Foscam C1 Login Spoofing: Leveraging ARP

Lorik Heijnen
l.heijnen1@student.tue.nl

Jeroen Hellenbrand
j.w.f.hellenbrand@student.tue.nl

Sam Nijsten
s.j.h.a.nijsten@student.tue.nl

***Abstract*—The use of security cameras in private homes is gradually increasing. Between 2013 and 2016, the IVMP estimated an Compound Annual Growth Rate (CAGR) in homes using security or surveillance cameras of 20% [8]. From 2019 to 2025 the CAGR is expected to be 14% [14]. A significant factor of this growth is the popularity of Internet Protocol (IP) cameras. Unfortunately, the Internet of Things (IoT), which IP cameras are a part of, is known for its vulnerabilities. Using the Foscam C1 IP camera as a representative of this category, we demonstrate the weaknesses of IP cameras. A threat scenario is given in which these attacks are used in a realistic manner. We will exploit the Address Resolution Protocol (ARP), a request-response protocol used in the link-layer of the OSI model. We leverage the weaknesses of ARP to establish a Man in The Middle (MiTM) position, from which we can perform Denial of Service (DoS) attacks and login spoofing. We find that measures can quite easily be taken to prevent such attacks from being successful, but that the default settings found in the commercial grade IP cameras are often lacking.**

I. Introduction

The market for security cameras designed for home usage is growing steadily. The Internet Protocol (IP) camera market is valued at over 8 billion USD in 2018, and is expected to maintain a Compound Annual Growth Rate of 14% from 2019 to 2025. [14]. This growth is mainly due to rising security threats and declining manufacturing costs. The steps taken to acquire unsolicited access to the live video feed of an IP camera, using the commercial grade Foscam C1 IP camera as illustrative for this group, will be presented in this paper.

We will use Address Resolution Protocol (ARP) poisoning to become a Man in The Middle (MiTM) between the HTTP server, running on the IP camera, and the victim. We will extract essential data, such as HTML, CSS and JavaScript files, using sniffing software such as Wireshark to create an identical login page. Using cookies, the attacker will be able to obtain the username and password from this identical login screen. With the victim's username and password, the attacker obtains unsolicited access to the account of the victim, and watch the camera's live video feed. Furthermore, we will use the MiTM position to conduct a series of Denial of Service attacks. From the victim's perspective, this will cause the video stream to be interrupted.

Finally, we will analyze our attack and propose certain countermeasures that can be deployed to protect the victim from future attacks.

II. Attack Description

A. Attack Scenario

The Attack Scenario consists of a security-conscious but digitally illiterate home or small business owner, henceforth referred to as the *victim*. The building which is to be protected has been fitted out with the commercially available Foscam C1 IP camera [5]. We assume the victim uses his or her browser to check the camera's video stream on a regular basis. Furthermore, we assume that the attacker has gained access to the Local Area Network (LAN) of the victim. This can be accomplished by connecting to an unprotected Wireless Access Point (WAP), utilizing software such as the Aircrack-NG suite [1] to get access to the WAP, or ensuring a physical connection.

The attacker intends to break into the building without arousing suspicion. To achieve this goal, an abundance of information, such as the building's layout, the movement of residents and the location of valuables, can be gathered from the video stream. When breaking into the building, it is to the attacker's advantage if the camera fails. As a result, the attacker intends to prevent the victim from viewing the video feed.

B. Attack Execution

The attack execution has been made intuitive by packaging the attack code into a simplistic applet. As a result of this ease of use, the attacker is only required to possess an elementary knowledge of networks. Prior to executing the attack, the attacker must learn the IP addresses of the victim machine, IP_v, and the camera, IP_c. Foscam provides a plugin to scan the network and discover IP_c. However, the notion of stealth is violated by using such software. For this purpose, software such as *Nmap* is recommended [12]. Furthermore, the attacker must establish a web-server, running a website identical to the login page used by the webcam.

Armed with IP_c, IP_v, the IP address of the web-server, and the IP address of the attacking machine, the attacker can launch the applet and is greeted by the User Interface (UI) seen in Figure 1. To launch the attack, the attacker enters the corresponding IP addresses and presses either of the buttons at the bottom of the page. Unambiguously, the button marked *DoS* will launch a Denial of Service attack, while the button marked *Login Spoof* will launch an attack designed to gain access to the victims login credentials.

