https://doi.org/10.53360/2788-7995-2025-2(18)-13

МРНТИ 81.93.29; 50.33.33

BY 4.0
Check for updates

L. Rzayeva[1], G. Abitova*[1], K. Niyazaliyev[1], A. Baitulakov[1], V. Nikulin[2]
[1]Astana IT University
010000, Republic of Kazakhstan, Astana, Mangilik El Avenue, 55/11
[2]State University of New York, NY, USA
4400 Vestal Pkwy E, Binghamton, NY 13902, United States
*email: abitova.gul@gmail.com

ANALYSIS OF METHODS FOR ENCRYPTING VIDEO RECORDER FILE SYSTEMS TO OPTIMIZE DIGITAL FORENSICS RESEARCH METHODS

Abstract: *Mobile phones have become not only a means of communication, but also significant sources of information. Despite the development of encryption technologies and methods, it remains important to develop special and obtain new knowledge, search, develop and improve methods and approaches to the study of mobile devices. This study solves problems aimed at solving problems of optimizing the data analysis and processing system, as well as effective decision-making during the investigation of digital forensic experts based on artificial intelligence technologies and machine learning algorithms. In view of this, as well as in order to create new methods and algorithms, a deep and comprehensive analysis and review of the encryption methods of video recorder file systems for optimizing digital forensics research methods was carried out. The analysis and review of the video recorder file system made it possible to justify the choice and development of a methodology for detecting encryption and restoring video data from digital video recorders and mobile devices. The review defines the structure and mechanism of the file system. The conducted analysis and review are useful for forensic experts and digital forensics experts in analyzing digital evidence related to video surveillance. This analysis and review were the first stage in the implementation of a scientific project on digital forensics.*

Key words: *mobile phones, encryption technologies and methods, optimizing the data analysis, digital forensics, video recorder file systems, methodology for detecting encryption, artificial intelligence technologies.*

Introduction

Introduction. Mobile digital forensics play an important role in crime investigation in the modern world. Mobile phones have become not only a means of communication, but also significant sources of information. Despite the development of encryption technologies and methods, it remains important to develop specialized and obtain new knowledge, search, develop and improve methods and approaches to mobile device research. By continuing to develop their skills, forensic specialists will be able to effectively use the opportunities provided by modern technologies and innovative methods and digital tools.

This project addresses the challenges of optimizing the data analysis and processing system, as well as effective decision-making during digital forensic investigations based on artificial intelligence technologies and machine learning algorithms. The proposed approaches combine advanced methods for extracting existing and restoring deleted information, as well as optimizing the analysis of the obtained data using artificial intelligence, thoroughly tested in various possible scenarios for effective subsequent decision-making.

The analysis and review of the video recorder file system provided the basis for the selection and development of a methodology for detecting encryption and recovering video data from digital video recorders and mobile devices. The review defined the structure and mechanism of the file system. The analysis and review are useful for forensic experts and digital forensic experts in analyzing digital evidence related to video surveillance.

To create new methods and algorithms, a detailed review of file system encryption methods for various video recorders and mobile devices was also conducted, which considered the basic principles of operation and basic transformations. The obtained results confirm the relevance and prospects of research areas in the field of digital forensics.

Main Parts:

Review of the Video Recorder File System

Video recordings of CCTV systems are the most useful evidence in forensic activities. However, it is difficult to analyze the hard drive using the recorder's file system, since there is

currently no relevant research. This study determines the structure and mechanism of the file system of the HiWatch DS-H104G recorder.

The presented study examines in detail the file system of the HiWatch DS-H104G video recorder. Key components of the file system, such as the «Master Sector» (MBR – Master Boot Record), «System Logs», «Video Data Area» and «HIKBTREE», are identified. Understanding the structure and mechanism of operation of these components allows us to develop methods for effective extraction and recovery of video data.

With the increasing use of video surveillance systems, the importance of video recordings as evidence in judicial practice also increases. However, the analysis and recovery of video data from digital video recorders (DVR) presents significant difficulties due to the use of proprietary file systems by manufacturers. This makes it difficult to use standard methods and tools for data extraction [1].

Many DVR manufacturers develop their own proprietary file systems. This is due to several reasons:

1. Performance optimization – specific file systems allow you to optimize the speed of writing and reading data, ensuring stable operation during continuous recording from multiple cameras.
2. Data Security – Native file systems make it difficult for unauthorized access to video data, increasing security;
3. Resource management – optimizes disk space usage through cyclic writing and efficient data fragmentation management.

Many DVRs use a block structure to store data. Video data is recorded in fixed-size blocks, making it easy to manage loop recording and overwriting of old data.

Video recordings are often encoded using efficient codecs such as H.264 or H.265. This allows for a reduction in the amount of data stored without loss of quality. Special container formats adapted to a specific device may be used within the file system.

To provide quick access to video recordings by time or event, special index structures are used. They can contain information about key frames, recording start and end times, channel numbers, and other parameters.

As for the HiWatch DS-H104G recorder and research methods, this is a four-channel video recorder for analog cameras (four BNC input connectors), and you can additionally connect two IP cameras, the recorder does not provide PoE power. Three analog cameras were connected to the recorder, one TVI format and two PAL format cameras. A 120 Gb SSD hard drive was installed. Video recording settings were enabled for continuous recording at a speed of 24 frames per second.

The recorder was flashed with the new, latest firmware. The firmware of the recorder was done by unsoldering the EEPROM memory chip 25Q128FVSG. The old firmware was read by the MiniPro programmer using the SOIC8/SOP8-DIP adapter with spring clips, then the chip was erased and flashed with the new firmware. The password 1q2w3e4r was set and the graphic password was set to Z. In the recorder settings, when requesting information about the system, the following data is displayed: 1. serial number 0420170526AAWR770642404WCVU, 2. firmware version V3.4.80, Build 170225, 3) confirmation code RRRWAR, 4) board version 0x9fa00 [1, 2].

