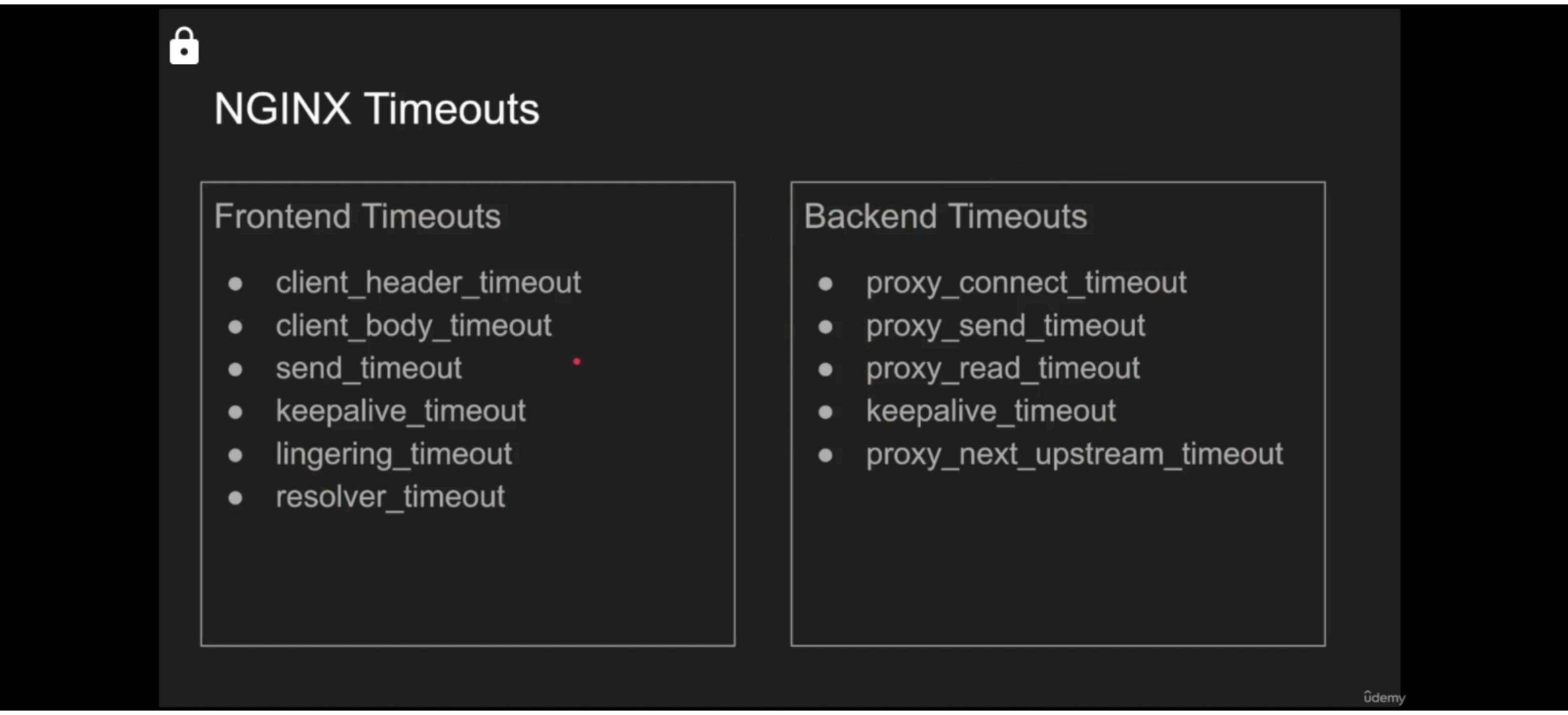


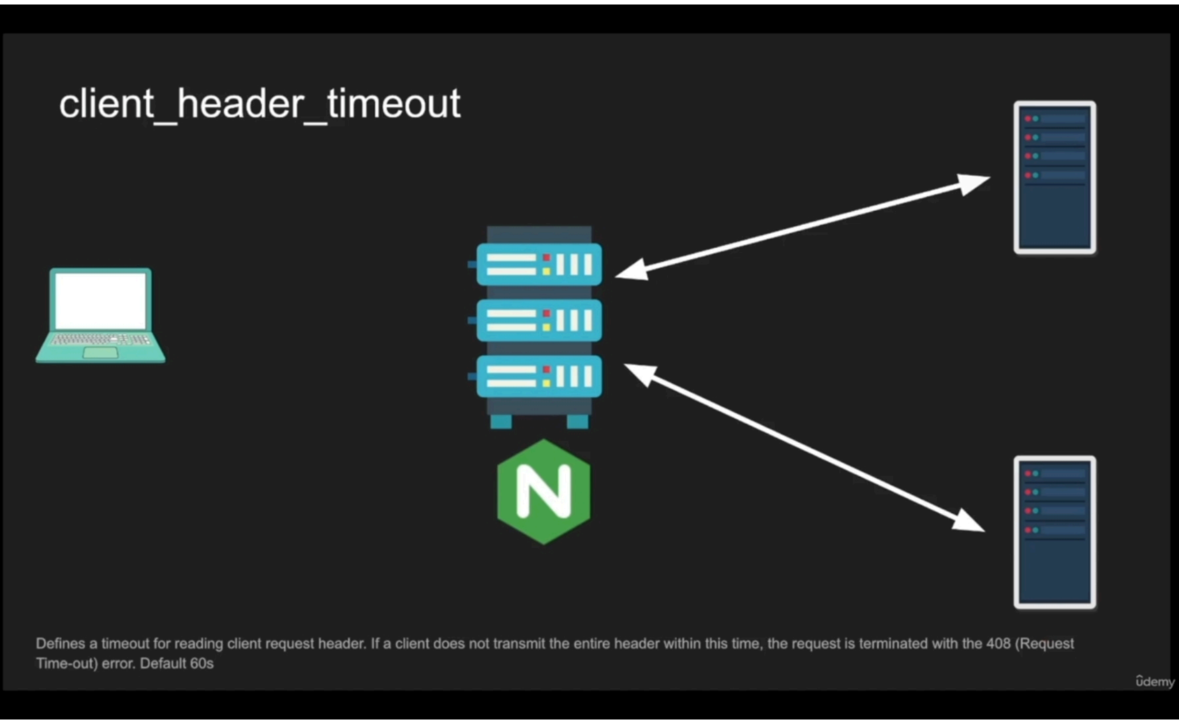
NGINX Timeouts

Frontend Timeouts

- client_header_timeout
- client_body_timeout
- send_timeout
- keepalive_timeout
- lingering_timeout
- resolver_timeout

