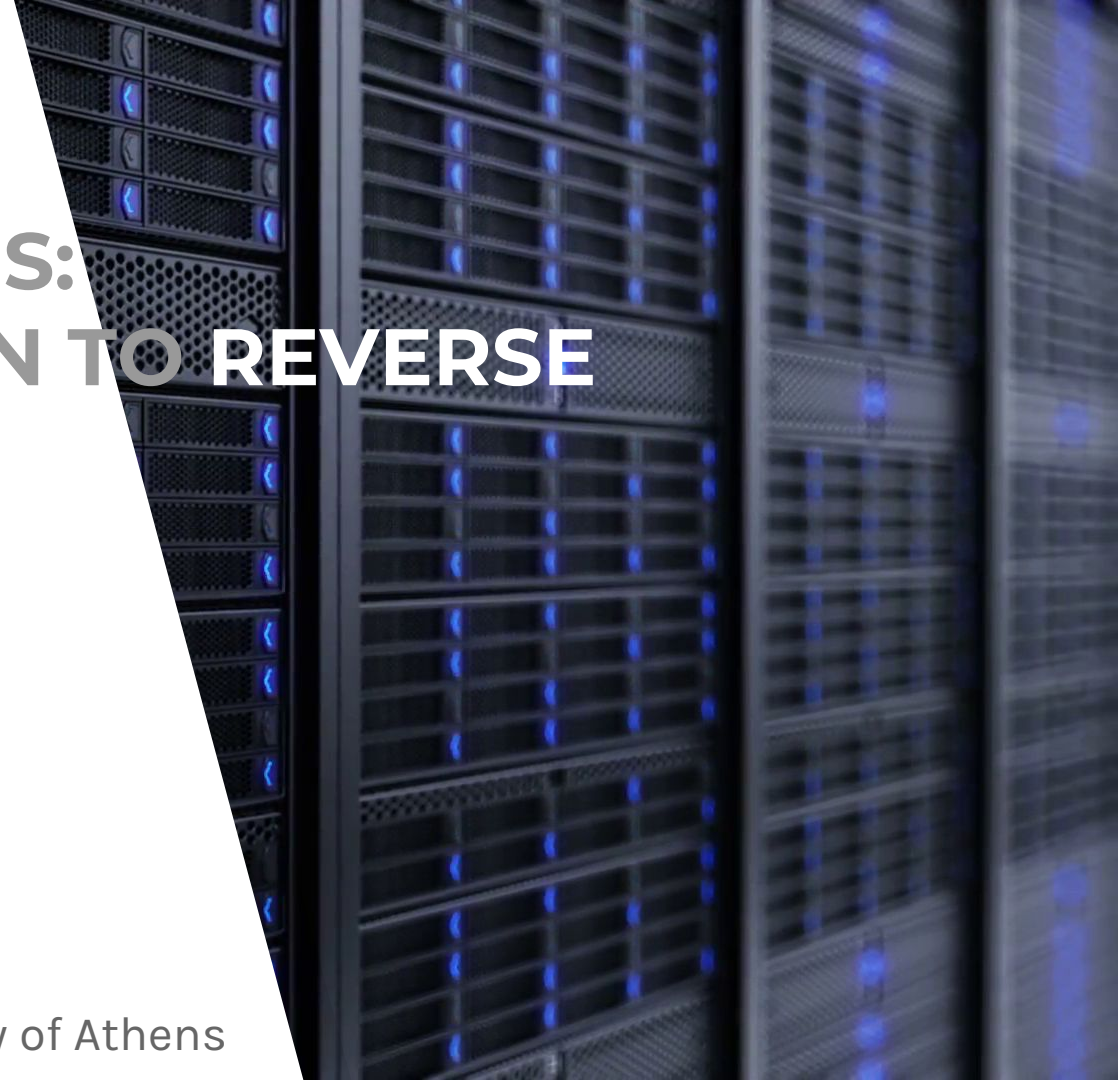
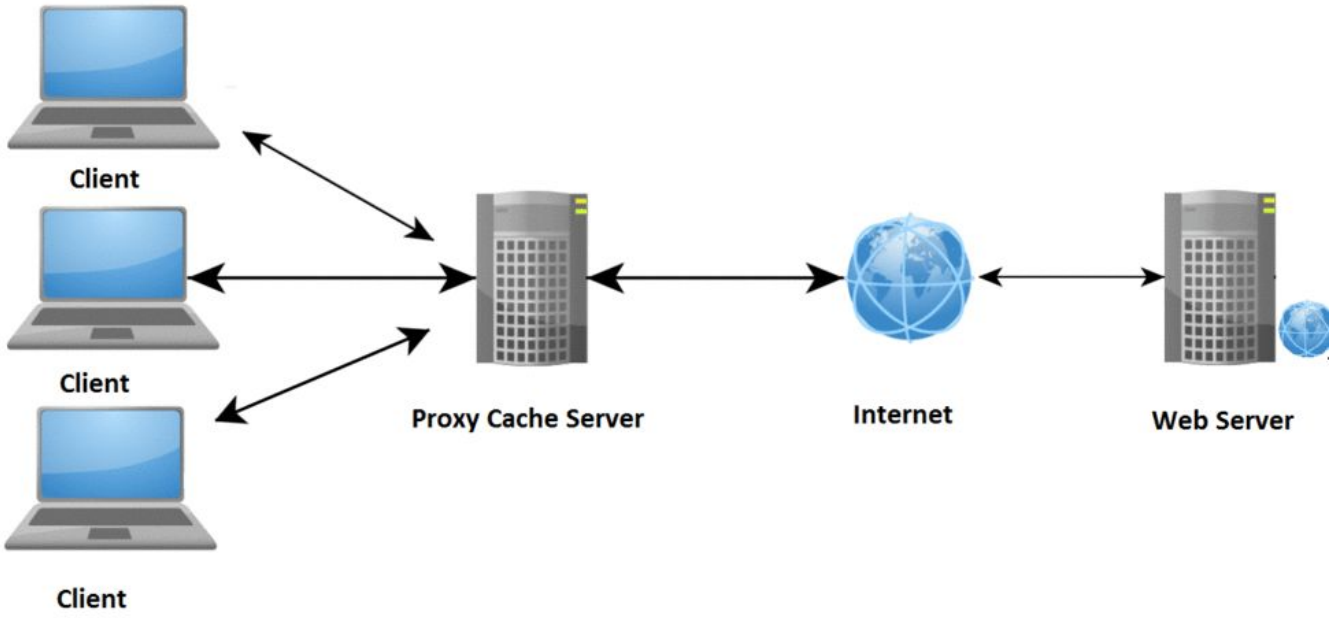
LAYERING DEVICES: AN INTRODUCTION TO REVERSE PROXIES



