Regular Paper

Dangers of IP Camera – An Observational Study on Peeping

Kazuki Tamiya[1] Aamir H. Bokhari[1,a)] Yuta Ezawa[1] Sou Nakayama[1] Ying Tie[1]
Rui Tanabe[2] Akira Fujita[2] Katsunari Yoshioka[3] Tsutomu Matsumoto[3]

Received: November 25, 2019, Accepted: June 1, 2020

Abstract: Existing research on information security for IP cameras has been primarily focused on issues with authentication or malware, but not on the peeping method itself. How cyber peeping is conducted in real world can further help in strengthening defenses accordingly and spread more awareness about dangers of IP camera. In this research, we observed peeps by setting up a honeypot using decoy cameras in two scenarios. First, where background information (handwritten URL and ID/password bait) can be read by humans. Second, simulating a living-room in a home environment. As a result, many examples of peeping into the decoy cameras were confirmed in reality. Also, a rapid increase in peeping (over 20,000 times/day) was seen after a decoy camera's feed got posted on a well-known website, showing a large scale peeping danger also exists due to such websites. The results of this study were used in several TV programs to show the dangers of using IP cameras over a national broadcasting station and also were directly shared with IP camera vendors, resulting in the improvement of IP camera security. Therefore, we believe that this study can further help in improving the security and awareness on the dangers associated with IP cameras.

Keywords: IP camera, authentication, peeping, privacy, honeypot, web

1. Introduction

In recent years, more and more various types of devices are getting connected to the Internet. A common term has been coined to call all such devices as "Internet of Things" (IoT). Such IoT devices offering various services are attracting a lot of attention. Among IoT devices, digital cameras that allow remote viewing and operations via internet can be collectively referred to as IP cameras. There are many IP camera devices connected to the Internet with vulnerability and authentication issues, and it is, therefore, possible for a third person to electronically peep into them. Such peeping actions via internet can be referred to as cyber peeping. Another issue is due to the presence of certain web sites, like Insecam [1], which provides video images of freely accessible IP cameras for anyone, thereby, creating security and privacy issues.

So far, most of the research on unauthorized access to IP cameras has been focused mainly on authentication, changing camera configuration information, and observing malware infections that exploit vulnerabilities [2], [3], [4], [5], [6], [7], [8]. However, there has been no investigation into the actual state of cyber peeping. Knowing the peeping techniques can help in further understanding the risks associated with using IP cameras and improving awareness among the general public and IP camera vendors from the security and the privacy point of view. Therefore, in this research study, in order to investigate and analyze the actual situation of cyber peeping in the real world, an IP camera was set up as a honeypot (hereinafter referred to as the "decoy camera") for conducting an observational experiment. Two types of observation environments were established for the experiment purposes.

In the first observation experiment, we prepared two cameras (hereinafter referred to as the "URL reflection type decoy camera"), that displayed a hand-written note for a specific URL and two sets of different ID/passwords (one for each decoy camera) for access confirmation purposes, and assigned 10 IP addresses to each camera for observations. The objective was to study an access by a human element, who can read the background information (reflected URL and ID/password) via peeping into the decoy camera and then use that information to successfully gain access to the reflected URL. On the URL side, we examined what ID/password was entered for determining if humans were attempting access after viewing the video of the decoy camera.

Although the first observational study helped in determining human element involvement, it was limited due to the fact that such a set up did not provide continuous peeping interest as the decoy camera was only showing a URL and its related ID/password. Therefore, a second observation experiment was setup where a room was prepared for observation that simulated a living-room of a home in which movement can be expected. Five decoy cameras (hereinafter referred to as the "living-room decoy camera") were installed to show the video of this room. These living-room decoy cameras were then exposed to internet

[1] Graduate School of Environment and Information Sciences, Yokohama National University, Yokohama, Kanagawa 240–8501, Japan
[2] Institute of Advanced Sciences, Yokohama National University, Yokohama, Kanagawa 240–8501, Japan
[3] Graduate School of Environment and Information Sciences, Yokohama National University/Institute of Advanced Sciences, Yokohama National University, Yokohama, Kanagawa 240–8501, Japan
[a)] aamir-bokhari-rd@ynu.jp

so that peeping methods could be examined in detail for a longer time. The objective was to attract the peeping entity to engage in more than one peeping event by providing a real life scenario.

As the result of this study, we were able to achieve the following:

(i) First observational study of IP camera peeping using real IP cameras in various environments.
(ii) Demonstrated that a peeping problem does exist with a high degree when IP cameras are insecure.
(iii) Revealed detailed access patterns on the insecure cameras, including the existence of automated accesses specifically tailored to find IP cameras efficiently.

Moreover, we had conversations with several IP camera vendors and explained the real world risks of insecure IP cameras. One of the vendors now deploy a security mechanism in their IP camera products that enforces users to set their own unique password.

2. IP Camera Peeping

2.1 Basics of IP Camera

An IP camera is a camera that can be viewed remotely via a network by connecting to the internet. A web browser or manufacturer-specific application software is used to view its video. There are also IP cameras with functions allowing operational control such as directional movement or zooming etc. that can be operated via the browsers and/or an application.

2.2 Peeping into the IP Camera

Many IP cameras are set to perform authentication via login ID/ password so that privacy can be maintained and images are not viewed by others than the authorized users. However, there are cases in which authentication may not be set or the login ID/password may not have been changed from the initial default setting, allowing unauthorized access from the outside and cyber peeping may be possible [9], [10], [11]. Also, even if the ID/password is set, there are many devices with weak or leaked passwords due to device vulnerabilities that can be peeked by outsiders [10], [12]. Similarly, there are many devices for which security measures may not be sufficient [9], [11].