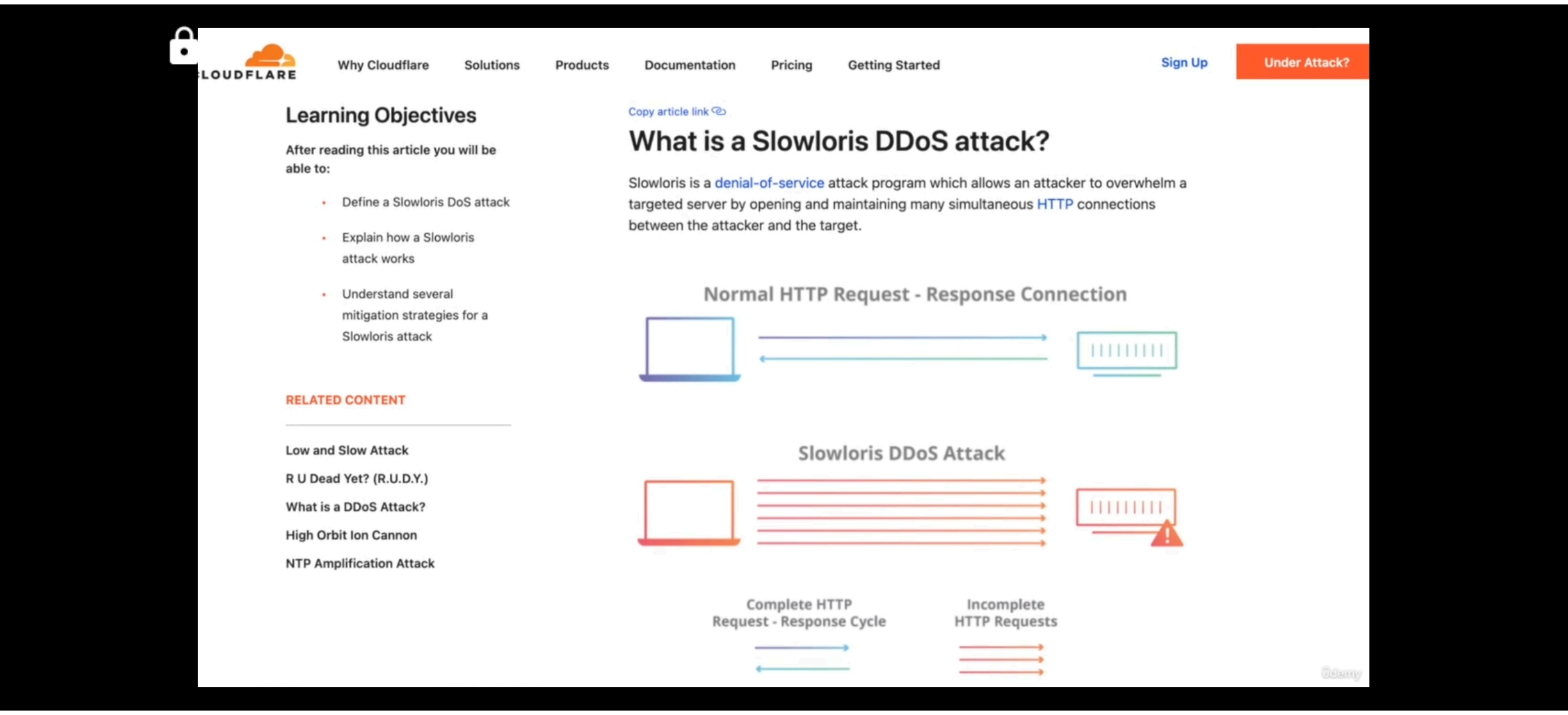
Backend Timeouts

- proxy_connect_timeout
- proxy_send_timeout
- proxy_read_timeout
- keepalive_timeout
- proxy_next_upstream_timeout



client_header_timeout

Defines a timeout for reading client request header. If a client does not transmit the entire header within this time, the request is terminated with the 408 (Request Time-out) error. Default 60s



Learning Objectives

After reading this article you will be able to:

- Define a Slowloris DoS attack
- Explain how a Slowloris attack works
- Understand several mitigation strategies for a Slowloris attack

