MODELS AND ALGORITHMS FOR PROTECTION AGAINST THREATS THAT VIOLATE THE INTEGRITY OF VIDEO INFORMATION

Gafurov A.A.
"Cybersecurity Center" State Unitary Enterprise, Head of UZSERT Department, Uzbekistan
Abdusalomov A.A.
Master's degree, Faculty of Cyber-Security,Tashkent University of Information Technologies named after Muhammad al-Khwarizmi, Uzbekistan

https://doi.org/10.5281/zenodo.7445078

Abstract. *Over the last decade, video surveillance systems have become a part of the Internet of Things (IoT). These IP-based surveillance systems now protect industrial facilities, railways, gas stations, and even one’s own home. Unfortunately, like other IoT systems, there are inherent security risks which can lead to significant violations of a user’s privacy. In this review, we explore the attack surface of modern surveillance systems and enumerate the various ways they can be compromised with real examples. We also identify the threat agents, their attack goals, attack vectors, and the resulting consequences of successful attacks. Finally, we present current countermeasures and best practices and discuss the threat horizon. The purpose of this review is to provide researchers and engineers with a better understanding of a modern surveillance systems’ security, to harden existing systems and develop improved security solutions.*

Keywords: *IP cameras, video surveillance, IoT, network security, physical security*.

МОДЕЛИ И АЛГОРИТМЫ ЗАЩИТЫ ОТ УГРОЗ, НАРУШАЮЩИХ ЦЕЛОСТНОСТЬ ВИДЕОИНФОРМАЦИИ

Аннотация. *За последнее десятилетие системы видеонаблюдения стали частью Интернета вещей (IoT). Эти IP-системы видеонаблюдения теперь защищают промышленные объекты, железные дороги, автозаправочные станции и даже собственный дом. К сожалению, как и в других системах IoT, существуют неотъемлемые риски безопасности, которые могут привести к серьезным нарушениям конфиденциальности пользователя. В этом обзоре мы исследуем поверхность атаки современных систем наблюдения и перечислим различные способы их компрометации на реальных примерах. Мы также определяем агентов угроз, цели их атак, векторы атак и результирующие последствия успешных атак. Наконец, мы представляем текущие контрмеры и лучшие практики и обсуждаем горизонт угроз. Цель этого обзора — предоставить исследователям и инженерам лучшее понимание безопасности современных систем наблюдения, укрепить существующие системы и разработать улучшенные решения для обеспечения безопасности.*

Ключевые слова: *IP-камеры, видеонаблюдение, IoT, сетевая безопасность, физическая охрана.*

Introduction. A threat agent may learn the target device’s credentials or vulnerabilities by the use of reverse engineering (RE). Reverse engineering is typically performed off-site using the same hardware/software used by the victim. RE is the process of analyzing compiled code or hardware to identify system’s components and their interrelationships. During this process, vulnerabilities and even hard-coded credentials can be discovered.

One approach is to analyze the pre-compiled firmware provided by the manufacturer. In a Black Hat lecture, the authors focused on IP cameras which face the Internet and analyze them through firmware images supplied by the camera's vendors. The authors found zero-day vulnerabilities in digital surveillance equipment from various firms including D-Link Corp., Cisco Systems, Linksys, TRENDnet, and more with the use of existing tools. The analysis revealed serious security vulnerabilities such as administrative passwords, remote code execution vulnerabilities, and more. Another case was found in Sony's IPELA surveillance camera series. By performing RE on the firmware, researchers from Sec Consult found a backdoor via two hard coded root level credentials. These backdoors have also been discovered in other cameras and DVRs. The hard-coding of credentials may occur intentionally, or by mistake (e.g., a developer forgot to remove the credentials after testing).

