

# Raspoređivači Opterećenja

---

Predmet: Distribuirani sistemi

Predavač: dr Dušan Stefanović

# RASPOREĐIVAČI OPTEREĆENJA

Raspoređivanje opterećenja (load balancing) je mehanizam za optimalnu raspodelu opterećenja među resursima.

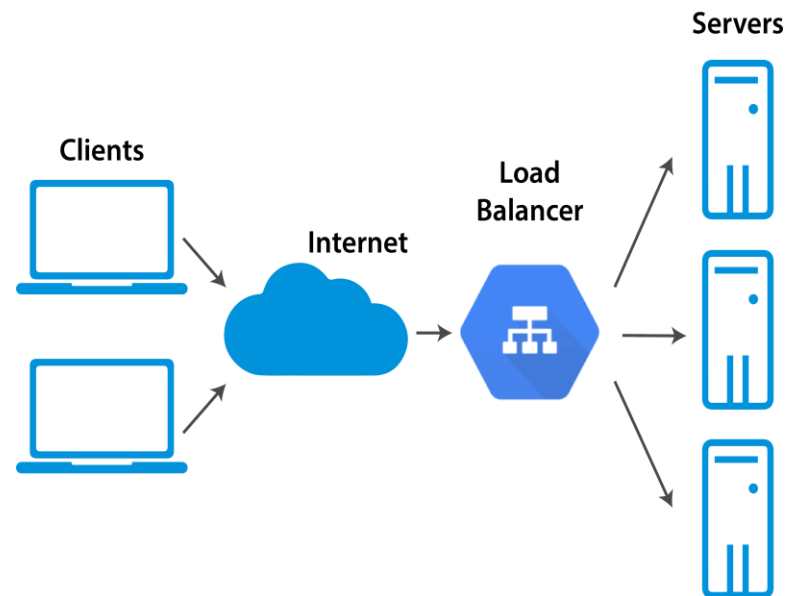
Raspoređivanje opterećenja odnosi se na ravnomernu distribuciju dolaznog mrežnog saobraćaja kroz grupu pozadinskih servera - farma servera ili grupa servera.

Savremene web aplikacije često opslužuju stotine hiljada, neretko i milione, istovremenih zahteva korisnika ili klijenata sa tekstovima, slikama, video zapisima na brz i pouzdan način.

Raspoređivanje opterećenja usmerava klijentske zahteve na sve servere sposobne da ispune zahteve i osigurava da nijedan server nije preopterećen.

Ako jedan od servera postane nedostupan, load balanser preusmerava saobraćaj na preostale servere.

Kada se doda novi server, load balanser automatski počinje da mu šalje zahteve.



# RASPOREĐIVAČI OPTEREĆENJA

Load balanser vrši sledeće funkcije:

- Efikasno distribuira zahteve klijenata ili opterećenje mreže na više servera
- Osigurava visoku dostupnost i pouzdanost slanjem zahteva samo serverima koji su na mreži
- Pruža fleksibilnost dodavanja ili oduzimanja servera prema zahtevima

**load balancing** - sprečava da jedan server postane preopterećen, što bi moglo uzrokovati njegovo usporavanje, ispuštanje zahteva, i eventualno rušenje.

# RASPOREĐIVAČI OPTEREĆENJA - ULOGA

Balansiranje opterećenja obavlja sledeće kritične zadatke:

- Upravlja povećanjem saobraćaja i sprečava skokove na jednom serveru
- Skraćuje vreme odgovora na zahtev korisnika
- Osigurava performanse i pouzdanost računarskih resursa, fizičkih i virtuelnih
- **FLEKSIBILNOST**: balansiranje opterećenja pruža fleksibilnost za dodavanje i uklanjanje servera prema zahtevima - omogućava održavanje servera bez izazivanja smetnji za korisnike jer se tokom održavanja saobraćaj preusmerava na druge servere.
- **SKALABILNOST**: Kako se novi serveri pojavljuju na mreži, uravnoteživač opterećenja ih prepoznaje i neprimetno uključuje u proces. Ovaj pristup je poželjniji od premeštanja veb lokacije sa preopterećenog servera na novi, što često zahteva određenu količinu zastoja.
- **REDUDANTNOST**: U distribuciji saobraćaja preko grupe servera, balansiranje opterećenja obezbeđuje ugrađenu redundantnost.

# UPROŠĆEN NAČIN RADA

---

Balanser opterećenja koristi unapred određeni obrazac - algoritam ili metoda za uravnoteženje opterećenja.

- ✓ Klijent, poput aplikacije ili pretraživača, pokušava da se poveže sa serverom.
- ✓ Balanser opterećenja prima zahtev i, na osnovu unapred podešenih obrazaca algoritma, usmerava zahtev na jedan od servera u grupi servera (ili farmi).
- ✓ Server prima zahtev za povezivanje i odgovara klijentu preko load balansera.
- ✓ Balanser opterećenja prima odgovor pronalazi IP adresu klijenta sa IP adresom izabranog servera. Zatim prosleđuje paket sa odgovorom.
- ✓ Kada je primenljivo, balanser opterećenja upravlja SSL-om, što je proces dešifrovanja podataka pomoću protokola za šifrovanje (Security Socket Layer), tako da serveri to ne moraju da rade.
- ✓ Postupak se ponavlja sve dok sesija ne završi.

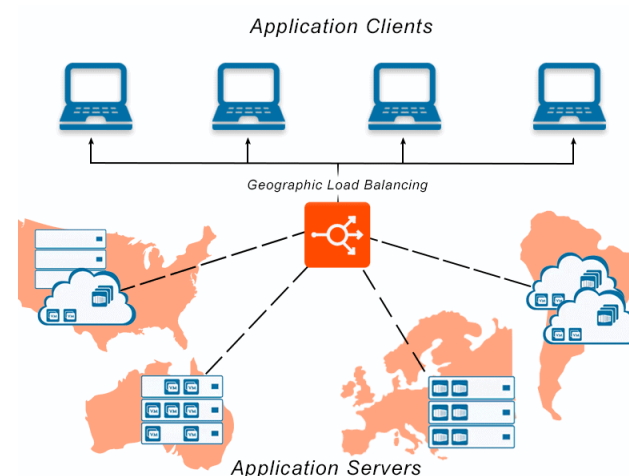
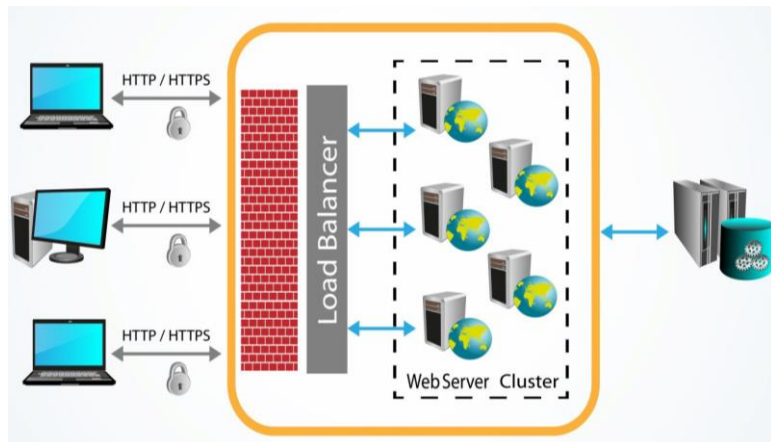
# SERVISI KOJI KORISTE RASPOREĐIVAČI OPTEREĆENJA

Uređaji Load balanseri podržavaju gotovo sve protokole zasnovane na TCP ili UDP, uključujući HTTP, HTTPS, FTP, SMTP, RDP, SIP, IMAP, POP, DNS ...

Balansiranje opterećenja se često koristi na:

- farmama servera na kojima se nalaze web lokacije sa velikim saobraćajem;
- servere baze podataka
- prenosa datoteka (FTP)
- sistema DNS (Domain Name Sistem).