The version of the board DS-80237 rev.5.2 is written on the recorder board. Four microcircuits are soldered on the board:

1. EEPROM memory chip 25Q128FVSG in SOIC8 package, it stores the device firmware.
2. TP2853 from TechPoint is a HD-TVI 3.0 video surveillance receiver chipset, which is a 4-channel multi-standard 3–8-megapixel receiver with support for 2 BT.656 outputs, designed to process video signals from video surveillance cameras.
3. nt5cb128m16ip-ek DDR3 SDRAM synchronous dynamic random-access memory. This is the RAM, works together with the CPU.
4. The processor itself has a glued radiator, upon removal of which there are inscriptions KY-2015-7.

When installing a hard drive in a recorder, it requires its initialization, because of which it writes its file system to it. The first 512 bytes (MBR sector) are written with only 0x00, which makes it unknown to common file systems, the recorder uses its own file system, presumably to improve the efficiency of video management and copy protection.

As a result of the study of the HiWatch recorder file system structure, it was revealed that all video data is placed in a data structure called a data block record, which contains time records, channels, and initial locations of the data block. Video data blocks can be found using a structure called HIKBTREE. The structure of the video recorder file system consists of four physical sections.

The first section, «Master Sector», contains information about the general structure of the file system. The second section contains «System Logs», which store information about events and the state of the DVR. The third section, «Video Data Area» has numerous data blocks for storing video data. The fourth section, HIKBTREE, contains video data metadata, including time records and others.

The «Master Sector» contains information about the overall structure of the file system. This area starts at offset 0x200, and the size of the master sector is 256 bytes. The file system signature values are «HIKVISION@HANGZHOU (0x48 49 4B 56 49 53 49 4F 4E 40 48 41 4E 47 5A 48 4F 55)», as shown in Figure 1.

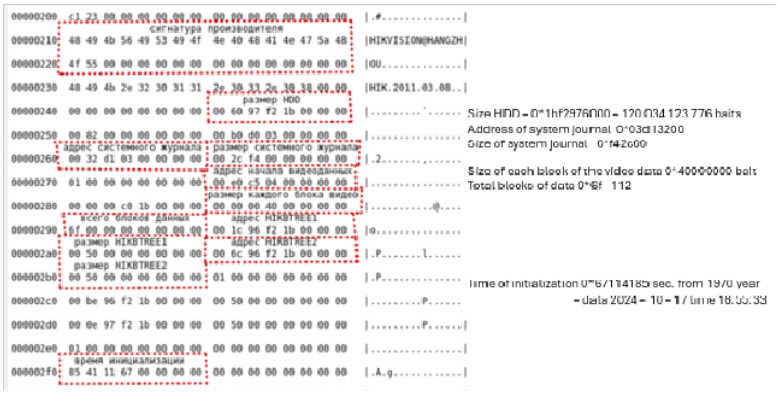

Figure 1 – File system signature values Source: completed based on [1]

HIKVISION video recorder. At the end of the video data block, there is an IDR table, which records the addresses of the beginning of the video recording segments with a duration of one second. Each entry of the IDR table is written in the direction of decreasing offset from the end of the data block. It starts with the signature «OFNI (0x4F 46 4E 49)» and has a fixed size of 56 bytes for each entry. By comparing the IDR table timestamps stored in the entries of the data block, it is possible to check the time of the IDR images. In general, a series of video data is stored in the same data block when it continues recording [1, 2].

HIKBTREE contains metadata of each video data in data blocks. HIKBTREE is a fundamental area for detecting data block offset, presence of video data and other additional information about records. Since it has a signature value of 'HIKBTREE (0x48 49 4B 42 54 52 45 45)', the term HIKBTREE will be used. The backup HIKBTREE is located after the previous one.

When the data volume exceeds the capacity of the hard disk, the old video data is overwritten with new data. Old video data may sometimes be saved together with new data in a data block if the device is suddenly stopped or turned off. When video data is overwritten, the following changes: the «Channel» and «Record Start/Stop Time» values in the data block entry. «Channel» is updated as the current channel, and «Record Start/Stop Time» is changed to «0xFF FF FF 7F 00 00 00 00» – 'start 2038-01-19 03:14:07' 'stop 1970-01-01 00:00:00'.

CCTV footage is the most useful evidence in forensic work. However, it is difficult to analyze a hard drive using the HIKVISION file system, as there is currently no relevant research. It is important to evaluate proprietary file systems, otherwise valuable digital evidence may lose its evidentiary value. Therefore, it is necessary to identify unknown file systems.

Thus, this review defines the structure and mechanism of the HIKVISION file system, which are not well known. Using the results of this analysis, forensic experts and digital forensic experts can analyze the hard drives of HIKVISION products with the integrity of digital evidence. In addition, the case analysis procedure can be useful to counteract anti-forensic actions such as system initialization or data overwriting. This review is conducted to provide useful results of the HIKVISION file system analysis for forensic experts and digital forensic experts in analyzing digital evidence related to video surveillance.

Overview of File System Encryption Methods for Hikvision and Dahua Video Recorders

Hikvision and Dahua are the world's leading manufacturers of video recorders (DVR/NVR), which are widely used in video surveillance systems. These devices are an important target for protection against unauthorized access, as they store large amounts of video data. Both systems have similar encryption methods that they use in their file systems.

If we talk about similar approaches to encryption, we can say that both systems encrypt data at the device level, encrypt data streams, and protect configuration data. When encrypting data at the device level, both systems use encryption algorithms to protect information on hard drives, as well as flash drives. These algorithms prevent unauthorized access in the event of loss or theft of the device. In addition, both systems use encryption already during recording and data transfer, which helps to avoid data leakage at intermediate stages.