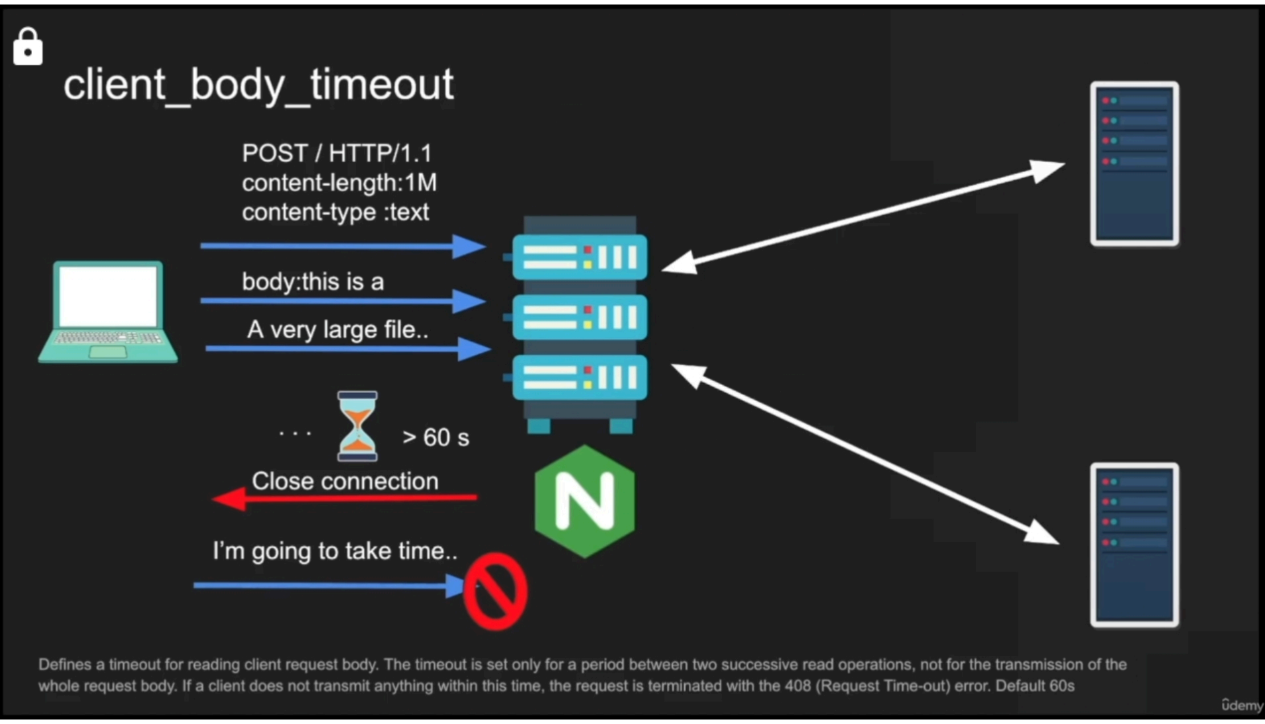
RELATED CONTENT

- Low and Slow Attack
- R U Dead Yet? (R.U.D.Y.)
- What is a DDoS Attack?
- High Orbit Ion Cannon
- NTP Amplification Attack

What is a Slowloris DDoS attack?

Slowloris is a denial-of-service attack program which allows an attacker to overwhelm a targeted server by opening and maintaining many simultaneous HTTP connections between the attacker and the target.

Normal HTTP Request - Response Connection
Slowloris DDoS Attack
Complete HTTP Request - Response Cycle
Incomplete HTTP Requests
Udemy



client_body_timeout

POST / HTTP/1.1
content-length:1M
content-type :text

body:this is a

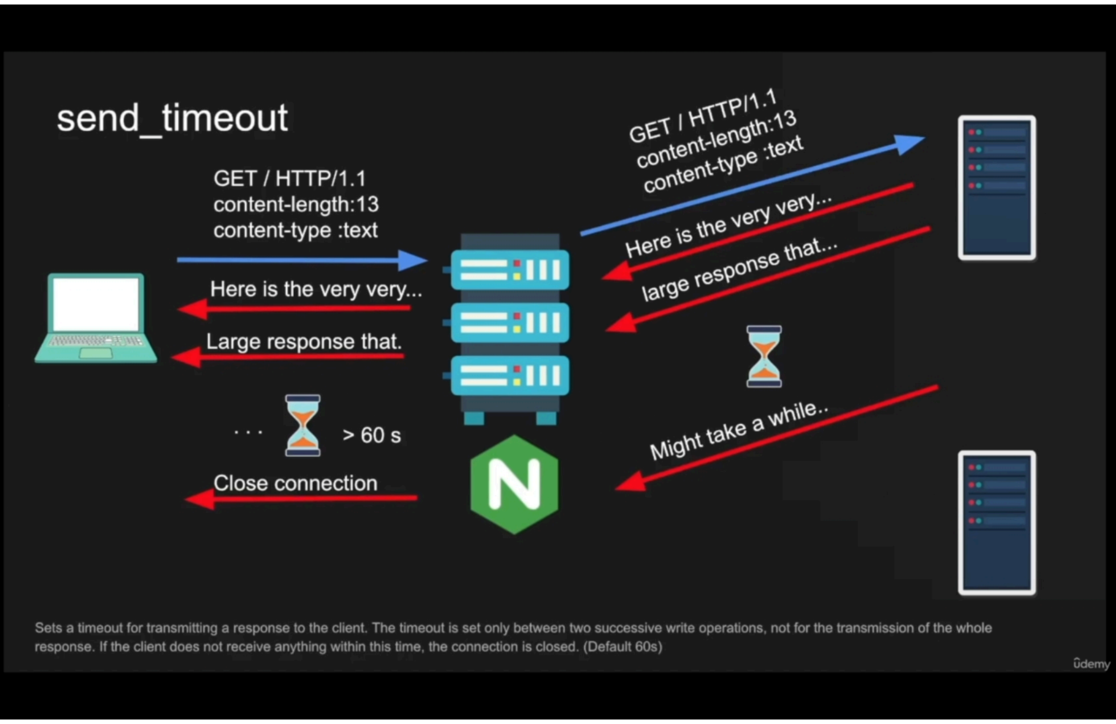
A very large file..

. . . > 60 s

Close connection

I'm going to take time..

Defines a timeout for reading client request body. The timeout is set only for a period between two successive read operations, not for the transmission of the whole request body. If a client does not transmit anything within this time, the request is terminated with the 408 (Request Time-out) error. Default 60s



send_timeout

GET / HTTP/1.1
content-length:13
content-type :text

Here is the very very...

Large response that.