Ravnomernim preusmeravanjem korisničkih zahteva kroz grupu servera, uravnoteživači opterećenja minimiziraju verovatnoću zastoja.

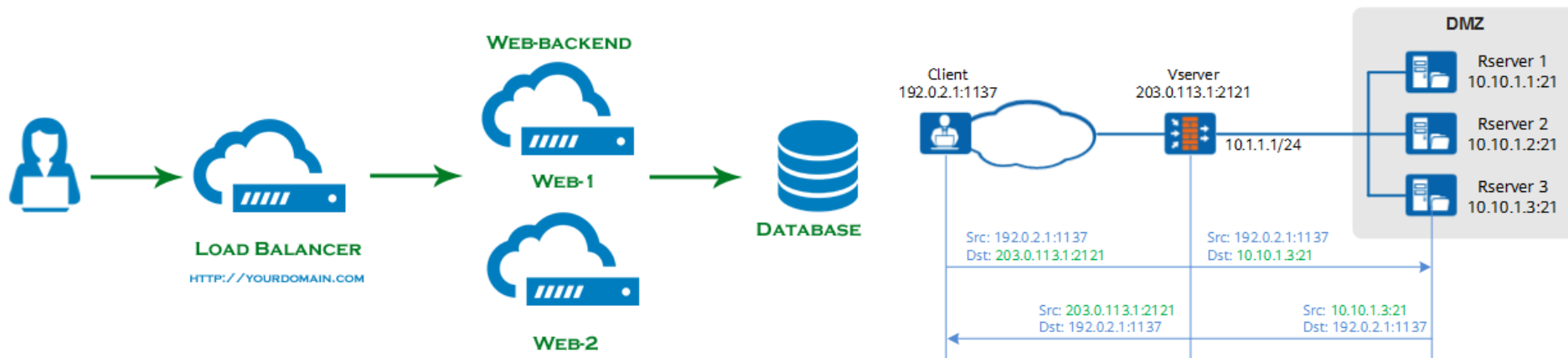


# BALANSIRANJE OPTEREĆENJA SLOJA 4

## Raspoređivači Opterećenja na sloju 4

- koristi informacije definisane na transportnom sloju mreže o načinu distribucije zahteva klijenata po grupi servera.
- zasniva odluku o balansiranju opterećenja na izvornim i odredišnim IP adresama i portovima koji se nalaze u zaglavlju paketa
- ne uzima u obzir sadržaj paketa i često je namenski hardverski uređaj
- zahteva manje računanja od load balancer-a sloja 7, ali su CPU i memorija sada dovoljno brzi i jeftini da su prednosti performansi za Layer 4 postale zanemarljive u većini situacija.

## LAYER 4 LOAD BALANCING

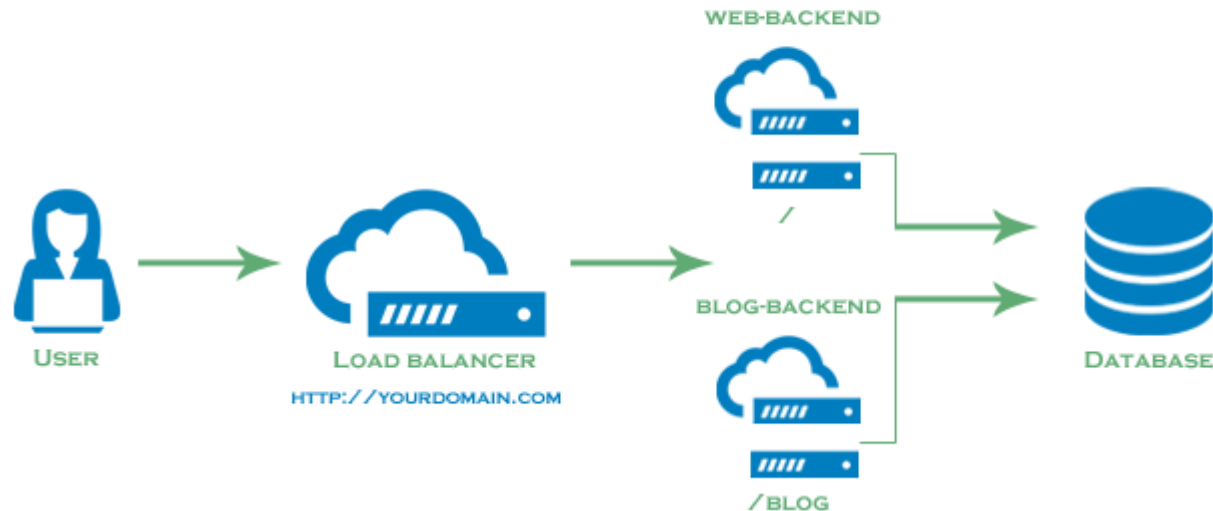


# BALANSIRANJE OPTEREĆENJA SLOJA 7

## Raspoređivači Opterećenja na sloju 7

- Balanseri opterećenja sloja 7 rade na najvišem nivou u OSI modelu, aplikacionom sloju.
- L7 balanseri opterećenja uključuju mnogo širi opseg podataka, uključujući HTTP zaglavlja i SSL sesije.
- Balanseri opterećenja sloja 7 svoje odluke o rutiranju zasnivaju na različitim karakteristikama HTTP zaglavlja i na stvarnom sadržaju poruke, kao što su URL, vrsta podataka (tekst, video, grafika) ili informacije u kolačićima.

## LAYER 7 LOAD BALANCING



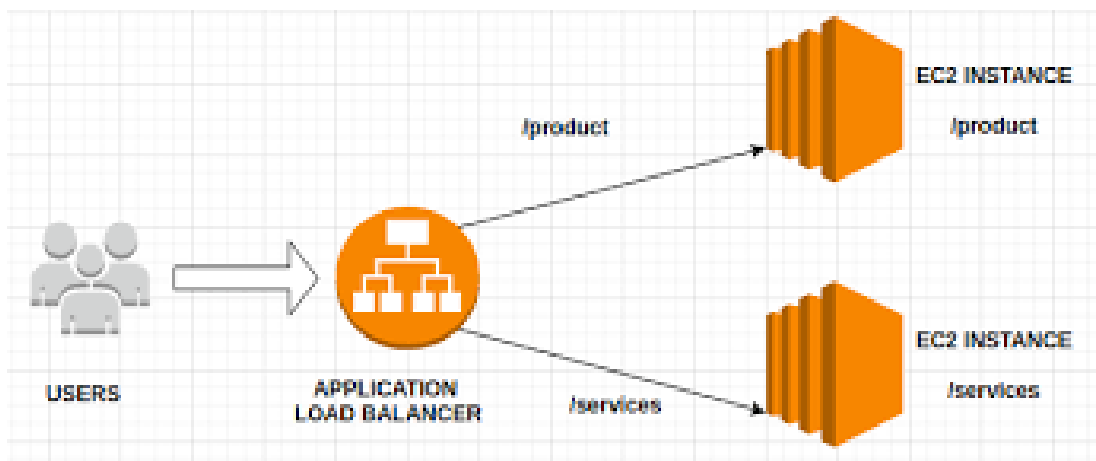


# BALANSIRANJE OPTEREĆENJA SLOJA 7

Balansiranje opterećenja Sloja 7 može biti skuplje od Sloja 4 u smislu vremena i potrebne računarske snage, ali ipak može dovesti do veće ukupne efikasnosti.

Balanser opterećenja Sloja 7 može da odredi koju vrstu podataka (video, tekst itd.) klijent zahteva, ne morate da se dupliraju isti podaci na svim serverima sa balansiranim opterećenjem.