When it comes to encryption of data streams, records can be encrypted before being saved to a hard drive or flash drive. This also provides additional security in the event of an attempt at unauthorized access to the storage medium, because it may be impossible to decrypt the data without the encryption key. At the same level, protection against interception and analysis of video streams occurs, which also provides a high level of security for video recordings. These methods reduce the risk of data compromise. Another similarity between the two surveillance systems is the protection of configuration files. Passwords, access keys and configurations are stored in encrypted form, the use of such measures also reduces the risk of replacement/change of configuration data, as well as their abuse. Thus, a comprehensive approach to encryption of data and configurations of both systems provides a high degree of security and increases resistance to threats [1, 3].

HikVision file systems use specific encryption methods that also provide maximum data protection. The system uses several encryption methods that include AES, TLS/SSL, hardware encryption and many more, which provide a high level of data security on HikVision devices. AES improves protection against loss or theft of storage media, using 128/256-bit encryption keys, which are well suited for encrypting video recordings and other data on the device. TSL/SSL is used for data transmission, providing encryption when accessing the device remotely over the network.

Within the framework of the file system under consideration, it is used to protect data when broadcasting video streams to client devices. The implementation of TSL/SSL protects against most network threats, plus it provides secure device management and viewing of video streams, which is especially important for video surveillance systems. The HikVision file system also uses hardware and cloud encryption. Hardware encryption is not implemented in all models, but it is designed to improve performance, as well as the use of specialized cryptographic processes.

It is important to note that in addition to everything, Hik-Connect services encrypt data before transferring it to the cloud, which also adds protection against data interception. HikVision devices also use TPM (Trusted Platform Module), which serves for secure storage of keys and increases protection for authentication. HikVision file systems combine many encryption methods, which increase data protection several times, all methods are aimed at secure storage and protection of data.

Dahua file system encryption methods are also focused on maximum data protection. They include AES-256, SHA (Secure Hash Algorithm), SSL/TLS, hardware encryption, P2P connection encryption, and proprietary algorithms. AES-256 provides reliable data encryption both on storage media and during network transmission, which demonstrates a high level of security. The SHA algorithm is used to check the integrity of data, with its help you can track whether a file has been changed or whether there has been an attempt to forge it. SSL/TSL creates secure channels between devices and clients, thanks to which it can prevent interception or substitution of transmitted information.

Dahua systems have implemented a comprehensive approach to data encryption and maximum data security. Hardware encryption reduces the load on the main processor and increases performance when encrypting data. P2P encryption, in turn, provides secure remote access via SmartPSS applications and the Dahua mobile application, and encryption of data before sending it to the network prevents its interception. It is worth remembering that some systems also have their own algorithms, which provide additional data protection. Together, all these methods guarantee a high level of confidentiality, integrity and security of data.

Thus, despite all this, these file systems have some shortcomings or points that need to be considered. For example, key management requires greater responsibility in terms of the security of storage and transmission of keys, since the reliability of encryption heavily depends on this. If the keys are compromised, the data becomes vulnerable. In addition, on some models with limited resources (some budget models), encryption can cause delays in recording or playback. Devices of both HikVision and Dahua file systems have been attacked due to vulnerabilities in software. There have been situations with password leaks and weak authorization mechanisms.

In general, the encryption methods of HikVision and Dahua are very similar to each other, the comparison of encryption methods is summarized in Table 1.

Table 1 – Comparison of Hikvision and Dahua encryption methods

Parameter	Hikvision	Dahua
Basic algorithm	AES-128/256	AES-256
Data transfer	TLS/SSL	TLS/SSL
Hardware acceleration	Used in high-end models	Used in high-end models
Cloud encryption	Hik-Connect	SmartPSS
Key management	TPM, embedded solutions	Hardware and software methods
Source: compiled by the author based on [1-2]		

Thus, based on all the above, it is shown that HikVision and Dahua use similar approaches to AES encryption and TLS technology for data transfer. HikVision, in turn, focuses on working with cloud services using encryption for data security. And Dahua video recorders offer broader support for hardware data encryption, making the devices of this system more effective in protecting data with limited resources.

Methodology

This study is represented by the analysis, comparison, synthesis and review of the methods of encryption of file systems used in Android operating systems (versions 7 and above) and iOS (versions 8 and above). The study does not affect devices with outdated versions of operating systems, as well as encryption methods, implemented exclusively at the level of third-party software. The main attention is paid to build-in data protection mechanisms and their influence on the possibility of digital forensics.

The research with review and analysis is based on the main encryption methods. There are two main encryption methods: full -disking encryption (FDE – Full Disk Encryption): All data on the device are encrypted with one key. Poprol encryption (FBE – File BASDED ENCRYPTION): Data is encrypted by files with different keys, which allows them to decrypt independently.

The use of encryption in mobile devices: Apple - full -disking encryption was used from iOS 3 to iOS 8, with iOS 8, pupillid encryption was introduced. Android – full-disking encryption was used with Android 4.4 and on some devices to Android 9, puffy encryption acts from Android 7 to Android 14.

Methods of obtaining data from «iOS» devices: backup ITUNES-is considered expanded logical data extraction. The full file system through the vulnerability of the loading mode (Bootrom Exploit) is the Checkm8 method, which supports the iPhone versions on A5 chips to A11 (iPhone 4s to iPhone X). Removing the full file system through the SSH (Secure Shell) protocol allows you to get deeper access to the operating system and its data. To use this method, it is necessary to conduct a jailbreak, which provides unlimited access to the iOS operating system.

Methods of obtaining data from «Android» – Throsa [3]: logical extraction by copying data; Data extraction using the vulnerabilities of the mobile device processor according to the manufacturer: Unisoc (previously Spreadtrum), Qualcomm, MTK, Samsung Exynos, Huawei Kirin; Android Physical, Full File System-the method allows to extract the physical image of the memory of the mobile device on old versions of Android (to Android 6), by obtaining a temporary Root-right; Backup ADB backup; The method of lowering the application version to extract data from them APK Download.

Analysis of Encryption Methods for Video Recorder File Systems

1) Analysis of file system encryption methods of Dahua Technology video recorders

Encryption of useful data of video surveillance systems manufactured by Dahua Technology is carried out at all available levels and processes: from data transmission via the local video surveillance network to their storage and transmission to the security post, as described in the relevant documentation on the security of video surveillance systems (Dahua Product Security, 2024). Also, this document contains a description of the video surveillance system diagram with a central key management server (Key Management Server) (Figure 2).