Dimitra Mavroforaki
National & Kapodistrian University of Athens

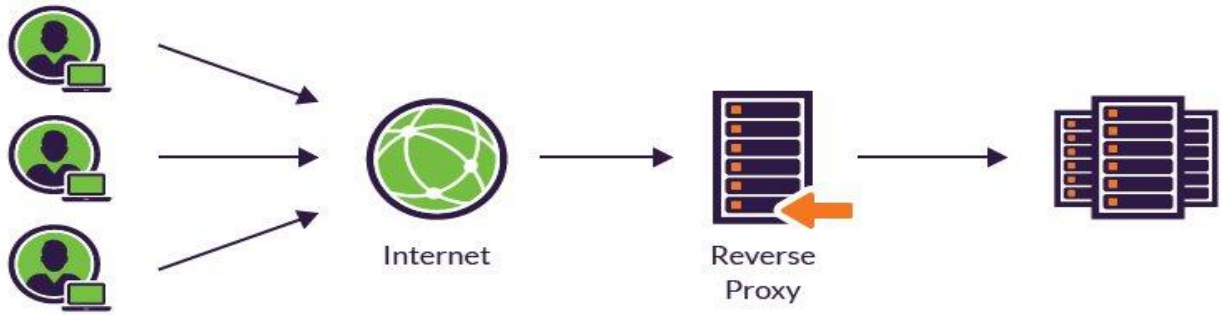
To begin with:

What is actually a forward proxy server?

