NGINX

Web Server & Reverse Proxy

huong.nguyen@ubitec.com | **Jan-19-2020**

ubitec AG / Engineering / onebase

spa.md

Here's where you'll find this repository's source files. To give your users an idea of what they'll find here, add a description

Source ▾ master ▾ 19cece9 ▾ Full commit

onebase / docs / distro / **spa.md**

onebase/spa

Webserver for Single Page Application.

This image is OpenShift compatible.

Supported tags

- `onebase/spa:1.18.0` (basing on `nginx:1.18.0-alpine`)

Usage example

```
FROM onebase/spa:1.18.0
COPY --chown=spa:spa --from=build /apps/build /usr/share/nginx/html/
COPY --chown=spa:spa nginx.conf.d/default.conf /etc/nginx/conf.d/default.conf

USER spa
EXPOSE 8080
```

ubitec AG / infras / nginx

sites-available

master ▾   Files ▾   Filter files

nginx / conf / **sites-available**

Name	Size	Last commit	Message
..			
abs.ubitec.io.conf	6.96 KB	2020-05-18	Allow access from offices to
anchore.ubitec.io.conf	951 B	2020-03-05	Prevent anchore.ubitec.io fr
assura-ubichat.ubitec.io.conf	5.07 KB	2020-08-03	MSG-2627	new ubichat er
balboa-ng.ubitec.io.conf	3.26 KB	2020-12-10	Add HTTPS redirection for
balboa.ubitec.io.conf	13.13 KB	2020-05-18	Allow access from offices to
cdn.ubitec.io.conf	898 B	2020-08-12	Add cdn.ubitec.io for servin
cic-cob-ng.ubitec.io.conf	6.51 KB	2020-05-18	Allow access from offices to
cic.ubitec.io.conf	9.01 KB	2020-05-18	Allow access from offices to
clx-ubichat.ubitec.io.conf	3.31 KB	2020-12-14	Create domain clx-ubichat.
docker.ubitec.io.conf	5.04 KB	2020-11-09	Consolate configuration on
docs.ubitec.io.conf	1.89 KB	2020-05-18	Allow access from offices to
erups.openshift.ubitec.io.conf	3.02 KB	2020-10-30	Config nginx to include pub

Agenda

Theory (10')

- Why NGINX matter

- Using NGINX as a Web Server

- Using NGINX as a Reverse Proxy

Live demo (40')

- Make use of basic directives at localhost

- Run a website on OpenShift (optional)

- Serve the website with domain <u>ubitec.io</u> (optional)

Q&A (10')

What and Why

Interview the NGINX's author …

"When did you start working on NGINX, and why?"

"**I** started the initial work in 2002, and in **2004 I opened NGINX to the public.** Back then, I was trying to overcome certain barriers of scaling the web infrastructure of a large online media company I worked for. In particular, the **difficulties of handling many concurrent connections, reducing latency** and offloading static content, SSL and persistent connections were my main interest. **There weren't any reliable production quality web server software to crack so-called C10K problem** (handling of at least 10,000 of concurrent connections, outlined by Dan Kegel). So in a sense I decided to solve both practical and "academic" problems. I was very curious and excited to try it out and I'm really happy it turned to be a successful attempt and that **NGINX is now used by the over 50,000,000 web sites** on the Internet"