Accordingly, such a system consists of end devices (in the diagram – a video surveillance camera, which is equipped with temporary data storage in case of failure of the central recording device), the central recording device itself, a key management server and an operator. Data from the video camera is saved to temporary storage in encrypted form and is also transmitted over the local network using frame encryption. The central recording device records the encrypted data received on hard drives (Figure 3).

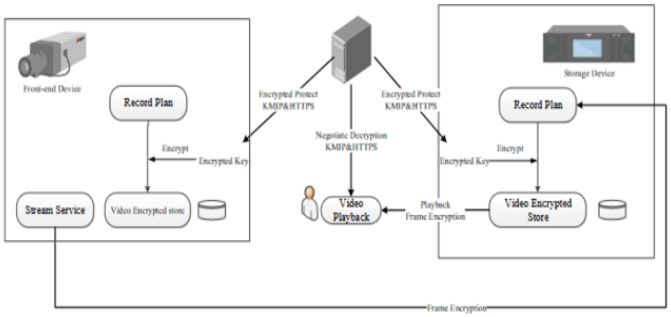

Figure 2 – Scheme of a video surveillance system with a central key management server
Source: completed based on [2]

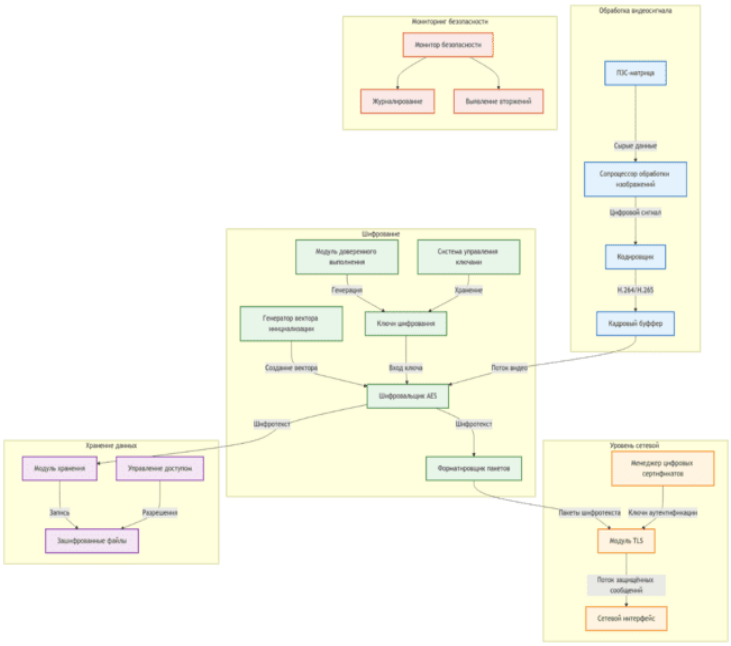

Figure 3 – Data flow graph in a video surveillance system with encryption
Source: completed based on [3, 4]

At the same time, the operator still has access to view the broadcast of the video camera signal, since the key management server provides the ability to issue an encryption key to the operator's device for viewing cameras in real time (the real-time video stream is encrypted by the central device itself, the key management server acts as an intermediary in the key agreement procedure) [1, 2, 11].

Important notes: The series can support basic HTTPS connection, but without full encryption of the video stream. Specific encryption features may vary depending on the firmware version. Enterprise models support all modern security protocols. Professional series has limited support for some encryption features. [2, 12].

The negotiation of encryption keys occurs using the KMIP protocol. KMIP is a standardized communication protocol developed by OASIS to provide interoperability between different cryptographic key management systems. The main goal of the protocol is to provide a unified way to exchange cryptographic material between clients and key management servers, regardless of the hardware or software manufacturer.

The protocol architecture is based on the client-server model, where the server acts as a centralized repository and manager of cryptographic keys, and the clients are various applications and devices that need access to key information. This architecture allows for a single point of control and management of all cryptographic materials in an organization. Figure 4 provides a typical diagram of the interaction between a client and a server.

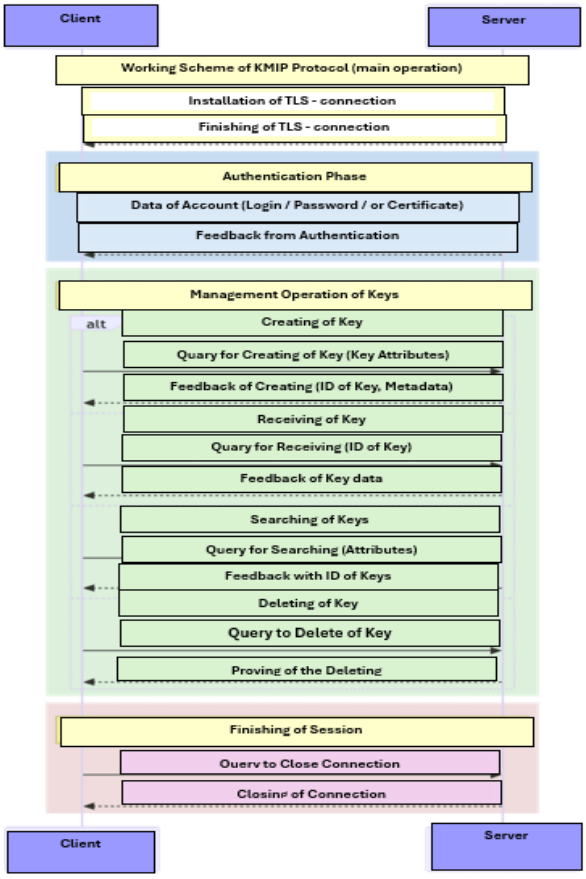

Figure 4 – KMIP Parties Interaction Sequence Diagram Source: completed based on [3-4]

The most important aspect of KMIP is the lifecycle management of keys. Each key in the system goes through several states: from creation to destruction. The initial state of the key is pre-active, when the key is created but not yet used. The key then moves to the active state, where it can be used for cryptographic operations. Upon expiration or if necessary, the key can be deactivated, which prevents its further use. In the event of a compromise, the key is marked as compromised, which requires immediate cessation of its use and possible replacement. The final stage is the destruction of the key, after which its recovery becomes impossible.