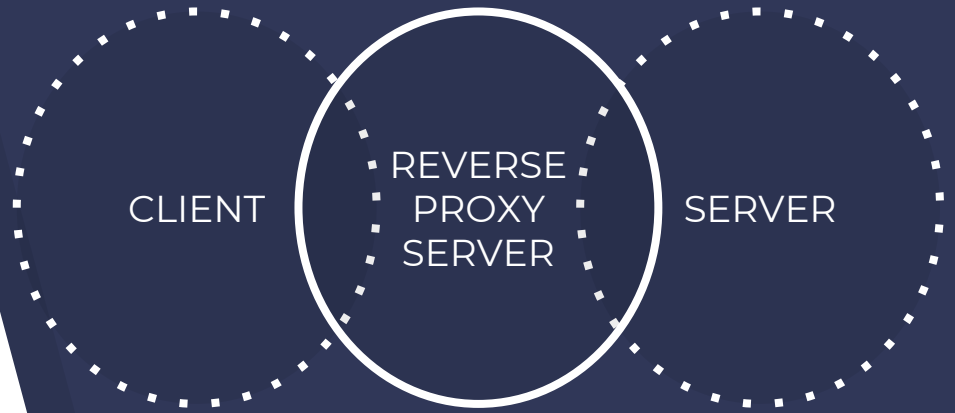


Reverse Proxy:

- Appears as an **ordinary** server in a network of web servers
- Is an extra layer so that the client has **no knowledge** of the origin server



How it works ?



Step 1:

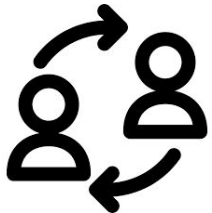
Client starts a connection with the proxy using a connection protocol

Step 2:

Proxy establishes the connection between the client and the main application server

Step 3:

- The client's output stream ends up to the backend server
- The output stream of the server ends up to the client



Why should we use reverse proxies ?

Load Balancing

A reverse proxy :

- Distributes requests → backend servers are not overloaded
- Controls traffic → rewrites url to connect to the desired server

Web Acceleration

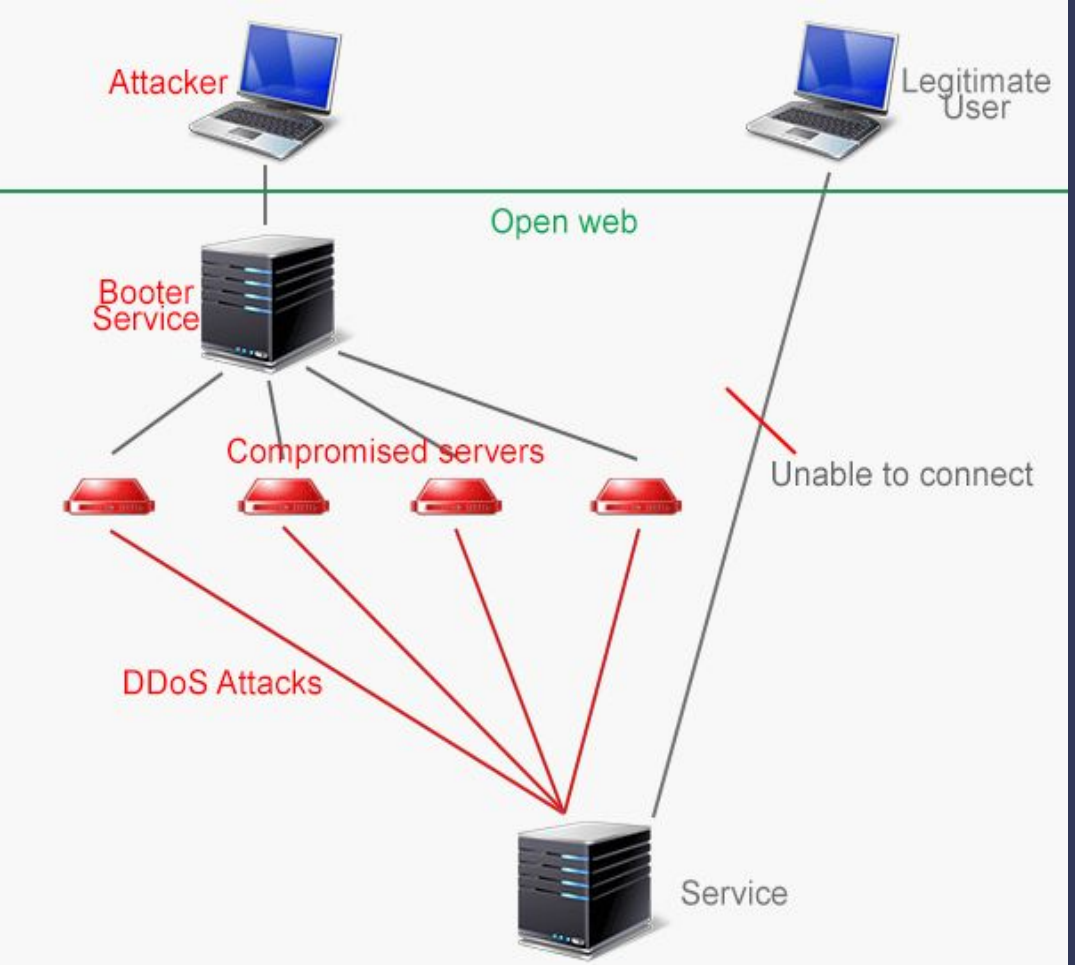
- Reverse proxy performs tasks to boost the performance of the main application server
  - Caching
  - Compressing inbound and outbound data
  - SSL encryption
  - A/B testing