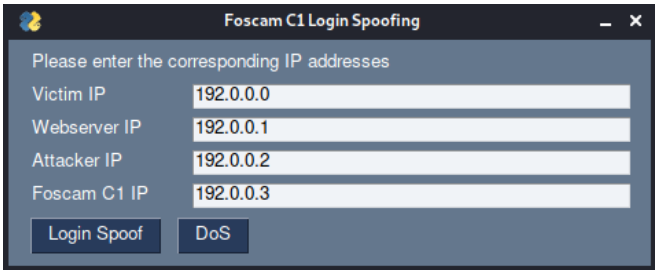

Fig. 1. The UI of the applet used to deploy the attack

III. Technical Setup

A. Victim

As mentioned in Section I, the target camera was the Foscam C1 IP camera running firmware version 1.12.5.4_2.82.2.33. As we assumed the owner of the target camera to be digitally illiterate, the instruction booklet [7] provided by Foscam was followed whilst installing the camera. As a result, the camera uses HTTP rather than HTTPS, because HTTP is the default protocol. Furthermore, the instruction booklet barely mentions the SD-card functionality of the camera. This, together with the fact that no SD-card was included in the box, leads to the conclusion that there was no SD-card in the camera at the time of the attack. Moreover, the camera was connected to the LAN via Ethernet.

The target machine was the Lenovo P1 running Windows 10 Enterprise. The browser used by the victim is irrelevant. A large number of devices, such as smartphones, smartwatches, PCs, and smart televisions, were connected to the LAN to simulate public WiFi or a home network in everyday use.

B. Attacker

The attacking machine was a virtual machine inside Oracle's VirtualBox running Kali Linux version 2021.1. The attacking machine was also connected to the LAN via Ethernet. The software used by the attacker was Ettercap version 0.8.3 [6], mitmproxy version 6.0.2 [10], and Apache HTTP server version 2.4.48 [2].

A web server is needed to host the false login page in order to carry out the attack, which will be achieved with an Ubuntu-based Virtual Machine. Apache2 [2] will be utilized to run said server. The fake login page and images are placed in the server's directory from which it reads. In order to redirect the HTTP request to the proxy server, the port configuration file needs to altered to accept traffic on port 88. The command `sudo systemctl start apache2.service` can be executed to start the server. The web server is now prepared for the attack.

IV. Attack Analysis

The attacks performed were twofold, each focusing on a different aspect of security. Initially, a Denial of Service (DoS) attack was carried out, resulting in the the security attribute *availability* being disintegrated. Secondly, the IP camera's password was obtained, resulting in the violation of the security attribute *confidentiality*. From the attacker's standpoint, we'll now go over both attacks in depth. Table I summarizes the outcome of the attacks.

TABLE I
Outcomes of the attacks

Attack	Outcome
ARP Poisoning	MiTM between the victim and the IP camera
Naive DoS	Breach of availability concerning the complete network
Directed DoS	Breach of availability concerning the IP camera
Login Spoofing	Breach of confidentiality

A. Address Resolution Protocol

To start the attack, a MiTM position must be achieved. As discussed in Section I, we will use the ARP to achieve this goal. ARP is a request-response link layer protocol [3] that does not provide any methods for confirming the authenticity of ARP replies. This weakness can easily be leveraged to perform an attack known as *ARP Poisoning*. By sending spoofed ARP replies to both the victim and the camera, replacing the corresponding MAC address by the attacker's MAC address, we have placed the attacker in between the victim and the camera.

B. Denial of Service

The attackers repertoire consists of two types of Denial of Service (DoS) attacks, a naive and a directed attack. The naive attack leaves the victim completely unable to connect to the network. From the attacker's perspective, this has clear drawbacks, as the victim will immediately be alerted to the possibility of an attack. The directed DoS will attempt to mitigate this disadvantage by simply blocking the victim's access to the camera, whilst allowing connections to the remainder of the network.

1) Naive DoS: In the naive version of the DoS attack, the attacker will flood the target unconditionally, causing the target to lose all network access. The attacker enacts such an attack by sending out an abundance of spoofed ARP reply packets, commonly referred to as ARP flooding or ARP storming. This flood of ARP replies will corrupt the ARP table for a wide range of IP addresses, effectively rendering the IP to MAC mapping ineffective.