The protocol's security is ensured by several levels of protection. At the transport level, TLS is required to encrypt all traffic between the client and the server. Authentication can be performed using X.509 certificates or other mechanisms, such as login/password. An important feature is the support of mutual authentication, when not only the server verifies the client's authenticity, but the client also verifies the server's authenticity.

All operations in KMIP are strongly typed and documented. The protocol supports a wide range of cryptographic operations, including key creation, derivation, search, and destruction. When creating a key, you can specify many attributes that define its characteristics and usage policies. Attributes can include validity, allowed operations, cryptographic algorithms, and other parameters.

The protocol places special emphasis on auditing and access control. Each operation with keys is logged, allowing their use to be tracked and potential security breaches to be identified. The access control system is based on a role model, where each user or application is assigned specific rights to perform operations with keys.

KMIP is widely used in enterprise environments to centrally manage encryption keys. This is especially important as encryption becomes increasingly used to protect data both at rest and in transit. In cloud environments, KMIP allows organizations to retain control over their cryptographic keys even when using external storage services [1, 2, 4, 12].

Modern KMIP implementations provide high scalability and fault tolerance. KMIP servers can operate in cluster mode, supporting data replication and load balancing. This allows processing large volumes of requests and ensuring continuous operation even if individual system components fail.

Integrating KMIP with existing systems is simplified by supporting common cryptographic standards and interfaces such as PKCS#11, JCE, and Microsoft CAPI. This allows organizations to gradually migrate to KMIP without having to replace their entire infrastructure at once.

In the context of the Internet of Things (IoT), KMIP provides mechanisms for securely managing keys for multiple devices. This includes initial key loading, key renewal, and revocation if necessary. The protocol allows centralized control of device cryptographic materials throughout their lifecycle.

The protocol's extensibility is ensured by the ability to define custom profiles and attributes. Organizations can tailor KMIP to their specific requirements by adding their own data types and operations while maintaining compatibility with the base protocol.

The primary benefit of using KMIP is to reduce the complexity and cost of managing cryptographic keys within an organization. Instead of maintaining multiple disparate key management systems, organizations can use a single, standardized protocol for all their cryptographic material management needs.

The actual encryption of data during its storage is performed using the AES-256 algorithm. The operating mode is not specified, but it can be hypothesized that the GCM mode (Galois/Counter Mode) is used.

AES is a symmetric block cipher adopted as the encryption standard by the US government in 2001. The algorithm operates on 128-bit blocks of data and supports keys of 128, 192, or 256 bits, which determines the number of transformation rounds (10, 12, or 14, respectively). Using a shorter key length slightly reduces the strength of the algorithm [1, 3, 4, 15].

Basic principles of operation. AES is based on the principle of a substitution-permutation (SP) network, where data is passed through a sequence of nonlinear and linear transformations. The input data is organized into a 4x4 byte state matrix, on which all operations are performed.

Basic transformations. SubBytes (Byte Substitution) is a non-linear substitution operation where each byte of the state is replaced by the corresponding value from a fixed substitution table (S-box, Table from [1, 2, 16]).

The S-box is built based on multiplicative inversion in the Galois field GF (2^8), affine transformation over bits, a special structure to counter cryptanalytic attacks. ShiftRows is a linear transformation in which the status rows are cyclically shifted to the left by a different number of positions:

- The first line remains unchanged.
- The second row is shifted one byte to the left.
- The third line is two bytes.
- The fourth one is three bytes.

MixColumns is a linear transformation that operates on each state column independently [3, 6]:

- each column is multiplied by a fixed matrix in the field GF(2^8).
- and special coefficients {02}, {03}, {01}, {01} are used.
- provides strong diffusion between bytes within each column.

AddRoundKey (Add Round Key) is a simple bitwise XOR operation of the current state with the round key [1, 2, 17-18]:

- Each state byte is combined with the corresponding round key byte.
- Ensure the introduction of key material into the encryption process.
- The only operation that uses an encryption key.

Round structure:

- Initial round: only AddRoundKey operation with initial key.
- Main rounds (9, 11 or 13 rounds depending on the key length): SubBytes, ShiftRows, MixColumns, AddRoundKey.
- Final round: SubBytes, ShiftRows, AddRoundKey (without MixColumns).

Algorithm security:

The cryptographic strength of AES is ensured by a combination of several factors:

- with strong nonlinearity of the SubBytes operation.
- efficient diffusion through ShiftRows and MixColumns.
- d the remaining number of rounds for complete mixing.
- mathematically sound design of all components.

There are currently no feasible attacks on full-round AES when implemented correctly. The main vulnerabilities are related to side-channel attacks on specific implementations, rather than cryptographic weaknesses in the algorithm itself [7, 8, 18, 19].

2) Analysis of encryption methods of file systems of Hikvision video recorders

An analysis of available sources of Hikvision (Hangzhou Hikvision Digital Technology Co., Ltd) [3, 10, 14] video surveillance system security documentation and user manuals for the most common digital video recorders did not reveal any mention of data encryption mechanisms during storage. Data protection functions on storage devices in this manufacturer's video recorders are limited to prohibiting file deletion and overwriting (during cyclic recording), switching individual hard drives to "read-only" mode, and setting up backup. However, IP cameras do have a backup SD card encryption function [4, 13]. According to the manufacturer, a proprietary encryption algorithm is used [5, 17, 20].