Oni upravljaju i manipulišu saobraćajem na osnovu potpunog razumevanja transakcije između klijenta i servera aplikacija



# Raspoređivači opterećenja na sloju 4 i sloju 7

Na sloju 4 primarni protokoli koji se koriste su TCP i UDP.

- Ovi protokoli nisu svesni protokola višeg nivoa, kao što su FTP, HTTP, HTTPS, DNS, RDP itd.
- Load balanseri odluku o raspodeli opterećenja mogu da donesu samo na osnovu detalja dostupnih na slojevima 4 i nižim, poput brojeva portova i IP adresa.

Na sloju 7, load balanser ima više informacija za donošenje odluka o balansiranju opterećenja, jer je dostupno više informacija o protokolima višeg nivoa.

- Balansiranje opterećenja sloja 7 koristi proxy na aplikacionom sloju (HAProxy).
- Zahtevi se završavaju na raspoređivaću opterećenja, a proxy generiše novi zahtev koji se prosleđuje izabranom serveru podataka.

## PERFORMANCE

- Zbog povećane količine informacija na sloju 7, performanse nisu tako brze kao na sloju 4.

# Raspoređivači opterećenja na sloju 4 i sloju 7

## POSTOJANOST

- Mogućnost da se osigura da se određeni klijent ponovo poveže na isti server u određenom vremenskom roku.
- Jednom kada korisnik stupi u interakciju sa određenim serverom, svi naredni zahtevi se šalju na isti server i na taj način traju na tom serveru.
  - Na sloju 4, postojanost izvornog IP -a je jedina opcija.
  - Na sloju 7 dostupne su dodatne metode, kao što je postojanost HTTP kolačića

## STANJE SESIJE

- Ako server aplikacija otkaže i automatski se skloni iz klastera, tada će se sve trenutne korisničke sesije preneti na druge servere u klasteru bez potrebe da se korisnici ponovo prijavljuju u aplikaciju.
- Standardni ASP i PHP podaci o sesijama se podrazumevano skladište lokalno,
- Ako server aplikacije otkaže, svi lokalni podaci o sesiji će biti izgubljeni, korisnik se mora ponovo prijaviti i eventualno izgubiti korpe za kupovinu ...
- Ovaj problem se rešava implementacijom zajedničkog trajnog skladišta podataka za klaster - zajednička pozadinska baza podataka

# RASPOREĐIVAČI OPTEREĆENJA

## OPCIJE POSTOJANOSTI

- IP izvora (podmreža)
- Kolačić (aktivan ili pasivan)
- ID sesije SSL
- X-Forwarded -Za zaglavlje
- Microsoft Connection Broker/Session Broker Integration

**Performanse** Balanser opterećenja može povećati performanse omogućavajući da se koriste nekoliko servera za rukovanje radnim opterećenjem jedne aplikacije.

**Pouzdanost** Pokretanje aplikacije na jednom serveru stvara jednu tačku kvara. Korišćenjem balansa opterećenja se tačka kvara premešta u balanser opterećenja. Savetuje se upotreba redundantnih balansera da bi uklonili jedinu tačku kvara.

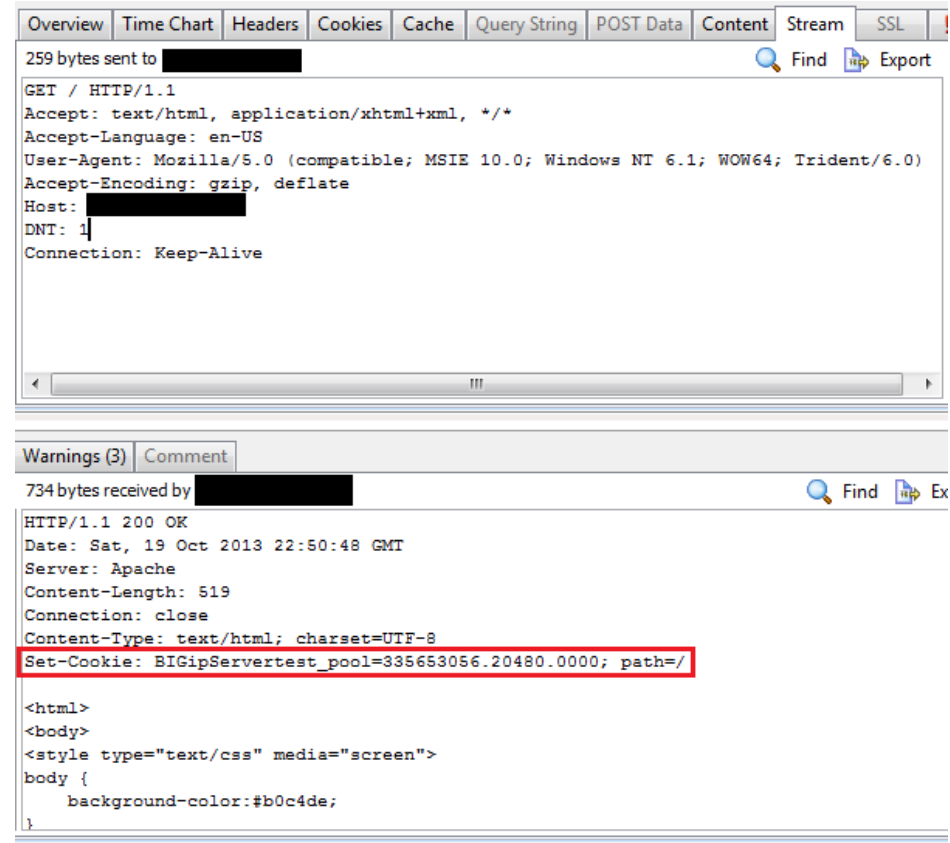
**Održavanje** Korišćenjem uređaja možete lako da dovedete servere na mrežu i van nje radi izvođenja zadatke održavanja, bez ometanja korisnika.

# RASPOREĐIVAČ OPTEREĆENJA ZASNOVAN NA KOLAČIĆU

- Popularan metod koji koriste hardverski raspoređivači opterećenja je umetanje kolačića u pretraživač klijenta.
- Kolačić vezuje klijenta za određeni server.
- Metod omogućava ravnomerniju raspodelu opterećenja na serverima jer veliki broj klijenata koji se nalaze iza istog NAT servera imati istu izvornu IP adresu.
- Kolačići koje podesi load balanser mogu da otkriju poverljive informacije
- Kriptovana vrednost koja sadrži naziv skladišta, IP adresu web servera i port
- Kolačić za F5 load balanser koristi format:

`BIGipServer<pool_name>=`

`<coded_server_IP>.<coded_server_port> .0000`



```
Overview | Time Chart | Headers | Cookies | Cache | Query String | POST Data | Content | Stream | SSL | ...
259 bytes sent to [redacted] Find Export
GET / HTTP/1.1
Accept: text/html, application/xhtml+xml, */*
Accept-Language: en-US
User-Agent: Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.1; WOW64; Trident/6.0)
Accept-Encoding: gzip, deflate
Host: [redacted]
DNT: 1
Connection: Keep-Alive

Warnings (3) | Comment
734 bytes received by [redacted] Find Export
HTTP/1.1 200 OK
Date: Sat, 19 Oct 2013 22:50:48 GMT
Server: Apache
Content-Length: 519
Connection: close
Content-Type: text/html; charset=UTF-8
Set-Cookie: BIGipServertest_pool=335653056.20480.0000; path=/
<html>
<body>
<style type="text/css" media="screen">
body {
    background-color:#b0c4de;
}
```