Another approach of RE is to interface with the device via its Universal Asynchronous Receiver/Transmitter (UART) ports. These ports are typically inside the device's casing, and used by the manufacturer for debugging purposes. UART ports can be used to expose vulneabilites, gain access to the firmware, run foreign applications, extract sensitive information, or upload custom firmware for further analysis. The authors analyzed several IoT devices including IP cameras, baby monitors thermostats, and doorbells for vulnerabilities. The authors presented a generic workflow in order to gain access to the software of IoT devices, run foreign applications, and extract secret information (i.e., credentials) with the UART ports. The authors reveal that many DVRs are vulnerable to an unauthenticated login disclosure and unauthenticated command injection via UART. Researchers connected to the UART and updated the firmware with code to find that the camera was running a vulnerable OpenSSL version (i.e., heartbleed) and discovered other vulnerabilities leading to remote code execution.

Materials. Video surveillance requires either a manual or automated way of reviewing the video content for events—for example, detecting live intrusions or locating suspects. Therefore, the domain of video analytics has been applied to minimize the human efforts in this task. In the case of large deployments, such as China's state surveillance system, automated methods are required. Some automated technologies include facial recognition, and object tracking. However, since most of these technologies rely on machine learning, they are susceptible to adversarial attacks. An adversarial attack is where a machine learning model is abused by either poisoning the model during training so that the mode will behave according to the attacker's will, crafting an input which will yield an unexpected output, or learning the training data or the model itself by observing the input–output relationship. Adversarial attacks on these technologies mean that an attacker may be able to evade detection, falsify the recognition of an object, or even cause a DoS attack by raising the technology's false positive rate.

A good example of an adversarial attack on Surveillance systems is the work done. There the authors generated colorful glasses rims, which, when worn, alters the identity of the individual in the perspective of the deep learning classifier monitoring the imagery. This attack can be used to not only evade detection, but impersonate individuals as well. Another attack on these systems is a DoS attack where the attacker spams the imagery with false positives—for example, by wearing clothes with crafted license plate images to overload traffic cams.

Another good example is where the attacker crafts adversarial images which are designed to consume significant resources. An attacker can cause a DoS attack by placing these 'sponge' samples in view of the camera to either slow down the device's processor until it becomes unresponsive or depleting the battery of remote cameras.