Furthermore, there are several websites that gather information on cameras and videos that can be viewed without authentication [8], [13], [14], tens of thousands of such cameras worldwide are posted on the Insecam web site [1].

3. Related Research

IP cameras are one of the mostly used devices among Internet of Things (IoT). In 2018, FBI warned [15] about the dangers of IoT devices, including IP cameras, to be used as “proxies for anonymity and pursuit of malicious cyber activities”. Most of the research primarily has been focused on vulnerabilities for exploitation of IP cameras. For example, in paper [7] the author has analyzed the security of cloud-based video cameras by focusing on vulnerabilities for exposing potential issues in IP cameras. Also, paper [6] discusses about the security of smart homes having IP cameras as well, but examines it from systems design and security vulnerabilities point-of-view. Another article [8] talks about peeping by exploiting default passwords, but not the peeping phenomenon itself. Similarly, server-type honeypots that monitor remote exploit attacks and collect malware specimens, have been researched as honeypots for observing attacks on web services by using general purpose responses to services [16], [17]. In addition, IoTPOT [2], [3], [4] and SIPHON [5], which are honeypots simulating IoT devices, have been proposed. However, in papers [2], [4], [5], [6], [7], [8], [16], [17] no detailed investigation has been conducted on the peeping technique for acquiring the IP camera video or images. Similarly, in paper [3], a successful cyber-attack by an attacker (who viewed the video through the camera that displayed or reflected an ID/password) was confirmed, but the actual method of peeping was not analyzed. Therefore, in this research study, in order to investigate this fact, we experimented with a honeypot of IP cameras (decoy cameras) to observe the peep and analyze the detailed method.

4. Observational Experiment

4.1 Environment Setup

In this research, the communication environment of the decoy camera was constructed by extending the method of IoTPOT proposed in paper [4]. In this honeypot, as shown in **Fig. 1**, a proxy script is running, and the received communication is transferred to the communication control machine.

In the communication control unit, the communication is transferred to the IP camera corresponding to each observation point, and the response is transferred to the proxy script at each observation point, such that an IP camera operating at each observation point is made to appear to an attacker as directly connected to the internet. To ensure that an attacker can always connect to the same IP camera, we used a static mapping of a set of 10 consecutive public IP addresses to each IP camera (except for one camera that was directly connected to the internet with one public IP address only). For example, observation point 1 proxies a set of 10 consecutive public IP addresses assigned to IP camera 1 with the help of communication control unit, observation point 2 proxies another set of 10 consecutive public IP addresses assigned to IP camera 2 with the help of communication control unit, and so on.

4.2 Experiment Overview

In this research, two types of decoy cameras were set up to observe peeping in IP cameras. The first one was a decoy camera (a URL reflection type decoy camera) that displays a hand-written

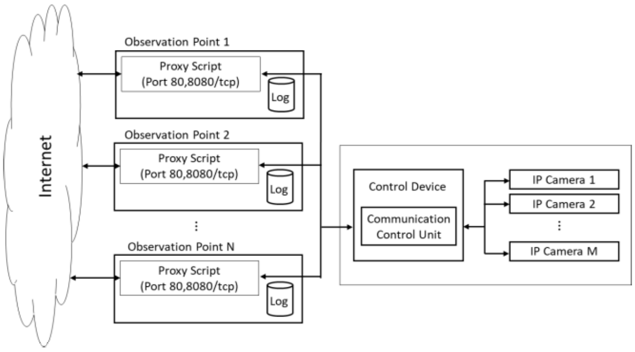


Fig. 1 Decoy camera network.

Table 1 URL decoy camera observation experiment.

IP Camera	Made in …	Authentication Feature	ID / Password	IP Address	Operational Functions	Observation Period	Observation Days
A	Japan	No		10	Yes	Jun.11〜 Jul.23, 2017	43 Days
B	Taiwan	Yes	admin/*****	10	No	Jun.11〜 Jul.23, 2017	43 Days

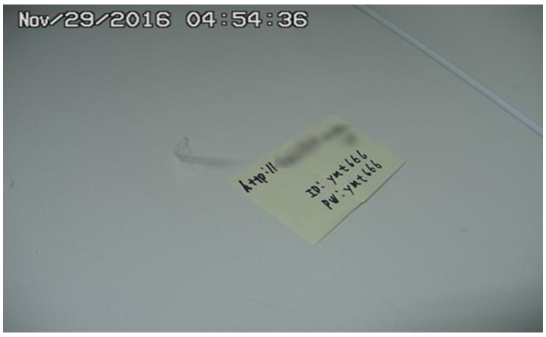


Fig. 2 Image from URL revealing camera A.

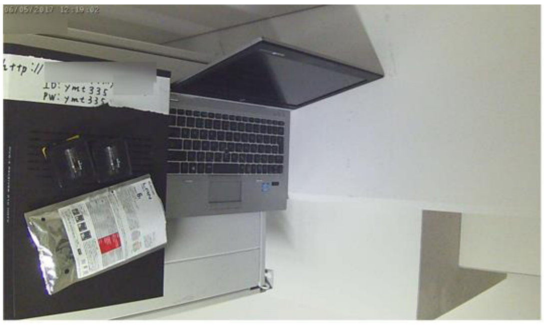


Fig. 3 Image from URL revealing camera B.

URL address and an ID/password for accessing that URL. In case of paper [3], the observation of access by the decoy camera with an ID/password was conducted, however in our research the URL is also reflected. In this URL, a script for basic authentication was run, and it was set to refuse login with any ID/password so that login challenges could be observed. Two cameras were used as URL reflection type decoy camera, each with a different ID/password set. **Table 1** shows the equipment used.

