

## ANALIZA PERFORMANSI HTTP SAOBRAĆAJA

Bojan Bakmaz<sup>1</sup>, Vuk Perović<sup>2</sup>, Zoran Bojković<sup>3</sup>, Miodrag Bakmaz<sup>4</sup>  
<sup>1</sup>Univerzitet u Beogradu - Saobraćajni fakultet, b.bakmaz@sf.bg.ac.rs  
<sup>2</sup>Endava d.o.o, Beograd, vuk.perovic@outlook.com  
<sup>3</sup>Univerzitet u Beogradu, z.bojkovic@yahoo.com  
<sup>4</sup>Univerzitet u Beogradu - Saobraćajni fakultet, bakmaz@sf.bg.ac.rs

**Rezime:** *Povećanje kompleksnosti web servisa pratilo je unapređenje HTTP-a, kao jednog od najzastupljenijih aplikativnih protokola. Za većinu aplikacija i kod velikog broja servera već je implementirana podrška za HTTP verzije 2, sa ciljem unapređenja performansi u domenu propusnosti i bezbednosti. Međutim, u realnom okruženju HTTP/2 ne obezbeđuje uvek očekivana svojstva. U radu su analizirane performanse saobraćaja generisanog preko HTTP/1.1 i HTTP/2 primenom Webpagetest alata, na primeru demo i realne web stranice. Pokazano je da struktura web stranice određuje i njene performanse, tako da se u slučaju demo stranice ostvaruje značajno manje vreme odziva primenom HTTP/2. U specifičnim mrežnim okruženjima, sa izraženim kašnjenjem i značajnim gubicima paketa, HTTP/2 ispoljava lošije performanse od očekivanih. Na osnovu dobijenih rezultata i aktuelnih istraživanja ukazano je na mogućnosti i potrebe daljeg unapređenja ovog protokola.*

**Ključne reči:** *Blokiranje zahteva, HTTP saobraćaj, vreme odziva, web servisi.*

### 1. Uvod

Sa povećanjem dostupnosti i funkcionalnosti web servisa raste i njihova kompleksnost. Tradicionalni protokoli, poput TCP (*Transport Control Protocol*) i HTTP (*HyperText Transfer Protocol*) nisu u mogućnosti da odgovore na sve kompleksnije zahteve korisnika. U proteklih dvadesetak godina HTTP je evoluirao od jednostavnog protokola za pristup web sadržajima do primarnog rešenja za distribuciju video servisa u heterogenim mrežnim okruženjima [1] i dominantnog izvora Internet saobraćaja. Zbog toga analiza ovog tipa saobraćaja dobija na značaju, sa ciljem unapređenja kvaliteta servisa, energetske efikasnosti i bezbednosti.

Inicijalne verzije ovog protokola pokazale su se kao neadekvatno rešenje za savremene web aplikacije, sa aspekta iskorišćenosti resursa TCP konekcije i problematike izraženog blokiranja zahteva [2]. Aktuelna verzija (HTTP/2) delimično prevazilazi pomenuta ograničenja, međutim i dalje ispoljava degradirane performanse, naročito u mrežnim okruženjima sa izraženim kašnjenjem i gubicima paketa [3,4].

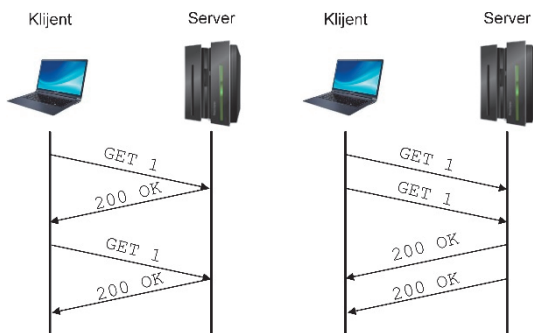
Nakon uvoda, u drugom delu rada prikazane su karakteristike HTTP-a, sa osvrtom na *pipelining* proceduru i problem blokiranja zahteva. Posebna pažnja posvećena je drugoj,

