International Journal of Scientific Research and Management (IJSRM)
||Volume||10||Issue||12||Pages||EC-2022-949-970||2022||
Website: www.ijsrm.in ISSN (e): 2321-3418
DOI: 10.18535/ijsrm/v10i12.ec01

Multimedia Data Secure Transmission: A Review

Hassan Elkamchouchi [1], Rosemarie Anton [1] and Yasmine Abouelseoud [2]

[1]Department of Electrical Engineering, Faculty of Engineering, Alexandria University, Alexandria, Egypt.
[2] Department of Engineering Mathematics, Faculty of Engineering, Alexandria University, Alexandria, Egypt.

Abstract

Encryption is a technique of encoding data so that they can only be recognized by authorized receivers. More interactive media information is communicated in the medical, business, and military fields because of the rapid advances in various multimedia transmission and networking technologies, which may containsensitive information that must be kept hidden from public users. Advanced encryption standards (AES) and data encryption standards (DES) are widely used encryption algorithms for text data. However, they are not appropriate for video data. To ensure that this information cannot be accessed by attackers, the demandfor efficient video-protection techniques has been raised. This article provides multimedia design requirements to maintain a secure multimedia system occupied with a threat model for detecting and rankingthe potential risks facing a multimedia system. the risks exposed to multimedia security and their impacts on users are typically described according to the textual description and also an overview of the current state-of-the-art video-encryption schemes are presented and their performance parameters have been examined. The relationship between encryption algorithms and compression techniques is also discussed and various multimedia applications have been presented in this paper; Additionally, as the synchronisationof real-time continuous streams is necessary for the interchange of these streams in multimedia conferencingservices, multiple synchronisation strategies have been given in this study along with video synchronisationchallenges.

Keywords: Cryptography, Multi-Media Encryption, Video Encryption Algorithms, Performance Parameters, Video Compression, Video Coding, Multimedia synchronization, video synchronization challenges, and Multimedia applications.

1. Introduction

Transmissions of multimedia files including images, sounds, and movies have proven to be more and more successful as a consequence of the quick development of network and multimedia technology. Our daily lives are increasingly dominated by digital video. Multimedia technologies such as computerized cameras and camcorders have consequently rapidly gained in popularity [1]. Encrypting the video data stream is a standard approach to satisfy the demand for protecting these multimedia files. The amount of data that can be securedis one of the key problems with securing video sharing because videos contain a large amount of data. Additionally, the video-encryption algorithm should be carefully selected based on the mobile device's capabilities, power requirements, and memory limitations. The encryption algorithm selected for real-time video transmission must take into account a variety of factors, including dependability and computing difficulties. For the goal of protecting video data, numerous encryption techniques have been developed [2].

Asymmetric-key cryptography and symmetric-key cryptography are the two main categories of cryptographic systems. The key property, which the encryption method utilizes to create the cipher text from the original data (plaintext) in order to make the data secure and only accessible to entities who have the corresponding key to retrieve it, is the cornerstone of cryptographic systems [2].

The message that needs to be encrypted and delivered is put through a number of steps, permutations, and/or substitutions in symmetric encryption. The message is encrypted during these procedures using the recipient's encryption key. The recipient decrypts the message using its own encryption key when it needs to be restored back to its original form. Therefore, the symmetric-key cryptographic algorithms use the same key for the encryption and decryption operations [3]. A schematic diagram is shown for symmetric-key cryptography in Figure 1. Many symmetric key encryption algorithms are available. The main algorithms for symmetric encryption are the following: advanced encryption standard (AES), DES, triple data encryption standard, and RC4 cipher. These algorithms are not suitable for direct use in video encryption; however, they can be employed by carefully adapting them.

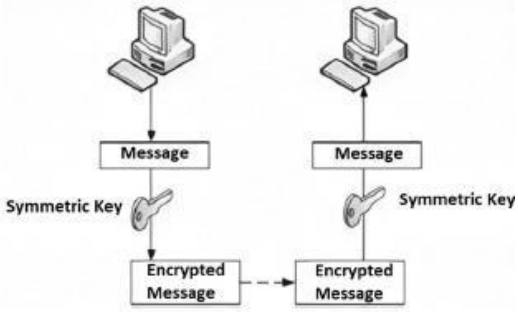

Figure 1. Symmetric key cryptography [3].

The key used in asymmetric-key cryptography is different for the sender and receiver. The key used for encrypting the message cannot be used while the message is being decrypted. For example, if the message is encrypted by the first person using key A (public key), the second person can only decrypt the encrypted message using a corresponding key B (private key), where A is generated from B by applying a one-way trapdoor function. In this type of encryption algorithm, the encryption and decryption keys are different [3]. Figure 2 shows how asymmetric key cryptography works. The most commonly used asymmetric key encryption algorithms are the following: Rivest–Shamir–Adleman (RSA), El Gamal, Diffie–Hellman, and their counterpart digital-signature algorithms. Asymmetric key encryption algorithms are slow and thus cannot be used for encrypting multimedia data, which are massive and the applications they are used in are real-timeand delay-sensitive. They can, however, be used for encrypting the keys used in symmetric key cryptography.

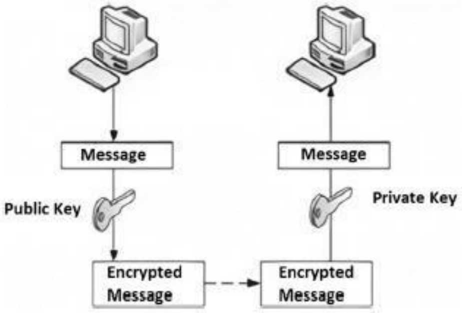

Figure 2. Asymmetric-key cryptography [3].

The work presented in this paper focuses on the issues exposed to multimedia security and the requirements needs to face it; also a methodical approach to identifying current threats to multimedia as there are several definitions of what makes a threat among disciplines. we will go over a few ways in which video data could be exposed to adversarial parties. In addition, a review of recent work on video encryption techniques is presented. Our review includes schemes suitable for videos in the form of moving pictures only.

The rest of this paper is organized as follows. In the next section, issues that face multimedia security in addition to the threat model demonstrating attacks against the illegal distribution of multimedia data transmission are discussed in section 2. Section 3 summarizes the metrics used for the evaluation of the performance of a video encryption algorithm. Different categories of video encryption algorithms aredescribed in Section 4. In Section 5 video coding schemes and video compression techniques are reviewed as they are essential tools for the efficient transmission of video streams. Moreover, they have a strong impact on the procedure used for encryption followed by video encryption techniques and examples of video encryption algorithms that belong to the various categories in Section 6. Multimedia synchronization algorithms and Video Synchronization Challenges are presented in section 7. Section 8 shows application areas for multimedia encryption. Finally, Section 9 concludes the paper.

2. Multimedia Security Issues

Multimedia security is often provided by a method or set of steps for preventing unauthorized access to multimedia content. These techniques, which focus entirely on cryptography, provide either communication security or anti-piracy security (Digital Rights Management and watermarking), or both. Digital images and text-based data can be transmitted securely using symmetric key cryptography. AES or DES can be used to encrypt all of the data in such medium, which can be represented as a binary sequence. The aforementioned methods often function effectively when the multimedia material is static (not real-time streaming) [4].

Choosing the appropriate level of security is more difficult than it seems. We must carefully balance the costof the multimedia data that needs to be secured against the cost of the actual protection in order to select the best security level. If the multimedia that has to be protected isn't all that important, to begin with, a simple level of encryption will do. On the other hand, the highest level of cryptographic security is necessary if the multimedia content is particularly valuable or involves a military or government secrets. For instance, the news might not be as helpful one hour later. In this case, it is sufficient to keep the data private for at least onehour despite minimizing the cost of supporting it privately for a long period of time [4].

Systems will be able to generate a continuous media stream due to the development of networked multimedia systems. Networked continuous media must be safeguarded from potential dangers like hackers and eavesdroppers, among others. Streaming has a plethora of applications. A whole linear programming package,a subscription service, or a pay-per-view service can all be offered via streaming (PPV). It can be used as partof an interactive website or as a standalone tool for video preview and film dailies. Internet broadcasting (corporate communications), education (lectures and remote learning), web-based channels (IP-TV, Internet radio), Video-on-demand (VOD), and content browsing on the Internet and intranet are just a few examples (asset management). These systems employ a variety of encryption techniques to improve the security of networked multimedia applications. When playing video streams via a network in real-time, the transmitted frames must have a short delay. Also, because video frames must be displayed at a specific rate, encrypted packets must be sent and received in a specified duration in order to take advantage of the allowable delay. Consider the following scenario: When using video-on-demand, the video stream must be played whenever the receiver requests it. As a result, the video stream has no buffering or playback notions (i.e. it runs in real-time). As a result, multimedia security faces numerous issues, including [5]:

- Even when using the best compression algorithms, the natural size of multimedia data after compression is frequently relatively enormous. The size of a two-hour MPEG-1 video is roughly 1 GB.
- Future multimedia apps will need to operate in real-time on processes like video on demand.
- Multimedia stream processing performance should be satisfactory (i.e. bounded by a certain value of delay).
- Encryption techniques should be efficient and have low overhead as compared to compression techniques.