Camera A was set to allow viewing of camera images without authentication, and Camera B was set to be able to access the video with the camera's default ID/password. Each camera was assigned 10 consecutive IP addresses as the observation points, and the traffic on 80/TCP and 8080/TCP from these points was set to be relayed to Cameras A and B.

Figures 2 and **3** show the images from the URL reflection type decoy camera A and the URL reflection type decoy camera B, respectively. Both of the URLs are the same, but the ID/password is different, so that it is possible to determine which camera was viewed by the peeping host who input that particular ID/password.

In the second observation experiment, we prepared a room for observation simulating a family living-room where the image is expected to change, because the first type of observational setup did not attract continuous peeping as the URL reflection type decoy camera is not much attractive for peeping toms if the camera is only showing the URL and ID/password. Therefore, another study with a decoy camera setup (living-room decoy camera) was used to simulate a real life living-room view (**Fig. 4**).

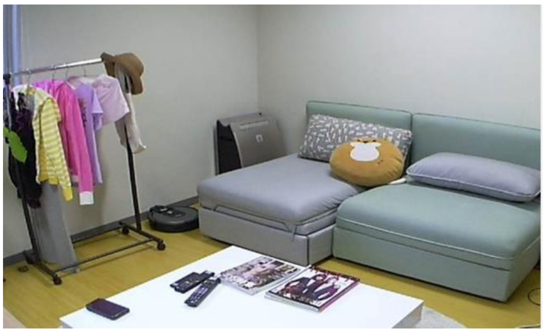


Fig. 4 Image from living-room decoy camera.

Table 2 Living-room decoy camera observation experiment.

IP Camera	Made in …	Authentication Feature	ID/ Password	IP Address	Operational Functions	Observation Period	Observation Days
A	Japan	No		10	Yes	Oct. 06〜 Nov.25, 2017	51 Days
C	Japan	No		10	Yes	Oct.06〜 Nov.25, 2017	51 Days
D	Japan	No		10	Yes	Oct.06〜 Nov.25, 2017	51 Days
E	Japan	No		10	No	Oct.06〜 Nov.25, 2017	51 Days
F	China	Yes	admin/***	1	Yes	Sept.21〜 Nov.25, 2017	66 Days

For the observation room, we used the home network test bed environment proposed in the paper [18]. For the living-room decoy camera, five cameras shown in **Table 2** were used.

The living-room decoy camera A was the same device as the URL reflection type decoy camera A. Living-room Decoy Cameras A, C, D, E, were set such that they could be browsed without authentication and living-room decoy camera F was set to allow login with default ID/password. In the living-room cameras A and C to E, 10 consecutive IP addresses were used and the communication addressed to 80/TCP and 8080/TCP was set to be transferred to those cameras. Whereas, the living-room decoy camera F does not get traffic transferred to it from the proxy, and therefore, can only be viewed via 1 IP address on 80/TCP port.

4.3 Experiment Results

4.3.1 Observation Results from URL Reflected Decoy Cameras

Figure 5 shows the transition of the number of hosts that sent the HTTP request to the URL reflection type decoy camera. Access to Camera B was almost constant, but access increased rapidly on Camera A on the third day of observation. When this case was investigated, many accesses using Insecam as a referrer were confirmed after June 13. Therefore, when we actually accessed Insecam website, the video of the camera posted there on June 13 was confirmed that explained the sudden increase in the number of peeping hosts.

Table 3 below shows the number of hosts that sent HTTP requests, the number of hosts that succeeded in authentication, the number of hosts that acquired camera view (peeped), and the number of hosts that operated the camera, as observed by the URL reflection type decoy camera experiment. Since the camera A is set to allow access to the image without authentication, the column for successfully authenticating hosts is not applicable (as marked by a diagonal line). Similarly, as camera B does

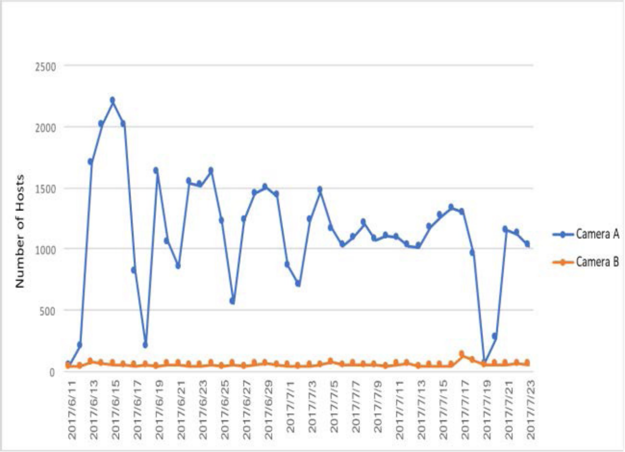


Fig. 5 Trend of hosts accessing URL reflective decoy camera.

Table 3 URL reflection type decoy camera observation results.

Camera	Request Hosts	Successfully Authenticating Hosts	Camera View Hosts	Camera Operations Hosts
A	27,116		25,585	499
B	1,280	4	0	

not have a camera operation function, the number of hosts that operated camera is not applicable (diagonal line).

Furthermore, the host that peeped in means "the host that sent the request to acquire a video or image of the camera view". Therefore, in case of Camera A that got exposed on Insecam, quite a large number of hosts (over 94%) targeted this camera for peeping, i.e., 85 hosts peeped out of the 27,116 hosts that sent the HTTP request. Among these peeping hosts, 1.7% (449 hosts) were observed as controlling the operation of the camera. We confirmed this from the fact that Web-UI browser access of Camera A was used in order to access the camera functions. For the camera B, 4 hosts out of the 1,280 hosts that sent the request succeeded in logging in, but there was no host that acquired the camera video, and no peeping attempt aimed at the device was observed.