aktuelnoj verziji ovog protokola. Sagledani su aspekti bezbednosti i energetske efikasnosti kao neizostavnih komponenti svakog savremenog protokola. U trećem delu izvršena je analiza HTTP saobraćaja primenom alata *Webpagetest*. Izvršeno je poređenje performansi HTTP saobraćaja koji generišu verzije 1.1 i 2 na primerima demo i realne *web* stranice. Na osnovu dobijenih rezultata i aktuelnih istraživanja u četvrtom delu ukazano je na mogućnosti unapređenja HTTP/2, kao i na koncepte razvoja novih verzija ovog protokola.

## 2. Karakteristike HTTP-a

HTTP predstavlja najdominantniji aplikativni protokol i ujedno jedan od primarnih izvora (generatora) Internet saobraćaja. Inicijalni HTTP koncipiran je kao prilično jednostavan i direktan protokol, koji je vremenom evoluirao preko više aspekata. Prva verzija ovog protokola [5], uvedena u upotrebu pre više od dvadeset godina, zahtevala je posebnu TCP konekciju za svaki objekat *web* stranice, što je neminovno prouzrokovalo veliko kašnjenje i gubitke.

HTTP/1.1 [6] je opsežniji ASCII protokol, sa više opcija dostupnih za nadogradnju. Na slici 1. prikazan je princip funkcionisanja HTTP/1.1, pri čemu klijent i server uspostavljaju TCP konekciju na osnovu GET zahteva. Svaki pojedinačan zahtev korespondira isključivo sa jednim *web* resursom. Server na zahteve odgovara sekvencijalno, tj. novi zahtev se šalje tek nakon odgovora na prethodni zahtev. U ovom slučaju vreme odziva *web* resursa ekvivalentno je proizvodu broja zahteva i kašnjenja.

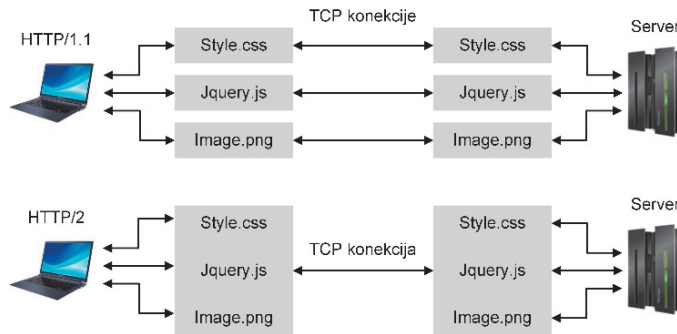


Slika 1. HTTP/1.1 uspostavljanje konekcije: standardna procedura (levo) i pipelining procedura (desno)

Kao potencijalno rešenje problema nametnula se *pipelining* procedura, pri kojoj klijent šalje više sukcesivnih zahteva bez čekanja na odziv servera. Međutim, u ovoj proceduri do izražaja dolazi fenomen HoL (*Head of Line*) blokiranja zahteva, gde prvi zahtev u nizu utiče na kašnjenje svih ostalih zahteva. Jedan od jednostavnijih mehanizama za izbegavanje blokiranja podrazumeva uspostavljanje više istovremenih TCP konekcija. Kao perspektivnije rešenje prihvaćena je nova verzija protokola, HTTP/2 [7].

