

Terminal--CURLing Master (part 2)

Solution

Normally HTTP2 connections may start with HTTP 1.1 and negotiate a change to HTTP2. However, Bushy's application only accepts HTTP2, which makes the curl command a little more difficult. If you look at the terminal's BASH history, you get a helpful clue.

```
elf@de00fc9cb69d:~$ history
 1 netstat -ant
 2 ncat --broker -nlvp 9090
 3 echo "\302\257\_(\343\203\204)\_/\302\257" >> /tmp/shruggins
 4 cat /tmp/shruggins
 5 curl --http2-prior-knowledge http://localhost:8080/index.php
 6 telnet towel.blinkenlights.nl
 7 fortune | cowsay | lolcat
 8 ps -aux
 9 sl
10 figlet I am your father
11 echo 'goHangasaLAmIimalaSagnaHoG' | rev
12 aptitude moo
13 aptitude -v moo
14 aptitude -vv moo
15 aptitude -vvv moo
16 aptitude -vvvv moo
17 aptitude -vvvvv moo
18 aptitude -vvvvvv moo
19 yes Giddyup
20 factor 512
21 aafire
22 history
elf@de00fc9cb69d:~$
```

On line 5, someone used the command

```
curl --http2-prior-knowledge http://localhost:8080/index.php
```

If you use curl --http2, the command will fail on this server since curl will start with HTTP 1.1 and attempt to negotiate a transition to HTTP2.

```
elf@17945497ca8c:~$ curl --http2 //localhost:8080/
curl: (3) <url> malformed
elf@17945497ca8c:~$
```

The flag --http2-prior-knowledge tells curl to skip the negotiation because we already know that the server is running HTTP2.

```
elf@de00fc9cb69d:~$ curl --http2-prior-knowledge http://localhost:8080/index.php
<html>
  <head>
    <title>Candy Striper Turner-On'er</title>
  </head>
  <body>
    <p>To turn the machine on, simply POST to this URL with parameter "status=on"

  </body>
</html>
elf@de00fc9cb69d:~$
```

This tells us we need to POST "status=on" to the server.

The [link with curl POST examples](#) shows us that the proper format to issue a POST request is to use -d followed by the data we want to POST, and then -X POST to tell curl we want a POST request.

```
curl -d "status=on" -X POST --http2-prior-knowledge
```

```
http://localhost:8080/index.php
```

```
elf@de00fc9cb69d:~$ curl -d "status=on" -X POST --http2-prior-knowledge http://localhost:8080/index.php
```

That does the trick!

```

                                okkd,
                                OXXXXX,
                                oXXXXXXo
                                ;XXXXXX;
                                ;FXXXXXXx
                                oXXXXXXO
                                .lKXXXXXXX0.
'MMMMO',,MMMMMMO',,MMMMMMK',,eccccOXXXXXXXXXXXXXkXXXXXXXXXXXX.
'MMMMN',,OMMMMMW',,OMMMMMW',,kxccccOXXXXXXXXXXXXXkX0KKKKK000d;
'MMMMl',,oMMMMMMo',,lMMMMMMd',,cMxccccOXXXXXXXXXXXXXOdK0000KKKK0x.
'MMMO',,MMMMMMO',,NMMMMMK',,XMxccccOXXXXXXXXXXXXXkXXXXXXXXXXXXX:
'MMN',,CMMMMMW',,kMMMMMW',,xMMxccccOXXXXXXXXXXXXXkKkx00000000x;.
'MMl',,lMMMMMMo',,cMMMMMMd',,MMxccccOXXXXXXXXXXXXXK00kd0XXXXXXXXXX0.
'MO',,MMMMMMO',,NMMMMMK',,XMMxccccKXXXXXXXXXX0FKKx0KKKXXXXXXXXk.
.c.....'ccccc.....'ccccc.....'ccc:ccc: .c0XXXXXXXXXX0x000000000c
                                ;XXXXXXXXX0XXXXXXXXXK.
                                ..,:ccllc:ccccc:'

Unencrypted 2.0? He's such a silly guy.
That's the kind of stunt that makes my OWASP friends all cry.
Truth be told: most major sites are speaking 2.0;
TLS connections are in place when they do so.

-Holly Evergreen
<p>Congratulations! You've won and have successfully completed this challenge.
<p>POSTing data in HTTP/2.0.

</body>
</html>
elf@de00fc9cb69d:~$
```

Up Next

After talking to Holly to collect her hints, move on to