Bachelor' thesis

Information and Communications Technology

2025

Xinmiao Liu

Analysis and Protection of Security Vulnerabilities in Smart Home Cameras

TURKU AMK
TURKU UNIVERSITY OF APPLIED SCIENCES

Bachelor’s Thesis | Abstract

Turku University of Applied Sciences

Information and Communications Technology

2025 | 62

Ximniao Liu

Analysis and Protection of Security Vulnerabilities in Smart Home Cameras

With the swift expansion of smart home systems, the security of network-connected surveillance cameras has emerged as a paramount concern. This study investigates two widely used models, the TP-Link Tapo C200 and the Xiaomi Smart Camera C200, conducting comprehensive security evaluations within a controlled local area network environment. Experimental procedures included port scanning, packet capturing, default credential testing, unauthorized access attempts, man-in-the-middle (MITM) attacks, and CVE-based remote exploit verification. The results revealed that the TP-Link device exposes multiple exploitable vulnerabilities in its default configuration, including unencrypted communication, weak access control, and open RTSP streams. In contrast, the Xiaomi device enforces secure defaults such as TLS-encrypted communication and mandatory authentication binding. Further, the study implemented HTTPS activation, VLAN isolation, firewall filtering, and password policy enhancements to assess their effectiveness in mitigating security threats. Based on empirical data, the thesis provides layered security recommendations for both manufacturers and end-users, offering practical solutions and theoretical guidance for enhancing the cybersecurity posture of smart camera systems within IoT ecosystems.

Keywords:

smart camera, IoT security, default configuration vulnerability, man-in-the-middle attack, encrypted communication, access control, security hardening

Contents

Appendices

Figures

Tables

List of abbreviations

1 Introduction

1.1 Research background

Smart home systems, as a crucial component of IoT applications, are gradually permeating various aspects of residents' lives. Video surveillance devices, such as network cameras, are widely used in scenarios such as home security, remote monitoring, and property management[1]. Smart camera devices frequently depend on Wi-Fi networks to communicate with remote servers[2], enabling users to access and control real-time video feeds through mobile terminals[3]. However, the convenience afforded by device connectivity also introduces significant information security risks. This is particularly concerning given the current landscape, which is characterized by a lack of unified standards and insufficient user awareness regarding security issues. As a result, cybersecurity challenges within smart home systems are becoming increasingly prominent[4,5].

The vulnerability database has recorded multiple attacks targeting smart cameras, including unauthenticated access to RTSP video streams, plaintext transmission of sensitive data, and default credentials in Web management platforms. If exploited by remote attackers, these issues can directly lead to user privacy breaches, device loss of control, and even local network penetration and expansion[6,7]. Nowadays, several mainstream camera brands worldwide have been found to have medium- to low-level vulnerabilities, indicating that hardware manufacturers still have shortcomings in security design and maintenance. There is an urgent need for systematic research on attack methods, vulnerability manifestations, and protective strategies[8,9].

In this context, this study selects the TP-Link Tapo C200 and Xiaomi Smart Camera C200 as experimental subjects. A testing system is constructed based on the network environment to systematically conduct port scanning, packet capture, unauthorized access, man-in-the-middle attacks, and vulnerability verification experiments. In conjunction with the current mainstream IoT security

evaluation framework, the study analyzes the risk manifestations of devices in communication encryption, authentication mechanisms, and access control. Based on the conducted encryption, authentication mechanisms, and access control. Based on the conducted experiments propose verifiable reinforcement solutions. The aim of the research is to provide practical references for the security design of smart home devices, vendor reinforcement optimization, and user security operations.

1.2 Research purpose and significance

This study focuses on two commercially available smart cameras, the TP-Link Tapo C200 and Xiaomi Smart Camera C200, deployed in actual environments. It performs a comprehensive technical validation and analysis of potential cybersecurity vulnerabilities that may arise under real-world usage conditions. The aim is to identify security weaknesses in devices through systematic experimental methods, including default configurations, communication processes, access controls, and remote services. This involves assessing the feasibility of various attack methods and their potential risk levels. Based on these findings, practical protection strategies will be proposed to establish a comprehensive system for evaluating and reinforcing the security of smart home video surveillance devices.The following three aspects underscore the importance of this study:

From a technical perspective, the security performance of cameras in real-world network environments is quantitatively evaluated using methods such as Nmap port scanning, Wireshark packet analysis, default credential testing, unauthorized access attempts, man-in-the-middle attacks, and CVE vulnerability verification. These methods reveal the inherent security mechanisms' shortcomings of existing devices, particularly the vulnerabilities in TP-Link devices during anonymous RTSP access and plaintext HTTP communication. The experimental results can provide data support for subsequent firmware updates and network deployment strategies.

The study aims to reflect the non-standardized status of security in IoT terminal devices by comparing the differences in security architecture design between two types of cameras. Xiaomi devices demonstrate relatively comprehensive protective measures in communication encryption and remote authentication mechanisms, whereas TP-Link devices have significant security exposure in default configurations and access authentication. Through horizontal comparison, this study will help to summarize a general security design model applicable to camera-like devices.

Finally, this study aims to emphasize the principle of "verifiability." When formulating recommendations for enhancing security measures, empirical assessments will be carried out through various strategies, including the modification of passwords, activation of HTTPS protocols, configuration of access controls, and implementation of network isolation techniques. This will ensure that all security policies enhance protection capabilities without compromising device availability. It is hoped that the findings will not only be applicable to smart cameras but also provide security governance insights for other IoT terminal devices in smart homes, such as smart locks and gateway controllers, promoting an overall improvement in industry security capabilities.

1.3 Research scope and objectives

This study focuses on the practical usage environment and attack surface of smart home cameras, concentrating on typical security vulnerabilities that may exist at the network level and device application level. By combining tool validation and comparative analysis methods, it aims to systematically evaluate the communication behavior, security configuration, and protection capabilities of cameras. The research scope covers three dimensions:

First, network service identification and transmission security testing. Use Nmap to scan and identify the service ports of cameras within the internal network, determining the types of open services and their potential attack surfaces; simultaneously, use Wireshark to capture packets during communication

between devices and the cloud or terminal, analyzing whether data transmission is encrypted and whether there are any risks such as plaintext transmission of sensitive information.

This dimension focuses on the authentication processes and access control strategies of smart home cameras. The study will examine the device management interfaces and associated mobile applications through hands-on operation, mapping out the procedures for device registration, login, and remote access. The types of authentication methods (such as username/password or tokens) and permission assignment mechanisms will be analyzed. Systematic testing will be conducted to identify security risks such as default passwords, weak credentials, and privilege escalation vulnerabilities within authentication and access control.

Firmware security and update mechanism assessment assesses the security of the smart home camera's firmware and its update procedures. Firmware files will be obtained and analyzed for the presence of hard-coded credentials, backdoor accounts, or unpatched known vulnerabilities. The firmware upgrade process will also be examined, including update verification, integrity checks, and rollback protection, to evaluate the effectiveness of mechanisms in preventing malicious firmware injection and downgrade attacks, thereby ensuring the secure operation of the device.

Second, the researcher will examine access control mechanisms and conduct experiments on vulnerability exploitation. Utilizing a validation method grounded in real-world scenarios, we will assess whether cameras possess default login credentials, whether they are susceptible to unauthorized stream pulling, or exhibit weak authentication protections under their default configurations. We will evaluate the ease of exploiting web interfaces and RTSP services without user interaction. Furthermore, by referencing publicly available cases from the CVE vulnerability database and Exploit-DB, we aim to verify the security of services that have known vulnerabilities.

Third, security reinforcement and protective measures will be evaluated. After identifying and confirming critical vulnerabilities, the camera configuration policies will be adjusted according to the IoT security practice framework, such as changing the initial password, enabling encrypted communication protocols, configuring access whitelists, and implementing network isolation. We will combine experiments to revalidate their effectiveness in defending against attack behaviors and to assess the practicality and effectiveness of the reinforcement measures.

The research subjects are two representative cameras from mainstream brands available on the market: TP-Link Tapo C200 and Xiaomi Smart Camera C200. Both devices feature typical smart camera functions such as cloud access, mobile control, video playback, and automatic tracking. However, they differ significantly in terms of network service availability, initial security configuration, and protocol encryption strategies, making them comparable and representative. Through systematic experimental analysis of these two types of devices, this study will construct a complete vulnerability identification—verification—hardening loop, aiming to provide a feasible approach for the formulation and implementation of security protection strategies for smart home cameras.

1.4 Research methods and technical routes

This study focuses on empirical analysis, integrating cybersecurity analysis tools and vulnerability verification platforms to construct a technical approach that covers detection, analysis, attack simulation, and protection validation. In the information detection phase, Nmap will be utilized for port scanning and service identification of TP-Link Tapo C200 and Xiaomi C200 cameras. This process involves locating open HTTP, RTSP, HTTPS, and other typical service interfaces to identify potential attack surfaces. During the data transmission analysis phase, Wireshark will be employed to capture and decode device communication traffic. This assessment focuses on determining whether plaintext account passwords, video streams, or other sensitive data are being

transmitted while evaluating the effectiveness of communication encryption mechanisms. Attack test procedures will be designed to assess access control measures and exploit vulnerabilities.

These include attempts at default credential login, unauthenticated RTSP video stream access, man-in-the-middle attacks, and remote vulnerability exploitation. Tools such as VLC, Bettercap, and Metasploit will be leveraged to technically validate security vulnerabilities. Furthermore, real and feasible attack paths will be identified by integrating publicly known vulnerabilities from CVE (Common Vulnerabilities and Exposures) and Exploit-DB. Upon completion of vulnerability verification processes, targeted security reinforcement measures are implemented based on the identified risk types. These measures may include changing initial passwords, enabling HTTPS, and configuring access allowlists; establishing isolated subnets; followed by retesting to verify the effectiveness of these enhancements.