# DETEKTOVANJE RASPOREĐIVAČA OPTEREĆENJA

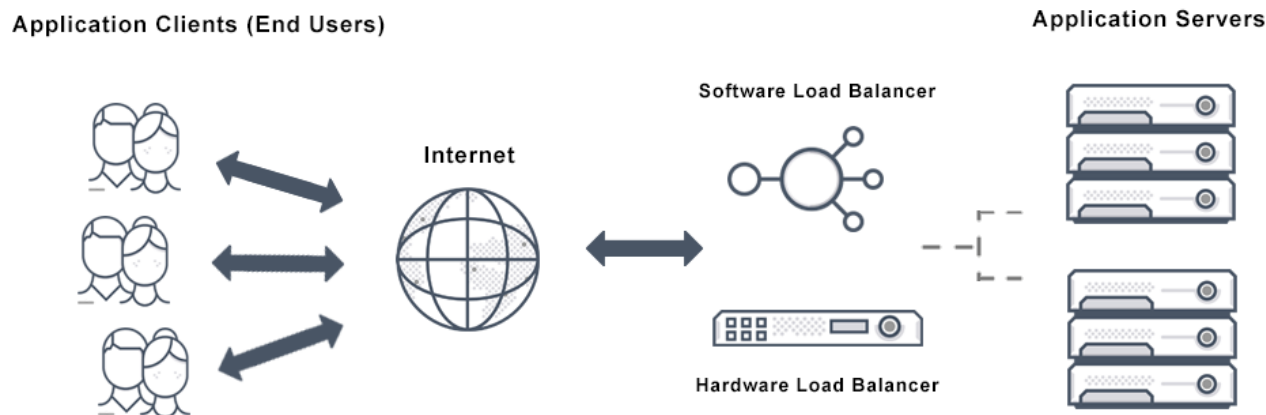
Pored umetanja kolačića postoje i drugi načini da se detektuje prisustvo load balansera.

- **Analiza SSL razlika između servera**
  - Mogu postojati male promene u SSL konfiguraciji na različitim web serverima, npr. vremenska oznaka na sertifikatu koja se izdaje web serverima može da varira.
- **Preusmeravanje na drugi URL** – preusmeravanje klijenta na drugi URL
  - Korisnik može da pretražuje web sajt [www.xxx.com](http://www.xxx.com) stin što se preusmerava na [www2.xxx.com](http://www2.xxx.com).
  - Zahtev od sledećeg korisnika će biti preusmeren na [www1.xxx.com](http://www1.xxx.com).
  - Ne primenjuje se često jer izaziva dodatno trošenje memorije za upravljanje i bezbedonosne probleme.
- **DNS zapisi za raspoređivače opterećenja**
- **Detektor raspoređivača opterećenja**
  - alat koji je uključen u Kali Linux
  - Komanda `lbd <ime web sajta>`

# VRSTE RASPOREĐIVAČA OPTEREĆENJA

## HARDVERSKI LOAD BALANSERI

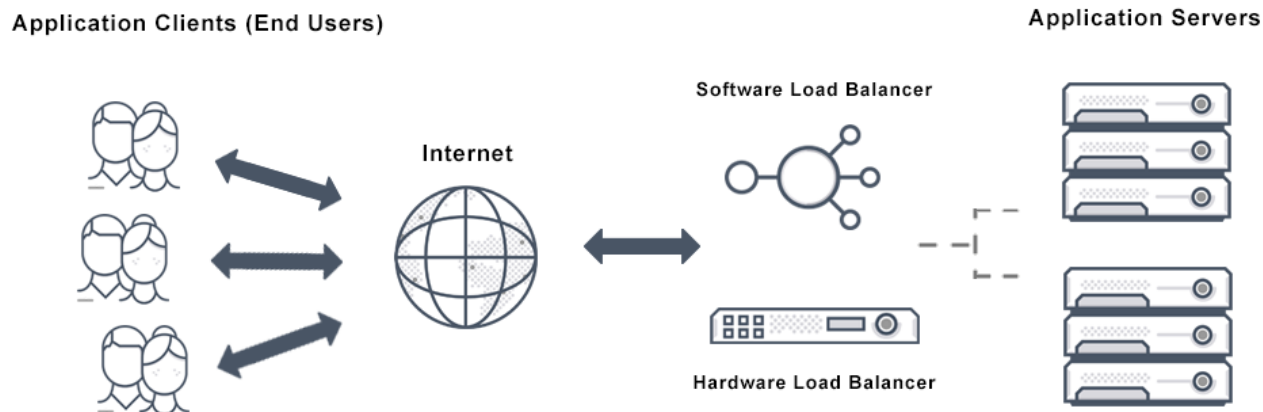
- Hardverski balanser opterećenja je hardverski uređaj sa specijalizovanim operativnim sistemom koji distribuira saobraćaj veb aplikacija preko grupe servera aplikacija.
- Pri odabiru balansera opterećenja, treba da imati u vidu troškove hardvera
- Kompanije koje koriste hardverske balansere opterećenja ne mogu iskoristiti prednosti računarstva u oblaku, što smanjuje troškove, dok nudi veće performanse, automatizaciju i elastičnost.
- hardverski balanseri i serveri aplikacija raspoređeni su u lokalnim centrima podataka, a broj balansiranih opterećenja zavisi od očekivane količine saobraćaja.



# VRSTE RASPOREĐIVAČA OPTEREĆENJA

## SOFTVERSKI LOAD BALANSERI

- Predstavljaju računarske aplikacije koje je potrebno instalirati u sistem gde će funkcionisati
- Postoje dve vrste - komercijalni i otvoreni kod (open source)
- Za hardverske balansere potrebni su vlasnički hardverski uređaji, dok se softverski balans opterećenja jednostavno instaliraju na standardne x86 servere ili virtuelne mašine.
- Softverski balanseri opterećenja se mogu elastično skalirati kako bi zadovoljili potražnju.
- Hardverski uređaji nisu kompatibilni sa cloud okruženjima, dok su softverski balanseri kompatibilni sa virtuelnim, kontejnerima i cloud platformama.





# VRSTE RASPOREĐIVAČA OPTEREĆENJA

---

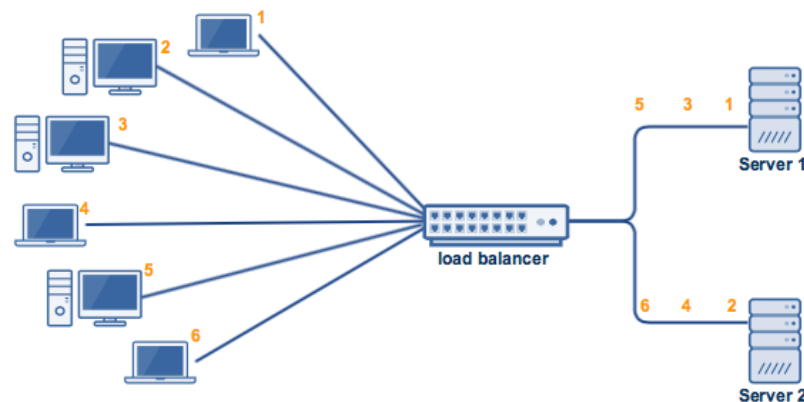
## VIRTUELNI LOAD BALANSERI

- Kombinacija programa hardverskog balansa opterećenja koji radi na virtuelnoj mašini
- Programska aplikacija hardverske opreme izvršava se na virtuelnoj mašini kako bi se saobraćaj na odgovarajući način preusmerio.
- Virtuelni balanseri opterećenja imaju slične izazove kao i hardverski, nedostatak centralnog upravljanja, manja skalabilnost i ograničena automatizacija.
- Balansi softverskog opterećenja potpuno su drugačija arhitektura dizajnirana za visoke performanse i agilnost.
- Balansi softverskog opterećenja takođe nude niže troškove bez zaključavanja u bilo kog dobavljača.

