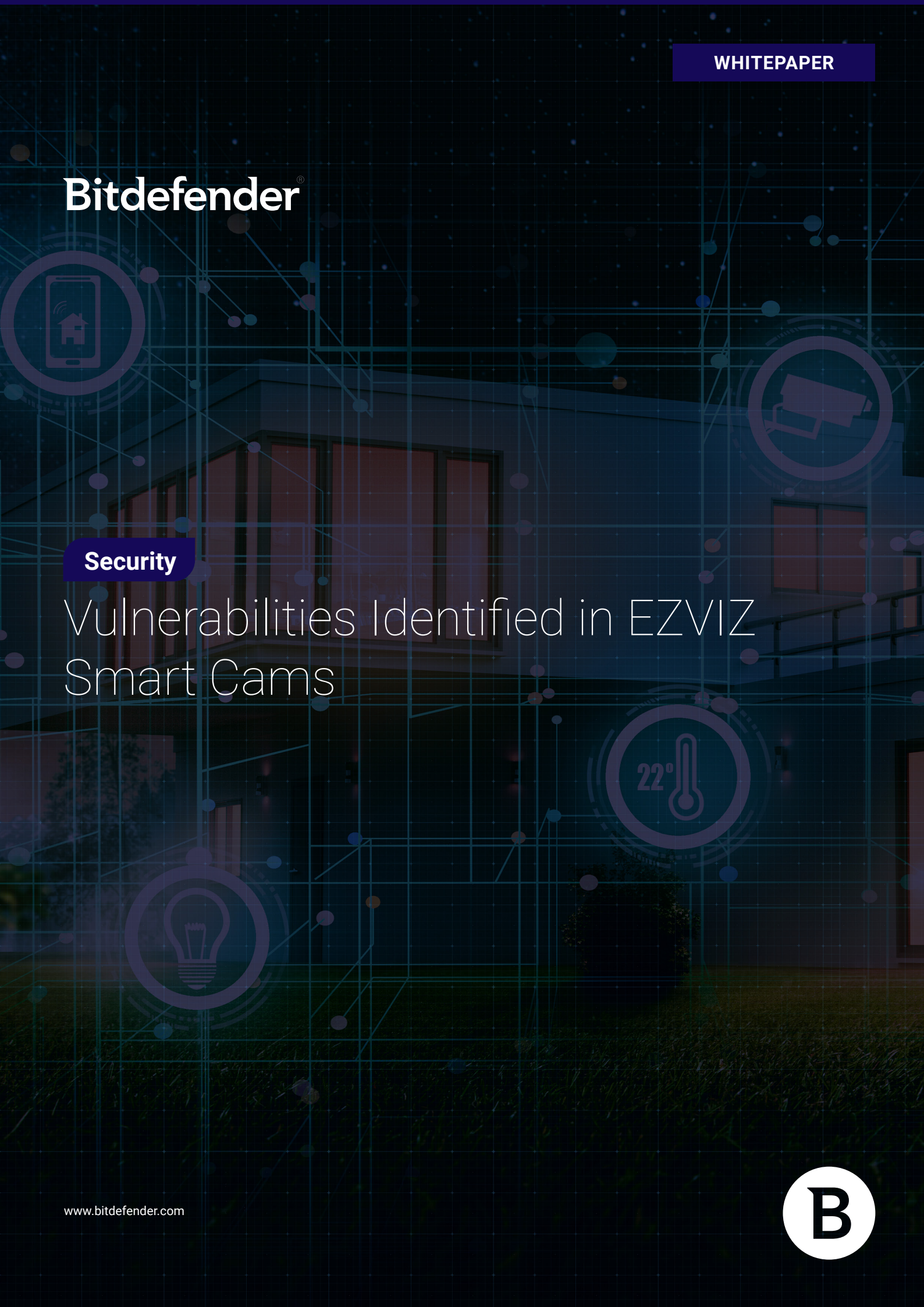
WHITEPAPER

Bitdefender



Security

Vulnerabilities Identified in EZVIZ Smart Cams

www.bitdefender.com

Contents

Foreword

As the creator of the world's first smart home cybersecurity hub, Bitdefender regularly audits popular IoT hardware for vulnerabilities that might affect customers if left unaddressed. This research paper is part of a broader program that aims to shed light on the security of the world's best-sellers in the IoT space. This report covers several camera models manufactured by EZVIZ. Vulnerable camera models and firmware versions that we based our research on are listed in Appendix A.