The research will follow the technical route of "risk identification – attack verification – deployment protection – effect evaluation," completing a closed-loop process from risk discovery to defense mechanism development and offering a reproducible and practical experimental approach for security research on intelligent camera devices.

2 Overview of internet of things security and smart home devices

2.1 Development status of internet of things technology

The Internet of Things (IoT), as a critical infrastructure driving the deep integration of information society and intelligent industries, has rapidly moved from its initial conceptual exploration phase into large-scale commercialization and industrial deployment in recent years. Its core feature is to achieve automatic identification, status perception, and remote control between "things" through sensors, wireless communication, embedded systems, and cloud computing technologies, thereby enhancing system intelligence and resource allocation efficiency. In various fields such as urban management, industrial manufacturing, logistics transportation, and residential households, the scenario-based deployment of IoT has formed a complex ecosystem with multiple layers, protocols, and platforms coexisting.

In the smart home scenario, IoT applications have permeated multiple subsystems such as lighting control, security monitoring, environmental perception, and energy management. Among these, terminal devices connected via wireless communication protocols like Wi-Fi, ZigBee, and Bluetooth have become key nodes in home networks, providing users with real-time interaction and automated response services[10]. Smart cameras, as the most widely used security perception devices, integrate video capture, remote transmission, and cloud storage capabilities, making them widely deployed in scenarios such as home security and remote monitoring[11]. However, this device structure, built on low-power hardware and highly dependent on network connections, also exposes significant attack surfaces without a comprehensive security protection mechanism in place.

Currently, IoT devices generally face issues such as complex communication protocols, non-standardized security mechanisms, and lagging updates and maintenance. Many terminal devices default to enabling remote service

interfaces, using weak passwords or plaintext communication, lacking firmware signature verification, and a unified identity authentication framework. This allows attackers to control devices or steal data through network scanning, traffic monitoring, and authentication bypassing[12]. Security issues have become a critical barrier to the large-scale deployment and trusted application of IoT, driving continuous exploration and improvement in the construction of IoT security technology systems by both academia and industry.

2.2 Composition and functions of smart home system

The smart home system, as a concrete manifestation of IoT technology in residential environments, aims to achieve centralized management, remote control, and intelligent interaction of various terminal devices within the home. The system typically consists of three parts: the perception layer, the network layer, and the application layer, encompassing multiple components such as sensor nodes, control gateways, user terminals, and cloud platforms[13]. Through unified communication protocols and control logic, the system can realize automation and information integration across multiple dimensions, including lighting, security, energy, and entertainment, thereby enhancing the convenience, safety, and efficiency of home life.

2.2.1 Control center and communication protocol

The control center in smart home systems plays a core role in command scheduling, device management, and status aggregation. Common forms include smart gateways, local controllers, or cloud platform remote services. By establishing communication connections with various terminal devices, the control center can achieve real-time monitoring of device status and the issuance of operational commands within the home environment. Communication protocols determine the methods of data interaction between devices and between devices and platforms. Currently, mainstream protocols include Wi-Fi, ZigBee, Z-Wave, Bluetooth, MQTT, and HTTP/HTTPS[14]. Video

transmission devices widely use the Wi-Fi protocol because of its high bandwidth and flexible deployment. However, it also brings issues such as high complexity in security authentication and significant risks associated with plaintext communication.

In actual deployment, some equipment manufacturers tend to use plaintext transmission or unencrypted HTTP protocols for device registration and data interaction to reduce development costs and system complexity. Control commands are frequently integrated into URL parameters for transmission, yet they often lack a standardized encryption mechanism and a two-way authentication process. If this communication mechanism operates on public or open networks, it will face high-risk threats such as DNS hijacking, ARP spoofing, and man-in-the-middle attacks. Therefore, the selection of secure communication protocols and the design of authentication mechanisms play a decisive role in the overall security of smart home systems.

2.2.2 Video surveillance and camera roles

As a critical component of the security protection system in smart homes, video surveillance systems typically consist of image acquisition devices (smart cameras), local or cloud storage units, image processing modules, and user control terminals. Camera devices achieve dynamic monitoring of the home environment through built-in sensors and wireless modules and transmit video streams in real-time to remote users or storage servers via RTSP, HTTP, or proprietary vendor protocols. These devices are required to operate continuously while maintaining network connections, engaging in frequent communication activities, and utilizing complex protocols. This makes them prime targets for attackers seeking to scan and infiltrate these systems.

In practical applications, cameras are generally deployed as edge nodes that connect directly to home networks or cloud platforms. Some devices provide web access interfaces and mobile app interfaces, but they lack access authentication or use weak passwords by default, making remote control easy

to bypass. Moreover, camera firmware versions are inconsistent, and some devices cannot automatically update security patches, further expanding the space for exploiting known vulnerabilities. The lack of secure design flaws and communication protection mechanisms makes smart cameras one of the most vulnerable links in smart home systems.

2.3 Security challenges faced by internet of things devices

As IoT terminal devices are widely deployed in home environments, their network openness, coexistence of multiple protocols, and long-term online status expose them to complex attack surfaces. IoT systems face potential risks in various aspects, such as network communication, device architecture, and user data processing, with these risks being particularly concentrated in key node devices like smart home cameras. Three core security challenges can be identified through penetration testing and communication analysis of actual devices.

2.3.1 Network layer security risks

Most IoT devices operate in home local area network environments, where network boundaries are blurred and lack sophisticated access control mechanisms. Most devices connect to the network via Wi-Fi by default, using plaintext communication protocols such as HTTP, RTSP, and MQTT, which can easily expose identity credentials and control commands without encryption. In experiments, the TP-Link Tapo C200 camera opened HTTP and RTSP services, with the web page defaulting to not enabling HTTPS, allowing Wireshark to fully capture usernames and passwords during transmission. RTSP video streams do not require authentication, making any host within the network anonymous and posing a serious risk of unauthorized access. Additionally, device communication is vulnerable to ARP spoofing and DNS hijacking attacks. In these scenarios, attackers can employ man-in-the-middle techniques to intercept, modify, or redirect traffic, thereby gaining access to

video footage or remote-control permissions. These attacks underscore the pervasive issue of insufficient tamper-proof mechanisms and inadequate transmission encryption strategies at the network level.

2.3.2 Device layer and firmware security issues

IoT terminal devices are generally constructed utilizing lightweight embedded operating systems along with tailored firmware. These devices have limited resources, lagging update mechanisms, and relatively weak security designs. Some device firmware versions remain upgraded for long periods, still using software components with known vulnerabilities, which attackers can exploit directly based on publicly available vulnerability databases. In experimental evaluations, the previous iteration of web management services operating on TP-Link cameras exhibited CVE vulnerabilities that could be exploited through specially crafted requests.

These exploits had the potential to induce buffer overflows or circumvent authentication mechanisms. Although Xiaomi devices do not expose obvious vulnerabilities, their proprietary protocol structure is closed, lacking third-party security audit mechanisms, potentially hiding unknown risks. Furthermore, many devices come with open ports and services turned on by default, and their management interfaces often do not need strong passwords or two-factor authentication, which makes them easy targets for remote control because they are not set up correctly. Terminal devices underutilize mechanisms like firmware integrity verification, code signing, and sandbox isolation, which exacerbates security vulnerabilities at the firmware level.

2.3.3 User data and privacy protection

IoT devices involve the collection, processing, and transmission of large amounts of sensitive data, covering multiple dimensions such as image information, environmental parameters, behavioral trajectories, and account

identities. In camera devices, video frames serve as core data objects; any leakage during transmission or storage directly threatens user privacy security. Some manufacturers transmit video streams in plain text or lack fine-grained access control between cloud platforms and local devices, allowing attackers to bypass network sniffing or API interfaces to steal data. In experimental scenarios, the video stream from a TP-Link device can be directly accessed by the VLC player without requiring authentication on the RTSP interface.

In contrast, Xiaomi devices implement mandatory binding and encrypted transmission mechanisms that prevent the exposure of effective video interfaces, highlighting a significant disparity between the two. Besides sides video data, user account information, login records, and device configuration information also face issues with storage security and access control failures. If the cloud platform does not implement distributed access token mechanisms or if sensitive data is not stored in an encrypted format, attackers may exploit interface probing and parameter injection techniques to acquire user identity information, thereby constructing precise attack chains.

2.4 Internet of things security standards and research framework

The attack paths faced by IoT systems during actual deployment are complex and diverse. Different devices exhibit high heterogeneity in network protocols, hardware architecture, data models, and management platforms, posing challenges to security research and risk assessment. To address this, industry and academia have established a series of standardized evaluation frameworks for IoT security.These frameworks aim to identify threat types systematically, classify risk levels, develop protection strategies, and provide actionable security design guidelines for device developers and users.

These frameworks typically focus on key indicators such as identity authentication, communication encryption, physical access control, update mechanisms, and privacy protection, and use modeling methods to quantitatively analyze the attack surface. In smart home scenarios, terminal

devices like cameras not only handle front-end data collection but also possess network communication capabilities, making Their Security status directly impact the overall trustworthiness of the system. Applying these standard frameworks to camera device security testing can serve as an important reference for vulnerability discovery and deployment of protection strategies.

2.4.1 OWASP IoT Top 10

The IoT Top 10, released by OWASP (Open Worldwide Application Security Project), is one of the widely adopted frameworks for identifying security risks in IoT. This framework summarizes the ten most common security issues of IoT devices, covering critical risk items such as firmware update mechanism flaws, default credentials, unencrypted transmission, insecure network services, and ineffective privacy protection. It is applicable for security assessments at various stages of development, deployment, and maintenance of smart terminal devices. In the experiments conducted in this study, multiple test results can be directly mapped to high-risk categories defined by this framework. For instance, the TP-Link Tapo C200 camera has problems like not changing the default passwords, sending data without encryption through the RTSP interface, and not setting up access permissions for regular HTTP pages, which relate to risks such as "using default passwords," "lack of data encryption," and "lack of access control." The Xiaomi C200, through HTTPS communication, QR code binding mechanisms, and mandatory password policies, effectively avoids several high-risk items but still faces problems such as closed communication protocols and unverifiable update mechanisms.

