



THE UNIVERSITY *of* EDINBURGH

Edinburgh Research Explorer

An IoT analysis framework: An investigation of IoT smart cameras' vulnerabilities

Citation for published version:

Alharbi, R & Aspinall, D 2018, An IoT analysis framework: An investigation of IoT smart cameras' vulnerabilities. in Living in the Internet of Things: Cybersecurity of the IoT - 2018. London, UK, Living in the Internet of Things: Cybersecurity of the IoT - 2018, London, United Kingdom, 28/03/18.
<https://doi.org/10.1049/cp.2018.0047>

Digital Object Identifier (DOI):

[10.1049/cp.2018.0047](https://doi.org/10.1049/cp.2018.0047)

Link:

[Link to publication record in Edinburgh Research Explorer](#)

Document Version:

Peer reviewed version

Published In:

Living in the Internet of Things: Cybersecurity of the IoT - 2018

General rights

Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.



An IoT Analysis Framework: An Investigation of IoT Smart Cameras' Vulnerabilities

*Rana Alharbi**, *David Aspinall*[†]

alharbi.rana@ed-alumni.net, david.aspinall@ed.ac.uk
University of Edinburgh, UK

Keywords: Internet-of-things, security, privacy, smart cameras.

Abstract

The significant increase in the number of applications that depend on Internet of Things concept is becoming more evident. It has been deployed in many areas in smart homes, smart cities and health monitoring applications. The means to secure these applications are slower than our growing dependence on them. The aim of this paper is to demonstrate the kinds of vulnerabilities that exist in home monitoring smart cameras and to demonstrate their effects on users' security and privacy, by proposing a threat model and a security and privacy analysis framework. The framework covers five major components of the smart camera system with a set of designed test cases. The framework is applied to five commodity smart cameras. Range of vulnerabilities are discovered with respect to the framework. The vulnerabilities discovered indicate that IoT devices continue to be shipped by vendors without putting enough effort on their security and with insufficient regard for the implications that they have on users' privacy. The work reported here has been part of the first author's MSc thesis [1].

Introduction

The Internet of things (IoT) phenomenon has received much attention in recent years due to its attractive nature of bringing convenience to people's lives and the promise of making various appliances smart. The significant increase in the number of applications that depend on this concept is becoming more evident. The IoT has inspired many possible applications and has been deployed in areas such as smart homes, smart cities, health monitoring applications, etc. [2]. The means to secure these applications are appearing more slowly than our growing dependence on them; more thoughts are put into the usability of such devices and less on their security.

This motivates us to propose an analysis framework for commodity devices such as IoT smart cameras. Smart cameras for home monitoring have been widely used and have been deployed by different manufacturers. Users utilise these smart cameras while away from home to monitor their home.

These cameras that are used for personal use of either indoor or outdoor security often endanger the lives or property of the people using them. Security and privacy are often overlooked, or come as afterthoughts. Alternatively, they may be applied as an extra layer after development. To ensure that IoT products do not compromise users' security and privacy, the UK government released a draft code of practise that guides manufacturers to ensure that security in these products is built in by design [3].

Threat Model and Smart-Camera Analysis Framework

First, we will define the context of the smart-camera system. The camera monitors its surroundings and sends notifications to the user's associated smartphone application whenever an event is detected. The user can also login into his or her account through the application and watch video streams of the camera. The camera sends the video to the user's application either directly or through a server. We examined associated Android apps as the most common case. Additional details are described below.

Some of the assets (AS) that need to be protected in smart-camera systems are the following:

- AS1. The video stream data. Videos can capture users' faces, behaviours, gestures, and many other types of identifiable information. These streams are transmitted from the camera to the Android application either directly or through the server.
- AS2. Personally Identifiable Information (PII) like telephone numbers, a user's full name, home address and so on.
- AS3. The smart camera which is the main physical asset of the system

A complete threat and risk analysis is beyond the scope of this paper. However, four types of threats are considered to create a smart-camera analysis framework that can be further extended.

These types of threats (TH) are as follows:

- TH1. Unauthorised video stream retrieval: The attacker can retrieve a video stream by capturing the traffic exchanged between the smart camera and other destinations.

