Edimax BR-6478AC mid-2015 Multiple Vulnerability Write-Up

Reported by Michael Winstead

July 2015

Table of Contents

Disclaimer

The views expressed in this report do not reflect the official policy or position of my employers past or present. This report is designed to be informative in nature and is not designed to bring discredit to the affected company or individuals.

Executive Summary

There exists in the Edimax BR-6478AC (firmware version 2.15) small office, home office (SOHO) WiFi router [1] a number of security flaws which allow an authenticated user to perform additional actions beyond what is permitted from the standard web interface at the highest privilege level. These security flaws may be exploited by a malicious actor in order to redirect critical personal internet traffic from its intended recipient to a location operated by said actor for nefarious purposes. Unfortunately, these flaws seem to have originated from a number of poor software development practices which are specifically addressed as the number one issue in the Open Web Application Security Project (OWASP) top web application security awareness document [2]. By allowing these flaws to go unpatched, it places the customers of Edimax at a greater level of risk for safe and private internet use.

Affected Products

Preliminary research demonstrates that the BR-6478AC may not be the only vulnerable router.

Disassembly of firmware and preliminary search for format string denoting presence of the /goform/mp command shell in the web server at /bin/webs:

- BR-6208ACD (v1.21)
- BR-6288ACL (v1.05)

Analysis of the GPL source code of demonstrates the same vulnerable functions as present within the following additional products::

- BR-6428nC (v1.07)
- BR-6228nC (v1.11)
- BR-6258n (v1.18)
- BR-6528ACL (unknown version, found source on GitHub)

Technical Vulnerability Descriptions

Vulnerability Discovery Method

All vulnerabilities found herein were discovered through a mix of static and dynamic analysis of a running system and open source code release. Dynamic analysis is assisted by directly soldering to the Universal Asynchronous Receiver/Transmitter (UART) header on the main board of the WiFi router in order to directly observe command output. Since the UART connector is only 3.3 volts, direct current, I used a raspberry pi and bus pirate in order to connect a PC and observe the output. Furthermore, the python programming language [3],

command-line cURL [4], GNU Is Not UNIX (GNU) screen [5], and vi improved (VIM) [6] were critical in order to replicate requests and automate processes.

Static analysis was completed by simply downloading the GNU Public License (GPL) source code release from Edimax's website and reading what was provided. Of note is the fact that the GPL release was not sanitized in any way and seemed to be a "developer's dump" of an entire development workspace including partially compiled objects, build scripts, and developer comments.

Web Server Vulnerabilities

Buffer Overflows and Command-injectable request parameters. Specified variables are written to stack buffers and also directly string-formatted and executed using the system() system call. These form handlers are additions to the boa web server and may be found in C source files under the RTL8197/AP/boa-0.94.14rc21/src directory within the GPL source release for this router. Each of the forms are accessed via the URL "http:///<IP address>/goform/<form name>" with a valid authentication string. After the form name is a designation of how large the stack buffer is in which the request variables are formatted. This should provide a starting point for proving that a buffer overflow exists by making a request which is a deal larger than the stack buffer.

- formHwSet (600-byte buffer)
    - Antenna
    - Mcs
    - regDomain
    - regDomain2
    - nic0Addr
    - nic1Addr
    - wlanAddr
    - wlanAddr2
    - wlanSSID
    - wlanChan
    - comd
        - This variable is simply directly ingested and ran by calling system()
    - initgain
    - txcck
    - txofdm

- formUSBStorage (200-byte buffer)
    - sub_dir

- mp (500-byte buffer)
    - command

- formSystempingtool (150-byte buffer)
    - ping_ip
    - ping_number

- formWpsStart (100-byte buffer)
    - pinCode (overflow AND command injection)
    - modeVal (overflow only with long number)
- formWlanMP (300-byte buffer)
    - NOTE: These can also just have each of these as empty strings with only one having the actual command injection.
    - ateFUNC
    - ateGain
    - ateTxCount
    - ateChan
    - ateRate
    - e2pTxPower1
    - e2pTxPower2
    - e2pTxPower3
    - e2pTxPower4
    - e2pTxPower5
    - e2pTxPower6
    - e2pTxPower7
    - e2pTx2Power1
    - e2pTx2Power2
    - e2pTx2Power3
    - e2pTx2Power4
    - e2pTx2Power5
    - e2pTx2Power6
    - e2pTx2Power7
    - strAteFreqOffset
    - ateMode
    - ateBW
    - ateAntenna
    - e2pTxFreqOffset
    - e2pTxPwDeltaB
    - e2pTxPwDeltaG
    - e2pTxPwDeltaMix
    - readE2P
    - e2pTxPwDeltaN
    - ateMacID
    - AbandTx1
    - AbandTx2
- formWlBasic

