**Name of Title:** Learning Nginx

**Video Name:** 04\_01 Reverse Proxy and Load Balancing

**Estimated Length:**

**Author Name:** Michael Jenkins

**Chapter\_Section\_Video:**

**Video Objective:**

At the end of this video the learner will know what proxying and load balancing are and how nginx is efficiently used for both.

**Speaking Points:**

1. Explain proxying
2. Explain load balancing
3. Compare proxying to load balancing
4. Give intro to upstreams and segue into next lesson

**Script:**

SLIDE: 04\_01 Reverse Proxy and Load Balancing

We know that nginx can be used as a web server for static files and as a platform for dynamic sites when paired with PHP and MySQL. Now let's take a look at two other ways nginx can be used: as a reverse proxy and a load balancer. I may refer to a reverse proxy as just proxy but know that I’m referring to the same thing.

SLIDE: Reverse Proxies and Load Balancers

Proxies and load balancers provide very similar functions. In both cases, a server running as a proxy or load balancer sits between a client and some resource that can fulfill requests from the client.

The client connects to the proxy or load balancer on the front end, then the proxy connects to the server or resource on the back end and returns the response to the client.

With proxies and load balancers providing what looks like a very similar function, You might be wondering what the difference is between the two. Let’s take a closer look.

SLIDE: Reverse Proxy

A reverse proxy applies to the case where there is only one server on the backend. This could be a web application implemented in python of Node JS for example. It could even be another web server running Apache or Tomcat.

In these cases, Nginx is great at simplifying things that might be harder to implement in other technologies like SSL termination and logging.

Nginx proxies are also good for accelerating the response from backend servers by caching content. By storing frequently requested content, Nginx can reduce the load on backend servers and respond to client requests more quickly.

Nginx also has capabilities to compress data which can reduce the amount of network bandwidth needed to fulfill a request.

SLIDE: Load Balancer

Acting as a load balancers, nginx offers the same capabilities as a reverse proxy. But instead of only connecting to one server on the back end, a load balancer connects to multiple servers.

This configuration is useful for applications that require more than one back end server to handle a large number of clients or a high volume of traffic.

SLIDE: Load Balancer 2

With more than one server, a site or application can be more reliable by continuing to respond to requests if one server goes down. This also gives site administrators some flexibility for maintenance since they can take one server offline to update software, for example, without taking the entire site offline. Depending on the needs of the application, as long as one server is online, the site will still be up.

SLIDE: Load Balancer 3

Another feature of load balancers is enabling session persistence. This is useful when clients need the same server to accept and respond to all of their requests. Let’s consider an example application that requires a client to log in and the user’s account information is stored locally on the server. If the client’s requests were routed to a different server each time, they might have to log in over and over again. Instead, session persistence routes all connections to the same server they started on until they log out.

Now that we have an understanding of proxying and load balancing, in the next lesson, we’ll discuss the upstream directive and how to define the servers that nginx connects to when acting as a reserver proxy or load balancer.

**Conclusion:**

Type your conclusion statement here.

**Script and Media:**

Break the script up into parts and align it with any media (slides, web, CLI, etc.)

| **Part** | **Script** | **Media** |
| --- | --- | --- |
|  |  |  |

**Exercise Files:**

**Basement:**

<http://nginx.org/en/docs/http/ngx_http_proxy_module.html>

<http://nginx.org/en/docs/http/load_balancing.html>

<https://www.nginx.com/resources/glossary/reverse-proxy-vs-load-balancer/>