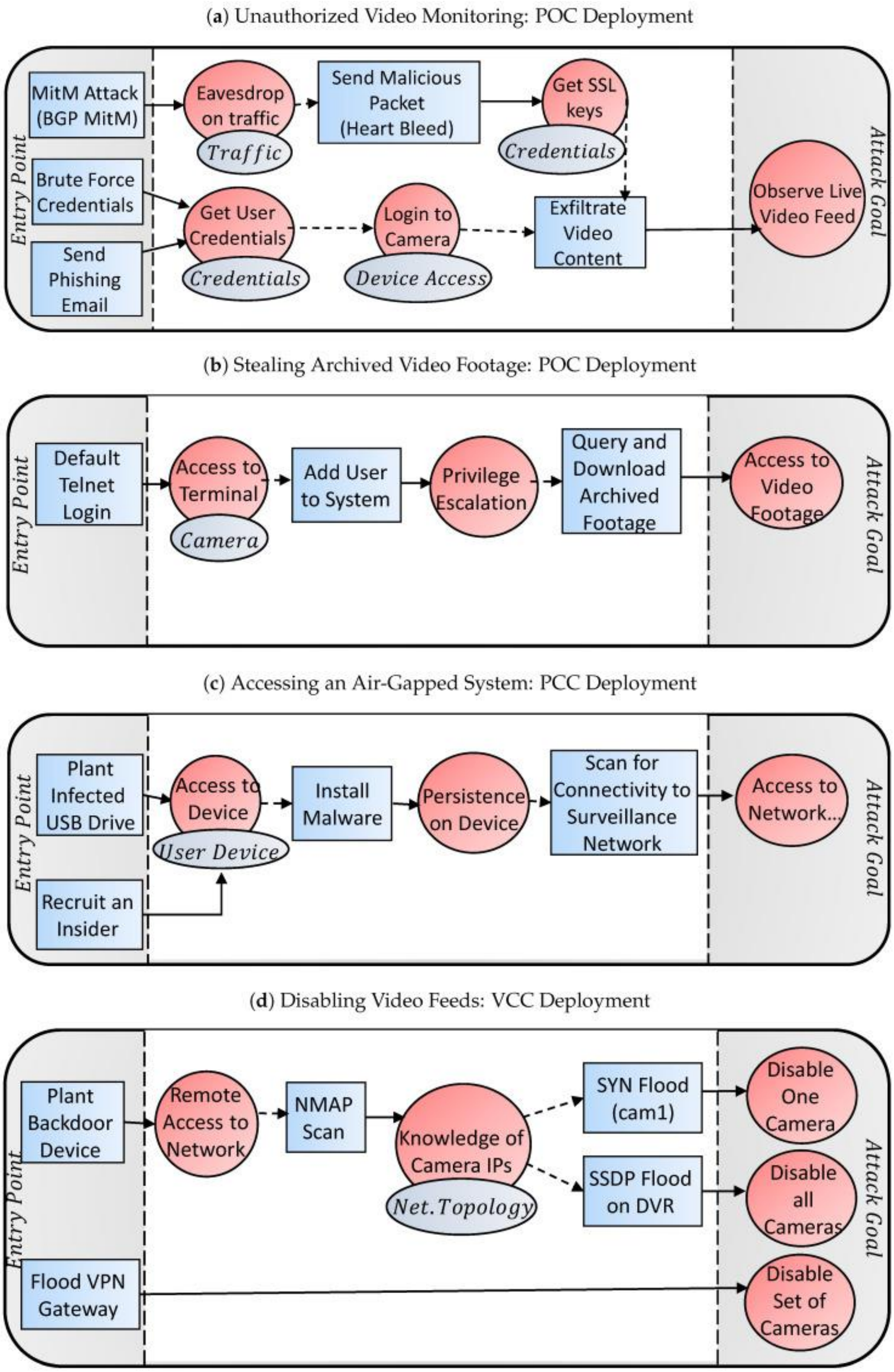
Another example is where AI-based surveillance systems setup to measure traffic can be fooled to reporting traffic jams or no traffic. This would give the attacker the ability to shape traffic to his needs, cause havoc, or block emergency routes to hospitals as an act of terrorism.

We suggest that an attacker may try to spam the system with millions of false alarms, burying important alerts and notifications from the response team's view. This can be accomplished by crafting adversarial images which contain the thousands of patterns that trigger the object detector. For example, a single picture containing numerous imperceivable patterns of weapons or faces.

Methods. The attacker may be able to watch/download live or pre-recorded video footage. Compared to compromising other IoT devices, this results in a significant privacy violation. The attacker could use the content to track people, observe their behaviors, find where valuables are stored, shoulder-surf to steal credentials, determine when to commit a crime, or blackmail an individual. Another concern is that the attacker will alter the contents to plant false evidence such as a prerecorded video loop, or use deep learning to insert an individual performing an activity, cover up an on-going crime, or permanently delete footage.

In some cases, an attacker may get implicit access to the video content through side-channel attacks. For example, video compression algorithms, such as H264, only send data when regions of the frames change. As a result, the bandwidth of the channel fluctuates in correlation to the motion in the camera's field of view. The authors demonstrate this concept on surveillance cameras using the CUMSUM algorithm on the data rates of the encrypted network traffic. CUSUM is a nonparametric algorithm which can detect anomalies in time series data.

Results. Consider an attacker who wants to view the video footage of a POC deployment with encrypted traffic. A few potential attack vectors are illustrated. A state actor may perform a BGP MitM routing attack, and cause all of the video surveillance traffic to pass through them first. This Figure 1 would give him access to the cameras' traffic enabling him to eavesdrop on the camera–server communications. Next, the attacker may exploit the Heartbleed vulnerability to get the SSL cryptographic keys and then decrypt the video traffic. A simpler way might be to get the camera's or DVR's login credentials by performing a brute-force login attack, or to send phishing emails to users of the system to have them unwittingly reveal their credentials. Once the credentials have been obtained, the attacker can access the camera and observe the live video feeds.