2.1. Multimedia Design Requirements

For multimedia data security, multimedia content encryption is essential. Multimedia encryption requires both perceptual and cryptographic security. Cryptographic security refers to defense against cryptographic attacks, whereas perceptual security emphasizes that the encrypted multimedia content is incomprehensible to human perception. A secure multimedia system was one of the necessary conditions to maintain a user-friendly but secure end-user experience [4]:

A. It must be secure to use while yet being easy to install in order to draw additional content producers and providers.

B. The entire supply chain should be secured by end-to-end system security.
C. In order to draw in more applications and customers, it is vital to manage the current and new heterogeneous environment.
D. It should be scalable over a range of consumer devices, distributed caches, and storage systems.
E. Should be secure enough to transition from PCs to mobile devices, enabling new, more adaptable business models.
F. Regeneration should be easy.
G. Should not degrade the quality of streaming media playback, i.e., it should not affect the system's continuous playing, loss-resilient capability, or scalability in real-time streaming applications.
H. Users should be able to fast-forward or rewind content without affecting the viewing or playback experience.

2.2. Threat Model

A threat model is a systematic approach to detecting and ranking the potential risks and weaknesses of a system from the attacker's perspective. Using a threat model that has been examined by both security expertsand developers, work is always being done to increase system security. Threat modelling helps in assessing the seriousness of an assault and deciding whether to act immediately or safely ignore it. Five processes or components make up threat modelling, each of which is essential and functions in concert to provide a full security assessment of the system. The following are the components of threat modelling [6]:

- **Assets:** At all times, the attacker will make an effort to get access to some of the system's assets. Before creating a security solution, it is essential to identify the system's most valuable assets and how they may attract an attacker [6].
- **Entry points:** Entry points are vulnerable or suspicious points where attackers can access a system [6].
- **Attacker model:** The features of the attackers are described by the attacker model. It recognizes the attackers, their causes for attacking, and their ability in attacking [6].
- **Threats and vulnerabilities:** The most crucial part of threat modeling is determining the system's threats and vulnerabilities. Threats and vulnerabilities are categorized in many well-known threat modeling approaches to organizing the study [6]. The types of threats and their effects on users are typically described according to the textual description in their related studies by various security organizations and academia. In table 1, we list various multimedia threats that an attacker can use to obtain sensitive information from a user's multimedia data that they share on an SNS (social network security) [7].
- **Mitigation strategies:** In order to increase the security of a system, mitigation techniques are methods for preventing attacks and resolving vulnerabilities with well-known security solutions [6].

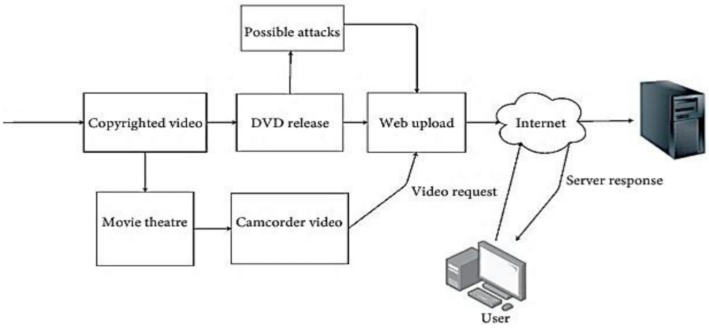

Figure 3. Illegal Distribution of Multi-Media data [8].

In Figure 3, illegal video distribution is shown, when videos are made available online after a DVD or movie is released. This problem resulted in significant losses for the film business. Real-time videos are gaining popularity on OTT (over-the-top) platforms like NETFLIX and AMAZON PRIME. As videos from these platforms are shared online, the concern over copyright protection reemerges, leading to a decrease in the number of individuals visiting these websites. It has been necessary to develop a safe method of identifying these unauthorised users and stopping illegal distribution. Video security is a crucialconcern for multimedia content makers who are experiencing new threats. Copyright protection is becoming more essential for multimedia content providers who are committed to producing creative mediaprograms. In order to help secure video data from unauthorised access during transmission, videoencryption techniques scramble the video stream [8].

3. Performance Parameters (evaluation Method) for Video Encryption

We need to develop a set of criteria that will allow us to compare and evaluate different video encryption techniques [9].

A. **Cryptographic security (CS):** Security is a fundamental requirement for multimedia data encryption and cryptographic security. This parameter determines whether the encryption algorithm is protected from attack by brute force and various plaintext/ciphertext attacks. For highly important multimedia use, the encryption algorithm must meet strict cryptographic requirements.
B. **Speed (S):** Security is the most important requirement for multimedia data encryption, and real-time encryption and decryption algorithms need to be quick enough to keep up with the demands of multimedia data applications.
C. **Visual degradation (VD):** This parameter restricts the algorithm's applicability to specific types of users. The perceptual distortion of multimedia data, such as video or pictures, is compared to plain data in this security standard. In addition, in some applications, it can be advantageous to make consumers pay to view content that isn't encrypted. For sensitive material, such as military photos, however, significant visual degradation may be necessary to totally hide the multimedia data.
D. **Compression Friendliness (CF):** In some applications, it is required that the size of encrypted data does not increase. As a result, an encryption system is known compression-friendly if it has little to no impact on the effectiveness of data compression.
E. **Format Compliance (FC):** In many applications, encryption algorithms should have a minimal overhead requirement. The encrypted bit stream should give compliance with the compressor with low overhead. A standard decoder should be designed to be able to decode the encrypted bit stream without decryption.
F. **Encryption Ratio (ER):** This parameter is used to determine the difference in data size between plaintext and cipher text. To put it another way, to reduce computational complexity, an algorithm's encryption ratio must be lowered. However, depending on the objective of their algorithms, authors of previously published video encryption algorithms take into account some of the aforementioned criteria. Table 2 summarizes the results of selected papers using the criteria outlined in the previous section.

4. Classification of Video Encryption Algorithms

They are classified into the following four categories:

A. **Fully layered encryption:** Layered Encryption is a compression method in which the entire video content is compressed first. After that, the data is encrypted with the AES or DES algorithms. This technique cannot be utilized for real-time encryption due to the higher temporal complexity. Furthermore, this condition could lead to a reduction in video quality [10].
B. **Permutation-based encryption:** A permutation-based video encryption technique encrypts video by permuting a specific component of the frame with another precise element. Video frame data contains a significant amount of pixel information. As a result, it's nearly impossible to recover all of the pixels' original places. A frame of 1920 x 1080 = 2, 073, 600 pixels in full high definition (FHD) resolution, which is commonly used for multimedia content, contains 1920 x 1080 = 2, 073, 600 pixels. Using a brute-force assault, a malicious attacker must rearrange and evaluate 2,073,600 frames to discover the originating frame. Without the permutation list, a brute-force attack on a permutated frame is essentially impossible. Video encryption based on permutations is substantially faster than video encryption based on block ciphers. On average, video data is significantly larger than text data. Block cipher-based

encryption has a high overhead since it repeats the same operation on the same portion of the data. As a result, using block cipher-based video encryption, real-time video streaming is almost impossible. The same component of the data is not permuted over and over again in permutation-based video encryption algorithms. As a result, it can encrypt video with less overhead than block cipher-based video encryption. The permutation-based video encryption methods are classified according to their encryption algorithm's position (see Figure 4). A video codec is used to compress the original video stream. The three forms of permutation-based video encryption algorithms are pre-compression, while-compression, and post-compression [11].

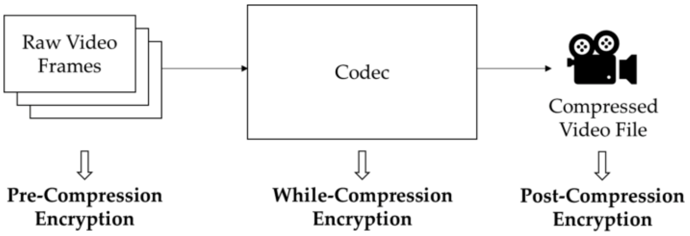
Raw Video Frames
Codec
Compressed Video File
Pre-Compression Encryption
While-Compression Encryption
Post-Compression Encryption

Figure 4. Permutation-based video encryption algorithms Classification [11].

C. **Selective (partial) encryption:** All methods that minimize computational complexity by encrypting a specific bitstream are included in this technique. Spatially selective encryption is a sort of selective encryption where the bits are chosen depending on spatial information [12]. fig 5 shows Video codec and security system process flow: codec-embedded SEAs.

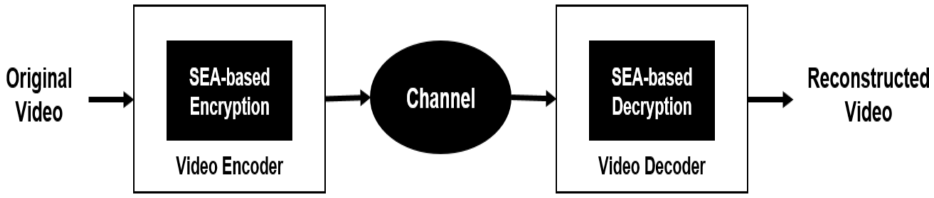

Figure 5. Process flow of video codec and security systems: codec-embedded SEAs [13].

D. **Perceptual encryption:** This technique keeps the low-quality perceptual information of the video content after encryption. This method can be used to change the audio/video quality. Perceptual encryption techniques have a low level of security in terms of content confidentiality. Although, they have a high level of security in terms of quality control reconstructions [14].

we present a comparison of these algorithms with respect to various parameters such as encryption technique, the security level provides for each technique, the computation needs, and finally the speed of each methodology as shown in Table 2.

Table 2. Classification of video encryption methodologies [15].

