FIRAT UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
TÜRKİYE

IOT SECURITY ISSUES: A CASE STUDY ON VULNERABILITIES IN SURVEILLANCE SMART CAMERAS

Peshraw Ahmed ABDALLA

Master's Thesis

DEPARTMENT OF SOFTWARE ENGINEERING

AUGUST 2020

FIRAT UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
TÜRKİYE

Department of Software Engineering

Master's Thesis

IOT SECURITY ISSUES: A CASE STUDY ON VULNERABILITIES IN SURVEILLANCE SMART CAMERAS

Author
Peshraw Ahmed ABDALLA

Supervisor
Assoc. Prof. Dr. Cihan VAROL

AUGUST 2020
ELAZIG

DECLARATION

I hereby declare that I wrote this Master's Thesis titled “IoT Security Issues: A Case Study on Vulnerabilities in Surveillance Smart Cameras” in consistent with the thesis writing guide of the Graduate School of Natural and Applied Sciences, Firat University. I also declare that all information in it is correct, that I acted according to scientific ethics in producing and presenting the findings, cited all the references I used, express all institutions or organizations or persons who supported the thesis financially. I have never used the data and information I provide here in order to get a degree in any way.

7 August 2020

Peshraw Ahmed ABDALLA

PREFACE

The purpose of this thesis is to investigate vulnerabilities of IoT type surveillance devices, such as IP cameras and baby monitors to display the types of vulnerabilities that exist in surveillance smart cameras and to explain their impacts on security and privacy of users. The means to protect these things are seeming more slowly than our increasing dependence on them; more ideas are inserted into the usability of the IoT devices and fewer on their security. This motivates us to offer a study framework for inspecting these devices. In this study, three IP cameras with three different brands will be tested to find vulnerabilities.

All praises and thanks are to Allah for giving me the ability to study, health and knowledge to finish my master thesis work. I would like to thank my supervisor, in Software Engineering Department Assoc. Prof. Dr. Cihan VAROL for his support, patience, motivation, all along with my thesis. Who has always helped me and his door was always open to answer me anytime when I got a problem with my thesis and wish him a long life with gladness.

I would also like to express my deepest gratitude to my dear parents and my whole family for their continuous support, motivation, and encouragement throughout my years of study, and they had a great role and always supported me to continue my studying. Also, I would like to thank my friends for all their support.

Finally, I dedicate this thesis to my parents who always support me; to my beloved wife; to my faithful brothers and sister; to my handsome son; to all persons who try to learn science.

Peshraw Ahmed ABDALLA

ELAZIG, 2020

TABLE OF CONTENTS

Page

ABSTRACT

IoT Security Issues: A Case Study on Vulnerabilities in Surveillance Smart Cameras

Peshraw Ahmed ABDALLA

Master's Thesis

FIRAT UNIVERSITY
Graduate School of Natural and Applied Sciences

Department of Software Engineering

August 2020, Page: xii + 52

The number of internet-connected devices is growing at a fast rate. A large number of these machines are called the internet of things (IoT) machines. IoT devices can be conveniently associated and used via the internet. However, a considerable number of these devices are made with security vulnerabilities and defects that result in them being obvious targets for hackers. This issue has been extensively discussed in the media recently.

An internet protocol (IP) camera is a type of digital video camera that can transmit information over the internet. These cameras are used for several purposes, such as home and industry monitoring, as well as monitoring for babies and the elderly. Also, with the remote access feature of these cameras, a live video stream can be viewed through a number of devices, including smart-phones, tablets, computers, or any other handheld device. IP cameras are usually strong, both in transmission bandwidth and computing, rendering them particularly appealing to hackers, who can exploit these devices with multiple attacks. Distributed Denial-of-Service is one of the attacks that IoT devices, such as IP cameras, are faced with.

The purpose of this thesis is to investigate the vulnerabilities of IP cameras and the impact of these vulnerabilities on the security and privacy of IP camera users. This task will be carried out using a Kali Linux penetration testing platform and its tools. To this end, a hands-on test was be performed on three different brands of IP camera (an Intelligent Onvif YY HD camera, a Panoramic IP camera, and a Dual-HD camera), as well as analyzing the security components of these cameras. The outcomes of this study indicate that IP cameras have weaknesses and security problems, that such flaws have various security effects on users. One security flaw of IP cameras is their default usernames and passwords. To prevent attackers that try to access these devices without permission (for instance, using Mirai malware), the end-user must modify the default credentials. The implications and effects of security gaps such as a weak password policy for setting passwords, encryption of sensitive information with weak methods, and sending data without any type of encryption, are discussed in detail. This thesis will also discuss solutions and recommendations to improve the security of IoT devices, especially that of IP cameras.

Keywords: Internet of Things, Security, Vulnerability, IP Camera, Kali Linux, Wireshark, Nmap, Mitmproxy, Bettercap, Penetration Testing, Ethical Hacking.

ÖZET

IoT Güvenlik Sorunları: Akıllı Kameralarda Sistem Açıkları Üzerine Deneysel Bir Çalışma

Peshraw Ahmed ABDALLA

Yüksek Lisans Tezi

FIRAT ÜNİVERSİTESİ
Fen Bilimleri Enstitüsü

Yazılım Mühendisliği Anabilim Dalı

Ağustos 2020, Sayfa: xii + 52

İnternete bağlı cihazların miktarları hızla artıyor. Bu makinelerin çoğuna IoT makineleri denir. IoT cihazları İnternet üzerinden rahatlıkla ilişkilendirilebilir ve kullanılabilir. Bununla birlikte, bu cihazların önemli bir kısmı, bilgisayar korsanları için belirgin kurbanlar olmalarını sağlayan güvenlik açıkları ve kusurları ile üretilmektedir. Bu son zamanlarda medyada yoğun bir şekilde tartışıldı.

IP kameralar gibi gözetim cihazları, genellikle uzaktan erişim özelliği ile internet üzerinden bilgi iletebilen, ev, sanayi, yaşlı ve bebek izleme gibi çeşitli ortamlarda ve amaçlar için kullanılan dijital video kameralardır. Canlı video akışı, akıllı telefon, tablet, bilgisayar veya diğer elde taşınır cihazlar gibi çeşitli cihazlardan görülebilir. IP kameralar genellikle hem bant genişliği hem de bilgi işlem açısından güçlüdür ve korsanlar birden fazla saldırı için bu cihazlardan yararlanabildikleri için kendilerine özellikle çekici hale gelir.

Bu tezin amacı, IP kameraların güvenlik açıklarını araştırmak ve Kali Linux sızma testi platformunu ve araçlarını kullanarak kullanıcıların güvenliği ve gizliliği üzerindeki etkilerini açıklamaktır. Bu amaçla, tez üç farklı IP kamera üzerinde uygulamalı bir test yapar ve bu kameraların güvenlik bileşenlerini analiz eder.

Bu çalışmanın sonuçları, IP kameraların zayıflıkları ve güvenlik eksiklikleri olduğunu ve bu kusurların kullanıcılar üzerinde çeşitli güvenlik etkileri olduğunu göstermektedir. IP kameralarının güvenlik kusurlarından biri fabrika çıkışlı ayarlanmış kullanıcı adları ve şifrelerinin kullanıcılar tarafından değiştirilmemesidir. Aygıtlara yetkisiz erişime çalışan saldırganların (örneğin, Mirai kötü amaçlı yazılımları) önlenmesi için, son kullanıcının varsayılan kimlik bilgilerini değiştirmesi gerekir. Şifrelerin ayarlanması için zayıf şifre politikası, hassas bilgilerin zayıf yöntemlerle şifrelenmesi veya herhangi bir şifreleme olmadan veri gönderilmesi gibi güvenlik açıklarının etkileri ve etkileri, güvenlik çözümleri ve IoT cihazlarının güvenliğini iyileştirmek için öneriler ile birlikte ayrıntılı olarak tartışılmaktadır.

Anahtar Kelimeler: Nesnelerin İnterneti, Güvenlik, Güvenlik Açığı, IP Kamera, Kali Linux, Wireshark, Nmap, Mitmproxy, Bettercap, Penetrasyon (Sızma) Testi, Etik Hackleme.

LIST OF FIGURES

Page

LIST OF TABLES

Page

ABBREVIATIONS

AES	: Advanced Encryption Standard
ARP	: Address Resolution Protocol
CRC	: Cyclic Redundancy Check
CVE	: Common Vulnerabilities and Exposures
DDNS	: Dynamic Domain Name System
DDoS	: Distributed Denial-of-Service
DES	: Data Encryption Standard
DoS	: Denial-of-Service
HD	: High-Definition
HTTP	: Hypertext Transfer Protocol
IoT	: Internet of Things
IP	: Internet Protocol
LAN	: Local Area Network
MAC	: Media Access Control
MITM	: Man-in-the-Middle
P2P	: Peer-to-Peer
PII	: Personal Identifiable Information
PTZ	: Pan–Tilt–Zoom
RSA	: Rivest-Shamir-Adleman
RTSP	: Real Time Streaming Protocol
SD	: Standard Definition
SSID	: Service Set Identifier
TCP	: Transmission Control Protocol
UDP	: User Datagram Protocol
UI	: User Interface
URL	: Uniform Resource Locator
VSS	: Video Surveillance System

1. INTRODUCTION

The appearing IoT or Internet of Things, while billons of smart machines are connected to each other, is assumed to the next internet generation and will become a charming goal for attackers [1]. In the IoT, any physical devices can communicate with or without the interference of humans [2]. Nowadays, a different type of applications with various infrastructures has been produced, for example, healthcare, logistics, manufacturing, industrial monitoring, etc. But, technologies for the IoT are however in their child steps and several technological challenges correlated with IoT must be defeated. Security is now one of IoT's most critical barriers, which includes the security at several layers such as infrastructure sensing security, the security of transmission network, security of applications, and security of the overall system [3].

IoT is based on the idea of numerous hardware devices (Things), including sensors, being fixed with software and apps able to process and transmit data. These new easy-to-use devices have the ability to make decisions at any necessary time. The IoT presents connectivity for anything at anyplace and anytime to anyone [4]. The advancement of IoT's intelligent machines growing quicker and quicker, today most persons use at least one or more machines joined to the Internet, these devices are located at different places to collect several types of information.

Surveillance devices such as IP cameras are kind of digital video cameras usually used for monitoring, which can transmit information over the internet also, with the feature of remote access, the live video stream can be seen through several devices, for instance, smart-phone, tablet, computer or any other handheld device.

IP cameras are commonly used in a variety of environments, and the safety of these devices continues to be an essential problem. Types of security concerns that could reveal IP cameras could be: default username and password, credentials that are simple to imagine, weak password policy for setting the passwords, encryption sensitive information and data with weak methods or send data without any type of encryption and more.

Different clients were aware of potential security jeopardies. In addition, if the IP camera is dispatched with the newest protection characteristics, the operations of users can begin to security issues. If the device's default settings are not modified by the user or missing to renew to the newest firmware can put the cameras in an unsafe situation [5].

In this study, three IP cameras (three different brands) will be tested to find vulnerabilities. Also, to implement this task and perform a hands-on test, these steps are required:

- Information gathering for the purpose of device identification, network traffic capture, port and service identification
- Traffic analysis
- Inspect mobile application (controller) vulnerabilities

1.1. Thesis Aims

The purpose of this thesis is to investigate vulnerabilities of IoT type surveillance devices, such as IP cameras and baby monitors to display the types of weaknesses that exist in surveillance IP cameras and to explain their impacts on the privacy and security of users. During the course of the thesis, the followings are conducted:

a. Exposing the vulnerabilities of IP surveillance cameras,

b. Inspecting the IP camera mobile applications,

c. Comparing the results obtained from the testing of cameras,

d. Disclosing vulnerabilities disclosure for vendors and users.

To reveal vulnerabilities in the cameras, this thesis will implement some penetration testing and ethical hacking methods by using Kali Linux tools. We will investigate via several Kali tools and ethical hacking methods such as Man in the middle (MITM) and sniffing to observe network traffic, in order to understand and analyze security components in three different brands of IP cameras (Intelligent Onvif YY HD camera, Panoramic IP camera, and Dual-HD camera).

In the end, the thesis will compare the results, address and explore to find ways to provide better security for these devices as well.

1.2. Motivation

The improvement of intelligent things identified as the (IoT) Internet of Things is growing faster and faster, one of the most IoT devices that widely used by the clients are IP cameras and baby monitors. A baby monitor can be used to remotely monitor if the kid is fine, by joining the device to a Wi-Fi with the remote-control feature. This could be performed through, for instance, a mobile phone app.

The means to protect these things are seeming slower than our increasing dependence on them; more ideas are inserted into the usability of the before-mentioned things and fewer on their security.

This motivates us to offer a study framework for inspecting devices, for instance, IoT smart cameras. Smart cameras for house surveillance have been broadly used and have been extended by various manufacturers.

In fact, the attacker tries to utilize IP cameras in DDOS attacks because these devices have large bandwidth, this makes these devices so suitable to be utilized by attackers in this type of attack. All these offerings for home monitoring cameras are really exciting IoT devices to analyze while it comes to security [6].

1.3. Significant Terminologies

Vulnerability

It is described within the devices or systems as a weakness or a flaw that is possible to be utilized to enter and access a system in an unauthorized way.

Threat

It introduces a potential hazard to IoT devices and systems. It is something a company does not wish to occur. Sometimes a threat may be malicious software or an attacker who attempts to obtain illegal entry to the IoT devices and systems.

Exploit

It is anything that tries to reveal the weakness in an environment to induce unwanted or unexpected activity in a specific framework that may be enabling an abuser to obtain sensitive data in a system.

Risk

It is described as the effect (impact) that results from an organization's productive compromise. For instance, an enterprise operating a weak server (e.g. apache tomcat), presents a danger to an environment and the loss or damage that is caused to the resource is described as a risk. Commonly, the below equation can be executed to measure risk [7]:

$$\text{Risk} = \text{Vulnerability} * \text{Threat} * \text{Impact}$$

1.4. Overview of Thesis

This thesis contains six major chapters: the first chapter includes a comprehensive introduction to the research and its components, the second chapter discusses related works and literature review in the field of IP cameras security, the third chapter demonstrates the general architecture of IoT and its security issues while presenting some securing mechanisms, the fourth chapter discusses the requirements and details on penetration tools, the fifth chapter presents the IP cameras tests and discusses the results obtained from the pen-testing process, finally the sixth chapter presents the conclusions, a list of suggestions for security solutions, and future works.

2. BACKGROUND AND RELATED WORK

According to previous studies, the number of IoT devices is growing at a fast rate [8], [9]. The amount of internet-connected devices in 2008 exceeded the world's population and is expected to exceed 50 billion by 2020 [8]. According to the IHS Markit research report, 108 million network surveillance cameras are estimated to be deployed globally in 2019 [9].

The authors, Seralathan *et al.* [6] essentially examined an IP camera that is joined to the internet and available to anybody on the internet (the brand name of the camera that is used in this research paper is not mentioned by the authors). They analyzed the device in order to find some vulnerabilities on it. Also, they discussed questions of how and why current malware directly attacks the IoT things. Their paper utilized two tools for the purpose of information gathering and analyze security components, Nmap and Wireshark. Nmap was used to find the camera's IP address and Wireshark was employed to monitor network traffic for the purpose of capturing all packets in network-level. According to their findings, the device was vulnerable in three main areas:

- Unencrypted communication: not utilizing a safe transmission network, all sensitive information and data have been sent in plaintext
- Brute Force attack to find RTSP URL
- Credentials of the IP camera app stored in plaintext

This article limits its study to the security implementation of IoT devices. Finally, it explained the value of protecting IoT by several suggestions to improve the security of IoT devices and present basic safety methods to decrease the chance of exploitation.

The authors, Alharbi and Aspinall [10] conducted a hands-on test on five different smart cameras, one of them is an outdoor camera (Ring Doorbell) and four of them are indoor cameras (Netatmo, BT, MA, and UA). They illustrated the types of weaknesses that exist in surveillance smart devices and explained their impacts on the privacy and security of users, via introducing a model of threat and a framework for privacy and security analysis.

Furthermore, this paper found many security issues in these cameras including default credentials, weak password policy, no account lockout, unencrypted video stream, encrypted video stream with poor key management, users' sensitive information leakage, unencrypted removable storage that saves the videos and security issues in android applications that used to control the cameras. Generally, this paper focused on four types of threats such as:

- Unauthorized streaming of video
- Manipulation of the intelligent camera
- The unauthorized hijacking of the account
- The capture of personally identifiable information data

Finally, the security gaps found in this article are enough signs that reflected safety is not a vendor's priority.

Tekeoglu and Tosun [11] examined the security problems of a specific camera type known as cloud-based cameras, the wireless IP camera that utilized in this article was mentioned as a "Belkin Netcam" camera. They inspected the traffic produced by this affordable device. Furthermore, their device has the ability to broadcast the video stream through the internet or via Wi-Fi to a smartphone.