(**a**) Unauthorized Video Monitoring: POC Deployment

(**b**) Stealing Archived Video Footage: POC Deployment

(**c**) Accessing an Air-Gapped System: PCC Deployment

(**d**) Disabling Video Feeds: VCC Deployment

Figure 1. Example attack vectors on IP-based surveillance camera systems deployed with POC, PCC, and VCC topologies.

Open in a separate window

Stealing Archived Video Footage

Another scenario is the case where an attacker wants to blackmail an individual by obtaining sensitive video footage. In Figure 5d, we illustrate one possible attack vector where an agent gains access to the DVR's terminal by performing a dictionary brute-force login attack on one of the camera's telnet servers. Having access to a camera's terminal prompt, the attacker will add a user to the system giving himself elevated privileges to query the DVR for archived footage (e.g., using RTSP replay commands).

Accessing an Air-Gapped System

In the case of a PCC deployment, direct access from the Internet is impossible. However, this does not mean that the network is impervious to infiltration. In Figure 1c, we illustrate how an attacker can install malware on one of the user's devices, such as a viewing terminal (tablet, console, etc.) This can be accomplished by surreptitiously placing an infected USB drive in the area, or by recruiting an insider. Once the malware is installed by the threat agent, the malware will scan the surveillance network to identify all of the assets. At this point, the malware may perform automated actions designed by the attacker (e.g., disable camera at a certain time) or it may communicate with the attacker directly over the air-gap via bridgeware.

Disabling Video Feeds

An attacker can disable video feeds in many different ways. Let's assume that the target system has a VCC deployment, so access is either physical or via a VPN gateway. In Figure 1b, we show how an attacker can disable one, all, or a subset of cameras in this scenario. First, an attacker may plant a backdoor device to gain remote entry. For example, the attacker may arrive under the pretext of a repairman and secretively connect a Raspberry Pi to the network, and then remotely connecting to the Pi's Wi-Fi access point from the parking lot. Next, the attacker will have the Pi scan the network to reveal the IP addresses of the cameras and the DVR. Finally, single cameras can be disabled via a TCP SYN flooding attack, or all cameras can be disabled by exploiting a potential SSP flood vulnerability in the DVR. Alternatively, instead of coming locally to plant a backdoor device, the attacker can attack remotely by performing a flood attack (e.g., ISAKMP flood) on the network's site-to-site VPN gateway. As a result, the set of cameras on one side of the VPN tunnel will be disconnected from the DVR on the other side of the tunnel.

Countermeasures and Best Practices

In the following section, we review existing countermeasures and best practices which can be used to protect modern surveillance systems.

Intrusion Detection and Prevention Systems

Basic cyber defense should be considered in every computer network. For example, to detect and prevent malware infections, anti-virus software should be installed on the user terminals and DVRs. In non-distributed POC topologies, a strict firewall should be deployed to pass the minimal network traffic required to use the system (e.g., block telnet, ICMP 'ping' packets, etc.).

In case the adversary evades the firewall, a network intrusion detection system (NIDS) can be used to detect malicious traffic patterns. In this case, free rule-based NIDS, such as Snort and Suricata, or commercial software can be used.

The authors propose a lightweight NIDS based on an ensemble of autoencoder neural networks, and evaluate it on a video surveillance system. Kitsune uses incremental statistics to track millions for network channels and then uses these summaries to extract a feature vector for every packet. The feature vector captures a snapshot of the network in the context of the given packet. The anomaly detection model (KitNET) is trained in real time and on site. The model uses autoencoders to detect anomalies. This is accomplished by training the autoencoder to recontruct feature vectors of benign traffic, and then flagging packets whose reconstruction errors are statistically high. KitNET is comprised of an ensemble of small autoencoders, where each one covers a different correlated subspace (set of features), and a single autoencoder which monitors the ensembles' recontruction errors to cover cross-subspace anomalies. The authors evaluated Kitsune on a commercial IP-based surveillance system and successfully detected two types of reconnaissance attacks, three types of man-in-the-middle attacks, and three types of DoS attacks.