- TH2. Tampering with the smart camera: The attacker can gather information about the device, reset it, access the device system's logs, or remove external storage.
- TH3. Unauthorised account hijacking: The attacker can brute-force access the user's account passwords.
- TH4. Unauthorised capture of PII data: The reckless design of the Android application can endanger personal data.

The first and fourth threats represent a breach of user confidentiality and privacy, while the second and third threats compromise the availability of the smart camera as well as user confidentiality.

Figure 1 shows the components of the smart-camera system: the smart camera and its associated gadgets, a smart phone with a smart-camera associated application installed on it, the web interface for the smart camera, the servers that the smart camera interacts with, and the possible communications within the system. The dotted square refers to the possibility of the camera and the phone being in the same network. Moreover, the figure illustrates the attack surface and the types of attacks on the components. These attackers (A) pose the kinds of threats described above. We take ideas for these attacks from previous security analysis of devices and applications [4, 5].

- A1. Eavesdropper: When an attacker can monitor network traffic and see unencrypted traffic.

- A2. Active attacker on the network: When an attacker can use brute force to access user accounts.
- A3. Man in the middle (MITM): When an attacker targets improper Secure Sockets Layer (SSL) implementation to route all of the communication into his or her station to see the encrypted traffic in clear text.
- A4. Man at the end (MATE) 1: When an attacker has physical and authorised access to the network on which the target under attack resides.
- A5. Man at the end (MATE) 2: When an attacker inspects or tampers with the physical device hardware or software.
- A6. Remote man at the end (RMATE): When an attacker has remote access to the physical device.
- A7. App developers: They may observe user behaviour by inserting trackers into the application or by inserting malicious code or may leak information.
- A8. Malware developer: Malware can be designed to target the associated smart-camera application or sensitive data on phone logs and can communicate with the camera if they are on the same network.
- A9. The malware can be designed to infect a component in the network, which will infect the whole network.
- A10. Third parties: Parties, like Amazon platforms for video streaming, web servers, and analytics servers, all have an effect on user privacy.
- A11. Wi-Fi Sniffer: When an attacker tries to discover router credentials and log in to the network as a trusted network component.

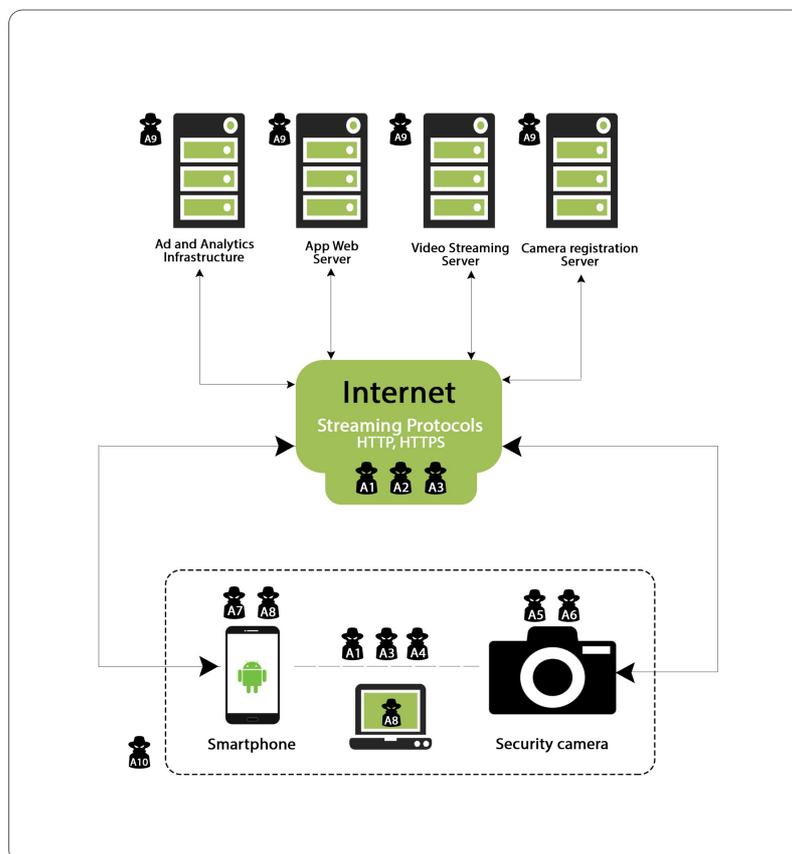


Figure 1 (Smart Camera Environment and Threat Model)

The threat model above suggests a security analysis of the smart-camera systems covering five major components of the system and attack surface: (1) the security of the transmitted video stream; (2) the physical and network security of the smart camera device; (3) the security of the associated web interface; (4) the vulnerability of the associated mobile application to potential information leakage or account hijacking; (5) the privacy policy and agreements which the user signs up to.

