Periodicals of Engineering and Natural Sciences ISSN 2303-4521
Vol. 8, No. 4, December 2020, pp.2405-2416

Securing software defined network transactions using visual cryptography in steganography

Al Baiati Ali Emad[1], Al Gburi Hussein Qahtan[2], Al Hamami Duaa Jaafar[3]
[1]Ministry of Higher Education, University of Technology, Computer Engineering Dep.
[2]Ministry of Education, Administrative Affairs, ID's Dep.
[3]Ministry of Education, Administrative Affairs, Biometrics Dep.

ABSTRACT

In the Software-Defined Network many transactions of data have been made, the need for securing this data very imperative. One of the most successful ways to secure data is by adding layers of both cryptography and steganography. In past years, many attempts to achieve this approach is done, but in this paper, a discussion made on how this approach works on Software-Defined Network, by first hiding secret messages inside the randomly generated image using LSB algorithm and then encrypting it using simple Visual Cryptography which based on creating two noisy images concluded from the original and only can be restored by stacking these images together. At the end of this research, results showed that two layers of security can be more secure than the one layer, this way the attacker can be obscured about the data hidden. The reason behind the implementation of this approach in SDN because of its importance since it's been used in many organizations and they need the data to be secured. Another problem that occurred in this system, is the latency to transfer the data from one node to another since this approach uses multimedia as a payload to transfer the data, a compression layer must be added, to decrease the amount of data sent through the network. After all this work, the data is now secured by applying two layers of security and faster to transfer hence this data is reduced by compression and simple visual cryptography is used and the BZIP2 compression technique to reduce processing of data.

Keywords: Steganography, Cryptography, Software-Defined Network, Visual Cryptography, BZIP2, LSB

Corresponding Author:

Al Baiati Ali Emad
Computer Engineering Department, University of Technology
Al-sinaa, Street, Baghdad, Iraq
Email: [1]120098@uotechnology.edu.iq

1. Introduction

Software-Defined Network-wide used in cloud computing and virtualization because of its combination of split control and data plane. As concluded, the key to SDN (Software-Defined Network) is the software of control of this system [1]. The software within control runs to control and analyze traffic thus is targeted by many third parties within the network. Thus, an enhancement on the security of SDN can be made by adding encryption and decryption software in all host parties [2]. Since the creation of SDN by the Clean State Project of Stanford University, the challenges for securing the data and network of SDN have appeared and many solutions have been proposed. One of these challenges is disguising the data for preventing any other parties within the network has been added on somehow [3].

In this paper, another layer of security, which is, have been used widely on some networks but not this type of network. The new layer is based on hiding secret messages inside the randomly generated image and encrypt this image using simple visual cryptography that uses two share images and stack them together for revealing the original image, which hides the original data [4].

This layer proved its functionality and security in many network types and scenarios and many other applications such as web-based applications. It provides two levels of security for information transmitting to

increase difficulties on the attacker to break the system security as well as it's easy to organize the data hence it's been encrypted in only two ways [5].
The phases of this system will be in three phases:

1- Generate a random image.
2- Hide data (text, or binary) in the generated image.
3- Encrypt the image using Visual Cryptography.
4- Compress data using BZIP2.

BZIP2 lossless compression technique, which combines Borrows-Wheeler transform with RLE and Huffman coding, is used to compress data before releasing it to the network [5].
These phases will be reversed in the receiver party.

2. Background

In 2015, Yogesh K. Meghrajani and Himanshu S. Mazumdar proposed a method to hide secret messages using both steganography and visual cryptography. They proposed that a random image should be generated and hide the text inside it by integrating a secret message within another apparently unobjectionable message. After hiding this message, a Visual Cryptography is implemented on the resulted image by splitting it into two shares. This system differs from another secure system because it has two levels of security in which complexity was achieved [4]. The previous journal was not the first attempt to combine visual cryptography with steganography, in 2012; Neha Chhabra proposed a similar approach to combine these two techniques together by also hiding secret information inside an image and split it into two shares for more complex security [6]. As for Visual Cryptograph, in 2014, used this approach to secure mailing services. As we know, the email messages are very popular and important uses in the global network. Ajish and Rajasree proposed that it is to apply this algorithm on the mail messages splitting it into two shares then encrypted using the Choas-Based image encryption algorithm using Wavelet transform [7]. Least Significant Bit steganography (LSB) is one of the most popular methods used for hiding messages inside images. In 2017, Nadeem, Vasim, and Hira explained this approach in detail and also put more improvement on this algorithm by using the modulus function which will break the data into two components. Each one is hidden cover using modulo function. And by using the "Peak Signal-to-noise ratio" (PSNR), the results showed that the new approach had improved the original [8]. This approach will be used to improve the technique in the proposed system.