    - key1
        - a specific buffer (ConnectTestKey, 200 bytes) has this strcpy'd into it, but since it is a global variable rather than a stack-local one, a traditional buffer-overflow may not be possible depending on the current memory layout.
    - repeaterSSID
        - not available for buffer overflow, command injection only with subshell

"Evilgrade" Attacks

These attacks are where an attacker could trick the device into accepting an "upgrade" which is actually either purely malicious or is the original software with malicious code added.

fmmgt.c

formUploadWebsite

Lack of secure download capability (cryptographic signatures or HTTPS+certificate checking) could allow an attacker to take the place of "www.planex.co.jp" and deliver a malicious software update

Injection Vulnerabilities

fmmgt.c

formUploadWebsite

In the response body from an upload, were an attacker to take the place of the legitimate upgrade site, they could issue system commands due to an injection in the url value which is then fed to the system command

Command Shells

While great for development, testing command shells should be removed for production. Attackers can use these to execute arbitrary commands on the target device.

fmmgt.c

mp

- URL: /goform/mp
- Methods: GET, POST
- This seems to be a shell to test wireless radio behavior, but can be exploited by inputting double-bars (||) and then any valid shell command. This method causes the originally intended script to fail and for the shell to proceed to the injected commands

Script Vulnerabilities

The below vulnerabilities exist within the scripts which are called by the custom form handlers within the boa web server.

/bin/urlblocking.sh

Within the iptable command, a url which is a command-substitution will execute upon boot on the device.

/bin/wps.sh

Within the iwpriv program invocation, a subshell command-substitution can be put into place from the wps pin.

Example HTTP Request

The below HTTP request, when sent to a vulnerable router, will cause the output from the ps command to appear on the device's UART serial port on standard error.

```
$ curl -u admin:1234 --data
'wlan-url=%2Fwireless_wps.asp&confMode=0&configOption=pin&pinCode=`ps
%3e%262; killall -9 webs`' http://192.168.2.1/goform/formWpsStart
```

This example exploits the command injection into the formWpsStart. When URL decoded, the relevant part of the exploit looks like this:

```
pinCode=`ps >&2; killall -9 webs`
```

While the first component seems to be relatively normal, the second component with killing the webs process keeps the wps pin process from holding up issuing additional commands to the web server.

In order to exploit the buffer overflow vulnerability within formWpsStart, simply create a pinCode which, when formatted into the temporary buffer, takes up more than the correct space and observe the UART serial port. In order to reliably test this, I would suggest making all the letters within the pinCode to be an ASCII "A". When there is a crash on the system in which it complains that the instruction pointer is 0x41414141 (the hexidecimal representation of "A"), the buffer has been overflowed enough to overwrite the function return pointer on the stack.

Vulnerability Disclosure Communications Log

- 7 June 2015

  Edimax,
  Does your company have a program to handle externally reported bugs? If so, please let me know how to embark upon one such report.

  I would like to open a dialogue regarding one of your WiFi routers (the BR-6478AC).

  Thank you for your time.

  - Mike Winstead

- 2340 17 July 2015

  Edimax,

I e-mailed on July 7th with regards to a possible bug I encountered in one of your SOHO WiFi routers and did not receive a reply.

I am hereby making a second attempt at contacting you to responsibly disclose security vulnerabilities within a number of your home WiFi routers. There is a shared codebase between many of your routers and the BR 6478AC which can lead an authenticated user to execute arbitrary commands and possibly malicious code on the router itself at the root level. By allowing these issues to go unpatched, users may be vulnerable to a large number of man-in-the-middle (MiTM) attacks from compromised home WiFi router.

An example of one such issue is an embedded injection vulnerability in your URL filtering web page which can allow a malicious command to be executed every time the router is rebooted and could be used to redirect a user's entire home internet traffic to malicious actors on the internet.

Please contact me for a full write-up of these flaws. I do not exact a fee for this write-up, I only ask that we can work to keep the internet as a whole a safer place. Additionally, I am prepared to make recommendations to mitigate the issues.

If you do not respond, I will work the issue with the US-CERT in order to release a public advisory to the issue.

Thank you for your time.

- Mike Winstead

- 0020 17 July 2015

  Correction to the below:

  My previous e-mail was on the 7th of June,

  Thank you.

  - Mike Winstead

  ---------- Forwarded message ----------
  From: Michael Winstead <mwinstead3790@gmail.com>
  Date: Thu, Jul 16, 2015 at 11:40 PM
  Subject: BR 6478AC Router Vulnerability Disclosure
  To: support@edimax.com

…

- 1140 17 July 2015

  Hi Michael,

  Sorry we did not reply to your first email. For some reasons, our tech support team did not receive your previous email (7th of June).