# ALGORITMI ZA BALANSIRANJE OPTEREĆENJA

## ROUND ROBIN ALGORITAM

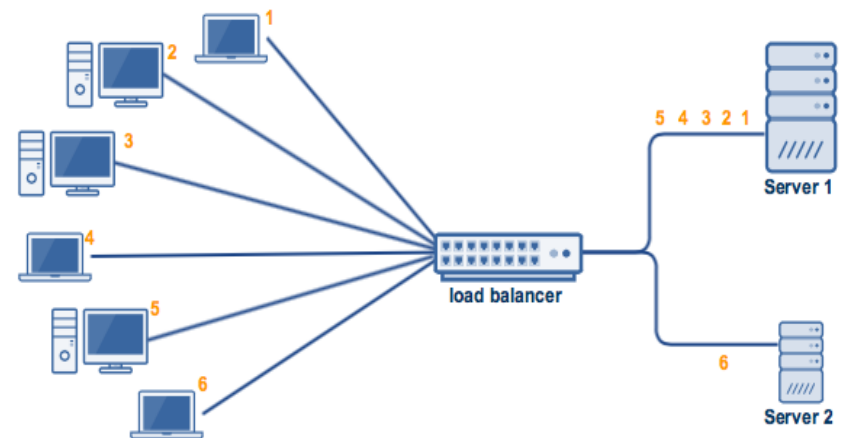
- Oslanja se na rotacioni sistem za sortiranje saobraćaja pri radu sa serverima iste težine.
- Zahtev se prenosi na prvi raspoloživi server, a zatim se taj server postavlja na dno čekaonice.
- Balansiranje opterećenja pomoću kružnih tokova jedna je od najjednostavnijih metoda za distribuciju zahteva klijenata preko grupe servera.
- Glavna prednost ovog algoritma je jednostavna implementacija.
- Često ne obezbeđuje najefikasniju raspodelu saobraćaja, jer **algoritam pretpostavljaju da su svi serveri isti**: da trenutno obrađuju isto opterećenje i sa istim skladišnim i računarskim kapacitetom
- Round Robin algoritam je najbolji za klastere koji se sastoje od servera sa identičnim specifikacijama.



# ALGORITMI ZA BALANSIRANJE OPTEREĆENJA

## WEIGHTED ROUND ROBIN ALGORITAM

- Algoritam se koristi za balansiranje opterećenja više servera sa različitim karakteristikama.
- Weighted Round Robin algoritam sličan je Round Robin algoritmu u smislu da je način na koji se zahtevi dodeljuju čvorovima i dalje cikličan, iako sa obrtom.
- Čvoru sa boljim specifikacijama biće dodeljen veći broj zahteva na osnovu veće konfigurisane težine.
- Ako je kapacitet Servera1, 5 puta veći od Servera2, tada Serveru1 dodeljujemo težinu 5, a Serveru2 težinu 1.
- Kada dolaze zahtevi, prvih 5 će biti dodeljeno čvoru 1, a 6-ti čvoru 2.

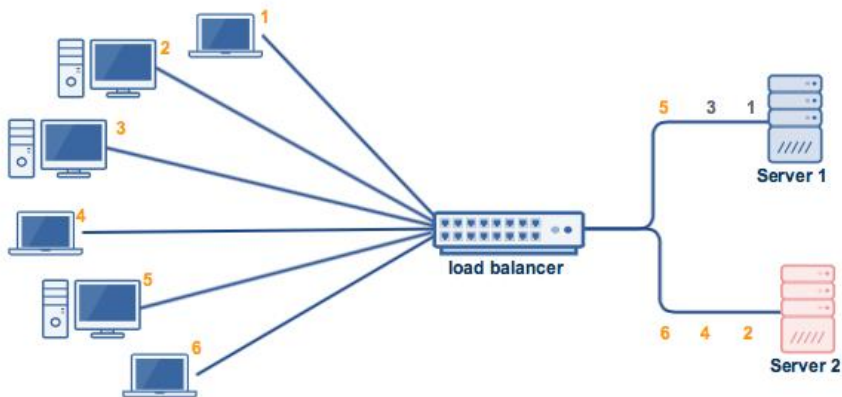


# ALGORITMI ZA BALANSIRANJE OPTEREĆENJA

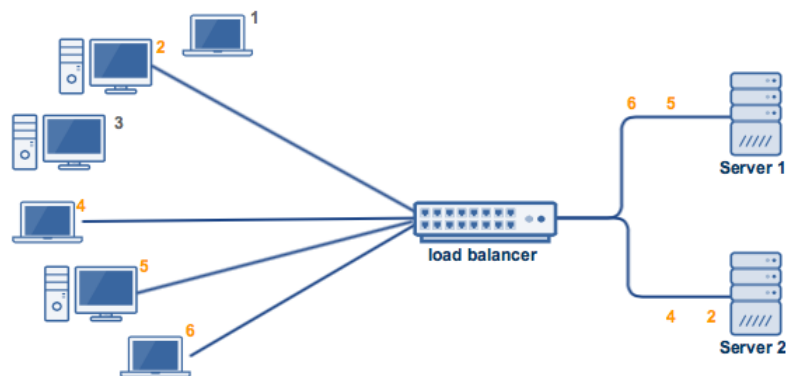
## LEAST CONNECTION ALGORITAM

- Zahtev je usmeren na server sa najmanje prometa.
- Ovo pomaže u održavanju optimizovanih performansi, posebno u vreme najvećih opterećenja održavanjem ujednačenog opterećenja na svim serverima.
- Mogu postojati slučajevi kada dva servera u klasteru imaju potpuno iste specifikacije, jedan server i dalje može biti preopterećen više od drugog.
- Jedan od mogućih razloga je taj što klijenti koji se povezuju na Server2 ostaju povezani duže od onih koji se povezuju na Server1.

Klijenti 1 i 3 su u međuvremenu prekinuli vezu, dok su 2, 4, 5 i 6 još uvek povezani. Resursi servera 2 su više zauzeti



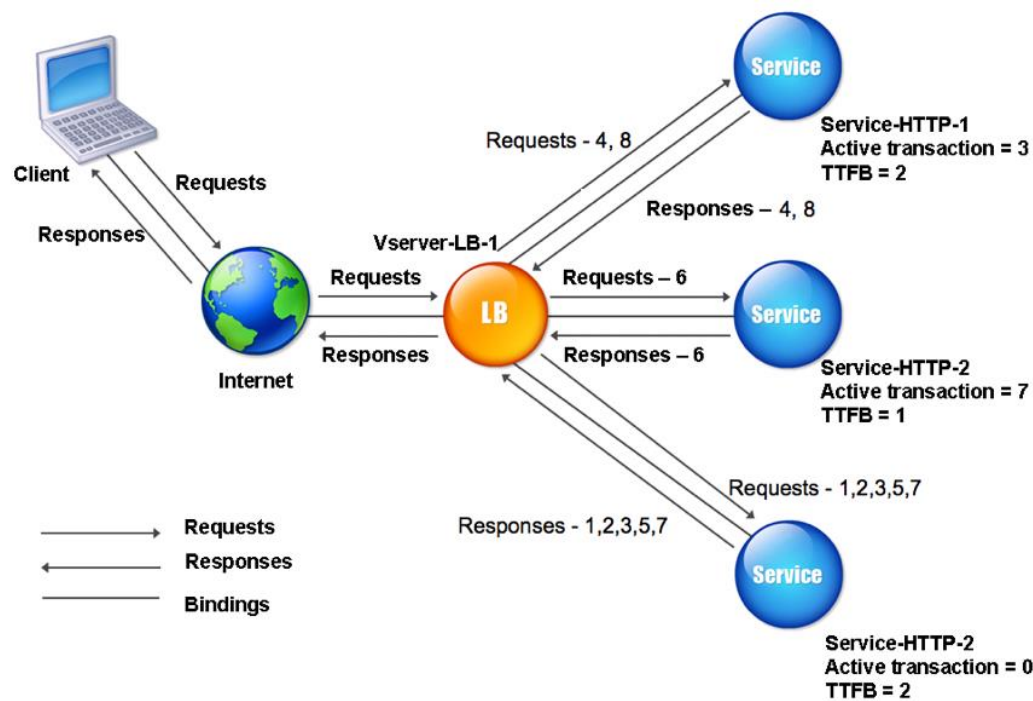
Balanser opterećenja će dodeliti klijenta 6 Serveru1 umesto Serveru2.