In each component, the focus will be on one of the abovementioned types of threats, albeit each component may be affected by more than one type of threat. Communications (TH1) exchanged over the Internet are often a big target for attackers and can be divulged to the public. These communications carry video streams either from the camera to the server, the server to the phone, or the camera to the phone directly. Smart cameras (TH2) threaten the environment if physical security is exposed to an attacker or if its network services are open to attackers. Tampering with these devices can let the attacker collect substantial information about the software running in the camera and its data or access unauthorised information remotely.

The web interface (TH3) is an important part of the system. Weak password policies and weak account lockout mechanisms can enable an attacker to hijack user accounts and compromise user confidentiality and privacy. The Android phone application (TH4) is the medium that the user uses to interact with the camera, which makes it a potential source of vulnerability in the system. Bad application design and implementation can leak and expose users' PII data. A privacy (TH4) analysis of the vendor privacy policy can determine whether the user's private data and the location of the user or device are being disclosed, further than might be expected.

Analysis framework

The Framework analysis components and their respective test cases are illustrated in Table 1. The framework will examine the security of the smart camera system by following some techniques culled mainly from the following sources: security analysis of Android applications for analysing the associated Android application in the smart camera system [4], OWASP testing guides [6, 7], IEEE IoT security best practices [8], Network attackers [5, 9], a privacy and security study of smart scales [10].

Component	Threat Imposed	Test Case Number	Test Case Name	Test Case Description
Communication	TH 1	Test Case I	Camera-Server	Whether the connection between the camera and the server exchanges the video stream.
		Test Case II	Server-Phone	Whether the connection between the server and the phone exchanges the video stream.
		Test Case III	Camera-Phone	Whether the connection between the camera and the phone exchanges the video stream.
		Test Case IV	Camera-Registration server	Whether the connection between the camera and the registration server can be intercepted by a man in the middle attack.
Smart Camera	TH 2	Test Case V	Camera Network Services	Check for open ports, exploitable outdated services running on these ports, URLs that access system information. Update process.
		Test Case VI	Camera Default Passwords	Check for the use of default passwords by the company manufacturer.
		Test Case VII	Camera Physical Security	Whether the device physical ports can be used to access the camera or insert something into it, whether the camera can be put into unsafe state, and whether it has a removable storage media.
Web Interface	TH 3	Test Case VIII	Weak Password Policy	Check whether the password policy along with account lockout mechanisms can add extra layer of security to protect user account from hijacking.
		Test Case IX	Account Lockout Mechanisms	
Android Application	TH 4	Test Case X	SSL Implementation	Check whether SSL implementation is flawed, causing the connection between the phone and server to be under risk of a man in the middle attack.
		Test Case XI	Log file Information Leakage	Check whether user PII is leaked due to bad logging mechanisms by developers.

Component	Threat Imposed	Test Case Number	Test Case Name	Test Case Description
		Test Case XII	Over-privileged Application	Check application manifest permissions with the application features to check for possible backdoors or signs of risky programming.
		Test Case XIII	App Signing	Check the application signing certificate parameters.

Table 1 (Framework Test Cases)

Case Studies

The framework is applied on two types of smart cameras: outdoor and indoor cameras. These smart cameras are:

- The outdoor smart camera product is the Ring Doorbell that is used to enable users to hear and speak to visitors through their smartphones [11].
- The indoor smart camera Netatmo Welcome, which enables users to monitor their home and recognize intruders through the face recognition feature [12].
- The indoor smart camera BT Smart Home Cam 100 that provides home monitoring through smartphones as well [13].

We considered three cameras in depth (Ring, Netatmo, BT) from the start of the study; a further two were added later and

they are MA and UA indoor smart cameras. Both have the same working mechanisms as the first three cameras with having an associated phone application that allow the user to control the smart camera by watching the stream and recording. We elide full details of those devices because the responsibility disclosures are in the process.

