutni zahtevi ciljanih sistema se uzimaju kao početna tačka, odnosno jedan od primera upotrebe, ali nikako ne smeju biti krajnji cilj, odnosno ograničenja trenutnih posmatranja.

Za sisteme pametnih kuća je veoma značajna velika propusna moć, mogućnost asinhrone komunikacije, odnos zaglavlja i sadržaja poruke (posebno kod kratkih poruka). To se mora uzeti u obzir i velika pažnja se mora usmeriti na **efikasnu razmenu poruka** između komponenti. To, zapravo, jeste polazna tačka i ono što definiše sisteme pametnih kuća: interakcija uređaja. Tako da su odabir i primena komunikacionih tehnologija izuzetno bitni.



Slika 1- Uređaji u sistemima pametnih kuća

Takođe, u ovom radu je predstavljen i način za povećanje efikasnosti kontrole i interakcije, pored ostalih, i u takozvanim *meš mrežama* [1]. U pomenutim mrežama se može dogoditi da propagacija razmene poruka traje duže nego što je očekivano od strane klijenta. Razlog uključivanja ove realizacije je široka zastupljenost protokola kao što su ZigBee[2] i Z-Wave[3], a koji formiraju upravo pomenuti tip mreže.




Slika 2- Primer povezivanja unutar meš mreže

Sigurnost sistema i adekvatna **verifikacija** učesnika u razmeni poruka je još jedna od osobina koja je veoma važna za sam sistem. Uvođenjem različitih tipova učesnika u razmeni poruka se javlja potreba za različitim pravima pristupa.

## 1.1 Proširivost komunikaciong sistema

Komunikacija između računara u mreži, kao i komunikacija između ljudi uživo ili putem društvenih mreža se može apstrahovati na komunikaciju između učesnika. Ti učesnici mogu da pitaju druge, dobiju odgovor na to pitanje, zahtevaju od drugih da izvrše određene radnje ili obaveste druge o događaju za koji su zainteresovani.

Komunikacioni sistem mora podržati laku integraciju novih klijenata, dok se sa druge strane razmena poruka mora biti uniformna bez obzira na broj klijenata ili njihov tip. Iz tog razloga su definisane dve osnovne komponente ovog protokola:

1. **Oznaka** klijenta i
2. **Tip razmene** poruka.

Oznaka je promenljiva u sistemu i vezuje se za klijenta. Sa globalnog stanovišta, oznaka je niz karaktera koji jednoznačno određuju klijenta. Kako ne bi došlo do kolizije u razmenama, na sistemu je da obezbedi jedinstvenost oznake.

Tip razmene poruka definiše ponašanje koje se očekuje od klijenta u komunikacionom sistemu. Za razliku od oznake, skup tipova je unapred definisan i uključuje razmene:

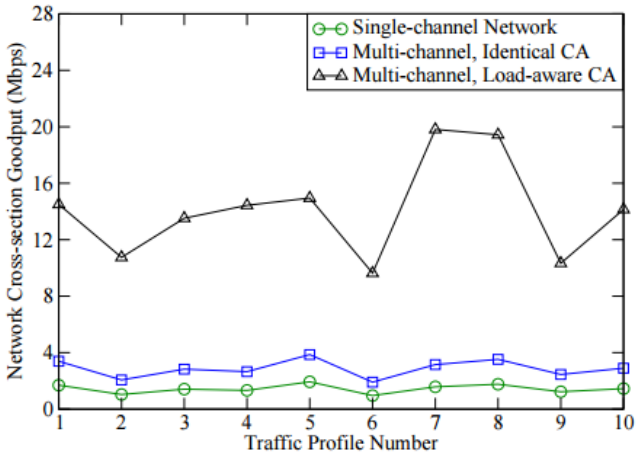
- Zahteva,
- Odgovora i
- Obaveštenja.

## 1.2 Efikasna razmena poruka

Sistemi za pametne kuće se u velikoj meri oslanjaju na asinhrone poruke koje obaveštavaju o promenama unutar sistema. Primer takvih promena su: izmerena temperatura sobe, nivo osvetljenosti, stanje prekidača, itd. Količina poruka koja se šalje tom prilikom je mnogo manja od najčešće korišćenih u mrežnoj komunikaciji (razmena HTML stranica ili video snimaka). Pored kratkih poruka, protokol mora prihvatiti podatke koje su veliki i po nekoliko megabajta. Isto tako, sistemi za pametne kuće se koriste na uređajima koji su ograničeni sa stanovišta računarskih resursa. Uzimajući u obzir različite osobine okruženja u kojem se vrši razmena poruka, sam protokol mora da odgovara veoma strogim zahtevima. Centralni kontroleri koji predstavljaju uređaje potrošačke elektronike, kao i baterijiski napajani mobilni uređaji kroz koje korisnici vrše interakciju zahtevaju opravdanu razmenu svakog bajta podataka. Time se doprinosi smanjenoj potrošnji uređaja, ali i brzini razmene koja ostavlja krajnji utisak na korisnike sistema.

Iako su meš tehnologije napravljene sa ciljem da podrže veliki broj međusobno povezanih uređaja, sama primena u sistemima za pametne kuće može naići na prepreke kao što je brzina u razmeni podataka. Performanse meš mreža opadaju sa porastom broja uređaja u mreži, ali je u samoj računici veoma bitan broj kanala [4] kojim se raspolaže. Kako je opisano u radu koji su napisali Raniwala, Ashish, Kartik Gopalan, i Tzi-cker Chiueh [5], oslanjanje na jedan kanal za kreiranje celokupne meš mreže, dovodi do značajnog gubitka performansi. Oni pokazuju da je prosečna brzina razmene podataka u meš mrežama, sa upotrebom jednog kanala, negde oko 256

kB/s (2 megabita po sekudni). Poređenja radi, uspostavom mreže na više kanala, može se doći do prosečne brzine od 2 MB/s (16 megabita po sekundi) (Slika 3).




Slika 3- Poređenje upotrebe jednog i vise kanala u meš mrežama pametnih kuća [5]

Zbog jednostavnije izrade, koordinatori meš mreža [2] u sistemima pametnih kuća se oslanjaju na konfiguraciju mreže upotrebom jednog kanala. Iako se u kontrolisanim uslovima funkcionalnost može održati, problem nastaje pri upotrebi sistema u mrežama sa velikim brojem krajnjih uređaja. Za adekvatnu upotrebu i ponašanje sistema, klijentu se mora dozvoliti da određena ograničenja prevaziđe. Potreba za povećanjem umreženih uređaja koji se koriste u jednoj kući je očekivana, posebno ako se uzme u obzir ubrzan razvoj i sve veći broj različitih tipova uređaja koji se mogu pronaći na tržištu. Mogućnost da se zahtevi tržišta ispune vodi ka većem prihvatanju i upotrebi sistema.

Sa ciljem da se poveća brzina razmene podataka i time direktno utiče na kvalitet, u ovom radu će biti predstavljen način sinhronog ponašanja distribuiranih jedinica u sistemu za pametne kuće. To će biti dodatak osnovnom radu, ali i jedan primer proširenja sistema sa novim fukcionalnostima kroz komunikacioni sistem. Očekuje se da će ovakav pristup značajno doprineti propusnosti i brzini odgovora, u poređenju sa korišćenim meš tehnologijama. Takođe, postoje ograničenja, poput prostorne udaljenosti, koja se ne mogu zaobići i koja ne dozvoljavaju upotrebu pomenutih tehnologija. Upravo u takvim okolnostima se razmena može preusmeriti na IP strukturu (bežičnim, ili putem kabla).

## 1.3 Sigurna razmena poruka

Kada se radi o integritetu poruka koji se obrađuju, može se reći da sistemi pametnih kuća uvode strožije kriterijume sigurnosti isporuke poruka nego ostali računarski sistemi. Zalaženje u identitet i lične osobine je direktno vidljivo korisnicima, što ne mora biti slučaj kod klijenata drugih tipova sistema. Zbog toga postoji posebna vrsta skepticizma i zahteva koje korisnici očekuju od posmatranih sistema.

Pored standardnog metoda šifrovanja poruka, u definisanje ovog protokola ulaze i dodatne metode autentifikacije i autorizacije učesnika (uređaja) u komunikaciji.

Autentifikacija učesnika se mora obaviti na način koji može garantovati identitet samog učesnika. Stoga je neophodno omogućiti autentifikaciju koja se razlikuje i koja je karakteristična za svakog njih.

Nakon što učesnici pristupe uspostavljenoj mreži, njihovo ponašanje mora biti kontrolisano. Pristup podacima, kao i sama interakcija moraju biti nadgledani. To se može uraditi kroz kategorizaciju učesnika, ali i prepoznavanjem njihovih individualnih ovlašćenja.

# 2. Analiza zahteva

Kako bi pokrili sve aspekte željenog načina razmene poruka, analizom četiri javno dostupna protokola: CoAP, XMPP, AMQP i MQTT [6], utvrđeno je da se najbolje rešenje može dobrim delom bazirati na MQTT protokolu. Neke od najbitnijih stvari koje su uzete u obzir su: asinhrona komunikacija, jednostavnost, kompatibilnost i fleksibilnost samog protokola. Postojanje stabilnih realizacija ostavlja mogućnost da se uz nadogradnju i adekvatno definisanje upotrebe protokola dođe do fleksibilnog rešenja koji ispunjava postavljene zahteve.

## 2.1 MQTT protokol

Dva osnovna pojma, definisana od strane MQTT protokola [7], su „tema“ (eng. “topic”) i „sadržaj“ (eng. “payload”) poruke. Sam protokol funkcioniše po principu pretplate i objave: kada su različiti učesnici u komunikaciji zainteresovani za određenu **temu**, oni će se na to pretplatiti, ili objaviti određeni **sadržaj**. Po MQTT protokolu, **broker** je centralna komponenta mreže, odnosno poslužilac u komunikaciji, a **klijenti** su korisnici specifičnih usluga brokera. Sve poruke dolaze do brokera koji služi da ih pri pristizanju preusmeri onim klijentima koji su na temu te poruke pretplaćeni. Na taj način je razdvojeno slanja poruka od njihove isporuke. Slaganjem poruka u redove za isporuku, *broker* u mnogome olakšava komunikaciju između dve strane. MQTT je protokol aplikativnog nivou iz familije internet protokola i oslanja se na isporuku garantovanu TCP porotoklom [8]. Pored toga, MQTT uvodi još tri tipa takozvanog „kvaliteta usluge“ (QoS):

1. QoS-0: U zavisnosti od stabilnosti mreže i veze sa učesnicima, isporuka može izostati, ali se može dogoditi i slučaj da ista poruka bude dostavljena više od jednog puta,
2. QoS-1: Poruka koja je poslata nakon što se neki učesnik pretplatio na tu temu, a još uvek nije otkazao pretplatu, će stići pretplatniku bez obzira na stabilnost i moguć

prekid veze. U ovom slučaju će poruka stići bar jednom, ali se mogu pojaviti duplikati poruke i

3. QoS-2: Poruka koja je poslata dok je neki od klijenata pretplaćen će sigurno stići jednom i samo jednom.

Prethodna osobina je veoma važna i može se iskoristiti pažljivom upotrebom, posebno u slučaju korišćenja na mobilnim telefonima. Zbog pokretljivost ovih uređaja, očekuje se veoma čest prekid veze. U takvim slučajevima je veoma bitno da se izbegne redudantnost poruka i smanji nepotrebna interakcija koja je uslovljena prekidima veza.

Dodatna garancija za isporuku se može primeniti na *obaveštenja*. Ukoliko se izvrši pretplata na obaveštenja, najverovatnije se očekuje da obaveštenje stigne do klijenta, bez obzira da li sa zakašnjenjem ili odmah. Upravo za taj tip razmene poruka se preporučuje upotreba QoS-1 kvaliteta usluga.

Za razliku od obaveštenja, pri slanju zahteva se očekuje što brže izvršenje. Za zahteve se ne očekuje isporuka ukoliko dođe do dugotrajnih prekida veze. Slanje zahteva je najčešće propraćeno odgovarajućim periodom u okviru kog se očekuje izvršenje, nakon čega se prijavljuje izostanak odgovora. Garancije kvaliteta usluge QoS-1 i QoS-2 uključuju nekoliko dodatnih razmena pri dostavi, što dodatno opterećuje resurse uređaja. Iz tog razloga je QoS-0 prihvatljivije rešenje za razmenu zahteva i odgovora .

MQTT teme se mogu formirati na osnovu komponenti razdvojenih znakom kose crte: „/“. U tom slučaju je omogućen mehanizam pretplate na više od jedne teme, uz pomoć samo jedne operacije (zahteva) poslate brokeru. Za jednostavniju pretplatu su ostavljena dva specijalna znaka:

- Znak plus („+“): svaka komponenta u strukturi teme može biti zamenjena ovim znakom. Na taj način se pretplata obavlja nad grupom tema, bez obzira na vrednost koja se nađe na mestu komponent „+“. Ovaj znak se može naći na mestu više komponenti, unutar iste teme za pretplatu.