  I will discuss with HQ (R&D team) and see if they are aware the security issue you addressed in your first email.
  Will keep you update on this.
  May I ask what is your firmware version of your BR-6478AC?

  And you also mentioned that this issue exist in other Edimax router? Can you please let me know the model number and firmware version?
  Thank you in advance.

  Best regards,
  Jeffrey Cheng | Product Manager | Edimax USA | 408-496-1105 Ext 105

- 1149 17 July 2015

  Mr. Cheng,

  Thank you for getting back with me.

  I can for sure confirm that the v2.15 firmware is vulnerable along with the GPL release for the same router.

  I don't have the specifics with me (travelling with only my phone for the next 10 or so hours). All you need to look for are your routers which have the same Edimax-custom form handlers written for the boa Web server as the BR6478AC. As for versions, they were the latest releases on the website. I can have more specifics when I get back to my hotel room tonight.

- 2033 17 July 2015

  Mr. Cheng,

  By disassembling the firmware updates of the routers and just looking for indications of a left-over developmental command shell, the following routers may share a vulnerable code base with the BR-6478AC:

- BR-6208ACD (v1.21)
- BR-6288ACL (v1.05)

Additionally, the GPL source code of the following seems to share some vulnerable code:

- BR-6528ACL (unknown version, found source on GitHub)
- The following had release firmwares available, but I did not have a big-endian unsquashfs on hand to unpack them:
  - BR-6428nC (v1.07)
  - BR-6228nC (v1.11)
  - BR-6258n (v1.18)

My method for checking these is to analyze the webs web server ELF file and search for a string which indicates it has an enabled developmental shell which allows for arbitrary command execution (It should have likely been removed for production). This method does not test for suseptibility to the rest of the vulnerabilities I found in the BR-6478AC.

Please note that since I do not have these routers in my possession, I can't provide a 100% recreation of the vulnerabilities. All I can do is statically analyze their firmwares and detect whether they should be running vulnerable code.

One last note: possible computer usernames of various employees of Edimax are also still present in the GPL releases including:

- hcjong
- ygtai
- John Huang

-Mike Winstead

- 0045 20 July 2015

  Hi Michael,

  Thanks for all the information. We already discussed with engineer team. They are investigating this issue now.
  I will keep you update if we have any patch released. Thank you so much!

  Best regards,
  Jeffrey

- 2039 20 July 2015

  Mr. Cheng,

  Thanks for getting with engineering. I haven't actually told you the vulnerabilities yet, so feel free to forward my information to the team so we can have the technical discussion.

  Also, with whom could I work to put together a timeline of patch release and public vulnerability disclosure.

  Thanks!

- 1513 23 July 2013

  Dear Michael,

  My team tested the BR-6478AC with Nessus and OpenVAS but they did not find vulnerability yet.
  Can you please let us know how did you find the vulnerability?
  We will follow your step and duplicate this issue and then solve it.

  Thank you in advance. Have a great day.

  Best regards,
  Jeffrey Cheng | Product Manager | Edimax USA

- 0907 23 July 2015

  Mr. Cheng,

  Please find what you've requested in the below Google Doc.

  https://docs.google.com/document/d/1fDnXf0ymgnCDf6pK46c64jyQZa4CqX4BBUGKrH00dVw/edit?usp=sharing

  To generate the HTTP requests in order to inject into the HTTP vulnerabilities, I suggest using cURL on the command line. An example is given, but if you would like me to go back and create specific cURL commands which replicate each vulnerability I can do that.

  Thank you for looking into this, Mr. Cheng. Have a good day as well.

- Mike Winstead

- 1229 24 July 2015

  Hi Mike,

  Thank you so much!
  Your note is received. I will forward it to our team and follow up with R&D team.

  I will keep you post if I have any updates. Thank you!

  Best regards,
  Jeffrey

- 1446 24 July 2015

  Thanks for keeping me posted, Mr. Cheng.

  Additionally, feel free to send back the document and say that you'd like specific steps and expected/unexpected output for these vulnerabilities.

  I'm working on this vulnerability disclosure as a template for future work, so thank you for working with me in figuring out what elements need to be in it.

  Best of hunting to your team.

  - Mike Winstead

- 1551 1 August 2015

  Mr. Cheng,

  I'd like to check in with you so that we can set a date for the public release of these vulnerabilities. Please let me know your timeline so we can coordinate the public side of this disclosure.

  Thanks!

  - Mike Winstead

- 1531 7 August 2015

  Dear Michael,

Sorry for late reply. I was out of office for few days.

I just got the confirmation from our engineers.
They told me that they do use those open source code you mentioned in the document but our routers are not vulnerable.

