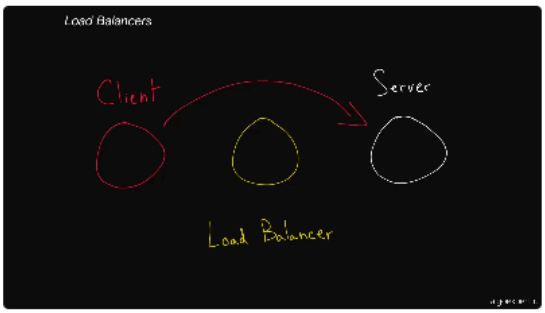
4/25/2021 -



Load Balancers

Relentlessly distributing network requests across multiple servers, these digital traffic cops act as watchful guardians for your system, ensuring that it operates at peak performance day and night.

Includes code example.

1 Prerequisite

Reverse Proxy
A server that sits between clients and servers and acts on behalf of the servers, typically used for logging, load balancing, or caching.

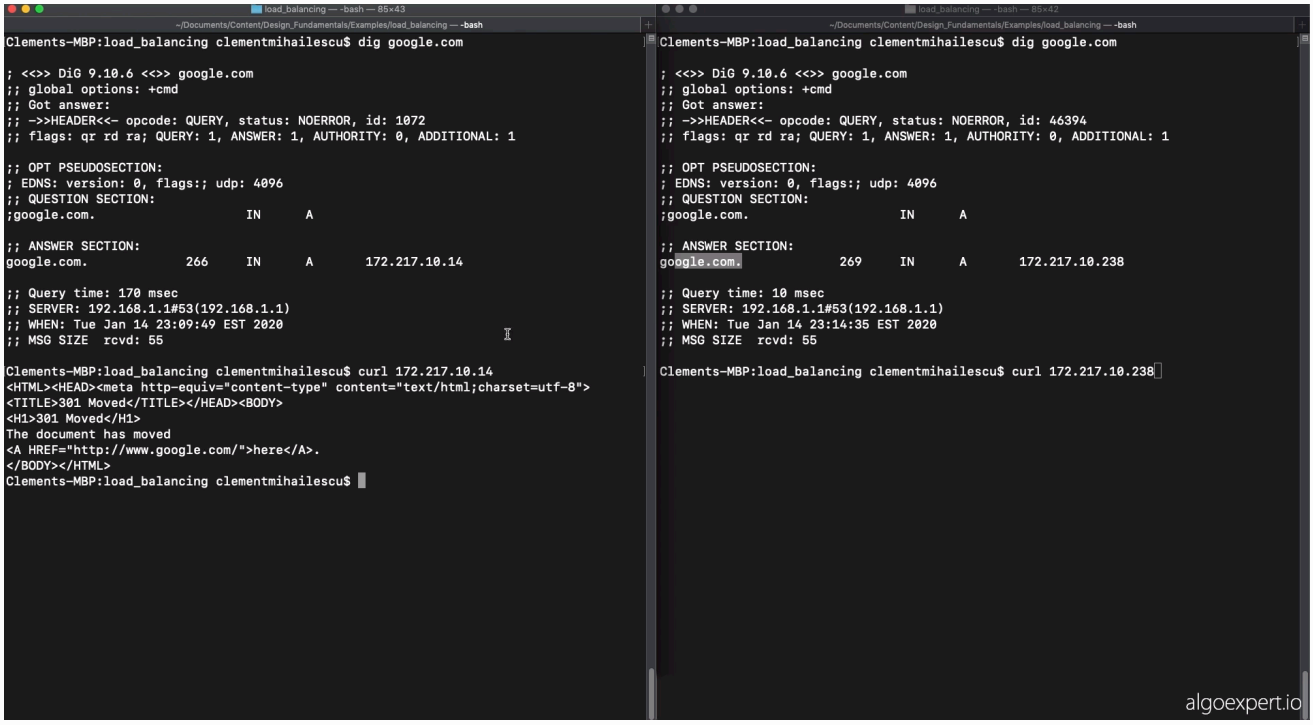
4 Key Terms

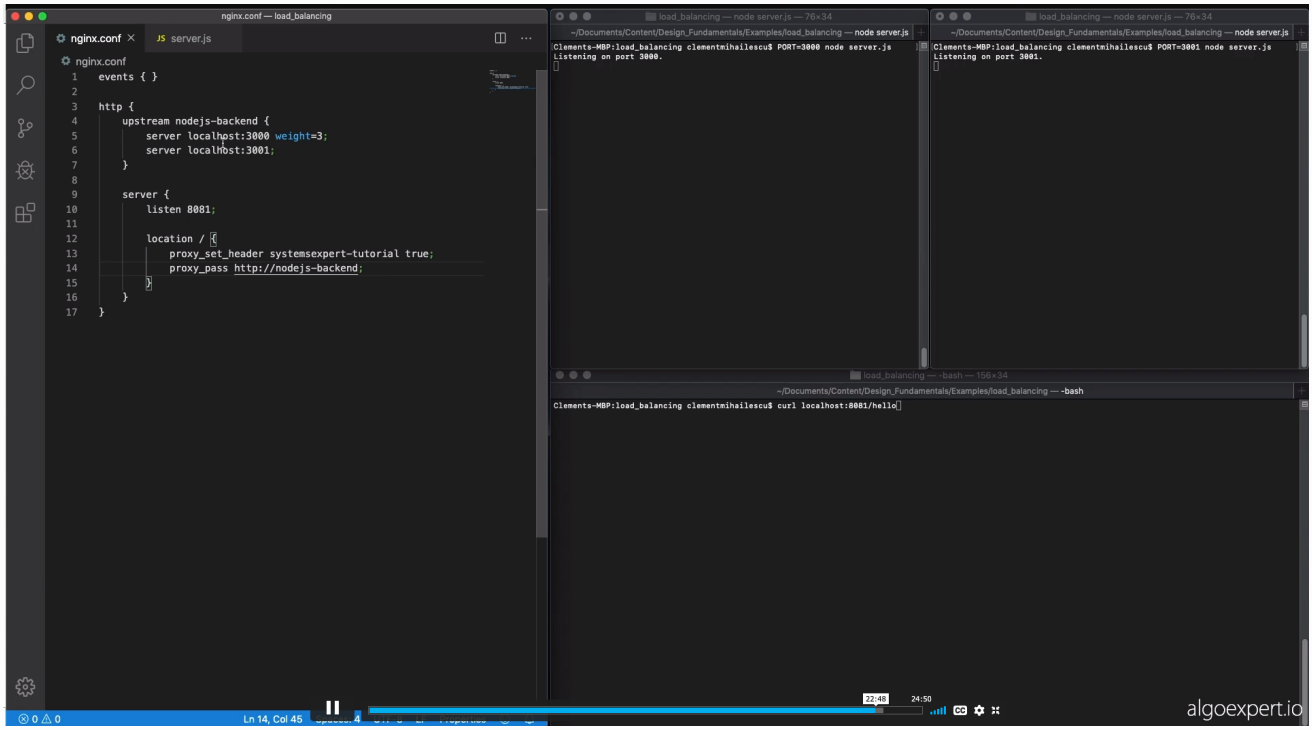
Load Balancer