  *Primer upotrebe znaka „+“*: pretplata na temu: „soba/12/obaveštenja“ se poklapa sa porukama koje obaveštavalju o događajima iz sobe 12, dok bi pretplata „soba/+/obaveštenja“ uključila obaveštenja koja dolaze iz svih soba, a ne samo iz sobe 12.
- Znak taraba („#“): koristi se kao zamena za bilo koju vrednost na jednoj ili više, krajnjih komponenti teme. Ovaj znak se može kombinovati sa znakom „+“, ali se unutar iste teme može naći samo jednom i to na kraju.

*Primer upotrebe znaka „#“*: pretplata na temu „soba/11/stanje“ se poklapa sa porukama koje opisuju trenutno stanje u sobi 11, dok bi pretplata na: „soba/#“ uključila poruke svih tipova (obaveštenja, stanja, itd.) koje stižu o svim sobama.

## 2.2 Razmena zahtev-odgovor preko MQTT protokola

Asinhrona razmena poruka jeste osnovna osobina MQTT protokola i to zadovoljava zahtev za razmenom *obaveštenja* među korisnicima. Međutim, slanje *zahteva* i dobijanje *odgovora* ne predstavlja asinhronu komunikaciju i nije pokriveno ovim protokolom. Sa druge strane, razmatrani AMQP protokol [9] je veoma sličan MQTT načinu razmene poruka. Za razliku od MQTT-a, ovaj protokol podržava razmenu po principu zagtev-odgovor, ali je njegova implementacija nešto zahtevnija. Sve to je prouzrokovano potrebama bankarskih sistema (iz kojih je AMQP potekao), koji nisu u potpunosti u skladu sa onim što se zahteva unutar posmatranih sistema. Takođe, trenutno javno dostupne realizacije AMQP broker nisu predviđene za ugradnju u ciljana okruženja, poput uređaja potrošačke elektronike.

Za razmenu zahtev-odgovor AMQP koristi metod koji bi se na MQTT mogao preslikati na sledeći način:

1. Učesnik u komunikaciji šalje poruku na temu na koju sluša učesnik od koga želi da dobije odgovor,
2. Pri slanju, pošiljalac u poruku umetne naziv teme na koju očekuje odgovor,
3. Nakon što pripremi odgovor, primalac ga šalje na temu koju je dobio u primljenoj poruci.

Metod, koji je veoma sličan predstavljenom, je iskorišćen u ovom radu i služi za ostvarivanje komunikacije po principu zahtev-odgovor. On je uklopljen u definisane formate i način razmene i kao takav u potpunosti odgovara zahtevima sistema.

## 2.3 Uloga brokera i klijenta

Pored osnovne uloge koju ima, a to je upravljanje pristiglim porukama, na brokeru je zadatak da formira mrežu sa drugim brokerima. Sa korisničke strane, broker je posrednik u razmeni podataka. Ono o čemu korisnici ne vode računa je način i put kojim poruke stižu do krajnjih ciljeva. Uspostavljanjem veza sa jednim od brokera, korisnici su u stanju da razmenjuju podatke sa bilo kojim od klijenata iz mreže. Ukoliko korisnici nisu povezani preko istog brokera, put kojim će poruka ići zavisi od prethodno uspostavljene mreže.

# 3. Definicija protokola

Logiku koja se zaniva na upotrebi *oznaka* i različitih *tipova razmene* je potrebno preslikati na MQTT protokolom definisane *teme* i *sadržaj* poruka. Primena protokola iz ovog odeljka se oslanja na realne komponente komercijalnog sistema za pametne kuće.

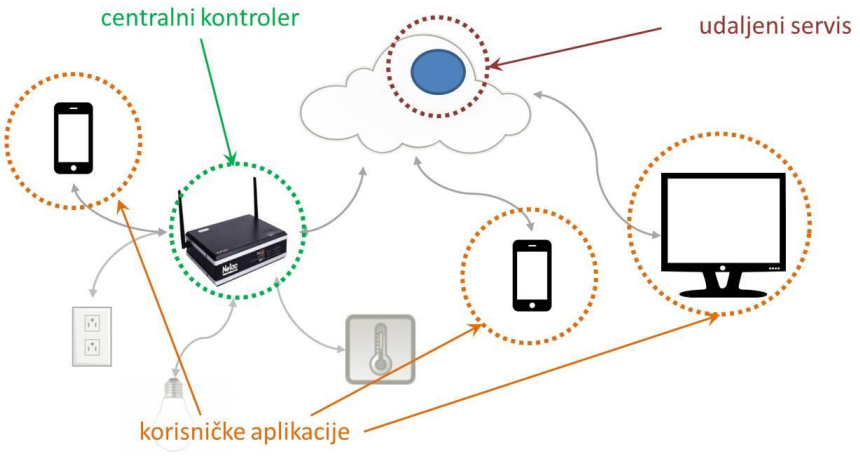
## 3.1 Format MQTT tema

Korisnici su osnova definisanog protokola. Korišćenjem oznaka unutar teme, omogućava se pretplata i slanje poruka na temu koja je vezana za određenog klijenta. Pored oznake, sa ciljem da se omoguće adekvatne pretplate i slanja, u temu je uključen i tip razmene poruka.Taj šablon je identifikovan i uspostavljen kako bi se omogućila lakša i kontrolisana razmena poruka. U tekstu ispod se nalaze primeri upotrebe tog šablona.

### 3.1.1 Oznaka učesnika

Među oznakama karakterističnim za svakog učesnika u razmeni, prepoznate su grupe sličnih. Te grupe se formiraju na osnovu uloge u samom sistemu, preslikano na komunikacionu mrežu. Za posmatrani sistem definisane su tri grupe učesnika:

1. Centralni kontroler sa modulima
2. Udaljeni servisi i
3. Korisničke klijentske aplikacije.




Slika 4- Tipovi učesnika u razmeni poruka

Sama oznaka je karakteristična za svaku od grupa, jer se naziv grupe nalazi unutar oznake učesnika. Jedini razlog raspodele klijenata po gupama je jednostavnija kontrola ponašanja sličnih učesnika. Za prethodno nabrojane grupe, Tabela 1 sadrži definiciju formata oznaka.

| Grupa | Format oznake |
|---|---|
| Kontroleri | knt/<knt-id>/<mod-id> |
| Udaljeni servisi | svs/<svs-id> |
| Klijentske aplikacije | kli/<kli-id> |

Tabela 1- Format oznaka za različite grupe

Podrazumeva se da se unutar iste grupe učesnici predstavljaju korišćenjem identifikatora koji je jedinstven za svakog od učesnika unutar grupe. Tako su identifikatori za kontrolera, modul unutar kontrolera, udaljeni servis i klijentsku aplikaciju predstavljeni koristeći: *knt-id*, *mod-id*, *svs-id* i *kli-id* respektivno.

### 3.1.2 Tip razmene poruka

Pored oznake učesnika, u format teme ulazi i tip razmene poruka. Za svaki od potrebnih i prepoznatih načina razmene je definisan po jedan tip. Pored već pomenutog tipa koji služi za razmenu *obaštenja*, uveden je dodatni koji je samo specifična pojava pomenutog. Obaveštenja koja govore o dostupnosti učesnika u mreži je izolovana od obaveštenja u posebnu grupu: *stanje*.

| Tip razmene poruka | Upotreba unutar teme |
|---|---|
| Zahtev | zht |
| Odgovor | odg |
| Obaveštenje | obv |
| Stanje | sta |

Tabela 2- Različiti tipovi razmene poruka i njihova integracija unutar teme

### 3.1.3 Primeri tema prema definisanom formatu

Format teme se dobija kada se na oznaku učesnika nadoda tip razmene poruka. Tabela 3, Tabela 4 i Tabela 5 sadrže primere tema koje su definisane za posmatrani sistem. Tabela 3 prikazuje upotrebu na primeru modula *mod-sis* za sistemske operacije kontrolera *007AC5236123*, Tabela 4 prikazuje teme predviđene za udaljeni servis koji vrši statistiku korišćenja uređaja i Tabela 5 prikazuje na koji način se teme upotrebljavaju sa strane korisničke aplikacije *12345674125412*.

| Format teme | Primer teme |
|---|---|
| oznaka/zht | knt/007AC5236123/mod-sis/zht |
| oznaka/odg | knt/007AC5236123/mod-sis/odg |
| oznaka/obv | knt/007AC5236123/mod-sis/obv |
| oznaka/sta | knt/007AC5236123/mod-sis/sta |

Tabela 3- Primer tema u slučaju modula kontrolera

| Format teme | Primer teme |
|---|---|
| oznaka/zht | svs/statistika/zht |
| oznaka/odg | svs/statistika/odg |
| oznaka/obv | svs/statistika/obv |
| oznaka/sta | svs/statistika/sta |

Tabela 4- Primer tema u slučaju udaljenog servisa

| Format teme | Primer teme |
|---|---|
| oznaka/zht | kli/12345674125412/zht |
| oznaka/odg | kli/12345674125412/odg |
| oznaka/obv | kli/12345674125412/obv |
| oznaka/sta | kli/12345674125412/sta |

Tabela 5- Primer tema u slučaju korisničke aplikacije

## 3.2 Format sadržaja poruka

U slučaju MQTT protokola ne postoji ograničenje po pitanju tipa podataka koji se može slati. MQTT prihvata niz bajta, do dužine od 256 megabajta. Za posmatrani sistem i pomenute tipove razmene je definisana upotreba tekstualnog sadržaja u JSON formata. Za svaki od prepoznatih tipova je definisan format sadržaja poruke.

### 3.2.1 Zajedničko za sve tipove

Zarad lakšeg sagledavanja, zajedniče osobine za svaki od tipova su izdvojeni i prikazani u tabeli ispod.

| Polje | Obavezno | Opis |
|---|---|---|
| id | da | Identifikator poruke koji je definisan od strane pošiljaoca poruku. |

| | | |
|---|---|---|
| parametri | ne | Proizvoljna JSON struktura koji dodatno opisuje samu poruku. |

Tabela 6- Zajednički delovi sadžaja poruka

Pored zajedničkih delova sadržaja, za svaki od tipova poruka su definisane specifičnosti.

### 3.2.2 Specifičnosti zahteva

| Polje | Obavezno | Opis |
|---|---|---|
| pošiljalac | da | Jedinstvena oznaka pošiljaoca koja se najčešće koristi kako bi se odgovor na zahtev preusmerio na adekvatnu temu. |
| tip | ne | Označava tip samog zahteva, kao što je komanda ili upit. |
| naziv | da | Naziv zahteva, kako bi strana na koju se šelje zahtev, pripremila adekvatan odgovor. |

Tabela 7- Polja unutar sadržaja zahteva

Primer sadržaja jednog zahteva:

```
{
        "id": 15,
        "pošiljalac": "kli/123",
        "tip": "komanda",
        “naziv”: “trenutniBrojUređaja”,
        "parametri": {
                “tipUređaja”: “sijalica”
        }
}
```

### 3.2.3 Specifičnosti odgovora

| Polje | Obavezno | Opis |
|---|---|---|
| kod | da | Broj koji omogućava brzu proveru validnost odgovora. Označava uspešnost postupka zahtev-odgovor, ili se koristi za identifikaciju greške. |
| opis | ne | Dodatno pojašnjenje odgovora. Najčešće dopunjava dobijeni kod. |

Tabela 8- Polja unutar sadržaja odgovora

Primer sadržaja jednog odgovora:

```
{
        "id": 15,
        "kod": 0,
        "opis": "uspešno obrađeno",
        "parametri": {
                “brojUređaja”: 12
        }
}
```

### 3.2.4 Specifičnosti obaveštenja

| Polje | Obavezno | Opis |
|---|---|---|
| naziv | da | Naziv obaveštenja, kako bi primalac obaveštenja znalo na koji način da postupi u vezi sa obaveštenjem. |

Tabela 9- Polja sadržaja obaveštenja

Primer sadržaja jednog obaveštenja:

```
{
        "id": 645,
        "naziv": "dodat novi uređaj",
        "parametri": {
                “idUređaja”: 45,
                “tipUređaja”: “termostat”
        }
}
```

### 3.2.5 Specifičnosti obaveštenja o stanju

| Polje | Obavezno | Opis |
|---|---|---|
| stanje | da | Stanje koje opisuje dostupnost učesnika na mreži. |

Tabela 10- Polja sadržaja obaveštenja o stanju

Primer sadržaja jednog obaveštenja o stanju:

```
{
        "id": 5435,
        "stanje": "dostupan"
}
```

# 4. Proširenje brokera

Kao jezgro realizacije brokera je iskorišćen *Mosquitto MQTT* broker. On u potpunosti podržava osobine MQTT 3.1.1 protokola [10]. Realizacija je pisana u C programskom jeziku i obajvljena pod EPL/EDL licencom [11]. Zahtevi protokola koji nisu deo MQTT-a su dodatno realizovani.

Proširanja uključuju:

1. Autentifikaciju učesnika,
2. Registraciju novih učesnika,
3. Autorizaciju klijenata i
4. Uspostavljanje mreže sa drugim brokerima.