Also, they studied the security effects obtained from the IP cameras producers and estimated the procedures of access control. In addition, in their study, they utilized several non-commercial and open source tools and found a number of issues in security when using these products, varying from mild to serious problems. According to this study, if an attacker or network-sniffer observes the network traffic among the camera, servers of the camera, and smartphone path, by traces packet capture it is possible to rebuild unencrypted JPEG images that transmitted via the network. Finally, the outcomes of this study demonstrate that currently common and affordable wireless IP cameras linked to the cloud do not enforce privacy and security as they should.

Liranzo and Hayajneh [12] illustrated the existing privacy issues by getting a cloud-based IP camera (D-Link DCS-930L) correlated to the internet in a specific location and bad security controls that could contribute to security and privacy breaches. This paper discussed the different security threats and forms in which the system may be compromised either by anyone using the same LAN network or from the outside of the local network. Also, this article suggested some mechanisms to decrease the existing challenges and protect the security and privacy of the users. Finally, it recommended the users to not buy those products that put their privacy at risk.

Bugeja *et al.* [13] in their research, used the search engine of the Shodan and CVE database to implement an evaluation of global vulnerability. Besides, they used the Shodan search engine to identify and detect intelligent on-line cameras on the Internet.

Also, they addressed whether the observed sensitive information can be used to jeopardize individuals' privacy and security furthermore describe certain mitigations that can be implemented. Their framework included three steps, information gathering, data extraction, and analyze the vulnerabilities. In the third stage (analyze the vulnerabilities), they found thousands of intelligent cameras that broadcast different parts of information utilizing the Shodan search engine with “uc-httpd” keyword. The findings of this paper demonstrated that there are certainly a huge number of smart cameras that are vulnerable to multiple flaws in security and privacy.

Costin [14] presented a comprehensive security review of CCTV, VSS, and IP cameras by identifying risks, flaws, assaults, and mitigations in-depth, according to information available to the public, taxonomies, and existing classifications. The author offered detailed information on methods of attacking and defending video monitoring devices at various levels. This made it

possible to use this structured knowledge to better explain and detect the security and privacy risks involved with developing, deploying and using these systems. Furthermore, the author compiled this information into a detailed table that can be used as a safety checklist when reviewing the cybersecurity standards of existing or new CCTV models and implementations in a practical manner.

This article also provided a set of guidelines and mitigation strategies capable of improving the privacy and security aspects of surveillance cameras such as network communication, hardware, and firmware.

The authors, Obermaier and Hutle [15] implemented a hands-on test on four different types of VSS and analyzed their security and privacy components. They found different types of security gaps and vulnerabilities in the devices by utilizing some reverse-engineering mechanisms. Two types of attackers are considered by the researchers, including internal (LAN network) attackers and external (remote) attackers.

Also, this paper explained how VSS's weaknesses can be abused by these attackers to exact users and organizations to DoS attacks, injecting streaming video, and monitoring private video data even without enter to the system physically. Their study revealed major deficiencies in the development and implementation of the testing devices, most security issues are caused by very basic security design errors, for instance, weak passwords, unsecured authentication, and proprietary security protocols.

3. INTRODUCTION TO IOT SECURITY ISSUES

This chapter explains the term (IoT) as well as demonstrates the general architecture of IoT and its security issues with presenting some securing mechanisms to prevent and protect the IoT devices from the possible attacks.

3.1. Internet of Things

When the internet appeared, just a few devices were connected to the internet. The number of internet-connected devices was started to grow quickly in the ’90s, and it looks like it is increasing at a faster rate and more things become attached. IoT works with linking hardware devices to the internet in order to transmit or receive customized data.

The term "Internet of Things" was mentioned by Kevin Ashton for the first time in 1999 through a presentation [16].

These devices (Things) involve individual computers, laptops, smartphones, tablets, PDAs, and smart surveillance devices. Largest portable devices install various actuators and sensors that have the ability to sense, make computation, take smart decisions, and send helpful gathered data and information across the Internet. Utilizing a network of such machines (Things) with various sensors have the ability to provide birth to huge wondrous services and applications that can bring important personal, economic, and professional advantages [17].

Most of these things are normal machines that we were previously unable to access the internet, such as a TV. These machines are sometimes referred to as "smart things" and we add the word "smart" before them, for instance, "smart TV".

The major variation within a normal TV and the named smart TV is that there is a Wi-Fi feature for smart TVs to be controlled via the internet. Intelligent things are often loaded with computing ability and some machines containing minimized operating system versions such as Linux, etc. [5].

3.2. IoT Architecture

Every layer in the IoT system is determined through their roles and the things that utilized in the layer. The number of layers in the IoT are varied from one research to another.

According to several studies, the IoT works primarily on four layers called (perception, communication, middleware, application) that are briefly described below [18].

3.2.1. Perception Layer

This layer in the IoT architecture contains various data sensors characters such as 2D-barcode, RFID and other types of sensors. This layer's primary goal is to recognize the individual things according to their gathered information and data collected from the environment with their corresponding sensor(s) assistant [19]. These sensors have several characteristics in terms of IoT such as:

- To gather information from the environment, the applications of IoT requires sensors
- These sensors may be small in size, low cost, and consume low energy
- Different sensing kinds are utilized in the development of intelligent applications

3.2.2. Network Layer

This layer is designed to transfer the collected data received from the layer of perception to layer of middle-ware via existing networks of delivery including the mobile networks, Internet and other types of a secure system. Also, it is possible to call this layer to "Communication Layer".

The communication mechanism may be wireless or wired and depending on the sensor types, the system may be 5G,4G,3G, Wi-Fi, Ethernet, Bluetooth, ZigBee and so on.

Generally, the IoT network layer and its components have some characteristics and roles such as:

- Identification and addressing each smart device need to be recognized with a specific address inside a network
- Routing protocol with the requirement of memory and effective protocols for communication, high rate and contact without delay

3.2.3. Middle-Ware Layer

This layer includes technologies to store data such as cloud storage with the database which presents storage capacities to the gathered information and also, consists of data processing methods. The middle-ware layer is service-oriented that provides the same kind of service between the connected devices [4].

3.2.4. Application Layer

This layer achieves many useful IoT implementations according to the user requirements and numerous types of applications like smart environment, smart home, smart hospital, smart surveillance system, smart transportation and so on [20].

3.3. Security Aims for IoT

IoT's main security aims are to guarantee that correct techniques of authentication are applied and that the data is confidential. The CIA or security triad is a famous model for the implementation of safety tools, executes security by using the three sectors including confidentiality, integrity, and availability [18].

3.3.1. Confidentiality of Data

Ensuring that the information is protected and only accessible to permitted clients is quite essential. Throughout IoT, a client can be a person and different things, as well as inner devices (inside the LAN network) or external devices (outside the LAN network).

For instance, ensuring that sensors do not expose the gathered information to nearby nodes is important. Another problem of confidentiality which needs to be considered is how to handle the information.

It is essential for IoT consumers to be mindful of the information management processes to be implemented, the system responsible for the administration, and to guarantee that the data is safe in the system[21].

3.3.2. Integrity of Data

Information may be changed by malicious hackers at the point of the interaction cycle or may be altered by numerous other reasons behind personal control such as server crash.

The integrity of data applies to the security of users' data from the attackers or the outer intervention at the time of transmission and reception with any methods of standard tracking so that without the system catching the threat, the data cannot tamper.

Techniques for maintaining authenticity and accuracy of information involve tools such as CRC and checksum that are basic processes for a portion of information to identify errors [22].

3.3.3. Availability of Data

Another main purpose of IoT security is to create information accessible to clients, whenever demanded (on demand). Availability of information means that licensed entities have quick access to their data assets not just to standard conditions, but also in severe situations.

Because of the company's reliance on it, it is important to present firewalls to countermeasure attacks on networks such as DoS attacks that may refuse the clients to access the system data [18].

3.4. Possible Attacks Against IoT Devices

IoT devices are vulnerable to several types of cyber-attacks this section explains the most common cyber-attacks against IoT devices and IP cameras.

3.4.1. MITM Attack

The man-in-the-middle attack prevents connection among two points. Two nodes are deceived into believing that they connect with one another safely, while the criminal simply sits among them and interacts with both.

The attacker in this type of attack is attacking privately to that data transfers and possibly changes the transmission between two things who think they are straight communication with each other. Two main types of man-in-the-middle attacks involve eavesdropping and modification [23].

3.4.2. DoS Attack

Denial-of-service is another type of attack against the IoT devices to make a device or network service inaccessible to its expected clients. Many systems in IoT are possible to resource enervation attacks due to low storage capacity and restricted processing power.

There is a large proportion of DoS attacks that could be conducted towards the IoT, like jamming streams, use of computer resources such as bandwidth, storage, disk space, and processing power [24].

3.4.3. DDoS Attack and Default Credentials

It can be a major issue to have default credentials such as passwords and usernames that come with the company's device. If a client does not modify an equipment's default passwords and usernames, it may be vulnerable to use via attackers in DDoS attacks and make them a botnet like Mirai. For the purpose of entering the systems and make the IoT devices to a botnet, the Mirai malware runs via the common log in information in a file for the particular vendors [25].

The distributed denial-of-service or DDoS attack is a well-known attack type against IoT devices to make botnets. Like the DoS attack, DDoS tries to make a device or network service inaccessible to its clients. A target's received traffic flood originally comes through various sources, making it impossible to avoid the cyber-criminals simply by preventing a single source.

For instance, “Mirai” is a malware that transforms connected machines running Linux becoming remote-controlled called "bots" which could be applied in large-scale network attacks as botnets. This specifically addresses IoT devices including IP cameras and other home-based devices [26].

3.4.4. Eavesdropping

Eavesdropping on traffic of network is definitely a feasible method if data is transmitted in plaintext including through an unencrypted Wi-Fi network.

By the way, to protect communications from eavesdroppers while the clients exchange the information and data over Wi-Fi or any network types, the communications always should be protected.[27].

3.4.5. Wi-Fi Sniffing

This method has only been used to track network traffic by intercepting frames forwarded by machines and monitoring for signals already accessible in the network without inserting any additional signals [28]. For instance, in some cases, the attackers attempt to log in to the network by finding the credentials of the router [10].

3.4.6. Brute-force Attack

The brute force term indicates the unauthorized attempt to break into a system's back end to have the log in detail such as username and password.

The attackers can break into a specific device or a system via utilizing the brute-force word list, usually, they make an educated guess since users often use their own names, the names of their kids, addresses, etc. Recently, most companies are adapting to this real issue and they are beginning to use two-factor authentication and biometrics to eliminate these threats [29].

3.5. IP Cameras Expose by the Cloud

Services are cloud-hosted so that devices and applications can be reached at any time and from anywhere. This removes access limits, but at the same time contains security risks [30]. While an IP camera begins, this IP camera joins to the cloud as a backend and holds the link alive while the device is connected to the internet and switched on. This connection permits the camera to be communicated and commanded appropriately via a PC or smartphone application from anywhere and anytime around the world. For instance, intelligent Onvif YY HD camera uses cloud-links.net / cloudlinks.cn as a backend.

Similar to common consumer-grade monitoring devices this IP camera has a “P2P Cloud” characteristic that is enabled by default. The camera joins to a server of cloud infrastructure and holds the link-up. All approved desktop and smartphone applications can receive a live video stream from the IP camera via the cloud.

From a usability view, this makes it more comfortable for users to communicate with this device because the user doesn't have to be in the same network (LAN) to access the camera.

Furthermore, no port forwarding commands, firewall controls, or DDNS service setup are needed upon the router. Nevertheless, this method has some security implications such as:

Unencrypted communication

If the process of information exchange is not correctly encrypted, anybody who can cut off the relationship can watch all the information that is transferred.

P2P cloud

This characteristic bypasses firewalls and completely enables teleconnections into local or private networks. Presently criminals cannot just hack things that have been deliberately/inadvertently shown to the web (classic "Shodan hacking" or the Mirai approach) but a huge number of things that are displayed via the "P2P Cloud".

A list of related cameras can be regained from the backend by the attackers

Camera IDs have a length of up to seven digits for Cloudlinks. The validity of all device IDs can be checked by the attackers.

Default and weak passwords can be enumerated

Several kinds of packets are sent by the backend only depend on the camera ID. By replaying packets with modified camera IDs, passwords can be methodically investigated by attackers upon all cameras. This permits for forwarding packets to all actual cameras. A vulnerability in the administration of these packets would come up with all cameras to active exploitation [31].
Figure 3.1 demonstrates the components of IP cameras.

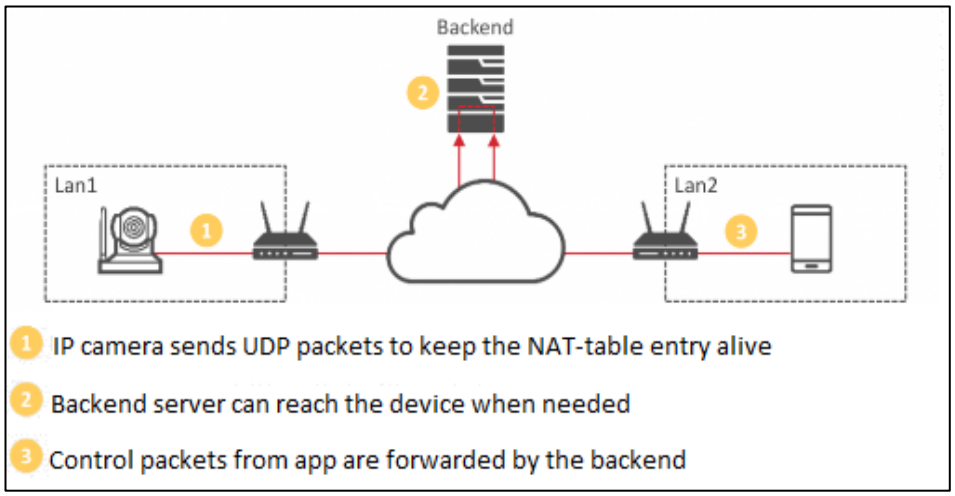

Figure 3.1. IP camera system components

3.6. Security Mechanisms

Security mechanisms include technological methods and procedures designed to enforce security services. To produce a specific service, a system can work either alone or with others. Types of basic security mechanisms are:

3.6.1. Password Policy

Password policy has the ability to help and improve security. It is possible to set password laws like the strength of a password, password expiration frequency and so on via password policies. Also, various rule types can be extended for specific platform applications [32]. The method of authentication remains the same including the use of passwords and usernames. Furthermore, most of the passwords include names with various combinations that an attacker can guess. The strength of the password is the indicator of defense to imagination and brute-force attacks, the number of characters that are used in a password (such as 8 characters) and a combination of numbers and symbols instead only alphabets, these are some steps to measuring the strength of a password [33].

3.6.2. Lockout Mechanism

Account lockout methods are utilized to minimize the risk of assaults via brute-force to achieve login into a system. It is a commonly used method to protect toward online password breaking attempts by domains or any other systems. When several failed authentications try towards an account are recorded in a short time for the purpose of login to the system, the login fields will be "locked" for a specific time and no one, not even the client with the right credentials, will be able to log into the system [34].

3.6.3. Authentication Factors

The factor of authentication is a piece of data and a method utilized for authenticating the identity of a user. Two-factor authentication is an additional security mechanism used to verify that a person who is trying to enter a system. The users cannot enter the system at once after adding their login credentials such as user name and password, also he/she will be needed to provide another piece of information, this additional information based on something one has and something one knows. There is also another security element that offers multi-factor authentication depending on what one is, namely biometric identification [35].

3.6.4. Encryption Algorithms

Throughout network transmission environments, unencrypted information flowing over various channels is vulnerable to various kinds of attacks and may be interpreted, manipulated or modified by someone with exposure to such data. Encryption and decryption methods are used to protect the data and to deter attacks. Encryption is the basic aspect in cryptography where one protects information and data by turning it into an unreadable script. Usually, encryption utilizes a given function or code to convert the information.

The length of the key in some types of encryption methods is must be equal to the length of the readable message, but some other mechanisms may work on much shorter keys compared to the text. This mechanism most widely practiced during exchange data within unprotected channels, like the Internet. Digital signatures can be created via encryption methods to authenticate a document. Encryption methods can be demonstrated in two main categories such as:

Symmetric encryption

This type utilizes a single key for both encryption and decryption. It means that the person who encrypts the messages must send the key to the recipient in order to decrypt the ciphertext. Examples of this type include DES, AES, and Blowfish.

Asymmetric encryption

This method is also known as public-key encryption, utilizes two distinct keys, one key for the encryption procedure identified as the public key and one key for the decryption procedure identified as the private key. Examples of this type include RSA and Digital Signature [36].

4. MATERIALS AND METHODS

This section introduces the hardware and software requirements to work on this thesis, such as penetration testing tools, devices, and application programs. Furthermore, some pen-testing tools used in this thesis are defined in this part briefly.