Vulnerabilities at a glance

- [REMOTE] Stack-Based Buffer Overflow Vulnerability can lead to remote code execution in the motion detection routine – CVE-2022-2471| CWE-121 **[1]**
- [REMOTE] Insecure Direct Object Reference vulnerability in multiple API endpoints allows an attacker to fetch images and issue commands on behalf of the real owner of the camera [2]
- [REMOTE] Storing Passwords in a Recoverable Format vulnerability in [3] **/api/device/query/encryptkey** allows an attacker to recover the encryption key for images
- [LOCAL] Improper Initialization vulnerability lets an attacker recover the administrator password and completely own the device - CVE-2022-2472 |CWE-123 **[4]**

Impact

When daisy-chained, the discovered vulnerabilities allow an attacker to remotely control the camera, download images and decrypt them. Use of these vulnerabilities can bypass authentication and potentially execute code remotely, further compromising the integrity of the affected cameras.

Bitdefender has been working closely with EZVIZ through all stages of vulnerability disclosure. We would like to extend our thanks for the prompt response time, communication, transparency and escalation.

Disclosure timeline

Apr 15, 2022: Bitdefender makes initial contact attempt via multiple public communication channels

Apr 16, 2022: Acknowledgement received, vendor requests additional information through OneDrive

Apr 18, 2022: Bitdefender submits documentation and proof of concept

Apr 20, 2022: Report received and acknowledged by vendor

May 05, 2022: Vendor informs that internal assessment is in progress

May 10, 2022: Vendor requests a 90-day extension for vulnerability fixing and patching

May 16, 2022: Vendor communicates the findings of internal assessment and confirms fix

Jun 20, 2022: Updates are still rolling out to vulnerable devices

Sep 15, 2022: This report becomes public as per the coordinated vulnerable disclosure guidelines

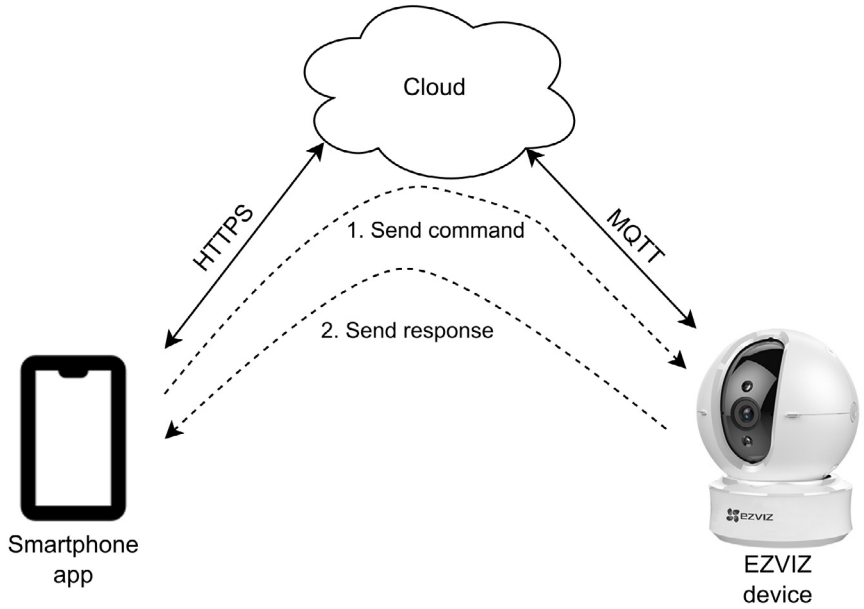
Vulnerability walkthrough

1. Remote code execution vulnerability in the /api/device/configMotionDetectArea API endpoint (CVE-2022-2471)

Our analysis uncovered several vulnerabilities in the EZVIZ smart devices and their API endpoints that could allow an attacker to carry out a variety of malicious actions, including remote code execution and access to video feed.