## 4.1 Autentifikacija klijenta

Sastavni deo MQTT protokola su identifikator klijenta, njegovo ime i lozinka. Za razliku od imena, identifikator je specifičan za klijenta i mora biti jedinstven na brokeru. Kao i identifikator, ime i lozinka se šalju pri uspostavi veze. Iako nije zahtevano po specifikaciji, u određenim slučajevima je korisno da identifikator i ime klijenta budu jednaki. Na taj način bi svaki od klijenata morao biti registrovan pod različitim imenom. To otvara mogućnost da se jedinstvenost klijenta proverava na osnovu, za njega karakterističnih, imena i lozinke.

Kada su svi korisnici jednoznačno identifikovani, ostavlja se mogućnost za njihovu proveru i eventualnu zabranu korišćenja sistema.

Slanjem sledećeg upita se može dobiti lista trenutno registrovanih klijenata:

```
{       "id": 1545,
        "pošiljalac": "kli/administrator",
        "tip": "upit",
        “naziv”: “listaKlijenata”       }
```

```
{
        "id": 1545,
        "kod": 0,
        "parametri": {
                “klijenti”: [
                        “kli/administrator”,
                        “kli/stefan”,
                        “kli/marko”,
                ]
        }
}
```

A zatim poslati zahtev za isključivanjem nekog od postojećih:

```
{       "id": 4324,
        "pošiljalac": "kli/administrator",
        "tip": "komanda",
        “naziv”: “izbrišiKlijenta” ,
        "parametri": {
                “oznakaKlijenta”: “kli/marko”
        }
}
```

## 4.2 Registracija klijenata

Proširivost novim korisnicima se može obaviti kroz postupake za registraciju klijenata. Protokol podržava vise načina registracije sa različitim nivoima sigurnosti. Ti načini su predstavljeni u tekstu ispod.

### 4.2.1 Registracija korišćenjem tajnog ključa

Procedura registrovanja novog korisnika se sastoji iz 8 koraka:

1. Prvo se zahteva uspostava inicijalne vezu sa brokerom. Tom prilikom klijent se povezuje upotrebom imena definisanog isključivo za namenu registracije novih klijenata (npr. „novi“, „klijent“ ili samo „“),
2. Nakon uspešno obavljene uspostave veze za registraciju, klijent šalje zahtev za registraciju brokeru:

```
{
        "id": 15,
        "pošiljalac": "klijent",
        "tip": "komanda",
        “naziv”: “registracija”,
        "parametri": {
                “identifikator”: “1354546”
        }
}
```

3. Kao potvrdu prihvaćene registracije, broker odgovara i dostavlja registracioni broj:

```
{
        "id": 15,
        "kod": 0,
        "parametri": {
                “registracioniBroj”: 455644456
        }
}
```

4. Nako uspošno uspostavljene veze, broker inicira mehanizam kojim proverava klijenta. U zahtevu, broker proseleđuje nasumično odabran broj, koji će klijent u narednom koraku iskoristiti kako bi dokazao pravo na korišćenje sistema:

```
{
        "id": 433,
        "pošiljalac": "broker",
        "tip": "komanda",
        “naziv”: “izazov”,
        "parametri": {
                “broj”: “75443457”
        }
}
```

5. Kako bi prihvatio novog klijenta, broker očekuje odgovor koji sadrži broj šifrovan pomoću standarda za napredno kodovanje (AES). Samo šifrovanje treba da bude obavljeno na osnovu sledećih parametar:
    - Ključ: SHA1 kod dobijen na osnovu posebno definisanog tajnog ključa,
    - Format zapisa: niz heksadecimalnih brojeva,
    - Dužina ključa: 128 bita (16 bajta),
    - Način kodovanja: ECB i
    - Način popunjavanja: PKCS5.
6. Rezultat šifrovanja je u binarnom obliku, tako da je potrebno taj rezultat prebaciti u tekstualni oblik i poslati kroz poruku odgovora. Za to je iskorišćen Base64 algoritam.

```
{
        "id": 433,
        "kod": 0
        "parametri": {
                “broj”: “mhstbFDdjkFndjts==”
        }
}
```

7. Ukoliko se broj iz odgovora poklopi, odnosno ukoliko je klijent upoznat za tajnim ključem i procedurom kodovanja, broker će nastaviti sa procesom registracije. Naredni zahtev se odnosi na promenu korisničkog imena i lozinke:

```
{
        "id": 766,
        "pošiljalac": "broker",
        "tip": "komanda",
        “naziv”: “promenaImenaILozinke”,
        "parametri": {
                “korisničkoIme”: “kli/1354546”,
                “lozinka”: “fsjhfknsdkfgfhdgf”,
                “periodVaženja”: 7
        }
}
```

8. Na klijentu je da odgovori i prosledi registracioni broj koji mu je dostavljen na početku procedure. Na taj način broker poseduje potrebne informacije o klijentu koji je započeo process registracije. Nakon ovog koraka, klijentu je dozvoljeno da uspostavi regularnu vezu sa brokerom.

```
{
        "id": 766,
        "kod": 0
        "parametri": {
                “registracioniBroj”: 455644456
        }
}
```

Tajni ključ može biti:

1. Predefinisan – dolazi uz brokera i može se naknadno promeniti,
2. Periodično obnavljan – automatski se osvežava i može se dobiti uz pomoć klijenata koji su već registrovani ili
3. Jednokratno generisan – generiše se po zahtevu nekog od registrovanih klijenata i važi za jednu registraciju.

### 4.2.2 Otvorena registracija

U ovom vidu registracije nije zahtevano da klijent potvrdi svoj identitet preko dodatnih informacija ili ključeva. Postupak je istovetan kao i u slučaju opisanom u 4.2.1, sa izuzetkom koraka 4, 5 i 6. Ovaj način predstavlja sigurnosnu manu i ne sme biti izabran kao osnovna postavka, ali se može aktivirati po potrebi.

Osnovni cilj je da se olakša i ubrza postupak registracije, dok se sigurnost prebacuje na postavku lokalne mreže. Prvenstveno, ovakav metod olakšava upotrebu korisničkih klijentskih aplikacija. S obzirom da je pristup lokalnoj mreži najčešće ograničen na klijente koji bi smeli pristupiti sistemu, ovaj metod može doprineti jednostavnijoj i lakšoj upotrebi celog sistema.

U kombinaciji sa obaveštenjima i mogućnosti izlistavanja i regulisanja registrovanih klijenata, ovakav način može biti prihvatljiv i primenljiv u velikom broju slučajeva.

## 4.3 Autorizacija klijenata

Oznake klijenata se koriste kako bi se proverilo njihovo ponašanje unutar sistema. Postoje određena prava koja važe za pojedinačne klijente, ali se kontrola pristupa u najvećoj meri obavlja na osnovu pripadnosti određenoj grupi. Kao što je napomenuto, naziv grupe je integrisan u oznaku klijenta.

### 4.3.1 Kontrola pristupa na osnovu grupe

Svaki modul unutar kontrolera ima pravo da:

1. Vrši pretplate na teme koje određuju njegove zahteve i odgovore,
2. Vrši objave na teme koje su vezane za njegova obaveštenja,
3. Objavljuje na teme koje su vezane za zahteve i odgovore drugih modula, udaljenih servisa, klijentskih aplikacija i neregistrovanih klijenata i
4. Vrši pretplate na teme koje su vezane za obaveštenja drugih modula, udaljenih servisa, klijentskih aplikacija i neregistrovanih klijenata.

Tabela 11 sadrži detaljan opis prava pristupa koja važe za module kontrolera.

| Grupa | Tema | Dozvola |
|---|---|---|
| Kontroler sa modulima | knt/< KNT-ID >/< *MODUL* >/zht | Pretplata |
| Kontroler sa modulima | knt/< KNT-ID >/< *MODUL* >/odg | Pretplata |
| Kontroler sa modulima | knt/< KNT-ID >/< *MODUL* >/obv | Objava |
| Kontroler sa modulima | knt/< KNT-ID >/< *MODUL* >/sta | Objava |
| Kontroler sa modulima | knt/< KNT-ID >/< *, mod_id ≠ *MODUL* >/zht | Objava |
| Kontroler sa modulima | knt/< KNT-ID >/< *, mod_id ≠ *MODUL* >/odg | Objava |
| Kontroler sa modulima | knt/< KNT-ID >/< *, mod_id ≠ *MODUL* >/obv | Pretplata |
| Kontroler sa modulima | knt/< KNT-ID >/< *, mod_id ≠ *MODUL* >/sta | Pretplata |
| Kontroler sa modulima | knt/< *, knt_id ≠ KNT-ID >/+/zht | Objava |
| Kontroler sa modulima | knt/< *, knt_id ≠ KNT-ID >/+/odg | Objava |
| Kontroler sa modulima | knt/< *, knt_id ≠ KNT-ID >/+/obv | Pretplata |
| Kontroler sa modulima | knt/< *, knt_id ≠ KNT-ID >/+/sta | Pretplata |
| Udaljeni servisi | svs/+/zht | Objava |
| Udaljeni servisi | svs/+/odg | Objava |
| Udaljeni servisi | svs/+/obv | Pretplata |
| Udaljeni servisi | svs/+/sta | Pretplata |
| Klijentske aplikacije | kli/+/zht | Objava |
| Klijentske aplikacije | kli/+/odg | Objava |
| Klijentske aplikacije | kli/+/obv | Pretplata |
| Klijentske aplikacije | kli/+/sta | Pretplata |
| Neregistrovani klijent | klijent/zht | Objava |
| Neregistrovani klijent | klijent/odg | Objava |
| Neregistrovani klijent | klijent/obv | Pretplata |
| Neregistrovani klijent | klijent/sta | Pretplata |

Tabela 11- Prava pristupa za module kontrolera

Svaki udaljeni servis ima pravo da:

1. Vrši pretplate na teme koje određuju njegove zahteve i odgovore,

2. Vrši objave na teme koje su vezane za njegova obaveštenja,
3. Objavljuje na teme koje su vezane za zahteve i odgovore drugih udaljenih servisa, modula i klijentskih aplikacija i
4. Vrši pretplate na teme koje su vezane za obaveštenja drugih udaljenih servisa, modula i klijentskih aplikacija.

Tabela 12 sadrži detaljan opis prava pristupa koja važe za udaljene servise.

| Grupa | Tema | Dozvola |
|---|---|---|
| Udaljeni servisi | svs/< *SERVIS* >/zht | Pretplata |
| Udaljeni servisi | svs/< *SERVIS* >/odg | Pretplata |
| Udaljeni servisi | svs/< *SERVIS* >/obv | Objava |
| Udaljeni servisi | svs/< *SERVIS* >/sta | Objava |
| Udaljeni servisi | svs/< *, svs_id ≠ *SERVIS* >/zht | Objava |
| Udaljeni servisi | svs/< *, svs_id ≠ *SERVIS* >/zht | Objava |
| Udaljeni servisi | svs/< *, svs_id ≠ *SERVIS* >/zht | Pretplata |
| Udaljeni servisi | svs/< *, svs_id ≠ *SERVIS* >/zht | Pretplata |
| Kontroler sa modulima | knt/+/+/zht | Objava |
| Kontroler sa modulima | knt/+/+/odg | Objava |
| Kontroler sa modulima | knt/+/+/obv | Pretplata |
| Kontroler sa modulima | knt/+/+/sta | Pretplata |
| Klijentske aplikacije | kli/+/zht | Objava |
| Klijentske aplikacije | kli/+/odg | Objava |
| Klijentske aplikacije | kli/+/obv | Pretplata |
| Klijentske aplikacije | kli/+/sta | Pretplata |

Tabela 12- Prava pristupa za udaljene servise

Svaka korisnička klijentska aplikacija ima pravo da:

1. Vrši pretplate na teme koje određuju njegove zahteve i odgovore,
2. Vrši objave na teme koje su vezane za njegova obaveštenja,
3. Objavljuje na teme koje su vezane za zahteve i odgovore modula kontrolera i
4. Vrši pretplate na teme koje su vezane za obaveštenja modula kontrolera.

Tabela 13 sadrži detaljan opis prava pristupa koja važe za korisničke klijentske servise.

| Grupa | Tema | Dozvola |
|---|---|---|
| Klijentske aplikacije | kli/< *KLIJENT* >/zht | Pretplata |
| Klijentske aplikacije | kli/< *KLIJENT* >/odg | Pretplata |
| Klijentske aplikacije | kli/< *KLIJENT* >/obv | Objava |
| Klijentske aplikacije | kli/< *KLIJENT* >/sta | Objava |
| Kontroler sa modulima | knt/+/+/zht | Objava |
| Kontroler sa modulima | knt/+/+/odg | Objava |
| Kontroler sa modulima | knt/+/+/obv | Pretplata |
| Kontroler sa modulima | knt/+/+/sta | Pretplata |

Tabela 13- Prava pristupa za klijentske aplikacija

Neregistrovani klijent ima pravo da:

1. Vrši pretplate na teme koje određuju njegove zahteve i odgovore,
2. Vrši objave na teme koje su vezane za njegova obaveštenja i

3. Objavljuje na teme koje su vezane za zahteve i odgovore funkcionalnosti za registraciju na brokeru kontrolera.

Tabela 12 sadrži detaljan opis prava pristupa koja važe za neregistrovane klijente.

| Grupa | Tema | Dozvola |
|---|---|---|
| Neregistrovani klijent | klijent/zht | Pretplata |
| Neregistrovani klijent | klijent/odg | Pretplata |
| Neregistrovani klijent | klijent/obv | Objava |
| Neregistrovani klijent | klijent/sta | Objava |
| Kontroler sa modulima | knt/KNT/broker/reg/zht | Objava |
| Kontroler sa modulima | knt/KNT/broker /odg | Objava |

Tabela 14- Prava pristupa za neregistrovane klijente

Kao što je prikazano kroz tabele iz ovog poglavlja, učesnicima u komunikaciji je dozvoljeno da:

- Prime zahtev,
- Prime odgovor,
- Pošalju obaveštenje i
- Pošalju poruku stanja.