Also, the list of compatible hard drives indicates that the use of hard drives with hardware encryption is not supported [6]. Data encryption on the NVR / DVR is present only when transferring data for remote access using the Hik - Connect (Android , iOS) [7] and iVMS (Windows , macOS) [8-10, 20, 29] applications. At the same time, data encryption when saving is performed in the Hikvision cloud service, which uses the Amazon AWS cloud platform [9-10, 21-24].

It is necessary to consider the possibility of deliberate concealment of information about the use of encryption algorithms and the deliberate absence of user settings, and, accordingly, the absence of mentions of them in the instructions. To test this hypothesis, it is necessary to analyze the hard disk image of the system under study [11-13, 25-28, 29-30].

Conclusion

Based on the analysis of video surveillance systems (Dahua, Hikvision), it was found that multi-level data processing must be provided to create an effective decryption module. At the transport level, it is necessary to implement full support for TLS 1.2/1.3 with processing of various cryptographic suites, management of X.509 certificates and correct processing of the TLS record layer. Particular attention should be paid to support for Perfect Forward Secrecy and secure processing of session keys, as well as protection against side-channel attacks when performing cryptographic operations.

At the file system level, the system must ensure decryption of data in the AES-256-XTS (XEX Tweakable Block Cipher with Ciphertext Stealing) format with correct processing of initialization vectors and authentication tags. AES-256-GCM is used for transmission over the network. The key components are mechanisms for restoring encryption keys through interaction with the TPM module, processing user passwords via PBKDF and receiving keys from the Key Management Server via the KMIP protocol. The system performance must be ensured by using AES-NI hardware acceleration and the ability to parallelize data blocks, while it is necessary to guarantee safe memory clearing after using the keys and full logging of all decryption operations. When working with HikVision video recorder systems, these requirements are not set, but future challenges are possible when new lines from the manufacturer are released.

References

1. Dahua Technology. Dahua Product Security White Paper. Dahua Technology. URL: https://material.dahuasecurity.com/uploads/soft/20240531/Dahua-Product-Security-White-Paper-V3.0.pdf (date of access: 24.11.2024).
2. Dahua Technology. Dahua Product Security White Paper. Dahua Technology. URL: https://material.dahuasecurity.com/uploads/soft/20240531/Dahua-Product-Security-White-Paper-V3.0.pdf (date of access: 24.11.2024). – С. 24.
3. Hangzhou Hikvision Digital Technology Co., Ltd. NVR Security Guide. Hikvision. URL: https://www.hikvision.com/content/dam/hikvision/en/cybersecurity/NVR%20Security%20Guide.pdf (date of access: 24.11.2024).
4. Hangzhou Hikvision Digital Technology Co., Ltd. Network Camera Firmware Version: V5.6.10 Release Note. Hikvision. URL: https://www.hikvisioneurope.com/eu//portal/portal/Technical%20Materials /00%20%20Network%20Camera/00%20%20Product%20Firmware/H3%20platform (5xxx,7xxx,6924,6DX4,8426)/DS-2CD7XXX/V5.6.10_Build190919(Released)/IPC%20H3%20V5.6.10%20Release%20Note--External.pdf (date of access: 24.11.2024)], [Hangzhou Hikvision Digital Technology Co., Ltd. UD21022B-E Network Camera User Manual. Hikvision. URL: https://www.hikvision.com/content/dam/hikvision/products/S000000001/S000000002/S000000016/

S000000027/OFR000044/M000051183/User_Manual/UD21022B-E_Network-Camera_User-Manual_5.5.111_20240204.pdf (date of access: 24.11.2024), C. 38.
5. Hangzhou Hikvision Digital Technology Co., Ltd. What encryption standard does Hikvision use for SD cards? Hikvision. URL: https://supportusa.hikvision.com/support/solutions/articles/17000129431-what-encryption-standard-does-hikvision-use-for-sd-cards- (date of access: 24.11.2024).
6. Hangzhou Hikvision Digital Technology Co., Ltd. HDD Compatible List for Hikvision DVR/NVR. Hikvision. URL: https://www.hikvision.com/content/dam/hikvision/en/support/notice/HDD-Compatible-List-for-Hikvision-DVR-NVR_20240315.pdf (date of access: 24.11.2024), c. 7.
7. Hangzhou Hikvision Digital Technology Co., Ltd. Enabling/Disabling Video and Image Encryption. User Authentication Center. URL: https://www.hik-connect.com/views/terms/helpAndroid/Hik%20Enabling%20or%20Disabling%20Video%20and%20Image%20Encryption.html (date of access: 24.11.2024).
8. SpyCameraCCTV. Hikvision IVMS-4200 – Stream is Encrypted. SpyCameraCCTV Helpdesk. URL: https://gethelp.spycameracctv.com/en-US/hikvision-ivms-4200---stream-is-encrypted-277209 (date of access: 24.11.2024).
9. Hangzhou Hikvision Digital Technology Co., Ltd. Hikvision Cloud-Based Platform Security White Paper. Hikvision. URL: https://pinfo.hikvision.com/hkwsen/unzip/20241029114559_91608_doc/Hikvision%20Cloud-Based%20Platform%20Security%20White%20Paper_20241027.pdf (date of access: 24.11.2024).
10. Android Open-Source Project. Trusty TEE | Android Open-Source Project . Electronic resource. URL: https://source.android.com/docs/security/features/trusty (date of access: 25.11.2024).
11. Android Open-Source Project. Support Direct Boot mode | Security | Android Developers . Electronic resources. URL: https://developer.android.com/privacy-and-security/direct-boot (accessed: 25.11.2024).
12. Android Open-Source Project. File-based encryption | Android Open-Source Project . Electronic resource. URL: https://source.android.com/docs/security/features/encryption/file-based (accessed: 25.11.2024).
13. Zebra Developer Portal. Android™ 13 Security Overview: Direct Boot, FBE and more... . Electronic resources. URL: https://developer.zebra.com/blog/androidtm-13-security-overview-direct-boot-fbe-and-more (date of access: 25.11.2024).
14. NIST. Recommendation for Block Cipher Modes of Operation: the XTS-AES Mode for Confidentiality on Storage Devices. NIST SP 800-38E . Gaithersburg: NIST, 2010. 12 p. Electronic resources. URL: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38e.pdf (accessed: 25.11.2024).
15. Crowley P. Adiantum: length-preserving encryption for entry-level processors / P. Crowley, E. Biggers // IACR Transactions on Symmetric Cryptology. – 2018. – P. 39-61. Electronic resources . https://doi.org/10.46586/tosc.v2018.i4.39-61 (date accesses : 11/25/2024).
16. Apple Support. Secure Enclave . Electronic resources . URL: https://support.apple.com/ru-ru/guide/security/sec59b0b31ff/web (accessed: 25.11.2024).
17. Hackaday. Apple's Secure Enclave Processor (SEP) Firmware Decrypted . Electronic resource. URL: https://hackaday.com/2017/08/18/apples-secure-enclave-processor-sep-firmware-decyrpted/ (date of access: 25.11.2024).
18. Apple Developer Documentation. Keychain services . Electronic resources. URL: https://developer.apple.com/documentation/security/keychain-services (accessed: 25.11.2024).
19. CSE. URL: http://www.cseweb.ucsd.edu/classes/fa10/cse120/lectures/CSE120-lecture.pdf
20. Apple developer's documentation. URL: https://developer.apple.com/library/ios/documentation/Miscellaneous/Conceptual/iphoneostechoverview/iOSTechOverview.pdf
21. Cambridge, development. URL: www.cl.cam.ac.uk/~acr31/p36/WP8%20Development%20Cambridge.pdf
22. ITU, Nokia and Symbian. URL: http://itu.dk/courses/ISOM/E2005/Nokia_and_Symbian_OS%5B1%5D.pdf
23. Gul M. A survey of anti-forensics techniques. 2017 International Artificial Intelligence and Data Processing Symposium (IDAP) / M. Gul, E. Kugu // Malatya. – 2017. – P. 1-6. Google Scholar.
24. Van Belle J.-P. A. Stander Anti-forensics: a practitioner perspective / J.-P. Van Belle, R. De Beer //Int. J. Cyber-Security. Digit. Forensics. – 2014. – № 4(2). – P. 391. Google Scholar.