3. Visual Cryptography

In 1994, Naor and Adi have developed the first attempt to split the image into a finite number of shares. The technique is based on encrypting any visual information and the decryption is dependent on humans, not a computer. The image will be break into n shares and stacked together to restore the original image. In this way, the second party needs to have all the shares in order to restore his information [9]. Figure 1 shows the idea of the process.

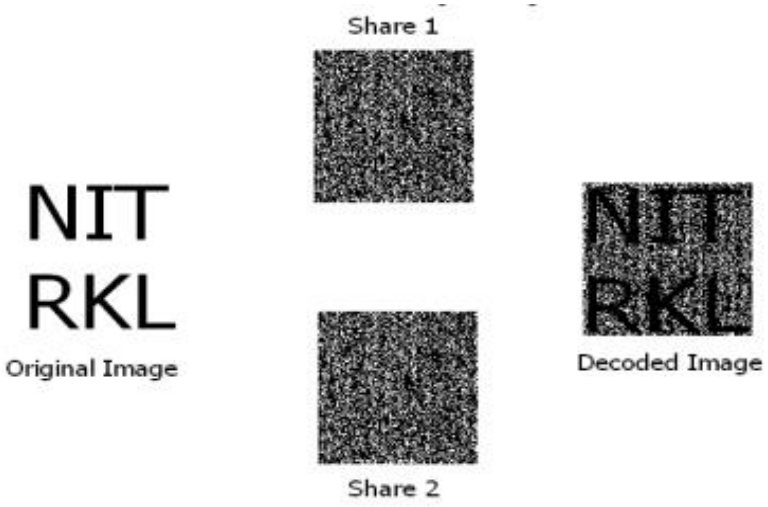

Figure 1. Visual cryptography technique

As we can see, Visual Cryptography is very useful especially when it comes to text images. In many applications, it is used for handwriting signature authentication. Figure1 shows that this image can reveal the secret information by stacking it using AND approach, while in our proposed work, an XOR applied to the approach. The reason behind this is that we need to restore the full original image not only the text. Since the information is visual text images, the generated shares are converted to binary, and by taking every pixel in the original, the pixel will be converted into a share block. The share block is randomly selected one of the possible blocks, except the share, 2 will be revers share 1, so when these shares are stacked by using XOR, the result will be cleared original image [10]. See Figure 2.

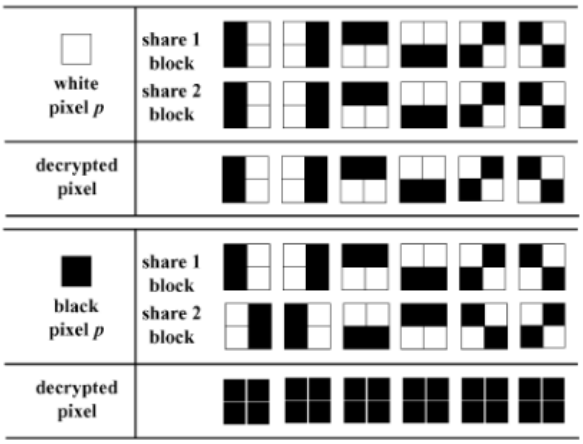

Figure 2. Blocks to be selected for each pixel in the original image

4. Proposed System

As been described at the beginning of this paper, all previous similar methods use steganography and visual cryptography with secret information. Our proposed system examines this combination of security for the transaction of data within SDN. The system will generate a random image, then it will use steganography to cover up the information interior of the generated picture, and finally, the resulted picture will be encrypted using visual cryptography for a transaction through the network. Data compression has been used as an additional feature for the fast transaction within the SDN network. See Figure 3.