Full Encryption	**Selective encryption**	**Permutation based**	**Perceptual encryption**
Uses standard algorithms to encrypt every byte	Only selected bytes are encrypted	Permutation of DCT coefficients	The quality of the video is degraded
Highly secure	Moderately secure	Not secure	Not secure
Needs heavy computation	Needs less computation	Needs less computation	Computations can be controlled
Slow	fast	Very fast	Speed can be controlled

5. Video Coding and Video Compression

Compression is a core part of every video encoder and one of the most significant processes in multimedia applications. Any video that is available on the Internet or that is made on a phone can be encrypted. Therefore, knowing how to apply an effective video encryption technique sometimes needs understanding a certain video format (coding standard). As a result, we'll give a brief overview of video-coding standards in what follows [16].

H.260, H.261, MPEG-1, MPEG-2, MPEG-4, H.264, and H.265 are some of the video coding formats available. H.260 was the first video-coding standard, having been introduced in 1984. Because of its lack of performance, this coding system was not used in the application. Based on motion-compensated DCT compression, H.261 was the first workable video-coding standard established. MPEG-1, designed by MPEG(Moving Picture Experts Group) for Video Home System compression, was the next standard. When used athigh bit rates, it outperformed H.261 in terms of quality [16]. Half-pixel motion and bi-directional motion prediction were added to H.261 with this specification. MPEG-1 was eventually succeeded by MPEG- 2/H.262, which was designed for high-data-rate broadcast formats and was widely adopted as the DVD standard. MPEG-2 was able to support interlaced scan images with a wide range of bit rates [16]. The following standard was MPEG 4/H.263, often known as MPEG-4 Part 2, which was established in 1999 and made additional advances in video compression. This standard included segmented shape coding, variable block size, spatial predictive Intra coding, temporal and spatial scalability, and overlapping block-motion correction [16]. Among these standards, video encryption researchers have used MPEG-1, MPEG-2, and MPEG-4, while the most often used video-coding standard for video encryption is H.264/AVC [16].

The advanced video coding (H.264/AVC) standard and high-efficiency video coding (HEVC) are commonly applied as video encoding standards. Both H.264/AVC and HEVC use a block-based video-compression scheme with advanced technology. First, each block is predicted using images that have already been compressed. Second, an original block is subtracted from the predicted block. The residuals in this subtracted block are then translated into frequency coefficients in the third step. Fourth, the transformed coefficients are quantized to reduce their magnitude. Finally, the quantized coefficients are lossless compressed using entropy encoding. When a decoder receives the coefficients sent from an encoder, entropy decoding, inverse quantization, and inverse transform are performed in order. For the reconstruction, the residuals generated after the inverse transform are added into a block predicted from previously reconstructed images. During thevideo-compression process, quantization suffers from irreversible data loss. Thus, inverse quantization cannotrecover the original data. Figure 6 shows a general video-compression process [17].

Both H.264/AVC and HEVC encoders can in general decide compression types based on the target applications. Lossless compression is employed when the videos are used in error-sensitive applications like medical imaging. In most cases, lossy compression is used instead. There will be no quality loss during compression if the original video is lossless compressed. As a result, the original and reconstructed videos will have the same quality. However, if a lossy compression technique is used, the reconstructed video's quality degradation will depend on the quantization parameter (QP) value, which defines how much the original video is compressed. The value of QP can range from 0 to 51. QP should have low values if the desired bitrate is high. QP should have large values if the desired bitrate is low. In most cases, high QP valuescause huge distortions [17].

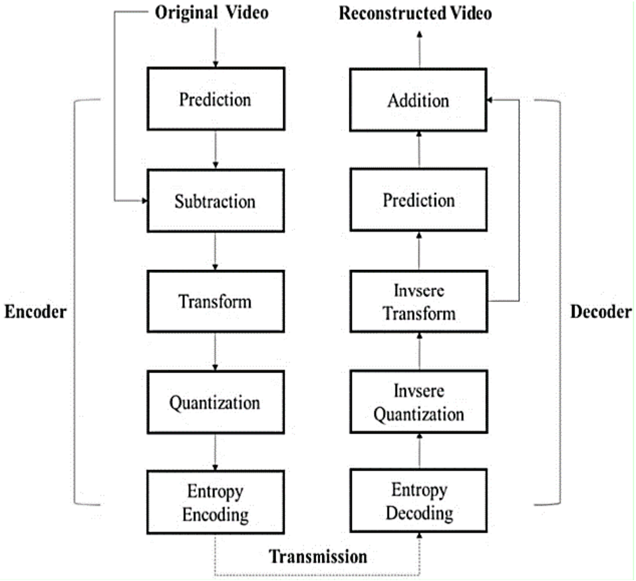

Figure 6. General video-compression process [17].

6. Video Encryption Techniques

The main drawback of the naive video encryption algorithm (NEA) is that it encrypts all data, therefore the computation time is proportional to the amount of data. A computationally complicated problem occurs when NEA is applied to a high-capacity video bit stream, such as a typical 2-hour movie that is stored and sent in gigabits after compression. As an alternative, selective encryption algorithms (SEAs) have been developed. A video bit stream is the smallest amount of data that allows the original video to be restored using the redundancy of the original video data. As a result, most (if not all) components of the video bit stream are interdependent, and a small proportion of the bit stream can be enough to destroy the entire bit stream, rendering the original video impossible to reconstruct. SEA solves this problem by encrypting only a portionof the video bit stream. When compared to NEA, this approach allows it to safeguard video data with far lessprocessing complexity [12].

Meyer and Gadegast [12, 18] proposed the Secure Moving Picture Experts Group approach for selective video encryption (SECMPEG). Maples and Spanos [12, 19] also proposed AEGIS, a selective video encryption approach. Both SECMPEG and AEGIS encrypt only the I-frame or keyframe data, which is required for a decoder to decode properly. Moreover, using a common encryption technique (DES) to encrypt only the I-frames is found to have an influence because it makes it impossible to fully reconstruct even the P- and B-frames that are reconstructed according to the I-frame. However, because the I-frames in the video bitstream typically account for 30 to 60% of the video size, the computational complexity is not much improved over that of NEA [12].

Tang [12, 20] proposed the zig-zag permutation algorithm, which reorganized the discrete cosine transform (DCT) coefficients in a zig-zag form during the video-compression step while generating an I-frame. The computational complexity of the zig-zag permutation algorithm is minimal because it rearranges data order in units of macroblocks that form I-frames. In zig-zag permutation algorithm applies the encryption function within the compression function; more specifically, after the quantization step. Experiments reveal that the zig-zag permutation method has just 1.56 percent of the computational cost of NEA. Despite its high encryption speed, the zig-zag permutation technique has a flaw in that it increases the size of the bitstream byabout 50% [12].

Shi and Bhargava [21] and Shi et al. [22] introduced SEAs such as VEA (Video Encryption Algorithm), improved VEA, and Real-time VEA (RVEA), that encrypt the sign bits of the DCT coefficient of the I-frame and the motion vector of the P- and B-frames [12]. Because the sign bits of the DCT coefficients and motion vectors occupied only a small portion of the entire bit stream, the computational complexity was evaluated to be only 10% compared with that of NEA. However, these methods cannot guarantee full security because useful video information can be recovered by simply changing all encrypted DC coefficients to 128 and all encrypted AC coefficients to positive numbers. The majority of SEAs are built

for the most recent video codec, HEVC, which encrypts the bit stream using syntax components. During the compression process, the syntax components selected for encryption are encrypted. As a result, video encryption based on syntactic elements is inextricably linked to video compression [12].
The I-frames, DCT coefficients of the I-frame, sign bits of the DCT coefficients of the I-frame, sign bits of the motion vectors of the P- and B-frames, and other elements make up the smallest amount of the overall bitstream encrypted by SEAs. When compared to NEA, this approach enhances encryption speed [12].

On the other hand, existing SEAs have significant weaknesses that must be repaired. When compared to NEA, the computational complexity is not much reduced. The encrypted part of the video bit stream expands its size. The level of protection becomes more fragile because the encrypted section is somewhat recoverable. Furthermore, most SEAs make it impossible to distinguish between encryption and compression techniques [12].
An example of selective video encryption is shown in Fig. 7. In this example, for a segment encoded using the H.264/AVC codec, the I-frames receive robust cryptography (e.g., uses more robust algorithms or larger keys), the P-frames are weakly encrypted (e.g., employs small keys), and the B-frames are not encrypted at all [23].

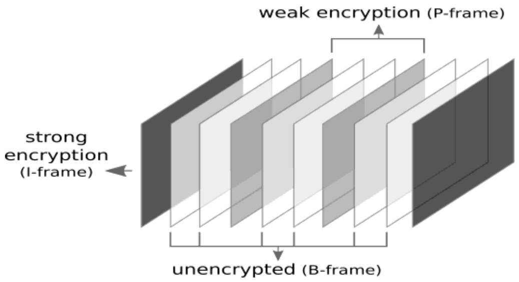

Figure 7. Example of selective encryption for predictive video coding [23].

6.1. Examples of recent video encryption algorithms

In what follows, examples of recent video encryption algorithms are described that belong to different categories [24].