# ALGORITMI ZA BALANSIRANJE OPTEREĆENJA

## LEAST RESPONSE TIME ALGORITAM

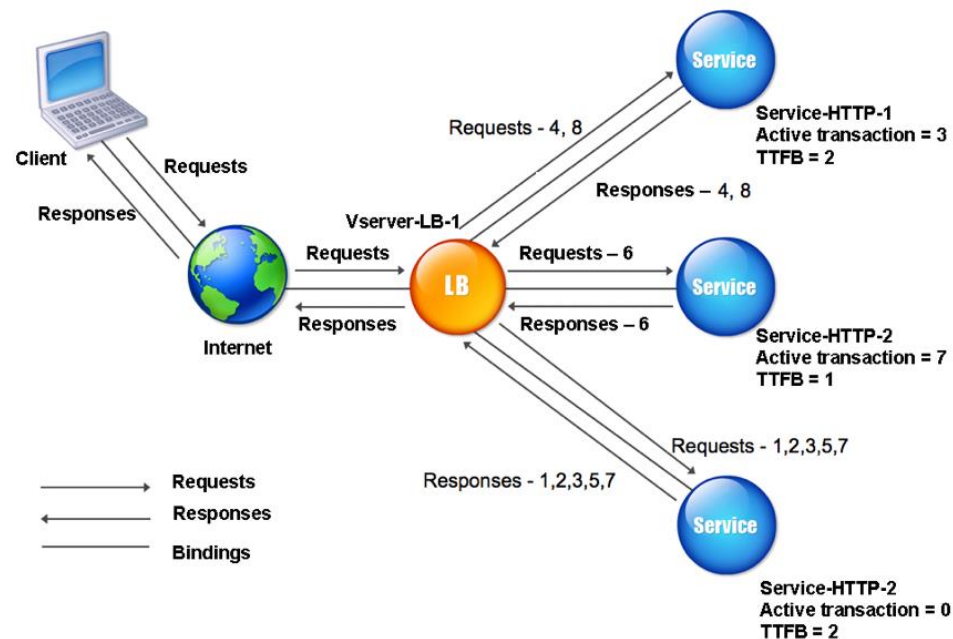
- Algoritam bira server sa najmanje aktivnih veza i najnižim prosečnim vremenom odziva.
- Algoritam se primenjuje samo za virtuelne servere za HTTP-a i *Secure Sockets Layer (SSL) servise*.
- Vreme odziva (takođe nazvano *Time to First Byte*, ili TTFB) je vremenski interval između slanja paketa zahteva servisu i prijema prvog paketa odgovora od servisa.



# ALGORITMI ZA BALANSIRANJE OPTEREĆENJA

## LEAST RESPONSE TIME ALGORITAM

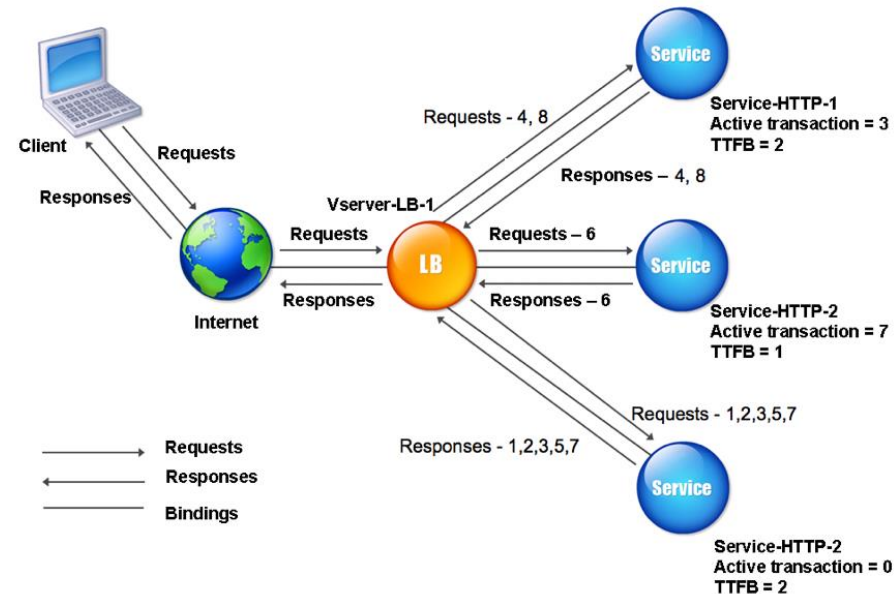
- Service-HTTP-1 upravlja sa tri aktivne transakcije, a TTFB je dve sekunde.
- Service-HTTP-2 obrađuje sedam aktivnih transakcija, a TTFB jedna sekunda.
- Service-HTTP-3 ne obrađuje nijednu aktivnu transakciju i TTFB je dve sekunde.
- Virtuelni server bira servis množenjem broja aktivnih transakcija sa TTFB za svaki servis.
- Odlučuje se za servis sa najmanjim rezultatom



# ALGORITMI ZA BALANSIRANJE OPTEREĆENJA

## LEAST RESPONSE TIME ALGORITAM

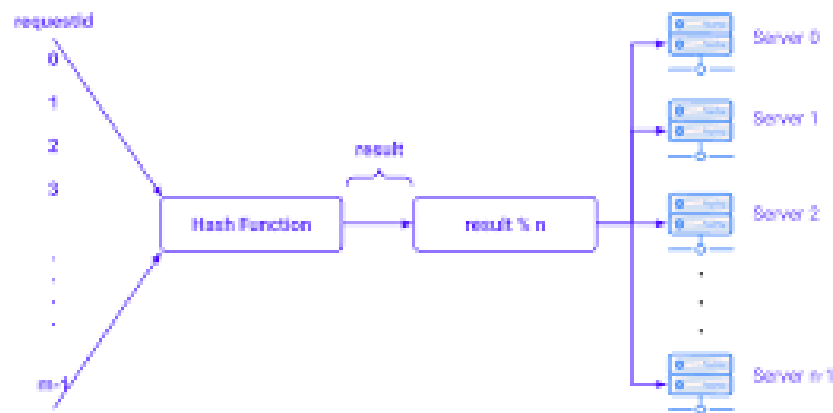
- Virtuelni server prosleđuje zahteve na sledeći način:
- Usluga-HTTP-3 prima prvi zahtev, jer usluga ne obrađuje nijednu aktivnu transakciju.
- Service-HTTP-3 takođe prima drugi i treći zahtev, jer je rezultat najniži od tri servisa.
- Service-HTTP-1 prima četvrti zahtev.
- Pošto Service-HTTP-1 i Service-HTTP-3 imaju isti rezultat, algoritam bira između njih primenom Round Robin metode.
- Service-HTTP-3 prima peti zahtev.
- Service-HTTP-2 prima šesti zahtev, jer u ovom trenutku ima najmanji rezultat.



# ALGORITMI ZA BALANSIRANJE OPTEREĆENJA

## IP HASH ALGORITAM

- Algoritmi heširanja se koriste u slučaju postojanih veza (vezati zahteve klijenta na određeni server).
- Algoritam koristi izvornu i odredišnu IP adresu klijenta i servera za generisanje jedinstvenog heš ključa.
- Ključ se koristi za dodeljivanje klijenta određenom serveru.
- Ključ se može regenerisati ako je sesija prekinuta, algoritam može osigurati da je klijent usmeren na isti server koji je ranije koristio.
- Ovaj algoritam dobro funkcioniše za TCP veze koje ne koriste kolačiće.