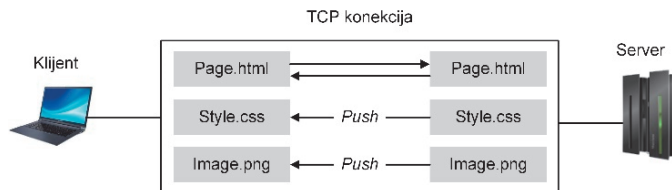
HTTP/2 uvodi nekoliko značajnih funkcija, ujedno zadržavajući semantiku izvornog protokola [8]. Sa ovom verzijom HTTP postaje binarni protokol u cilju pojednostavljenja formiranja okvira. Takođe, uvedena je kompresija zaglavlja primenom HPACK algoritma [9], koji se oslanja na *Huffman*-ovo kodovanje. Sa ciljem otklanjanja problema blokiranja zahteva, HTTP/2 sprovodi tehniku multipleksiranja zahteva, koja je

prikazana na slici 2. Svaki zahtev dodeljuje se posebnom toku, a tokovi se multipleksiraju preko zajedničke TCP konekcije. U ovom slučaju zahtevi nemaju međusobni uticaj, dok server simultano odgovara na njih. Pored rešavanja problema blokiranja zahteva, redukovan je broj TCP konekcija.



Slika 2. Multipleksiranje zahteva

Najznačajnije poboljšanje koje donosi HTTP/2 ogleda se u server *push* funkciji. Ukoliko klijent zahteva određeni resurs, server može predvideti prateće resurse i proslediti ih klijentu bez prethodnih zahteva, kao što je prikazano na slici 3. Ova funkcija uvodi značajnu kompleksnost u procedure podešavanja i održavanja, međutim značajno smanjuje vreme odziva (učitanja stranice).



Slika 3. Server push funkcija

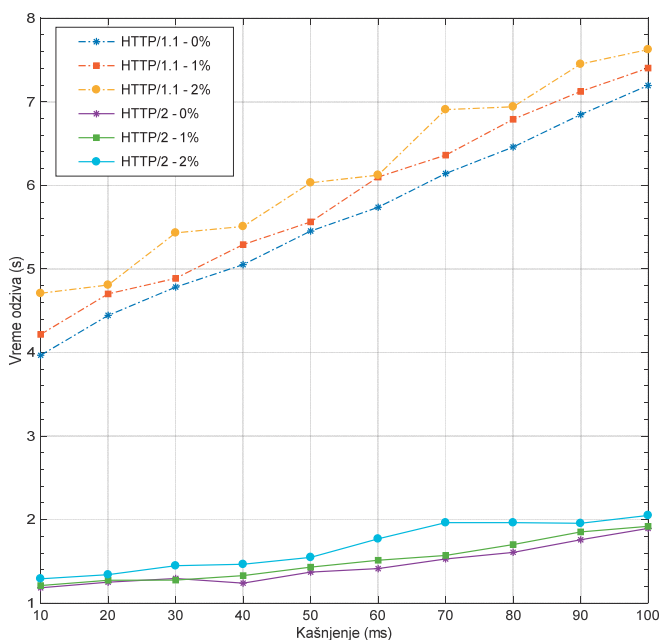
Povećani zahtevi za bezbednošću rezultovali su uvođenjem HTTPS, tj. HTTP preko TLS (*Transport Layer Security*) i u ranijim verzijama protokola. Integracija bezbednosnih funkcija neminovno dovodi do povećanja kašnjenja, kao i redukovanja efikasnosti keširanja u mreži (*in-network caching*) i drugih pratećih servisa. HTTP/2 komunikacija se uglavnom odvija preko TLS ekstenzije, što kao posledicu ima degradaciju performansi, kao i u slučaju primene HTTPS.

Pored bezbednosti, energetska efikasnost takođe predstavlja jedan od bitnih aspekata savremenih komunikacionih mreža. Na osnovu istraživanja sprovedenih u [10], utvrđeno je da postupak kriptovanja na transportnom sloju prouzrokuje veću potrošnju energije nego što čini sam HTTP/1.1. Takođe, HTTP/2 prezentuje slične performanse kao prethodna verzija protokola za male vrednosti RTT (*Round Trip Time*), dok se za veće vrednosti ovog parametra može smatrati energetske efikasnijim protokolom. Na serverskoj strani je slična situacija i do uštede energije dolazi time što server ostvaruje samo jednu TCP konekciju. Ukoliko se koristi server *push* funkcija, može se ostvariti ušteda energije i do 17% [11].