A. Full Encryption Algorithms

1. In [25], the authors describe an implementation of completely encrypted video protection utilizing Elliptic Curve Cryptography (ECC) on a mobile device. This is accomplished using the Android platform. The results show that the two most crucial conditions for video mobile encryption are provided: a low computation time and a high level of confidentiality.
2. Design and implementation of a network video-encryption system based on STM32 are described in DVEMD [26]. The goal of this paper is to ensure the security of network video surveillance data and to prevent criminals from accessing and stealing valuable information.
3. Dual-layer video encryption using RSA is presented in DINVESS [27]. This paper provides a video encryption technique based on RSA and Pseudo Noise (PN) sequence, which is targeted at applications that need sensitive video data transmissions. The system is intended to operate with files encoded using the Audio-Video Interleaved (AVI) codec, but it may simply be modified to work with files encoded with the MPEG standard.
4. A modified AES-based algorithm for MPEG video encryption is developed in DLVRSA [28]. On MPEG video data, heavyweight encryption is applied in this paper.
5. Separable reversible data hiding and encryption for HEVC video are considered in PZCIVEA [29]. In this paper, a reversible data masking strategy including encryption for HEVC videos is suggested to secure videos while preserving their originality.
6. A video encryption technique using the AES algorithm is presented in SRDHEM [30]. Another approach for video encryption based on modified AES is presented in JLVEA [31]. This paper suggested a new

AES modified form that is better suited to encrypt digital video. The change focuses on replacing the slowest transformations in the original AES, which are mix columns transformations, with a new Henon map chaotic-based mask and one mix columns transformation.

7. Securing compressed video streams using the RC4 encryption scheme is examined in NAVEMAES [32]. The RC4 algorithm is a stream cipher that is faster than block ciphers. RC4 generates a pseudorandom keystream that is incomprehensible without the input key, making cryptanalytic attacks more difficult.

B. Permutation-Based Algorithms

1. Puzzle is an efficient, compression-independent video-encryption algorithm in MABAVE [33]. The puzzle was inspired by the children's game jigsaw puzzle. Puzzling and obscuring are two simple encryption processes with little computational complexity. In comparison to traditional encryption techniques like AES, the scheme significantly minimizes the encryption overhead, especially for high-resolution video.
2. JLVEA is a lightweight real-time video stream encryption algorithm for IoT VEAES [34]. It is a new permutation-based video encryption technique. It employs a crypto-protected pseudo-random number generator to refresh the permutation list for each frame without dramatically increasing memory use. As a result, the technique becomes resistant to known-plaintext attacks, which is a typical issue with current permutation-based video encryption schemes.
3. Video-encryption algorithm and key management using perfect shuffling are considered in SCVSRC4 [35]. The authors propose a computationally efficient and safe video encryption algorithm in the study, which makes encryption possible for real-time applications without a large computational cost and minimizes key management through the use of block shuffling.

C. Selective-encryption Algorithms

1. A fast, selective video encryption technique based on randomly selecting data for encryption using RC4 pseudo-random number generators is presented in VEAKMSH [36].
2. A selective-encryption scheme to protect H.264/AVC video in a multimedia network is developed in MPAMSVE [37], in which the scheme encrypts the sign of intra-macro block non-zero DCT coefficients, the sign of trailing ones (T1 s), the intra prediction modes (IPMs), and the sign of motion vector difference (MVD) to protect the texture and motion information of H.264/AVC after analyzing the impact of the quantization parameter (QP) on the encryption of the sign of T1 s and the impact of encrypting inter-macro block non-zero coefficient.
3. Selective encryption with multiple security levels for the H.264/AVC video coding standard is proposed in SESPHMSN [38]. The authors propose a novel technique with several levels of protection. In fact, five alternative cryptographic scenarios are proposed based on the nature of the encrypted coefficients. They maintain effective confidentiality while meeting the demands of real-time processing.
4. An efficient format-compliant video-encryption scheme for HEVC bitstreams is presented in SEMSLHCS [39]. The suggested approach first identifies the most important syntactic elements. Four syntactic elements are required to ensure that the suggested encryption method is highly efficient and that the encrypted bitstream is consistent with the HEVC standard.
5. In EFCVESH [40], an effective commutative encryption technique and a data concealing technique for HEVC videos are proposed. The commutative property enables ciphering a steganography video without interfering with the embedded signal, as well as performing steganography on an encrypted video while maintaining accurate decryption.
6. An encryption algorithm for H.264 video transmissions based on the blowfish algorithm is described in CEPHHVC [41], which addresses the issue of achieving a fair balance of security and encryption efficiency.
7. Encryption for high-efficiency video coding (HEVC) with video adaptation is discussed in EHEVCVA [42], where several encryption options for the HEVC standard are considered.
8. In RVTETBFA [43], four sketch attacks on H.264/AVC encrypted-compressed video are presented. The concept of a sketch attack is first discussed, and then traditional sketch attacks, which are meant for still images, are used to sketch the frames of H.264/AVC compressed video. They then suggest four sketch attacks that use partially decoded information from the H.264/AVC compressed video, such as residue

DC coefficients, residue AC coefficients, motion vectors, and macroblock bitstream size, to build an outline of the original frame.

9. The authors in ESEHIHEV [44] propose an enhanced selective encryption approach for H.264/advanced video coding (AVC) (CABAC) and HEVC streams, which addresses the fundamental security issue with SEAs.
10. A fast video encryption technique using the H.264 error-propagation property for mobile devices is developed in SKATVEM [45]. The paper suggests a specific encryption approach for H.264 that encrypts only the DC/ACs of I-macroblocks and the motion vectors of P-macroblocks and the motion vectors of P-macroblocks to assure format compliance and security.
11. A highly secure and fast video encryption with minimal overhead for H.264/AVC bitstreams is provided in FVEPSMD [46], which encrypts residues data and motion vectors after entropy encoding.
12. A joint selective encryption and data-embedding technique for HEVC videos are proposed in HSFVEMO [47]. The suggested technique is separable, with independent decryption and data extraction procedures and low parsing overhead.
13. A novel approach for real-time video encryption to secure multimedia transfer is suggested in SSEDETHV [48]. In this paper, they use pixel encryption. Both shuffling and modifying pixel values are used to perform pixel encryption.
14. In RVE/DSE [49], a selective video-encryption scheme based on coding characteristics is presented. By merging the video coding method with the encryption algorithm, the authors suggest an encryption technique that protects video information with higher security levels.
15. A format compliant, visual protection technique for HEVC videos based on selective encryption of CABAC bin strings is presented in VPHVSE [50].

D. Perceptual-Encryption Algorithm

1- A new perceptual evaluation methodology for selective HEVC video encryption is provided in FSVEA [51], where subjects' opinion is taken to assess the visual degradation of the encrypted video at different bit rates.

2- A new design of multiple transforms for perceptual video encryption is presented in SVESBCD [52], which incorporates random sign-flips into the DCT's implementation structure at later phases.

3- In NDMTPVE [53], Multiple Description Coding (MDC) is employed to offer a format-compliant perceptual encryption technique. The suggested method takes advantage of the MDC's scalable characteristic.

4- An improved perceptual video encryption technique based on the S-transform is developed in FCPVEMD [54]. The authors expanded their work by evaluating the rotation of blocks to be utilized to produce the S- transform.

5- In IPVEST [55], perceptual video encryption using SPIHT (Set Partitioning in Hierarchical Tree) transform is proposed. It gets good PSNR values and it is more trustworthy to transfer videos.

6- A perceptual video encryption technique that uses a rotation matrix and a unit anti-diagonal matrix to visually degrade video data with different perceptibility levels is presented in PVEUADM [11].

7- The study in PVEMSC [56] highlights the benefits of transparent encryption, which leaves the lower-quality base layer in the clear, in terms of reducing time and increasing distortion.
Table 3 summarizes the performance of selected papers according to the measures described in Section 4.

7. Multimedia synchronization algorithms

Synchronization of these streams is necessary for the sharing of real-time continuous streams in multimedia conferencing services. In multimedia conferencing systems, synchronization refers to the processes that control the timing of occurrences (such as the playback of an audio sample or the display of a visual frame). This section outlines several synchronization strategies and describes the synchronization forms in detail [57].

A. Intra-stream synchronization

Sequences of samples with a defined size are produced at regular intervals. It is necessary to maintain the timing relationship between the stream units in order to provide a high-quality display. The following elements, however, could have an impact on how time relates to stream units:

- Processing and network delay jitter (i.e., the variance in delay)
- Variations in the speeds of recording and playback.
- Unreliable transmission of stream data units.

The incoming stream data units can be delayed at the receiver to combat network jitter. Buffering and appropriate process scheduling can eliminate the discontinuity brought on by processing in the end nodes. If the same constant rate is guaranteed at the source and sink, jitter can be reduced. Small differences between the source and sink, however, could result in a buffer overflow [57].

Additionally, losing packets can cause the sink to run out of buffers. To detect late or lost packets, further techniques must be used. It is necessary to understand the maximum jitter induced by processing and transportation for intra-stream synchronisation. Buffering and scheduling can reduce jitter in the network and during processing. This necessitates that there be enough buffer area at the sink. There will always be a trade-off between buffer space and acceptable delay jitter, though, as more buffer space will be needed to reduce jitter. The timeliness of data, or the validity time of data, should be taken into consideration while determining the permissible delay jitter, especially with live media [57].

By ensuring that the source and sink have rate-synchronized clocks (or other devices), intra-stream synchronisation can be readily accomplished. A quick method to handle minor differences in source and sink rates is buffer monitoring. Although they will add communication overhead because of the corresponding feedback messages and the time reference, the feedback and global clock solutions can offer an alternative [57].

We can use the temporal link between Media Data Units (MDUs) in a video sequence as an illustration. In the event that the video was generated at a rate of 25 frames per second, the visualisation device must display each frame for 40 milliseconds. The synchronisation requirement for a video sequence featuring a jumping ball is displayed in Fig. 8. The Media Data Units MDUs will be kept in a reception buffer when they arrive atthe receiver in order to ensure intra-stream synchronisation. It will be necessary to ensure that MDUs are present in the buffer throughout the playing process (to prevent buffer underflow situations, which could result in starvation), as well as to ensure that the buffer is not already full when new MDU arrive (to prevent buffer overflow situations, which could result in flooding). Additionally, the playout procedure needs to be able to consume the MDUs at the same appropriate rate [58].