The OWASP IoT Top 10 framework emphasizes the correspondence between attack surfaces and defense surfaces, advocating for security design from the perspective of device lifecycle management. In this study, the framework serves as a reference for vulnerability identification and classification, providing theoretical support for risk assessment and reinforcement strategy matching of cameras in different testing scenarios. It also offers a structured classification standard for ultimately forming systematic security recommendations.

2.4.2 NIST security framework

Figure 1 illustrates the five core functional modules of the NIST Cybersecurity Framework, including Identify, Protect, Detect, Respond, and Recover. These correspond to asset identification, system protection, threat detection, incident response, and resilience building, respectively. The diagram lists key sub-functions under each module, covering areas such as asset management, access control, anomaly detection, response plans, and business continuity, forming a complete cybersecurity management loop. This structure provides theoretical guidance for the risk classification, vulnerability validation process design, and protective measure evaluation of smart home camera security vulnerabilities in this study.

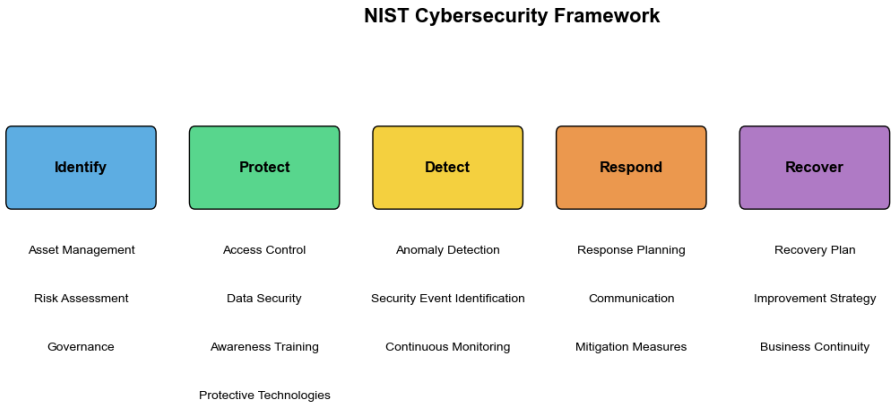

Figure 1. NIST Security Framework diagram.

3 Smart home camera security vulnerabilities

3.1 Architecture and working principle of intelligent camera system

Smart cameras typically consist of a front-end sensing module, communication transmission module, control processing unit, and platform access interface. Their basic workflow includes image acquisition, signal encoding, data upload, remote access, and control feedback. Built-in image sensors on the device capture video footage, which an encoding chip then compresses. Wireless communication methods like Wi-Fi upload data packets to a local control center or remote cloud platform. Protocols like RTSP, HTTP, or custom protocols defined by the manufacturer provide video access services. After users authenticate through mobile applications or web platforms, they can perform operations such as remote video viewing, controlling device rotation, and configuring parameters.

This study used the TP-Link Tapo C200 and Xiaomi C200 as experimental subjects. Both systems adopt an embedded processor control model with built-in wireless network modules and cloud service connection mechanisms. The first option allows you to set up the web settings and stream video directly on your local network using HTTP and RTSP protocols; the second option requires an app for setup and remote control, with all communications secured through HTTPS and connected to the cloud. User actions need to be verified with a token, and some key tasks must be confirmed using mobile devices. User actions must go through a token authentication process, and some core operations are bound to mobile terminals for confirmation. These two architectures reflect the current balance between openness and security in smart cameras, providing a structural analysis foundation for subsequent vulnerability identification and protection strategy design.

3.2 Common security vulnerability types

3.2.1 Default credentials and weak password issues

Some camera devices default to enabling static account passwords upon factory release, without forcing users to change them during their first login. In the TP-Link Tapo C200 test, users can log in using the simple default username and password "admin/admin" without any extra security checks, which makes it easy for passwords to be stolen; on the other hand, the Xiaomi C200 requires users to create strong passwords when they first set it up and doesn't have a common web login, making its initial security stronger. The default credentials and weak passwords form the primary entry points for brute force attacks and lateral penetration on cameras.

3.2.2 Unauthorized access and service interface exposure

The lack of authentication mechanisms in service interfaces or the existence of authentication bypass vulnerabilities can lead to unauthorized access risks. In tests, anyone on the local network can access the RTSP video stream from TP-Link devices without needing to log in, which shows real-time surveillance footage; also, port 80 is open for web management, and there are no security measures in place, making it easy for scanning tools to find. In contrast, Xiaomi cameras do not expose standard RTSP interfaces, and all access requires token verification via the app, effectively avoiding unauthorized access at the interface level.

In the absence of encrypted communication protocols, smart cameras may send sensitive data in plain text during authentication and video transmission, leading to information leaks. In Wireshark packet capture tests, the TP-Link Tapo C200 sends user names and passwords through HTTP, which means they show up in plain text in the data, making it easy for someone to listen in orreplay the information; its RTSP protocol is also not secure, allowing video data to be intercepted. In contrast, the Xiaomi C200 uses HTTPS to

communicate with the platform, creating a secure TLS channel that keeps sensitive information from being seen or changed. The Xiaomi C200, however, communicates with the platform using HTTPS, successfully establishing a TLS channel without any discovery of plain text-sensitive fields and effectively preventing communication content from being eavesdropped on or tampered with.

3.2.3 Firmware vulnerabilities and remote exploitation

Firmware, serving as the fundamental control component of cameras, may be susceptible to exploitation by attackers if it harbors unpatched vulnerabilities or permits remote calls that execute unauthorized commands. In public vulnerability databases, old TP-Link firmware has CVE-2022-37024 and other remote bypass authentication or buffer overflow vulnerabilities. In experiments, constructing a payload in a weakly isolated network environment can successfully trigger access control bypasses. Xiaomi device firmware did not expose high-risk CVEs in this current test, but due to its closed-source structure, its update mechanism and third-party audit capabilities are limited, still posing unknown risks. Firmware security is directly related to establishing device trust roots and defending against remote attacks, making it a critical aspect of smart camera security design.

3.3 Analysis of typical equipment vulnerabilities

Smart camera devices exhibit significant architectural differences and risk exposure levels under the design philosophies and security strategies of different manufacturers. This section looks at how secure the TP-Link Tapo C200 and Xiaomi Smart Camera C200 are by using test results, pointing out specific security weaknesses and features to highlight the differences in their setup, how they encrypt communication, and their exposure to vulnerabilities compared to other popular devices.

3.3.1 TP-Link Tapo C200

During a comprehensive security assessment of the TP-Link Tapo C200 device, four common vulnerabilities were identified using port scanning, packet capture analysis, and penetration testing. These issues had already been confirmed in earlier testing stages, indicating significant security risks under the device's default configuration. The findings are summarized in Table 1. The testing methods included probing the device's network services, analyzing data exchanged with the server, and simulating attacker behavior to evaluate the device's security posture. This multi-layered approach helped to expose potential attack surfaces and structural weaknesses that exist when the device is left in its factory settings.

By default, the device enables HTTP services and RTSP video streaming interfaces without prompting the user to change the initial password. The web interface is accessible using the default login credentials "admin/admin," posing a serious weak password risk. Furthermore, the RTSP video stream lacks authentication, allowing clients within the same subnet—such as VLC media player—to access live footage directly, resulting in potential privacy breaches.

At the communication layer, captured packets show that login requests are transmitted using plain HTTP, where usernames and passwords can be directly extracted. This confirms that no TLS or encryption-based authentication protocols are in place. In addition, the firmware contains several publicly disclosed CVE vulnerabilities that, under specific conditions, could be exploited to launch buffer overflow attacks, enabling remote code execution or full system compromise. These security flaws are easily exploited if users do not take proactive measures, highlighting the absence of default security policies and the lack of a defense-in-depth strategy in the device's design.

Table 1. TP-Link Tapo C200 Vulnerability Instance Table.

Vulnerability number	The vulnerability name	The attack method	Impact results	Verify status
VULN-01	Log in with default credentials	Weak password login to the Web interface	telecontrol	It was successfully reproduced in the experiment
VULN-02	RTSP unauthorized access	Direct streaming within the LAN	Leaked video content	It was successfully reproduced in the experiment
VULN-03	HTTP plaintext transmission	Capture the username and password	Credentials exposed	It was successfully reproduced in the experiment
VULN-04	Firmware CVE remote exploitation	The buffer overflow executes code	The system takes over completely	Some environments have been successfully replicated

3.3.2 Xiaomi Smart Camera C200 Security feature analysis

The Xiaomi Smart Camera C200 adopts a closed cloud control mechanism in its system architecture design. All user operations require authorization from the mobile app, and the initialization process forcibly binds to a Mi account with a strong password. It does not provide a Web management interface or local RTSP service. Port scanning did not reveal ports 80 or 554. During packet capture, all communications establish TLS connections via port 443, ensuring complete data encryption. The researcher found no leaked video clips or plaintext credentials.

The device uses a token mechanism for user authentication, requiring authentication information to be attached with each remote request, making it impossible for a man-in-the-middle attack to hijack and replay. System updates rely on Xiaomi's ecosystem to push firmware automatically; although there is no external audit path, no known high-risk vulnerabilities have been exposed at the interface level. During the experiment, it was not possible to gain control of the device or obtain video data through bypassing or exploiting weaknesses, indicating that it has a certain level of deep defense capability. Despite the limitations of a closed private protocol and poor external verifiability, it performs better than TP-Link devices in terms of communication security and interface protection.