One of the main features of these devices is the ability to be accessed from anywhere the user has an Internet connection. To accomplish this, user-device communication is relayed through servers in the cloud. When the smartphone app needs to contact a device, the cloud servers relay the messages back and forth.

Two communication channels are implemented: one for streaming audio-video, and one for control and configuration changes. The latter is used to send commands to the camera, such as pan-tilt-zoom movement, enabling or disabling sound recording, and performing basic configuration. To achieve this, the device always stays connected to the cloud through a MQTT tunnel. When the smartphone app connects to the cloud and issues a command, the server forwards it to the device through the active connection. The device processes and executes the command and sends the result through the same channels, in reverse order.



1. The app sends a command; 2. The device receives the command, processes it, then sends the result back

One of the multitude of available commands allows the user to set up an area in which the device will check for motion. This area is set through the **/api/device/configMotionDetectArea** API endpoint. The command is then forwarded by the cloud server to the device through the MQTT connection. After the command is received, the device tries to decode the base64-encoded payload into a local stack buffer, without checking if the result will fit into the allocated size. This corrupts the stack and may lead to remote code execution.

2. Insecure Direct Object Vulnerabilities in several API endpoints

We also found that multiple API endpoints are vulnerable to insecure direct object reference vulnerabilities. These types of vulnerabilities allow an attacker to access resources that belong to other users by simply using the IDs of those resources instead. This is possible due to a failure to check if the person making the request has permission to access the resource. To find IDs of resources belonging to other users, an attacker can simply increment a known ID, as they are sequential. In the case of those devices, their serial numbers are used as unique IDs.
The motion detection area API endpoint located at **/api/device/configMotionDetectArea** does not check if the client making the request has access to the device with the specified serial number. This allows any authenticated client to send arbitrary payloads to any compatible device. This vulnerability can be combined with the buffer overflow exploit presented above to deliver malicious payloads to vulnerable devices and accomplish remote code execution.
Another vulnerable API endpoint is at **/api/panoramic/devices/pics/collect**. This endpoint allows client to request a panorama from a camera. However, it does not check if the client making the request has access to the device with the specified serial number. This allows any authenticated client to request a panoramic view from any compatible device.