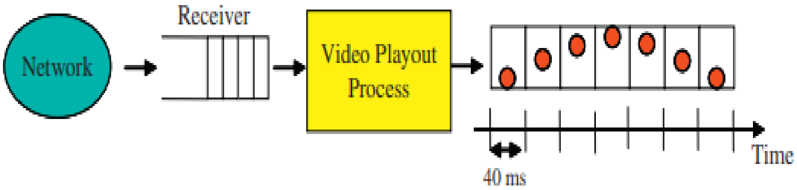

Figure 8. Intra-stream synchronization [58].

B. Inter-stream synchronization

There might be a temporal connection between different continuous streams at the source. This link must continue even after the streams are moved over conceivably different pathways. Applying an inter-stream synchronisation technique ensures this [57].

Getting inter-stream synchronisation is simple and accurate using multiplexing. Different QoS requirements for individual streams cannot be met since the continuous streams are sent as one stream (e.g., different jitterbounds for audio and video). In this situation, the synchronisation marker offers a good substitute that is simpleto use and has a comparable level of precision [57].

Fig. 9 (display-time bar char of a presentation's temporal schedule) provides an illustration of the temporal links between media streams in a multimedia application. A playout is displayed, starting with a video, then an audio sequence, many static graphics (slides), and finally, an animation with pertinent vocal commentary.Intra-stream synchronisation is also required in this situation, and after that, during playout, steps should be made to fix any potential differences between the playout processes to ensure inter-stream synchronisation. [58].

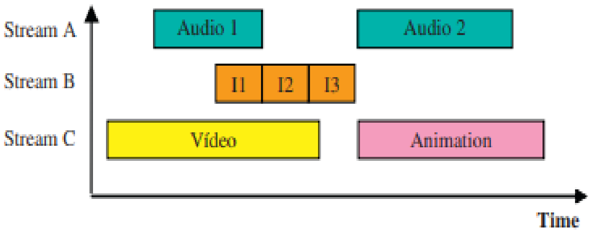

Figure 9. Inter-stream synchronization [58].

C. Spatial synchronization

A distributed conferencing service will have a number of recipients. To maintain a fair conference, it is crucial that all participants in it receive the audio and video data at the same time, especially when it comes to live audio-visual data. The employment of spatial synchronisation mechanisms based on global clocks, synchronisation channels, or feedback approaches as recommended for inter-stream synchronisation is done for this purpose. Mechanisms based on global clocks can achieve the most precise spatial synchronisation when these clocks are available [57].

D. Applied synchronization techniques

A buffer monitoring method is used for intra-stream synchronisation. Synchronization markers are used to establish precise inter-stream synchronisation. A centralised synchronisation manager, which is capable of using all three strategies for spatial synchronisation, will carry out spatial synchronisation. It would make sense to use the feedback technique to the three different types of synchronisation. On the Ethernet, however,that is used for our approach, the network delays are not precisely understood. This indicates that the feedback technique's accuracy is insufficient and that it is thus not employed. Similar justifications can be established for global clock-based techniques because nodes in a distributed computing environment do not always have synchronised clocks. Additional reasons for selecting the synchronisation methods employed are mentioned in the engineering specification [57].

7.1. Video Synchronization Challenges

A. Speed Control

The speed at which the video plays might not need to be controlled in some simple implementations. These simple implementations rely on the video source to continuously provide video frames. As soon as a video frame arrives, it is displayed. A conventional television broadcast is one illustration of this. Uncompressed video data is sent at a fixed bit rate during TV broadcasts. Although there is relatively little delay, this uses alot of bandwidth. For compressed video, the narrative is different. The playback side frequently needs speed control. One explanation for this is that MPEG-2, MPEG-4, and H.264 all have quite varied frame sizes.

I-frame, P-frame, and B-frame are the three frame types that are most frequently employed. They are typically mixed in with the video stream. The I-frames are the largest of the three. They take longer to send and use more bandwidth. The smallest frames are the B-frames. The video decoder will wait longer for I-frames duringconstant bit rate transmission than it will for P- or B-frames. As a result, each frame is received at a differenttime than when it should be displayed. Because of the non-deterministic nature of Ethernet and TCP/IP, transmission times are unpredictable even when using 100 Megabit Ethernet or higher capacity. Accordingly,the arrival time will vary greatly depending on the network situation. Further, if When a compression algorithm uses B-frames, the transmission frame order differs from the order in which the frames were taken.For instance, a B1 frame may be collected before a P2 frame yet sent after the P2 frame. Due to the reordering of the video data, speed management is necessary to display each image at the proper moment and for the appropriate amount of time. We will need time information in the video stream in order to execute speed control [59].

B. Jitter and Choppy Video

Jitter and choppy video are signs that a video picture is not being shown at the appropriate time and for the appropriate length of time. Additionally, it can mean that numerous frames are being lost and that the video's movement won't be fluid. These signs also point to a fluctuating frame rate. This issue could have a variety of causes. For instance, a system overload prevents the CPU or DSP from encoding or decoding a frame in time. This may result in a lag in transmission or a lag in the display of the images. Playback will be sluggish as a result. Later, the system will play quicker than usual as it tries to catch up. Internal buffer underrun or overflowis another potential source of this problem. Another factor is an insufficient bandwidth that exists temporarily.The delay between the video encoder and decoder never disappears. The latency is the result of adding the encoding, decoding, transmission, and propagation delays. The delays will cause the video to flicker and choppy. The time stamp information provided in the video stream allows us to statistically calculate deviationvalues, such as the average deviation and the maximum deviation. The presentation time of each frame can also be altered using the deviation value. Assuming there are no missing frames, this will lead to smoother video. Dropped frames cannot be recovered if choppiness is the result of them. A significant delay is typicallyundesirable for real-time video playing, even though postponing the video's presentation will also smooth it out. A longer delay must be balanced against less jitter and choppiness in the software. In any event, the divergence must be kept to a minimum during system installation. Otherwise, the divergence can be too great, which would make compensating for it impossible. Unavoidable jitter and choppy visual playback would result from this [59].

8. Multimedia Encryption Applications

Multimedia encryption is being more widely used in practically every area of daily life. Multimedia data (pictures, videos, audio, etc) is increasingly being used in applications like video-on-demand, video conferencing, and broadcasting. The following are a few of the most regularly utilized applications [60].

A. Secure Media Player

Encryption approaches for multimedia can be embedded into the media player. As a result, the player can decrypt as well as decompress and render media files. The decryption key and encrypted media data are included in the input data for this type of player. It allows service providers to create their own players with security features built-in. The secure player, as opposed to a regular player, has a decryption operation. The decryption operation should be carefully engineered to maintain the original performance, such as real-time playing [60].

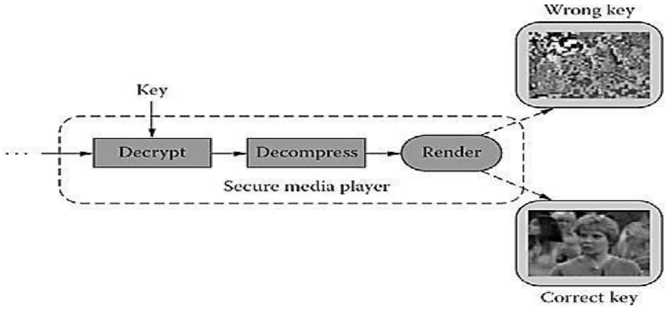

Figure 10. Secure media player [60].

Decryption and decompression operations can be ordered in a variety of modes, depending on the encryption algorithm used. The decryption operation should be fast enough to be used in real-time. Because media data is first decompressed and then rendered during playback, the added decryption operation may create delays in the decompression process. In general, the asymmetric cipher is more extensively employed, and the encryption and decryption operations are symmetric. As a result, for the safe player, a high-encryption-efficiency media encryption approach is selected.

B. Secure Media Streaming

Video-on-demand, IPTV, mobile TV, and other forms of real-time entertainment now rely heavily on media streaming. Streaming media offers obvious advantages over download-based methods. Before a media program can be played back using download-based approaches, it must be completely downloaded and saved on the user's device. In contrast, the media program can be played back while it is downloading in media streaming. Streaming media strives to deliver real-time media content transmission, according to this property. Secure media streaming is a service that ensures the security of media streaming. According to Figure 11, the sender encrypts the media content before streaming it, and only an authorized user can watch it. Two aspects are focused on secure media streaming: security and efficiency. The most basic requirement for secure media streaming is security. For secure content transmission, several solutions have been proposed. However, not every one of them is appropriate for multimedia data.

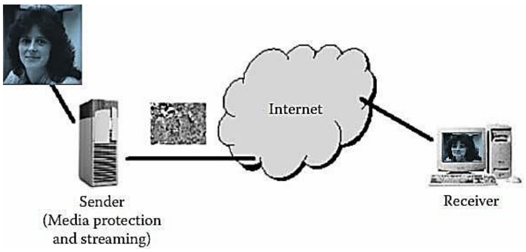

Figure 11. Secure media streaming [60].