4.1. Hardware and Software Requirements

This section explains the hardware and software requirements in this study

4.1.1. Hardware

- A computer (with at least 4 GB of RAM, 20 GB of available hard-disk space,1.6 GHz or faster processor)
- Three different IP cameras (Intelligent Onvif YY HD camera, Panoramic IP camera, and Dual-HD camera)
- Wireless adapter that must support monitor mode (Alfa 802.11 n)
- A smartphone (controller)

4.1.2. Software

- Kali Linux as an operating system and pen-testing platform (version 2019.3)
- Some tools for penetration testing such as Wireshark, Nmap, Netdiscover, Ettercap, Arpspoof, Mitmproxy and Driftnet
- Mobile applications which support the cameras (YYP2P, YOSSEE, V380, NetCam)
- Windows applications which support the cameras (CMS Client, V380, Device Client)

4.2. Penetration Testing

A penetration test is a part of ethical hacking [37], sometimes described as a ‘white hat hacking’ or ‘pen test’. It provides a collection of protocols and processes designed to test the systems. At the same time, the foremost purpose of penetration testing in this study is to recognize vulnerabilities in a specific IoT device and organization’s security, also, it can be practiced to detect the threat, risk and how the devices and systems breakdown.

The penetration tests are useful for identifying an organization's weaknesses and testing if a hacker can manipulate them to obtain access to the system since if we discovered the vulnerabilities, we can find ways to protect the security of the system and IoT devices [7].

4.2.1. Penetration Testing Methodologies

The procedure of conducting a pen-test in a system must be decided before checking the vulnerabilities in the devices and systems. The process of penetration testing system contains three steps:

Determining the scope

At the first step of the penetration testing process, it is important to identify the testing range. The criteria of the testing may cover the entire system and devices, or it may be simply targeting the IoT devices that working on the system. Some items should be defined to identify the test scope properly such as the number of targets at the test and test range.

Implementing the penetration testing

Each company ensures that the processes they are implementing for a penetration test are appropriate. This is the second step in the penetration testing process, it includes collecting all the important information to reveal vulnerabilities in security. It is the penetration tester's responsibility to ensure that programs, networks or devices are not exposed to a security gap that may permit unauthorized access to the network, by testing and inspecting the system.

Reporting and presentation of outcomes

After the penetration test has been performed, all data obtained from the procedure of testing will be reviewed by penetration testers. The presentation report includes these parts:

- Demonstration of risks and vulnerabilities found by security tester at the system
- Details about the strengths and weaknesses of the existing security points
- Threats classified as low, medium and high
- Details about the weaknesses of each device

Finally, penetration-testers offer suggestions for fixing detected security gaps and provide detailed information about how to solve the weaknesses discovered in the system [38, 39].

4.2.2. Penetration Testing Tools

To implement the penetration testing process successfully the following software and tools are used:

Kali Linux

Kali Linux is a platform for checking the security threats and penetration testing with professional techniques for detecting, identifying and leveraging any bugs found in the specific system. Implementing an effective research technique prepared with well-defined business goals

and a planned test strategy can result in the network's rigorous penetration check. Kali is a Linux distribution created to concentrate on the task of penetration testing. The previous Kali Linux generation was referred to as BackTrack. On 13 March 2013, the first version (1.0) of Kali was published as a new open-source and powerful platform for penetration testing [40].

Nmap

Network Mapper or Nmap is a network discovery and safety checking tool. It is also found to be helpful by several systems and penetration testers for duties like the inventory of networks, overseeing schedules of service updates, and testing availability of hosts and services. Nmap utilizes raw IP packets in innovative methods to identify the hosts are operating on the network, the name/version of application these hosts provide, determine the type of operating system with its version, which kinds of packet firewalls or filters are already in use, and many other functions.

It was developed to search wide area networks quickly but operates very well with individual hosts. Nmap operates on all main computer operating systems such as Mac OS X, Linux, and Windows [41].

Wireshark

Wireshark is still arguably the best powerful open-source and accessible protocol/packet analyzer in the world. It is generally utilized for debugging, testing, and development of protocols by penetration testers and security experts. Wireshark's main purpose is to collect network activity and show packet information as accurately as possible, allowing experts to interpret network traffic material at a microscopic scale.

It is an interactive packet analyzer that helps the tester to graphically simulate the operation of the network and supports experts in monitoring problems at the network level. Nevertheless, for the purpose of achieving sensitive information in the format of plaintext, Wireshark is often utilized by attackers to inspect data packets [42].

Netdiscover

Netdiscover is another tool in Kali Linux [43], employed by pen-testers, it is a perfect method to discover possible IP addresses for more analysis on the network. This is accomplished by forwarding ARP messages to the specified network. The tester can find all devices on any network type, wireless and wired via running this tool. It also tries to determine the vendor through its MAC address, which could be very useful in identifying security flaws to exploit [44].

Arpspoof

Arpspoof, an element of the dsniff package, is a package with several tools that could be used to implement MITM attacks. The arpspoof is a tool that could be utilized in a switch system to track the network packet stream and observe network traffic.

Generally, if host-A decides to connect with host-B (gateway), an ARP request to obtain the MAC address of host-B will be broadcast by host-A. After that, host-B will respond by giving its MAC address as such an ARP Response packet to the request. Host B is conducting the same operation. Now, host-A will be able to connect with host-B. When an attacker C tries to monitor the traffic of a network between A and B, the ARP answers must be sent to A by telling the B's IP address, and vice versa from B to A. Via the below commands in kali, the attacker will start to sniffing the complete network traffic between A and B will first go through C.

"arpspoof -i [interface_name] -t [target-A] [target-B]"

"arpspoof -i [interface_name] -t [target-B] [target-A]"

To receive and monitor all the packets in the network traffic, the wireless Wi-Fi adapter or network card must support monitor mode with the below commands we can enable monitor mode on the computer running Kali Linux:

“ifconfig [interface_name] down”

“ifconfig [interface_name] mode monitor”

“ifconfig [interface_name] up”

Also, the command “iwconfig” can check if the network card is in the right mode.

Another important point is the IP forwarding must be enabled on the Kali machine by this command [45]:

"echo 1 > /proc / sys/ net / ipv4/ip forward".

Bettercap

Bettercap is a new generation of Ettercap, including advanced sniffing and spoofing features at the network level. Both of them are man-in-the-middle software designed to examine and improve the network security or any machines attached to these networks by pen testers [46].

Driftnet

Driftnet is another tool in Kali used to display pictures taken through a MITM attack. There are other methods of obtaining further valuable information; but, when we concentrate on displaying pictures, this tool may be helpful. If Driftnet not installed on Kali, the command “apt-get install driftnet” will download it.

The command "driftnet -i [interface_name]" will open a new empty console screen. All photos accessed by a target during the MITM attack will show on this screen [47].

4.3. Malicious Software and Tools

4.3.1. Shodan

The search tool of Shodan is programmed to walk straight the Internet network and try to classify and recognize the internet-connected machines. Shodan has the ability to find millions of smart on-line things related to ICSs (industrial control systems). Significant security issues have been raised by the ability to recognize machines that control and supervise significant infrastructure resources [48]. According to previous researches, Shodan is the internet's most dangerous search engine .

4.3.2. Mirai

The Mirai malware and its versions are a wake-up alarm to the enterprise to improve the security of IoT machines or to reveal the digital infrastructure for potentially destructive assaults on DDoS attacks [49].

The despiteful Mirai program is looking for machines with accessible IP addresses, therefore it will identify the hardware kind. To achieve the log in detail information it uses a checklist that contains default credentials usually utilized by varying companies in several IoT gadgets. When it breaks the system, a source code that makes the IoT things a slave will install via Mirai. Upon deploying this software on a huge amount of IoT machines, the hacker has the ability to demand the established software and require that all of these machines send a request to a specific web page and makes it down. This could result in a DDoS assault that renders the particular web page inaccessible [50].

4.3.3. BrickerBot

In the same manner as Mirai, BrickerBot detects insecure machines on the Internet, but it tries to destroy and bricks them rather than inserting malware. The bricked machines are not going to be able to operate and hence are no longer a danger to be a member of a botnet governed by such as Mirai [51].

5. EXPERIMENTS

This chapter will explain all the properties of the Device used in this study. Moreover, it will also provide an explanation about all test procedures which have been done for this experiment to investigate the vulnerabilities of IP cameras and explain their impacts on the security and privacy of users via utilizing the Kali Linux penetration testing platform and its tools.

5.1. Setting up the Environment

First of all, Kali Linux operating system is installed on VMware work station 15, Kali is the famous and number one OS that used to penetration testing and it contains various tools that help us during the process of penetration testing, after that, all the devices (laptop, mobile phone, and IP cameras) are connected to the internet on the same LAN.

Next, all the mobile applications that support the cameras are installed on the mobile phone (such as Yoosee, YYP2P, V380, and NetCam), according to the user manual, these apps are used to control the IP cameras through a smartphone. Finally, windows applications, such as CMS Client, V380, and Device Client, that support the cameras are installed on the laptop to control the IP camera through the computer.

5.2. Intelligent Onvif YY HD Camera

Table 5.1 reflects detail information about the first device used in the experiment part of this thesis.

Table 5.1. Information about the intelligent Onvif YY HD camera

Device Name	Intelligent Onvif YY HD camera	**Device's Picture**
Device ID	8861084	
User Name	None	
Device Password	123	
MAC Address	54:C9:DF:4B:3F:F7	
PTZ Control	Yes	
LAN IP (router)	192.168.1.101	
Mobile App	YYP2P – YOOSEE	
Windows App	CMS Client	

5.2.1. Information Gathering

This section deals with collecting different types of data by practicing Kali Linux tools, for instance, Netdiscover and Nmap. In the first step we need to know how many devices are working on the network, the below command in Kali Linux displays all devices that work on the network:

root@kali-Test:~# netdiscover

Figure 5.1 shows all the devices that work on the LAN network with detailed information about IP address, MAC address, and hostname.

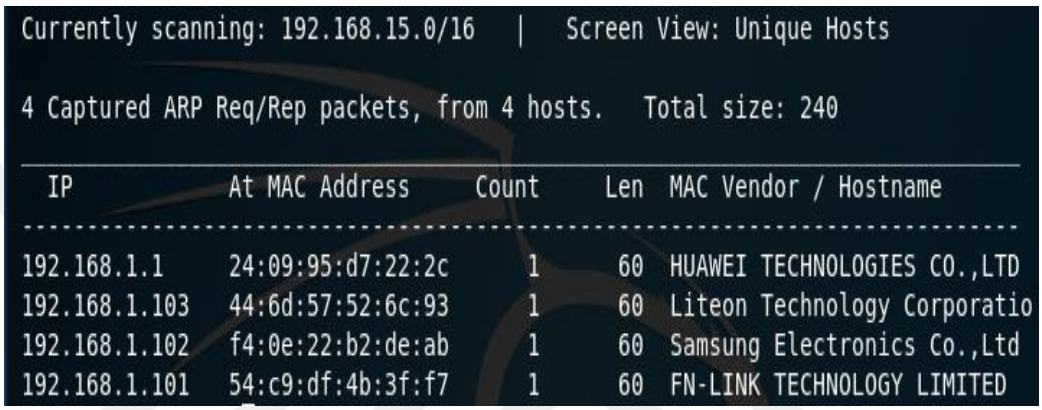

Figure 5.1. Netdiscover scan inside the LAN

In order to check the open ports in this camera, the Nmap scan is needed to inspect all the ports to test if it is open or not from 1 to 65535. Figure 5.2 shows the result of the Nmap scan with the command:

root@kali-Test:~# nmap -p1-65535 192.168.1.101

Figure 5.2. Nmap scan to find open ports

From the figure, we can observe a TCP open port 554 hosting an RTSP (real-time streaming protocol) service.

To discover additionally point by point data about the camera and its working framework, one more scan of Nmap with the alternative -O is required. Figure 5.3 clarifies the camera running Linux kernel 2.6.32 operating system:

root@kali-Test:~# nmap -O 192.168.1.101

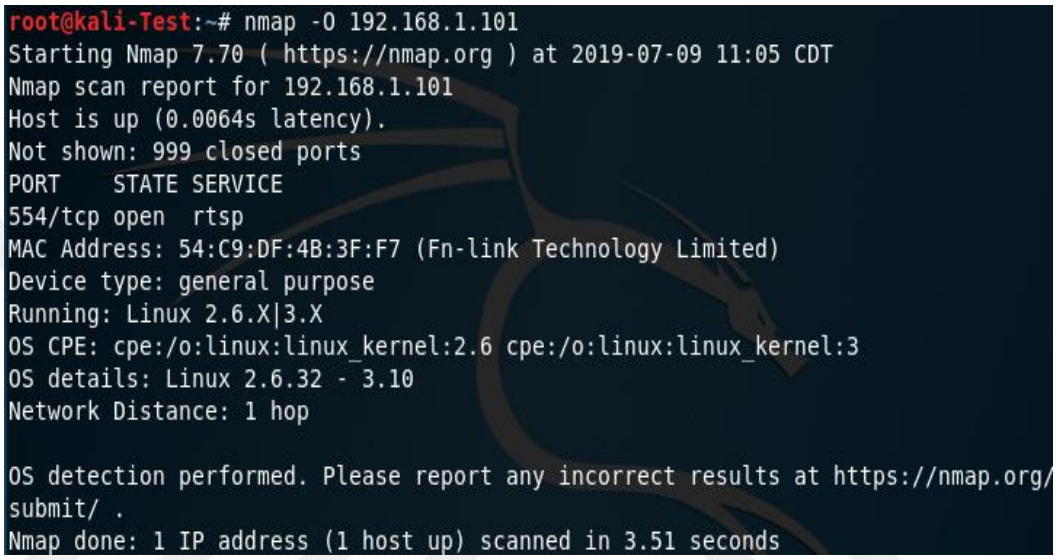

Figure 5.3. Nmap scan to indicate operating system type

5.2.2. Device ID and its Password

The feature of "P2P Cloud" works with the device ID and its password. On the back of the camera, there is an ID number and a password (e.g. ID: 8861084, password: 123). In the supported app (e.g. YYP2P) there is an option to "Add online device" that permits the user to join the camera. Regrettably, the camera ID and default password do not seem so safe.

Except if the user modifies the password to a strong one, anyone can log in and communicate with the device by "examining" diverse cloud IDs because ID and Password of the camera is the only credential to enter the device.

The password policy of the YYP2P mobile application is very weak since it has not any password policy to set a password for the device. If the user sets a password with a length of only one character it will accept it. The below figure shows the screenshot of back of the camera and option to "add online device" in the YYP2P application.

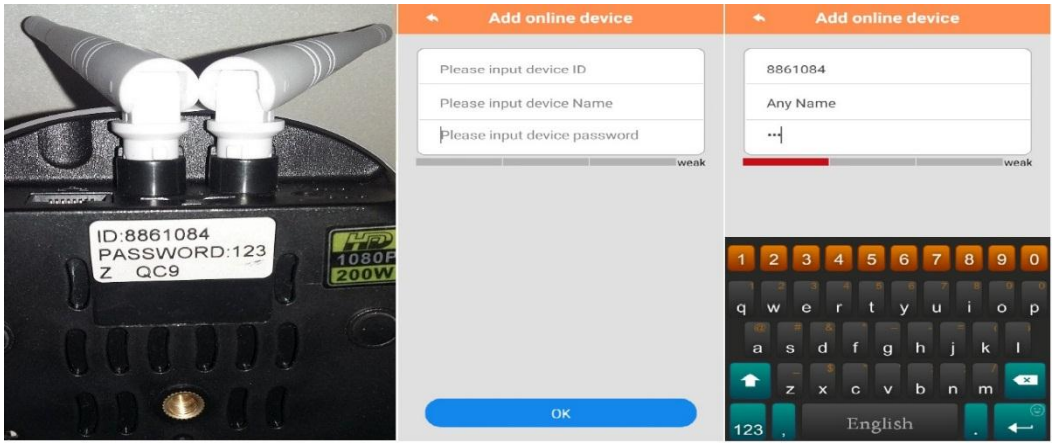

Figure 5.4. Add online device option in YYP2P app

5.2.3. Brute Forceable RTSP URL to Stream Video

The findings of the Nmap test show that an RTSP service working on port 554 in the information-gathering portion. RTSP is identical to HTTP because it has commands for control and URLs. RTSP streaming should create a URL that looks like "rtsp:/IPaddress/ video". RTSP authentication works correspondingly to the authentication of HTTP. A basic strategy to discover the URL will be to simply URL brute force for basic stream names. Nmap performs the URL brute force attack to discover a rundown of ordinarily known RTSP URL's and tests in the event that they operate with the camera, the below figure indicates the result of Nmap scan with the command:

root@kali-Test:~# nmap –script rtsp-url-brute –p [port] [device_IP]

```
root@kali-Test:~# nmap -p 554 --script rtsp-url-brute 192.168.1.101
Starting Nmap 7.70 ( https://nmap.org ) at 2019-07-21 21:17 CDT
Nmap scan report for 192.168.1.101
Host is up (0.0072s latency).
PORT    STATE SERVICE
554/tcp open  rtsp
| rtsp-url-brute:
|   discovered:
|     rtsp://192.168.1.101/onvif1
|     rtsp://192.168.1.101/onvif2
|     rtsp://192.168.1.101/videoinput_1:0/h264_1/onvif.stm
|_    rtsp://192.168.1.101/user=admin_password=tlJwpbo6_channel=1_stream=0.sdp?real_stream
MAC Address: 54:C9:DF:4B:3F:F7 (Fn-link Technology Limited)
Nmap done: 1 IP address (1 host up) scanned in 13.32 seconds
```

Figure 5.5. URL brute force to discover possible RTSP URL

Also, through attaching the RTSP URL to the network stream URL page in the VLC media player, the live video stream of the camera can be achieved. The convenient RTSP URLs that are working with this camera are both “rtsp://IPaddress/onvif1” for HD quality and “rtsp://IPaddress/onvif2” for SD quality, without any authentication just with a simple copy and paste the beforementioned link to the VLC media player, it shows a live video of the camera. Anybody having access to the network can watch the camera with VLC and “rtsp://192.168.1.101/onvif1” or “rtsp://192.168.1.101/onvif2”. As explained in Figure 5.6, the RTSP URL added to VLC after that live video stream obtained in Figure 5.7.