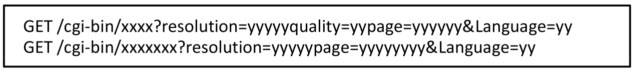
Next, **Table 4** shows further peeping attempts for the reflected URL in the URL reflection type decoy cameras by a number of access hosts attempting logins using the domain of the reflected URL rather than the direct input of the IP address. It also shows the number of hosts that entered the ID/password set reflected in Camera A, and the number of hosts that entered the ID/password set reflected in Camera B. Of the 583 hosts that accessed the URL, 422 were the access hosts that used the domain of the URL, that is, the host whose content of the HTTP header of the HTTP request matched the domain in the URL. Since this URL has not been disclosed outside this experiment, it is highly likely that access using that domain was made with visual observation of the reflected background information from the camera image. In addition, 217 hosts trying to login to the URL and entering the ID/password reflected on the camera A were observed. Therefore, we can say that certain number of people took the next action such as further accessing the reflected URL after reading the reflected ID/password information from the camera A video.

Table 4 Access to the URL displayed on the camera.

Number of Hosts that Requested Access	Hosts that Requested Access via URL	Number of Hosts that Attempted Login	Hosts that Requested Access using Info from Camera A	Hosts that Requested Access using Info from Camera B
583	422	235	217	0

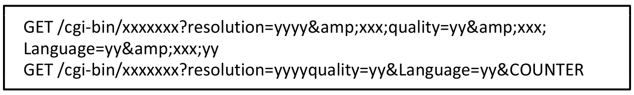
Table 5 Living-room decoy camera observation results.

Camera	Number of Hosts that Requested Access	Hosts Successfully Authenticated	Hosts that Browsed Camera	Hosts that Operated Camera
A	1,755		33	8
C	1,998		66	18
D	1,806		13	1
E	1,749		4	
F	876	51	32	6



```
GET /cgi-bin/xxxx?resolution=yyyyyquality=yypage=yyyyyy&Language=yy
GET /cgi-bin/xxxxxxx?resolution=yyyyypage=yyyyyyyy&Language=yy
```

Fig. 6 Image acquisition request to camera A by the browser.



```
GET /cgi-bin/xxxxxxx?resolution=yyyy&amp;xxx;quality=yy&amp;xxx;
Language=yy&amp;xxx;yy
GET /cgi-bin/xxxxxxx?resolution=yyyyquality=yy&Language=yy&COUNTER
```

Fig. 7 Automatic image acquisition request to camera A.

4.3.2 Observation Results from Living-room Decoy Cameras

Table 5 shows the number of hosts that sent HTTP requests, the number of hosts that succeeded in authentication, the number of hosts that browsed the camera (peeped), and the number of hosts that operated cameras, as observed by the living-room decoy camera experiment. Similar to Table 3, here also for the cameras that can access the video without authentication, the column for the number of hosts successfully authenticated was not applicable (diagonal line). Similarly, for the camera with no camera operation function, the number of operated hosts was not applicable (diagonal lines) either. Although none of the living-room decoy cameras were listed on Insecam website, still multiple peeping accesses were observed.

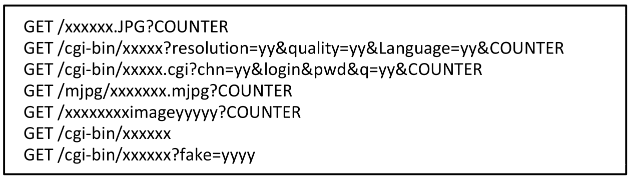
4.3.3 Peeping Characteristics

The peeping characteristics observed with each IP camera device are described below. In Figs. 6 to 13, a part of the character string in the request has been masked for security and privacy purposes.

Camera A

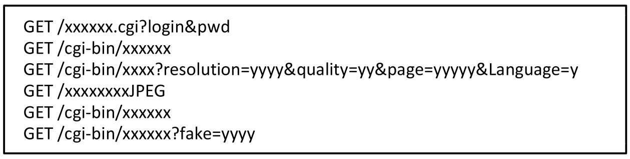
When this device is accessed by a general web browser, the video or image being captured is acquired with the requests as shown in **Fig. 6**. We benchmarked this pattern as the normal case when accessed by a general web browser. Among the hosts who peeped into camera A, some hosts acquired images multiple times with the requests as shown in **Fig. 7**.

On comparing the two requests, it can be seen that the request shown in Fig. 7 is very different from that of the browser access (Fig. 6) because it includes the character strings such as amp and COUNTER, and the argument page is not added. From this, it is



```
GET /xxxxxx.JPG?COUNTER
GET /cgi-bin/xxxxx?resolution=yy&quality=yy&Language=yy&COUNTER
GET /cgi-bin/xxxxx.cgi?chn=yy&login&pwd&q=yy&COUNTER
GET /mjpg/xxxxxxx.mjpg?COUNTER
GET /xxxxxxxximageyyyyy?COUNTER
GET /cgi-bin/xxxxxx
GET /cgi-bin/xxxxxx?fake=yyyy
```

Fig. 8 Request for automatic search of multiple types of cameras.



```
GET /xxxxxx.cgi?login&pwd
GET /cgi-bin/xxxxxx
GET /cgi-bin/xxxx?resolution=yyyy&quality=yy&page=yyyyy&Language=y
GET /xxxxxxxxJPEG
GET /cgi-bin/xxxxxx
GET /cgi-bin/xxxxxx?fake=yyyy
```

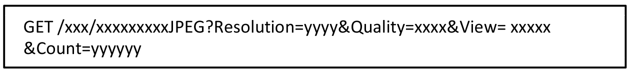
Fig. 9 Part of long-term access to camera A.

considered that this access uses a tool or script that acquires images automatically. Hence it can be concluded that there is a host that targets a specific manufacturer's equipment (IP camera) and performs peeping and image acquisition automatically.