. . . > 60 s

Close connection

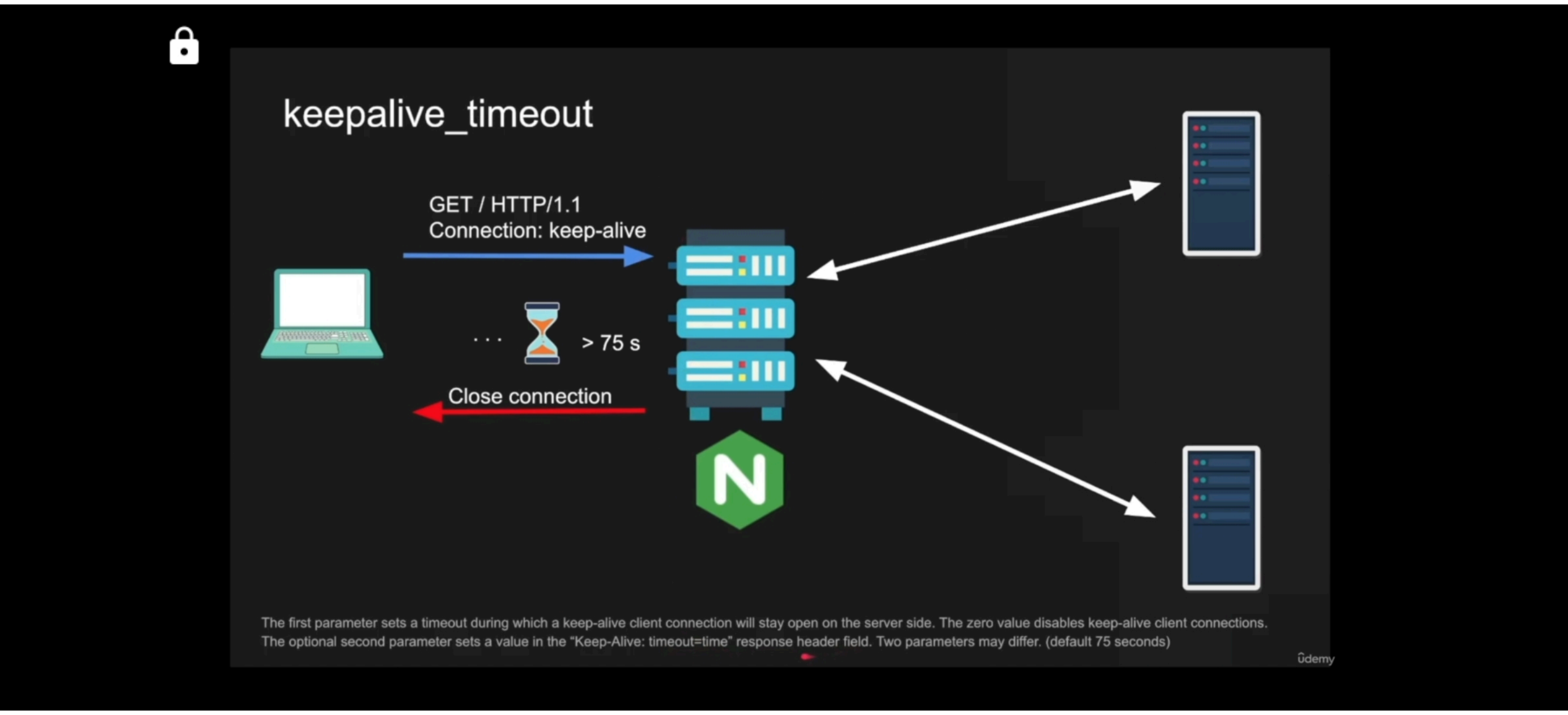
GET / HTTP/1.1
content-length:13
content-type :text

Here is the very very...

large response that...

Might take a while..

Sets a timeout for transmitting a response to the client. The timeout is set only between two successive write operations, not for the transmission of the whole response. If the client does not receive anything within this time, the connection is closed. (Default 60s)