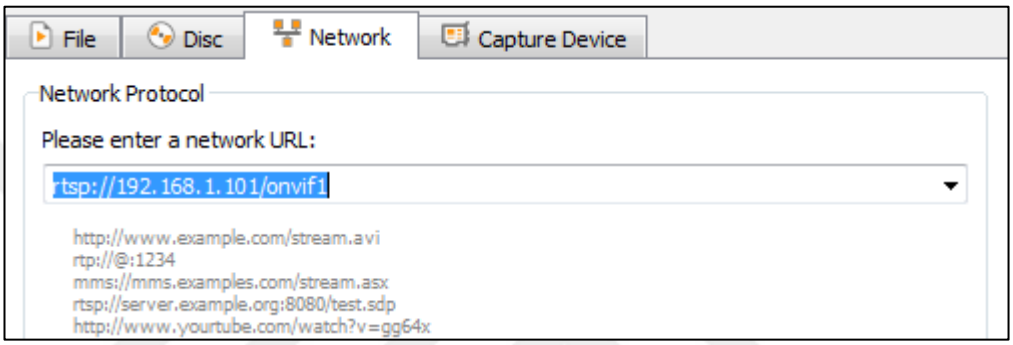

Figure 5.6. Network stream in VLC

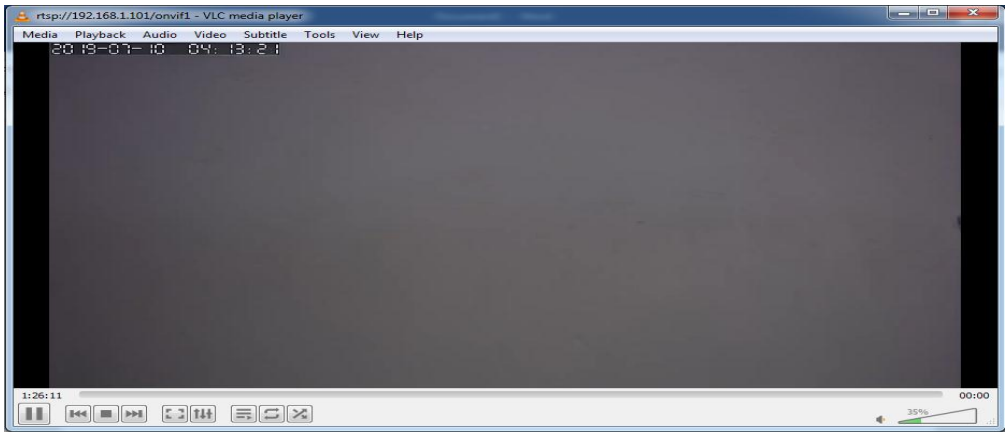

Figure 5.7. Video stream obtained from right RTSP URL

5.2.4. Inspecting Network Traffics

By utilizing Kali tools of network packet capture such as Mitmproxy and Wireshark, it is possible to observe and monitor all network-level packet information. As described in section 5.1 the mobile device, as well as the cameras, are connected to the internet on the same LAN network, so Wireshark and Mitmproxy can be used to monitor all LAN packets, for accomplishing this task. A simulated MITM attack implemented by the following steps:

First of all, we must enable IP forwarding by changing the ip_forward variable from 0 to 1, if the value was changed from 0 to 1, it means that this feature is enabled:

root@kali-Test:~# echo 1 > /proc/sys/net/ipv4/ip_forward

Now, with arpspoof we can implement MITM attack, by writing these two lines of code in Kali Linux Terminal:

root@kali-Test:~# arpspoof -i wlan2 -t 192.168.1.1 192.168.1.101

root@kali-Test:~# arpspoof -i wlan2 -t 192.168.1.101 192.168.1.1

During the capturing of network data, we can note that all the pieces of information were transferred as plaintext. This indicates that the data and information which was transferred among the mobile app (such as Yoosee or YYP2P) and camera, the mobile application and the server of IP camera, camera and IP camera server are can be read in plaintext. The mobile apps sent sensitive data, for instance, login info and enrollment of the device are sent in plaintext which can command to jeopardize all the cameras connected to that account.

Transferring sensitive information in plaintext does not just endanger the security of the camera but also the LAN security in which the device is present. In the first steps of registering the camera, the Yoosee mobile app pushes the SSID (service set identifier) of Wi-Fi and its password. These sensitive data were sent in readable text. If an attacker makes an MITM attack and observes the network traffic they would be ready to enter the Wi-Fi network and sniff all additional information. The data demonstrated was achieved utilizing the Wireshark filter "Follow UDP stream", which picks all the UDP packets of the appropriate packet selected and dumps all of its data. The data captured by Wireshark shown in Figure 5.8. The red enclosed rectangle shapes indicate the Wi-Fi SSID and its password.

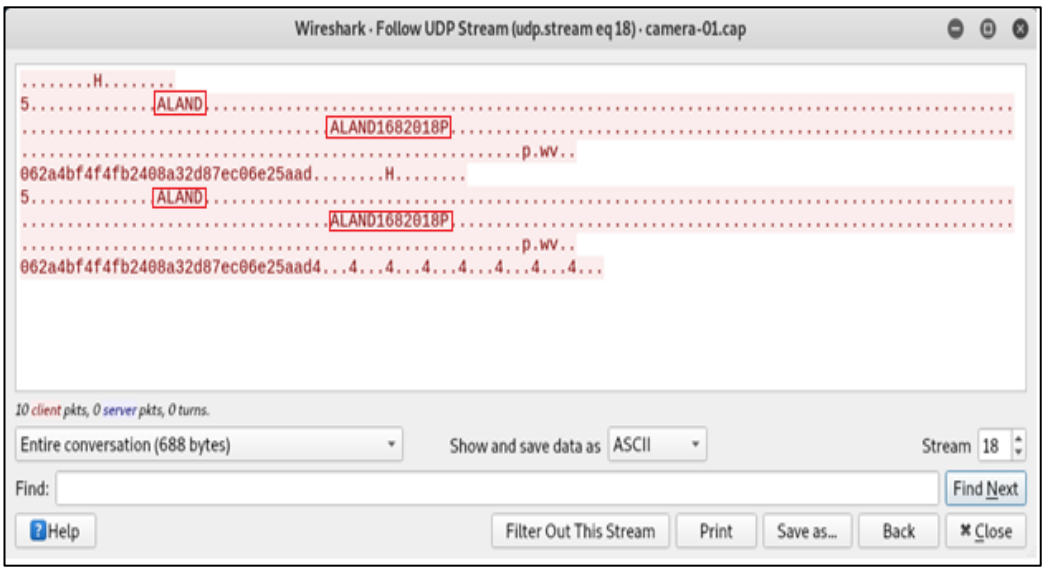

Figure 5.8. Wi-Fi credentials captured by Wireshark in plaintext

Packets recorded with Mitmproxy and Wireshark included some sensitive information in plaintext as pointed in Figure 5.9 and Figure 5.10. This information may contain sensitive data such as login details (username and password). The password login is only protected by an MD5 hash algorithm which can be decoded from (5416d7cd6ef195a0f7622a9c56b55e84) to the original password (1q2w3e4r) and User_ID=031573046 via using online hash decryption platforms.

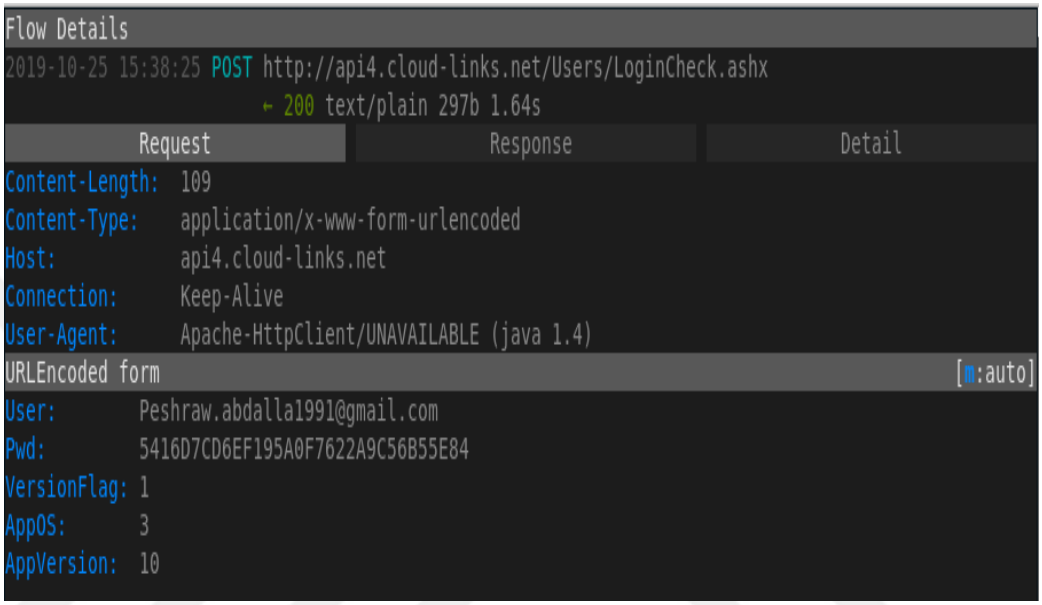

Figure 5.9. Login credentials captured by Mitmproxy

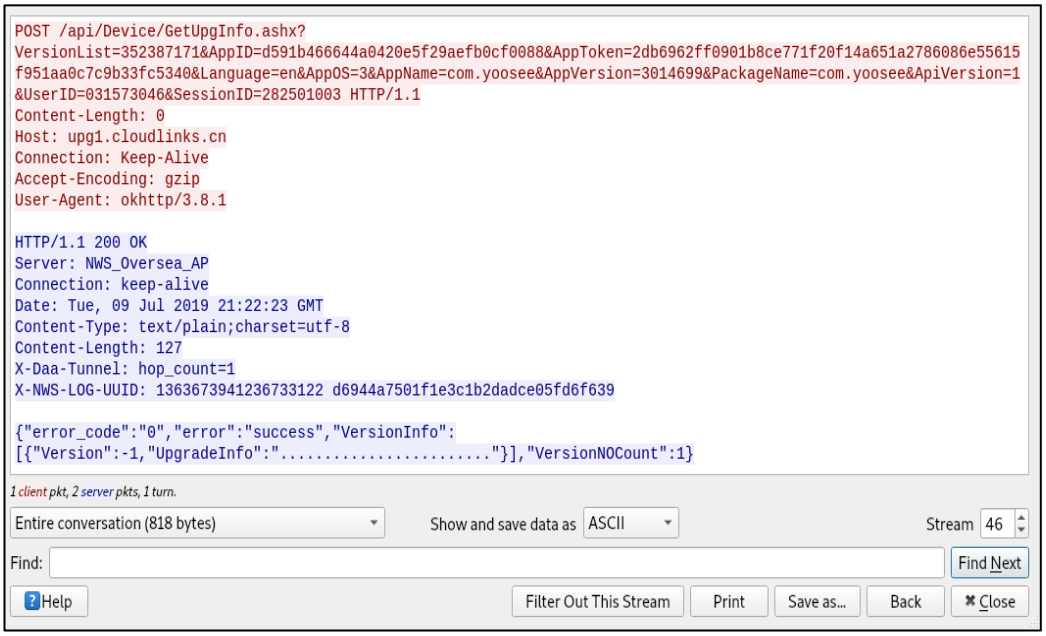

Figure 5.10. User_ID captured by Wireshark

A mobile phone number is another sensitive data sent by the mobile application to the server. As shown in Figure 5.11, the phone number was captured by Mitmproxy when the mobile number was added to the account.

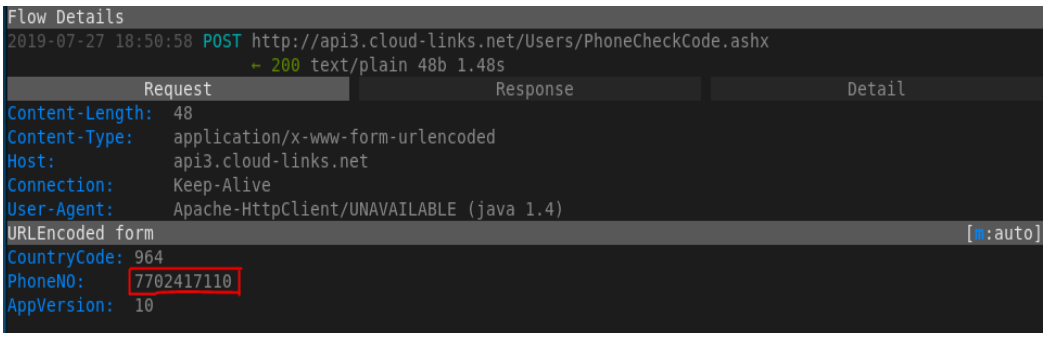

Figure 5.11. Phone number captured by Mitmproxy

CMS Client application sends all data in plaintext to the server. The below figure shows HTTP requests from the application to the server that contains some sensitive data such as SharerName and DeviceID.

Figure 5.12. CMS Client application sends Device ID in plaintext

A further finding by Bettercap is that every media streaming servers of this IP camera are in the form of (p#p#.cloudlinks.cn and p#p#.cloud-links.net) as reflected in Figure 5.13. By replacing the "#" symbol into an integer between (1 to 10) in the (p#p#.cloudlinks.cn and p#p#.cloud-links.net) real servers of the device can be found, for instance, (p2p1.cloudlinks.cn) is one of the real servers of this camera.

The aforementioned information is so helpful for criminals since all the videos from all the cameras are included in this server stream. Several IP cameras presumably use the same servers, which would display a specific case of breakdown of all the company's cameras if the IP camera's servers were unsafe.

Figure 5.13. Media streaming servers of the IP camera

5.2.5. DE-Authentication Attack

This attack sends de-authentication packets to the victim which is directly connected to a specific access point in order to disable a Wi-Fi security camera from a network. The below command can find the SSID of the victim's Wi-Fi and collect detail information, as shown in Figure 5.14.

root@kali-Test:~# airodump-ng wlan0

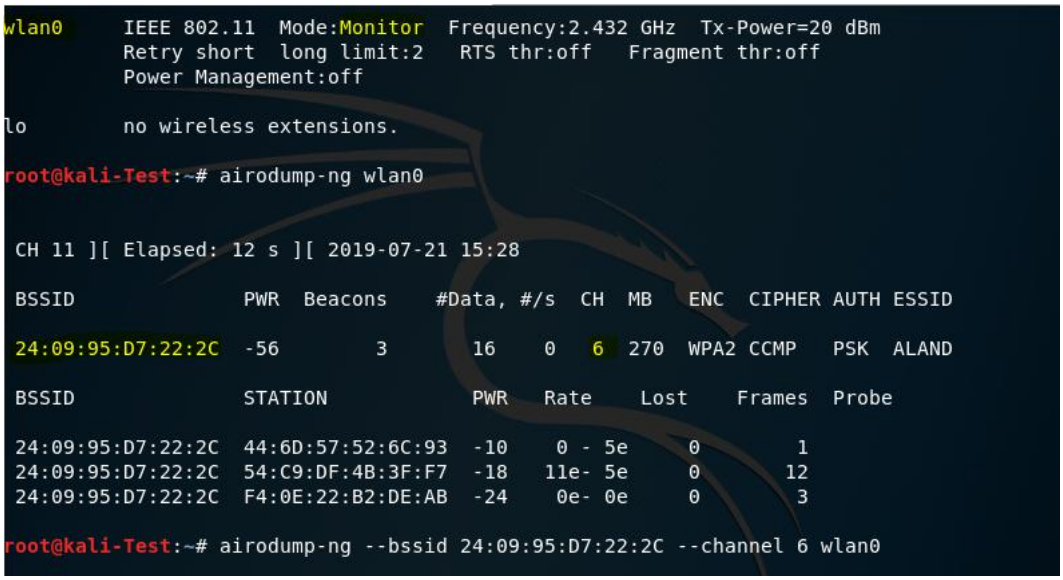

Figure 5.14. The airodump-ng command to find the target's channel

As we can see from Figure 5.14 the target works on channel 6, which this information is beneficial for the attacker. Also, we can see the BSSID of the target, by using these two pieces of information we can write another command in order to find all the devices that are working in this network and gain detail information about them via:

root@kali-Test:~# airodump-ng --bssid 24:09:95:D7:22:2C --channel 6 wlan0

Figure 5.15 shows the result of the command. Our target IP camera can be recognized by its mac address (54:C9:DF:4B:3F:F7), furthermore, we can observe the network mac address that is (24:09:95:D7:22:2C).