Furthermore, as shown in **Fig. 8**, in addition to the image acquisition request of the relevant device, a host was observed sending a request to acquire the image of the IP camera of other manufacturers. This host sends similar requests to multiple IP addresses and it seems that many IP addresses are being tried to access for collecting images from various IP cameras.

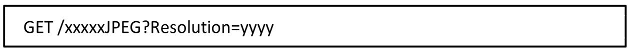
In addition to above patterns, a host that peeps for a long time was also observed. The access flow of this host is shown in **Fig. 9**.

First, a request (shown in the first line of Fig. 9) was transmitted 18 times in total by changing the value of the argument "pwd". However, it seems that these requests are targeted to other IP cameras because this device returns "404 Not Found" error message. After 1 minute 36 seconds, the second line request is sent twice, and the captured image of this device is acquired. Since this request is obviously different from the request observed in Fig. 6, it is considered to be an automated access by a tool/script or the like. 52 seconds later, the video is acquired with the request on the third line. At almost the same time we observe requests of JavaScript, CSS, image files etc., and since "User–Agent" is also a relatively new version of the actual browser, therefore It seems that some human has accessed using a web browser. After 3 minutes 17 seconds, image acquisition is performed again with the request of lines 4 and 5. 14 seconds later, the request on line 6 was intermittently transmitted at 1–3 second intervals with a different "User-Agent" than before. The value entered in the argument fake was different each time, but the intention to add the argument is not clear because the camera display image at the time of transmission is acquired regardless of the value of fake and whether or not it exists. The image acquisition by this request was observed intermittently until reaching 42 hours. When analyzing the access flow of this host, it seems that the requests in lines 1 to 5 are for searching the camera first, and continuous image collection is started when the camera is found. Moreover, in the camera A, 8 hosts were observed operating the camera in addition to viewing the images. Both hosts include requests for JavaScript, CSS, image files, etc. that are generated by access using a browser, and therefore, it seems that humans accessed the camera with a browser and operated the camera. However, the two hosts sometimes were observed to send automated image acquisition requests before and after an access by a browser. Thus, performed peeping by combining an automated access using a tool and checking the video using a browser.



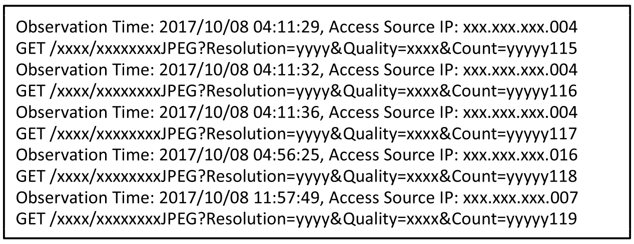
```
GET /xxx/xxxxxxxxxJPEG?Resolution=yyyy&Quality=xxxx&View= xxxxx
&Count=yyyyyy
```

Fig. 10 Image acquisition request to camera C by browser.



```
GET /xxxxxJPEG?Resolution=yyyy
```

Fig. 11 Automatic image acquisition request to camera C.



```
Observation Time: 2017/10/08 04:11:29, Access Source IP: xxx.xxx.xxx.004
GET /xxxx/xxxxxxxxJPEG?Resolution=yyyy&Quality=xxxx&Count=yyyyy115
Observation Time: 2017/10/08 04:11:32, Access Source IP: xxx.xxx.xxx.004
GET /xxxx/xxxxxxxxJPEG?Resolution=yyyy&Quality=xxxx&Count=yyyyy116
Observation Time: 2017/10/08 04:11:36, Access Source IP: xxx.xxx.xxx.004
GET /xxxx/xxxxxxxxJPEG?Resolution=yyyy&Quality=xxxx&Count=yyyyy117
Observation Time: 2017/10/08 04:56:25, Access Source IP: xxx.xxx.xxx.016
GET /xxxx/xxxxxxxxJPEG?Resolution=yyyy&Quality=xxxx&Count=yyyyy118
Observation Time: 2017/10/08 11:57:49, Access Source IP: xxx.xxx.xxx.007
GET /xxxx/xxxxxxxxJPEG?Resolution=yyyy&Quality=xxxx&Count=yyyyy119
```

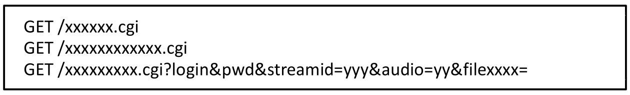
Fig. 12 Part of group of accesses appears to be from the same person.

Camera C

In camera C, it was confirmed that the image was acquired by a request (**Fig. 11**) different from the request for image acquisition using a browser (**Fig. 10**).

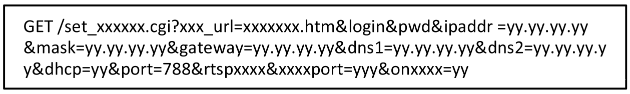
This request is significantly different from the request by the browser in that some arguments are not assigned, and therefore it seems that the request for access is by a tool or the like that automatically collects images. In addition, one host sent the request pattern as shown in Fig. 11, intermittently between October 21 and November 19, acquired a total of 2,528 images continuously, and performed long-term peeping access. Of the 66 IP addresses that peeped, 18 IP addresses operated the camera, and 9 IP addresses were concentrated in comparatively close address range within the same AS [19]. A detailed analysis of the access from these 9 IP addresses showed that the "User-Agent" used to acquire the images was the same, and the iPhone web browser Safari [20] was used. In addition, "GET / apple - touch - icon-icon.png" has been observed, and this request is an icon image acquisition request that iPhone uses to create a site shortcut on the home screen, and this request is unique to iPhone or iPad. From these facts, there is a high possibility that access from 9 IP addresses is using the iPhone. A part of such access from these 9 IP addresses is shown in **Fig. 12**.