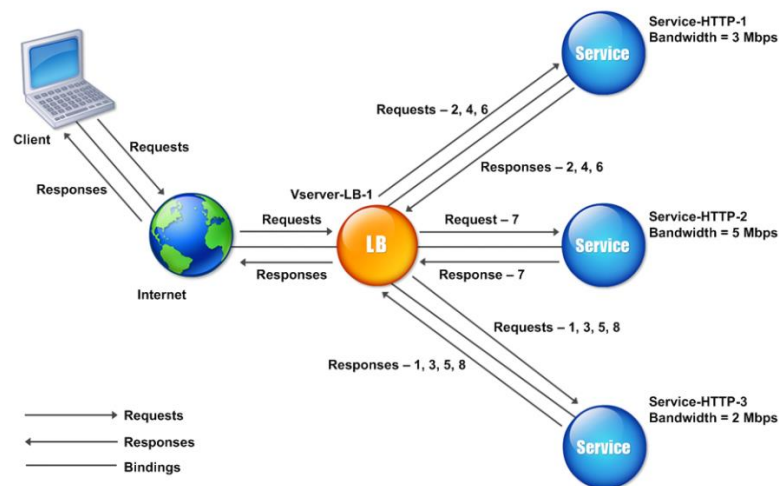




# ALGORITMI ZA BALANSIRANJE OPTEREĆENJA

## Least Bandwidth Algorithm (algoritam najmanje propusnosti)

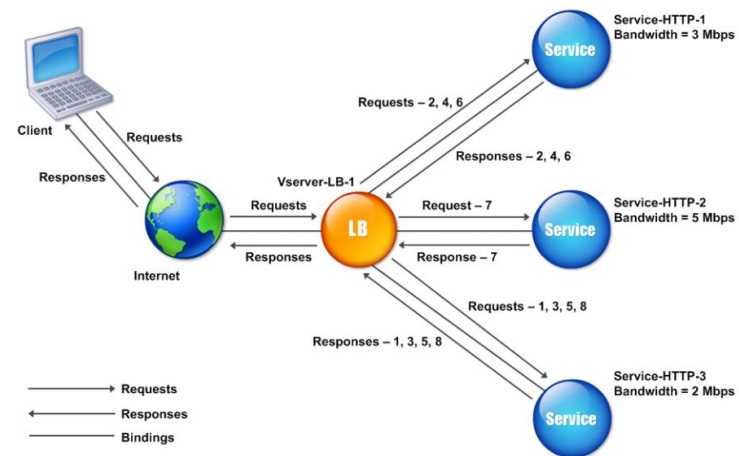
- Algoritam zahtev klijenta šalje serveru sa najmanje Mbps saobraćaja.
- Algoritam bira servis koji trenutno opslužuje najmanju količinu saobraćaja, merenu u megabitima u sekundi (Mbps).
- Service-HTTP-1 ima propusnost od 3 Mbps.
- Service-HTTP-2 ima propusnost od 5 Mbps.
- Service-HTTP-3 ima propusnost od 2 Mbps.



# ALGORITMI ZA BALANSIRANJE OPTEREĆENJA

## Least Bandwidth Algorithm (algoritam najmanje propusnosti)

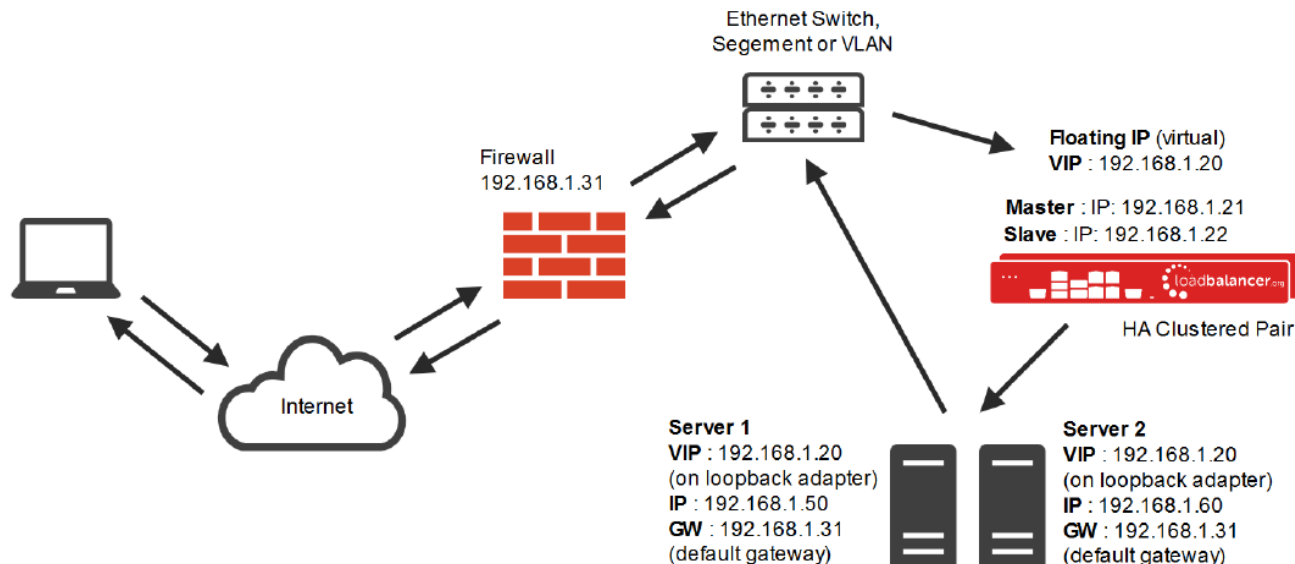
- Virtuelni server bira servis korišćenjem vrednosti propusnog opsega (N) što je zbir broja bajtova poslanih i primljenih u prethodnih 14 sekundi.
- Ako svaki zahtev zahteva propusnost od 1 Mbps, uređaj isporučuje zahteve na sledeći način:
- Service-HTTP-3 prima prvi zahtev, jer ova usluga ima najmanju N vrednost.
- Pošto Service-HTTP-1 i Service-HTTP-3 sada imaju istu N vrednost, virtuelni server se prebacuje na metodu ponavljanja za ove servere, naizmenično između njih.
- Service-HTTP-1 prima drugi zahtev, Service-HTTP-3 prima treći zahtev, Service-HTTP-1 prima četvrti zahtev, Service-HTTP-3 prima peti zahtev, a Service-HTTP-1 prima šesti zahtev .