Takođe, dozvoljeno im je da:

- Šalju zahteve nekim učesnicima,
- Šalju odgovore nekim učesnicima,
- Slušaju na obaveštenja nekih klijenata i
- Slušaju na obaveštenja o stanju nekih klijenata.

### 4.3.2 Prava pristupa na funkcionalnosti

Kao dodatak prethodnom definisanom pravu pristupa, korisnicima se dozvoljava da definišu sebi svojstvene *funkcionalnosti*. Naziv same funkcionalnosti se nadodaje na oznaku klijenta. Tako se zahtevi mogu slati na određenu funkcionalnost, ali se, takođe, mogu dobijati obaveštenja vezana za određenu funkcionalnost.

Kako bi se omogućila odgovarjuća prava pristupa, od klijenta je zahtevano da pored naziva funkcionalnosti navede i klijente, ili grupe klijenata koji tim funkcionalnostima mogu pristupati. Primer takvog zahteva:

```
{
        "id": 53423,
        "pošiljalac": "kli/64574444",
        "tip": "komanda",
        “naziv”: “prijavaFunkcionalnosti”,
        "parametri": {
                “naziv”: “sistemskaKontrola”,
                “korisniciZahteva”: [“aplikacije/*”],
                “korisniciObaveštenja”: [“aplikacije/*”, “kli/*”]
        }
}
```

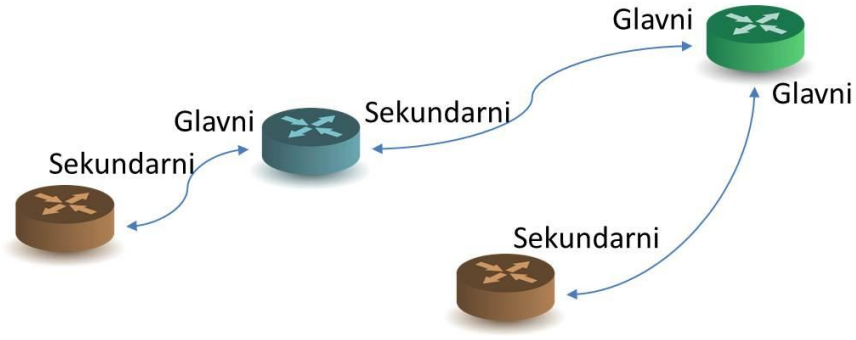
Funkcionalnosti dozvoljavaju prava pristupa koja izlaze iz okvira prethodno definisanih „opštih“ prava za grupe klijenata. Jedan od primera iskorišćenih funkcionalnosti je *registracija*, kojoj pristup imaju isključivo neregistrovani korisnici.

## 4.4 Formiranje mreže

Uspostavom veze sa jednim brokerom, klijentu se otvara mogućnost razmene poruka sa dostupnim klijentom iz mreže. Koliko će klijenata biti uključeno u razmenu, zavisi isključivo od prethodno uspostavljene mreže. Za njeno formiranje i održavanje su zaduženi brokeri, a njihove akcije se mogu inicirati kroz definisanu spregu.

Ovaj dodatak predstavlja prosleđivanje potrebnih poruka kroz vezu do drugog brokera. Pri pristizanju na broker, pored isporuke klijentima, poruke se isporučuju i povezanim brokerima. Na taj način se poruke distribuiraju i dolaze do klijenata koji nisu u bliskoj vezi.

Za potrebe pravilnog definisanja, brokeri su podeljeni u dve kategorije. Broker koji uspostavlja vezu se naziva „glavni“, dok je broker na koji je povezan „sekundarni“. Pojam glavni i sekundarni se odnosi samo na vezu dva brokera. Ukoliko postoji više brokera u mreži, za svaku vezu postoji po jedan glavni. Slika 5 prikazuje primer mreže brokera sa njihovim ulogama.

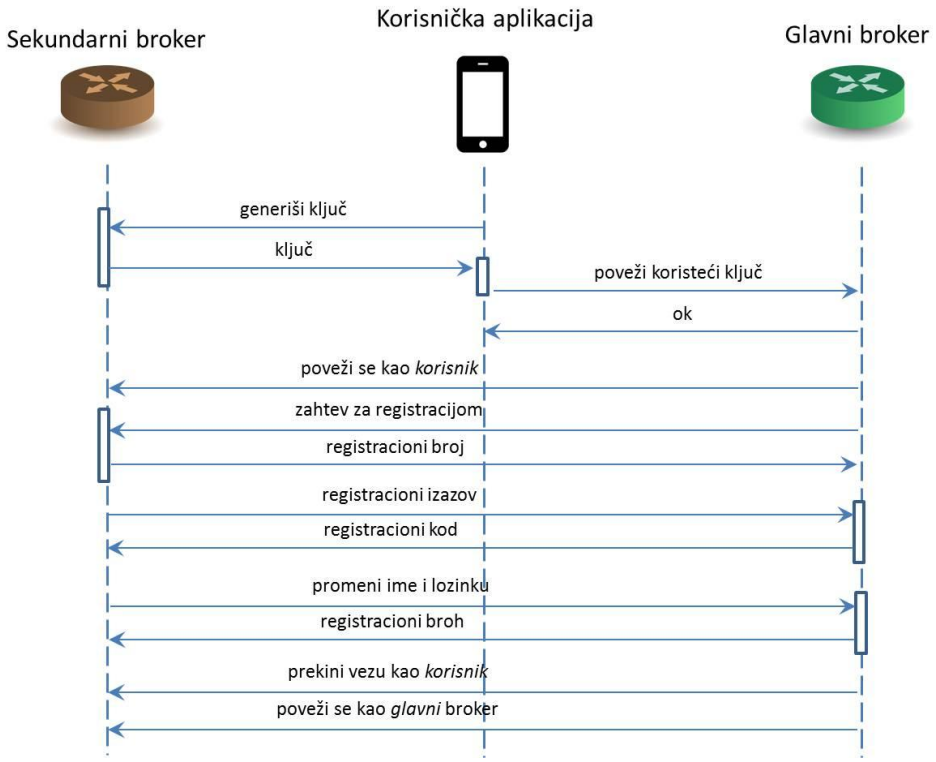



Slika 5- Mreža brokera

Realizacija uspostave mreže zahteva da se:

- Omogući pretraga dostupnih broker u lokalnoj mreži korišćenjem SSDP protokola,

- Održava veza sa sekundarnim brokerom:
    - Uključje postupak ponovne uspostave veze (od trenutka prekida) i
    - Ponovna pretraga u slučaju promene adrese sekundarnog brokera,
- Realizuje sprega za izlistavanje trenutnih veza,
- Realizuje sprega za povezivanja sa novim brokerom (Slika 6),
- Realizuje sprega za poništavanje pretodno uspostavljenih veza
    - Uz potvrdu zahteva i
    - Slanje obaveštenja o izvršenom zahtevu,
- Pravilno prosleđuju poruke kroz vezu
    - Slanje zahteva i odgovora od glavnog ka sekundarnom brokeru, ukoliko te poruke nisu namenjene korisnicima direktno povezanim sa glavnim brokerom i
    - Slanje svih obaveštenja koji dolaze sa sekundarnog na glavni broker, osim onih poruka koji su potekli od klijenata direktno povezanih sa glavnih brokerom.



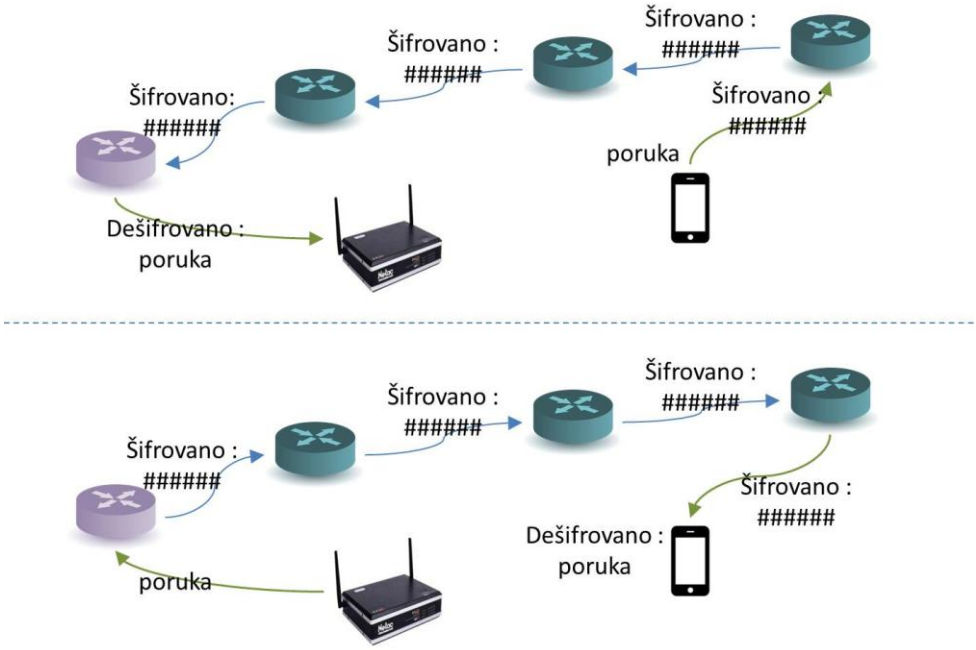

Slika 6- Postupak za povezivanje brokera

## 4.5 Šifrovanje između brokera i klijenata

Pored podrazumevane upotrebe SSL/TLS protokola na nižim nivoima veze, ostavlja se mogućnost za upotrebu dodatne metode zaštite podataka. To je poželjno, jer se, pored sigurnosti, u određenim situacijama mora garantovati privatnost podataka sa kojima se raspolaže. Korisnici posmatranog sistema mogu biti osetljivi, ili strogo protiv upotrebe podataka u neke druge svrhe (marketing, naučna istraživanja, itd.). Konstantna upotreba uređaja u posmatranim sistemima

dovodi do velike količine podataka koji se adekvatnom analizom mogu iskoristiti. Ono što je zahtev je da se onemogući "trećoj strani" da do tih podataka dođe. Zbog toga se ostavlja prostor za dodatnu zaštitu podataka.

Ono što je karakteristično za definisani protokol je da učesnici u komunikaciji ne moraju biti povezani posredovanjem jednog brokera, već se na putu podataka može naći više njih. Sa druge strane, nije nužno da klijenti tokom vremena uvek budu povezani sa samo jednim brokerom. U toku njihove migracije na različite lokacije, oni mogu menjati brokere. Kako bi se zaštitila privatnost podataka pri prolasku kroz posrednike, može se koristiti šifrovanje podataka između klijenata i krajnjeg brokera. Slika 7 sadrži primer razmene zaštićenih podataka.



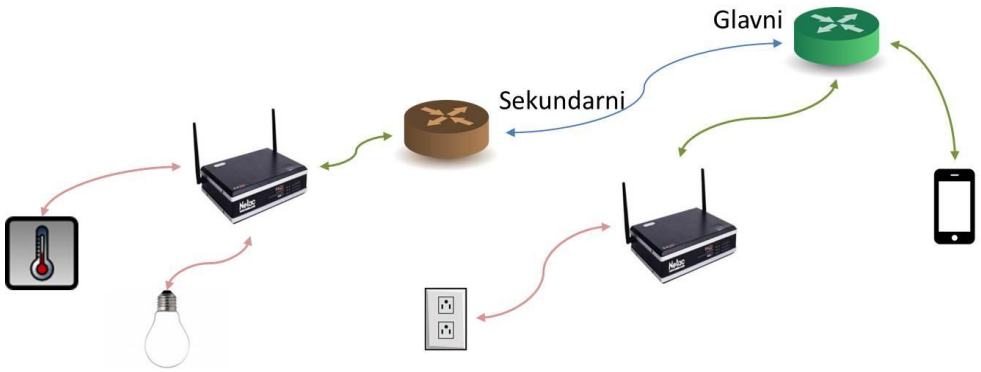

Slika 7- Zaštita podataka pri prolasku kroz više brokera

U proces registracije (odeljak 4.2), uključena je završna faza u kojoj se za svakog klijenta dodeljuje za njega karakteristična lozinka. Ta lozinka je dodeljena u direktnoj razmeni između brokera i klijenta i niko drugi nema pristup tom podatku (prema pravima pristupa iz odeljka 4.3). U cilju zaštite privatnosti podataka koji prolaze kroz posrednike do krajnjeg brokera, klijent svoje poruke može šifrovati korišćenjem lozinke koju je dobio prilikom registracione procedure sa krajnjim brokerom. Na taj način će porka biti dešifrovana samo na krajnjem, bez obzira na broj brokera preko kojih prolazi. Pored zaštite razmene podataka, na ovaj način se garantuje i autentičnost klijenata koji su u mrežu uključene preko drugih brokera. Da bi se ovakva zaštita mogla primeniti, neophodno je da klijent ima lozinku koja je validna na krajnjem brokeru.