Table 2 summarises the case studies under the framework. The table column gives a view of the security issues found for the cameras. The row view gives an indication of how each camera deals with a certain test case. Three classifications are introduced based on the severity of the issues found. Red means that the issues are very dangerous and have a huge effect on user security and privacy. Yellow means it is a warning, and the issues are medium on the severity scale or that it requires some certain settings to be considered a red issue. Green suggests that no issues were discovered regarding this test case.

Test Case Name	Ring Doorbell Smart Camera	Netatmo Welcome Smart Camera	BT Smart Home Camera	MA Smart Home Camera	UA Smart Home Camera
Communication					
Camera-Server	✗ RTP unencrypted video stream.	! SD card contains traces of cryptographic keys that might affect communication.	✓ Encrypted over TCP.	N/A	N/A
Server-Phone	✗ SRTP encrypted with bad key management.	✗ Depends on application SSL implementation that is flawed.	✓ Encrypted over TCP.	N/A	N/A
Camera-Phone	N/A	! HTTP not safe against man at the end attackers or external attackers who try Wi-Fi discovery.	N/A	✓ Over UDP, no footage was directly retrievable but there might be cases that were not covered.	✓ Over UDP, no footage was directly retrievable but there might be cases that were not covered.
Camera-Registration Server	✓ (However, before the update, the man in the middle was applicable).	✓ Does not accept proxy certificate.	✓ Does not accept proxy certificate.	N/A	N/A

Test Case Name	Ring Doorbell Smart Camera	Netatmo Welcome Smart Camera	BT Smart Home Camera	MA Smart Home Camera	UA Smart Home Camera
Smart Camera					
Camera Network Services	✓ No unnecessary open ports. Automated updates.	✓ No unnecessary open ports. Automated updates.	✗ Port 53 runs an outdated version of dnsmasq, system logs leakage, accessible setup page that expose router credential, user intervention needed to update.	✗ An accessible page that prompt the user a password. This password can be the admin password or the password that the user changed it to.	✗ Accessible system log: http://IP-Address/debug/syslogd.txt, open ports (49152/tcp unknown- 554/tcp rtsp- 80/tcp http).
Camera Default Passwords	✓ Depending on the working dynamic of the camera no use of default passwords was apparent; as far as it was tested it seems good.	✓ Depending on the working dynamic of the camera no use of default passwords was apparent; as far as it was tested it seems good.	✗ Username = "admin"; password = "admin".	✗ Makes use of default password "admin" that can be changed by the user but can be reset by the device reset button to "admin".	✗ Makes use of default password "admin" that can be changed by the user and it cannot be reset by the device reset button.
Camera Physical Security	! Can be put into unsafe state, exposes MAC address and serial number of the device.	✗ Removable unencrypted media storage.	✗ Can be put into unsafe state making some pages accessible, exposes MAC address and serial number of the device.	! The SD card is not recognizable, exposes the ID and password of the device.	✗ Removable unencrypted media storage, exposes UID, password and serial number of the device.
Web Interface					
Weak Password Policy	! No password complexity enforced, there are account lockout mechanisms.	✓ Adequate password policy, adequate lockout mechanisms.	✗ No password complexity enforced, no account lockout.	✗ No password complexity enforced, no account lockout.	! No password complexity enforced, (no lockout mechanisms due to the fact that the account is accessible by the phone application).
Account Lockout Mechanisms					
Android Application					
SSL Implementation	✓ Uses a set of certificates defined in pem file.	✗ Flawed SSL implementation, TLS settings does not support the current best practices.	✗ Flawed certificate validation, TLS settings do not support current best practices.	✗ No SSL is used. The user is logged in via HTTP.	N/A doesn't connect to the server it communicates with the camera directly using UDP.
Log File Information Leakage	✗ User PII leakage: home address, cryptographic keys, etc.	✓ No leakage detected.	✓ No leakage detected.	✗ Leaks encryption keys, Wi-Fi credential.	✓ No leakage detected.