A type of **reverse proxy** that distributes traffic across servers. Load balancers can be found in many parts of a system, from the DNS layer all the way to the database layer.

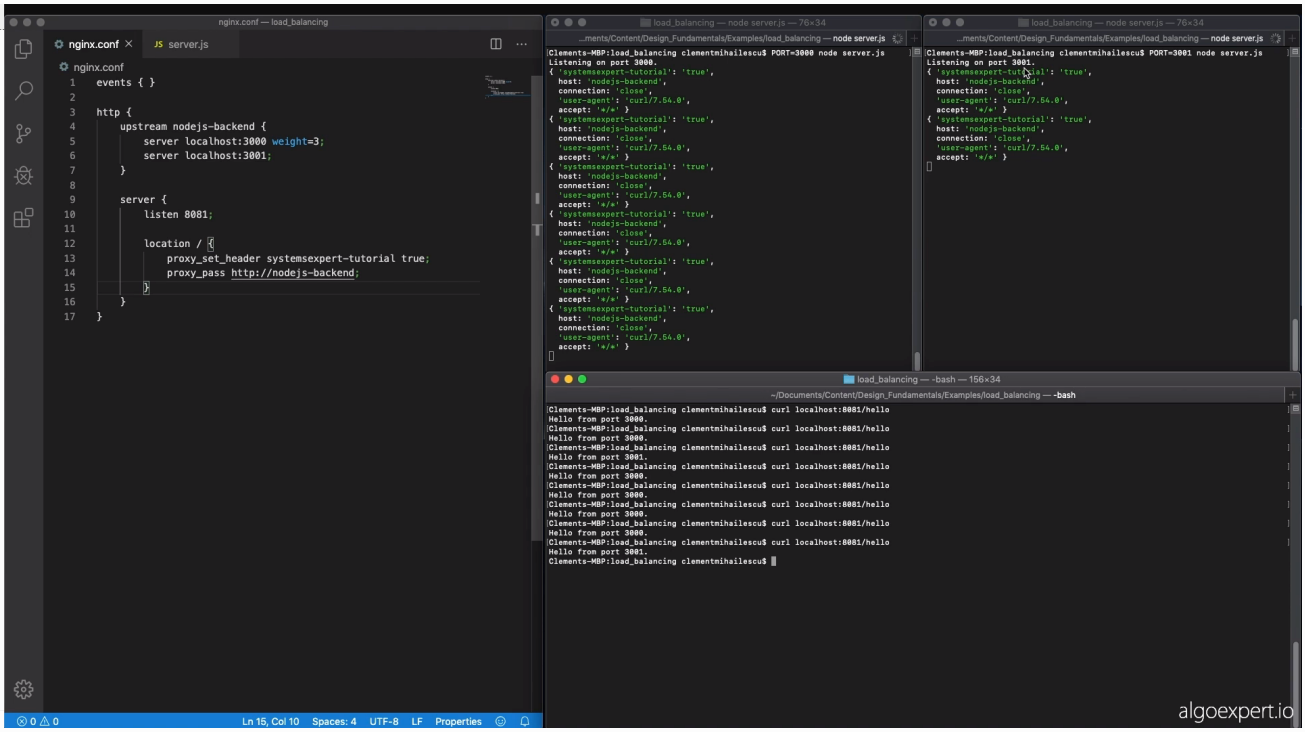
Server-Selection Strategy
How a **load balancer** chooses servers when distributing traffic amongst multiple servers. Commonly used strategies include round-robin, random selection, performance-based selection (choosing the server with the best performance metrics, like the fastest response time or the least amount of traffic), and IP-based routing.