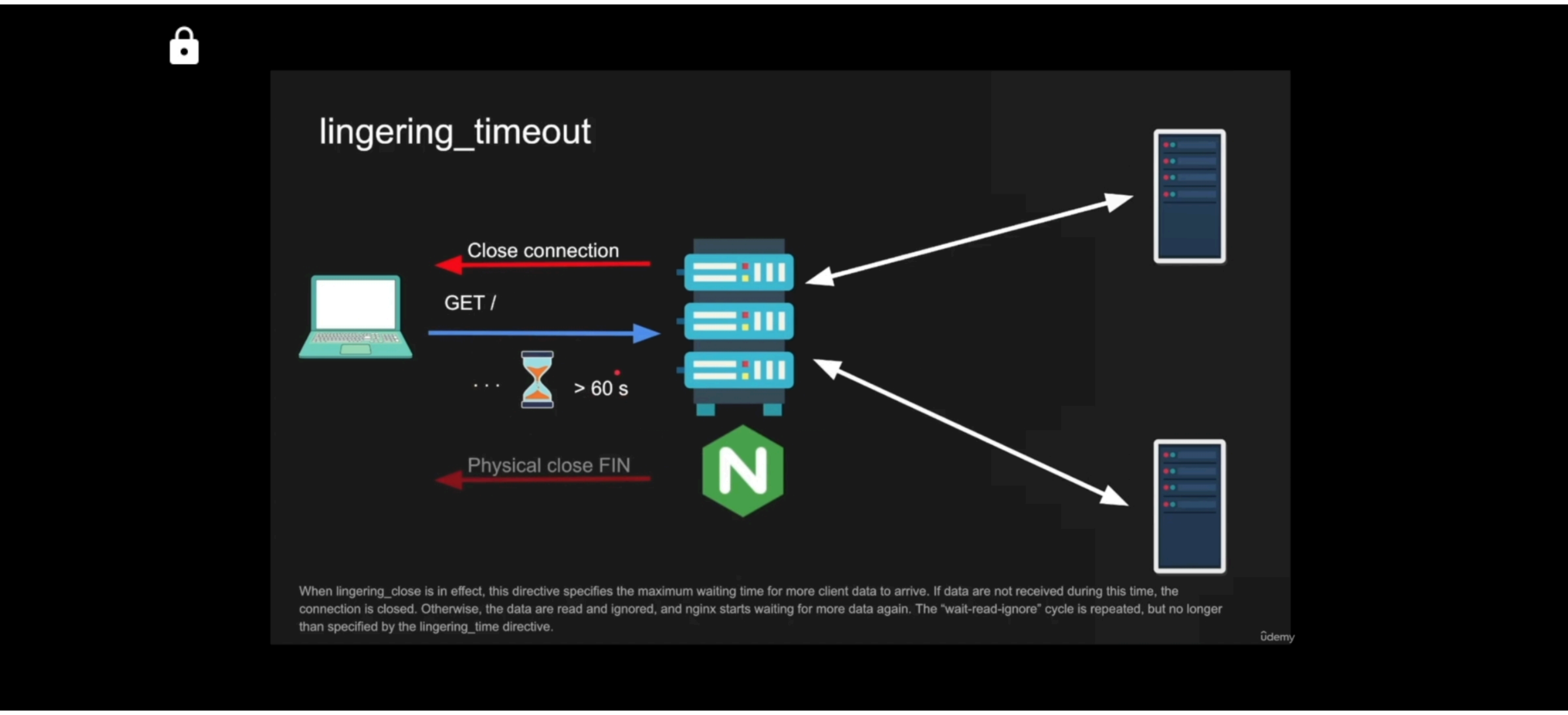
keepalive_timeout

GET / HTTP/1.1
Connection: keep-alive

. . . > 75 s

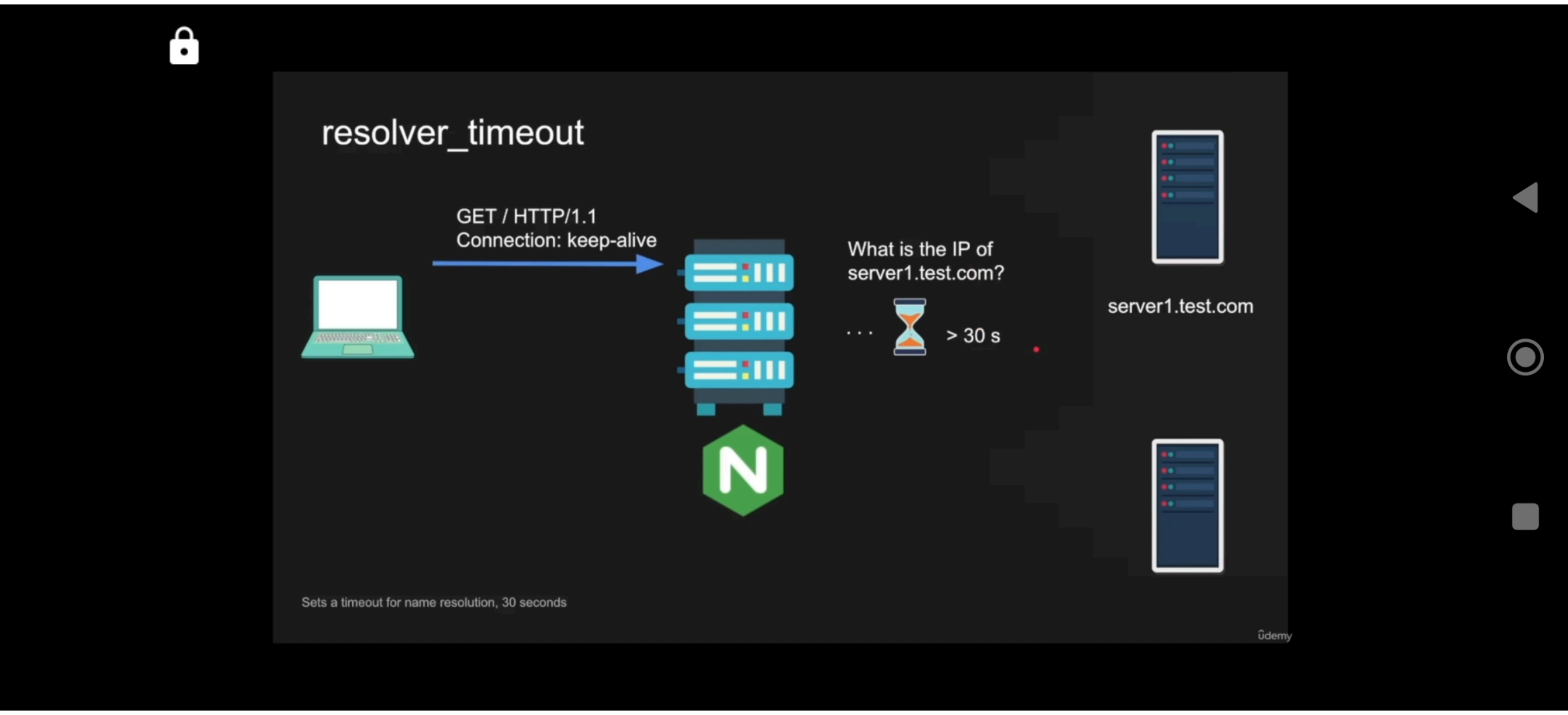
Close connection

The first parameter sets a timeout during which a keep-alive client connection will stay open on the server side. The zero value disables keep-alive client connections. The optional second parameter sets a value in the "Keep-Alive: timeout=time" response header field. Two parameters may differ. (default 75 seconds)



lingering_timeout

When lingering_close is in effect, this directive specifies the maximum waiting time for more client data to arrive. If data are not received during this time, the connection is closed. Otherwise, the data are read and ignored, and nginx starts waiting for more data again. The "wait-read-ignore" cycle is repeated, but no longer than specified by the lingering_time directive.