In the case of access by the browser in the camera C, the display image is acquired every 3 seconds by the request shown in Fig. 10, and each time the argument Count of the request increases by 1 until the page is reloaded. In Fig. 12 with 5 image acquisition requests, the Count is incremented by 1 for each request, and it seems that the access is from the same client. However, the source IP address is different for the 1–3, 4th and 5th requests. Further analysis of this phenomenon shows that there is an interval of about 45 minutes between the 3rd and 4th accesses and about 7 hours between the 4th and 5th accesses. So in the process of accessing the device using the iPhone, it can be considered that this is due to the fact that the allocation of IP address



```
GET /xxxxxx.cgi
GET /xxxxxxxxxxxx.cgi
GET /xxxxxxxxx.cgi?login&pwd&streamid=yyy&audio=yy&filexxxx=
```

Fig. 13 Part of peeping access using vulnerability.



```
GET /set_xxxxxx.cgi?xxx_url=xxxxxxx.htm&login&pwd&ipaddr =yy.yy.yy.yy
&mask=yy.yy.yy.yy&gateway=yy.yy.yy.yy&dns1=yy.yy.yy.yy&dns2=yy.yy.yy.y
y&dhcp=yy&port=788&rtspxxxx&xxxxport=yyy&onxxxx=yy
```

Fig. 14 Part of port number change request.

has changed due to a physical movement or time lapses. In other words, all of these accesses are expected to be by the same person using the same device.

Camera D

We also observed several peeks at camera D. The image acquisition request by the browser of the camera D is the same as the request of the camera C (Fig. 10). Camera D observed several peeping accesses, but most of them were accesses to obtain images automatically using the same tools as in Fig. 11. In addition, one host was observed operating the camera. This host also operated camera C.

Camera E

Camera E also observed a group of requests (Fig. 8) to search for multiple cameras, similar to those observed with camera A. In addition, no automated access or long-term access targeting only the relevant device was observed.

Camera F

In camera F, 32 out of 51 hosts that successfully logged in were observed peeping. This camera had a known vulnerability in which the default ID/password was leaked by sending a specific request to camera F device, and hosts that peeped exploiting such vulnerability were observed. **Figure 13** shows a part of the access flow.

An ID/password is acquired by sending the request of lines 1 and 2. We noticed from the "User-Agent" that these requests are presumed to be accessed by tools etc. other than the browser. After that, this host logged in based on the obtained information and browsed the video using the request shown in line 3. At the time of this access, besides reading CSS and image files etc. similar to the browser access pattern, it was also observed that the "User-Agent" acquired the video using the concerned camera-specific plug-in of Internet Explorer [21]. This indicates that a human accessed via the browser and peeped in using the plug-in.

Moreover, some of the other hosts obtained videos automatically instead of using a browser after obtaining an ID/password, and host peeping at a specific device was also observed. Furthermore, in addition to peeping, we also noticed a request sent (as shown in **Fig. 14**) and observed an attack from one host to change the port number for video delivery from 80/TCP to 788/TCP. If the intention was to exclude other intruders, then this action will also obstruct browsing by an authorized user. Therefore, this intention is questionable due to the effect of changing the settings in such a way.

5. Summary of Observations and Discussion

As a result of the first observational study, access to view the video of the URL reflection type decoy camera was observed from 25,585 IP addresses in 43 days. In particular, when one of the URL reflection type decoy cameras was posted on Insecam two days after starting the experiment, the observed access immediately increased from 1,701 IP addresses per day to more than 20,000 times, with 94% peeping. From this, it can be stated that a large amount of cyber peeping occurred because of the Insecam website. On the other hand, though the other URL reflecting camera was accessed by a few hosts, we did not observe a peeping host acquiring its camera view. Thus we can say that the weakness of login authentication and type of device can make a difference in the number of peeps. Moreover, 217 IP addresses, i.e., 0.8% of total hosts who acquired video of reflected URL decoy cameras attempted login with the reflected information. Though a very smaller number than expected, however, we were able to observe the human element by confirming hosts accessing the URL displayed on the URL reflection type decoy camera A and executing further (after viewing the URL information from the accessed camera) by logging-in with that reflected ID/password.

In the case of the second observational study, even though none of the five living-room decoy cameras were posted on Insecam, we still confirmed that peeping access from multiple hosts appears to be due to the nature of living-room image triggering curiosity. We observed a host that periodically acquires a video image of the camera automatically with a dedicated tool specialized for image acquisition. In addition, access to operate cameras and access for a long time (as much as 42 hours for a single camera) were also observed. Furthermore, we observed an attacker who peeped at the camera after breaking through the authentication and an attack that changed the TCP port number of the user interface for viewing.

As a result of analyzing these accesses, we were able to study the real world situation of different methods of peeping such as automated access by hosts that search for cameras and acquire images, or hosts that automatically acquire video targeting specific devices. We also observed hosts where a human seems to conduct further peeping after reading the reflected hand-written information and logging into the reflected URL successfully. Moreover, cyber peeping by exploitation of known vulnerabilities of cameras with default or weak authentication and operating them was also confirmed. This study also highlighted the importance of strengthening security parameters in IP cameras so that they can avoid unnecessary exposure to websites collecting and showing images of easily accessible IP cameras. Furthermore, through this study we were able to observe several automated attack patterns for cyber discovery and peeping purposes that can be used as detection rules in network-based intrusion detection systems (NIDS).