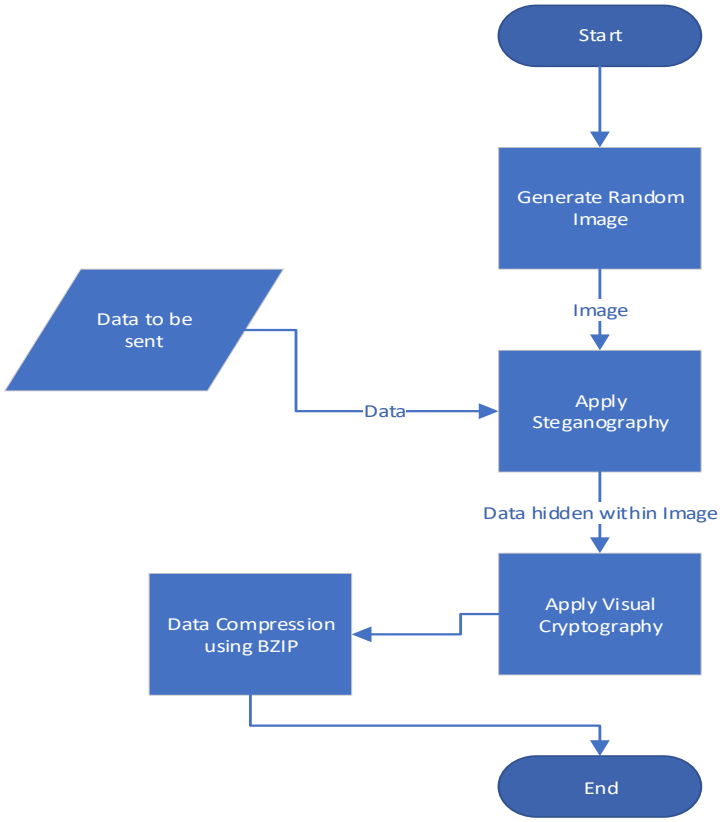

Figure 3. Workflow of the system

5. System Phases

The first step will randomly generate an image, which means the resulted image will be completely noisy and not clear. See Figure 4.