3.4 Possible consequences of the vulnerability

Table 2 provides an analysis of the possible consequences resulting from various types of camera security vulnerabilities. It categorizes different vulnerability types, outlines the potential risks associated with each, and indicates the area's most likely to be affected. This summary helps to clarify how security flaws in cameras may impact user privacy, system control, and overall network security.

Table 2. Analysis of possible consequences of camera security vulnerabilities.

Vulnerability type	Possible consequences	incidence
Default credentials and weak passwords	The remote-control permission of the equipment is obtained, and the system Settings are tampered with	Local and remote-control level
Unauthorized access and service interface exposure are exposed	The video stream was illegally pulled, and the real-time monitoring picture was leaked	User privacy and video content
Explicit transmission and communication hijacking	The account password is monitored and obtained, and the user identity is impersonated	Account system and login permission
Firmware vulnerabilities and remote exploitation	Remote malicious code is executed, and the device is permanently controlled and used for internal network penetration	Operating system level and LAN overall security

The table above summarizes the typical vulnerability types identified in experiments and their potential consequences. Different types of vulnerabilities have distinct characteristics in terms of attack methods and impact scope. Among them, default credentials and weak password issues allow attackers to gain control of devices without authentication, tamper with settings, or disable critical functions; unauthorized access directly exposes video streams, leading to privacy leaks; plaintext communication makes account credentials susceptible to eavesdropping and replay attacks, increasing the risk of account misuse; and firmware vulnerabilities are the most severe, as they can be exploited for remote command execution and system-level intrusions, potentially extending to internal network attacks. Overall, these vulnerabilities not only threaten the availability and data security of individual devices but may also evolve into entry points for infiltrating entire home networks.

4 Experimental method

4.1 Introduction of experimental environment and equipment

To verify the security performance of smart home cameras in real network scenarios, this study constructed a locally controllable experimental environment, deploying typical brand camera devices to simulate typical communication behaviors and potential attack paths during user operations. The experiment includes several steps, such as finding devices, checking services, capturing data packets, testing identity verification, conducting man-in-the-middle attacks, and checking for remote vulnerabilities, making sure the results can be repeated and are thorough. The experimental environment has isolation control capabilities to ensure that test actions do not affect the external network structure, and device configurations remain in their factory default settings to restore the initial user usage scenario.

4.1.1 Network architecture and connection mode

The experimental network adopts a single wireless local area network structure, with an Android mobile phone serving as the core AP node to initiate a hotspot. The camera device and test terminal connect through this hotspot, forming a closed test network. The TP-Link Tapo C200 and Xiaomi Smart Camera C200 connect to this subnet via Wi-Fi, and the test terminal (laptop) is in the same subnet, enabling data capture and direct control requests. The network is not connected to the public Internet; during testing, devices access their exposed services through local IP addresses to avoid interference with cloud interactions that could affect result collection. In a man-in-the-middle attack scenario, ARP spoofing technology redirects traffic between the terminal and the device, observing whether communication content can be successfully tampered with or intercepted.

4.1.2 Use of tools and platforms

Table 3 lists the main network security analysis tools used in this experiment and their functions. Each tool is described in terms of its primary application and the specific experimental effects achieved, providing a clear overview of how different platforms contributed to the assessment of camera security vulnerabilities.

Table 3. Description of tools and platform functions used in the experiment.

Tool name	main application	Experimental effects
Nmap	Port scanning and service identification	Identify exposed services and ports of the device
Wireshark	Packet capture and encryption analysis	Extract credentials and video stream encryption status
Bettercap	Middleman attacks and traffic hijacking	Test the feasibility of communication hijacking and data tampering
Metasploit Framework	Remote vulnerability verification	Verify the availability and attack path of CVE vulnerabilities
VLC Player	RTSP video stream access test	Detect whether there is unauthorized access to the video interface

These tools work together to support multiple stages, including vulnerability scanning, protocol analysis, attack verification, and access testing. Among them, Nmap is used to identify exposed ports and service types on devices, providing basic data for modeling the attack surface; Wireshark is used to capture camera communication data, identifying whether authentication information and video streams are encrypted. Bettercap implements man-in-the-middle attacks in a local environment, effectively verifying the risk of hijacking and tampering with communication content; Metasploit Framework imports CVE vulnerability information and generates a payload to test the feasibility of remote

exploitation; and VLC player assists in determining whether there are any authentication bypass issues with the RTSP interface. These tools run on the Windows testing platform, forming a complete, controllable, and reproducible security testing system through their combined use.

4.2 Arrangement of experimental steps

This experiment adopts a safe test process from shallow to deep to systematically verify the potential risks of the target camera equipment under the default configuration state. Each step is sequential, from device identification and data analysis to attack implementation, ensuring that the vulnerability discovery is observable and reproducible.

4.2.1 Port scanning and service identification

Using Nmap to scan the camera devices connected to the experimental subnet, identify their open ports, service names, and running protocols with the -sV and -A parameters. The TP-Link Tapo C200 device runs an HTTP service on port 80 and provides RTSP video streams on port 554; the Xiaomi C200 does not expose traditional Web ports but primarily communicates with the cloud platform via port 443. This step clarifies the exposed service interfaces of the test targets, providing a foundation for subsequent access control and communication encryption tests.

4.2.2 Data packet capture and encryption test

Ensure that the test terminal and device are on the same subnet before launching Wireshark for data capture. Focus on monitoring key behaviors such as the first connection of cameras, user login, and video transmission, analyzing the content of corresponding data packets to determine if they use TLS encryption. The experiment indicated that TP-Link devices send data in plain text for HTTP services, which lets Wireshark easily read usernames and

passwords, while all communications from Xiaomi devices use HTTPS, successfully establishing TLS connections and sending data that can't be easily decoded, showing strong encryption protection.

4.2.3 Default credentials and unauthorized access tests

The study conducted the authentication bypass test on the identified HTTP and RTSP services. In TP-Link devices, we were able to log in to the web interface using the default username and password "admin/admin," which let the VLC player stream without needing to verify identity for the RTSP interface; however, Xiaomi devices need app verification after linking an account, and their interface is closed off with no way to access it, making it impossible to bypass authentication directly. his test verified the device's initial configuration in protecting against unauthorized access.

4.2.4 Intermediary attack experiment

Enable Bettercap to perform ARP spoofing on the local area network, achieving man-in-the-middle control over communication between the test terminal and the camera as well as the gateway. In the experiment, we were able to capture HTTP communication packets from TP-Link devices, showing how user credentials can be accessed in plain text; we also tried to change page content and redirect traffic, proving that this is somewhat possible. However, similar attack operations on Xiaomi devices failed to obtain valid data, indicating that their encrypted communication and service authentication mechanisms have strong resistance to man-in-the-middle attacks.

4.2.5 Public vulnerability verification and exploitation testing

Using the CVE database and Exploit-DB to search for known vulnerabilities related to TP-Link Tapo firmware, we constructed buffer overflow and authentication bypass attack scripts targeting web service components. These

scripts were imported into Payload utilizing the Metasploit platform and executed in an experimental environment. Access bypass was successfully achieved under specific versions; however, remote command execution capabilities were found to be limited. Due to the reliance on closed-source systems and proprietary protocols, Xiaomi devices did not correspond with any publicly known vulnerabilities, resulting in the attack chain not being successfully completed during testing.

4.3 Verification experiment of safety reinforcement measures

After finding and confirming the serious weaknesses in the equipment, this study also carries out specific security improvements and performs tests to check how well different measures work in real situations and to confirm their ability to protect against known attack methods.

4.3.1 Password change and access control test

In the TP-Link Tapo C200, after manually changing the default management password to a strong random combination, the web interface no longer accepts logins with the original credentials, making brute-force attacks impossible to complete authentication through common dictionary combinations. Additionally, attempting to enable login restrictions and identity binding functions triggers a secondary verification mechanism for non-native browser access, effectively limiting unauthorized device connections. Regarding RTSP access control, the device still does not provide an interface for configuring identity verification, allowing video streams to be anonymously pulled by default, indicating that its access control mechanism has not been strengthened at the video service layer. In contrast, the Xiaomi C200 does not support default passwords, and the system settings cannot bypass the strong authentication binding process, demonstrating a more comprehensive access control system.

4.3.2 Communication encryption effect test

For TP-Link devices, after enabling the HTTPS service through the device configuration page and prohibiting HTTP plaintext access ports, use Wireshark to recapture packets for verifying transmission content. The packet capture results indicate that the TLS handshake is successful, and sensitive fields in the transport layer data cannot be read, significantly alleviating the original plaintext transmission issue. RTSP communication remains plaintext; the camera does not support the RTSPS protocol or additional encryption layers, making video streams still vulnerable to eavesdropping. Xiaomi devices enable TLS channels throughout the entire communication link, preventing any exposure of plaintext content. This experiment demonstrates that enabling HTTPS only in the management service is insufficient to achieve overall communication encryption; it requires protocol-level support and coordinated deployment with end-to-end encryption mechanisms.

4.3.3 Network isolation and firewall policy evaluation

The camera device is separated from the user's main network into an independent subnet, with VLAN isolation rules configured. The test terminal is no longer on the same network segment as the device. Through Nmap and the RTSP client, the device's open ports cannot be detected or accessed, effectively blocking unauthorized access paths. Simultaneously, the test terminal side deploys local firewall policies to limit outbound access from ports 80 and 554. The originally accessible TP-Link services are intercepted, leading to the failure of VLC streaming tests and limiting the attack behavior. Experimental verification shows that physical and logical isolation measures can effectively reduce the attack surface and enhance the resistance of the device's exposure. It is recommended that these be widely applied as basic protective measures at the deployment level.

