Latest FTP Vulnerabilities

In discussing the latest vulnerabilities, we will focus this section and the following ones on one of the previously shown attacks and present it as simply as possible without going into too much technical detail. This should help us facilitate the concept of the attack through an example related to a specific service to gain a better understanding.

In this case, we will discuss the `CoreFTP before build 727` vulnerability assigned CVE-2022-22836. This vulnerability is for an FTP service that does not correctly process the `HTTP PUT` request and leads to an `authenticated directory/path traversal`, and `arbitrary file write` vulnerability. This vulnerability allows us to write files outside the directory to which the service has access.

The Concept of the Attack

This FTP service uses an HTTP `POST` request to upload files. However, the CoreFTP service allows an HTTP `PUT` request, which we can use to write content to files. Let's have a look at the attack based on our concept. The exploit for this attack is relatively straightforward, based on a single `cURL` command.

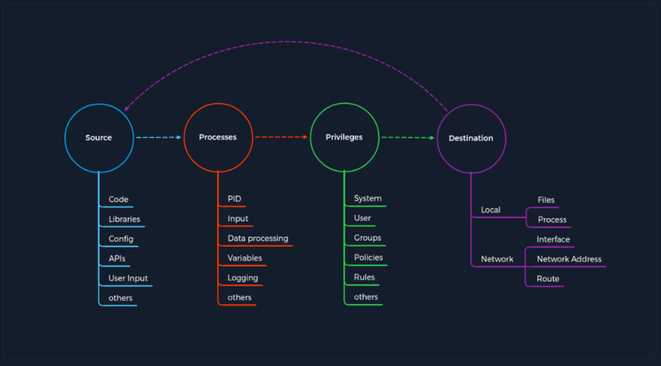
CoreFTP Exploitation

CoreFTP Exploitation

```
ipp@htb[/htb]$ curl -k -X PUT -H "Host: <IP>" --basic -u <username>:<password> --data-binary
```

We create a raw HTTP `PUT` request (`-X PUT`) with basic auth (`--basic -u <username>:<password>`), the path for the file (`--path-as-is https://<IP>/../../../../../whoops`), and its content (`--data-binary "PoC."`) with this command. Additionally, we specify the host header (`-H "Host: <IP>"`) with the IP address of our target system.

The Concept of Attacks



In short, the actual process misinterprets the user's input of the path. This leads to access to the restricted folder being bypassed. As a result, the write permissions on the HTTP `PUT` request are not adequately controlled, which leads to us being able to create the files we want outside of the authorized folders. However, we will skip the explanation of the `Basic Auth` process and jump directly to the first part of the exploit.

Directory Traversal

Step	Directory Traversal	Concept of Attacks - Category
1.	The user specifies the type of HTTP request with the file's content, including escaping characters to break out of the restricted area.	`Source`
2.	The changed type of HTTP request, file contents, and path entered by the user are taken over and processed by the process.	`Process`
3.	The application checks whether the user is authorized to be in the specified path. Since the restrictions only apply to a specific folder, all permissions granted to it are bypassed as it breaks out of that folder using the directory traversal.	`Privileges`
4.	The destination is another process that has the task of writing the specified contents of the user on the local system.	`Destination`

Up to this point, we have bypassed the constraints imposed by the application using the escape characters (`../../../../`) and come to the second part, where the process writes the contents we specify to a file of our choice. This is when the cycle starts all over again, but this time to write contents to the target system.

Arbitary File Write

Step	Arbitary File Write	Concept of Attacks - Category
5.	The same information that the user entered is used as the source. In this case, the filename (`whoops`) and the contents (`--data-binary "PoC."`).	`Source`
6.	The process takes the specified information and proceeds to write the desired content to the specified file.	`Process`
7.	Since all restrictions were bypassed during the directory traversal vulnerability, the service approves writing the contents to the specified file.	`Privileges`
8.	The filename specified by the user (`whoops`) with the desired content (`"PoC."`) now serves as the destination on the local system.	`Destination`

After the task has been completed, we will be able to find this file with the corresponding contents on the target system.

Target System

Target System

```
C:\> type C:\whoops

PoC.
```

Previous Next

Mark Complete & Next

Cheat Sheet

Resources

Table of Contents

Introduction

Interacting with Common Services

Protocol Specific Attacks

The Concept of Attacks

Service Misconfigurations

Finding Sensitive Information

FTP

Attacking FTP

Latest FTP Vulnerabilities

SMB

Attacking SMB

Latest SMB Vulnerabilities

SQL Databases

Attacking SQL Databases

Latest SQL Vulnerabilities

RDP

Attacking RDP

Latest RDP Vulnerabilities

DNS

Attacking DNS

Latest DNS Vulnerabilities

SMTP

Attacking Email Services

Latest Email Service Vulnerabilities

Skills Assessment

Attacking Common Services - Easy

Attacking Common Services - Medium

Attacking Common Services - Hard

My Workstation

OFFLINE

Start Instance

∞ / 1 spawns left