6. Ethical Considerations

Although websites like "Insecam" have succeeded to demonstrate there are many insecure IP cameras on the Internet that have no password set, our study further revealed that there are indeed a considerable number of unwanted accesses to such insecure cameras. In order to improve the situation, we tried to inform two main stakeholders: end-users and IP camera vendors. We consider that publishing our work is one of the main channels to inform the end users. It is worth noting that the work acquired some media attention and the experimental results were introduced in several TV programs and news by the national broadcasting station [22], [23], from which we believe that we have somewhat contributed in improving public awareness on the risks and dangers of IP cameras. Moreover, we had conversations with several major IP camera vendors to inform them of the increasing cyber threats. One of the vendors now adopts an improved security mechanism. In order to minimize the possible harm to the vendors of the IP cameras, we anonymized the vendor names and tried to redact identifiable information as much as possible from the analysis results. Though we have taken precautions, we believe that the benefits this study brings would exceed the harm that it might have caused.

7. Conclusion and Future Work

In this study, two scenarios for observation of peeps were tested. First one by two URL reflecting type decoy cameras and second by five living-room cameras. The results of this study provided us a better understanding on peeping methods through IP cameras, both via automation and human involvement. The techniques helped us in understanding the risks and dangers of using IP cameras with no/default or weak access authentication. This study also showed how public websites showing easily accessible IP cameras can drastically increase number of peeps into those IP cameras in the real world. Similarly, we were able to confirm that secondary information in viewable areas in front of IP cameras or reflected background information (URL and ID/password as bait via the decoy camera in our case) in IP camera images can be used by a peeping tom to further exploit and gain additional access, thereby exposing the dangers of using IP cameras. Furthermore, through this study we were able to observe several automated attack patterns for the discovery and peeping purposes that can be used as detection rules in the network-based IDS. Such protection by NIDS would be particularly useful when deployed at the gateway of heterogeneous networks, such as university networks, where IP cameras are massively used and owned by different individuals. The network administrator can use these patterns to detect and if necessary, block suspicious accesses to the cameras while advising the owners to use stronger authentication methods.

As this study utilized different devices in two types of observational environments to study the actual state of peeping by humans and automated accesses, for future study work it will be interesting to further broaden this study by utilizing same devices in multiple observational environments. Additionally, since this study was limited to confirming access attempt (by rejecting access and logging attempt only) to reflected URL web server with the login info from the peeped decoy cameras, it would be interesting to further carry out a detailed behavioral study for observing the actions of peeping hosts after successfully accessing a reflected URL.

Acknowledgments Part of this research was carried out with the support of the Ministry of Education, Culture, Sports, Science and Technology's National University Reform Improvement Promotion Project. Part of the results of this research were obtained with the support of a research commissioned by the National Institute of Information and Communications Technology (NICT)'s Research and development for practical application of web mediated attack countermeasure technology. We would also like to thank the production staff of "NHK Special – Home appliances are being targeted" program for their cooperation in the observation experiment.

References

[1] Insecam, available from ⟨http://www.insecam.org/⟩ (accessed 2019-02-25).

[2] Pa, Y.M.P., Suzuki, S., Yoshioka, K., Matsumoto, T., Kasama, T. and Rossow, C.: IoTPOT: Analyzing the Rise of IoT Compromises, *Proc. 9th USENIX Workshop on Offensive Technologies* (*WOOT'15*) (2015).

[3] Suzuki, S., Pa, Y.M.P., Ezawa, Y., Tie, Y., Nakayama, S., Yoshioka, K. and Matsumoto, T.: Improving IoTPOT for Observing Various Attacks Targeting Embedded Devices, *IEICE Technical Report*, *ICSS2015-47*, Vol.115, No.488, pp.1–6 (2016).

[4] Ezawa, Y., Tamiya, K., Nakayama, S., Tie, Y., Yoshioka, K. and Matsumoto, T.: An Analysis of Attacks Targeting WebUI of Embedded Devices by Bare-Metal Honeypot, *Proc. Computer Security Symposium 2017* (*CSS2017*), pp.211–217 (2017).

[5] Guarnizo, J.D. Tambe, A. Bhunia, S.S. Ochoa, M., Tippenhauer, N.O., Shabtai, A. and Elovici, Y.: SIPHON: Towards scalable high-interaction physical honeypots, *Proc. 3rd ACM Workshop on Cyber-Physical System Security* (*CPSS'17*), pp.57–68 (2017).

[6] Stelma, J.: *Securing the Home Network*, Master thesis, Eindhoven University of Technology (2015), available from ⟨https://pure.tue.nl/ws/files/47037062/799535-1.pdf⟩ (accessed 2019-03-18).

[7] Bogaard, C.V.: Security analysis of cloud-based video cameras (2017), available from ⟨https://pdfs.semanticscholar.org/17cb/8f89320c9ca31d93b0e9e8ede68b6d03ff74.pdf⟩ (accessed 2020-03-30).

[8] Smith, M.: Peeping into 73000 unsecured security cameras thanks to default passwords, CSO Online (2014), available from ⟨https://www.csoonline.com/article/2844283/peeping-into-73-000-unsecured-security-cameras-thanks-to-default-passwords.html⟩ (accessed 2019-03-18).

[9] How safe are home security systems? An HP study on IoT security (2015).

[10] Vivotek IP Cameras - RTSP Authentication Bypass: HelpSystems (2013), available from ⟨http://www.coresecurity.com/advisories/vivotek-ip-cameras-rtsp-authentication-bypass⟩ (accessed 2019-02-17).

[11] Donohue, B.: Urban surveillance camera systems lacking security, Blog, Kaspersky (2015), available from ⟨https://blog.kaspersky.co.jp/urban-surveillance-not-secure/7781/⟩ (accessed 2019-02-25).