# 5. Distribuirani uređaji

U cilju poboljšanja performansi sistema pri upotrebi velikog broja uređaja, treba iskoristiti prednost koju donosi definisani protokol. Jedna od osnovnih osobina ovog protokola je mogućnost razmene podataka između različitih klijenata u formiranoj mreži.

Centalni kontroleri su zaduženi za komunikaciju sa uređajima, između ostalih i onih uključenih u ZigBee i Z-Wave meš strukture. Ukoliko se u razmatranje uzmu korisnici koji predstavljaju te kontrolere, odnosno njihove module, uočava se mogućnost njihove interakcije, bez obzira na različite brokere preko kojih su priključeni.




Slika 8- Raspodela uređaja na više kontrolera

Ako se sistem od više kontrolera posmatra kao jedan, oni međusobno moraju deliti zahteve i ponašanje. Jedan od primera jeste simultano aktiviranje svih uređaja istog tipa (npr. utičnica) sa povezanih kontrolera. Iako je zahtev usmeren ka kontroleru na glavnom brokeru (sa korisničke aplikacije), on se može propagirati do kontrolera koji je povezan na sekundarnog brokera. Pri prihvatu takvog zahteva, kontroler može isto zahtevati od drugih kontrolera iz uspostavljene mreže. Na taj način se ponašanje jednog kontrolera presikava na ponašanje preostalih iz definisane strukture, a da ne mora voditi računa o njihovoj poziciji unutar globalne mreže.

Sa uspostavljenom mrežom i jednostavnom razmenom podataka između kontrolera, ostavlja se mogućnost da se željene performanse postignu nadogradnjom na osnovu potreba, a da se pri tom ne izgubi smisao uređaja koji zajednički deluju.

# 6. Merenja

Osnovni cilje ovih merenja je da se pokaže očekivano vreme odgovora na zahteve, pri različitoj veličini poruka i broju klijenata koji istovremeno učestvuju u razmeni. Pored zahteva i odgovora, u razmeni se koriste i obaveštenja, ali se ponašanje pri njihovoj upotrebi može proceniti na osnovu upotrebe zahteva i odgovora.

Merenje je rađeno na brokeru pokrenutom na RaspberryPi B+ platformi, sa 512 MB RAM memorije i jednojezgarnim procesorom takta 700 MHz. Mobilne aplikacije su oponašane na PC računaru sa procesorom Intel(R) Core(TM) i7-4510U, 2.00 GHz radnog takta i 2 fizička jezgra i radnom memorijom od 8 GB.

Svaki od klijenata koji su oponašali mobilne klijentske aplikacije je pokrenut u zasebnoj programskoj niti i podrazumeva se da je svaki od njih za vreme merenja konstantno razmenjivao poruke. Pauza između prijema odgovora i slanja narednog zahteva, sa strane jednog klijenta, je 5 sekundi. Svako od merenja je samo određeni vremenski period imalo aktivirane sve modele klijenata. Programske niti (koje odgovaraju korisnicima) su pokretane postepeno do definisanog intervala za puno opterećenje.

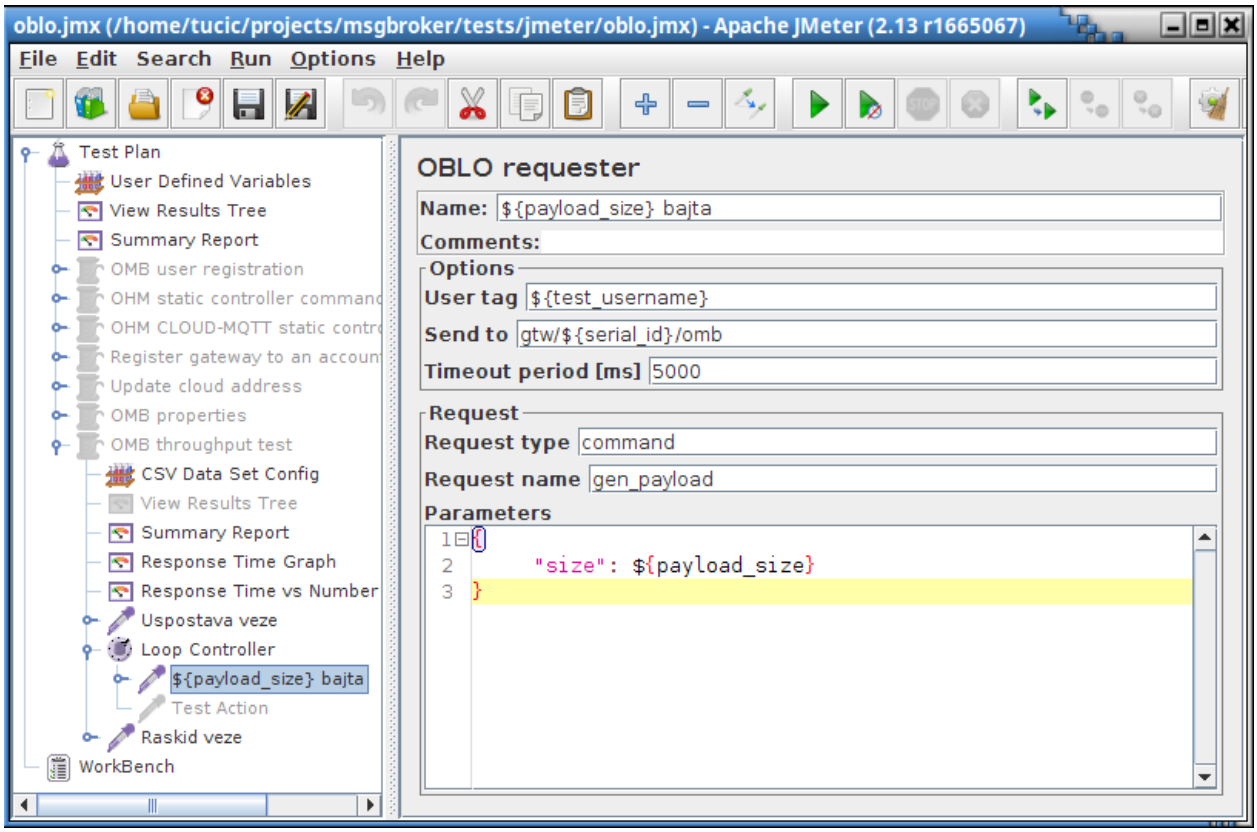
Na RaspberryPi platformi je pokrenut modul (model modula centralnog kontrolera) koji je prihvatao zahteve i slao odgovore nazad na brokera, ka korisnicima. Zahtevi su se odnosili isključivo na generisanje dodatnog sadržaja određene veličine u poruci odgovora. Po prijemu zahteva, sadržaj zahtevane veličine je generisan i umetnut u odgovor. Vreme koje je potrebno za samo generisanje sadržaja je dovoljno malo da se može reći da ne utiče na krajnji rezultat merenja. Primera radi, pri zahtevu za 4 MB podataka je za tu operaciju potrebno manje od 1 ms, dok je u slučaju 120 B potrebno oko 2 µs. Za sisteme pametnih kuća se očekuje vreme odgovora koje je manje od 1s za upotrebu u lokalnoj mreži, sa oko 50 istovremeno povezanih klijenata.

PC računar je bežičnim putem povezan sa lokalnim mrežnim kontrolerom, a RaspberryPi je na mrežu priključen pomoću mrežnog kabla.

Za potrebe merenja je korišćen javno dostupan alat *JMeter* [12]. On je namenjen simulaciji klijenata, a napravljen je za potrebe merenja performansi Apache Tomcat poslužioca [13]. Zasniva se na univerzalnoj, lako proširivoj strukturi. Ovaj alat dolazi sa podrškom za različite protokole, kao što su: HTTP(S), SOAP/REST, FTP, JDBC, LDAP, JMS, SMTP(S), POP3(S), IMAP(S) kao i TCP. U potpunosti je pisan u Java programskom jeziku. U osnovi se oslanja na paralelno izvršavanje. Poseduje jednostavanu grafičku korisničku spregu. Omogućava prikaz i analizu rezultata na različite načine. Komponente kojima se ovaj alat može proširiti su:

- Dodaci za protokole,
- Dodaci za merenja vremena,
- Dodaci za prikaz rezultata merenja, kao i
- Proširenja na nivou skript jezika.

Za potrebe ovih merenja je realizovan dodatak za definisani protokol (Slika 9). Sama integracija i korišćenje ovog alata je u mnogome olakšalo i ubrzalo postupak merenja.




Slika 9 – Korisnička sprega JMeter alata

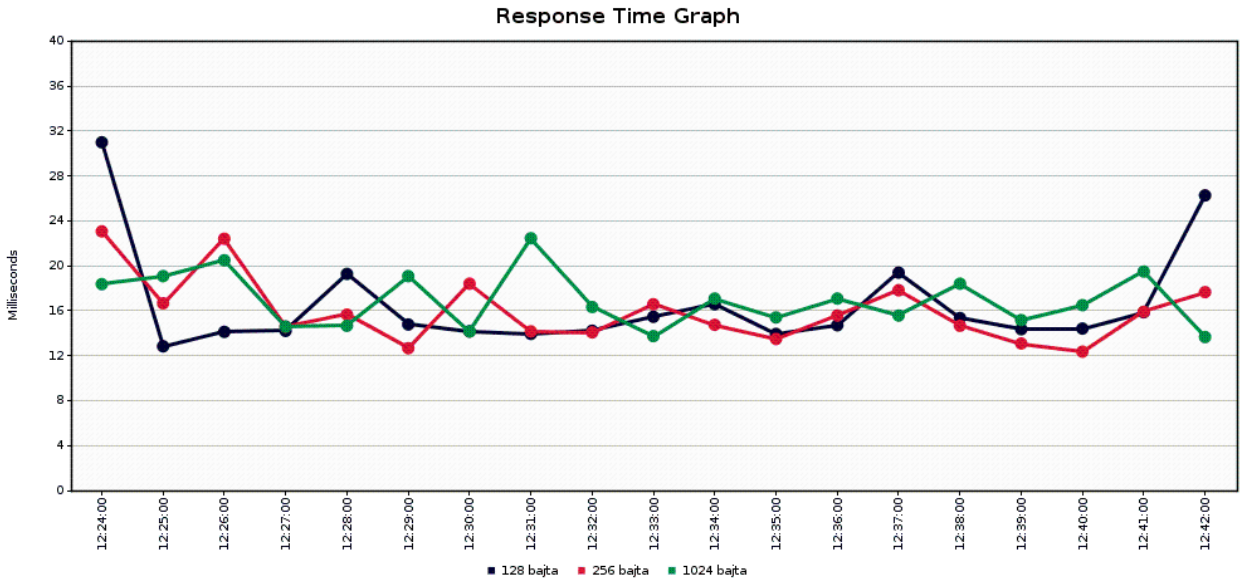
## 6.1 Kratke poruke sa 5 klijenata

Ovo merenje je trajalo oko 18 minuta, sa 5 modela korisničkih aplikacija u paraleli. Za to vreme je dostavljeno po 1000 zahteva i odgovora. Prosečna veličina razmene koja uključuje 1 zahtev i 1 odgovor je u ovom slučaju 433 bajta. Dobijeno prosečno vreme odgovora, od trenutka slanja zahteva je **16 ms**.

| Zahtev | Uzorak | Prosečno vreme izvršavanja [ms] | Najkraće vreme izvršavanja [ms] | Najduže vreme izvršavanja [ms] | Standardna devijacija [ms] | Greška [%] | Propu-snost [kom/s] | Protok [KB/s] | Prosečna veličina [B] |
|---|---|---|---|---|---|---|---|---|---|
| Uspostava veze | 5 | 126 | 117 | 135 | 6.95 | 0.00 | 0 | 0 | 0 |
| 128 bajta | 600 | 15 | 10 | 298 | 16.62 | 0.00 | 0.5 | 0.12 | 232.5 |
| 256 bajta | 200 | 15 | 10 | 80 | 8.07 | 0.00 | 0.2 | 0.07 | 360.6 |
| 1024 bajta | 200 | 16 | 12 | 116 | 9.14 | 0.00 | 0.2 | 0.22 | 1128.5 |
| Raskid veze | 5 | 1 | 1 | 2 | 0.4 | 0.00 | 0 | 0 | 0 |
| UKUPNO | 1010 | 16 | 1 | 298 | 15.98 | 0.00 | 0.9 | 0.37 | 433 |

Tabela 15- Rezultati za kratke poruke i 5 klijenata

Tokom vremena, prosečno vreme izvršavanja je bilo konstantno, sa standardnom devijacijom od 15,98 ms.