Configurations and Encryption

One should carefully review the configurations of the cameras, routers, terminals, and DVR. For example, weak or default passwords should be changed, and different passwords should be used among different devices if possible. Moreover, APIs and other similar features should be disabled if not needed. One should also periodically check for new CVEs that the software/firmware of all devices are up to date.

It is also important to enable secure communication wherever possible and not just on the video stream itself. This is because an attacker would still be able to hijack a video stream (e.g., redirect or pause video in an RTSP stream) or compromise the DVR through leaked credentials. We have also noted that several vendors of DVR software use self-signed SSL certificates (a common default setting). This is a significant risk and should be corrected since it enables an attacker to perform an SSL man in the middle redirection attack.

To ensure the integrity of the video content, digital watermarks (DW) can be used. A DW is a subtle signal hidden within the imagery (pixels) which is corrupted if the image is tampered with. In this way, the viewer can verify that every frame is legitimate. The advantage of using DWs is that they do not require changes to the video or networking protocols; however, they may add noise to the image and get corrupted in the presence of video compression.

Restrict Physical Access

The most basic perimeter defense is to restrict physical access to the system's assets. If possible, wiring should not pass through public areas, all networking equipment (switches, routers, etc.) should be protected under lock-and-key, and access to the system should be managed, logged, and monitored.

Defense against DoS Attacks

There are many protocols and vulnerabilities that can be abused to perform a DoS attack. As a result, there are many different defense mechanisms which can be deployed. Good protection involves the following steps: (1) detect the attack's initiation, (2) select the malicious/harmful packets, and (3) filter/log the detected packets. For the attack detection, machine learning and statistical methods can be used such as lightweight anomaly detection and many more.

Conclusion. We have identified two main emerging threats to IP-based video surveillance systems. The first is adversarial machine learning. Advanced machine learning techniques, mainly based on deep learning, are being researched and integrated within today's video surveillance systems for automating various tasks including: weapon detection, fire detection, in-store shopping, face recognition, and anomaly detection. In parallel, there has been an increase of research on adversarial machine learning meaning that these systems are vulnerable to attacks. The second emerging threat is how these systems are being infected and recruited into botnets, leading to attacks on the internal network (e.g., data exfiltration, spying or using the surveillance system for lateral movement) or on other external networks (e.g., DDoS, SPAM).

New attacks are constantly emerging. As a result, have noted in our review that a recent research trend has been securing surveillance systems has been the use of advanced anomaly detection. With anomaly detection researchers are able to identify man-in-the-middle-attacks, video injection, OS fingerprinting, fuzzing and ARP poisoning attacks, and DDoS attacks. Although anomaly detection is vulnerable to adversarial machine learning, previous works in the domain have mostly focused on attacking classifiers. The key difference between classifiers and anomaly detectors is that classifiers have a decision boundary built into the model (are closed-world) and anomaly detection models only capture the normal behaviors (are open-world). Therefore, more research is needed in understanding how an attacker can potentially craft a sample which is detected as normal while still achieving his or her goal of something that is abnormal.

Updating the software of such systems is also a challenging task since manufactures are focused on their next product, and in many cases do not have the capability of performing remote patching. Therefore, we believe future research should focus on providing an external continuous protection that can be easily updated with information on newly discovered attacks. One way to collect intelligence on emerging threats to surveillance systems is to use an advanced honeypot system. Moreover, by identifying emerging exploits, administrators can protect their systems before they get infected.