Insecure media streaming; and encryption efficiency is a major concern. In general, a secure transmission mechanism that operates at a higher layer can achieve greater security than one that operates at a lower layer. ISMACryp, for example, can provide end-to-end security, but IPSec can only provide peer-to-peer security. SRTP and IPSec are both meant for broad data transfer, but ISMACryp is developed for multimedia transmission. Some partial encryption methods can be added to the ISMACryp architecture to selectively encrypt the parameters in the MPEG4 data stream. Additionally, better ciphers like VEA can be used to encrypt the data packets. As a result, the volume of encrypted data will be substantially reduced, and encryption efficiency will be greatly enhanced [59].

C. Secure Media Preview

In a Secure media preview, the sender uses perceptual encryption methods to decrease the quality of the original material before uploading the degraded content to the online portal. The degraded content is freely previewable by users. If a user is interested in the content, he will connect with the sender in some way, such as by paying for it and obtaining the key. He can retrieve the media content and view a high-quality copy after receiving the key. As demonstrated in Figure 12, this method can be applied to video-on-demand services. Naturally, depending on the transmission mode, such as streaming or downloading, the decryption procedure might take place online or offline. Because perceptual encryption algorithms employ partial encryption approaches, the decryption efficiency is frequently sufficient for real-time applications.

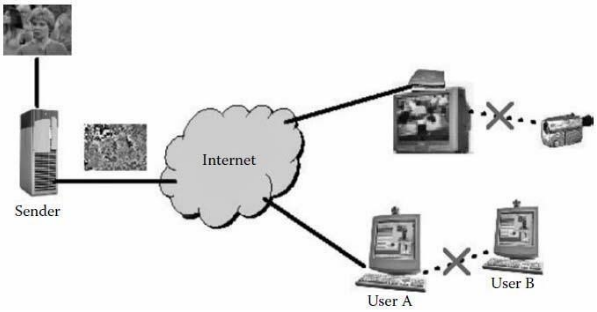

Figure 12. Secure media preview [60].

D. Secure Media Transcoding

Today, it appears that watching a TV show from any location is in high demand. One of the most pressing concerns is its safety. It mostly relies on two methods: multiple network convergence and scalable coding. Figure 13 depicts a combined example of the two methods. First, scalable coding is used to compress the TV show. The data stream is then compressed and sent over the Internet. The transcoder gets the data stream from the Internet, reduces the bit rate of the data stream by truncating specific bits, and then delivers it to mobile networks. Finally, the data stream is received by the mobile terminal, which decodes it and shows the TV program in low resolution. The bit rate of the original data stream is adjusted in this instance to accommodate the mobile channel's limited bandwidth. In most cases, the data stream does not need to be decompressed by the transcoder. The transcoder does not know the secret and can only truncate the encrypted data stream to achieve secure transcoding. As a result, a scalable encryption method that enables direct bit rate conversion should be utilized here. In addition, partial encryption mode can be used at the mobile terminal to save energy. Only the most important levels of the scalable data stream, for example, are encrypted, while other layers remain unencrypted. Of course, the security should be confirmed by adhering to secure partial encryption standards.

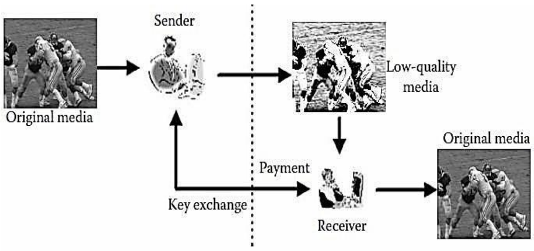

Figure 13. Secure media transcoding [60].

E. Secure Media Distribution

Secure media distribution safeguards the media content's different attributes, such as secrecy and copyright, by transmitting it from the sender to the user in a secure manner. An encryption mechanism will be employed to maintain confidentiality. Watermarking schemes such as the commutative watermarking (CWE) scheme or fingerprinting schemes such as the joint fingerprinting embedding and decryption system (JFD) scheme can be used to protect copyright. In the case depicted in Figure 14, the sender encrypts the TV program, which the authorized user can receive and decrypt. He is unable to redistribute the decrypted application to other unauthorized users, though. User A, for example, is unable to send his copy to User B. If this is the case, the distribution can be identified. In another case, the user cannot record the program with the capture and then send it out, for example, over the Internet. If so, the user can be traced [60]. For example, during decryption on the receiver side, the user's unique code, such as the set-top box ID or the user's registration code, is subtly placed into the TV show. As a result, the unique code can be found in the decrypted software. The unique code can be extracted and used to identify the unlawful user if the user transmits it out. The computational cost is not as critical if the decryption and watermarking operations are integrated into the set-top box. However, efficient algorithms are recommended if they are implemented on a PC or mobile terminal [60].

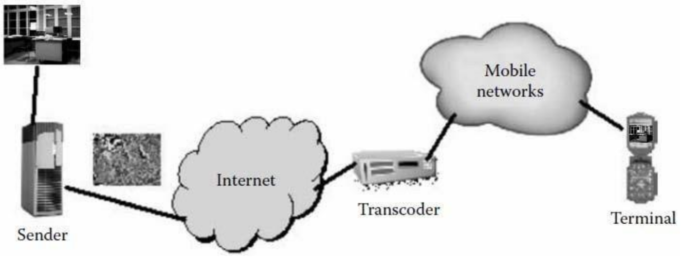

Figure 14. Secure media distribution [60].

9. Conclusion

A survey on video encryption has been presented in this paper, which describes various multimedia threats that can be used by an attacker to obtain a user’s sensitive information from the multimedia data that they share on network security; The type of threats and their impact on users are typically described according to the textual description in their related studies by various security organizations and academia; also multimediarequirements has been illustrated. Numerous video encryption algorithms are presented and examined their performance parameters. After the investigation made in this study, we found that the choice of the video encryption algorithm should be based on the application requirements. This is because different algorithms provide distinct tradeoffs among performance parameters. Finally, a classification of the main synchronizationtechniques for multimedia systems and the challenges facing video data synchronization has been presented.

Table 1. Multimedia content threats, description, impacts, and related studies.

Type of Multimedia threats	Description	Impacts	References
Multimedia content exposure	Shared multimedia data on social networking sites can immediately reveal a large quantity of sensitive information about users, such as their home address and recent activities.	Information leakage, reputational damage, location leakage, cyber harassment, profiling, and safety loss.	61,62
Shared ownership	Multimedia data published on social media sites can be linked to several people, but only one person can choose the multimedia data's desired privacy settings.	Loss of content ownership.	63
Manipulation of multimedia content	In SNSs, a malicious user can tamper the personal pictures of legitimate users to harm or ridicule them.	Reputation loss, Extortion/Blackmailing, Cyber harassment.	64,65
Steganography	A malicious user can share malicious information by concealing it within multimedia data such as a picture.	Reputation loss, Information disclosure, Safety loss.	66
Metadata	Because multimedia contents may expose other valuable data such as IDs and location, they operate as metadata.	Information disclosure, Location leakage, Reputation loss, Cyberstalking, Profiling, Safety loss.	67,68
Shared links to multimedia content	SNSs provide a function that allows users to share multimedia content in formats that aren't supported, such as GIFs, by posting a link to the content. A malicious person can make use of this functionality and alter the link's associated content with malicious external stuff.	Reputation loss, Information disclosure, Account loss.	69
Static links	The majority of SNS users share multimedia data via static links. A rogue user can simply copy and paste the static link to publish multimedia files outside of social networking sites.	Multimedia data disclosure, Data ownership loss.	70
Outsourcing and transparency of data centres	The SNSs do not encrypt the multimedia data they store. As a result, an unauthorized person can access the data without going through any authorization process. Small SNSs also use third-party storage, such as cloud-based data centres, to store their data. There could be a lot of privacy and security issues.	Multimedia data disclosure, Profiling, Data ownership loss.	71
Video conference	By exploiting the consequences of failing in the underlying communication architecture, an unauthorized user could intercept the broadcast video stream.	Reputation loss, Information disclosure, Blackmailing, Cyberbullying, and Cyberstalking.	72
Tagging	People who are not members of any SNSs and do not want to reveal any of their personal information may be linked with SNSs through tagging.	Multimedia data disclosure, Location leakage, Reputation loss, Cyberbullying, Cyberstalking.	73,74,75
Unauthorized data disclosure	A user on a social networking site (SNS) can share a photo with a specific group of people. However, any group member can retrieve the shared photo and re-upload it with his new privacy settings. As a result, a photograph may simply be displayed in public.	Reputation loss, Information disclosure, Location leakage, Content ownership loss, Identity theft, Extortion/Blackmailing, Cyber stalking, Profiling, Safety loss.	76

Table 3: Performance Metrics of Previously Proposed Video Encryption Algorithms.