Test Case Name	Ring Doorbell Smart Camera	Netatmo Welcome Smart Camera	BT Smart Home Camera	MA Smart Home Camera	UA Smart Home Camera
Over-privileged Application	✓ Reasonably-privileged.	! Semi-privileged READ_CONTACTS WAKE_LOCK FINE_LOCATION.	! Semi-privileged READ_CONTACTS WAKE_LOCK.	✓ Reasonably-privileged.	✓ Reasonably-privileged.
App Signing	✓ Good certificate parameter.	✗ Bad certificate parameter: SHA1WITHRSA.	✗ Bad certificate parameter: SHA1WITHRSA.	✗ Bad certificate parameter: SHA1WITHRSA.	✗ Bad certificate parameter: SHA1WITHRSA.

Table 2 (Framework Component Applied to Commodity Cameras)

From Table 2 we can see that many of the security issues identified affect user privacy such as unencrypted video stream, encrypted video stream with bad key management, leakage of user sensitive information and unencrypted removable storage that stores user's videos. The fifth component involved privacy policy compliance with the OECD principles [14]. Vendors usually dedicate sections to their products' privacy policy. These policies cover data collection, data use, data storage, and the security of user data. The OECD proposed a set of principles to ensure that user privacy is protected with the utmost best practice possible. For example, one of the companies excels in the collection limitation principle by only collecting relevant information, while one of the other companies gathers demographic information and other irrelevant information.

Summary of the results

- **Unencrypted video stream.** Protocols used in video streaming that do not provide any kind of encryption need to be abandoned. Alternatively, a more secure version of the protocols needs to be used. For example, the more secure version, which is the SRTP, needs to be used instead of the RTP used on one of the investigated cameras.
- **Encrypted video stream with bad key management.** Protocols used in video streaming that do not provide secure key management need to be considered while using the protocol. For example, the SRTP that exchanges the keys in plaintext in the SDP/SIP protocols need secure key management using authenticated key establishment (AKE) protocols like TLS, IPsec, etc. [15], [16]. Bad key management renders the encryption mechanisms useless. More efforts are needed to choose the key management of the protocol and encryption mechanisms used.
- **Video stream unencrypted due to blindly trusting internal network.** Loose trust boundaries endanger the video stream and make it more exposed to risk. When the camera and phone reside within the same network, they exchange the video in an unencrypted

way, which means that the camera trusts the network on which it operates. Thus, the threats come from two entities: the man at the end attacker that has a trust relationship with the network or an external attacker who can perform Wi-Fi discovery of the router keys and can access unencrypted traffic of the video stream. Vendors need to tighten the product trust boundaries as much as possible and not put the product trust on another entity. Especially as the assumption that internal WiFi networks are secure is unrealistic in shared environments and increasingly challenged by vulnerabilities such as the recently discovered KRACK key reinstallation attack [17].

- **Accessible URLs that can access system information.** During set up, when cameras create an access point and the Android application is supposed to be connecting to it to perform the setup procedure, the vendors need to secure the access point and provide an authentication for the device connected to it to prevent attackers from impersonating the Android application and inferring a lot of information about the system through a set of URLs.
- **Default passwords.** Default passwords need to be considered seriously by vendors because, if an attacker knows that a product uses default passwords, the number of users who will be affected will be huge. Consequently, vendors need to be contacted to change these default passwords and limit attacker accessibility to these accounts.
- **Video exposure through unencrypted removable physical storage.** When the camera uses an external removable storage, the content of the removable storage needs to be protected, for example, to let the SD card be password protected or to encrypt its content or make the SD card write-protected to protect its content from being altered or deleted. Moreover, access control rules need to be applied to removable data storage [18].
- **Weak password policy and No account lockout.** Strong password policies along with strong lockout mechanisms need to be applied by vendors to ensure that user accounts are safe from hijacking and that

accounts are safeguarded against brute-force attacks.

- **Improper SSL implementation.** Bad SSL implementation needs to be patched by application developers. The code needs to be modified to not trust all certificates. This bad implementation exposes user to being a victim of man in the middle attacks. Certificate pinning needs to be implemented by application developers following available guidelines provided by Moxie Marlinspike, the former head of security at Twitter and the founder of Open Whisper Systems, [19] to eliminate any chances of being vulnerable to a man in the middle attack.
- **Sensitive information leakage through phone logs.** Developers need to prepare the application for release and remove the unnecessary logging information used for debugging, like user sensitive information or cryptographic keys.
- **Over-privileged application.** Developers need to ask for the minimum permissions required, which enables the app to conduct its own tasks without requesting unused or unneeded permissions.
- **Bad certificate parameter.** Developers need to generate or request certificates that use more secure

parameters and comply with the latest safe certificate settings.