### 3. Analiza HTTP saobraćaja primenom *Webpagetest* alata

Trenutna statistika zastupljenosti HTTP/2 zahteva na Internetu je oko 57% [12], uz primetan konstantan priraštaj, imajući u vidu da su najaktuelniji *web* servisi (*Google*, *Yahoo*, *YouTube*, *eBay*, *Amazon*, *Instagram*, *Twitter*, i sl.) već obezbedili delimičnu ili potpunu podršku za ovaj protokol. Takođe, 90% pretraživača podržava HTTP/2. Osnovna analiza HTTP/2 saobraćaja se može izvršiti korišćenjem naprednih razvojnih opcija u samom pretraživaču (*Developertools* za *Google Chrome* i *WebDeveloper* u okviru *Firefox* pretraživača).

Za kompleksniju analizu performansi HTTP saobraćaja može se primeniti *Webpagetest* alat [13], koji pruža niz mogućnosti testiranja vremena odziva *web* sadržaja, korišćenjem različitih geografskih lokacija i mrežnih okruženja (DSL, 3G, LTE, itd.). U ovom istraživanju za analizu saobraćaja korišćena je demonstraciona stranica <<https://http2.golang.org/gophertiles>>. Stranica se sastoji od slike *Gophera*, maskote Go programskog jezika, podeljene u 180 pojedinačnih segmenata. Za svaki segment šalje se GET zahtev. Pored stranice sa podrškom za HTTP/2, postoji identična stranica koja podržava prethodnu verziju protokola <<https://http1.golang.org/gophertiles>>. Analizirano je vreme odziva obe stranice pri kašnjenju koje varira od 10 do 100 ms i gubicima paketa od 0%, 1%, i 2%. Kao relevantna korišćena je metrika *DocumentComplete* koja ne uključuje vreme izvršavanja *Java* skripti. Rezultati merenja prikazani su na slici 4.

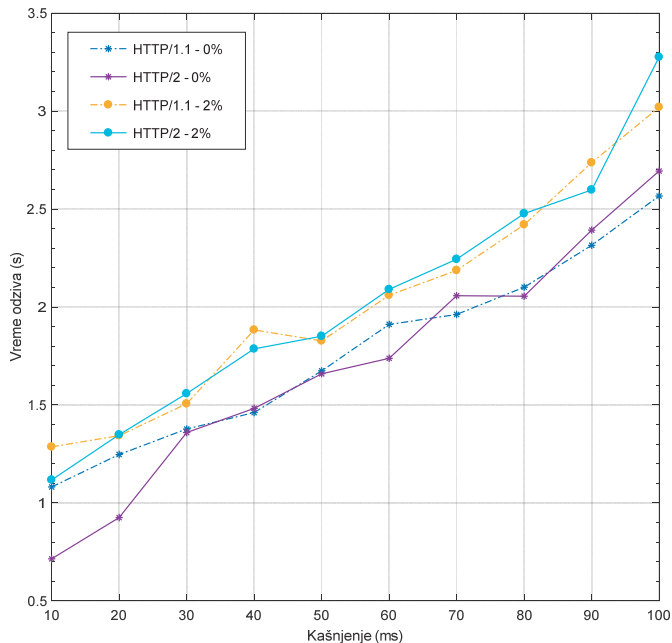


Slika 4. Zavisnost vremena odziva od kašnjenja i gubitaka paketa u slučaju demonstracione stranice

U ovom slučaju HTTP/2 demonstrira značajno bolje performanse zbog sposobnosti multiplexiranja više zahteva po jednoj TCP konekciji u odnosu na HTTP/1.1,

koji ima ograničenje od 6 TCP konekcija po klijentu. Pošto se struktura demo stranica *Gophertiles* sastoji od 180 malih slika, ova prednost dolazi do izražaja. Takođe, uočeno je da gubitak paketa ima manji uticaj na varijacije vremena odziva kod HTTP/2 saobraćaja.

