

**ATTACK**

**DEFENSE**

by PentesterAcademy

<b>Name</b>	Nginx Software
<b>URL</b>	<a href="https://attackdefense.com/challengedetails?cid=2038">https://attackdefense.com/challengedetails?cid=2038</a>
<b>Type</b>	DevOps Basics: Building Project

**Important Note:** This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

## Challenge Description

The project building process is to get the software/webapp ready for deployment. The build process depends on the project and technology/language/framework used.

A Kali machine is provided with make utility and project dependencies installed on it. The source code of Nginx webserver is provided in the home directory of the root user.

**Objective:** Configure, build the Nginx from source code and run it on the local machine!

### Instructions:

- The source code of web applications is provided at /root/github-repos

## Solution

**Step 1:** Check the contents of the current working directory.

**Command:** ls -l

```
root@attackdefense:~# ls -l
total 16
drwxr-xr-x 3 root root 4096 Sep 18 07:42 github-repos
-rw-r--r-- 1 root root 293 Feb 7 2019 README
drwxr-xr-x 1 root root 4096 Nov 21 2019 tools
drwxr-xr-x 2 root root 4096 Feb 7 2019 wordlists
root@attackdefense:~#
```

**Step 2:** Change to the nginx directory and check its contents.

**Commands:**

```
cd github-repos/nginx-1.18.0/
```

```
ls -l
```

```
root@attackdefense:~# cd github-repos/nginx-1.18.0/
root@attackdefense:~/github-repos/nginx-1.18.0#
root@attackdefense:~/github-repos/nginx-1.18.0# ls -l
total 784
drwxr-xr-x 6 root root 4096 Sep 18 07:40 auto
-rw-r--r-- 1 root root 302863 Apr 21 14:09 CHANGES
-rw-r--r-- 1 root root 462213 Apr 21 14:09 CHANGES.ru
drwxr-xr-x 2 root root 4096 Sep 18 07:40 conf
-rwxr-xr-x 1 root root 2502 Apr 21 14:09 configure
drwxr-xr-x 4 root root 4096 Sep 18 07:40 contrib
drwxr-xr-x 2 root root 4096 Sep 18 07:40 html
-rw-r--r-- 1 root root 1397 Apr 21 14:09 LICENSE
drwxr-xr-x 2 root root 4096 Sep 18 07:40 man
-rw-r--r-- 1 root root 49 Apr 21 14:09 README
drwxr-xr-x 9 root root 4096 Sep 18 07:40 src
root@attackdefense:~/github-repos/nginx-1.18.0#
```

In the list, one can observe the “configure” executable file but there is no Makefile.

**Step 3:** Run the configure executable file. This script will run checks on the local machine (e.g. architecture, install dependencies, etc) and then generate a Makefile.

**Command:** ./configure

```
root@attackdefense:~/github-repos/nginx-1.18.0# ./configure
checking for OS
+ Linux 4.15.0-72-generic x86_64
checking for C compiler ... found
+ using GNU C compiler
+ gcc version: 9.3.0 (Debian 9.3.0-15)
checking for gcc -pipe switch ... found
checking for -Wl,-E switch ... found
checking for gcc builtin atomic operations ... found
checking for C99 variadic macros ... found
```



```
Configuration summary
+ using system PCRE library
+ OpenSSL library is not used
+ using system zlib library

nginx path prefix: "/usr/local/nginx"
nginx binary file: "/usr/local/nginx/sbin/nginx"
nginx modules path: "/usr/local/nginx/modules"
nginx configuration prefix: "/usr/local/nginx/conf"
nginx configuration file: "/usr/local/nginx/conf/nginx.conf"
nginx pid file: "/usr/local/nginx/logs/nginx.pid"
nginx error log file: "/usr/local/nginx/logs/error.log"
nginx http access log file: "/usr/local/nginx/logs/access.log"
nginx http client request body temporary files: "client_body_temp"
nginx http proxy temporary files: "proxy_temp"
nginx http fastcgi temporary files: "fastcgi_temp"
nginx http uwsgi temporary files: "uwsgi_temp"
nginx http scgi temporary files: "scgi_temp"

root@attackdefense:~/github-repos/nginx-1.18.0#
```

On successful execution of the configure script, shows summary at the end. In case of failure, check the reason/error and try to fix that. Check the contents of the directory again. This time, the Makefile file will be present.

**Command:** ls -l

```
root@attackdefense:~/github-repos/nginx-1.18.0# ls -l
total 792
drwxr-xr-x 6 root root  4096 Sep 18 07:40 auto
-rw-r--r-- 1 root root 302863 Apr 21 14:09 CHANGES
-rw-r--r-- 1 root root 462213 Apr 21 14:09 CHANGES.ru
drwxr-xr-x 2 root root  4096 Sep 18 07:40 conf
-rwxr-xr-x 1 root root  2502 Apr 21 14:09 configure
drwxr-xr-x 4 root root  4096 Sep 18 07:40 contrib
drwxr-xr-x 2 root root  4096 Sep 18 07:40 html
-rw-r--r-- 1 root root  1397 Apr 21 14:09 LICENSE
-rw-r--r-- 1 root root   376 Sep 22 07:33 Makefile
drwxr-xr-x 2 root root  4096 Sep 18 07:40 man
drwxr-xr-x 3 root root  4096 Sep 22 07:33 objs
-rw-r--r-- 1 root root    49 Apr 21 14:09 README
drwxr-xr-x 9 root root  4096 Sep 18 07:40 src
root@attackdefense:~/github-repos/nginx-1.18.0#
```

**Step 4:** Compile the code and create the Nginx binary by using make command. The make command reads Makefile in order to compile the Nginx.