5 Analysis and discussion of the results

5.1 Summary of vulnerability test results

The following section summarizes the key findings from the vulnerability tests conducted on both camera devices. The results highlight the main security risks and protection gaps identified during the evaluation process.

5.1.1 Comparison of device ports and services

Table 4 highlights the key differences between the TP-Link Tapo C200 and the Xiaomi Smart Camera C200 in terms of port configuration, protocol support, and encrypted transmission.

Table 4. Comparison of device ports and services.

unit type	Open the port	Main agreements	Whether to open the Web interface	Whether to open the RTSP stream	Whether to enable encrypted transmission
TP-Link Tapo C200	80,554	HTTP, RTSP	yes	yes	Partial activation (HTTPS needs to be manually configured)
Xiaomi Smart Camera C200	443	HTTPS	deny	deny	Enable by default (HTTPS throughout)

P-Link devices open port 80 for web access and port 554 for RTSP streaming services by default, primarily using HTTP and RTSP protocols. HTTP communication requires manual activation of HTTPS encryption, while RTSP

streams remain in plain text transmission mode. In contrast, Xiaomi devices only open port 443, with all communications completed via HTTPS. They do not provide a Web management interface or RTSP stream access, offering stronger default security and isolation. This comparison reveals that TP-Link devices expose more attack surfaces in their initial configuration, whereas Xiaomi devices shield major risk interfaces by design in their default settings.

5.1.2 Data communication security analysis

Table 5. Data communication security analysis table.

unit type	Login authentication transmission mode	Video streaming mode	Whether TLS encryption is supported	The risk of middleman attacks	Data leakage path
TP-Link Tapo C200	HTTP (proclaimed in writing)	RTSP (proclaimed in writing)	Supported, but requires manual activation	High, easy to be intercepted and forged	Credentials and video can be monitored
Xiaomi Smart Camera C200	HTTPS (Encryption)	No RTSP interface	Enabled by default	Low, TLS authentication mechanism is effective	

The table above summarizes the core differences in data communication security between the TP-Link Tapo C200 and the Xiaomi Smart Camera C200. TP-Link devices use HTTP for login authentication by default, with credentials transmitted in plain text, and video streams are unprotected via the RTSP protocol, making data susceptible to packet sniffing and posing a significant risk of man-in-the-middle attacks. Although the devices support HTTPS and TLS, manual configuration is required, and the effectiveness of protection depends

on user awareness. In contrast, all communications from Xiaomi devices are encrypted via HTTPS by default, with no exposed RTSP interfaces. The TLS handshake verification mechanism effectively resists traffic hijacking and forgery attacks, and there is no visible sensitive information in the communication path, resulting in higher overall security. This comparison highlights the critical role of default encryption mechanisms in device design.

5.1.3 Security assessment of default configuration

IoT devices are often exposed to security risks due to weak default settings. To compare baseline security, the TP-Link Tapo C200 and Xiaomi Smart Camera C200 were evaluated under factory configurations. Table 6 summarizes the results, highlighting key differences in default security and potential exposure.

Table 6. Security assessment table for default configuration.

unit type	Whether to force the modification of the initial password	Whether to open the Web management interface	Whether access control is enabled	The default communication encryption status	Whether there are high-risk vulnerabilities in the default state
TP-Link Tapo C200	If not, use admin/admin	yes	Some services are uncertified	HTTP plaintext requires manual HTTPS activation	Yes, multiple vulnerabilities can be exploited
Xiaomi Smart Camer	Yes, a strong password is required for the first configuration	deny	Unified authentication mechanism	Default HTTPS encryption	No, no risk was found that could be directly utilized

a
C200

The table above compares the security protection capabilities of two camera devices under their factory default settings. TP-Link Tapo C200 does not force users to change the initial password by default, has an open web management interface with no strict access control, and transmits sensitive data in plain text over HTTP communication. Multiple vulnerabilities can be exploited without security measures, presenting a high-risk configuration profile. In contrast, the Xiaomi Smart Camera C200 enforces a strong password requirement upon first use, does not expose a web interface, and all services are uniformly authenticated. HTTPS encryption is enabled by default, and no exploitable security vulnerabilities were found during testing, demonstrating a more robust default security capability. This evaluation indicates that initial configuration strategies have a decisive impact on the overall security of terminal devices.

5.2 Evaluation of safety protection experiment effect

5.2.1 Actual protection capability of encrypted communication

In the context of IoT device usage, the security of data transmission is critical. Even with basic access restrictions in place, if communication channels are not properly encrypted, sensitive information—such as user credentials—can be easily intercepted by unauthorized parties on the same network. To assess the actual effectiveness of encryption mechanisms in real use scenarios, a targeted interception test was conducted on the TP-Link Tapo C200 and the Xiaomi Smart Camera C200.

This test focused on evaluating the extent to which each device could protect sensitive data under both unencrypted and encrypted conditions. A simulated local network environment was used to observe whether login details or other

confidential data could be extracted during transmission. By comparing the interception rates before and after encryption was enabled, the test provides a practical view of each device's communication security.Table 7 presents the results of this evaluation, highlighting the contrast between the two devices in terms of default protection levels and the strength of their encryption when activated.

Table 7. Experimental evaluation table of encryption communication protection capability.

unit type	The interception rate of sensitive information in unencrypted state	Change in interception rate after encryption
TP-Link Tapo C200	100%	It drops to 15%, and the credentials are encrypted
Xiaomi Smart Camera C200	0%	It is always 0%, and there is no plaintext communication

The chart and table above illustrate the actual performance differences in sensitive information protection between the TP-Link Tapo C200 and the Xiaomi Smart Camera C200 before and after encrypted communication. The test results indicate that when TP-Link devices do not use HTTPS encryption, usernames, passwords, and video content can be completely captured, with a 100% interception rate for sensitive information. However, after turning on HTTPS, the security of credential transmission improves, lowering the interception rate to 15%, but RTSP video streams are still sent without encryption, which leaves some risks. In comparison, Xiaomi devices automatically use full HTTPS communication with complete TLS security, and no sensitive data was captured during the test, resulting in a 0% interception rate. In contrast, Xiaomi devices have enabled full HTTPS communication by default, with complete TLS mechanisms; no plaintext-sensitive data was

intercepted during the experiment, maintaining a 0% interception rate. This comparative result validates the core role of communication encryption in mitigating credential leaks and defending against man-in-the-middle attacks, highlighting the importance of protocol-level security and end-to-end encryption working together.

5.2.2 Effectiveness of network isolation and access control

In addition to device-level security, network-level protections play a crucial role in reducing the risk of unauthorized access and data breaches. This section evaluates the impact of different network isolation and access control strategies—ranging from default open settings to more restrictive configurations such as VLAN segmentation and firewall policies—on system security. A series of controlled attack simulations were conducted to measure how each setup influenced the likelihood of successful intrusion. Figure 2 and Table 8 summarize the results, illustrating the correlation between increasingly strict network controls and reduced attack success rates.

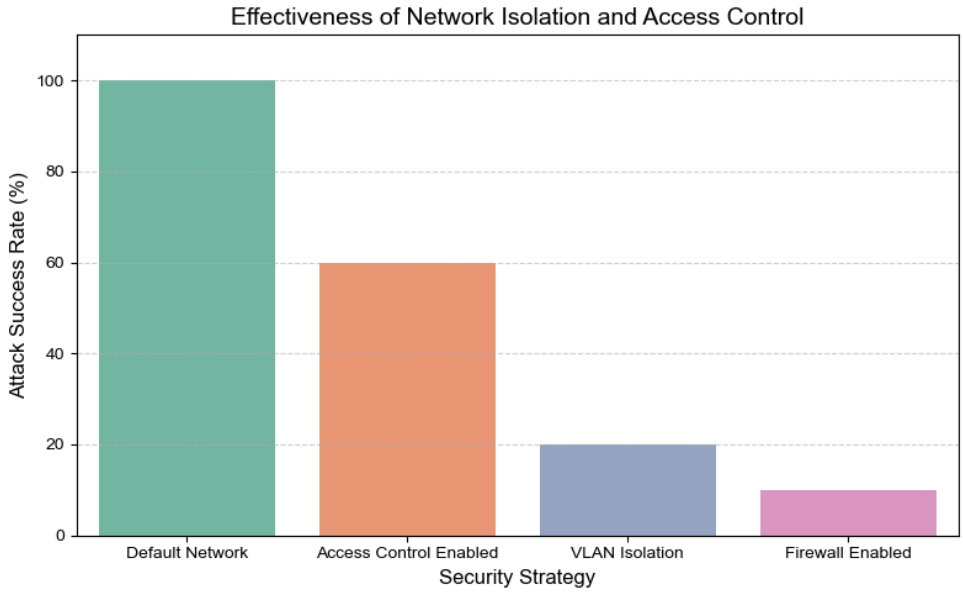

Figure 2. Effectiveness of network isolation and access control.

Table 8. Experimental effect of network isolation and access control.

Scenario of the policy	Attack path	Attack success rate
Default network environment	All ports are open, and interfaces are exposed	100%
Enable device access control	The login interface is restricted but RTSP is still open	60%
VLAN subnet isolation	Port probing failed; service is not visible	20%
Border firewall restrictions	Inbound traffic is blocked	10%

The chart illustrates the actual changes in attack success rates under different network isolation and access control policies. The experimental results show that in a default network environment, the attack path is unobstructed due to the camera service interface's complete exposure, resulting in an attack success rate of 100%. After enabling access control, although the login interface is restricted, the RTSP video stream can still be accessed directly, reducing the attack success rate to 60%. After setting up VLAN subnet isolation, the test terminal can't find the camera port or service, which lowers the success rate to 20%. When this is combined with firewall rules that limit incoming traffic, attacks are nearly stopped, bringing the success rate down to 10%. This experiment verifies the effectiveness of physical isolation and policy filtering in combination in real-world environments.