Slika 10- Kretanje vremena odgovora za kratke poruke i 5 klijenata

## 6.2 Kratke poruke sa 20 klijenata

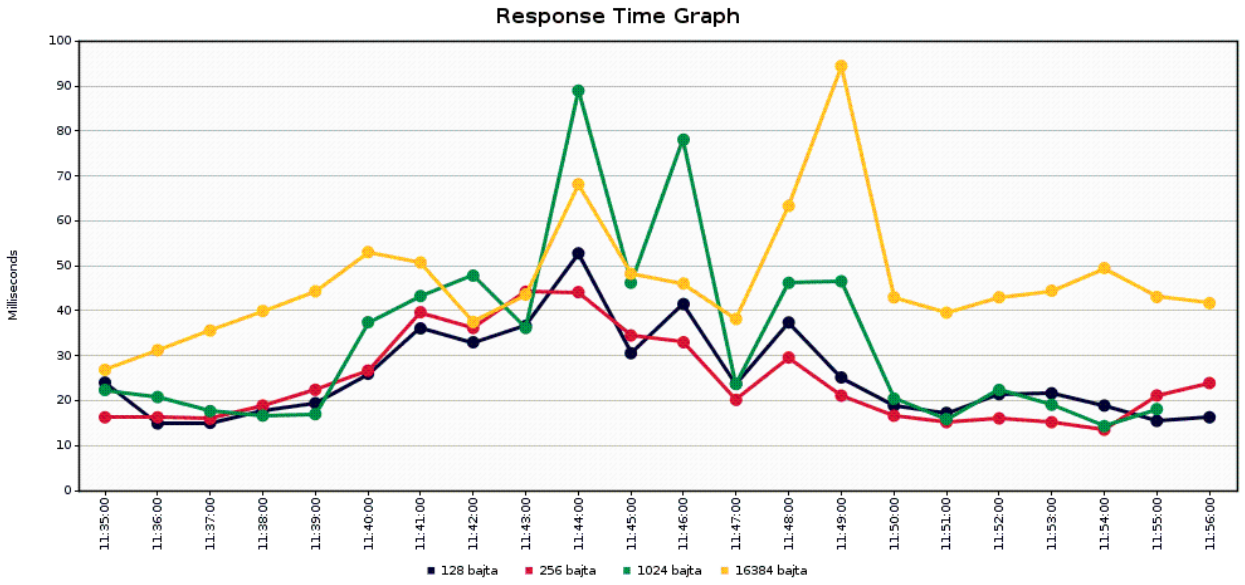
Ovo merenje je trajalo oko 21 minut, sa 20 modela korisničkih aplikacija u paraleli. Za to vreme je dostavljeno po 4000 zahteva i odgovora. Prosečna veličina razmene koja uključuje 1 zahtev i 1 odgovor je u ovom slučaju 1168 bajta. Dobijeno prosečno vreme odgovora, od trenutka slanja zahteva je **31 ms**.

| Zahtev | Uzorak | Prosečno vreme izvršavanja [ms] | Najkraće vreme izvršavanja [ms] | Najduže vreme izvršavanja [ms] | Standardna devijacija [ms] | Greška [%] | Propu-snost [kom/s] | Protok [KB/s] | Prosečna veličina [B] |
|---|---|---|---|---|---|---|---|---|---|
| Uspostava veze | 20 | 379 | 120 | 5131 | 1090.02 | 0.00 | 0.1 | 0 | 0 |
| 128 bajta | 2000 | 27 | 11 | 258 | 28.87 | 0.00 | 1.6 | 0.36 | 232.5 |
| 256 bajta | 1400 | 26 | 11 | 240 | 25.57 | 0.00 | 1.1 | 0.39 | 360.5 |
| 1024 bajta | 400 | 36 | 12 | 232 | 34.15 | 0.00 | 0.4 | 0.41 | 1128.6 |
| 16384 bajta | 200 | 49 | 31 | 278 | 30.66 | 0.00 | 0.2 | 3.2 | 16488. |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | 4 |
| Raskid veze | 20 | 2 | 1 | 4 | 0.78 | 0.00 | 0.1 | 0 | 0 |
| UKUPNO | 4040 | 31 | 1 | 5131 | 85.57 | 0.00 | 3.1 | 3.57 | 1168 |

Tabela 16- Rezultati za kratke poruke i 20 klijenata

Standardna devijacija je u ovom slučaju veća, nego sa 5 klijenata i iznosi 85,57 ms. Puno opterećenje (svih 20 klijenata) je dostignuto nakon 8 minuta od pokretanja, a nakon 4. minuta se primećuje postepeno povećanje vremena koje je potrebno za izvršenje.



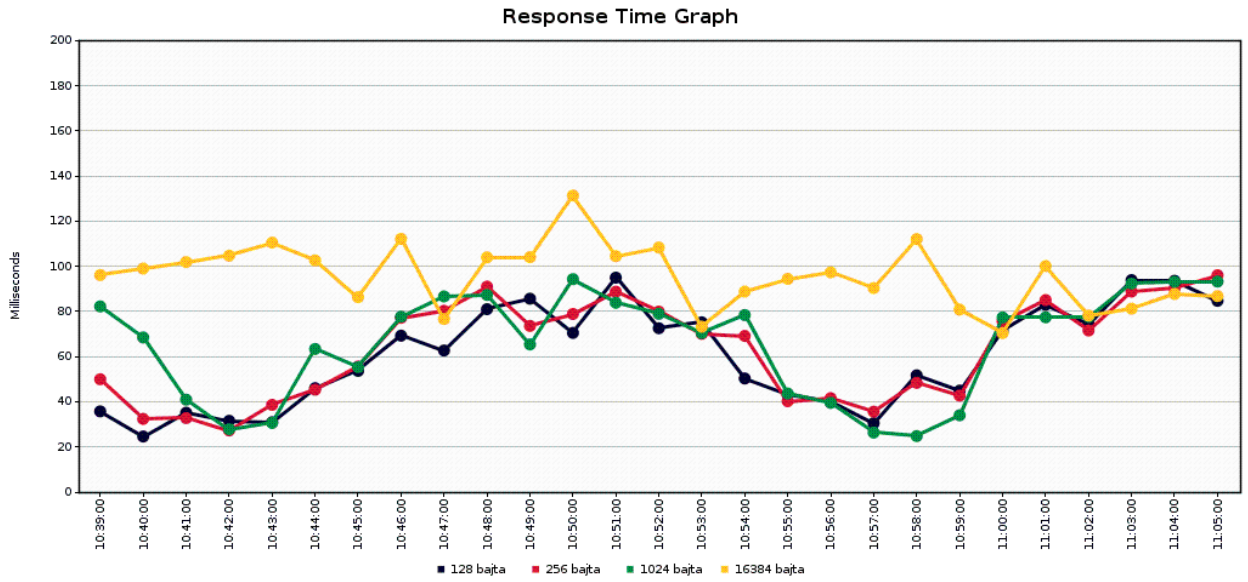

Slika 11- Kretanje vremena odgovora za kratke poruke i 20 klijenata

## 6.3 Kratke poruke sa 47 klijenata

Ovo merenje je trajalo oko 25 minuta i za to vreme je poslato 9400 zahteva i isto toliko odgovora. U merenju je učestvovalo 47 modela mobilnih aplikacija, koji su slali zahteve za sadržajem veličine od 128 do 16384 bajta. Prosečna razmena je iznosila oko 1,4 kB. Dobijeno prosečno vreme odgovora je **65 ms**.

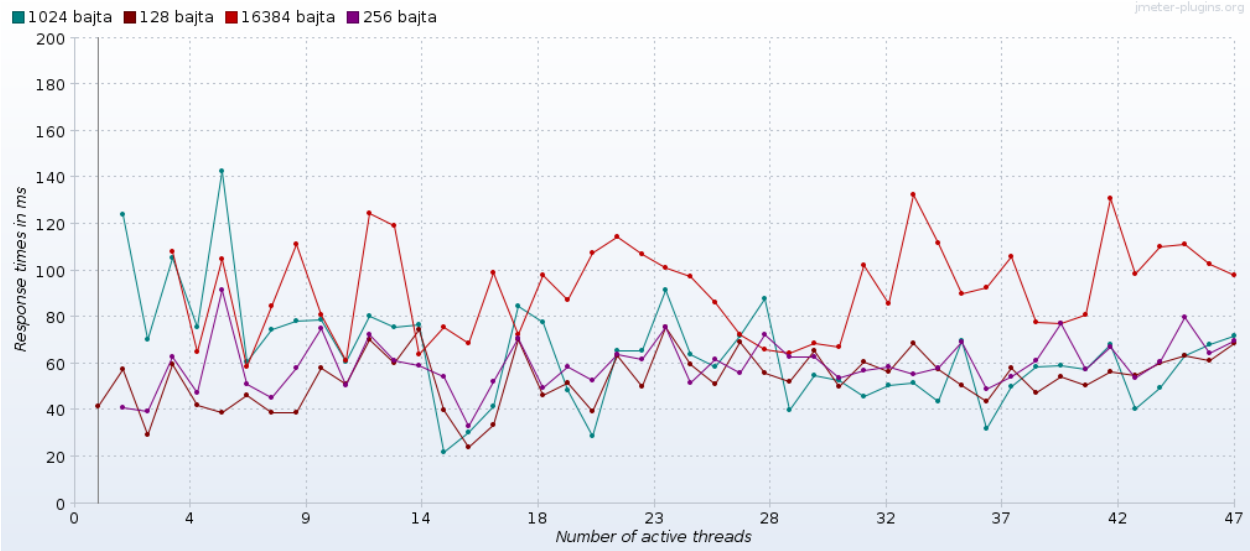
| Zahtev | Uzorak | Prosečno vreme izvršavanja [ms] | Najkraće vreme izvršavanja [ms] | Najduže vreme izvršavanja [ms] | Standardna devijacija [ms] | Greška [%] | Propu-snost [kom/s] | Protok [KB/s] | Prosečna veličina [B] |
|---|---|---|---|---|---|---|---|---|---|
| Uspostava veze | 47 | 147 | 114 | 256 | 36.6 | 0.00 | 0.1 | 0 | 0 |
| 128 bajta | 4200 | 61 | 11 | 622 | 52.01 | 0.00 | 2.6 | 0.6 | 232.5 |
| 256 bajta | 3400 | 64 | 11 | 667 | 54.47 | 0.00 | 2.2 | 0.77 | 360.5 |
| 1024 bajta | 1200 | 64 | 12 | 626 | 53.11 | 0.00 | 0.8 | 0.89 | 1128.5 |
| 16384 bajta | 600 | 95 | 31 | 256 | 40.16 | 0.00 | 0.4 | 7.17 | 16488.5 |
| Raskid veze | 47 | 1 | 1 | 5 | 0.88 | 0.00 | 0.1 | 0 | 0 |
| UKUPNO | 9494 | 65 | 1 | 667 | 53.32 | 0.00 | 5.9 | 8.19 | 1416.6 |

Tabela 17- Rezultati za kratke poruke i 47 klijenata




Slika 12- Kretanje vremena odgovora za kratke poruke i 47 klijenata

Korisnici su postepeno pokretani u roku od 10 minuta. To znači da je puno opterećenje, sa svih 47 klijenata bilo prisutno nakon 10. minuta merenja. Primetno je postepeno povećanje vremena odgovora, sa povećanjem broja klijenata koji su istovremeno aktivni.