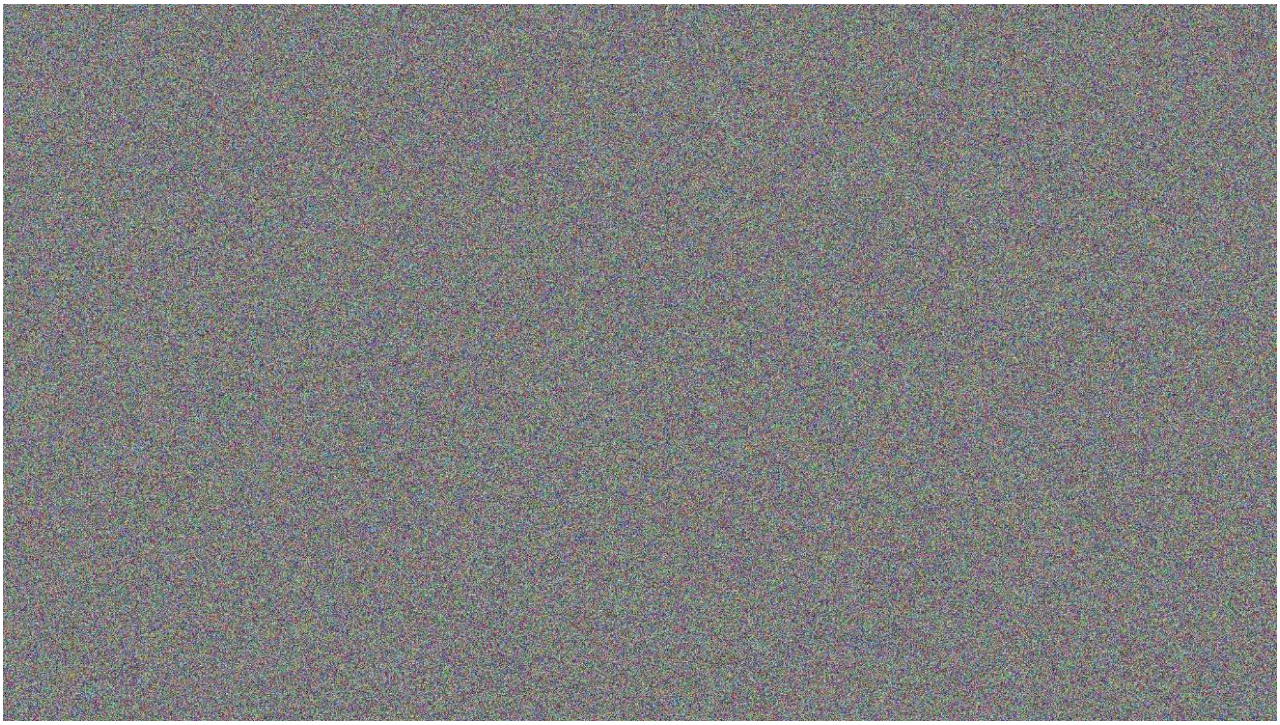

Figure 4. Random Image Generated

It's important to generate the image randomly than picking an image up from some gallery, it will add additional randomization and distract the attacker from the hidden data.

Next, the system will apply LSB (Least Significant Bit) to cover up the information interior of the generated picture from the previous picture, that's the main objective of Steganography. Images are best for data hiding. due to the huge quantity of excess space is made inside the storing of pictures. Steganography is composed of strategies for transporting mystery messages. Those mystery messages are transported via unknown cowl transporters in a manner that the very presence of the included messages is imperceptible. transporters envelop pictures; text; audio; video; or any other transportation or codes which can be represented digitally. The covered-up message can be ciphertext, plaintext, or anything else that will be appears as a bitstream [11].

LSB meaning that the 8th bit inside a picture is altered to a bit of the mystery message. one can be keeping three-bits in each pixel by when the utilize of a twenty-four-bits photo by utilizing altering one bit of all the red, blue, and inexperienced coloration elements, where they're each represented as "byte". An (800 × 600 pixels) can be keeping a total sum of (1,440,000) bits, which equivalent to (180,000 bytes) of inserted insights, for the occasion, a framework for three pixels of a twenty-four- bits picture can be as takes after:

(01010101 01011100 11011000)

(10110110 11111100 00110100)

(11011110 101100101 01101011)

When a binary representation (101101100) which is equivalent to the quantity (300) is inserted into the least full-size bits of this portion of the picture, the coming about the framework is as takes after:

(00101101 00011100 11011101)

(10100111 11000100 00001101)

(11010011 10101100 01100010)

Here the assortment 300 changed into inserted into the primary eight bytes of the framework; as it were, the five bits had to be changed in keeping with the inserted message. On rate, the most excellent half bits in a photo will ought to be altered to cover a mystery message utilizing the most cover size. Where, there are "256" conceivable densities of each number one color, altering the (LSB) of pixel comes about in little alters within the concentrated of the colors. Human being's eyes cannot see these alter - as a result, the message is efficaciously covered up. When choosing a good picture, one may indeed mask the message inside the (LSB) without taking note of the contrast. The algorithm below shows how the implementation of the (LSB) Steganography is:

1. Read the data needed to be covered up within the picture.
2. Data bits must be shifted to a cover-up within the cover picture via “X” bits.
3. A cover picture with (240) that is (11110000) so four MSb’s assign to 0. Four LSB are considered for this reason.
4. The shifted covered up the picture and the outcome of step three are bitored. This makes alters as it were within the “X” Lsb bits so that the picture is covered up within the original picture.