Discussion

The range of vulnerabilities discovered based on the proposed threat model and the smart-camera framework on five cameras indicate the strength of the analytical framework components and the corresponding test cases. Table 3 gives a general overview of the smart-camera analysis framework with the vulnerabilities successfully discovered.

A responsible disclosure has been followed to inform the companies about the vulnerabilities discovered in their products. Some vendors accepted direct email reports, others asked us to submit information via shared bug bounty programmes. Some of the companies asked for a more detailed Proof of Concept (PoC) to investigate the security flaws. one of the vulnerabilities was due to the device capabilities that limited it from being able to prevent the vulnerability from existing. Some of the vulnerabilities were known by the vendors and they were in the fix mode and some of them were new.

Component	Threat Imposed	Test Case Number	Test Case Name	Vulnerabilities Discovered
Communication	TH 1	Test Case I	Camera-Server	Unencrypted video stream
		Test Case II	Server-Phone	Encrypted video stream with bad key management
		Test Case III	Camera-Phone	Loose trust boundaries endanger the video stream to internal and external attackers
		Test Case IV	Camera-Registration Server	Launching man in the middle attack * (the vulnerability was launched before firmware update)
Smart Camera	TH 2	Test Case V	Camera Network Services	Open port that runs outdated version of dnsmasq, accessible webpages that expose system information (log, Wi-Fi credentials)
		Test Case VI	Camera Default Passwords	The use of default credential with username = 'admin'; password = 'admin' during setup
		Test Case VII	Camera Physical Security	User videos exposed through the external removable storage, devices can be put into unsecure state
Web Interface	TH 3	Test Case VIII	Weak Password Policy	No password policy and no account lockout mechanisms
		Test Case IX	Account Lockout Mechanisms	
Android Application	TH 4	Test Case X	SSL Implementation	Flawed SSL implementation that caused video stream exposer and enabled man in the middle attacks
		Test Case XI	Log File Information Leakage	User PII is leaked due to bad logging mechanisms by developers

		Test Case XII	Over-privileged Application	Over-privileged application that uses unnecessary permissions
		Test Case XIII	App Signing	Application developer signing certificate parameter with hash algorithm that has collision problems

Table 3 (Framework and Test Cases Evaluation)

A. Related work

A lot of efforts in the last few years have been focused on outlining security and privacy guidelines on Internet of things. In February 2015 “Five Star Automotive Cyber Safety Framework” was proposed [20]. Following that in May 2015 the code of conduct for IoT designers and developers to help produce more secure products was designed [21]. In January of 2016 security and privacy guidelines were proposed for Connected Medical Devices [22]. GSM Alliance created their IoT guidelines in February of 2016 [23]. Following after that many security and privacy guidelines in the year of 2016 had been proposed: the OWASP security guidelines [6], [24] by OneM2M, [25] by Broadband Internet Technical Advisory Group, [26] by the US Department of Homeland Security, [27] and [28] by Industrial Internet Consortium. In 2017 the IoT Security & Privacy Trust Framework by the Online Trust Alliance was proposed [29]. These guidelines go over the same material of the kinds of vulnerabilities that can be encountered in an IoT device [30].

Researchers began to create frameworks for analysing these kinds of vulnerabilities in IoT. The work of Tekeoglu et al. in [31] suggested a test bed to analyse the security and privacy of IoT devices. Their testing environment was based on creating two access points, one to connect all the devices to it and the other one to connect the phone to it and by making use of Kali Linux operating system and its pre-installed tools captured all the traffic generated from the IoT devices. They used their testbed in examining a set of commercial IoT devices like IP cameras, drones, activity trackers and smartwatches. However, their work was more focused on the testbed setting without providing a methodological way of examining the devices. Dhanjani in [32] carried out specific kinds of attacks on specific kinds of IoT devices such as exploiting Default credential in Foscam and SSL certificate validation in SmartThings etc. Their discussions provide details on certain types of attack that could or could not necessarily work on other devices. Tekeoglu et al. in [33] investigated the security and privacy of NetCam camera and went into great depth to focus on specific attacks against that camera but their investigation was limited to one kind of camera. We find that the work in this field is either too general, i.e., the security and privacy guidelines provide a general overview of the kinds of vulnerabilities that should be avoided or too specific, i.e., the researchers focused on discovering vulnerabilities in specific IoT devices leading to no generalizations. Therefore, our work aims in the middle and provides a framework for analysing security cameras and provides test cases that might be generalized to any IoT device that has the same working mechanisms.