resolver_timeout

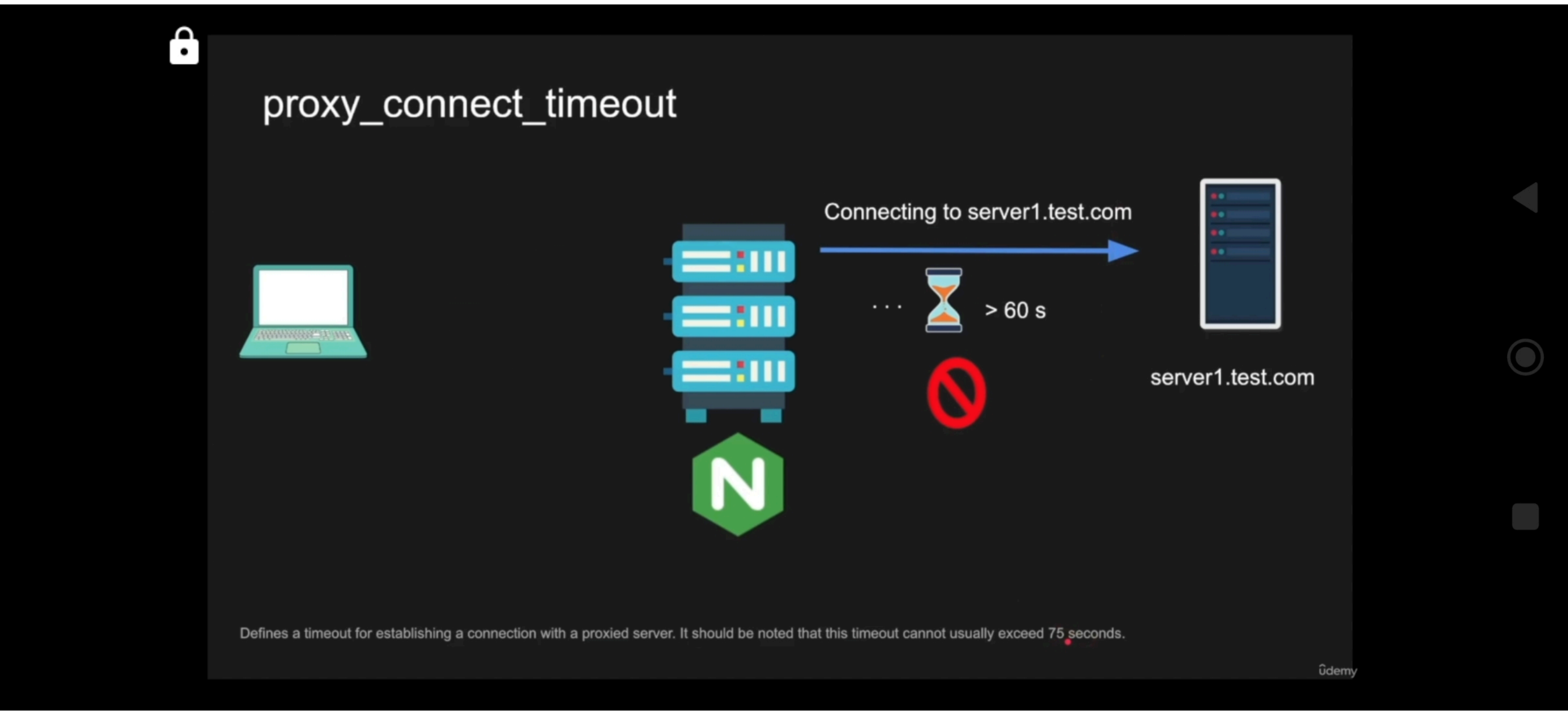
GET / HTTP/1.1
Connection: keep-alive

What is the IP of
server1.test.com?

. . . > 30 s

server1.test.com

Sets a timeout for name resolution, 30 seconds



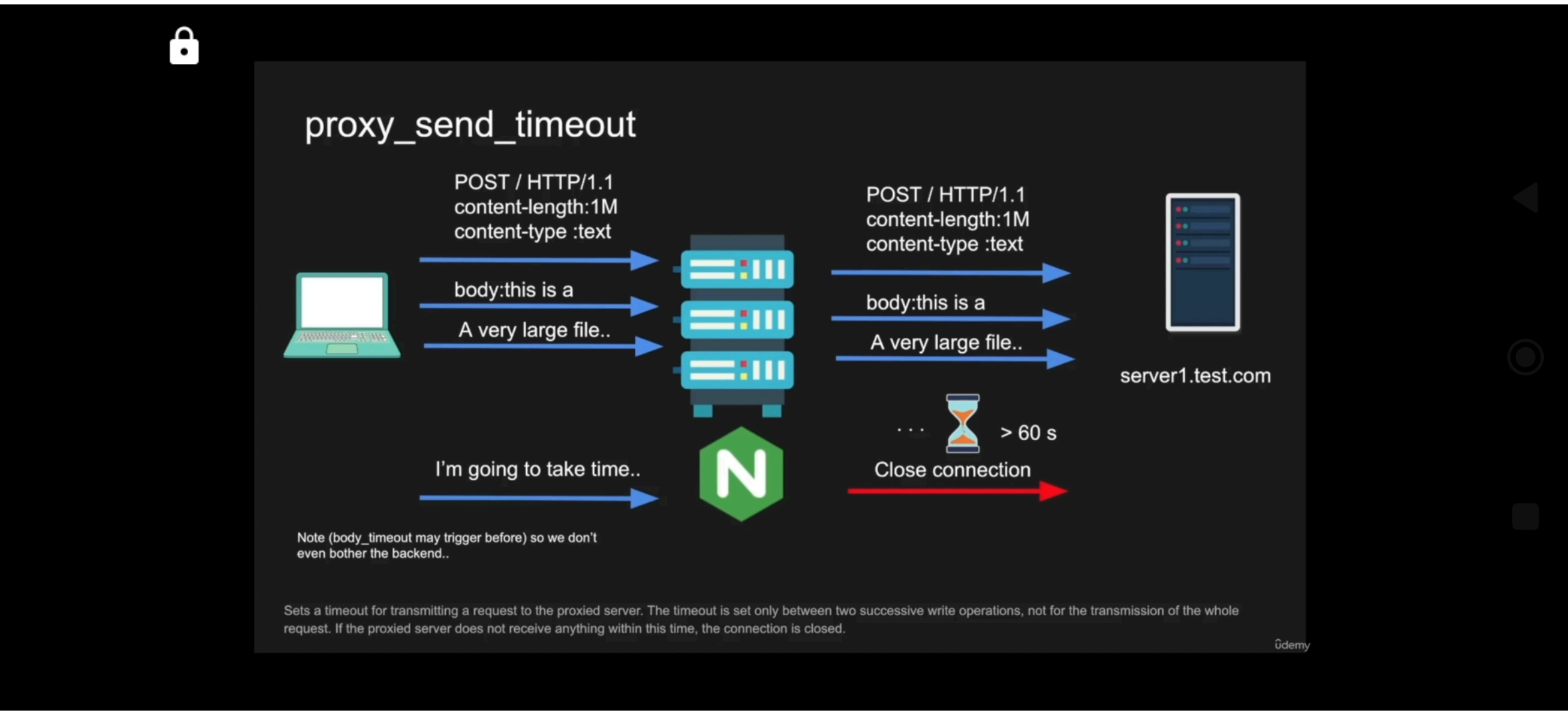
proxy_connect_timeout

Connecting to server1.test.com

· · · > 60 s

server1.test.com

Defines a timeout for establishing a connection with a proxied server. It should be noted that this timeout cannot usually exceed 75 seconds.



proxy_send_timeout

POST / HTTP/1.1
content-length:1M
content-type :text

body:this is a

A very large file..