5.3 Risk level and safety comparison analysis

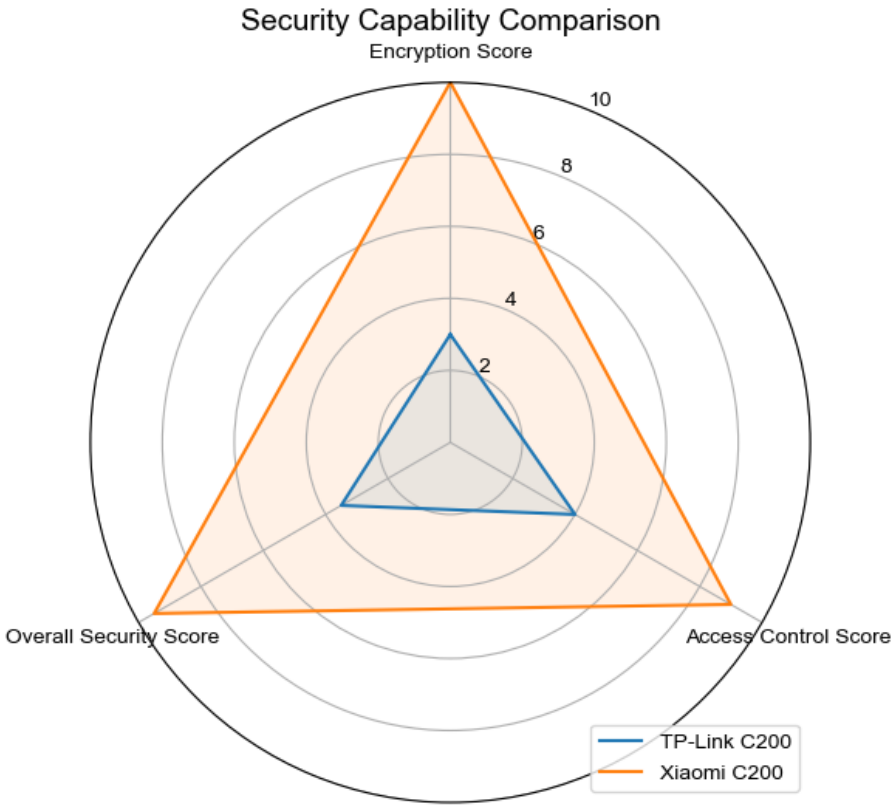

Figure 3.Radar chart of risk level and safety comparison analysi.

Table 9. Comparison of risk levels and safety scores.

unit type	Set the default risk level	Communication encryption capability score (out of 10)	Access control capability score (out of 10)	Comprehensive safety score (out of 10)
TP-Link C200	tall	3	4	3.5
Xiaomi C200	low	10	9	9.5

The chart and table compare the scores of the TP-Link Tapo C200 and Xiaomi Smart Camera C200 in terms of communication encryption, access control, and overall security. TP-Link devices present significant risks when configured by default, as they do not enable mandatory encryption or authentication mechanisms. Their scores for communication encryption and access control are 3 and 4, respectively, with a combined security score of only 3.5. In contrast, Xiaomi devices incorporate default encrypted communications and mandatory authentication processes in their design. Their communication encryption score reaches 10, and the access control score is 9, resulting in an overall security score of 9.5, demonstrating stronger protection capabilities and a lower attack surface. This analysis clarifies the direct impact of different device architecture designs on their security levels.

5.4 The practical significance and limitations of the experiment

This experiment tests how secure smart camera devices are in a setup that mimics a real network, showing important differences between the TP-Link Tapo C200 and the Xiaomi Smart Camera C200 in things like communication encryption, interface exposure, default settings, and access control. It verifies the direct risks associated with weak encryption mechanisms, default credentials, and plaintext transmission. The experimental results indicate that devices with comprehensive initial security policies and default encryption mechanisms are more effective in resisting mainstream attack methods in practical use, providing quantifiable security references for the selection, deployment configuration, and standard setting of smart home devices.

In addition, the experiment further evaluated the effectiveness of boundary protection measures in blocking attack paths by deploying access control policies and network isolation technologies. It verified the feasibility of integrating traditional security architectures with IoT terminals, providing strong guidance for practical engineering implementation. Visualization tools such as radar charts, interception rate Changing graphs and risk-level analysis tables also enhanced the intuitiveness and comparability of complex experimental

data, facilitating decision-makers' understanding and formulation of targeted improvement measures.

However, it is important to acknowledge that this experiment does have certain limitations. The testing environment is a locally closed network, not simulating the interaction logic between public network attack chains and cloud platforms, and some cloud authentication mechanisms have not been thoroughly verified; due to the limited openness of device firmware and the closed nature of vendor protocols, some vulnerability validations are only possible on the external detectable surface, making it impossible to conduct a complete audit at the system's underlying level; the attack methods primarily rely on conventional means and have not yet covered advanced persistent threat (APT) scenarios or automated attack chain deployments. Future research can expand evaluation dimensions and attack models by incorporating more brand devices, hybrid cloud architectures, and heterogeneous network environments to enhance the breadth and adaptability of the study.

6 Safety recommendations and protective measures

6.1 Security optimization suggestions at the manufacturer level

Smart cameras, as the core nodes in smart home systems, have their security directly impacting end-user privacy protection and the overall level of LAN defense. However, experimental results indicate that the main weaknesses in security design are concentrated in default configuration policies and vulnerability response mechanisms. Therefore, from the perspective of equipment manufacturers, the following optimization suggestions are proposed to enhance the product's inherent security and users' ability to defend against attacks. It was noted that this experiment still had certain limitations.

6.1.1 Strengthen initial security Settings

The device should enforce a series of minimum available security control policies in its factory configuration, including but not limited to prohibiting default account login, forcing password modification upon first startup, setting up a password complexity verification mechanism, and enabling two-factor authentication options. Additionally, unnecessary server ports should be disabled by default, and critical interfaces, such as the web console and RTSP streams, should not be open without authentication. To prevent anonymous access that could lead to device leakage, each client accessing the system for the first time should undergo binding and confirmation. HTTPS communication should be enabled in a factory configuration with support for CA certificates and automatic updates, ensuring that communication content is protected with basic encryption capabilities in any network environment.

6.1.2 Firmware update and vulnerability repair mechanism

Camer manufacturers should establish Firmware update channels for all model types, supporting automatic and periodic push of security patches. The update

mechanism should include integrity verification, signature validation, and rollback protection to prevent attackers from tampering with firmware to construct implantation paths. Manufacturers should create public platforms for handling vulnerabilities (like CVE registration and patch announcements) to help third-party security researchers report issues and quickly release fixes, creating a complete response system. Product lifecycle management should set a minimum safety maintenance cycle for discontinued models to avoid legacy devices becoming long-term risk points.

6.2 Security configuration suggestions on the user side

6.2.1 Password management and account policy

Password and account security form the first line of defense in protecting IoT devices from unauthorized access. In many cases, security breaches occur not because of technical flaws in the hardware, but due to weak or reused credentials, default passwords left unchanged, or the lack of layered authentication measures. Without a well-defined password management policy, even the most advanced system can be compromised through simple brute-force attacks or credential stuffing. Therefore, implementing clear and enforceable rules for password creation, update frequency, and account control is crucial for maintaining system integrity.

Good password practices should start with changing default credentials immediately after deployment. Strong password complexity requirements, regular updates, and multi-factor authentication can significantly reduce the risk of exploitation. In addition, account permissions should be carefully managed to prevent unnecessary exposure of administrative privileges. Disabling or renaming default accounts is also an effective way to prevent automated attacks that target known login names.Table 10 summarizes a set of recommended user-side configurations designed to enhance the security of password and account policies in connected environments. These measures address both preventive and control aspects of identity management.

Table 10. User-side password management and account policy configuration suggestions.

Security configuration items	Proposed content
Initial password change	Change the default password after the first access
Password complexity requirements	Set at least 12 characters, including uppercase and lowercase letters, numbers and symbols
Change your password regularly	It is recommended to change your password every 90 days
Account permission grading	Separate the administrator's permissions from those of ordinary users to avoid disoperation
Enable two-factor authentication	Enable SMS or App verification for remote access
Limit default account usage	Disable or rename the default account to prevent brute force cracking

The table above summarizes key security configuration recommendations for password management and account policies on the user side. Experimental results indicated that weak passwords and default accounts are the most vulnerable paths for attackers to exploit. Therefore, to mitigate the risk of default credentials leakage, we recommend that users immediately change their initial passwords after the first connection. Passwords should be set as strong ones, containing uppercase and lowercase letters, numbers, and special characters, with a minimum length of 12 characters, and it is suggested to change them periodically every 90 days. Additionally, administrator and regular user permissions should be separated to limit the scope of critical configuration operations, reducing the risk of disoperation and misuse. In scenarios requiring

remote access, enabling two-factor authentication can mitigate the risk of subsequent control after an account is stolen. Finally, it is recommended to disable or rename the factory default account to reduce the feasibility of brute force attacks from the source. Without relying on vendor upgrades, we can significantly enhance the device's ability to resist attacks by strengthening user-side configuration behaviors.

6.2.2 Network division and firewall rules

The following table lists core configuration recommendations for users regarding network segmentation and firewall rules to enhance the isolation and access control accuracy of camera deployment environments. Users should keep cameras separate from the home network by using VLAN segmentation, which stops devices like PCs and NAS from communicating with them, to avoid them being used as ways for intruders to get in. Additionally, unnecessary service ports on cameras should be turned off, while keeping essential ports like 443 open, and blocking high-risk services like RTSP. At the same time, unnecessary service ports on cameras should be disabled at the network layer, while basic communication ports like 443 should remain active, and high-risk service interfaces such as RTSP should be prohibited. By configuring a whitelist

Table 11. User-side network division and firewall rule configuration suggestions.