Security / Anonimity

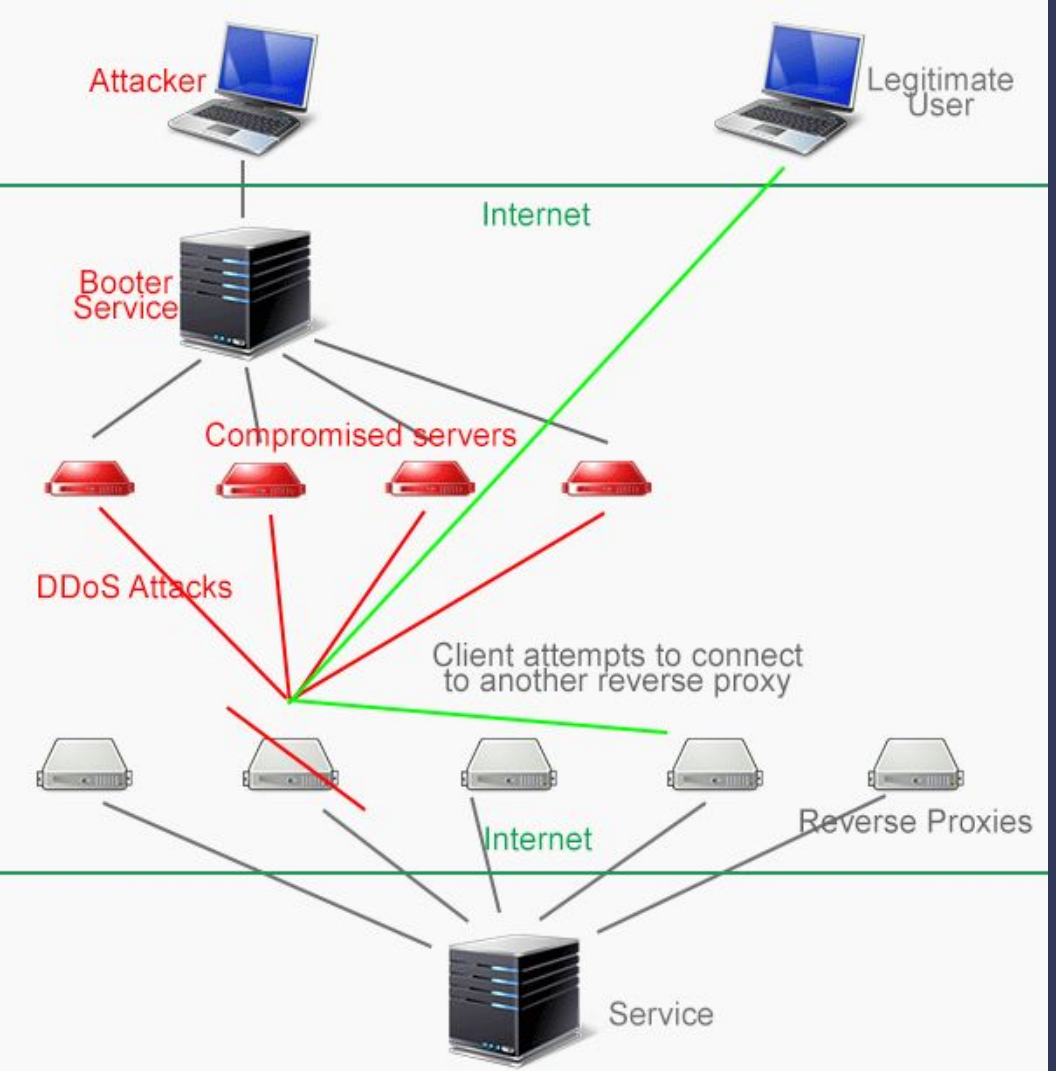
- The local network is protected from an additional layer of servers