I'm going to take time..

POST / HTTP/1.1
content-length:1M
content-type :text

body:this is a

A very large file..

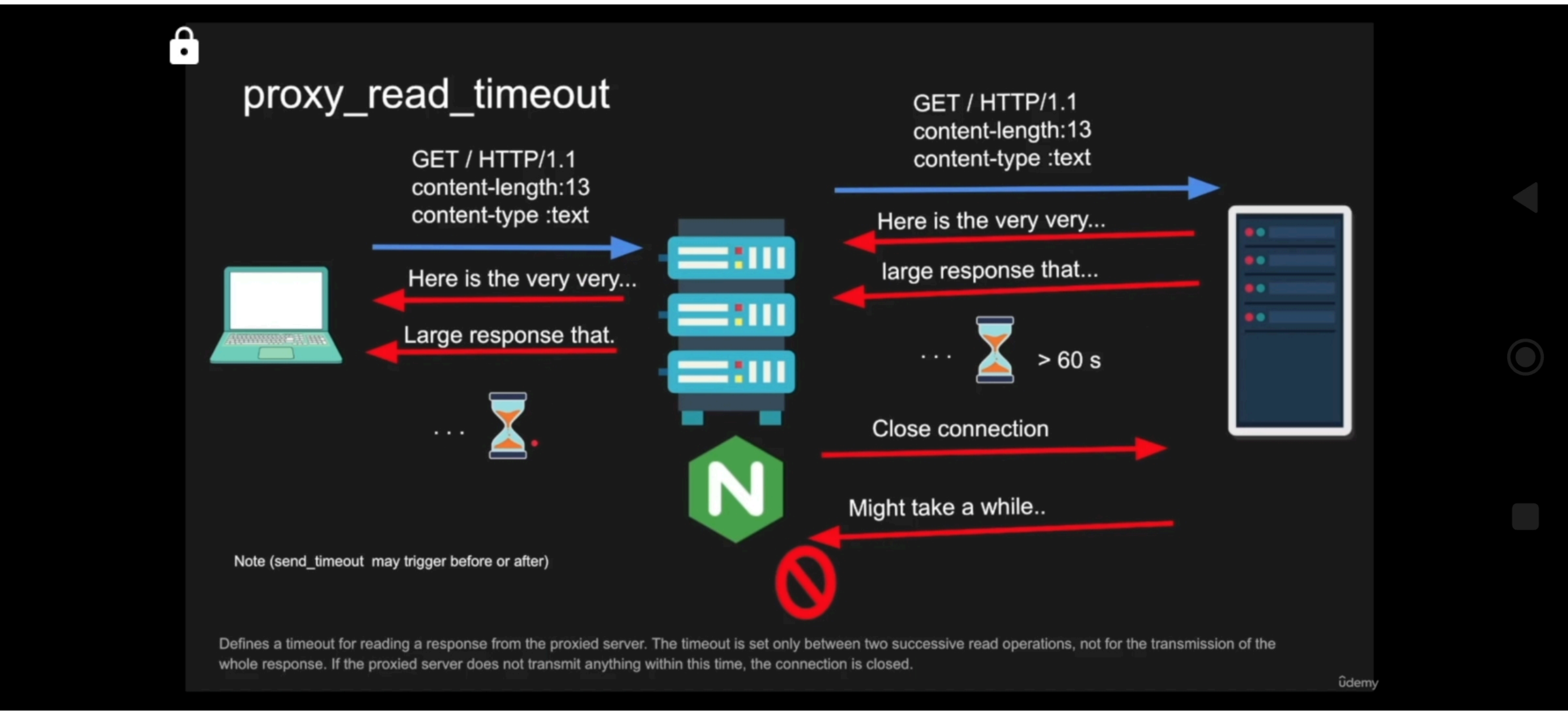
server1.test.com

. . . > 60 s

Close connection

Note (body_timeout may trigger before) so we don't even bother the backend..

Sets a timeout for transmitting a request to the proxied server. The timeout is set only between two successive write operations, not for the transmission of the whole request. If the proxied server does not receive anything within this time, the connection is closed.



proxy_read_timeout

GET / HTTP/1.1
content-length:13
content-type :text

Here is the very very...

Large response that.

. . .

GET / HTTP/1.1
content-length:13
content-type :text

Here is the very very...

large response that...