# REŽIMI RADA RASPOREĐIVAČA OPTEREĆENJA

## Direktno Rutiranje (DR)

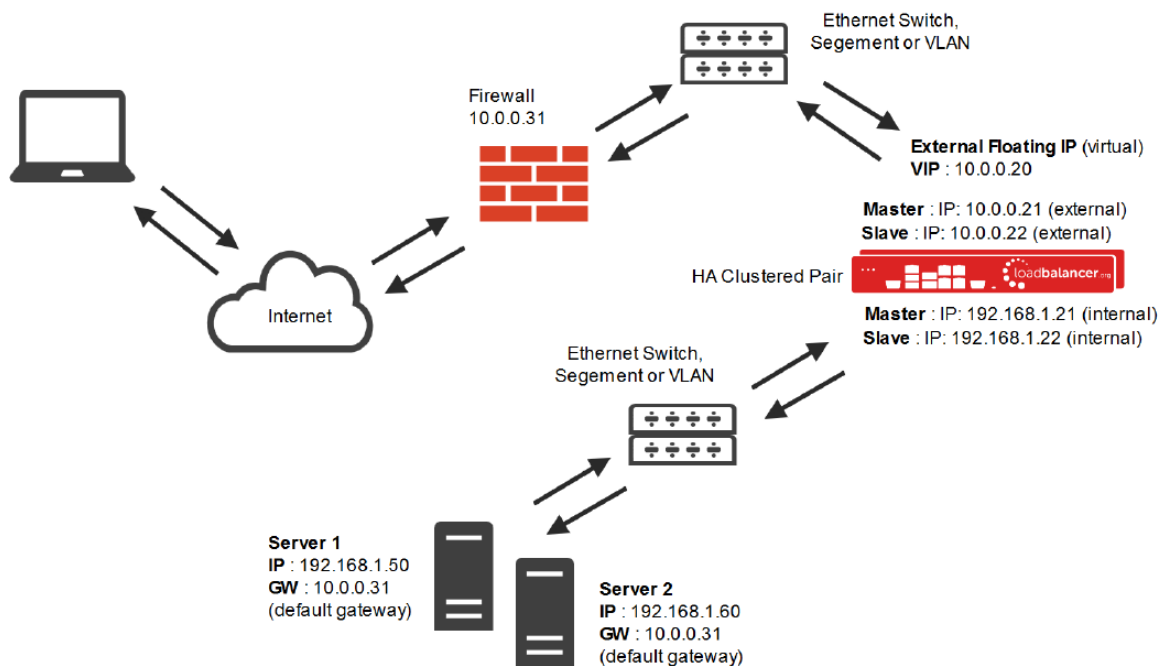
- U proseku, režim DR je 8 puta brži od NAT -a za HTTP.
- Balanser opterećenja mora imati interfejs u istoj podmreži kao i pravi serveri kako bi se osigurala povezanost sloja 2 potrebna za rad DR režima.
- DR je transparentan, odnosno pravi server će videti izvornu IP adresu klijenta.
- Odgovori servera se vraćaju klijentu direktno a ne preko load balansera



# REŽIMI RADA RASPOREĐIVAČA OPTEREĆENJA

## PREVOĐENJE ADRESE MREŽE IZVORA (Layer 4 source network address translation L4 SNAT)

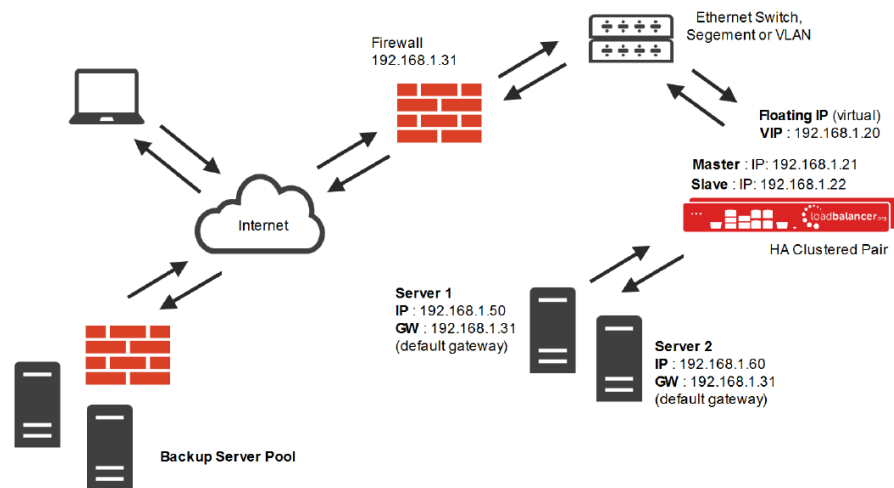
- Sloj 4 SNAT režim nije transparentan, pravilo SNAT prevodi izvornu IP adresu u IP adresu load balansera umesto u originalnu IP adresu klijenta.
- Prevođenje portova nije moguće u sloju 4 SNAT režimu, tj. Koji ima drugačiji RIP port od VIP porta.



# REŽIMI RADA RASPOREĐIVAČA OPTEREĆENJA

## PREVOĐENJE ADRESE MREŽE IZVORA-Layer 7 source network address translation

- Sloj 7 SNAT režim koristi proxy (HAProxy) na aplikacionom sloju.
- Dolazni zahtevi se prekidaju na load balanseru.
- HAProxy generiše novi zahtev izabranom tzv real serveru.
- Kao rezultat toga, Sloj 7 je sporija tehnika od DR ili NAT režima na Sloju 4.
- Sloj 7 se obično bira kada su potrebne poboljšane opcije kao što su prekid SSL-a, postojanost zasnovana na kolačićima, prepisivanje URL adresa, umetanje/brisanje zaglavlja.
- Sloj 7 SNAT režim nije podrazumevano transparentan, tj. Pravi serveri neće videti izvornu IP adresu klijenta



# RASPOREĐIVAČI OPTEREĆENJA

---

## ZAKLJUČAK

- Balansiranje opterećenja inteligentno distribuira saobraćaj sa klijenata na više servera.
- Klijenti ne moraju da razumeju koliko servera je u upotrebi ili kako su konfigurisani.  
Raspoređivač opterećenja se nalazi između klijenata i servera
- Poboljšavaju korisničko iskustvo pružajući dodatnu sigurnost, performanse, otpornost i pojednostavljajući skaliranje vaše veb stranica.
- Klijenti se povezuju na jedan URL (npr. [www.xxx.com](http://www.xxx.com)), a raspoređivač opterećenja će osigurati vezu i odlučiti koji od servera je u stanju da najbolje opslužuje klijenta (najmanje zauzet, najmanji broj konekcija, najbolje vreme odziva itd.).

# RASPOREĐIVAČI OPTEREĆENJA

---

## ZAKLJUČAK

### POBOLJŠANE PERFORMANSE

- Balansi opterećenja smanjuju dodatno opterećenje na određenom serveru
- *SSL OFFLOAD* - Zaštita saobraćaja pomoću SSL-a (Secure Sockets Layer) na load balanseru uklanja troškove sa veb servera što rezultira time da je više resursa dostupno za vašu veb aplikaciju
- Keširanje saobraćaja(*Traffic Caching*) Balanser opterećenja će zadržati kopiju često pristupanih elemenata web lokacije, poput slika što rezultira smanjenjem broja veb servera i bržom isporukom sadržaja korisnicima
- HTTP 2.0 je poboljšanje HTTP protokola što rezultira mnogo bržim veb stranicama. Balanser opterećenja može komunicirati sa klijentima koristeći HTTP 2.0 čak i ako to ne podržavaju web serveri

# RASPOREĐIVAČI OPTEREĆENJA

---

## ZAKLJUČAK

### ELASTIČNOST

- Pokvarene i loše performanse komponente mogu se odmah zameniti bez prekida pružanja usluge
- Ako je veb server zauzet, load balanser će to otkriti i preusmeriti saobraćaj na manje zauzete web servere
- Balanseri opterećenja mogu se rasporediti u parovima koji su dostupni, tako da ako jedan otkáže, drugi odmah preuzima bez uticaja na korisnike



# RASPOREĐIVAČI OPTEREĆENJA

---

## ZAKLJUČAK

### SIGURNOST

- Firewall web aplikacija (*The Web Application Firewall- WAF*) u balanseru opterećenja štiti web lokaciju od napada i uključuje dnevna ažuriranja pravila isto kao i skener virusa
- Balanser opterećenja može zatražiti korisničko ime i lozinku pre nego što odobri pristup web lokaciji radi zaštite od neovlašćenog pristupa
- Balanser opterećenja može otkriti i sprečiti (DDoS) napad pre nego što stigne na web lokaciju.

# RASPOREĐIVAČI OPTEREĆENJA

---

## ZAKLJUČAK

### SKALABILNOST

- Pojednostavljen kontinuitet poslovanja, load balanser može otkriti prekid na web lokaciji i neprimetno preusmeriti korisnike na drugu lokaciju.
- Jednostavno dodavanje dodatnih servera u load balanser povećava se kapacitet web lokacije umesto nadogradnjom postojećih servera što je skupo i uključuje prekid rada.