Igor Sysoev

Web Server

NGINX Documentation

Web Server

```
http {
    server {
        listen 127.0.0.1:8080;
        server_name localhost;
        set $default_server huong.io;

        root html;

        location / {
            sub_filter "<h1>Welcome to nginx!</h1>" "<h1>Welcome to NGINX!</h1>";
        }

        location ~ \.js? {
            return 403;
        }

        location /users/ {
            rewrite ^/users/(.*)$ /show?user=$1;
        }

        location /show {
            return 302 http://$default_server;
        }

        error_page 404 /404.html;
    }
}
```
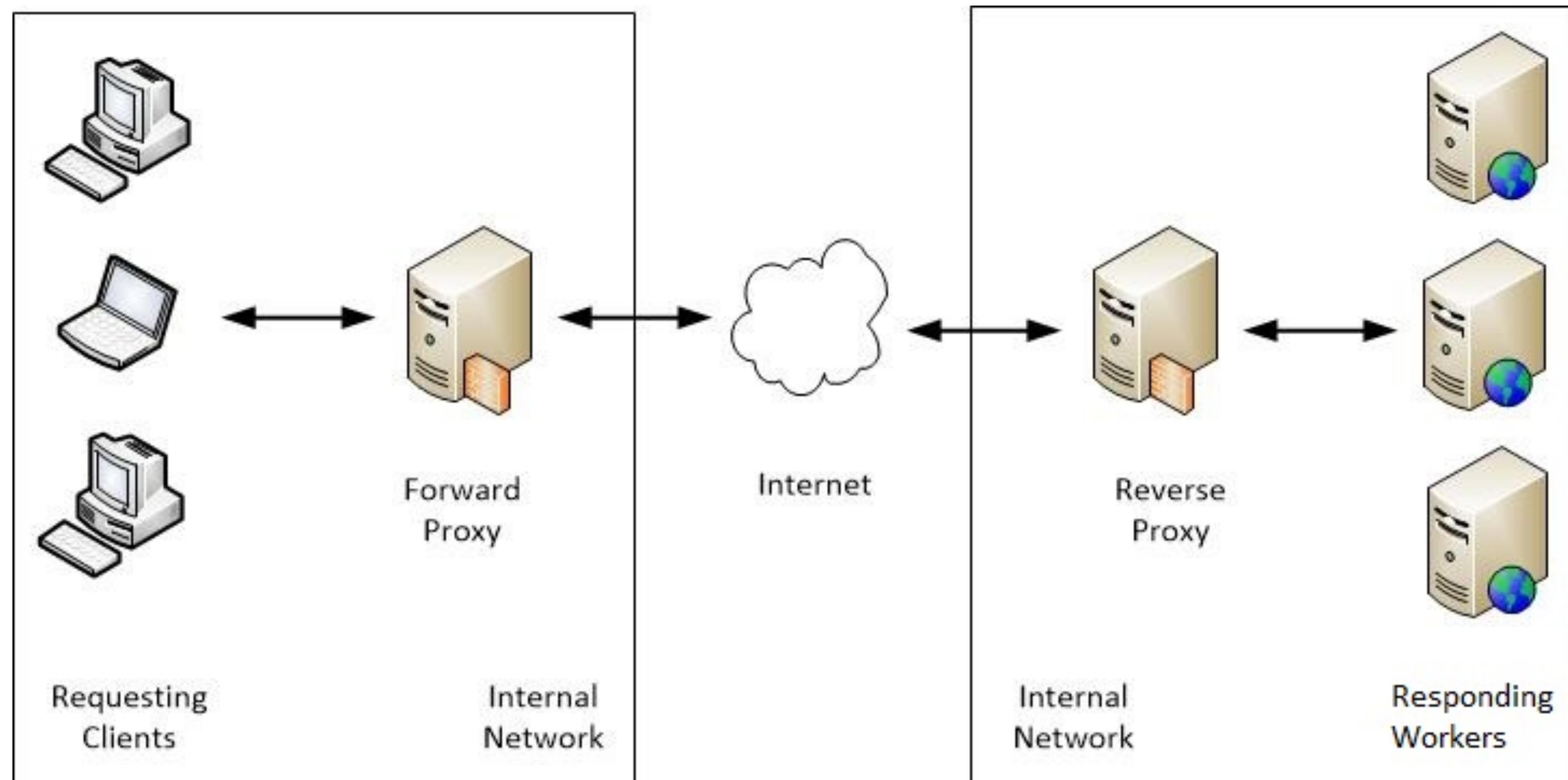
Reverse Proxy

Forward Proxy

Internet

Reverse Proxy

Requesting Clients

Internal Network

Internal Network

Responding Workers

https://docs.microsoft.com/en-us/iis/extensions/configuring-application-request-routing-arr/creating-a-forward-proxy-using-application-request-routing

Reverse proxies have several use cases, a few are:

- **Load balancing**: distribute the load to several web servers,
- **Cache static content**: offload the web servers by caching static content like pictures,
- **Compression**: compress and optimize content to speed up load time.

```
http {
    upstream backend {
        server node01.example.com:3000;
        server node02.example.com:3001;
    }

    server {
        listen 8080;
        server_name api.example.com;

        location / {
            proxy_set_header X-Real-IP $remote_addr;
            proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
            proxy_set_header X-Forwarded-Proto $scheme;
            proxy_pass http://backend;
        }
    }
}
```

Live Demo

References

- https://en.wikipedia.org/wiki/Nginx

- https://docs.nginx.com/nginx/admin-guide/web-server

- https://developer.mozilla.org/en-US/docs/Web/HTTP/Proxy_servers_and_tunneling

- https://docs.microsoft.com/en-us/iis/extensions/configuring-application-request-routing-arr/creating-a-forward-proxy-using-application-request-routing

- https://web.archive.org/web/20131019145106/http://www.freesoftwaremagazine.com/articles/interview_igor_sysoev_author_apaches_competitor_nginx

- http://www.aosabook.org/en/nginx.html

Thank You!