Ettercap sends out ARP replies for all IP addresses in the current net mask. It is possible to extend this to include addresses from outside of the net mask [13]. Because the connection with the gateway router is already interrupted by spoofing all the addresses inside the net mask, extending the attack to include addresses from outside the current net mask does not result in a stronger attack.

2) Directed DoS: The MiTM position is used to launch the directed DoS attack. Recall that the victim isn't connected to the IP camera directly. The attacker collects packets from the victim and passes them to the camera, which then sends packets back to the victim. The victim is then fed these

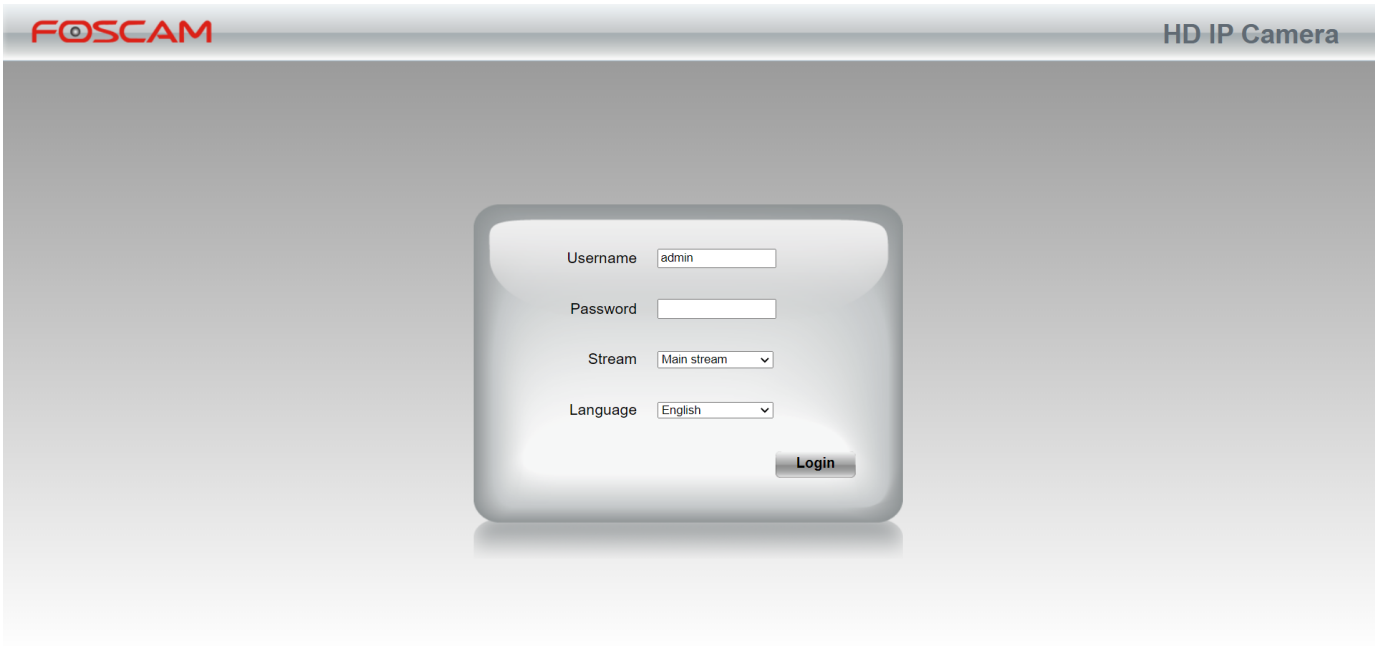

Fig. 2. The identical fake login page

packets, giving the appearance of a connection. The attacker only needs to stop relaying packets to 'disconnect' the victim from the camera. The victim will no longer be able to connect to the camera, but will still have full access to the rest of the network. As a result, a targeted DoS has been achieved.

Both DoS attacks were executed successfully. In particular, note the ease with which these attacks can be executed as a result of the MiTM position.

C. Login Spoofing

On the original login page of the Foscam C1 IP camera, there is JavaScript present which scrambles the user input before completing the request. Because of this security feature, the password cannot be retrieved simply by becoming a man in the middle. Hence, a fake login page, as shown in Figure 2, has been created. As soon as the user enters their credentials, a pop-up will appear telling the user that something went wrong, as shown in Figure 3. A similar style pop-up will appear on the original page if the username and/or password is incorrect.