Slika 13- Vreme odgovora po broju klijenata za kratke poruke i 47 klijenata

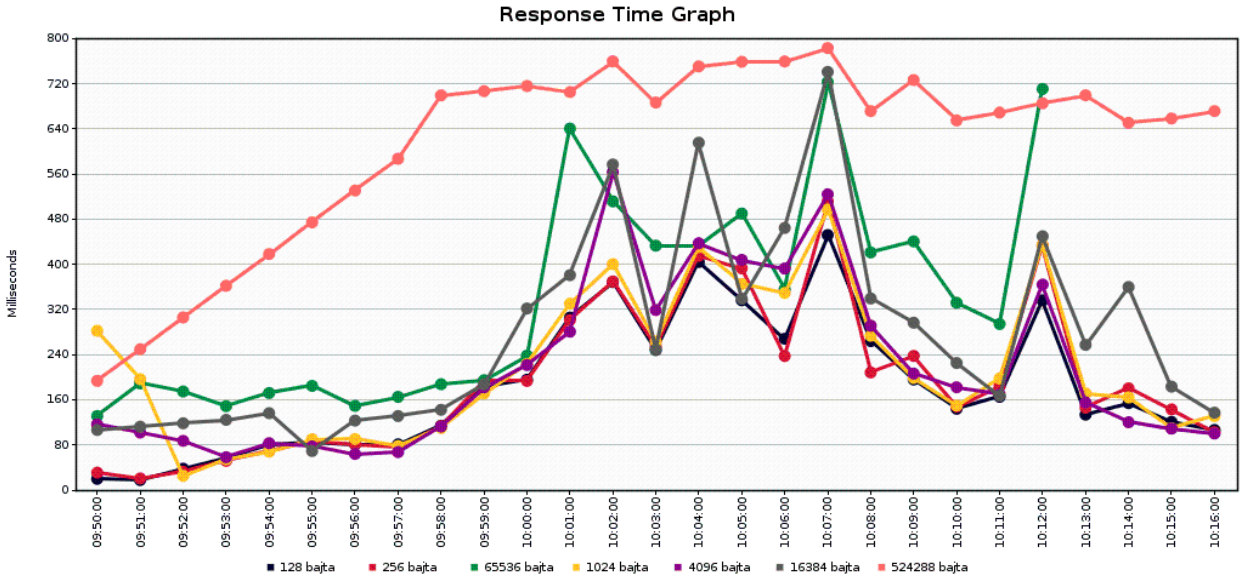
## 6.4 Duže poruke sa 47 klijenata

Ovo merenje je trajalo oko 26 minuta i za to vreme je poslato 9400 zahteva i isto toliko odgovora. U merenju je učestvovalo 47 modela mobilnih aplikacija, koji su slali zahteve za sadržajem veličine od 128 B do 512 kB. Prosečna razmena je iznosila oko 15 kB, a prosečno vreme do pristizanja odgovora je **254 ms**.

| Zahtev | Uzorak | Prosečno vreme izvršavanja | Najkraće vreme izvršavanja | Najduže vreme izvršavanja | Standardna devijacija [ms] | Greška [%] | Propu-snost [kom/ | Protok [KB/s] | Prosečna veličina [B] |
|---|---|---|---|---|---|---|---|---|---|

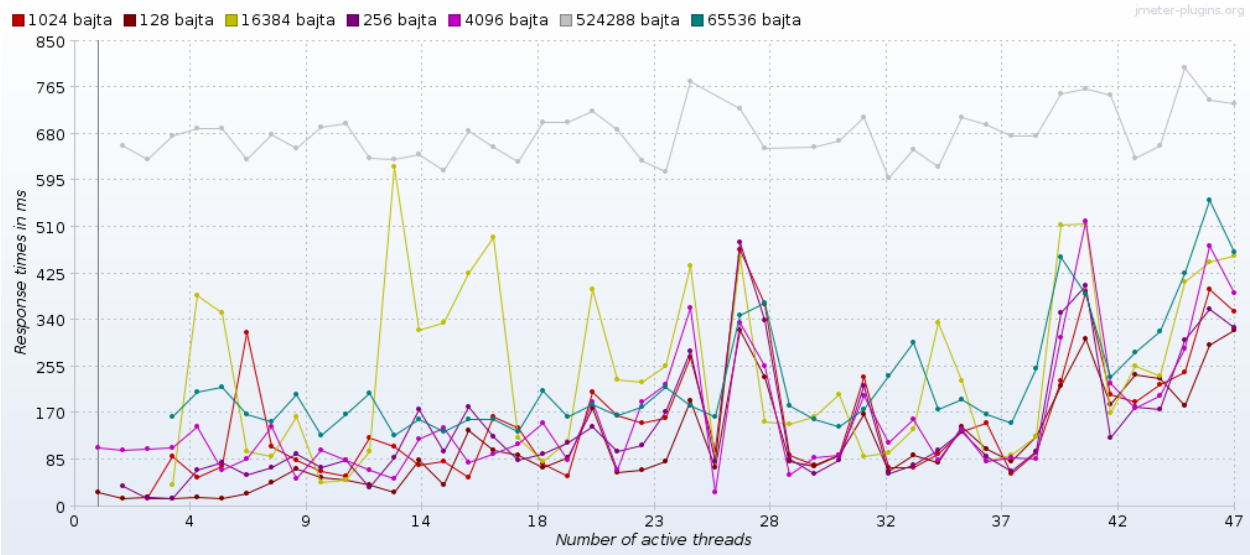
| | | [ms] | [ms] | [ms] | | | s] | | |
|---|---|---|---|---|---|---|---|---|---|
| Uspostava veze | 47 | 162 | 114 | 407 | 54.68 | 0.00 | 0.1 | 0 | 0 |
| 128 bajta | 3800 | 221 | 11 | 996 | 254.59 | 0.00 | 2.4 | 0.55 | 232.5 |
| 256 bajta | 2400 | 238 | 11 | 994 | 270.42 | 0.00 | 1.5 | 0.53 | 360.5 |
| 1024 bajta | 1200 | 252 | 12 | 999 | 271.26 | 0.00 | 0.8 | 0.87 | 1128.5 |
| 4096 bajta | 800 | 268 | 16 | 999 | 281.15 | 0.00 | 0.5 | 2.19 | 4200.5 |
| 16384 bajta | 600 | 334 | 31 | 1000 | 294.43 | 0.00 | 0.4 | 6.87 | 16488.5 |
| 65536 bajta | 400 | 338 | 85 | 943 | 258.88 | 0.00 | 0.3 | 20.12 | 65640.5 |
| 524288 bajta | 200 | 705 | 553 | 950 | 78.28 | 0.00 | 0.2 | 89.95 | 524394.4 |
| Raskid veze | 47 | 1 | 1 | 5 | 0.87 | 0.00 | 0.1 | 0 | 0 |
| UKUPNO | 9494 | 254 | 1 | 1000 | 273.31 | 0.00 | 5.7 | 87.04 | 15535.2 |

Tabela 18- Rezultati za duže poruke i 47 klijenata




Slika 14- Kretanje vremena odgovora za duže poruke i 47 klijenata

I u ovom slučaju su korisnici postepeno pokretani u roku od 10 minuta. Razlika pri prisustvu većeg broja klijenata je izraženija, u poređenju sa slanjem kraćih poruka.




Slika 15- Vreme odgovora po broju klijenata za duže poruke i 47 klijenata

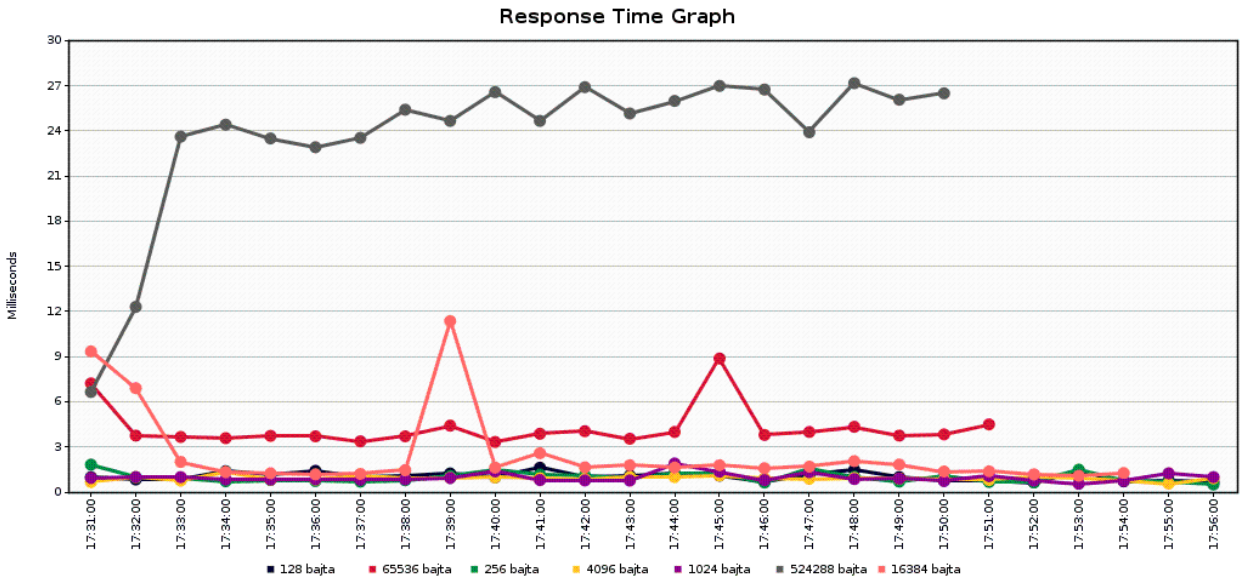
## 6.5 Duže poruke sa 47 IPC klijenata

Pri ovom merenju i model modula centralnog kontrolera je bio pokrenut na PC računaru, sa ciljem da se oponaša razmena podataka između procesa jednog računara. Merenje je trajalo oko 25 minuta i za to vreme je poslato 9400 zahteva i isto toliko odgovora. U merenju je učestvovalo 47 modela mobilnih aplikacija, pokrenutih na istom računaru na kom je pokrenut i model modula centralnog kontrolera. Zahtevi za sadržajem su se kretali od 128 B do 512 kB. Prosečna razmena je iznosila oko 15kB, a prosečno vreme do pristizanja odgovora je **1 ms**.

| Zahtev | Uzorak | Prosečno vreme izvršavanja [ms] | Najkraće vreme izvršavanja [ms] | Najduže vreme izvršavanja [ms] | Standardna devijacija [ms] | Greška [%] | Propusnost [kom/s] | Protok [KB/s] | Prosečna veličina [B] |
|---|---|---|---|---|---|---|---|---|---|
| Uspostava veze | 47 | 9 | 3 | 84 | 11.72 | 0.00 | 0.1 | 0 | 0 |
| 128 bajta | 3800 | 1 | 0 | 46 | 3.44 | 0.00 | 2.4 | 0.54 | 231.5 |
| 256 bajta | 2400 | 1 | 0 | 101 | 3.47 | 0.00 | 1.6 | 0.55 | 359.5 |
| 1024 bajta | 1200 | 0 | 0 | 43 | 2.68 | 0.00 | 0.9 | 0.94 | 1127.5 |
| 4096 bajta | 800 | 0 | 0 | 9 | 0.64 | 0.00 | 0.5 | 2.2 | 4199.5 |
| 16384 bajta | 600 | 2 | 1 | 303 | 12.55 | 0.00 | 0.5 | 7.64 | 16487.4 |
| 65536 bajta | 400 | 4 | 3 | 129 | 6.6 | 0.00 | 0.3 | 21.34 | 65639.5 |
| 524288 bajta | 200 | 25 | 21 | 37 | 3.52 | 0.00 | 0.2 | 102.18 | 524393.4 |
| Raskid veze | 47 | 1 | 0 | 4 | 0.86 | 0.00 | 0.1 | 0 | 0 |
| UKUPNO | 9494 | 1 | 0 | 303 | 5.84 | 0.00 | 6 | 90.67 | 15534.2 |

Tabela 19- Rezultati za duže poruke i 47 IPC klijenata

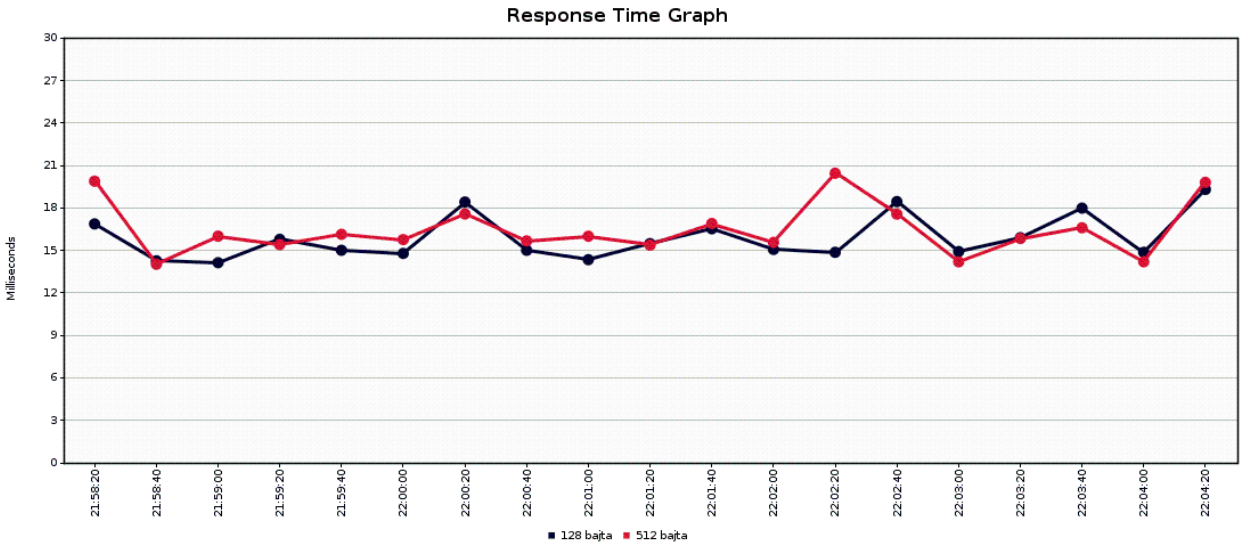
Vreme odgovora je za sve vreme merenja bilo gotovo istovetno, sa standardnom devijacijom od 5,84 ms. Varijacija broja klijenata koji su istovremeno bili aktivni, u ovom slučaju, gotovo da nije imalo uticaja.



