

## Curling In The Cloud with localhost

Nous avons face à nous la page contenant le flag. Cependant, un message nous indique que l'accès est refusé

**Access Denied, Your IP: 108.161.172.123**

Allons à l'index

```
<html
:UTF-8>
<head>

<!DOCTYPE
s="no-js">

content="width=device-width,

lang="fr-CA"

"viewport">
<meta
```

Enter a URL to fetch

URL:

Nous avons encore le même principe que le défi précédent, mais le flag se trouve : /flag.php

Essayons d'y accéder par les adresses de loopback. Le serveur va tenter d'accéder à ses propres ressources. Donc, nous ne devrions pas se faire bloquer par flag.php ??

**That IP is a blacklisted cidr (127.0.0.1/24)!**

<http://127.0.0.1/flag.php>

**That IP is a blacklisted cidr (169.254.0.0/16)!**

<http://169.254.169.254/flag.php>

**That IP is a blacklisted cidr (0.0.0.0/8)!**

<http://0.0.0.0/flag.php>

Inspectons le code source !

```
</div>
<!--parse_url-->
<!--debug=false-->
</body>
'html>
```

Hum essayons de mettre le paramètre debug=true dans notre requête POST.

☒ Enable Post data ☐ Enable Referrer

Post data

r=http%3A%2F%2Fcsr.dept-info.crosemont.quebec%2F&debug=true

I

array(3) { ["scheme"]=> string(4) "http" ["host"]=> string(31) "csr.dept-info.crosemont.quebec" ["path"]=> string(1) "/" }

<!--DOCTYPE

<!--html

html&gt;

<!--no-js&quot;&gt;

<!--lang=&quot;fr-CA&quot;

<!--head&

<!--meta

<!--t=&quot;UTF-8&quot;&gt;

<!--name=&quot;viewport&quot;

<!--content=&quot;width=device-width

Enter a URL to fetch

URL: http://csr.dept-info.cros

Envoyer

Voilà que le schéma de parse\_url nous apparaît ! Encore, faisons une petite recherche sur parse\_url et ses vulnérabilités reliées.

## Open redirection vulnerability in the Drupal API function drupal\_goto (Drupal 6.15 and 5.21)

From: Martin Barbella <martybarbella () gmail com>

Date: Thu, 4 Mar 2010 10:06:07 -0500

Open redirection vulnerability in the Drupal API function drupal\_goto (Drupal 6.15 and 5.21)

Discovered by Martin Barbella <martybarbella () gmail com>

Description of Vulnerability:

Drupal is a free software package that allows an individual or a community of users to easily publish, manage and organize a wide variety of content on a website (<http://drupal.org/about>).

The drupal\_goto API function is meant to "send the user to a different Drupal page. This issues an on-site HTTP redirect. The function makes sure the redirected URL is formatted correctly" ([http://api.drupal.org/api/function/drupal\\_goto](http://api.drupal.org/api/function/drupal_goto)).

This function will also check \$REQUEST['destination'] and \$REQUEST['edit']['destination'], and if either of these variables are set, will override any specified path with the path element of the associative array returned when passing either request variable through parse\_url.

When a URL such as "trickparseurl:<http://cwe.mitre.org/data/definitions/601.html>"; is passed to PHP's parse\_url function, it will return:

```
array(2) (
  ["scheme"]=>
  string(13) "trickparseurl"
  ["path"]=>
  string(46) "http://cwe.mitre.org/data/definitions/601.html";
)
```

Nous trouvons qu'il est possible de manipuler la syntaxe du schéma construit par parse\_url .  
Regardons la capture d'écran du Writeup précédent.

### PHP curl\_exec() url is controlled by user

**Web Vulnerabilities** / 700000 **Medium Severity** / 700000 PHP curl\_exec() url is controlled by user

**Description**

Manual confirmation is required for this alert.

This script is using the PHP function `curl_exec()`. The url used by curl is based on user input. This is not recommended as it can lead to various vulnerabilities.

For example, an attacker can use the `file://` protocol to read arbitrary files from the server (by using an url like `file:///etc/passwd`). It's also possible to access computers behind the firewall using URLs like `http://192.168.0.1` or `ftp://192.168.0.1`.

An older version of libcurl compiled to support SCP can get tricked to get a file using embedded semicolons, which can lead to execution of commands on the given server. "scp://name:passwd@host/a" ;date >/tmp/test"" ;"

**Severity**

LEVEL 1 - MEDIUM

**Classification**

**CWE** **CVE-2009-0037**

**CWE** **CWE-352**

Essayons de combiner les deux.

&lt;!DOCTYPE  
 PUBLIC  
 HTML  
 &quot;-//W3C//DTD  
 2.0//EN&quot;  
 Found&lt;title&gt;  
 Not  
 &lt;title&gt;404  
 &lt;/head&gt;&lt;body&gt;

Enter a URL to fetch

URL:  Envoyer

r=http://salut:salut@csir.dept-info.crosemont.quebec/flag.php&debug=true

Pouvons-nous ajouter 127.0.0.1 :80 devant @csir.dept-info.crosemont.quebec/

```
array(5) (["scheme"]=> string(4) "http" ["host"]=> string(31) "csir.dept-info.crosemont.quebec" ["user"]=> string(5) "sahut" ["pass"]=> string(18) "sahut@127.0.0.1:80/" ["path"]=> string(9) "/flag.php")
```

FLAG-33C3WASFUN

Enter a URL to fetch

URL: <http://csir.dept-info.cros> 

Magie !

Curl interprète notre url de cette façon !

Lors de la validation, l'host sera `csir.dept-info.crosemont.quebec`.

Cependant, Curl interprétera que l'host est 127.0.0.1 :80 et donc nous il nous permettra d'accéder à flag.php puisque la requête sera faite par lui-même

<http://seclists.org/fulldisclosure/2010/Mar/103>

<https://archive.aachen.ccc.de/33c3ctf.ccc.ac/index.html>