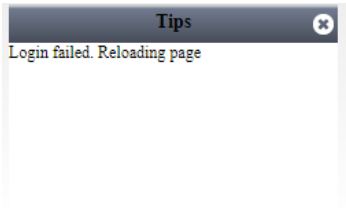

Fig. 3. The pop-up after logging in on the fake page

After the pop-up appeared, the page reloads to the actual login page. Meanwhile, the attacker can examine the target's credentials by sniffing the network packets, as they have been saved as cookies.

To get the fake login page to the user, a web-server with the malicious login page is setup. Recall that Ettercap [6] is used to ARP poison the target PC and the Foscam camera. Since all communication between the IP camera and the target PC goes via the attacker, the HTTP request for the website can be changed using *mitmproxy* [10]. The *mitmproxy* tool sets up a proxy server and redirects the request to the web-server, as described in section III-B, and sends the malicious website to the target PC as though it originated from the camera. After the victim presses the log-in button on the malicious website, the scrip stops redirecting the HTTP request and the target automatically transmits a new request for the original website, since the page is automatically refreshed. The proxy server will forward the request to the camera and forward the response from the camera. The schematic of this attack is illustrated in Figure 4.

D. Prevention Methods

Because our assault relies entirely on a MiTM position gained by exploiting the shortcomings of ARP, the most effective countermeasures are those taken to prevent ARP poisoning. The victim machine, for example, can use MAC address white-listing, which prevents the victim machine from accepting ARP replies from MAC addresses it has not white-listed. The victim might also prefer to assign a static ARP entry to IP_c, as a result of the dynamic nature of IP_c, this might not be a permanent solution.

If the attacker's MAC address is known, a firewall operating at the link-layer of the OSI stack can be installed, for example

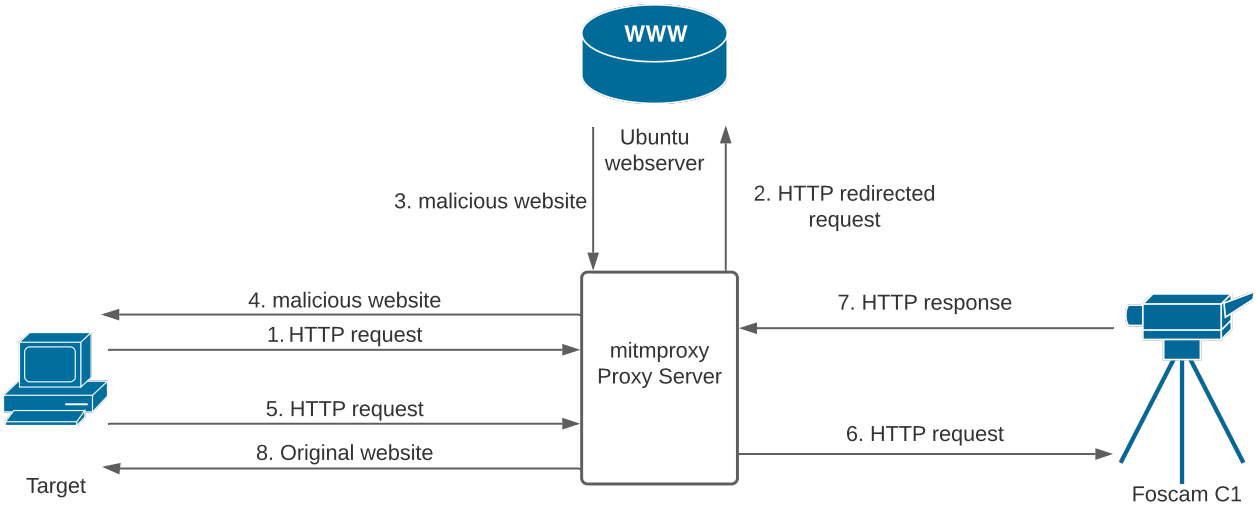

Fig. 4. mitmproxy proxy server redirection of HTTP request.

by studying the ARP table and detecting duplicate MAC addresses. The attacking machine's MAC address can be easily changed, resulting in a cat-and-mouse game between the attacker and the victim. Foscam also provides a built-in firewall for the camera in which you can either white- or blacklist IP addresses [7]. Another option for the victim would be to deploy an Intrusion Detection System (IDS). As the victim is presumed to have limited resources, the IDS must have a low overhead. Such an IDS based on Linear Temporal Logic (LTL), is proposed by Mitra et al. [11] Mitra et al. published a paper with encouraging experimental results, including excellent accuracy and detection rate as well as low overhead.