Reference Article	**PS/VD** Perceptual Security/Visual Degradation	**ER/DR** Encryption /Decryption Ratio	**S** **Speed**	**MSE** Mean Square Error	**FC** Format Compliance	**CS** Cryptographic Security
PVEUADM [11]	✓	✓	✗	✗	✗	✓
EHEVCVA [41]	✗	✓	✓	✗	✗	✓
VPHVSE [49]	✓	✓	✓	✓	✓	✓
ESEHIHEV [43]	✓	✗	✓	✓	✗	✓
DVEMD [25]	✗	✗	✓	✗	✗	✓
DINVESS [26]	✗	✗	✓	✗	✗	✓
DLVRSA [27]	✓	✓	✓	✗	✗	✓
PZCIVEA [28]	✗	✓	✓	✗	✓	✓
SRDHEM [29]	✓	✓	✓	✓	✓	✓
JLVEA [30]	✓	✓	✓	✗	✗	✓
NAVEMAES [31]	✗	✓	✓	✗	✗	✓
MABAVE [32]	✗	✓	✓	✗	✗	✓
VEAES [33]	✗	✓	✓	✗	✗	✓
SCVSRC4 [34]	✗	✗	✗	✗	✓	✓
VEAKMSH [35]	✓	✓	✓	✓	✗	✓
MPAMSVE [36]	✓	✗	✓	✓	✗	✓
SESPHMSN [37]	✓	✓	✗	✓	✓	✓
SEMSLHCS [38]	✗	✓	✓	✓	✓	✓
EFCVESH [39]	✓	✓	✓	✓	✓	✓
CEPHHVC [40]	✓	✓	✗	✓	✓	✓
RVTETBFA [42]	✗	✓	✓	✗	✗	✓
SKATVEM [44]	✗	✗	✗	✗	✓	✓
FVEPSMD [45]	✓	✓	✓	✗	✓	✓
HSFVEMO [46]	✓	✗	✗	✓	✓	✓
SSEDETHV [47]	✓	✓	✗	✓	✗	✗
RVE/DSE [48]	✗	✓	✓	✗	✗	✓
FSVEA [50]	✗	✓	✓	✗	✓	✓
SVESBCD [51]	✗	✓	✓	✗	✗	✗
NDMTPVE [52]	✓	✓	✓	✓	✗	✗
FCPVEMD [53]	✗	✗	✗	✓	✗	✗
IPVEST [54]	✓	✓	✓	✓	✗	✗
PVEMSC [55]	✓	✗	✓	✓	✓	✓

Legend for Table 3: ☑Metrics Measured ☒ Metrics Not Measured

10. References

1. Anbu, T., M. Milton Joe, and G. Murugeswari. "A Comprehensive Survey of Detection of Tampered Video and Localization of Tampered Frame." Wireless Personal Communications (2021): 1-34.
2. M. A. Saleh, N. M. Tahir, E. Hisham, & H. Hashim., "An analysis and comparison for popular video encryption algorithms" IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE). IEEE; 2015.

3. S. Balli & M. Yilmaz, "Multi-criteria usability evaluation of symmetric data encryption algorithms in fuzzy environment". SN Appl Sci, vol. 2, pp. 1–12, 2020.
4. Saikia, M., S. J. Bora, and Md A. Hussain. "A review on applications of multimedia encryption." national conference on Network Security-issues, challenges and Techniques, 2012.
5. Elkilani, Wail S., and Hatem M. Abdul-Kader. "Performance of encryption techniques for real time video streaming." 2009 International Conference on Networking and Media Convergence. IEEE, 2009.
6. Hoque, Mohammad Aminul, and Ragib Hasan. "Towards a threat model for vehicular fogcomputing." 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), IEEE, 2019.
7. Rathore, Shailendra, et al. "Social network security: Issues, challenges, threats, and solutions." Information sciences 421 (2017): 43-69.
8. Sharma, Chirag, et al. "A secured frame selection based video watermarking technique to address quality loss of data: combining graph based transform, singular valued decomposition, and hyperchaotic encryption." Security and Communication Networks 2021 (2021).
9. Mekki, Neila, et al. "A real-time chaotic encryption for multimedia data and application to secure surveillance framework for IoT system." 2018 International Conference on Advanced Communication Technologies and Networking (CommNet). IEEE, 2018.
10. Khorsheed, Khattab O., Omar G. Abood, and Shawkat K. Guirguis. "Enhancing the performance of video encryption used for security and privacy protection in secure multimedia transfer." International Journal of Engineering & Technology, (2018).
11. Yun, Junhyeok, and Mihui Kim. "JLVEA: Lightweight Real-Time Video Stream Encryption Algorithm for Internet of Things." Sensors 20.13 (2020): 3627.
12. S. Alhassan, M. M. Iddrisu & M. I. Daabo, "Perceptual video encryption via unit anti-diagonal matrix". Appl Math Inf Sci, vol. 12, pp. 923–930, 2018.
13. M. K. Lee & E. S. Jang, "Start code-based encryption and decryption framework for HEVC". IEEE Access, vol. 8 202918, 2020.
14. Li, Shujun, et al. "On the design of perceptual MPEG-video encryption algorithms." IEEE Transactions on Circuits and Systems for Video Technology 17.2 (2007): 214-223.
15. A survey of Video Encryption Methodologies. https://1library.net/document/yro02evy-a-survey-of-video-encryption-methodologies.html
16. M. Dalal & M. Juneja, "A survey on information hiding using video steganography". Artif Intell Rev, pp. 1–65, 2021.
17. J. Y. Lee, "Impact of video compression and multimodal embedding on scene description". Electronics, vol. 8, p. 963, 2019.
18. J. Meyer & F. Gadegast, ``Security mechanisms for multimedia data with the example MPEG-1 video". Proj Descript SECMPEG Tech UnivBerl, Berlin, Germany, 1995.
19. G. A. Spanos & T. B. Maples, ``Performance study of a selective encryption scheme for the security of networked, real-time video". In: Proc. 4th International Conference Comput Commun Netw, Las Vegas, NV, USA; 1995:210.
20. L. Tang, ``Methods for encrypting and decrypting MPEG video data efficiently". In: Proc. 4th ACM Int. Conference Multimedia, Boston, MA, USA, 1996 229:219.
21. C. Shi & B. Bhargava, "An efficient MPEG video encryption algorithm," in Proc. 17th IEEE Symposium Reliable Distrib Syst, West Lafayette, IN, USA, vol. 386, p. 381, Oct. 1998.
22. C. Shi, S. Y. Wang & B. Bhargava, ``MPEG video encryption in realtime using secret key cryptography". In: Proc. International Conference Parallel Distributed process Techn Appl, Las Vegas, NV, USA 2828; 1999:2822.
23. D. G. Costa, S. Figuerêdo & G. Oliveira, "Cryptography in wireless multimedia sensor networks: A survey and research directions". Cryptography, vol. 1, p. 4, 2017.
A. A. Adenowo & L. F. Oderinu, "A comparative study of video cryptographic algorithms and the performance metrics used in the literature to measure the algorithms". P-ISSN 2006-1781 Afr J, comp. & ICT 13; 2020:43 – 61.
24. A. K. Naji & S. N. Alsaad, "Data (Video) Encryption in Mobile devices". Kurdistan J Appl Res, vol. 2, pp. 32–39, 2017.

25. X. Tian, C. Fan, J. Liu & Q. Ding, "Design and implementation of network video encryption system based on STM32 and AES algorithm". Smart Innov Syst Technol, vol. 82, pp. 51–58, 2018.
26. A. Chadha, S. Mallik, A. Chadha, R. Johar & M. M. Mani Roja, "Dual-layer video encryption using RSA algorithm". Int J Comput Appl, vol. 116, pp. 33–40, 2015.
27. P. Deshmukh & V. Kolhe, "Modified AES based algorithm for MPEG video encryption" International Conference on Information Communication and Embedded Systems, ICICES 2014 2015; 2014.
28. M. Long, F. Peng & H. y. Li, "Separable reversible data hiding and encryption for HEVC video". J Real-Time Image Process, vol. 14, pp. 171–182, 2018.
29. D. M. Dumbere & N. J. Janwe, "Video encryption using AES algorithm" 2nd International Conference on Current Trends in Engineering and Technology, ICCTET 2014; 2014:332–337.
30. S. Ali Abaas & A. K. Shibeeb, "A new approach for video encryption based on modified AES algorithm". IOSR JCE (IOSR-JCE), vol. 17, pp. 44–51, 2015.
31. S. S. Giradkar & A. Bhattacharya, "Securing compressed video streams using RC4 encryption scheme" Global Conference on Communication Technologies, GCCT 2015; 2015:640–644.
32. F. Liu & H. Koenig, "Puzzle - an efficient, compression independent video encryption algorithm". Multimedia Tool Appl, vol. 73, pp. 715–735, 8 10, 2014.
33. J. Yun & M. Kim, " JLVEA: lightweight real- time video stream encryption algorithm for Internet of things". Sensors (Basel), vol. 20, pp. 3627–3641, 2020.
34. S. F. Sultana & D. C. Shubhangi, "Video encryption algorithm and key management using perfect shuffle". Int J Eng Res Appl, vol. 07, pp. 1–5, 2017.
35. Z. Li, X. Wang & W. Yang, "A fast selective video encryption algorithm by selecting data randomly" Sixth International Conference on Electronics and Information Engineering 9794; 2015.
36. F. Peng, X. q. Gong, M. Long & X. m. Sun, "A selective encryption scheme for protecting H.264/AVC video in multimedia social network”. Multimedia Tool Appl, vol. 76, pp. 3235–3253, 2017.
37. F. Sbiaa, S. Kotel, M. Zeghid, R. Tourki, M. Machhout & A. Baganne, "A selective encryption scheme with multiple security levels for the H.264/AVC video coding standard”. Proceedings 16th IEEE International Conference on Computer and Information Technology, CIT 6th International Symposium on Cloud and Service Computing 2016; 2016, institute of electrical and electronics engineers, “SC2 2016 and” International Symposium on Security and Privacy in Social Network 2017; 2016:391–398.
38. M. Yang, L. Zhuo, J. Zhang & X. Li, "An efficient format compliant video encryption scheme for HEVC bitstream". Proc. 2015 IEEE.
39. International Conference on Progress in Informatics and Computing, PIC, vol. 2015, pp. 374–378, 2016.
40. D. Xu, "Commutative encryption and data hiding in HEVC video compression". IEEE Access, vol. 7, pp. 66028–66041, 2019.
41. R. Huang & C. Lu, "Research of H.264 video transmission encryption technology based on blowfish algorithm". Proc. 2015 4th International Conference on Computer Science and Network Technology, ICCSNT 2015; 2016:931–935.
42. G. VanWallendael, A. Boho, J. De Cock, A. Munteanu & R. Van De Walle, ``Encryption for high effciency video coding with video adaptation capabilities".IEEE Trans Consum Electron, vol. 59, p. 634–642_642, Aug. 2013.
43. K. Minemura & K. Wong, "Sketch attacks: A note on designing video encryption method in H.264/AVC”, vol. 2014, 2014 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2014.
44. B. Boyadjis, C. Bergeron, B. Pesquet-Popescu & F. Dufaux, ``Extended selective encryption of H.264/AVC (CABAC)- and HEVC-Encoded Video Streams” of H.264/AVC (CABAC)- and HEVC-encoded video streams. IEEE Trans Circuits Syst Video Technol, vol. 27, p. 892–906_906, Apr. 2017.
45. Y. Chung, S. Lee, T. Jeon & D. Park, "Fast video encryption using the H.264 error propagation property for smart mobile devices". Sensors (Basel), vol. 15, pp. 7953–7968, 2015.
46. J. M. Joshi & U. D. Dalal, "Highly secure and fast video encryption using minimum overhead in H.264/AVC Bitstream”. J Test Eval, vol. 44, pp. 140–332, 2016.
47. Y. Tew, K. Wong & R. C. Phan, "Joint selective encryption and data embedding technique in HEVC video", vol. 2017, 2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2016.