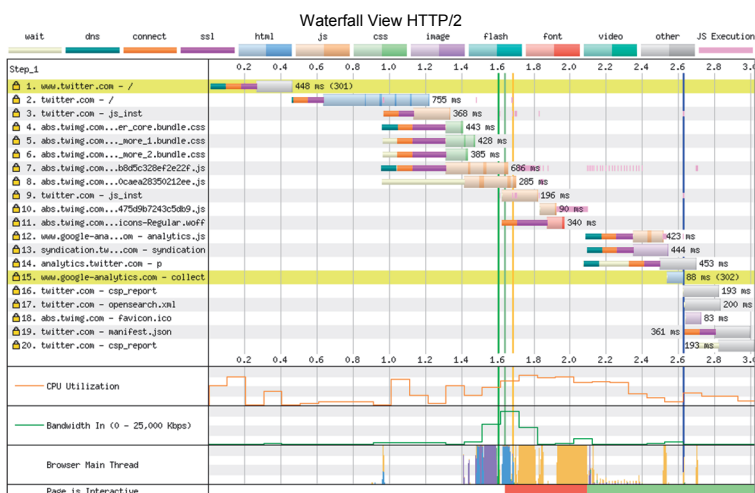
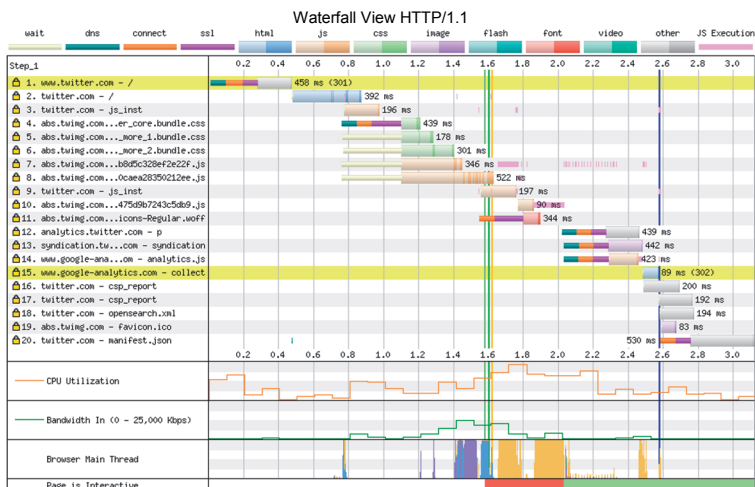
U drugom slučaju korišćena je stranica <www.twitter.com> kao reprezentativni primer realnog okruženja, imajući u vidu broj korisnika. Takođe, analizirano je vreme odziva za HTTP/1.1 i HTTP/2 pri kašnjenju koje varira od 10 do 100 ms i gubicima paketa od 0% i 2%. Isključivanje podrške za HTTP/2 moguće je izvršavanjem komandne linije – `-disable-http2`. Rezultati merenja vremena odziva realne *web* stranice prikazani su na slici 5. U ovom slučaju prednost HTTP/2 u odnosu na HTTP/1.1 nije izražena, a za veća kašnjenja stariji protokol pokazuje čak i bolje performanse.



Slika 5. Zavisnost vremena odziva od kašnjenja i gubitaka paketa u slučaju realne stranice

### 3.1. Analiza HTTP saobraćaja na nivou zahteva

Analiza na nivou zahteva pruža uvid u uticaj strukture *web* sadržaja na HTTP saobraćaj. Na slici 6, moguće je uočiti da u slučaju HTTP/1.1 saobraćaja klijent grupiše više zahteva i prosleđuje ih istovremeno, čekajući odgovor pre nego što započne slanje novih, jer postoji ograničenje od strane pretraživača od šest istovremenih konekcija po hostu. Takođe, može se primetiti da je pretraživač keširao odgovor DNS-a (*Domain Name System*) i iskoristio postojeću enkriptovanu konekciju TLS za slanje zahteva sa rednim brojevima od 5 do 8. Na primeru realne stranice ne uočava se značajna razlika u redosledu slanja zahteva zbog same strukture stranice, te su i vremena učitavanja približna za obe vrste saobraćaja. Analiza na nivou zahteva takođe obezbeđuje identifikaciju prioriternih zahteva.