Hot Spot
When distributing a workload across a set of servers, that workload might be spread unevenly. This can happen if your **sharding key** or your **hashing function** are suboptimal, or if your workload is naturally skewed: some servers will receive a lot more traffic than others, thus creating a "hot spot".

Nginx
Pronounced "engine X"—not "N jinx", Nginx is a very popular webserver that's often used as a **reverse proxy** and **load balancer**.


load_balancing — -bash — 85×43
~/Documents/Content/Design_Fundamentals/Examples/load_balancing — -bash
Clements-MBP:load_balancing clementmihailescu$ dig google.com
; <<>> DiG 9.10.6 <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1072
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;google.com. IN A
;; ANSWER SECTION:
google.com. 266 IN A 172.217.10.14
;; Query time: 170 msec
;; SERVER: 192.168.1.1#53(192.168.1.1)
;; WHEN: Tue Jan 14 23:09:49 EST 2020
;; MSG SIZE rcvd: 55
Clements-MBP:load_balancing clementmihailescu$ curl 172.217.10.14
<HTML><HEAD><meta http-equiv="content-type" content="text/html;charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.com/">here</A>.
</BODY></HTML>
Clements-MBP:load_balancing clementmihailescu$
load_balancing — -bash — 85×42
~/Documents/Content/Design_Fundamentals/Examples/load_balancing — -bash
Clements-MBP:load_balancing clementmihailescu$ dig google.com
; <<>> DiG 9.10.6 <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 46394
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;google.com. IN A
;; ANSWER SECTION:
google.com. 269 IN A 172.217.10.238
;; Query time: 10 msec
;; SERVER: 192.168.1.1#53(192.168.1.1)
;; WHEN: Tue Jan 14 23:14:35 EST 2020
;; MSG SIZE rcvd: 55
Clements-MBP:load_balancing clementmihailescu$ curl 172.217.10.238
algoexpert.io


nginx.conf — load_balancing
nginx.conf
server.js
events { }
http {
upstream nodejs-backend {
server localhost:3000 weight=3;
server localhost:3001;
}
server {
listen 8081;
location / {
proxy_set_header systemsexpert-tutorial true;
proxy_pass http://nodejs-backend;
}
}
}
load_balancing — node server.js — 76×34
Clements-MBP:load_balancing clementmihailescu$ PORT=3000 node server.js
Listening on port 3000.
load_balancing — node server.js — 76×34
Clements-MBP:load_balancing clementmihailescu$ PORT=3001 node server.js
Listening on port 3001.
load_balancing — -bash — 156×34
Clements-MBP:load_balancing clementmihailescu$ curl localhost:8081/hello
Ln 14, Col 45
22:48
24:50
algoexpert.io


nginx.conf — load_balancing
nginx.conf
server.js
events { }
http {
upstream nodejs-backend {
server localhost:3000 weight=3;
server localhost:3001;
}
server {
listen 8081;
location / {
proxy_set_header systemsexpert-tutorial true;
proxy_pass http://nodejs-backend;
}
}
}
Ln 15, Col 10 Spaces: 4 UTF-8 LF Properties
load_balancing — node server.js — 76×34
Clements-MBP:load_balancing clementmihailescu$ PORT=3000 node server.js
Listening on port 3000.
Clements-MBP:load_balancing clementmihailescu$ PORT=3001 node server.js
Listening on port 3001.
load_balancing — -bash — 156×34
Clements-MBP:load_balancing clementmihailescu$ curl localhost:8081/hello
Hello from port 3000.
Hello from port 3001.
algoexpert.io