48. N. Dilkash, A. Gupta & A. Jain. “Real Time Video Encryption for Secure Multimedia Transfer: A Novel Approach. pdfs.semanticscholar.org 8; 2018:17077–17080.
49. S. Cheng, L. Wang, N. Ao & Q. Han, "A selective video encryption scheme based on coding characteristics". Symmetry, vol. 12, pp. 332–346, 2020.
50. Z. Shahid & W. Puech, ``Visual protection of HEVC video by selective encryption of CABAC binstrings". IEEE Trans Multimedia, vol. 16, p. 24–36_36, Jan. 2014.
51. N. Sidaty, W. Hamidouche & O. Deforges, "A new perceptual assessment methodology for selective HEVC video encryption," ICASSP, IEEE International Conference on Acoustics, Speech and Signal processing. Proceedings, pp. 1542–1546, 2017.
52. S. K. Au Yeung & B. Zeng, "A new design of multiple transforms for perceptual video encryption". Proc. - International Conference on Image Processing, ICIP; 2012:2637–2640.
53. N. Kamnoonwatana, "Format-compliant perceptual video encryption based on multiple description coding" 11th Malays International Conference on Communications, MICC 2013. IEEE; 2013:75–80.
54. A. Kirthanaa, N. Mathan & T. Vino, "Improved perceptual video encryption and decryption using S-transform" International Conference on Control Instrumentation Communication and Computational Technologies, ICCICCT 2015 2016; 2015:145–148.
55. T. Bernatin, S. Kuzhaloli, M. S. Premi & L. B. Queen, "Perceptual video encryption in multimedia secure communication". Proc. 2016 Online International Conference on Green Engineering and Technologies, IC-GET 2016; 2017.
56. M. N. Asghar, R. Kousar, H. Majid & M. Fleury, "Transparent encryption with scalable video communication: lower-latency, CABAC-based schemes". J Vis Commun Image Represent, vol. 45, pp. 122–136, 2017.
57. Jacobs, Martin, and Peter Leydekkers. "Specification of synchronization in multimedia conferencing services using the TINA lifecycle model." *Distributed Systems Engineering* 3.3 (1996): 185.
58. Boronat, Fernando, Jaime Lloret, and Miguel Garcia. "Multimedia group and inter-stream synchronization techniques: A comparative study." *Information Systems* 34.1 (2009): 108-131.
59. Yang, Hsueh-szu, and Benjamin Kupferschmidt. "Time Stamp Synchronization in Video Systems." International Foundation for Telemetering, 2010.
60. Saikia, M., S. J. Bora, and Md A. Hussain. "A review on applications of multimedia encryption." national conference on Network Security-issues, challenges and Techniques.
61. M. Kandias, L. Mitrou, V. Stavrou, D. Gritzalis, which side are you on? A new Panopticon vs. privacy, in: Proceedings of the International Conference on Security and Cryptography (SECRYPT), IEEE, 2013, pp. 1–13.
62. N.N.G. de Andrade, A. Martin, S. Monteleone, All the better to see you with, my dear: facial recognition and privacy in online social networks, IEEE Secur. Priv. 11 (3) (2013) 21–28.
63. L. González-Manzano, A.I. González-Tablas, J.M. de Fuentes, A. Ribagorda, cooped: co-owned personal data management, Comput. Secur. 47 (2014) 41–65.
64. A. Mendelson, does social media distort reality? (http://www.scoop.it/t/social-media-versus-reality). Online; accessed 04 April 2017.
65. Software informer, Image Distortion Tool Downloads, (http://image-distortion-tool.en.informer.com/). Online; accessed 03 April 2017.
66. A. Viejo, J. Castella-Roca, G. Rufián, Preserving the user’s privacy in social networking sites, in: Proceedings of the International Conference on Trust, Privacy and Security in Digital Business, Berlin Heidelberg, Springer, 2013, pp. 62–73.
67. J.C. Dressler, C. Bronk, D.S. Wallach, Exploiting military OpSec through open-source vulnerabilities, in: Proceedings of the Military Communications Conference, MILCOM, IEEE, 2015, pp. 450–458.
68. O. Van Laere, S. Schockaert, B. Dhoedt, Georeferencing Flickr resources based on textual meta-data, Inf. Sci. 238 (2013) 52–74.
69. S. Lee, J. Kim, Warningbird: a near real-time detection system for suspicious urls in twitter stream, IEEE Trans. Dependable Secure Comput. 10 (3) (2013) 183–195.
70. B. Sams, Facebook photo exploit allows you to view any albums of non-friends, (https://www.neowin.net/news/ facebook-photo-exploit-allows-you-to-view-any-albums-of-non-friends). Online; accessed 04 April 2017.

71. S. Singh, Y.S. Jeong, J.H. Park, A survey on cloud computing security: issues, threats, and solutions, J. Netw. Comput. Appl. 75 (2016) 200–222.
72. N. Ramzan, H. Park, E. Izquierdo, Video streaming over P2P networks: challenges and opportunities, Signal Process. Image Commun. 27 (5) (2012) 401–411.
73. A.C. Squicciarini, M. Shehab, J. Wede, Privacy policies for shared content in social network sites, VLDB J. 19 (6) (2010) 777–796.
74. F. Ahmed, M. Abulaish, A generic statistical approach for spam detection in Online Social Networks, Comput. Commun. 36 (10) (2013) 1120–1129.
75. L. González-Manzano, A.I. González-Tablas, J.M. de Fuentes, A. Ribagorda, Cooped: co-owned personal data management, Comput. Secur. 47 (2014) 41–65.
76. J.D. Lee, C.H. Sin, J.H. Park, PPS-RTBF: Privacy protection system for right to be forgotten, J. Converg. 5 (2014) 37–40. S. Rathore et al. / Information Sciences 421 (2017) 43–69 67.

HASSAN ELKAMCHOUCHI was Born on May 29, 1943, in Alexandria, Egypt. He graduated from the Faculty of Engineering, Alexandria University, First Honor Class, Communication Department, in June 1966. Included his certificates; B. Sc. That's right. Applied mathematics, University of London, England, Faculty of Science, August 1970, M. Sc. That's right. Communications, Sept. 1969, Engineering Faculty, University of Alexandria, and March, Ph.D. Communications Engineering. 1972, University of Alexandria, Faculty of Engineering. His scientific degrees were from September 1966 to August 1972, Professor of Electrical Engineering from September 1972 to March 1979, Professor of Assistance from April 1979 to April 1984, Professor of Communications and Wave Propagation from May 1984 to September 2003., and from May 2003 until now, Emeritus Professor of Communications and Wave Propagation. In more than 23 national and international conferences, he published numerous scientific papers in Antenna, Propagation, and data encryption

ROSEMARIE ANTON received a B.Sc. degree in communication and electronics engineering, and an M.Sc. degree in electrical engineering from the faculty of Engineering Alexandria University, Egypt in 2009 and 2015 respectively. She is currently a PH.D. student since 2016 in the Faculty of Engineering, at Alexandria University, Egypt also she was working as a teaching assistant in the faculty of engineering, at Pharos University in Alexandria from 2010 till September 2021. Her main research interests are in information security, engineering optimization, wireless communications, signal processing, and image processing.

YASMINE ABOUELSEOUD was Born in Alexandria, Egypt in the year 1978. She received a B.Sc. Degree in Computer Science, and M.Sc. And Ph.D. in Mathematical Engineering from Alexandria University, Egypt in 2001, 2005, and 2008 respectively. She is currently a Professor with the Department of Mathematics and Physics Engineering, Faculty of Engineering, Alexandria University, Egypt. Her primary research priorities are security of knowledge and optimization of technology. She has more than 40 publications in these fields. Her papers co-authored with her students won top papers / best paper awards at the 2012 International Conference on Industrial Engineering and Operations Management (IEOM), Istanbul, Turkey, 2012, IEOM 2015, Dubai, United Arab Emirates, and the 2012 San Francisco, USA, World Congress on Engineering and Computer Science.