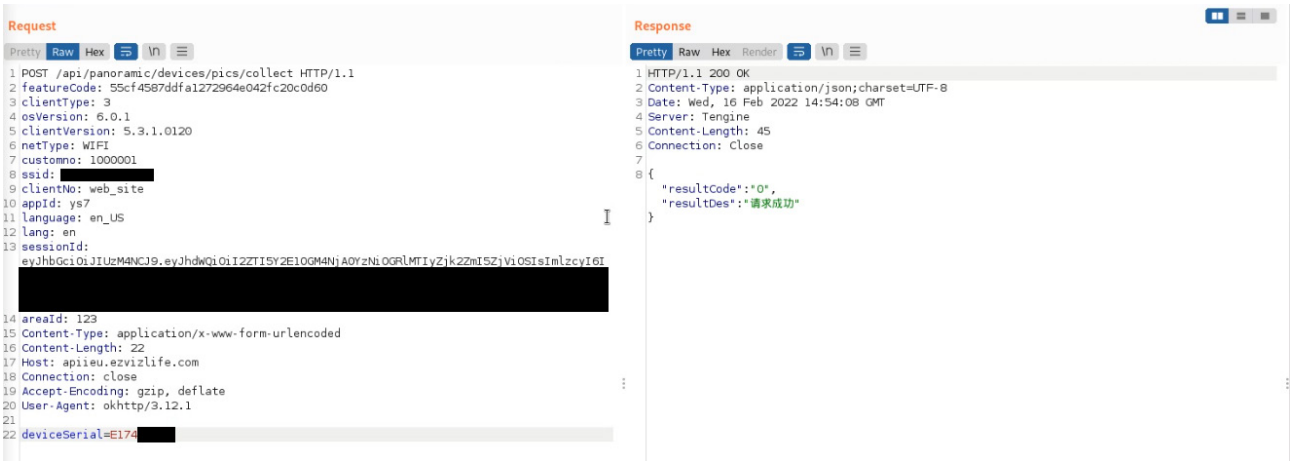


Figure 1: User requests a panorama from the camera

After requesting a panorama, the device takes several pictures, uploads them to the cloud, and reports on the progress. To access the uploaded images, we have to make another request to the **/api/panoramic/devices/pics** API endpoint, which will return the address from where we can download the images.

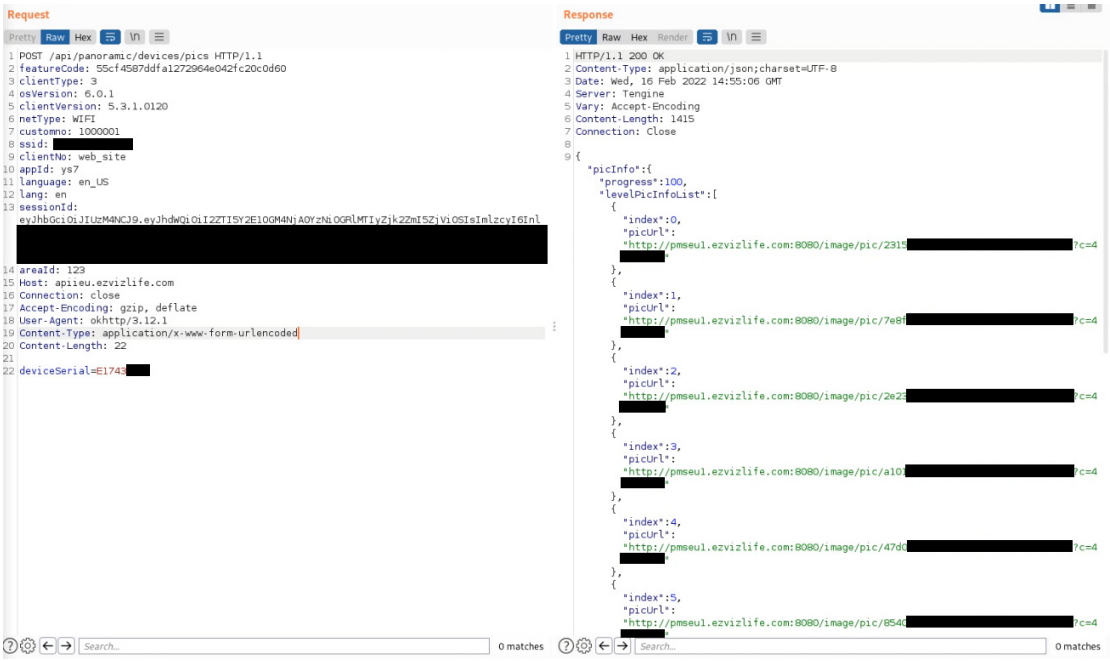


Figure 2: After the camera uploads the pictures, the server responds with their location

3. Storing Passwords in a Recoverable Format vulnerability in /api/device/query/encryptkey

After downloading the images, we learn that they are encrypted. Each device has a random verification code. This code is six characters long, all uppercase, and is used to encrypt the images. Due to its small entropy, it is trivial to bruteforce the code and decrypt the pictures. However, another security feature can be present: after setting up the device, the user is prompted to set a custom password for it. This password, which has to be at least eight characters long, will be used to encrypt the images instead of the verification code.
We have found that an API endpoint located at **/api/device/query/encryptkey** allows you to retrieve the secret password by only specifying the serial number of the device. This endpoint will not check if the device belongs to you and will return the password in plaintext. We will then use this password to decrypt and access the images.