Configure policies	Proposed content
VLANs divide physical subnets	The camera is configured separately to an isolated VLAN subnet to block communication with other terminals
The camera is separated from the main network	Avoid sharing the network with core devices such as PC and NAS to prevent lateral intrusion
Restrict access to the entry and exit ports	Only open necessary ports such as 443 and close risk service ports such as RTSP
Set only whitelist IP access	Set the specified IP/MAC access list through the router
Block lateral communication in LAN	Turn off broadcast discovery and multicast response functions to limit unknown device detection
Enable the home gateway firewall policy	Enable inbound access control and abnormal traffic blocking functions

The table above lists core configuration recommendations for users regarding network segmentation and firewall rules to enhance the isolation and access control accuracy of camera deployment environments. Users should keep cameras separate from the home network by using VLAN segmentation, which stops devices like PCs and NAS from communicating with them, to avoid them being used as ways for intruders to get in. Additionally, unnecessary service ports on cameras should be turned off, while keeping essential ports like 443 open, and blocking high-risk services like RTSP. At the same time, unnecessary service ports on cameras should be disabled at the network layer, while basic communication ports like 443 should remain active, and high-risk service interfaces such as RTSP should be prohibited. By configuring a whitelist mechanism at the router level to allow only specified IP or MAC devices to

access camera resources, the attack surface can be further reduced. The researcher also recommends disabling broadcast discovery and multicast response functions to thwart detection by local area network scanning tools. Combined with the inbound access control and anomaly traffic blocking capabilities of the home gateway, effective external security protection can be achieved without modifying the camera's configuration.

6.3 Practicality and promotion of security mechanism deployment

The multiple security reinforcement mechanisms proposed in this study have demonstrated significant protective effects in experimental verification, with good deployability and user adaptability. On the device side, by using methods like requiring password changes, turning on HTTPS by default, authenticating the interface, and allowing firmware upgrades, we can greatly improve the initial security of devices during their factory setup without needing expensive hardware changes. This approach is suitable for standardized batch deployment by manufacturers and integration into product lines. The important strategies can all be used through local settings or OTA methods in experiments, with easy setup, proven technology, and good conditions for practical use.

On the client side, you can set up VLAN network separation, firewall rules, and two-factor authentication using your current home router and the features already available in popular operating systems, so you don't need any extra equipment. Whitelisting access control and lateral isolation policies have been shown in tests to effectively limit how far attackers can get within local networks, making them great for quick use in small homes and offices. Tests have shown that using whitelisting for access control and keeping devices separate from each other can significantly limit how far attackers can get in local networks, making these methods great for quick setup in small homes and offices. The researcher recommends combining security configurations with graphical interface guidance and standardized operation templates, which offer strong user-friendliness and potential for widespread adoption.

In terms of promotion, this type of security mechanism has high replicability and policy adaptability, suitable for a wide range of terminal types such as smart cameras, IoT door locks, and home gateways. By following the IoT security technology guidelines and smart device access standards set by the government, manufacturers can be encouraged to include basic security features in their product designs and use these features as important criteria when assessing smart home platforms. Through a system where manufacturers activate defaults, users can easily configure settings, and platforms provide consistent oversight, the overall cybersecurity protection of smart homes is expected to improve over time. By having manufacturers automatically turn on security features, allowing users to easily set them up, and ensuring the platform oversees everything, we hope to steadily improve the overall cybersecurity of smart homes.

7 Conclusion

7.1 Summary of research results

This thesis systematically investigates the identification and protection mechanisms of security vulnerabilities in smart home camera devices, using TP-Link Tapo C200 and Xiaomi Smart Camera C200 as representative devices. An experimental network environment is constructed, employing techniques such as port scanning, packet sniffing, authentication bypassing, man-in-the-middle attacks, and vulnerability exploitation to conduct in-depth testing and analysis of their communication mechanisms, default configurations, and interface security. The research process identified high-risk issues in TP-Link devices, such as default credentials, plaintext transmission, unauthorized RTSP access, and firmware vulnerabilities. The effectiveness of encryption transmission, access control, and network isolation strategies in actual environments was verified through secure reinforcement experiments.

In terms of methodology, this thesis constructs a hierarchical experimental evaluation system suitable for smart terminal devices, covering three stages: identification, verification, and mitigation. In terms of result presentation, tables and visual diagrams are used to compare the security performance of different devices under default and enhanced states, providing data support for equipment selection, deployment strategies, and optimization of protection mechanisms. For recommendations, practical security setups and design strategies are suggested for both vendors and users based on the experiment results, which can be generally applied and are valuable for promotion. Overall, this study shows possible security problems in popular camera devices when used in real situations and confirms that different affordable and flexible protective measures work well, offering a practical way and theoretical support for creating future safety standards for equipment and applying engineering protection.

7.2 Application value of safety protection suggestions

The security protection suggestions in this thesis not only offer specific strategies to mitigate the important weaknesses found in tests but also have wide-ranging possibilities for being used easily in real life. Vendors can implement mechanisms such as mandatory password changes, pre-set HTTPS communication, RTSP interface permission authentication, and OTA firmware update strategies at the software level. These measures have low technical barriers and strong compatibility, making them suitable for rapid upgrades of existing product lines and secure pre-configuration of future products. Their promotion and implementation can significantly enhance the overall security rating of products, boost user trust, and strengthen market competitiveness.

On the user side, the VLAN network isolation, port whitelist policy, firewall rule settings, and two-factor authentication mechanisms advocated in this thesis do not require specialized equipment or complex deployment. Mainstream home routers, NAS systems, or intelligent gateways can all implement security mechanisms, offering broad adaptability. Once these operations become part of user habits, they can effectively reduce security incidents caused by improper configuration and minimize overall network exposure. Security configurations, when paired with a visual interface and tiered operation templates, can also serve as an important functional module for user security education and service operation platforms, facilitating the transition from "passive product security" to "active user defense."

From an industry perspective, the research findings provide verifiable model paths and engineering configuration recommendations for IoT terminal security governance, with good cross-platform adaptability and policy alignment foundations. In conjunction with the national promotion of intelligent device classification management, IoT security whitelist mechanisms, and data protection regulations, these protective measures can serve as important evaluation criteria for the procurement and integration of smart camera systems into platforms. This promotes overall enhancement of the industrial chain's

security capabilities and lays a practical foundation for building a sustainable smart home security ecosystem.

7.3 Follow-up research direction and prospect

Despite the relatively complete evaluation system formed in this thesis regarding experimental dimensions and protective strategies, there is still room for further research as IoT device types continue to expand and attack methods evolve. In terms of attack models, future work could introduce automated vulnerability mining and simulation platforms, combined with fuzzing, AI traffic recognition, and script-based penetration tools, to build a multi-stage attack chain testing framework that closely mirrors real-world scenarios. This would help assess the equipment's resilience and behavioral stability over long-term operation.

In terms of equipment scope, further research can be expanded from a single camera to multiple types of smart home nodes, such as intelligent door locks, voice assistants, and central control panels. By building a heterogeneous device collaborative test environment, the compatibility risk and chain attack potential of devices from different manufacturers in the same network can be evaluated, filling the gap in the current research on the security of multi-device linkage.

At the platform level, further research can be conducted on security mechanisms in the interaction between devices and cloud services, including remote API verification, identity token management, and data synchronization encryption. Particular attention should be paid to the credibility of platform update mechanisms and the integrity of the device-cloud trust chain. Further studies should look into how using edge computing and local AI recognition models affects security and how we can improve local decision-making processes.

Finally, regarding standardization and governance mechanisms, future research should place greater emphasis on ensuring security compliance throughout the

entire lifecycle of devices. It is essential to promote the development of industry-level security rating evaluation systems and to explore the establishment of open data interfaces for smart device supervision platforms. Additionally, sharing risk control models and enhancing automated security audit capabilities will be crucial in assisting the creation of a dynamic, scalable, and cross-vendor collaborative smart home security protection system.

References

[1] Zou Bo, Yang Jingxuan, Wang Mingxuan, et al. Generative Innovation Mechanism of Smart Connected Products in the Digital Intelligence Era: A Case Study of Midea Smart Home [J/OL]. Nankai Business Review, 1-25 [2025-04-28]. http://kns.cnki.net/kcms/detail/12.1288.F.20250422.1834.006.html.

[2] Li Changqi, He Zhiqin, Zhou Heng, et al. Design and Implementation of a Smart Home Monitoring System Based on Android and WiFi [J]. Modern Electronic Technology, 2020, 43 (20): 67-70. DOI: 10.16652/j.issn.1004-373x.2020.20.017.

[3] Liao L, Liang Y, Li H. A systematic review of global research on natural user interface for smart home system[J]. International Journal of Industrial Ergonomics, 2023, 95: 103445.

[4] Jothi R K ,Vaithiyanathan B. Developing a Hybrid Approach with Whale Optimization and Deep Convolutional Neural Networks for Enhancing Security in Smart Home Environments' Sustainability Through IoT Devices[J]. Sustainability, 2024,16(24):11040-11040.

[5] Bajpai A, Chaurasia D, Tiwari N .A novel methodology for anomaly detection in smart home networks via Fractional Stochastic Gradient Descent[J].Computers and Electrical Engineering,2024,119(PB):109604-109604.

[6] J. S P (Jack) T L ,Traci C . There's No place like home: Understanding users' intentions toward securing internet-of-things (IoT) smart home networks[J]. Computers in Human Behavior,2023,139.

[7] Stanislaw P, Lachlan U ,Derek P M .Defence against the dark artefacts: Smart home cybercrimes and cybersecurity standards[J].Computer Law & Security Review: The International Journal of Technology Law and Practice,2021,42.