[12] CVE-2017-5674: National vulnerability database, NIST (2017), available from ⟨https://nvd.nist.gov/vuln/detail/CVE-2017-5674⟩ (accessed 2018-12-20).

[13] Shodan, available from ⟨https://www.shodan.io/⟩ (accessed 2018-11-11).

[14] Censys, available from ⟨https://censys.io/⟩ (accessed 2019-01-25).

[15] Seals, J.: Cyber Actors Use Internet of Things Devices as Proxies for Anonymity and Pursuit of Malicious Cyber Activities, *Internet Crime Complaint Center* (*IC3*) (2018), available from ⟨https://www.ic3.gov/media /2018/180802.aspx⟩ (accessed 2019-03-21).

[16] Glastoph, Web Application Honeypot, available from ⟨https://github.com/mushorg/glastopf⟩ (accessed 2018-11-21).

[17] Dionaea low interaction honeypot, available from ⟨https://github.com/rep/dionaea⟩ (accessed 2018-11-21).

[18] Yang, Z., Xiong, J., Tie, Y., Tamiya, K., Nishida, S., Yang, D., Fujita, A., Yoshioka, K. and Matsumoto, T.: Observation and Analysis of Cyber Attacks in Home Network Testbed, *Proc. Computer Security Symposium 2017* (*CSS2017*) (2017).

[19] AS number assignments: Japan Network Information Center (JPNIC), available from ⟨https://www.nic.ad.jp/ja/ip/asnumber.html⟩ (accessed 2018-11-11).

[20] Safari - Apple Support, available from ⟨https://support.apple.com/ja_JP/downloads/safari⟩ (accessed 2018-11-14).

[21] Internet Explorer - Microsoft Download Center, available from ⟨https://www.mic rosoft.com/ja-jp/download/internet-explorer.aspx⟩ (accessed 2018-11-14).

[22] Home Gadgets at Risk: NHK Documentary, *Season 2018*, Episode 4 (2018).

[23] Home Gadgets at Risk: Science ZERO, NHK Educational TV

(2017), available from ⟨https://www2.nhk.or.jp/archives/chronicle/pg/page010-01-01.cgi?recId=0001000000000000%4000000000000000000000000%2D50%2D21%2D4800000000000000000000⟩ (accessed 2019-03-18).

Kazuki Tamiya received his master's degree from Yokohama National University in 2019. His research interests cover computer and network security including malware analysis and IoT security.

Aamir H. Bokhari received his M.Sc. degree in cybersecurity from University of Dallas, an excellence center of USA's department of homeland security (DHS) and national security agency (NSA). He is currently pursuing his Ph.D. in Informatics from Yokohama National University. He is also working on EU-Japan project of multi-layered security (M-Sec) solutions for smart cities. His background and research interests cover computer and network security, IoT security, cybersecurity, data and information security, cloud security, IT and telecommunications.

Yuta Ezawa received his master's degree from Yokohama National University in 2018. His research interests cover computer and network security including malware analysis and IoT security.

Sou Nakayama received his master's degree from Yokohama National University in 2018. His research interests cover network security including malware analysis and IoT security.

Ying Tie received her Ph.D. from Yokohama National University in 2018. Her research interests cover the area of network security including malware analysis and IoT security. She is currently a researcher at Toyota Motor Corporation, and is working on motor vehicle safety and security.

Rui Tanabe received his Ph.D. in information sciences from Yokohama National University in 2017. After working at Yokohama National University as a researcher, Dr. Tanabe is currently working as a project assistant professor at Yokohama National University. His research interests include information security and network security. He received the Yamashita Memorial Research Award from IPSJ in 2017.

Akira Fujita received his B.A., M.Sc. and Ph.D. in information sciences from Yokohama National University in 2008, 2009, and 2012, respectively. After working at Yokohama National University and National Institute of Informatics as a researcher, Dr. Fujita is currently a project assistant professor at Yokohama National University. His research interests include network security, natural language processing and cognitive science.

Katsunari Yoshioka is an Associate Professor at Yokohama National University since 2011. Before that, he was a researcher at National Institute of Information and Communications Technology, Japan. His research interests cover wide area of system security and network security including malware analysis and IoT security. He received the commendation for science and technology by the minister of MEXT, Japan in 2009, the award for contribution to Industry-Academia-Government Collaboration by the minister of MIC, Japan in 2016, and the Culture of Information Security Award in 2017.

Tsutomu Matsumoto is a professor of the Faculty of Environment and Information Sciences, Yokohama National University, and directing the Research Unit for Information and Physical Security at the Institute of Advanced Sciences. Prof. Matsumoto also serves as the Director of the Cyber Physical Security Research Center (CPSEC) at the National Institute of Advanced Industrial Science and Technology (AIST). Starting from Cryptography in the early '80s, Prof. Matsumoto has opened up the field of security measuring for logical and physical security mechanisms. He received a Doctor of Engineering degree from the University of Tokyo in 1986. Currently, he is interested in research and education of Embedded Security Systems such as IoT Devices, Cryptographic Hardware, In-vehicle Networks, Instrumentation and Control Security, Tamper Resistance, Biometrics, Artifact-metrics, and Countermeasure against Cyber-Physical Attacks. He serves as the chair of the Japanese National Body for ISO/TC68 (Financial Services) and the Cryptography Research and Evaluation Committees (CRYPTREC) and as an associate member of the Science Council of Japan (SCJ). He was a director of the International Association for Cryptologic Research (IACR) and the chair of the IEICE Technical Committees on Information Security, Biometrics, and Hardware Security. He received the IEICE Achievement Award, the DoCoMo Mobile Science Award, the Culture of Information Security Award, the MEXT Prize for Science and Technology, and the Fuji Sankei Business Eye Award.