25. San Bernardino Shooters Tried to Destroy Phones, Hard Drives, Sources Say, 2015. https://abcnews.go.com/US/san-bernardino-shooters-destroy-phones-hard-drives-sources/story?id=35570286. Accessed on: 12 September 2020. Google Scholar
26. Discarded laptop yields revelations on network behind Brussels, Paris attacks, 2017. https://edition.cnn.com/2017/01/24/europe/brussels-laptop-revelations/index.html . Accessed on: 12 September 2020.
27. Data Recovery E-Book V1.5 (Visit http://www.easeus.com for more information).
28. Peterson, Siberschaz, Galvin, “Secondary Storage Structure, Advanced Operating Systems”, 6th Edition.
29. Andrew S. Tanenbaum, «Modern Operating Systems» Prentice Hall, Dec. 2007.
30. Techniques in Computer Forensics: A Recovery Perspective / B.P. Battula et al // International Journal of Security (IJS). – Volume 3: Issue 2.

Acknowledgments

This study was carried out with the financial support of the Committee of Science of the Ministry of Science and Higher Education of the Republic of Kazakhstan under Contract №388/PTF-24-26 dated 01.10.2024 under the scientific project IRN BR24993232 «Development of innovative technologies for conducting digital forensic investigations using intelligent software-hardware complexes».

Л. Рзаева[1], Г. Абитова[*1], К. Ниязалиев[1], А. Байтулаков[1], В. Никулин[2]
[1]Astana IT University
010000, Қазақстан Республикасы, Астана қаласы, Мәңгілік Ел данғылы, 55/11
[2]State University of New York, Нью-Йорк, США
4400 Vestal Pkwy E, Binghamton, NY 13902, АҚШ
*e-mail: abitova.gul@gmail.com

ЦИФРЛЫҚ СОТ-САРАПТАМАЛЫҚ ЗЕРТТЕУЛЕРДІ ОҢТАЙЛАНДЫРУ ҮШІН БЕЙНЕТІРКЕГІШ ФАЙЛДЫҚ ЖҮЙЕЛЕРДІҢ ШИФРЛАУ ӘДІСТЕРІН ТАЛДАУ

Ұялы телефондар тек байланыс құралы ғана емес, маңызды ақпарат көздеріне де айналды. Шифрлау технологиялары мен әдістерінің дамуына қарамастан, арнайы әзірлеу және жаңа білім алу, мобильді құрылғыларды зерттеу әдістері мен тәсілдерін іздеу, әзірлеу және жетілдіру өзекті болып қала береді. Бұл зерттеу деректерді талдау мен өңдеу жүйесін оңтайландыру, сондай-ақ жасанды интеллект технологиялары мен машиналық оқыту алгоритмдері негізінде цифрлық сот сарапшылары жүргізген тергеулер кезінде тиімді шешім қабылдау мәселелерін қарастырады. Осыған байланысты, сондай-ақ жаңа әдістер мен алгоритмдерді құру үшін сандық криминалистикалық зерттеу әдістерін оңтайландыру үшін бейнетіркегіштердің файлдық жүйелерін шифрлау әдістеріне терең және жан-жақты талдау және шолу жүргізілді. Бейнетіркегіш файлдық жүйесін талдау және шолу шифрлауды анықтау және цифрлық бейнетіркегіштер мен мобильді құрылғылардан бейне мәліметтерді қалпына келтіру әдісін таңдауды және әзірлеуді негіздеуге мүмкіндік берді. Шолу файлдық жүйенің құрылымы мен жұмыс механизмін анықтайды. Ұсынылған талдау және шолу сот-медициналық және цифрлық сот сараптамасы сарапшылары үшін бейнебақылаумен байланысты сандық дәлелдемелерді талдауда пайдалы. Бұл талдау және шолу цифрлық сот сараптамасы бойынша ғылыми жобаны жүзеге асырудағы алғашқы қадам болып табылады.