Slika 6. Analiza HTTP saobraćaja na nivou zahteva

### 3.2. Analiza HTTP saobraćaja na nivou konekcije

Analiza na nivou konekcije omogućuje pregled TCP konekcija ka serverima. U slučaju HTTP/2 saobraćaja to je uglavnom jedna konekcija ka jednom hostu (Slika 7). U datom primeru za HTTP/1.1 uočava se više konekcija ka <www.twitter.com> i <abs.twimg.com> (poddomen izvorne stranice), dok je broj TCP konekcija u slučaju HTTP/2 saobraćaja manji, jer se multipleksiranje vrši nad tokovima podataka u okviru jedne TCP konekcije. Kako je broj konekcija koje se ostvaruju manji od 6 po istom hostu, ne dolazi do značajnije prednosti primenom tehnike multipleksiranja. Ukoliko bi struktura stranice bila takva da pretraživač mora da uputi više od 6 zahteva ka istom hostu istovremeno, došlo bi do blokiranja slanja sedmog zahteva dok se prvi ne završi. Sve

navedeno bi rezultovalo dužim vremenom učitavanja stranice. Može se zaključiti da zbog strukture razmatrane stranice prednosti novije verzije protokola ne dolaze do izražaja.

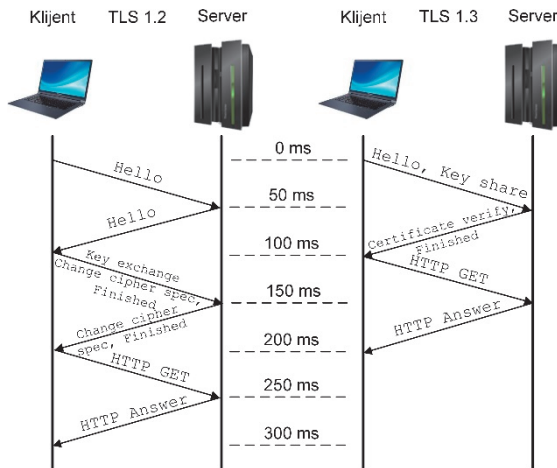


Slika 7. Analiza HTTP saobraćaja na nivou konekcije

#### 4. Mogućnosti unapređenja HTTP

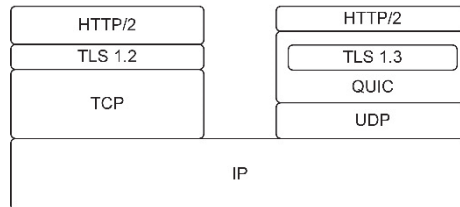
Aktuelna istraživanja koja se odnose na mogućnosti unapređenja HTTP/2 usmerena su ka optimizaciji TCP i TLS protokola, kao i na razvoj nove verzije samog aplikativnog protokola. TCP *Fast Open* (TFO) [14] mehanizam omogućuje prenos podataka u okviru SYN i SYN-ACK paketa, obezbeđujući raspoloživost korisničkog sadržaja u početnoj fazi uspostave konekcije. Ovim se postiže ušteda u vidu jednog RTT, u poređenju sa standardnom TCP procedurom. Međutim, TFO odstupa od standardne TCP semantike zbog toga što podaci iz SYN paketa mogu biti preneti do aplikacije u specifičnim okolnostima.