Conclusions and Future Work

Prior work in the IoT field is either too generic [34] or too specific to the kind of IoT or the brand of IoT device [32]. Alternatively, it focuses on aspects of IoT and leaves other aspects [35]. Since it is an active area of research and since guidelines are still under development [6, 8] and several more, there is no guide for the analysis framework for smart cameras for domestic use.

Our work has provided an analysis framework for future studies for assessing domestic smart cameras. The analysis framework can be generalised to other kinds of IoT devices and can be used by security experts to analyse products or by vendors to prevent security breaches from vulnerabilities. The trivial kinds of vulnerabilities found, such as unencrypted video stream, removable unencrypted storage media, applications with flawed SSL implementation, leakage of PII user data, and so on, emphasise the need for this kind of research and the need to spread awareness about security among vendors.

The kinds of breaches discovered by this paper are sufficient indicators that security is not a priority for vendors. The way that products are advertised gives the impression that they ensure security and protect user privacy. Moreover, the frequently asked questions section does not often reflect what happens behind the scenes with the products in general. This emphasises the need for such security analysis to better secure such devices and protect users’ privacy, and the need to provide independent kite-marking or similar to certify that device has passed tests.

References

- [1] R. Alharbi, “IoT under Lock and Key: An Investigation into Smart Camera Vulnerabilities,” University of Edinburgh, 2017.
- [2] C. Koliass, A. Stavrou, J. Voas, I. Bojanova, and R. Kuhn, “Learning Internet-of-Things Security ‘Hands-On,’” *IEEE Secur. Priv.*, vol. 14, no. 1, pp. 37–46, 2016.
- [3] “Secure by Design : Improving the cyber security of consumer Internet of Things Report.”
- [4] K. Knorr and D. Aspinall, “Security testing for Android mHealth apps,” *2015 IEEE 8th Int. Conf. Softw. Testing, Verif. Valid. Work. ICSTW 2015 - Proc.*, no. Sectest, 2015.