**Command:** make

```
root@attackdefense:~/github-repos/nginx-1.18.0# make
make -f objs/Makefile
make[1]: Entering directory '/root/github-repos/nginx-1.18.0'
cc -c -pipe -O -W -Wall -Wpointer-arith -Wno-unused-parameter -Werror -g \
    -o objs/src/core/nginx.o \
    src/core/nginx.c
cc -c -pipe -O -W -Wall -Wpointer-arith -Wno-unused-parameter -Werror -g \
    -o objs/src/core/nginx_log.o \
    src/core/nginx_log.c
cc -c -pipe -O -W -Wall -Wpointer-arith -Wno-unused-parameter -Werror -g \
    objs/src/http/modules/nginx_http_upstream_zone_module.o \
    objs/nginx_modules.o \
    -ldl -lpthread -lcrypt -lpcrc -lz \
    -Wl,-E
sed -e "s|%%PREFIX%%|/usr/local/nginx|" \
    -e "s|%%PID_PATH%%|/usr/local/nginx/logs/nginx.pid|" \
    -e "s|%%CONF_PATH%%|/usr/local/nginx/conf/nginx.conf|" \
    -e "s|%%ERROR_LOG_PATH%%|/usr/local/nginx/logs/error.log|" \
    < man/nginx.8 > objs/nginx.8
make[1]: Leaving directory '/root/github-repos/nginx-1.18.0'
root@attackdefense:~/github-repos/nginx-1.18.0#
```

The Nginx binary and configuration files are now ready on the machine.

**Step 5:** Check the generated nginx object file and run it to check the version.

**Commands:**

```
ls -l objs/nginx
objs/nginx -v
```



```
root@attackdefense:~/github-repos/nginx-1.18.0# ls -l objs/nginx
-rwxr-xr-x 1 root root 5179616 Sep 22 07:35 objs/nginx
root@attackdefense:~/github-repos/nginx-1.18.0#
root@attackdefense:~/github-repos/nginx-1.18.0#
root@attackdefense:~/github-repos/nginx-1.18.0# objs/nginx -v
nginx version: nginx/1.18.0
root@attackdefense:~/github-repos/nginx-1.18.0#
```

**Step 6:** If we try to run this binary then it will throw some error.

**Commands:** objs/nginx

```
root@attackdefense:~/github-repos/nginx-1.18.0# objs/nginx
nginx: [alert] could not open error log file: open() "/usr/local/nginx/logs/error.log" failed (2: No such file or directory)
2020/09/22 07:35:53 [emerg] 2598#0: open() "/usr/local/nginx/conf/nginx.conf" failed (2: No such file or directory)
root@attackdefense:~/github-repos/nginx-1.18.0#
```

**Step 7:** These errors are self explanatory. The nginx binary is looking for these two files which are not currently available. Fix the issues and run then binary again by using following commands.

**Commands:**

```
mkdir -p /usr/local/nginx/
cp -a conf /usr/local/nginx/
mkdir -p /usr/local/nginx/logs/
objs/nginx
```

```
root@attackdefense:~/github-repos/nginx-1.18.0# mkdir -p /usr/local/nginx/
root@attackdefense:~/github-repos/nginx-1.18.0# cp -a conf /usr/local/nginx/
root@attackdefense:~/github-repos/nginx-1.18.0#
root@attackdefense:~/github-repos/nginx-1.18.0# mkdir -p /usr/local/nginx/logs/
root@attackdefense:~/github-repos/nginx-1.18.0#
root@attackdefense:~/github-repos/nginx-1.18.0#
root@attackdefense:~/github-repos/nginx-1.18.0# objs/nginx
```

**Step 8:** Nginx process can be observed in the process listing.

**Commands:** ps -ef

```
root@attackdefense:~/github-repos/nginx-1.18.0# ps -ef
UID          PID    PPID  C STIME TTY          TIME CMD
root           1        0  0  07:28 ?           00:00:00 /bin/bash /startup.sh
root           6         1  0  07:28 ?           00:00:01 /usr/local/bin/ttyd -p 45654 bash
root           7         6  0  07:29 pts/0       00:00:00 bash
root        2611         1  0  07:38 ?           00:00:00 nginx: master process objs/nginx
nobody       2612      2611  0  07:38 ?           00:00:00 nginx: worker process
root        2613         7  0  07:38 pts/0       00:00:00 ps -ef
root@attackdefense:~/github-repos/nginx-1.18.0#
```

**Step 9:** Check if Nginx is listening on local port 80 by making a request with curl.

**Commands:** curl localhost:80

```
root@attackdefense:~/github-repos/nginx-1.18.0# curl localhost:80
<html>
<head><title>404 Not Found</title></head>
<body>
<center><h1>404 Not Found</h1></center>
<hr><center>nginx/1.18.0</center>
</body>
</html>
root@attackdefense:~/github-repos/nginx-1.18.0#
```

## Learning

Building and running Nginx webserver