Najnovija verzija TLS protokola, 1.3 [15], predstavljena je kao rešenje za neke od uočenih problema, koji se odnose na brzinu uspostavljanja veze, privatnost i bezbednost. Pored unapređene enkripcije, izvršena je i redukcija broja potrebnih RTT za uspostavljanje bezbedne konekcije, sa dva na jedan (tzv. 0-RTT pristup), čime se postižu kraća vremena odziva (Slika 8). Ova osobina je od velikog značaja kod ponavljanja zahteva, pošto pretraživač pamti informacije o bezbednom sadržaju.



Slika 8. Uspostavljanje veze preko TLS

Pored pomenutih unapređenja, radi se i na uvođenju nove verzije HTTP, čiji je razvoj započeo u okviru *Chromium* projekta QUIC (*Quick UDP Internet Connections*), a koja predviđa primenu UDP umesto TCP na transportnom sloju [16]. Modifikacija HTTP steka prikazana je na slici 9.



Slika 9. Modifikacija HTTP steka

Poboljšanja koja donosi QUIC, uglavnom se odnose na smanjeno vreme uspostavljanja veze, unapređenu kontrolu zagušenja, korekciju grešaka, multipleksiranje bez mogućnosti HoL blokiranja, itd. Pošto je enkapsuliran u okviru UDP, QUIC izbegava zavisnost od proizvođača mrežne opreme i operatora, čime se otvaraju mogućnosti za bržu implementaciju preko izmena u samoj aplikaciji. Obezbeđujući pouzdanost na nivou toka i kontrolu zagušenja na nivou konekcije, QUIC može značajno da unapredi performanse HTTP saobraćaja.

## 5. Zaključak

Analiza saobraćaja uvek predstavlja dobru osnovu za unapređenje performansi mreža, protokola i optimizaciju korisničkog sadržaja. U ovom radu prikazan je okvir za analizu HTTP saobraćaja u cilju daljeg unapređenja ovog najzastupljenijeg aplikativnog protokola. Kao osnovna metrika korišćeno je vreme odziva *web* sadržaja, a kao uticajni parametri razmatrani su kašnjenje i gubici paketa. U slučaju demonstracionog izvora



saobraćaja potvrđene su bolje performanse novije verzije protokola zbog njegove multipleksne prirode pri obradi velikog broja simultanih zahteva. Takođe, utvrđen je uticaj strukture *web* sadržaja na performanse HTTP saobraćaja. Međutim, pri realnom saobraćaju razlika između analiziranih sukcesivnih verzija protokola nije toliko značajna. U specifičnim mrežnim okruženjima novija verzija protokola ne pokazuje očekivane performanse. Moguće objašnjenje odnosi se na činjenicu da se sa povećanim gubicima paketa gube podaci iz više tokova, što prouzrokuje kašnjenje više zahteva istovremeno.

U radu su takođe sagledane mogućnosti optimizacije HTTP/2 saobraćaja primenom TCP *FastOpen* specifikacije, gde je moguće napraviti uštedu u vidu jednog RTT, koristeći sinhronizacione pakete za prenos podataka u ranoj fazi uspostavljanja konekcije. Još jedna od mogućih optimizacija je korišćenje funkcionalnosti 0-RTT u okviru TLS 1.3 protokola za kriptu zaštitu na transportnom sloju, gde se klijentu omogućuje da identifikuje bezbedne konekcije za kasniju upotrebu. Perspektiva HTTP protokola zasniiva se na izmenama u okviru transportnog sloja uvođenjem UDP, čime se obezbeđuje kontrola transporta na nivou aplikativnog protokola. Nagovešteno je da QUIC protokol pokazuje bolje performanse u okruženjima sa izraženim kašnjenjem, kao što su mobilne mreže. Jedan od pravaca daljih istraživanja podrazumeva analizu energetske efikasnosti, jer naprednije verzije protokola angažuju znatno više procesorskih resursa.