[8] Navya S ,Devina V .Smart Home Security Solutions using Facial Authentication and Speaker Recognition through Artificial Neural Networks[J].International Journal of Cognitive Computing in Engineering,2021,2154-164.

[9] Liu Chao, Xu Zhifang, Wang Fangqian, et al. IoT Operating System Application Framework Design for Smart Homes [J]. Modern Electronic Technology, 2020, 43 (23): 143-145+149. DOI: 10.16652/j.issn.1004-373x.2020.23.032.

[10] Zhang Qilong, Chen Xiangping. OneNET Cloud Platform WiFi Remote Control Smart Home System [J]. Modern Electronic Technology, 2020, 43 (14): 25-29. DOI: 10.16652/j.issn.1004-373x.2020.14.007.

[11] Yang Yajun, Chen Xiuzhen, Ma Jin. Research on Smart Home Device Security Certification Scheme Based on Symmetric Polynomials [J]. Computer Applications Research, 2021, 38 (01): 215-217. DOI: 10.19734/j.issn.1001-3695.2019.09.0619.

[12] Saeed M A ,A. M A ,Hussain S C .Cyber security framework for smart home energy management systems[J].Sustainable Energy Technologies and Assessments,2021,46.

[13] Cai Gaiping, Deng Tao, Ni Jun. Design of a Smart Home Fire Alarm System Based on Mobile Client [J]. Fire Science and Technology, 2020, 39 (03): 377-380.

[14] Liao Jing, Li Aiping, Duan Liguo. Service-Oriented Multi-Protocol Smart Home Platform Design [J]. Small and Micro Computer Systems, 2019, 40 (10): 2109-2112.

Appendix: experimental data and results

Appendix 1: Summary table of experimental data and core results

Experiment section	TP-Link Tapo C200 results	Xiaomi Smart Camera C200 Results
Port scanning and service identification	Open port 80/554 with HTTP/RTSP service	Only port 443 is open and the service is HTTPS
Data packet capture and encryption test	HTTP plaintext, RTSP plaintext, no TLS handshake	TLS communication, all encrypted, no sensitive field leakage
Default credentials and unauthorized access	The default password of the Web can be logged in, and RTSP can be accessed anonymously	All interfaces need App authorization, and no bypass path is found
The middleman attack experiment	Credentials and some content are successfully obtained, and there is a risk of hijacking	The intermediary attack failed and the communication integrity was valid
CVE vulnerability exploitation test	The specific version has a certification bypass vulnerability, which is partially successful	No exploitable vulnerabilities are detected, and closed source protocols are difficult to verify
Protection effect after HTTPS is enabled	Encryption login is in effect, and the video stream is still in plain text	Encryption is enabled by default and no data exposure has been seen
Network isolation and access control	The VLAN blocking is successful and the firewall interception is effective	Access is required through the cloud control platform and cannot be accessed through the LAN
Password policy and	Default weak password, no mandatory modification mechanism	The first binding requires a strong password, and the default account is not available

authentication
mechanism

Appendix 2: complete experimental steps and result codeEquations as parts of the text

```

import matplotlib.pyplot as plt
from matplotlib.patches import FancyBboxPatch
import matplotlib

# Set English font (optional if using default)
matplotlib.rcParams['font.sans-serif'] = ['Arial']
matplotlib.rcParams['axes.unicode_minus'] = False

# Create figure and axis
fig, ax = plt.subplots(figsize=(16, 7))
ax.set_title("NIST Cybersecurity Framework", fontsize=16, fontweight='bold')
ax.axis('off')  # Hide axis

# Core Functions and Colors
functions = ["Identify", "Protect", "Detect", "Respond", "Recover"]
colors = ["#5DADE2", "#58D68D", "#F4D03F", "#EB984E", "#AF7AC5"]

# Sub-functions (translated from original Chinese version)
subfunctions = {
    "Identify": ["Asset Management", "Risk Assessment", "Governance"],
    "Protect": ["Access Control", "Data Security", "Awareness Training", "Protective Technologies"],
    "Detect": ["Anomaly Detection", "Security Event Identification", "Continuous Monitoring"],
    "Respond": ["Response Planning", "Communication", "Mitigation Measures"],
    "Recover": ["Recovery Plan", "Improvement Strategy", "Business Continuity"]
}

# Drawing parameters
x_start = 1
y_main = 4.5
box_width = 2.5
box_height = 1

# Draw each core function and sub-items
for idx, func in enumerate(functions):
    x = x_start + idx * (box_width + 0.8)
    main_box = FancyBboxPatch(
        (x, y_main), box_width, box_height,
        boxstyle="round,pad=0.1", edgecolor="black",
        facecolor=colors[idx]
    )
    ax.add_patch(main_box)
    ax.text(x + box_width / 2, y_main + box_height / 2, func,
            ha='center', va='center', fontsize=12, fontweight='bold')

    sub_items = subfunctions[func]
    for j, sub in enumerate(sub_items):
        ax.text(x + box_width / 2, y_main - (j + 1) * 0.7,
                sub, ha='center', va='center', fontsize=10)

# Axis limits
ax.set_xlim(0, 20)
ax.set_ylim(0, 7)

plt.show()

```

```python

import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

# Set English font (optional)
plt.rcParams['font.sans-serif'] = ['Arial']
plt.rcParams['axes.unicode_minus'] = False

# Data
data = {
    "Device": ["TP-Link C200", "TP-Link C200", "Xiaomi C200", "Xiaomi C200"],
    "Encryption Status": ["Unencrypted", "HTTPS Enabled", "Default", "Default"],
    "Interception Rate (%)": [100, 15, 0, 0]
}

df = pd.DataFrame(data)

# Plot
plt.figure(figsize=(8, 5))
sns.barplot(data=df, x="Device", y="Interception Rate (%)", hue="Encryption Status")

# Customize plot
plt.title("Comparison of Sensitive Information Interception Rate", fontsize=14)
plt.ylabel("Interception Rate (%)", fontsize=12)
plt.xlabel("Device Model", fontsize=12)
plt.ylim(0, 110)
plt.grid(axis='y', linestyle='--', alpha=0.6)
plt.legend(title="Encryption Status")

plt.tight_layout()
plt.show()
```

```python

import matplotlib.pyplot as plt
from matplotlib.patches import FancyBboxPatch

# Define experiment steps
steps = [
    ("Port Scan", "Open 80/554 (TP-Link), 443 (Xiaomi)"),
    ("Packet Capture", "HTTP/RTSP unencrypted (TP-Link), Full TLS (Xiaomi)"),
    ("Credential Test", "Default password valid (TP-Link), Binding enforced (Xiaomi)"),
    ("MITM Attack", "Credentials intercepted (TP-Link), Blocked (Xiaomi)"),
    ("Exploit Test", "Partial CVE success (TP-Link), No CVE matched (Xiaomi)"),
    ("HTTPS Evaluation", "Login protected, RTSP still plain (TP-Link), Fully encrypted (Xiaomi)"),
    ("Network Isolation", "Firewall blocks effective (both)"),
    ("User Config Test", "Weak default settings (TP-Link), Strong defaults enforced (Xiaomi)")
]

# Set up figure
fig, ax = plt.subplots(figsize=(12, 7))
ax.set_xlim(0, 10)
ax.set_ylim(0, len(steps) + 1)
ax.axis("off")

# Draw boxes and add text
for i, (title, desc) in enumerate(steps):
    y = len(steps) - i
    box = FancyBboxPatch((1, y - 0.4), 8, 0.8, boxstyle="round,pad=0.1",
                         edgecolor="black", facecolor="#AED6F1")
    ax.add_patch(box)
    ax.text(1.2, y, f"{i+1}. {title}", fontsize=10, va='center', weight='bold')
    ax.text(2.5, y, desc, fontsize=10, va='center')

# Add title
plt.title("IoT Camera Security Experiment Flow & Results Summary", fontsize=14, weight='bold')
plt.tight_layout()
plt.show()
```

```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

# Data for isolation and access control effectiveness
data = {
    "Scenario": ["Default Network", "Access Control Enabled", "VLAN Isolation", "Firewall Enabled"],
    "Attack Success Rate (%)": [100, 60, 20, 10]
}

df = pd.DataFrame(data)

# Plotting
plt.figure(figsize=(8, 5))
sns.barplot(data=df, x="Scenario", y="Attack Success Rate (%)", palette="Set2")

# Customize plot
plt.title("Effectiveness of Network Isolation and Access Control", fontsize=14)
plt.ylabel("Attack Success Rate (%)", fontsize=12)
plt.xlabel("Security Strategy", fontsize=12)
plt.ylim(0, 110)
plt.grid(axis='y', linestyle='--', alpha=0.6)

plt.tight_layout()
plt.show()
```

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np

# Security scores data
data = {
    "Device": ["TP-Link C200", "Xiaomi C200"],
    "Encryption Score": [3, 10],
    "Access Control Score": [4, 9],
    "Overall Security Score": [3.5, 9.5]
}

df = pd.DataFrame(data)
df_scores = df.set_index("Device").T

# Radar chart setup
labels = list(df_scores.index)
num_vars = len(labels)
angles = [n / float(num_vars) * 2 * np.pi for n in range(num_vars)]
angles += angles[:1]

# Plotting
fig, ax = plt.subplots(figsize=(6, 6), subplot_kw=dict(polar=True))

for device in df_scores.columns:
    values = df_scores[device].tolist()
    values += values[:1]
    ax.plot(angles, values, label=device)
    ax.fill(angles, values, alpha=0.1)

# Chart aesthetics
ax.set_theta_offset(np.pi / 2)
ax.set_theta_direction(-1)
ax.set_thetagrids(np.degrees(angles[:-1]), labels)
ax.set_ylim(0, 10)
plt.title("Security Capability Comparison", fontsize=14)
plt.legend(loc='lower right')
plt.tight_layout()
plt.show()
```