Figure 5.15. The airodump-ng –bssid command in Kali terminal

A de-authentication attack is shown in Figure 5.16. After running the following command, the IP camera stopped to send the live video for a specific time. In this attack, the stop duration depends on the number of de-authentication packets being sent. Once the attack is over, the camera will go back to its normal situation and it starts to send the live video stream as demonstrated in Figure 5.17.

root@kali-Test:~# aireplay-ng --deauth 1000000 -a 24:09:95:D7:22:2C -c 54:C9:DF:4B:3F:F7 wlan0

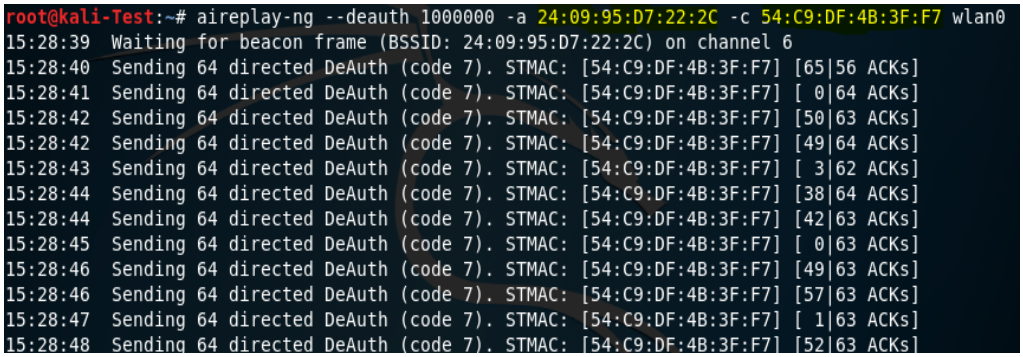

Figure 5.16. De-Authentication attack against the device

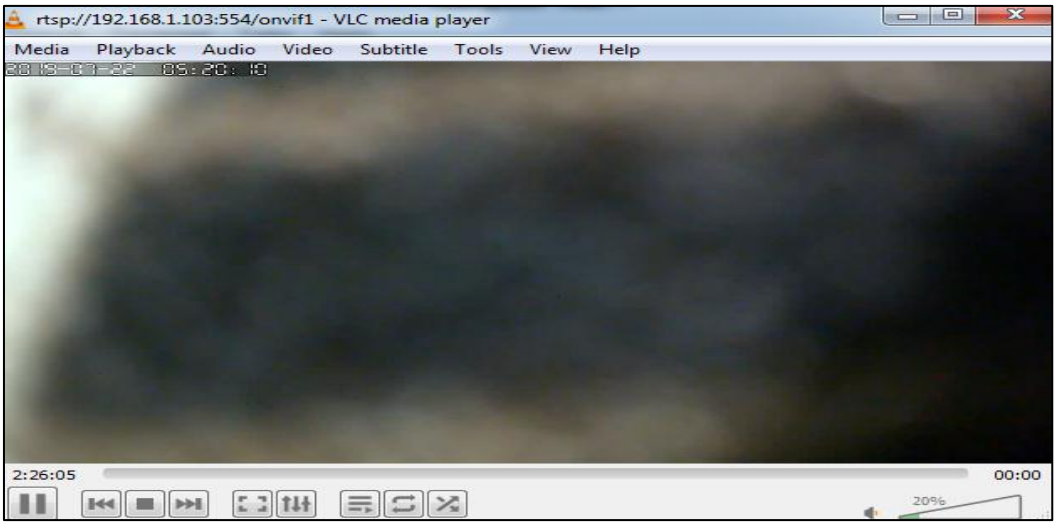

Figure 5.17. The IP camera stopped to send the live video stream after the attack

5.2.6. Driftnet Tool

This tool in Kali Linux is used to retrieve a picture or video from the target's device. This tool usually works with an HTTP request. The command opens the driftnet tool on wlan2:

root@kali-Test:~# driftnet -i wlan2

The result shows that driftnet cannot retrieve the picture or video in this case since only parts of the Web UI itself could be retrieved with this tool.

5.2.7. Inspecting IP Camera Account Credentials

When Inspecting the IP camera account in the mobile application, the credentials are kept in plaintext. Figure 5.18 shows login details of the YOOSEE account that used to control the IP camera with a mobile device that uses an android operating system. Detailed and sensitive information can be found via this directory pass:

/data/data/com.yoosee/shared_prefs/gwell.xml

As can be seen from the figure, Email ID can be found under <string name=”recentName_emailorphone”> that is:

Peshraw.abdalla1991@gmail.com

We can also find the password of account after <string name= ”recentPass_emailorphone”> that is: **1q2w3e4r**

Also, The Wi-Fi name and its password can be found under <string name=” WIFI_PWDALAND”> that is:

WIFI Name = **ALAND** and Password = **ALAND1682018P**

The device ID can be found in <int name=”**8861084**:21.1.0.99”>

Figure 5.18. Login detail and Wi-Fi credentials stored in plaintext at YOOSEE app

Furthermore, when inspecting the IP camera account with another application YYP2P, the results of this mobile application seems to have the same vulnerabilities as the YOOSEE application, such as credentials being stored in plaintext. Figure 5.19 shows the login details for the YYP2P account. We can see the detailed information at:

/data/data/com.yyp2p2/shared_prefs/gwell.xml

We can find the Email ID after <string name=”recentName_email”> that is:

Peshraw.abdalla1991@gmail.com

We can find the password of account after <string name=”recentPass_email”> that is:

1q2w3e4r

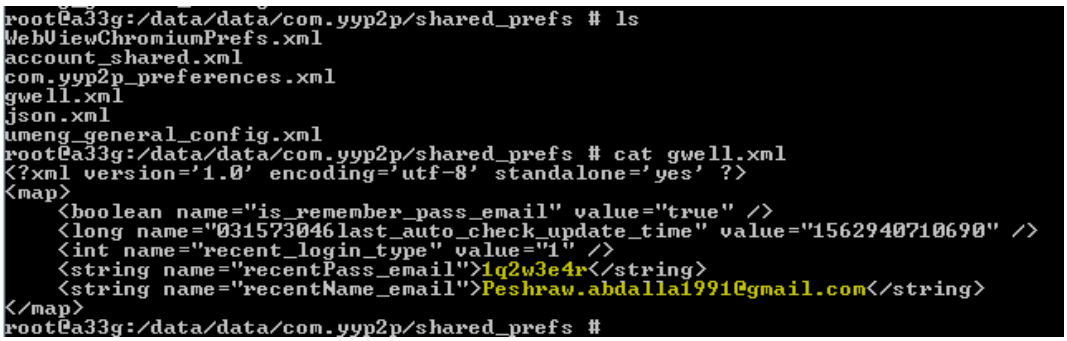

Figure 5.19. Login detail stored in plaintext at YYP2P app

5.3. Panoramic IP Camera

Table 5.2 explains detailed information about the second device in this work by the name panoramic IP camera.

Table 5.2. Information about the panoramic IP camera

Device name	Panoramic IP Camera	**Device's Picture**
Device id	27319433	
Username	Admin	
Device password	None	
MAC Address	14:6B:9C:2F:D1:93	
PTZ control	No	
LAN IP	192.168.1.104	
Mobile App	V380	
Windows App	V380	

5.3.1. Information Gathering

This part deals with collecting different types of information by using Kali Linux tools, the first step is to know how many devices are working on the network via:

root@kali-Test:~# netdiscover

Figure 5.20 shows all the devices that work on this network with detailed information about IP address, MAC address, and hostname.

```
Currently scanning: 172.24.95.0/16   |   Screen View: Unique Hosts

954 Captured ARP Req/Rep packets, from 7 hosts.   Total size: 40086

  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------------
192.168.1.59    44:6d:57:52:6c:93    374   15708  Liteon Technology Corporatio
0.0.0.0         cc:af:78:9b:24:f7      3     126  Hon Hai Precision Ind. Co.,L
192.168.1.162   cc:af:78:9b:24:f7     20     840  Hon Hai Precision Ind. Co.,L
192.168.1.164   54:35:30:9f:4c:13      1      42  Hon Hai Precision Ind. Co.,L
192.168.1.1     0c:b6:d2:89:d8:91    554   23268  D-Link International
192.168.1.104   14:6b:9c:2f:d1:3f      1      42  SHENZHEN BILIAN ELECTRONIC C
192.168.1.11    44:6d:57:52:6c:93      1      60  Liteon Technology Corporatio
```

Figure 5.20. Netdiscover scan inside the LAN

In order to check the open ports in this camera, the Nmap scan will inspect all the ports to check whether it is open or not from 1 to 65535. Figure 5.21 shows the result of the Nmap scan with the command:

root@kali-Test:~# nmap -p1-65535 192.168.1.104

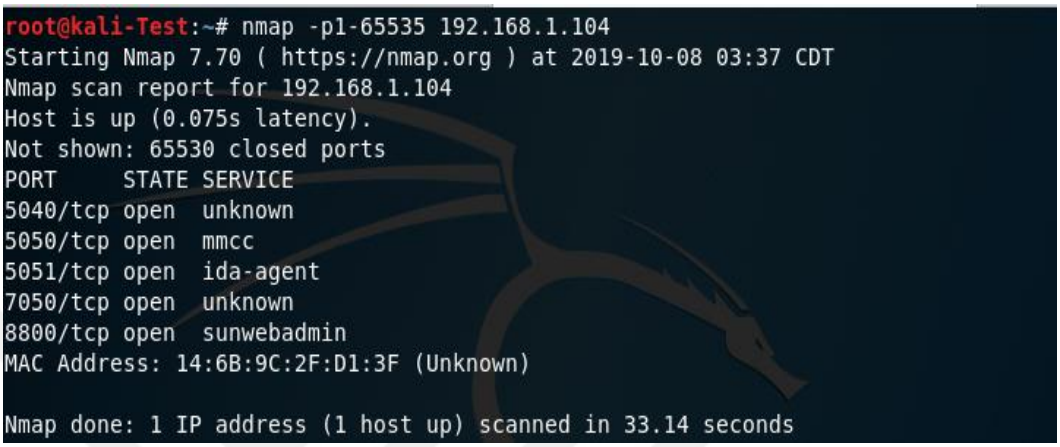

Figure 5.21. Nmap scan to find open ports

The scanned ports with Nmap indicate that there are five services open on TCP ports, such as ports 5040, 5050, 5051, 7050 and 8800.

5.3.2. Device ID and its Password

On the back of this camera, three pieces of information were written (ID, username, and password). There is an ID number with the length of up to eight digits (ID: 27319433) and default username (admin), its password seems to be more secure compared to the first camera since this device doesn't have default password and it must be set by the user during the process of configuration.

The password policy of the V380 mobile application obligates the user to set a password with a length of at least six characters and the combination with numbers and alphabets.

Also, in the supported app (e.g. V380) there is an option to "Add online device" that permits the user to join the camera. Fortunately, the password seems safe despite camera ID and default username do not seem so safe. Except the user has set a strong password and modified the username to another one, also anyone cannot log in and communicate easily with this device by "examining" diverse cloud IDs because ID and Password of the camera are not the only credentials to enter to this device, he/she needs device username also. The below figure shows the back of the camera and option to "add device" in the V380 mobile application.

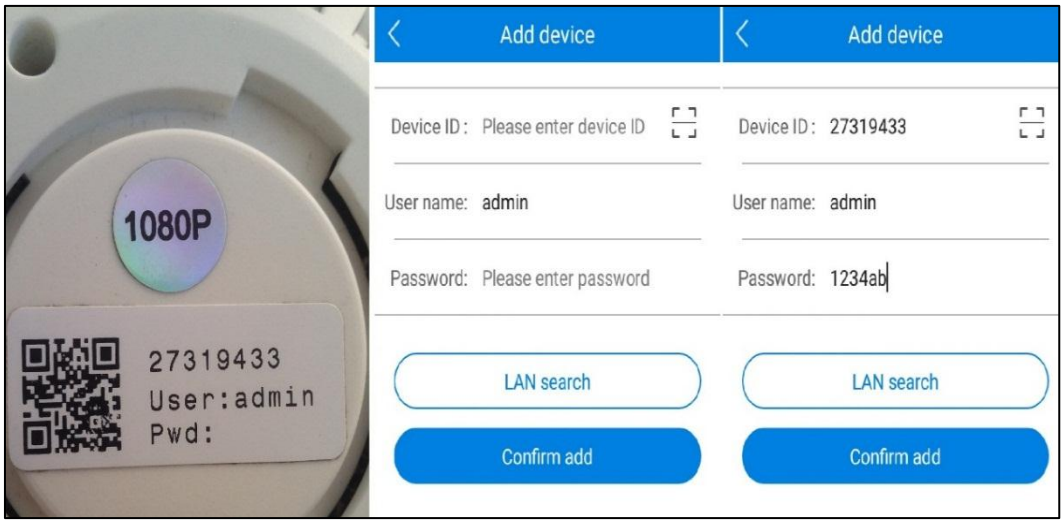

Figure 5.22. Add online device option in V380 app

5.3.3. Brute Forceable RTSP URL to Stream Video

The result of the Nmap test is to find valid RTSP URL shows that the RTSP protocol is closed in this camera, it indicates the device is not possible to RTSP URL brute force attack. The below figure indicated the result of Nmap scan with the command:

root@kali-Test:~# nmap –script rtsp-url-brute –p [port] [device_IP]

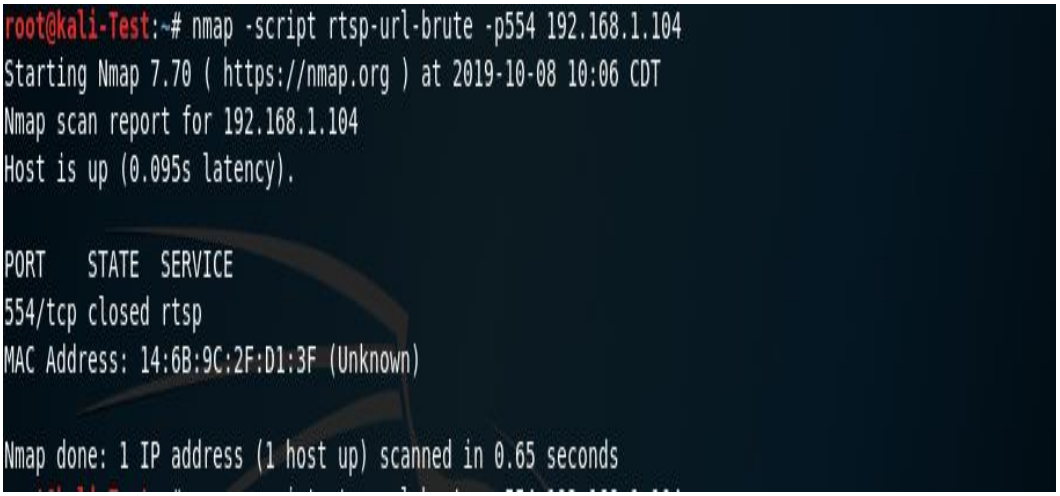

Figure 5.23. Nmap scan to find possible RTSP URL

5.3.4. Inspecting Network Traffics

A simulated man in the middle attack was done by the procedures that are shown in section 5.2.4. Also, by utilizing Kali tools of network packet capture such as Mitmproxy and Wireshark, it is possible to observe and monitor all network-level packet information.

During capturing the network data, we can note that all pieces of information were transferred in plaintext, and in some cases encoded with bad keys such as V380. Also, the Windows application encoded the device password with base64 which can be easily decoded to the original text. This indicates that the data and information which was transferred among the mobile app (e.g. V380) and camera, the mobile application and the server of IP camera, camera, and IP camera server can be read in plaintext. The mobile apps sent sensitive data, for instance, login info and enrollment of the device in plaintext which can be used to jeopardize all the cameras connected to that account.

The V380 mobile app pushes the SSID (service set identifier) of Wi-Fi and its password in the first steps of registering the camera. These sensitive data were sent in readable text. If an attacker (Wi-Fi Sniffer) makes an MITM attack to observe the Wi-Fi credentials to enter the LAN network he/she can sniff all additional information. The data demonstrated was achieved utilizing the Wireshark filter "Follow TCP stream (tcp.stream eq 16)" during the recording network traffic. As shown in Figure 5.24 the red enclosed rectangle shapes reflect the Wi-Fi "SSID = DSL-224-d890" and its "password = 53425331".