Finally, although in most cases the communication of advanced video surveillance systems is encrypted, the confidentially of entities can be compromised using side channel attacks. A vulnerability of encrypted video streams is that the compression algorithm (video codec) generates more data when there is motion. Researchers have shown how content can be inferred from these encrypted streams by learning/correlating network bandwidth patterns. The authors showed how to detect what a drone’s surveillance camera is looking at by monitoring the encrypted WiFi video stream, triggering a visual stimulus such as a flashing light, and detecting correlated bandwidth fluctuations in the wireless signal. Therefore, we suggest that future research should address the detection and elimination of these side channels.

In this article, we have reviewed the security of modern video surveillance systems. First, we presented a security overview of these systems along with their components and their deployments. Using this information, we identified the system’s attack surface comprising of its attack agents, vulnerabilities, actions, and consequences. We then used this information to exemplify several attack vectors. Having described the attacker’s capabilities, we then discussed recent research on countermeasures and best practices which can be implemented to better secure IP-based surveillance systems. Finally, we concluded the review with a discussion on the threat horizon, and suggested future work in this domain.

In summary, this article provided the reader with a greater understanding of the attack surface and recent advances made by both the attackers and defenders over the last ten years. We hope that this information will aid researchers and engineers in securing the surveillance systems of today and tomorrow.

REFERENCES

1. Cabasso J. Analog vs. IP cameras. Aventura Technol. Newsl. 2019;1:1–8. [Google Scholar]
2. Statista Security & Surveillance Technology Statistics & Facts. Technical Report. [(accessed on 21 August 2020)];2019 Available online: https://www.statista.com/topics/2646/security-and-surveillance-technology/
3. Mukkamala S., Sung A.H. Detecting denial of service attacks using support vector machines; Proceedings of the 12th IEEE International Conference on Fuzzy Systems; St. Louis, MO, USA. 25–28 May 2020; pp. 1231–1236. [Google Scholar]
4. Antonakakis M., April T., Bailey M., Bernhard M., Bursztein E., Cochran J., Durumeric Z., Halderman J.A., Invernizzi L., Kallitsis M., et al. Understanding the mirai botnet; Proceedings of the USENIX Security Symposium; Vancouver, BC, Canada. 16–18 August 2017. [Google Scholar]
5. Peeping into 73,000 Unsecured Security Cameras Thanks to Default Passwords. [(accessed on 21 August 2020)]; Available online: https://www.csoonline.com/article/2844283/peeping-into-73-000-unsecured-security-cameras-thanks-to-default-passwords.html
6. Dangle a DVR Online and It'll Be Cracked in Two Minutes. [(accessed on 21 August 2020)]; 2017 Available online: https://www.theregister.com/2017/08/29/sans_mirai_dvr_research/
7. The Internet of Things: An Overview. [(accessed on 21 August 2020)]; Available online: https://www.internetsociety.org/wp-content /uploads/2017/08/ISOC-IoT-Overview-20151221-en.pdf
8. Ling Z., Liu K., Xu Y., Jin Y., Fu X. An end-to-end view of iot security and privacy; Proceedings of the GLOBECOM 2017—2017 IEEE Global Communications Conference; Singapore. 4–8 December 2017; pp. 1–7. [Google Scholar]
9. Khan M.A., Salah K. IoT security: Review, blockchain solutions, and open challenges. Future Gener. Comput. Syst. 2018;82:395–411. doi: 10.1016/j.future.2017.11.022. [CrossRef] [Google Scholar]
10. Akhtar N., Mian A. Threat of adversarial attacks on deep learning in computer vision: A survey. IEEE Access. 2018;6:14410–14430. doi: 10.1109/ACCESS.2018.2807385. [CrossRef] [Google Scholar]
11. Costin A. Security of cctv and video surveillance systems: Threats, vulnerabilities, attacks, and mitigations; Proceedings of the 6th International Workshop on Trustworthy Embedded Devices; Vienna, Austria. 28 October 2020; pp. 45–54. [Google Scholar]