- [5] A. Akhunzada *et al.*, “Man-At-The-End attacks: Analysis, taxonomy, human aspects, motivation and future directions,” *J. Netw. Comput. Appl.*, vol. 48, pp. 44–57, 2015.
- [6] “IoT Testing Guides - OWASP.” [Online]. Available: https://www.owasp.org/index.php/IoT_Testing_Guides. [Accessed: 02-Aug-2017].
- [7] “Authentication Cheat Sheet - OWASP.” [Online]. Available: https://www.owasp.org/index.php/Authentication_Cheat_Sheet#Implement_Proper_Password_Strength_Controls. [Accessed: 06-Aug-2017].
- [8] G. Corser, G. Fink, M. Aledhari, and J. Bielby, “IEEE Internet Technology Policy Community White Paper INTERNET OF THINGS (IOT) SECURITY BEST PRACTICES,” no. February, 2017.
- [9] J. F. Kurose and K. W. Ross, *Computer Networking A Top-Down Approach*, no. 5. 2013.
- [10] M. Krämer, “Health Monitors Under The Magnifying Glass: A Privacy And Security Study,” no. October, 2016.
- [11] “HD Video Doorbells With Two-Way Talk and Advanced Motion Detection | Ring.” [Online]. Available: <https://ring.com/videodoorbells>. [Accessed: 15-Aug-2017].
- [12] “Netatmo Welcome | Home security camera with face recognition.” [Online]. Available: <https://www.netatmo.com/en-GB/product/security/welcome>. [Accessed: 15-Aug-2017].
- [13] “BT Smart Home Cam 100 (077232) | BT Shop.” [Online]. Available: <https://www.shop.bt.com/products/bt-smart-home-cam-100-077232-9C5D.html>. [Accessed: 15-Aug-2017].
- [14] OECD, “Recommendation of the Council concerning Guidelines governing the Protection of Privacy and Transborder Flows of Personal Data (2013),” *OECD Priv. Framew.*, pp. 11–37, 2013.
- [15] D. Wing, F. Andreasen, and M. Baugher, “Session Description Protocol (SDP) Security Descriptions for Media Streams,” pp. 1–44, 2006.
- [16] J. Arkko, F. Lindholm, M. Naslund, and K. Norrman, “Key Management Extensions for Session Description Protocol (SDP) and Real Time Streaming Protocol (RTSP),” pp. 1–30, 2006.
- [17] M. Vanhoef and F. Piessens, “Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2 Mathy,” *Proc. 2017 ACM SIGSAC Conf. Comput. Commun. Secur. - CCS '17*, pp. 1313–1328, 2017.
- [18] “Top 5 Ways to Protect a MicroSD Card | eBay.” [Online]. Available: <http://www.ebay.com/gds/Top-5-Ways-to-Protect-a-MicroSD-Card-/10000000178723271/g.html>. [Accessed: 11-Aug-2017].
- [19] M. Marlinspike, “Moxie Marlinspike - Your app shouldn't suffer SSL's problems.” [Online]. Available: <https://moxie.org/blog/authenticity-is-broken-in-ssl-but-your-app-ha/>. [Accessed: 11-Aug-2017].
- [20] O. August *et al.*, “Five Star Automotive Cyber Safety Framework,” no. February, pp. 1–5, 2015.
- [21] R. van der Vleuten *et al.*, “IOT Design Manifesto: Guidelines for responsible design in a connected world,” *Beyond.io*. 2015.
- [22] I am The Cavalry, “Hippocratic Oath for Connected Medical Devices,” pp. 1–5, 2016.
- [23] GSMA, “IoT Security Guidelines Overview Document Antitrust Notice,” 2016.
- [24] ONEM2M, “ONEM2M TECHNICAL REPORT,” vol. 1, pp. 1–35, 2016.
- [25] F. Baker *et al.*, “Internet of Things (IoT) Security and Privacy Recommendations,” no. November, 2016.
- [26] U.S. Department of Homeland Security, “Strategic Principles for Securing the Internet of Things (IoT),” pp. 1–17, 2016.
- [27] “Principles, Practices and a Prescription for Responsible IoT and Embedded Systems Development,” pp. 1–22, 2016.
- [28] Industrial Internet Consortium, “Industrial Internet of Things Volume G4 : Security Framework,” *Ind. Internet Consort.*, pp. 1–173, 2016.
- [29] Internet Society Initiative, “IoT Security & Privacy Trust Framework v2 . 0,” pp. 1–6, 2017.
- [30] “Security and Privacy Guidelines for the Internet of Things - Schneier on Security.” [Online]. Available: https://www.schneier.com/blog/archives/2017/02/security_and_pr.html. [Accessed: 17-Aug-2017].
- [31] A. Tekeoglu and A. Ş. Tosun, “A Testbed for Security and Privacy Analysis of IoT Devices,” *Proc.*

- 2016 IEEE 13th Int. Conf. Mob. Ad Hoc Sens. Syst. MASS 2016, pp. 343–348, 2017.

- [32] N. Dhanhani, *Abusing the Internet of Things - Blackouts, Freakouts, and Stakeouts*. 2015.
- [33] A. Tekeoğlu and A. Ş. Tosun, “Investigating security and privacy of a cloud-based wireless IP camera: NetCam,” *Proc. - Int. Conf. Comput. Commun. Networks, ICCCN*, vol. 2015–Octob, 2015.
- [34] A. Sedrati and A. Mezrioui, “Internet of Things challenges: A focus on security aspects,” *2017 8th Int. Conf. Inf. Commun. Syst. ICICS 2017*, pp. 210–215, 2017.
- [35] K. A. Hua, “Internet of Things: Challenges and opportunities for collaborative technologies (invited Talk),” *Proc. - 2016 Int. Conf. Collab. Technol. Syst. CTS 2016*, pp. 613–614, 2016.