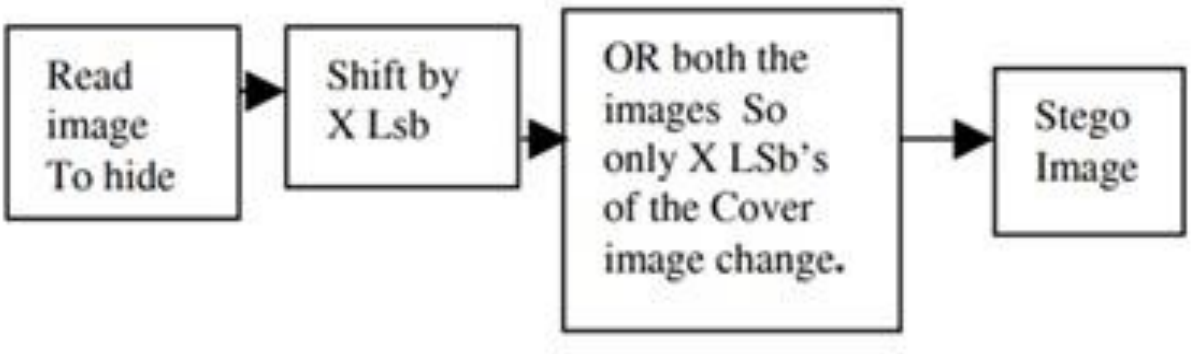

Figure 5. LSB Block Diagram

The next step is by applying visual cryptography to the resulted image. As we mentioned before, the system uses simple visual cryptography which will sperate the image into two shares and the recovered image only can be recovered by stacking up the two shares together by apply XOR operations on each block to regain the image. The final step, and in order to make our system faster transportation into the network, compression of the image bits is applied in a lossy manner to minimize the image size. The algorithm used to achieve this goal is BZIP2, which combines BWT (Burrows-Wheeler Transform) with run-length encoding, and Huffman encoding [12]. The main objective of compression is by reducing the frequent data inside the original data.

6. Experimental results

In this paper, an evaluation of each phase of the system starting from the generation of the image until the compression has been implemented. Also, in this section, the various scenarios of the SDN have been used for this system will be discussed. The system is implemented by Python as a programming language to implement the proposed system and evaluation of its metrics, and the simulation used is Mininet as an SDN network simulator that supports Python in its environment. Note that, Mininet can only be run and implement its scenarios in the Linux platform. The first phase after generation the image is to hide the data we need to be

secreted inside the resulted image from random generation. LSB is one of the best choices for this goal. It has the ability to hide data inside images without affecting the original image or change in its contents. An MSE (Mean Square Error) formula as a metric to calculate the changes created by using LSB. Objective excellent measurements or measurements of comparison are of extraordinary importance in the subject of photo processing. These measurements may be valuable for the evaluation and comparison of distinctive algorithms, outlined to unravel a specific issue. For illustration, one in all the attainable applications is the differentiate of different channels for picture commotion decrease. It is famous that classical extraordinary measurements, together with the peak sign to noise ratio (PSNR) or the mean rectangular error (MSE), not continuously match to visible observations. In this manner, various analysts are looking for seeking out new exceptional modern uncommon measurements, superior adjusted to human discernment. The current likeness measurements are all pixel-based and include in this manner no longer continuously first-class comes about. To manage with this downside, we prescribe a likeness degree based on a neighborhood, so that the pertinent frameworks of the photos are found exceptionally well. The new likeness degree is planned particularly for the utilize in picture handling [13]. The classic MSE is characterized as Let (X) and (Y) be the reference picture with culminating visual quality and misshaped picture with the same girth, and x(i,j) and y(i,j) represent to the pixel esteem at arranging (i,j) individually. The MSE between (X) and (Y) can be computed as takes after:

$$MSE(X,Y)=\frac{1}{M\times N}\sum_{i=1}^{M}\sum_{j=1}^{N}(x_{i,j}-y_{i,j})^2 \qquad (1)$$

Where (M) and (N) represent respectively the width and height of a picture [14].
The results showed that the MSE between the original picture and the new picture is 0, and the similarity between these two images is 100%. See Figure 6.

Figure 6. MSE and Similarity between Original Image (Left image) and Steganography image (Right image)

The resulted image of LSB is been processed further for more security by using Simple Visual Cryptography. The main idea of Visual Cryptography is to generate multiple shares from the original image, which will be stacked later in the decryption process on the other side of the network. The simple Visual Cryptography is the simplest implementation of this process; it splits the image into two shares by randomly picking up one of the 16 block orders (discussed before). See Figures 7 and 8.