A reverse proxy:

- Hides main application server’s identity
- Ensures that multiple servers can be accessed from a single url



Reverse proxies is a preventive measure against **DDoS attacks**



- Attackers are only able to see the IP of the proxy server so they cannot target a backend server
- There is always a server available to route and serve a client's request

SETTING UP A REVERSE PROXY

Main Goal

I want to set up an apache2 server as reverse proxy for an application that has Apache Tomcat as main application server.

Application’s URL without reverse proxy :
localhost:8080/

Application’s URL with reverse proxy:

localhost:/

1. Download apache

```
dimitra@dimitra-Lenovo-U310 /etc/apache2 $ cd sites-available/
dimitra@dimitra-Lenovo-U310 /etc/apache2/sites-available $ ls
000-default.conf  default-ssl.conf
```

2. Edit configuration files

Sites-enabled folder:
Contains all the rules applied to the websites (read-only)

Sites-available folder:

Specify the rules needed (editable)

```
<VirtualHost *:80>
        SSLProxyEngine on
        ProxyPass / https://localhost:8080/#/
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf

</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

3. Activate modules

```
dimitra@dimitra-Lenovo-U310 /etc/apache2/sites-enabled $ a2enmod
Your choices are: access_compat actions alias allowmethods asis auth_basic auth_digest auth_form authn_anon authn_core authn_dbd authn_dbm auth
n_file authn_socache authnz_fcgi authnz_ldap authz_core authz_dbd authz_dbm authz_groupfile authz_host authz_owner authz_user autoindex buffer
cache cache_disk cache_socache cgi cgid charset_lite data dav dav_fs dav_lock dbd deflate dialup dir dump_io echo env expires ext_filter file_c
ache filter headers heartbeat heartmonitor ident include info lbmethod_bybusyness lbmethod_byrequests lbmethod_bytraffic lbmethod_heartbeat lda
p log_debug log_forensic lua macro mime mime_magic mpm_event mpm_prefork mpm_worker negotiation php7.0 proxy proxy_ajp proxy_balancer proxy_con
nect proxy_express proxy_fcgi proxy_fdpass proxy_ftp proxy_html proxy_http proxy_scgi proxy_wstunnel ratelimit reflector remoteip reqtimeout re
quest rewrite sed session session_cookie session_crypto session_dbd setenvif slotmem_plain slotmem_shm socache_dbm socache_memcache socache_shm
cb speling ssl status substitute suexec unique_id userdir usertrack vhost_alias xml2enc
Which module(s) do you want to enable (wildcards ok)?
```

4. Restart apache

We have set up our own
REVERSE PROXY !

REFERENCES

https://www.youtube.com/watch?v=ozhe___GdWC8
https://en.wikipedia.org/wiki/Proxy_server
https://www.nginx.com/resources/glossary/reverse-proxy-server/
https://www.youtube.com/watch?v=Dgf9uBDX0-g
https://www.youtube.com/watch?v=9rOU_P5YN_Q
https://www.slideshare.net/proxiesforrent/reverse-proxy
https://www.youtube.com/watch?v=A3Prx_2YEm8
https://www.ericzhang.me/reverse-proxy-ddos-protection/

THANKS FOR YOUR TIME!

Any questions?