Түйін сөздер: *ұялы телефондар, шифрлау технологиялары мен әдістері, деректерді талдауды оңтайландыру, сандық криминалистика, бейне жазба файлдық жүйелері, шифрлауды анықтау әдістемесі, жасанды интеллект технологиялары.*

Л. Рзаева[1], Г. Абитова[*1], К. Ниязалиев[1], А. Байтулаков[1], В. Никулин[2]
[1]Astana IT University
010000, Республика Казахстан, г. Астана, проспект Мангилик Ел, 55/11
[2]State University of New York, Нью-Йорк, США
4400 Vestal Pkwy E, Бингемтон, Нью-Йорк 13902, США
*e-mail: abitova.gul@gmail.com

АНАЛИЗ МЕТОДОВ ШИФРОВАНИЯ ФАЙЛОВЫХ СИСТЕМ ВИДЕОРЕГИСТРАТОРОВ ДЛЯ ОПТИМИЗАЦИИ МЕТОДОВ ЦИФРОВЫХ КРИМИНАЛИСТИЧЕСКИХ ИССЛЕДОВАНИЙ

Мобильные телефоны стали не только средством связи, но и значимыми источниками информации. Несмотря на развитие технологий и методов шифрования, остается актуальным развитие специальных и получение новых знаний, поиск, разработка и совершенствование методов и подходов к исследованию мобильных устройств. В данном исследовании решаются задачи, направленные на решение задач оптимизации системы анализа и обработки данных, а также эффективного принятия решений в ходе расследования цифровыми судебными экспертами на основе технологий искусственного интеллекта и алгоритмов машинного обучения. В связи с этим, а также для создания новых методов и алгоритмов был проведен глубокий и всесторонний анализ и обзор методов шифрования файловых систем видеорегистраторов для оптимизации методов исследования цифровой криминалистики. Анализ и обзор файловой системы видеорегистратора позволил обосновать выбор и разработку методики обнаружения шифрования и восстановления видеоданных с цифровых видеорегистраторов и мобильных устройств. В обзоре определены структура и механизм работы файловой системы. Проведенный анализ и обзор полезны судебным экспертам и экспертам цифровой криминалистики при анализе цифровых доказательств, связанных с видеонаблюдением. Данный анализ и обзор стали первым этапом в реализации научного проекта по цифровой криминалистике.

Ключевые слова: *мобильные телефоны, технологии и методы шифрования, оптимизация анализа данных, цифровая криминалистика, файловые системы видеорегистраторов, методология обнаружения шифрования, технологии искусственного интеллекта.*

Information about the authors

Leila Rzayeva – PhD, Associate Professor; Astana IT University; Republic of Kazakhstan, Astana; e-mail: leila2186@mail.ru. ORCID: https://orcid.org/0000-0002-3382-4685.

Gulnara Abitova* – PhD, Associate Professor; Astana IT University; Republic of Kazakhstan, Astana; e-mail: abitova.gul@gmail.com. ORCID: https://orcid.org/0000-0003-3830-6905.

Kyandyk Niazaliyev – MSc., Researcher; Astana IT University; Republic of Kazakhstan, Astana; e-mail: gulya.abitova@gmail.com.

Anuar Baitulakov – MSc., Senior-Lecturer; Astana IT University; Republic of Kazakhstan, Astana; e-mail: a.baitulakov@gmail.com. ORCID: https://orcid.org/0009-0003-1155-4914.

Vladimir Nikulin: PhD, Associate Professor, State University of New York, NY, USA; e-mail: v.nikulin@binghamton.edu. ORCID: https://orcid.org/0000-0003-4977-0332.

Авторлар туралы мәліметтер

Лейла Рзаева – PhD докторы, доцент; Астана IT университеті; Қазақстан Республикасы, Астана; e-mail: leila2186@mail.ru. ORCID: https://orcid.org/0000-0002-3382-4685.

Гүльнара Абитова* – PhD докторы, доцент; Астана IT университеті; Қазақстан Республикасы, Астана; *e-mail: abitova.gul@gmail.com. ORCID: https://orcid.org/0000-0003-3830-6905.

Қуандық Ниязалиев – магистр, ғылыми қызметкер; Астана IT университеті; Қазақстан Республикасы, Астана; e-mail: gulya.abitova@gmail.com.

Ануар Байтулақов – магистр, аға оқытушы; Астана IT университеті; Қазақстан Республикасы, Астана; e-mail: a.baitulakov@gmail.com. ORCID: https://orcid.org/0009-0003-1155-4914.

Владимир Никулин – PhD, Нью-Йорк мемлекеттік университетінің доценті, Нью-Йорк, АҚШ; e-mail: v.nikulin@binghamton.edu. ORCID: https://orcid.org/0000-0003-4977-0332.

Сведения об авторах

Лейла Рзаева – PhD, доцент; Astana IT University; Республика Казахстан, Астана; e-mail: leila2186@mail.ru. ORCID: https://orcid.org/0000-0002-3382-4685.

Гульнара Абитова* – PhD, доцент; Astana IT University; Республика Казахстан, Астана; e-mail: abitova.gul@gmail.com. ORCID: https://orcid.org/0000-0003-3830-6905.

Куандык Ниязалиев – магистр, научный сотрудник; Astana IT University; Республика Казахстан, Астана; e-mail: gulya.abitova@gmail.com.

Ануар Байтулаков – магистр, старший преподаватель; Astana IT University; Республика Казахстан, Астана; e-mail: a.baitulakov@gmail.com. ORCID: https://orcid.org/0009-0003-1155-4914.

Владимир Никулин – PhD, доцент, Государственный университет Нью-Йорка, штат Нью-Йорк, США; e-mail: v.nikulin@binghamton.edu. ORCID: https://orcid.org/0000-0003-4977-0332.

Received 11.05.2025
Revised 22.05.2025
Accepted 23.05.2025