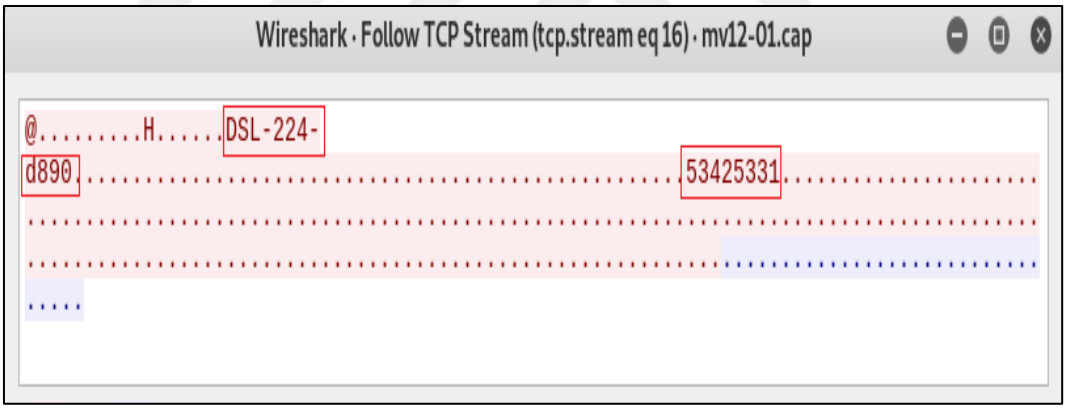

Figure 5.24. Wi-Fi credentials were captured by Wireshark in plaintext

Packets documented by (Mitmproxy and Wireshark) included information in plaintext as shown in Figure 5.25, Figure 5.26 and Figure 5.27. This information may contain sensitive data such as login details "user login = peshraw.abdalla1991@gmail.com, device information "username = admin", "device id = 27319433" and "device password = 1234ab".

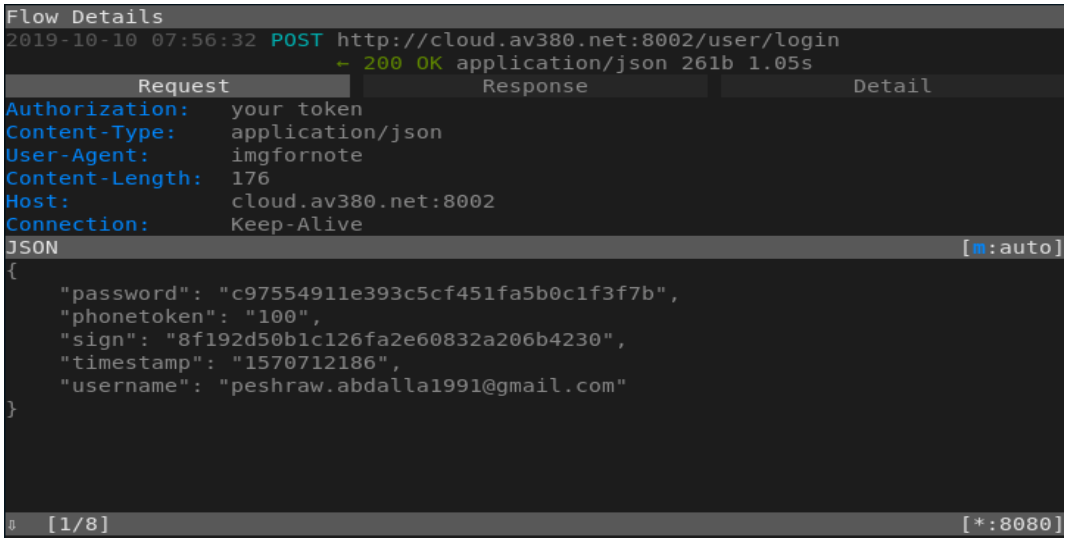

Figure 5.25. Login credentials captured by Mitmproxy

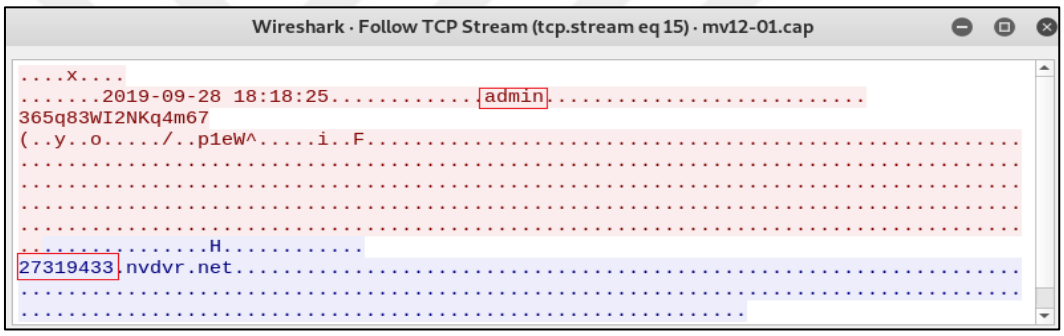

Figure 5.26. Device ID and its username captured by Wireshark

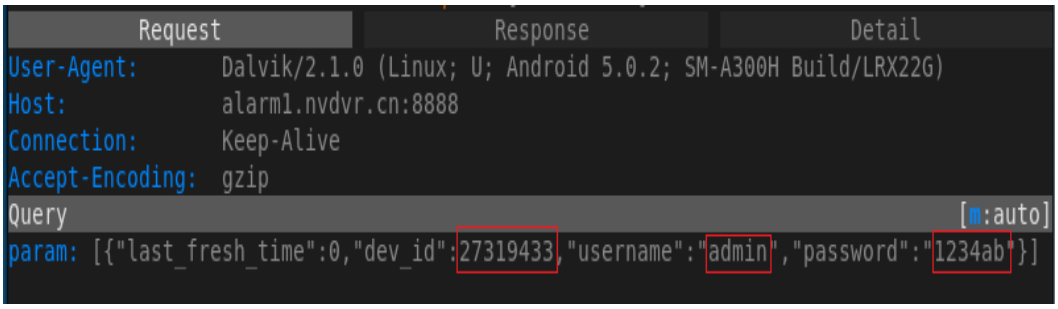

Figure 5.27. Device ID, device password and its username captured by Mitmproxy

V380 windows application sends all data in plaintext to the server, as shown in Figure 5.28. Also, Figure 5.29 shows HTTP request from the application to the server that contains some sensitive data such as UserName, UserID, Device ID and Device account, just device password (MTIzNGFi) encoded over Base64 which can be easily converted to the original password that is (1234ab).

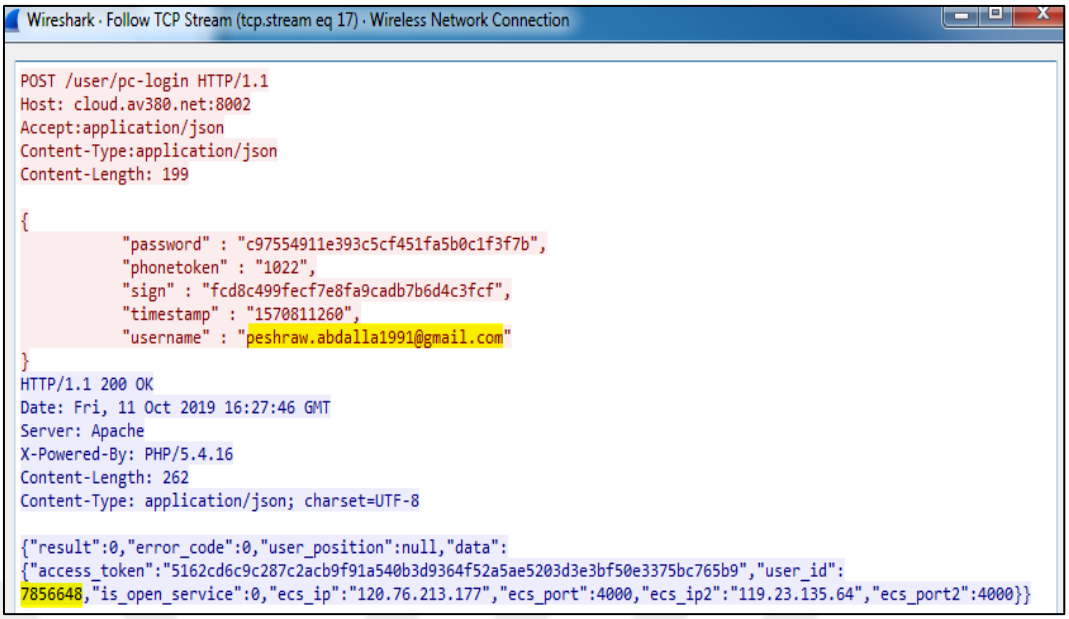

Figure 5.28. V380 windows application sent login username in plaintext

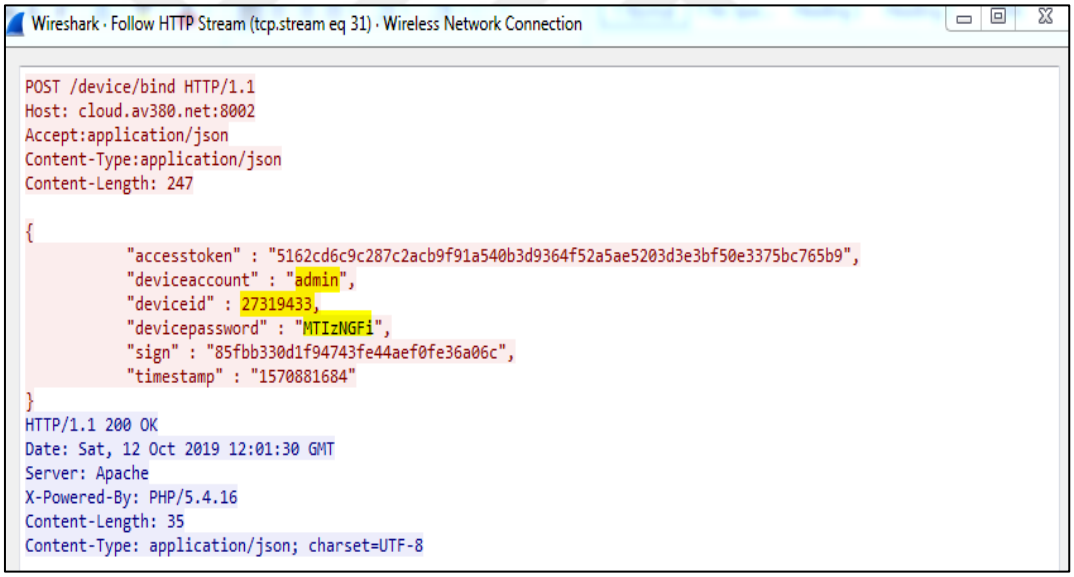

Figure 5.29. Device ID and username captured in plaintext and its password encoded with base64

5.3.5. DE-Authentication Attack

De-authentication attack was successfully done against this device like the first camera. Results show that this device is vulnerable to this attack. The de-authentication process was done by the steps that are shown in section 5.2.5. This attack disables the Wi-Fi security camera from the network and stops the camera to send the live video stream to the user for a period of time that the attackers dictate.

5.3.6. Inspecting IP Camera Account Credentials

When inspecting the IP camera account (V380), it is observed that the login username has been kept in plaintext. All the directories that are shown in Figure 5.30 are checked in order to find login credentials. Username login stored in plaintext but login password cannot be found as plaintext.

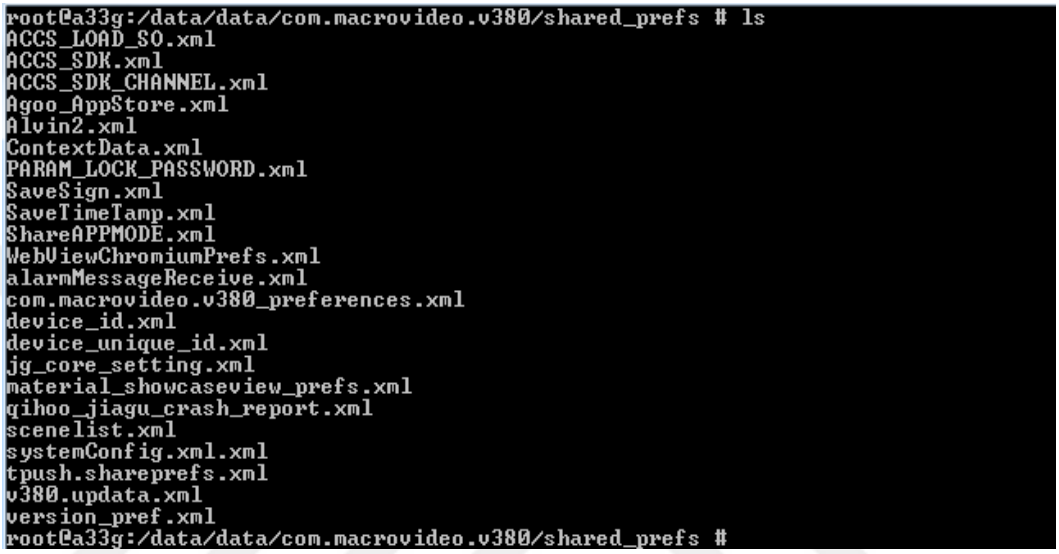

Figure 5.30. All the directories of the V380 app

Figure 5.31 shows the e-mail login of the V380 account that was used to control the IP camera. This sensitive information can be found at:

/data/data/com.macrovideo.v380/shared_prefs/SaveSign.xml

Email ID can be found under <string name=”save_username”> that is:

peshraw.abdalla1991@gmail.com

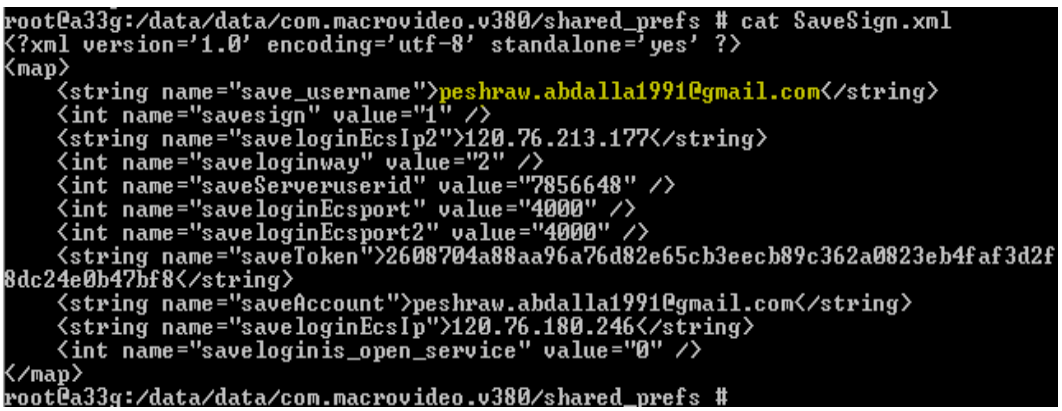

Figure 5.31. Login username stored in plaintext at V380 app

As shown in Figure 5.32 the device ID was found under the below directory in plaintext:

/data/data/com.macrovideo.v380/shared_prefs/device_id.xml

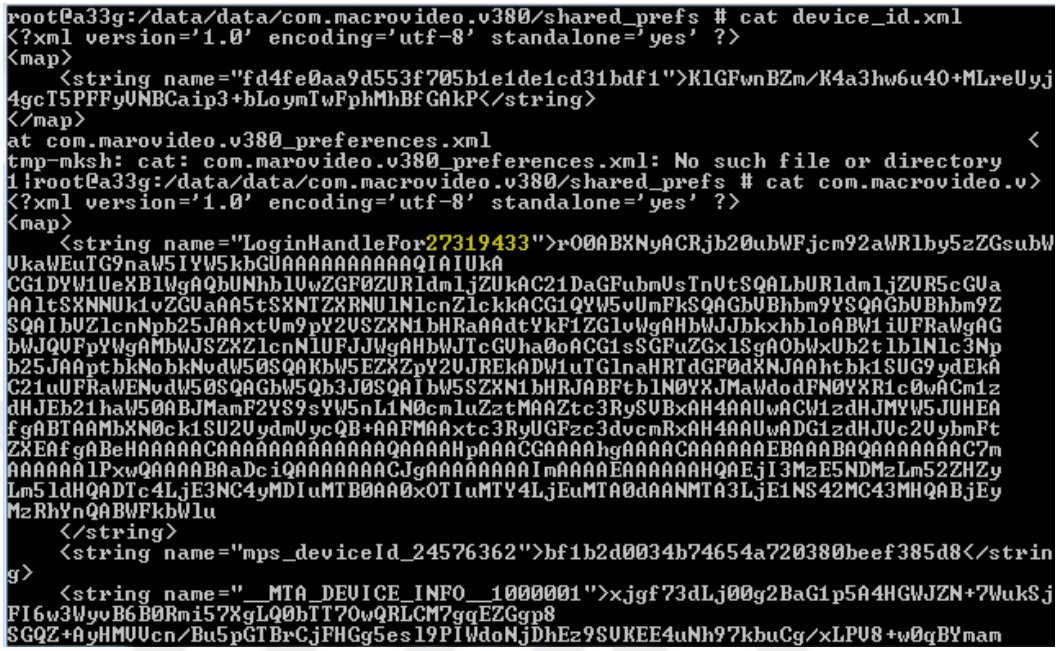

Figure 5.32. Device ID stored in plaintext at V380 app

5.4. Dual-HD Camera

Table 5.3 provides detail information about the last device that is used in this work.