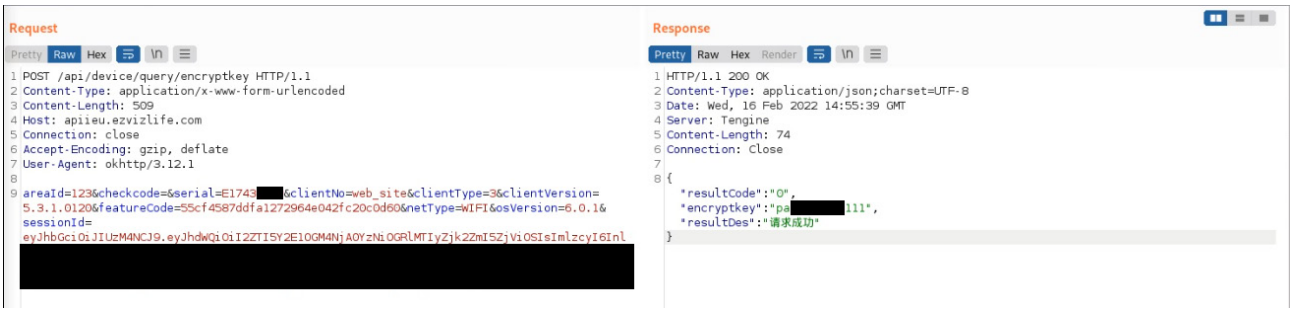


Figure 3: The vulnerable API returns the password to the device in plaintext

4. Administrator Password Recovery Vulnerability (CVE-2022-2472)

Another issue is present on a service running on the camera and listening on local port 8000. This service is used to control and configure the camera while connected on the same local network. When a client authenticates on this port, the server initializes some memory that holds details about the connected client, such as user ID, MAC address, IP address. However, if no client has authenticated since the service started, that memory is not initialized and will contain predictable data. In that case, the client can send a forged request which contains data that will be in that memory and the server will validate it. This bypass works only on a subset of server functionalities, such as NETCMD_GET_USERCFG_V30. After performing this request, the device will return the admin password, XORed with value 0x738b5544. Once the admin password is obtained, the attacker has full access to the device.

Appendix: Vulnerable camera models

The vulnerabilities were found on firmware version V5.3.0 build 201719 (previous versions might also be vulnerable but untested). Affected device models are listed in the table below – please note that there may be other device models and integrations that we have not tested:

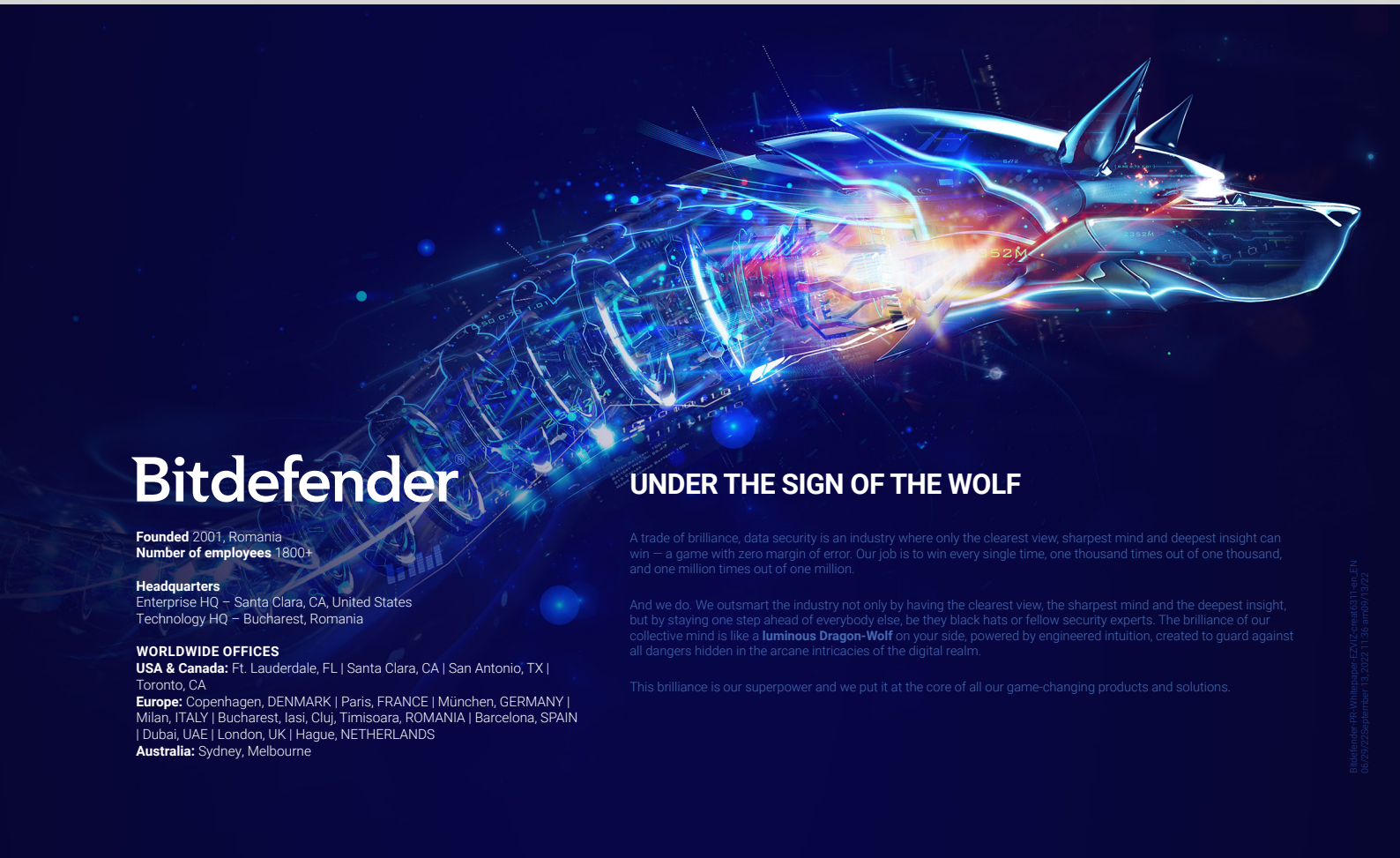
CS-CV248 [20XXXXX72] - V5.2.1 build 180403
CS-C6N-A0-1C2WFR [E1XXXXX79] - V5.3.0 build 201719
CS-DB1C-A0-1E2W2FR [F1XXXXX52] - V5.3.0 build 211208
CS-C6N-B0-1G2WF [G0XXXXX66] - v5.3.0 build 210731
CS-C3W-A0-3H4WFRL [F4XXXXX93] - V5.3.5 build 220120

About Bitdefender

Bitdefender is a cybersecurity leader delivering best-in-class threat prevention, detection, and response solutions worldwide. Guardian over millions of consumer, business, and government environments, Bitdefender is one of the industry's most trusted experts for eliminating threats, protecting privacy and data, and enabling cyber resilience. With deep investments in research and development, Bitdefender Labs discovers over 400 new threats each minute and validates around 40 billion daily threat queries. The company has pioneered breakthrough innovations in antimalware, IoT security, behavioral analytics, and artificial intelligence, and its technology is licensed by more than 150 of the world's most recognized technology brands. Launched in 2001, Bitdefender has customers in 170+ countries with offices around the world.

For more information, visit https://www.bitdefender.com.

All Rights Reserved. © 2022 Bitdefender.
All trademarks, trade names, and products referenced herein are the property of their respective owners.



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.

Bitdefender-PR-Whitepaper-EZVIZ-creat5311-en-EN
06/29/22September 13, 2022 11:36 am09/13/22