## Literatura

- [1] N. Barman, M. G. Martini, "QoE Modeling for HTTP Adaptive Video Streaming—A Survey and Open Challenges," *IEEE Access*, vol. 7, pp. 30831-30859, 2019.
- [2] E. Casilari, F. J. González and F. Sandoval, "Modeling of HTTP Traffic," *IEEE Communications Letters*, vol. 5, no. 6, pp. 272-274, June 2001.
- [3] N. Oda, S. Yamaguchi, "HTTP/2 Performance Evaluation with Latency and Packet Losses", *Proc. 15th IEEE Annual Consumer Communications & Networking Conference (CCNC 2018)*, Las Vegas, NV, Jan. 2018, pp. 1-2.
- [4] J. Min, Y. Lee, "An Experimental View on Fairness between HTTP/1.1 and HTTP/2", *Proc. The 33<sup>rd</sup> International Conference on Information Networking (ICOIN 2019)*, Kuala Lumpur, Malaysia, Jan./Feb. 2019, pp. 399-401.
- [5] T. Berners-Lee, R. Fielding, H. Frystyk, "Hypertext Transfer Protocol - HTTP/1.0", *IETF RFC 1945*, May 1996.
- [6] R. Fielding, et al., "Hypertext Transfer Protocol - HTTP/1.1", *IETF RFC 2616*, June 1999.
- [7] M. Belshe, R. Peon, M. Thomson, "Hypertext Transfer Protocol Version 2 (HTTP/2)", *IETF RFC 7540*, May 2015.
- [8] H. de Saxcé, I. Opreescu, Y. Chen, "Is HTTP/2 Really Faster Than HTTP/1.1?", *Proc. IEEE Conference on Computer Communications Workshops (INFOCOM)*, Hong Kong, Apr./May 2015, pp. 293-299.
- [9] R. Peon, H. Ruellan, "HPACK: Header Compression for HTTP/2", *IETF 7541*, May 2015.
- [10] S. A. Chowdhury, V. Sapra, A. Hindle, "Client-Side Energy Efficiency of HTTP/2 for Web and Mobile App Developers", *Proc. IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, Osaka, Japan, Mar. 2016, pp. 529-540.

- [11] S. Wei, V. Swaminathan, M. Xiao, "Power Efficient Mobile Video Streaming Using HTTP/2 Server Push", *Proc. IEEE 17th International Workshop on Multimedia Signal Processing (MMSP)*, Xiamen, China, Oct. 2015, pp. 1-6.
- [12] <https://httparchive.org>
- [13] <https://www.webpagetest.org>
- [14] Y. Cheng, J. Chu, A. Jain, "TCP Fast Open", *IETF 7413*, Dec. 2014.
- [15] E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.3", *IETF 8446*, Aug. 2018.
- [16] Y. Cui, et al., "Innovating Transport with QUIC: Design Approaches and Research Challenges", *IEEE Internet Computing*, vol. 21, no. 2, pp. 72-76, Mar.-Apr. 2017.

**Abstract:** *The complexity growth of web services was followed by the evolution of HTTP as one of the most representative application layer protocols. In order to improve performances in the domain of throughput and security, HTTP/2 is already implemented in the great number of servers and applications. However, in the real environment HTTP/2 does not always provide expected properties. The performances of HTTP traffic in the case of the demo end real web page are analyzed using the Webpagetest tool. It is shown that the structure of the web page determines its own performance so that in the case of demo page significantly less load time using HTTP/2 is perceived. In the specific network environments with significant latency and packet losses, HTTP/2 demonstrates poorly performances compared to the expected ones. Taking into consideration the obtained results and the contemporary researches, the possibility of this protocol further improvement is pointed out.*

**Keywords:** *Blocking, HTTP traffic, page load time, web services.*

**PERFORMANCE ANALYSIS OF HTTP TRAFFIC**  
Bojan Bakmaz, Vuk Perovic, Zoran Bojkovic, Miodrag Bakmaz