Table 5.3. Information about the Dual-HD camera

Device Name	Intelligent Onvif YY HD camera	**Device's Picture**
Device ID	8861084	
User Name	None	
Device Password	123	
MAC Address	54:C9:DF:4B:3F:F7	
PTZ Control	Yes	
LAN IP (router)	192.168.1.101	
Mobile App	YYP2P – YOOSEE	
Windows App	CMS Client	

5.4.1. Information Gathering

This part deals with collecting different types of information by using Kali Linux tools. In the first step, to find the answer of how many devices are working on the network, the below command in Kali Linux is used.

root@kali-Test:~# netdiscover

Figure 5.33 shows all the devices that work on the LAN network with detailed information about IP address, MAC address, and hostname.

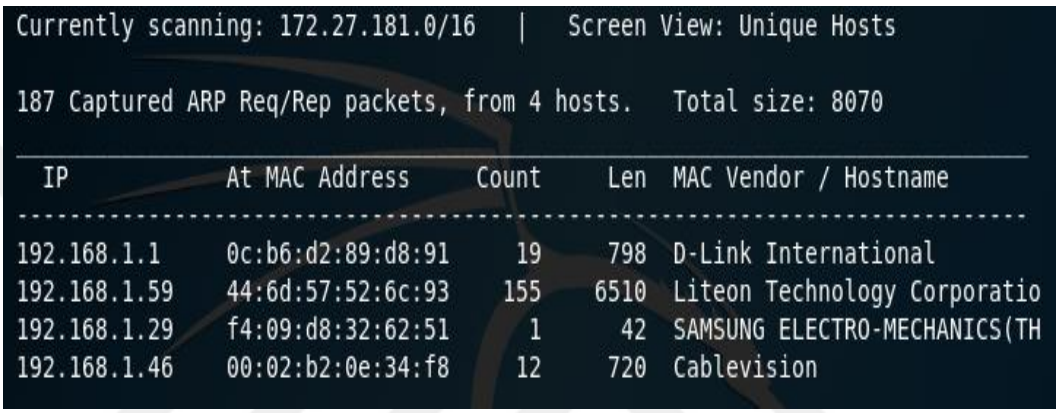

Figure 5.33. Netdiscover scan inside the LAN

In order to check the open ports in this camera, the Nmap scan inspects all the ports to check if is it open or not from 1 to 65535. Figure 5.34 shows the result of the Nmap scan with the command:

root@kali-Test:~# nmap -p1-65535 192.168.1.46

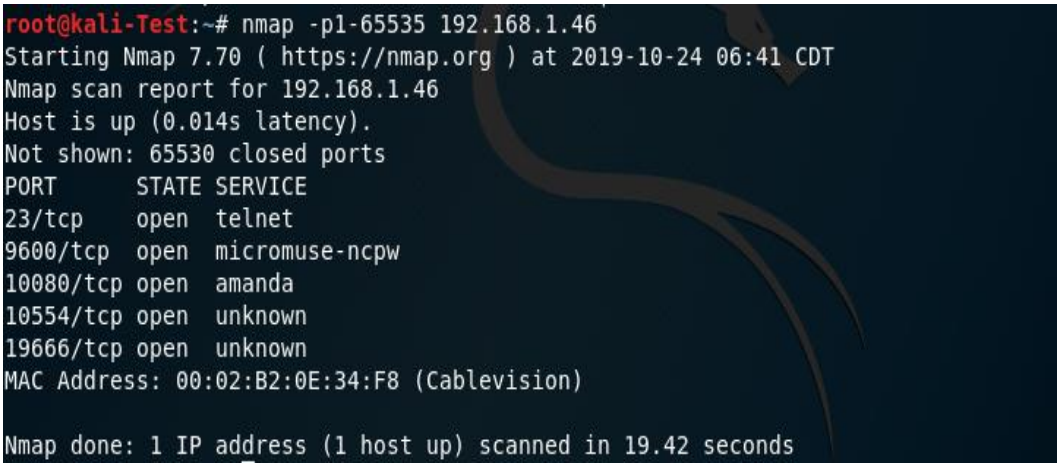

Figure 5.34. Nmap scan to find open ports

The scanned ports with Nmap indicate that there are five services open on TCP ports, such as port 23, 9600, 10080, 10554 and 19666 can be seen on the camera.

To discover additionally point by point data about the camera and its working framework, one more scan of Nmap with the alternative -O is required. Figure 5.35 clarifies the camera running Linux kernel 2.6.32 – 3.5 operating system.

root@kali-Test:~# nmap -O 192.168.1.46

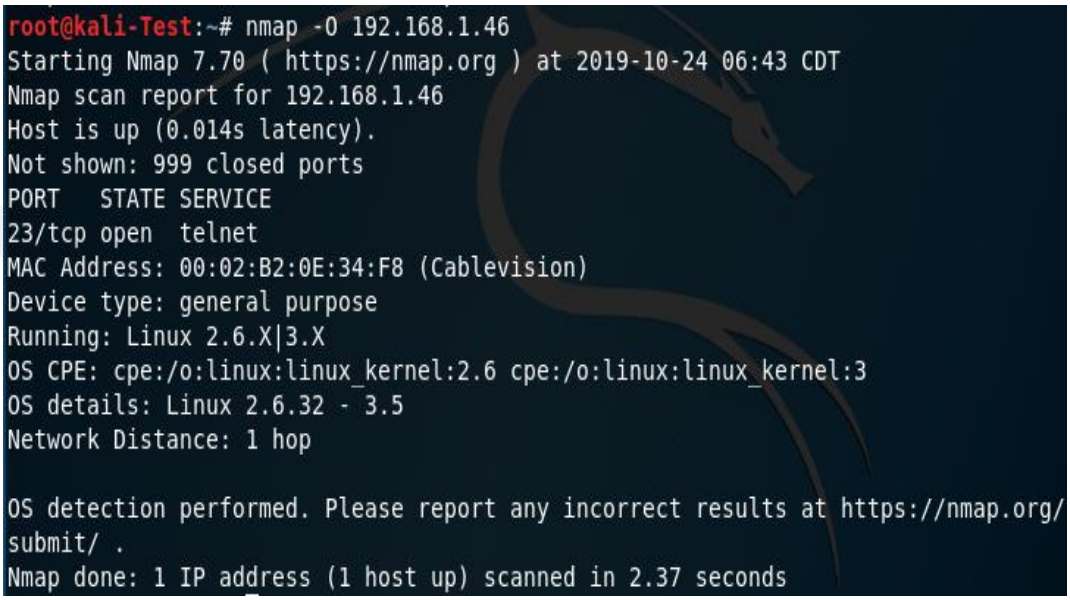

Figure 5.35. Nmap scan to detect operating system type on the device

5.4.2. Device ID and its Password

On the back of this camera, three pieces of information were written (P2P ID, username and password). There is an ID number with a length of up to sixteen characters (P2P ID: HSL-144328-DFDED) and default username (admin). The device ID and its password seem to be more secure compared to the previous cameras since device ID is very strong which includes sixteen characters with a combination of numbers and letters. This device does not have any default passwords and it must be set by the user during the process of configuration.

Also, in the supported app (e.g. NetCam) there is an option to “Add device” that permits the user to join the camera. Fortunately, the camera ID and password seem to be safe despite of the default username that does not seem so safe. Except the user has to set a strong password and modify the username to another one. Anyone cannot log in and communicate easily with this device by “examining” diverse cloud IDs because ID and password of the camera are not the only credentials required to enter this device, he/she needs device username also. Figure 5.36 shows the back of the camera and option to “add device” in NetCam mobile application.

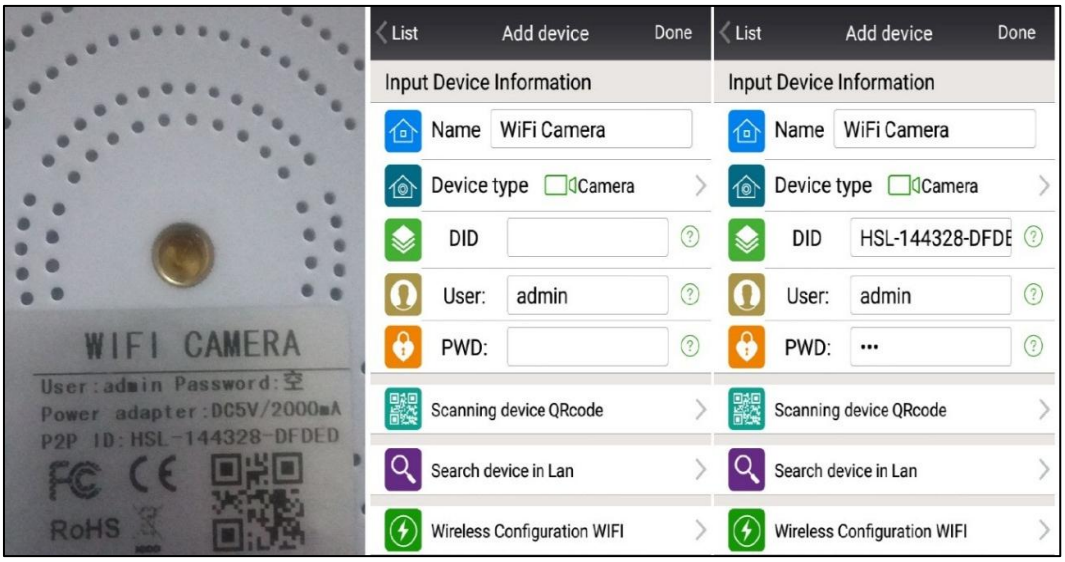

Figure 5.36. Add online device option in NetCam app

5.4.3. Brute Forceable RTSP URL to Stream Video

The findings of the Nmap test show that it was found to be an RTSP service running on port 10554. A basic strategy to discover the URL will be to simply URL brute force for basic stream names with the below command:

root@kali-Test:~# nmap –script rtsp-url-brute –p [port_number] [device_IP]

The result of the Nmap test finds a valid RTSP URL that indicates the device is vulnerable to the RTSP URL brute force attack. Also, through attaching the RTSP URL to the network stream URL page in VLC media player the live video stream of the camera can be watched. The convenient RTSP URL working with this camera are both "**rtsp://IP:10554/tcp/av0_0**" for HD quality and "**rtsp://IP:10554/tcp/av0_1**" for SD quality.

5.4.4. Inspecting Network Traffics

A simulated MITM attack was done by the procedures shown in section 5.2.4. Also, by utilizing Kali tools of network packet capture such as Mitmproxy, Bettercap, and Wireshark, it is possible to observe and monitor all network-level packet information.

At the beginning of the configuration process of this camera, the device must be connected to the internet through an RJ-45 wire cable, this makes NetCam mobile application to not send its SSID and password to the camera through the shared access point of the camera since the device itself does not share any access point for the purpose of configuring the device instead the configuration process takes place on the internet, which makes it more secure compared to the previous cameras.

During capturing of network data from NetCam and Device Client windows application and all other communication parts, only device ID captured in plaintext as shown in Figure 5.37. The device ID captured by Mitmproxy in plaintext highlighted with yellow color.

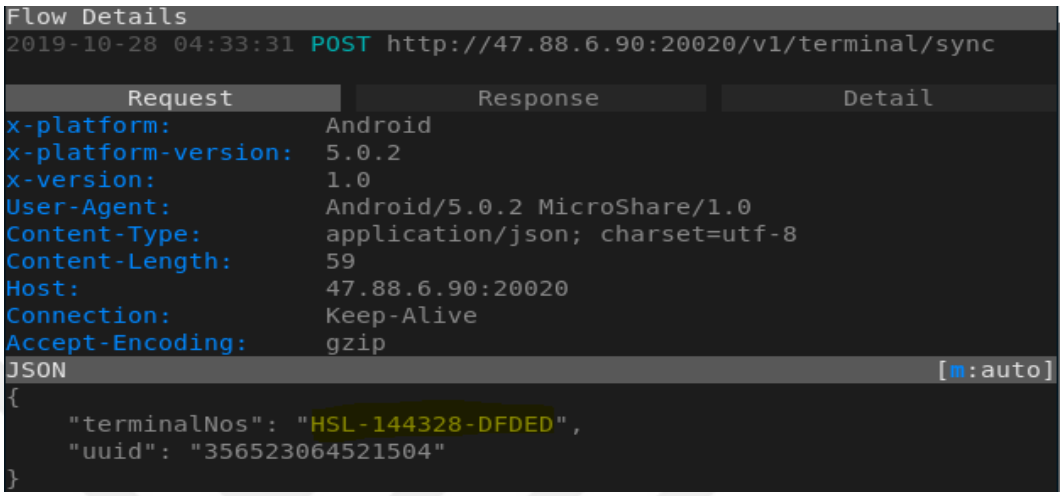

Figure 5.37. Device ID captured by Mitmproxy

5.4.5. DE-Authentication Attack

The de-authentication attack was successfully done against this device like the first and second cameras which indicate that this device is vulnerable to this attack. The de-authentication process was done by the steps that are shown in section 5.2.5. This attack disables the Wi-Fi security camera from the network and stops the camera to send the live video stream to the user for a period of time that the attackers want.

5.4.6. Inspecting IP Camera Account Credentials

When Inspecting IP camera account in NetCam, the Wi-Fi name and its password were stored in plaintext. Figure 5.38 shows the name of Wi-Fi (DSL-224-d890) and its password (53425331) as highlighted by a yellow color. These sensitive data can be found under the below directory path:

/data/data/hsl.p2pipcam/shared_prefs/wlan.xml

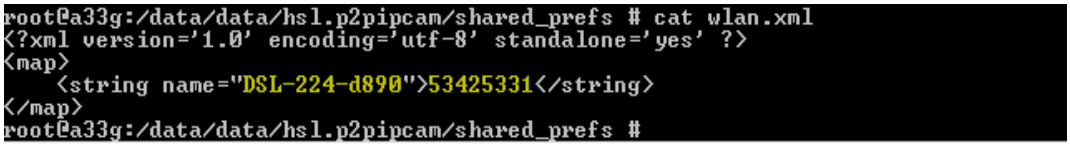

Figure 5.38. Wi-Fi SSID and its password stored in plaintext at NetCam app

Another vulnerability found in the NetCam app is that it does not have any login page, it means that everybody can use the application without any authentication.

5.5. Discussion and Comparison of Results

According to the outcomes of this thesis, there are many vulnerabilities exist in IP cameras. The first observation is that these devices are vulnerable by having default usernames and passwords which puts that devices to unsafe state, for instance, the first IP camera has a default password that is (123) and the second and third devices have default usernames that are (admin). On the other hand, the device ID seems unsafe to be found by an attacker. For instance, the first device has an ID (8861084) that does not seem to be so safe since it contains just seven digits (without a combination of letters and symbols). While the second device is also so similar to the first one, the third device ID seems stronger compared to the other devices since the length of this device ID is longer and contains letters, numbers, and special characters.

In addition, inspecting network traffic indicates that these devices have a lack of encryption. There are many sensitive types of information sent without any type of encryption or send with bad key encryption that can be easily decoded by the criminals. This type of vulnerability observed with the first and second device but fortunately, the third device seems good since just devices' ID captured in plaintext.

Also, the first and third devices are vulnerable to the RTSP URL brute force attack. URL can be used to obtain live video streams inside the LAN without any authentication.

Furthermore, a lack of security mechanisms is observed in the applications that support the devices in both smartphone apps and windows applications since some of the application has not set any policy for the password for the account and the devices (such as YYP2P and CMS Client). One another interesting thing that is observed with NetCam smart-phone application and Device Client windows applications are they do not have any login page meaning that everyone can log in to the account without any authentication. Also, all the applications do not have any lockout mechanism, it helps the attacker to conduct a brute force attack to obtain login credentials. Finally, the types of vulnerabilities that are found in this thesis briefly explained in the following points:

a. Default username and password
b. The device IDs are vulnerable to find by attackers
c. Vulnerable services (such as RTSP URL that could be used to obtain live video steam without any authentication)
d. Unencrypted communication (most of the sensitive data are sent without any type of encryption such as router login, device ID, device password, etc.)
e. Lack of Encryption (sensitive data are sent with bad key encryption which can be decoded by attackers)

f. Lack of security mechanisms (such as password policy, lockout mechanism, and authentication factors)

g. Vulnerabilities in smartphone applications (sensitive data and login credentials are stored in plain text)

Table 5.4 contains detailed information about the devices and comparison the finding from the penetration testing process.