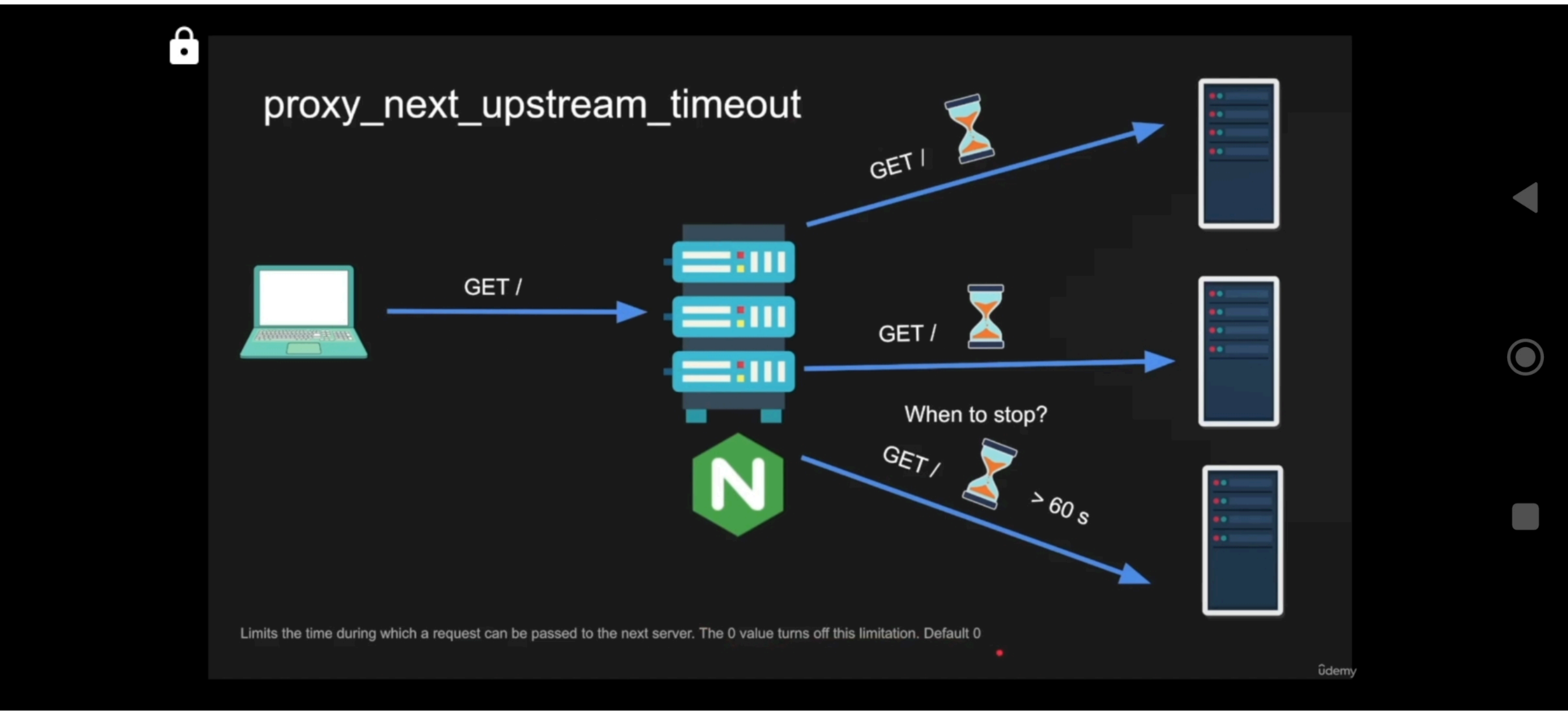
. . . > 60 s

Close connection

Might take a while..

Note (send_timeout may trigger before or after)

Defines a timeout for reading a response from the proxied server. The timeout is set only between two successive read operations, not for the transmission of the whole response. If the proxied server does not transmit anything within this time, the connection is closed.



proxy_next_upstream_timeout

GET /

GET /

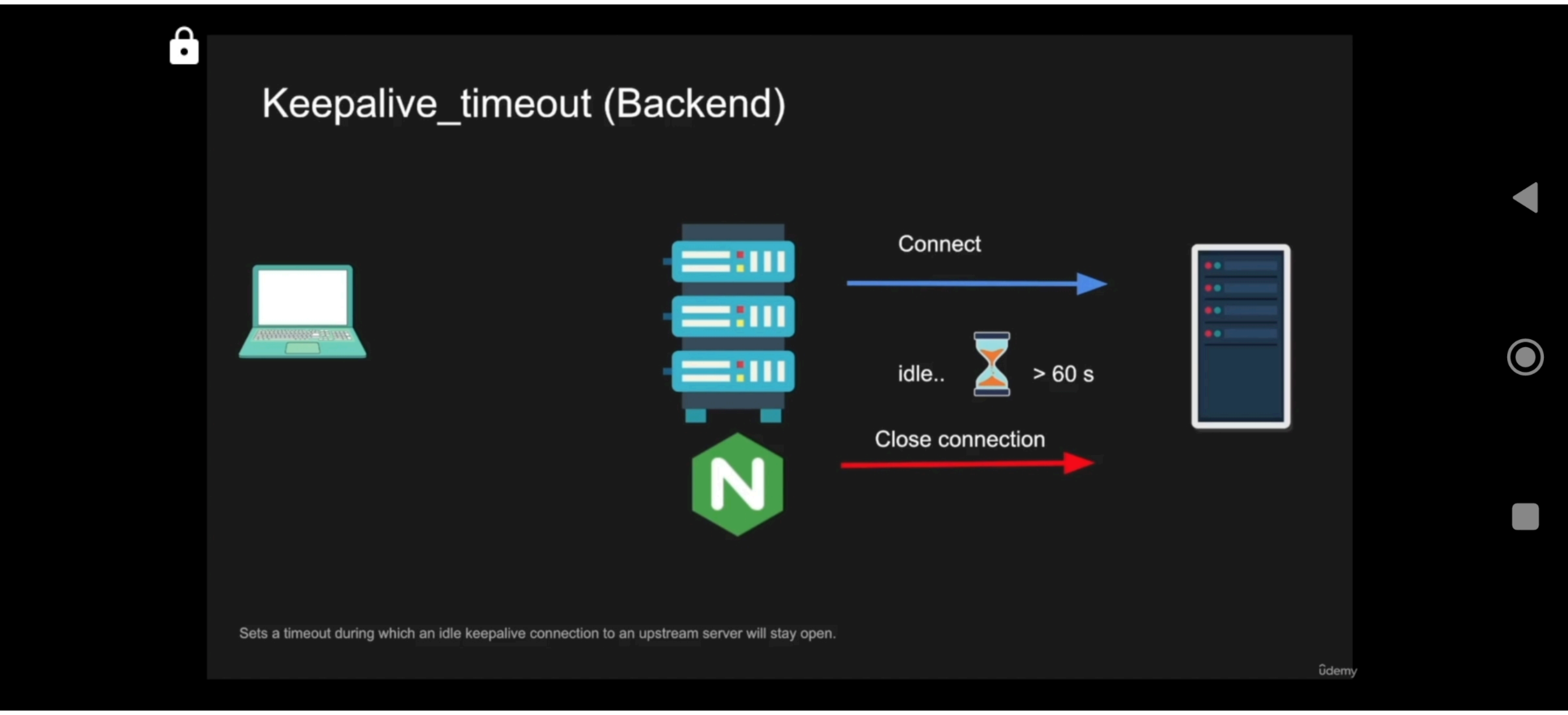
GET /

When to stop?

GET /

> 60 s

Limits the time during which a request can be passed to the next server. The 0 value turns off this limitation. Default 0



Keepalive_timeout (Backend)

Sets a timeout during which an idle keepalive connection to an upstream server will stay open.