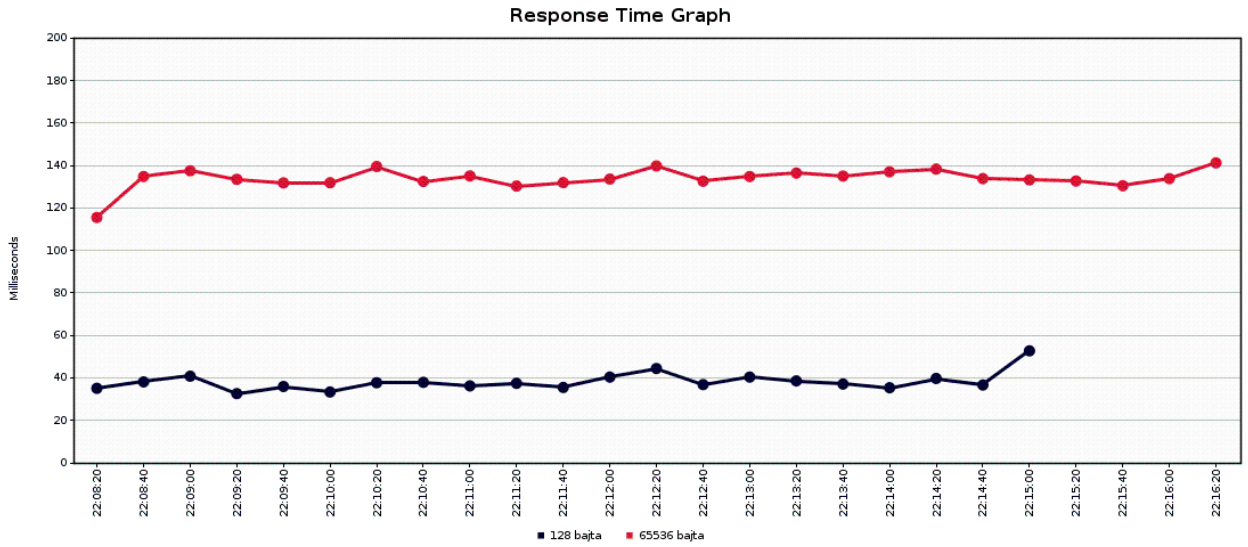

Slika 16- Vreme odgovora za duže poruke i 47 IPC klijenata

## 6.6 Poređenje dve veličine poruka

Merenje za razmenu između klijenata na PC računaru i modela modula na RaspberryPi platformi je izvršeno sa pauzom smanjenom na oko 400 ms i na uzorku od 1000 razmena po jednoj veličinu. Pri slanju odgovora sa 128 i 512 bajta podataka se dobija veoma slično vreme, dok je razlika mnogo izraženija u slučaju 128 i 65536 bajta.

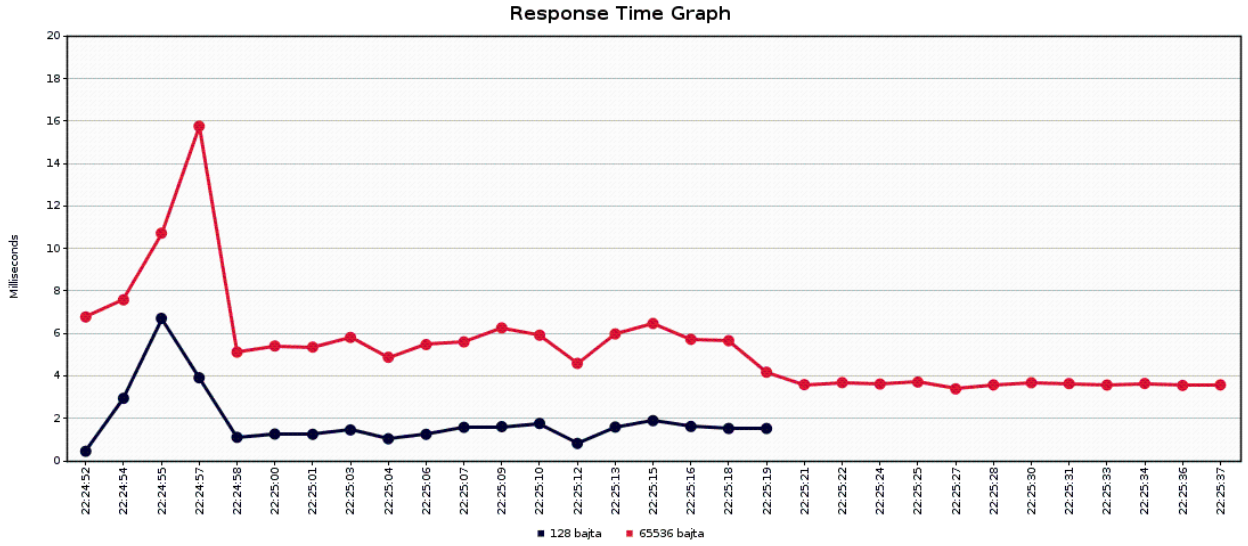



Slika 17- Vreme odgovora pri zahtevu za 128 i 512 bajta podataka




Slika 18- Vreme odgovora pri zahtevu za 128 i 65536 bajta podataka

Merenje razmene između procesa istog računara je izvršeno bez pauze i na uzorku od 5000 razmena po jednoj veličinu. Učinak pri razmeni između procesa i razmeni u lokalnoj mreži se razlikuje za približno 20 puta.




Slika 19- Vreme odgovora pri IPC zahtevu za 128 i 65536 bajta podataka

## 6.7 Merenje protoka

U cilju merenja protoka, iz postavke je isključena pauza od 5 sekundi, korišćena za merenje vremena odgovora. 47 modela korisničkih aplikacija je besprekidno slalo zahteve na model modula centralnog kontrolera pokrenutog na RaspberryPi platformi. Prosečna veličina razmene je oko 15 kB. Ovim merenjem je postignut protok od ~710 kB/s.

| Zahtev | Uzorak | Prosečno vreme izvršavanja [ms] | Najkraće vreme izvršavanja [ms] | Najduže vreme izvršavanja [ms] | Standardna devijacija [ms] | Greška [%] | Propu-snost [kom/s] | Protok [KB/s] | Prosečna veličina [B] |
|---|---|---|---|---|---|---|---|---|---|

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Uspostava veze | 47 | 416 | 114 | 1054 | 253.15 | 0.00 | 4.7 | 0 | 0 |
| 128 bajta | 3800 | 958 | 23 | 1755 | 135.72 | 0.00 | 18.7 | 4.26 | 232.5 |
| 256 bajta | 2400 | 959 | 79 | 1750 | 127.2 | 0.00 | 11.9 | 4.18 | 360.5 |
| 4096 bajta | 800 | 961 | 54 | 1667 | 123.4 | 0.00 | 4 | 16.3 | 4200.5 |
| 1024 bajta | 1200 | 964 | 267 | 1760 | 118.51 | 0.00 | 6 | 6.58 | 1128.5 |
| 16384 bajta | 600 | 973 | 402 | 1739 | 112.43 | 0.00 | 3 | 48.45 | 16488.5 |
| 65536 bajta | 400 | 968 | 106 | 1915 | 161.46 | 0.00 | 2 | 127.08 | 65640.7 |
| 524288 bajta | 200 | 989 | 653 | 1807 | 101.09 | 0.00 | 1 | 511.51 | 524394.4 |
| Prekid veze | 47 | 1 | 0 | 6 | 1.54 | 0.00 | 2.4 | 0 | 0 |
| UKUPNO | 9494 | 954 | 0 | 1915 | 151.55 | 0.00 | 46.8 | 709.93 | 15535.3 |

Tabela 20- Rezultati merenja protoka podataka

# 7. Zaključak

Uzimajući u obzir ponašanja različitih tipova uređaja u srednjem sloju komercijalnog sistema za pametne kuće, definisan je protokol koji pokriva sve aspekte zahtevanog načina razmene poruka. Asinhrona komunikacije definisana MQTT protokolom je iskorišćena u svom osnovnom obliku za poruke obaveštenja, ali se adekvatnim sagledavanjem mogućnosti došlo do efikasnog rešenja za razmenu koja izlazi iz okvira MQTT protokola: razmena poruka po modelu zahtev-odgovor. Korišćenjem realizacija koje su dostupne za MQTT protokol (klijenti i broker), u mnogome je ubrzana realizacija željenog komunikacionog sistema. Takođe, kako se te realizacije konstantno upotrebljava i održava od strane velikog broja članova zajednice otvorenog koda, stabilnost se iz dana u dan poboljšava. Dodatno su realizovane specifičnosti protokola, ali pri tome se vodilo računa da se baza (MQTT) ne menja, već da se na osnovu nje gradi ono što je potrebno.

Definisan je način za prepoznavanje učesnika pomoću *oznaka*, kao i univerzalni *tipovi razmene* zahteva, odgovora i obaveštenja. To su osnovni elementi protokola i dovoljni su za pravilnu upotrebu. Sama felksibilnost se ogleda u mogućnosti prosleđivanje različitih parametara unutar sadržaja poruka. Dodatno, primenjeno je pravilo po kom je naziv grupe sastavni deo oznake. Tako su predstavljeni primeri grupa za module centralnih kontrolera, korisničke mobilne aplikacije i udaljene servise. Upravo takav pristup je olakšao kontrolu ponašanja klijenata. Ona se obavlja na osnovu oznake i to u najvećoj meri uz korišnjenje naziva grupe kojoj klijent pripada. Time je definisan protokol koji pokriva ponašanje i zahteve različitih uređaja na jedinstven način.

Zaključeno je da se jedinstvenost klijenta može potvrditi samo u slučaju korišćenja lozinki koje su karakteristične za svakog od njih. Zbog toga je obezbeđena procedura za registraciju klijenta pri kojoj se te lozinke dodeljuju.

Veoma dobre performanse su pokazane kroz niz merenja, u skladu sa zahtevima i upotrebom koja se očekuje od ovakvog sistema. Pored osnovne upotrebe u lokalnoj mreži, više nego dobri su rezultati koji su postignuti u razmenama između procesa istog računara. To merenje je pokazalo da se razmena zahtev-odgovor od oko 15kB podataka, između dva klijenta (jedan realizovan u C++, a drugi u Java programskom jeziku), može obaviti za 1ms. Ovo dozvoljava i nameće upotrebu predstavljenog rešenja u IPC svrhe.

U lokalnoj mreži, pri merenju obavljenom na RaspberryPi platformi, je dostignut protok podataka od približno 710kB/s, što je gotovo 3 puta bolji rezultat od onog koji se u meš mrežama može postići korišćenjem 1 kanala (256kB/s). Treba napomenuti da je merenje obavljeno na osnovu očekivane, realne upotrebe pri intenzivnim razmenama u sistemima pametnih kuća i da se nije dostigao teorijski maksimum protoka. Očekuje se da će pri direktnom poređenju sa nekim od meš protokola rezultati biti više nego 3 puta bolji.

# 8. Literatura


[1] Akyildiz, Ian F., Xudong Wang i Weilin Wang. "Wireless mesh networks: a survey." Computer networks 47.4 (2005): 445-487.

[2] Baronti, Paolo i ostali "Wireless sensor networks: A survey on the state of the art and the 802.15. 4 and ZigBee standards." Computer communications 30.7 (2007): 1655-1695.

[3] Gomez, Carles i Josep Paradells. "Wireless home automation networks: A survey of architectures and technologies." IEEE Communications Magazine 48.6 (2010): 92-101.

[4] Kim, Su Min i ostali "Experiments on Interference and Coexistence between Zigbee and WLAN Devices Operating in the 2.4 GHz ISM band." Proc. NGPC (2005): 15-19.

[5] Raniwala, Ashish, Kartik Gopalan i Tzi-cker Chiueh. "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks." ACM SIGMOBILE Mobile Computing and Communications Review 8.2 (2004): 50-65.

[6] Milan Tucić, "IoT proširenje za povezivanje sistema za automatizaciju kuće korišćenjem M2M protokola", Završni (Bachelor) rad.

[7] Banks, Andrew i Rahul Gupta. "MQTT Version 3.1. 1." OASIS Standard (2014).

[8] I. Bašićević, M. Popović, V. Kovačević, "Osnovi računarskih mreža 1", FTN, 2013.

[9] Vinoski, Steve. "Advanced message queuing protocol." IEEE Internet Computing 6 (2006): 87-89.

[10] Version, M. Q. T. T. "3.1. 1. Edited by Andrew Banks and Rahul Gupta. 12 December 2013. OASIS Committee Specification Draft 01/Public Review Draft 01."

[11] Mosquitto- http://mosquitto.org/, posećeno 30.05.2016.

[12] Nevedrov, Dmitri. "Using JMeter to Performance Test Web Services." Published on dev2dev (http://dev2dev. bea. com/) (2006).

[13] Chopra, Vivek, Sing Li, and Jeff Genender. Professional Apache Tomcat 6. John Wiley & Sons, 2007.