Measures can be made to avoid the attacks detailed in Sections IV-B and IV-C in addition to preventing ARP poisoning. The approach provided by Vidya and Bhaskaran [13] can be used to prevent the naive DoS attack, whose efficacy is reliant on ARP flooding. To prevent the connection to the falsified login-page, the camera should operate on HTTPS, as this requires the attacker to fake certificates.

It is worth noting that the majority of these measures necessitate some technical expertise. Most of these procedures are ineffective in our threat scenario since we envisioned a victim who is digitally illiterate. Nonetheless, it's worth mentioning them here because they can easily be set as the default option on IP cameras of this type.

V. Social Engineering

The login spoofing attack, as explained in Section IV-C, relies on the victim being unaware that the login page has been faked. To achieve this, the fake login page looks and behaves identically to the original login page. As a result, the victim is persuaded by the attacker to provide his/her login credentials [4]. Due to certain cognitive biases, such as the belief bias and cognitive dissonance [9], the victim will most likely be unaware that the current website is not legitimate. It is easier for the victim to believe he entered incorrect credentials than it is to accept the unpleasant reality that an attacker is trying to gain unsolicited access.

VI. Future Work

When logged into the camera, the user can gain insights into his usage of the camera by inspecting a log of IP addresses that have been used to log into said camera. As a result of the attacker logging in to the target's account, the IP address of the attacker is recorded and visible in the account settings. To resolve this, IP spoofing can be deployed so that the camera record does not show a suspicious IP address, but rather the IP address of the target.

Furthermore, as a result of gaining the log-in credentials and hence gaining access to the settings of the camera, the settings could be altered to enable even more powerful attacks. The encoding of the packets could be broken and hence static images could be send to the victim, who then perceives nothing out of the ordinary when looking at the 'live' video feed.

VII. Project Code

The code used in this project is uploaded to Github and can be found in this repository.

REFERENCES

[1] Aircrack-ng. (2020). Retrieved 12 June 2021, from https://www.aircrack-ng.org/

[2] Apache HTTP Server Project. Retrieved 15 June 2021, from https://httpd.apache.org/

[3] Plummer, D. (1982), "An Ethernet Address Resolution Protocol: Or Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware", STD 37, RFC 826, DOI 10.17487/RFC0826. Retrieved 14 June 2021, from https://www.rfc-editor.org/info/rfc826

[4] Allodi, L. (2021). L9 - Hacking a human. Offensive Computer Security.

[5] Foscam C1 HD IP camera. (2016). Retrieved 11 June 2021, from https://www.foscam.nl/c1-b.html

[6] Ettercap. (2019). Retrieved 29 May 2021, from https://www.ettercap-project.org/

[7] Foscam Home Security. (1998). User Manual: Indoor HD IP Camera. Retrieved from https://foscam.com/downloads/user_manual.html.

[8] Honovich, J. (2016). Home Security Camera Statistics 2016. Retrieved 9 June 2021, from https://ipvm.com/reports/home-security-camera-statistics-2016

[9] Kahneman, D., Egan, P. (2011). Thinking, fast and slow. New York: Random House Audio.

[10] mitmproxy - an interactive HTTPS proxy. (2013). Retrieved 8 June 2021, from https://mitmproxy.org/

[11] Mitra, M., Banerjee, P., Barbhuiya, F., Biswas, S., Nandi, S. (2013). IDS for ARP spoofing using LTL based discrete event system framework. Networking Science, 2(3-4), 114-134. doi: 10.1007/s13119-013-0019-1

[12] Nmap: the Network Mapper. Retrieved 13 June 2021, from https://nmap.org/

[13] Vidya, S., Bhaskaran, R. (2011). ARP Storm Detection and Prevention Measures. IJCSI International Journal of Computer Science Issues, 8(2), 456-460. issn: 1694-0814

[14] Wadhwani, P., Gankar, S. (2019). IP Camera Market Size By Product. Global Market Insights.