If someone wants to generate the HTTP requests in order to inject into the HTTP vulnerabilities, they will need to (1) connect to LAN network (which they cannot because they don't have password) and (2) has admin's password.
If they don't have those 2 information (wireless security password and admin's password), they are not able to generate the http requests.
Therefore we don't think they are vulnerabilities.

Please let me know if you have any other questions or suggestions.
Thank you.

Best regards,
Jeffrey Cheng | Product Manager | Edimax USA

- 2191 14 August 2015

  Mr. Cheng,

  While I disagree with those findings, I respect that at least you were willing to look at the vulnerabilities which I reported and have been openly working with me during this disclosure.

  I will release this information to the Zero Day Initiative in order to complete routing and handling of these vulnerabilities and an appropriate public release. My plan for release to them will be by Wednesday August 19th.

  It has been good working with you Mr. Cheng and thank you for your and your team's time.

  - Mike Winstead

- 1853 17 August 2015

  Hi Mike,

  Thanks for letting me know.
  I will talk with RD tonight again about this issue.

Best regards,
Jeffrey

- 1837 20 August 2015

  Hi Mike,
  First, thanks for sharing the Vulnerability with Edimax again. I have discussed this issue with senior engineers this week again. Edimax will fix those vulnerabilities which are listed in your Write-Up (Google document):

  (1) Web Server vulnerabilities
  (2) “Evilgrade” Attacks
  (3) Injection Vulnerabilities
  We will take below actions to fix the security issues in about 6-8 weeks.

  a. When updating firmware, we will check cryptographic signatures or using HTTPS+certificate checking
  b. We will remove testing command shells (/goform/mp) for production.
  c. We will fix the issue that attackers could issue system commands due to an injection in the url value.

  After we fix all issues, we will release the patch (new firmware) as soon as possible. Please let us know if you have any question of findings. Thanks for your help on these vulnerability issues.

  Best regards,
  Jeffrey Cheng | Product Manager | Edimax USA | 408-496-1105 Ext 105

- 1919 20 August 2015
  Mr. Cheng,

  I'm very glad to hear that you're taking these measures! Please be assured - this is the right thing to do. Fixes a and b in my experience are the correct way of protecting those systems.

  May I inquire as to a sub-section of fix c? In the proposed fix, you say you're going to fix system command injection while that was half of the issue. The other half of the issue is that the HTTP POST values are written to stack buffers which may be exploited by making a request with a parameter of appropriately long length and value (a classic buffer overflow). If your engineers are already in that code base making changes, I'd suggest also taking care to modify the handling of the values so that all cover all of your bases.

6-8 weeks puts the date of fix release between 1 and 15 October. I'll check the firmware update page around that time to ensure that the fix has been released before releasing these vulnerabilities publicly.

I was more than glad to help with the vulnerability issues. Thank you for working with me to gain responsible reporting experience.

Respectfully,
- Mike Winstead

- 1308 25 August 2015

  Hi Mike,

  Thanks for your remind. For sub-section of fix c, our engineers will check the length of parameter to solve the buffer overflow issue.
  Will keep you update if we release the firmware for this security issue.

  Thank you,

  Best regards,
  Jeffrey Cheng | Product Manager | Edimax USA | 408-496-1105 Ext 105

- 2008 14 October 2015

  Dear Michael,

  Hope everything is well with you.

  I just want to let you know that we already released the new firmware to fix CGI Vulnerability for BR-6478AC.
  The new firmware version is v2.20. you can download it from the link below:

  http://us.edimax.com/edimax/download/download/data/edimax/us/download/for_home/wi-fi_range_extenders/wi-fi_range_extenders_ac1200/br-6478ac

  In addition, we also updated firmware for these routers below:

  1. BR-6478AC v2.20
  2. BR-6208AC v1.28
  3. BR-6288ACL v1.10
  4. BR-6228nS_v2 v1.22

5. BR-6228nC_v2 v1.22
6. BR-6428nS_v2 v1.16
7. BR-6428nC v1.16

If you have any suggestion or feedback, please let us know. Thanks you!

Best regards,
Jeffrey

- 2152 14 October 2015

  Mr. Cheng,

  This is excellent news! Thank you for working with me throughout this process.

  Feel free to give me feedback as well in case there's anything which you'd like to bring up.

  Respectfully,
  - Mike Winstead

References

1. Manufacturer's Product Page
   http://www.edimax.com/edimax/merchandise/merchandise_detail/data/edimax/au/wireless_routers_ac1200/br-6478ac/

2. OWASP Top 10 2013
   http://owasptop10.googlecode.com/files/OWASP%20Top%2010%20-%202013.pdf

3. Python Programming Language
   https://python.org

4. cURL Project
   http://curl.haxx.se/

5. GNU Screen
   http://www.gnu.org/software/screen/

6. Vi Improved
   http://www.vim.org/