Table 5.4. Comparison of results obtained from the penetration testing process

Component	Properties	Onvif	Panoramic	Dual-HD
Cameras	**Open ports**	554	5040, 5050, 5051, 7050, 8800	23, 9600, 10080, 10554, 19666
	Default password	Yes = 123	No (set by user). But it has default username = admin	No (set by user). But it has default username = admin
	Device ID	7 digits (8861084)	8 digits (27319433)	16 digits (HSL-144328-DFDED)
Mobile applications	**Application's name**	YYP2P - YOOSEE	V380	NetCam
	Login detail and other data how stored?	Username, Password, Device ID, Wi-Fi SSID and its password stored in plaintext	Username and Device ID stored in plaintext	It hasn't login credentials, also Wi-Fi SSID and its password stored in plaintext
	Password policy for the applications	YYP2P - Low (without any policy) YOOSEE - High (at least 8 digits with numbers & letters)	Middle (at least 6 digits)	It has not user login page
	Password policy for the camera	YYP2P - Low (without any policy) YOOSEE - High (randomly set by the app)	High (at least 6 digits with numbers & letters)	Low (without any policy)
	Authentication factors	Single-factor	Single-factor	Without authentication
	Account lockout mechanisms	No	No	No

Table 5.4. Comparison of results obtained from the penetration testing process (continue)

Computer applications	**Application's name**	CMS Client	V380	Device Client
	Password policy for the applications	Low (without any policy)	High (at least 6 digits with numbers & letters)	It has not login credentials page
	Password policy for the camera	Low (without any policy)	High (at least 6 digits with numbers & letters)	It has not change password option
	Authentication Factors	Single-factor	Single-factor	Without authentication
	Account Lockout Mechanisms	No	No	No
Communication	**Is sensitive data encrypted?**	Several sensitive data that sent without any encryption such as Username, Device ID, PII, Wi-Fi SSID, and Wi-Fi Password Account Password protected by MD5	Many sensitive data such as Username, Device ID, Device Password, Wi-Fi SSID and Wi-Fi Password are unencrypted. Just in some cases, Device ID encoded by Base64	Just Device ID captured in plaintext by Mitmproxy
RTSP URL to Stream Video		rtsp://IP/onvif1 (HD) rtsp://IP/onvif2 (SD)	RTSP closed	rtsp://192.168.1.46:10554/tcp/av0_0 (HD) rtsp://192.168.1.46:10554/tcp/av0_1 (SD)

6. CONCLUSIONS AND FUTURE WORK

6.1. Conclusion

The internet of things is a current model that appears to shift the world by the joint of different intelligent devices to the Internet such as smart-cameras, smart-plugs, smart-bulbs, etc., to build one intelligent and combined ecosystem. The history of the network is identified and the security issues, which still exist, are visible. This thesis declared a short review of the general IoT structure and define the functionality of each layer and popular IoT communication security along with the major security elements.

This thesis examined the security of three different IP cameras with three different brand names (Intelligent Onvif YY HD camera, Panoramic IP camera, and Dual-HD camera) via penetration testing methods and tools. The outcomes of this study demonstrated that the IP cameras have different vulnerabilities such as default username and password, the device IDs are vulnerable to find by attackers, vulnerable services (such as RTSP URL that could be used to obtain live video steam without ant authentication), unencrypted communication (most of the sensitive data are sent without any type of encryption such as router login, device ID, device password, etc.), lack of encryption (sensitive data are sent with bad key encryption which can be decoded by attackers), lack of security mechanisms (such as password policy and lockout mechanism), vulnerabilities in smartphone applications (sensitive data and login credentials are stored in plain text).

In order to improve the security of the devices, these weaknesses and vulnerabilities must be accounted for by the manufactures of the cameras. On the other hand, users must have a security background to protect their devices from cyber-attacks since weak or default passwords are one of the growing vulnerabilities for many IoT devices. If passwords are not modified to a strong one, an attacker will easily guess them. The types of vulnerabilities found in this study are clear signs that safety is not a company priority. In order to improve the security of the IoT devices and systems, especially IP cameras this thesis presented some solutions and suggestions. The suggested solutions in this study can be utilized to decrease or cut off the attacks on the IoT system.

Finally, the types of vulnerabilities found in this study are clear signs that safety is not a company priority.

6.2. Future Work

Internet of Things is a movement towards a significant accomplishment in the transmission network, but it has various difficulties at the same moment. To deal with it big network powerful resource management and preparation are demanded. Furthermore, nowadays the applications of IoT are increased, there are several different IoT devices that connected to the internet may have several vulnerabilities, the new researches in this filed may deal with other types of devices for the purpose find vulnerabilities and analyze security components such as smart-plugs and smart-bulbs. In the future, many other possible attacks may occur against IoT devices, for this purpose analyze other possible attacks and determine the other solutions in the field of IoT security is so important since the vulnerabilities did not find now, may be found by others in the future.

SUGGESTIONS

This thesis suggested some security solutions for protecting the IoT devices also fix the vulnerabilities and security gaps contained in the IP cameras:

- The manufactures of the IoT devices must improve the login credentials of the devices and change it to a stronger one (e.g. device ID and its password)
- OpenSSL library or other popular open-sourced SSL library can be used in the communication process for encrypting all the data among the applications, cameras, and servers
- Utilizing Simple authentication mechanisms in the RTSP protocol such as username and password can slow down the brute-force attackers that trying to find convenient URL to achieve live video streams
- Android KeyStore to save the login credentials can be used in android and other related iOS tools that encode and store the login details and sensitive data in hidden locations
- Remove the default passwords for the IoT devices and obligate the users to set a strong one in the configuration process to decrease or cut off the devices to be used in DDoS attacks
- Block specific IP address because of several wrong tries to log in to an IoT device
- Using some mechanisms to increase the security of an account from unauthorized access such as lockout and multi-factor authentication
- Utilize reliable protocols like HTTPS for communicating with IoT machines
- The users need to check router logs for malicious activity and undetermined IP address connections also, if the devices have default user names and passwords, this must be modified via users.

REFERENCES

[1] Roman, R., Najera, P., and Lopez, J. (2011). Securing the internet of things, Computer, C Vol. (9), 51-58.

[2] Bi, Z., Da Xu, L., and Wang, C. (2014). Internet of things for enterprise systems of modern manufacturing, IEEE Transactions on industrial informatics, C Vol. 10 (2), 1537-1546.

[3] Li, S. and Da Xu, L., (2017). Securing the internet of things, Syngress,

[4] Khan, R., Khan, S.U., Zaheer, R., and Khan, S. (2012). Future internet: the internet of things architecture, possible applications and key challenges. in 2012 10th international conference on frontiers of information technology, 257-260, IEEE, Islamabad.

[5] Bjørneset, K.J.W., (2017). Testing Security for Internet of Things: A Survey on Vulnerabilities in IP Cameras, University of Oslo. Master, Department of Information.

[6] Seralathan, Y., Oh, T.T., Jadhav, S., Myers, J., Jeong, J.P., Kim, Y.H., and Kim, J.N. (2018). IoT security vulnerability: A case study of a Web camera. in 2018 20th International Conference on Advanced Communication Technology (ICACT), 172-177, IEEE, Chuncheon-si Gangwon-do, Korea (South).

[7] Baloch, R., (2015). Ethical hacking and penetration testing guide, CRC Press, New York

[8] Swan, M. (2012). Sensor mania! the internet of things, wearable computing, objective metrics, and the quantified self 2.0, Journal of Sensor and Actuator networks, C Vol. 1 (3), 217-253.

[9] Markit, I. (2019). The top trends of 2019: powered by transformative technologies, URL: https://ihsmarkit.com/Info/0119/top-tech-trends-2019.html. Access: 3 March 2020.

[10] Alharbi, R. and 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, 1-10, IEEE, London.

[11] Tekeoglu, A. and Tosun, A.S. (2015). Investigating security and privacy of a cloud-based wireless IP camera: NetCam. in 2015 24th International Conference on Computer Communication and Networks (ICCCN), 1-6, IEEE, Las Vegas.

[12] Liranzo, J. and Hayajneh, T. (2017). Security and privacy issues affecting cloud-based IP camera. in 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), 458-465, IEEE, New York.

[13] Bugeja, J., Jönsson, D., and Jacobsson, A. (2018). An Investigation of Vulnerabilities in Smart Connected Cameras. in 2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), 537-542, IEEE, Athens.

[14] Costin, A. (2016). Security of cctv and video surveillance systems: Threats, vulnerabilities, attacks, and mitigations. in Proceedings of the 6th international workshop on trustworthy embedded devices, 45-54, ACM, Vienna.

[15] Obermaier, J. and Hutle, M. (2016). Analyzing the security and privacy of cloud-based video surveillance systems. in Proceedings of the 2nd ACM International Workshop on IoT Privacy, Trust, and Security, 22-28, ACM, Xi'an.

[16] Buyya, R. and Dastjerdi, A.V., (2016). Internet of Things: Principles and paradigms, Elsevier,

[17] Zhou, Q. and Zhang, J. (2011). Research prospect of Internet of Things geography. in 2011 19th International Conference on Geoinformatics, 22-28, IEEE, Barcelos, Portugal.

[18] Muhammad, M.F., Anjum, W., and Mazhar, K.S. (2015). A Critical Analysis on the Security Concerns of Internet of Things (IoT), International Journal of Computer Applications (0975 8887), C Vol. 111 (7).

[19] Matharu, G.S., Upadhyay, P., and Chaudhary, L. (2014). The internet of things: Challenges & security issues. in 2014 International Conference on Emerging Technologies (ICET), 54-59, IEEE, Islamabad.

[20] Mahmoud, R., Yousuf, T., Aloul, F., and Zualkernan, I. (2015). Internet of things (IoT) security: Current status, challenges and prospective measures. in 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST), 336-341 IEEE, London.

[21] Miorandi, D., Sicari, S., De Pellegrini, F., and Chlamtac, I. (2012). Internet of things: Vision, applications and research challenges, Ad hoc networks, C Vol. 10 (7), 1497-1516.

[22] Atzori, L., Iera, A., and Morabito, G. (2010). The internet of things: A survey, Computer networks, C Vol. 54 (15), 2787-2805.

[23] Hossain, M.M., Fotouhi, M., and Hasan, R. (2015). Towards an analysis of security issues, challenges, and open problems in the internet of things. in 2015 IEEE World Congress on Services, 21-28, IEEE, New York.

[24] Abomhara, M. and Køien, G.M. (2014). Security and privacy in the Internet of Things: Current status and open issues. in 2014 international conference on privacy and security in mobile systems (PRISMS), 1-8, IEEE, Alborg.

[25] Fraunholz, D., Krohmer, D., Anton, S.D., and Schotten, H.D. (2017). Investigation of cyber crime conducted by abusing weak or default passwords with a medium interaction honeypot. in 2017 International Conference on Cyber Security And Protection Of Digital Services (Cyber Security), 1-7, IEEE, London.

[26] Zhang, C. and Green, R. (2015). Communication security in internet of thing: preventive measure and avoid DDoS attack over IoT network. in Proceedings of the 18th Symposium on Communications & Networking. Society for Computer Simulation International.

[27] Gilchrist, A., (2017). IoT Security Issues, Walter de Gruyter, Berlin

[28] Siby, S., Maiti, R.R., and Tippenhauer, N. (2017). Iotscanner: Detecting and classifying privacy threats in iot neighborhoods, arXiv preprint arXiv:1701.05007.

[29] Sinha, S., (2018). Beginning Ethical Hacking with Kali Linux: Computational Techniques for Resolving Security Issues, Apress, Howrah, West Bengal, India

[30] Abdalla, P.A. and Varol, A. (2019). Advantages to Disadvantages of Cloud Computing for Small-Sized Business. in 2019 7th International Symposium on Digital Forensics and Security (ISDFS), 1-5, IEEE, Barcelos, Portugal.

[31] Balthasar and Fabian. The Cloud exposes your private IP cameras. https://srlabs.de/bites/cloud-cameras/, Access: 2 Jan 2020.

[32] Sezov, R., (2008). Liferay Administrator's Guide, Lulu.

[33] Reddy, S., (2017). Secure Password Policy. Current Situation and Solutions, Grin Publishing,

[34] Lu, B., Zhang, X., Ling, Z., Zhang, Y., and Lin, Z. (2018). A measurement study of authentication rate-limiting mechanisms of modern websites. in Proceedings of the 34th Annual Computer Security Applications Conference, 89-100, ACM, London.

[35] Liou, J.-C. and Bhashyam, S. (2010). A feasible and cost effective two-factor authentication for online transactions. in The 2nd International Conference on Software Engineering and Data Mining. IEEE.

[36] Ezeofor, C. and Ulasi, A. (2014). Analysis of network data Encryption & Decryption techniques in communication systems, International Journal of Innovative Research in Science, Engineering and Technology, C Vol. 3 (12), 17797-17807.

[37] Najera-Gutierrez, G. and Ansari, J.A., (2018). Web Penetration Testing with Kali Linux: Explore the methods and tools of ethical hacking with Kali Linux, Packt Publishing Ltd,

[38] EC-Council, (2011). penetration Testing: Procedures and Methodologies, Cengage Learning,

[39] Cardwell, K., (2016). Building Virtual Pentesting Labs for Advanced Penetration Testing, Packt Publishing Ltd, Birmingham

[40] Allen, L., Heriyanto, T., and Ali, S., (2014). Kali Linux–Assuring security by penetration testing, Packt Publishing Ltd, Birmingham

[41] Calderon, P., (2017). Nmap: Network Exploration and Security Auditing Cookbook, Packt Publishing Ltd, Birmingham

[42] Mishra, C., (2018). Wireshark 2 Quick Start Guide Secure your network through protocol analysis, Packt Publishing Ltd, Birmingham

[43] Parasram, S.V., Samm, A., Boodoo, D., Johansen, G., Allen, L., Heriyanto, T., and Ali, S., (2018). Kali Linux 2018: Assuring Security by Penetration Testing: Unleash the full potential of Kali Linux 2018, now with updated tools, Packt Publishing Ltd, Birmingham

[44] Beltrame, J., (2017). Penetration Testing Bootcamp, Packt Publishing Ltd, Birmingham

[45] Johansen, G., Allen, L., Heriyanto, T., and Ali, S., (2016). Kali Linux 2–Assuring Security by Penetration Testing, Packt Publishing Ltd,

[46] Velu, V.K., (2017). Mastering Kali Linux for Advanced Penetration Testing, Packt Publishing Ltd,

[47] McPhee, M. and Beltrame, J., (2016). Penetration Testing with Raspberry Pi, Packt Publishing Ltd, Birmingham

[48] Bodenheim, R., Butts, J., Dunlap, S., and Mullins, B. (2014). Evaluation of the ability of the Shodan search engine to identify Internet-facing industrial control devices, International Journal of Critical Infrastructure Protection, C Vol. 7 (2), 114-123.

[49] Kolias, C., Kambourakis, G., Stavrou, A., and Voas, J. (2017). DDoS in the IoT: Mirai and other botnets, Computer, C Vol. 50 (7), 80-84.

[50] Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., Durumeric, Z., Halderman, J.A., Invernizzi, L., and Kallitsis, M. (2017). Understanding the mirai botnet. in 26th USENIX Security Symposium, 1093-1110, London.

[51] Kambourakis, G., Anagnostopoulos, M., Meng, W., and Zhou, P., (2019). Botnets: Architectures, Countermeasures, and Challenges, CRC Press, United States.

CURRICULUM VITAE

Peshraw Ahmed ABDALLA

PERSONAL INFORMATION

Place of Birth : Sulaimani
Date of Birth : 12/6/1991
Nationality : Iraqi
Address : Sulaimani - Iraq
E-mail : Peshraw.abdalla1991@gmail.com
Languages : Turkish(B1), English(B2)

EDUCATION

Bachelor : Sulaimani University, College of Basic Education, Department of Mathematics and Computer, Year 2013
High School : Azmar High School, Sulaimani City, Year 2009

RESEARCH EXPERIENCES

- ✓ Programming Languages (C++, C#, Java, Visual Basic)
- ✓ Object-Oriented Programming
- ✓ Web Design

WORK EXPERIENCE

2013 – 2017: Research Assistance (University of Sulaimani)
2017 – 2018: Research Assistance (University of Halabja)

ACADEMIC ACTIVITIES

1. Abdalla, P.A. and Varol, A. (2019) *Advantages to Disadvantages of Cloud Computing for Small-Sized Business*. in *2019 7th International Symposium on Digital Forensics and Security (ISDFS)*, 1-5, IEEE. Portugal.
2. Abdalla, P.A. and Varol, C. (2020) *Testing IoT Security: The Case Study of an IP Camera*. in *2020 8th International Symposium on Digital Forensics and Security (ISDFS)*, 1-6, IEEE. Lebanon.