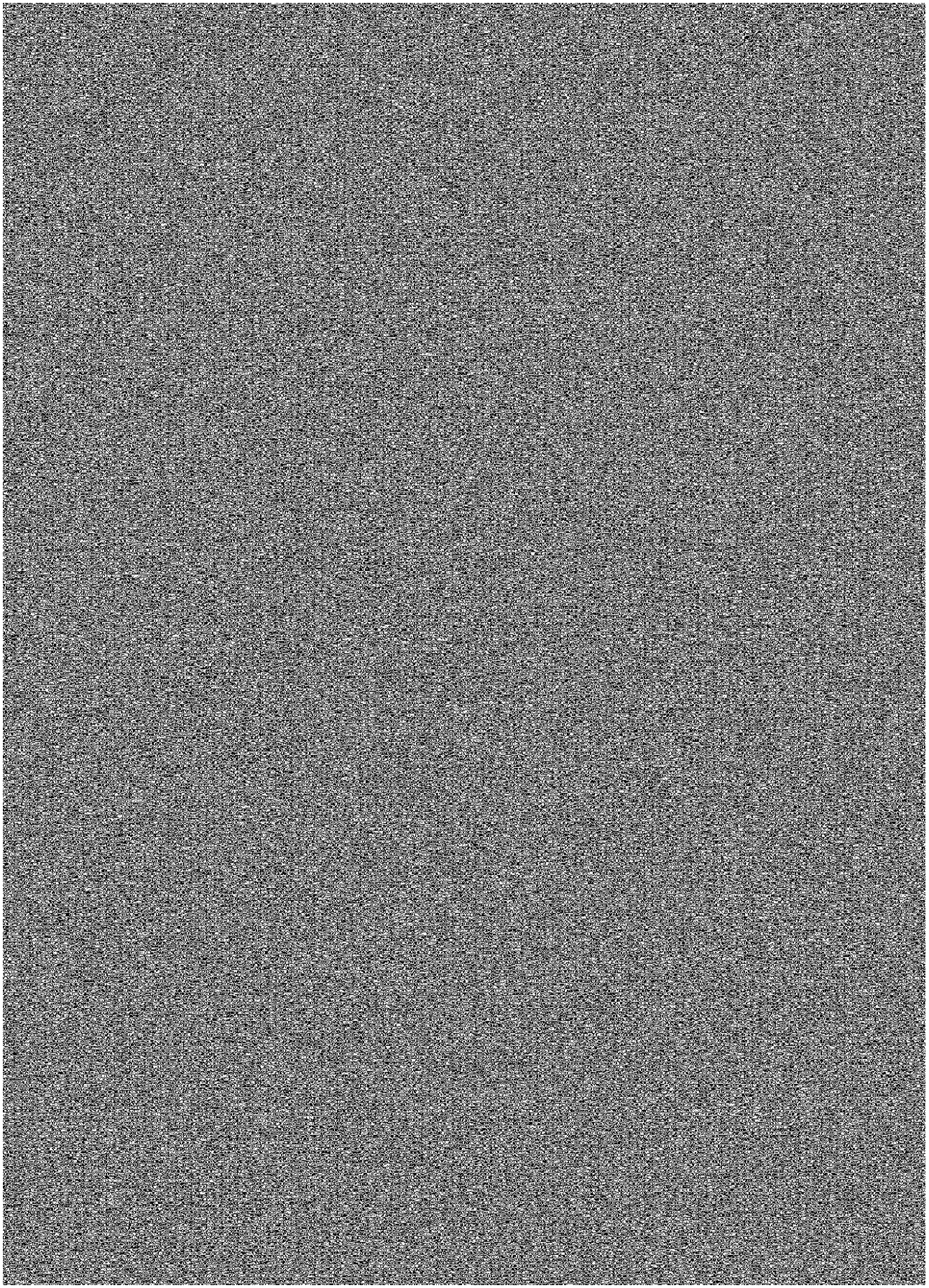

Figure 7. Share 1

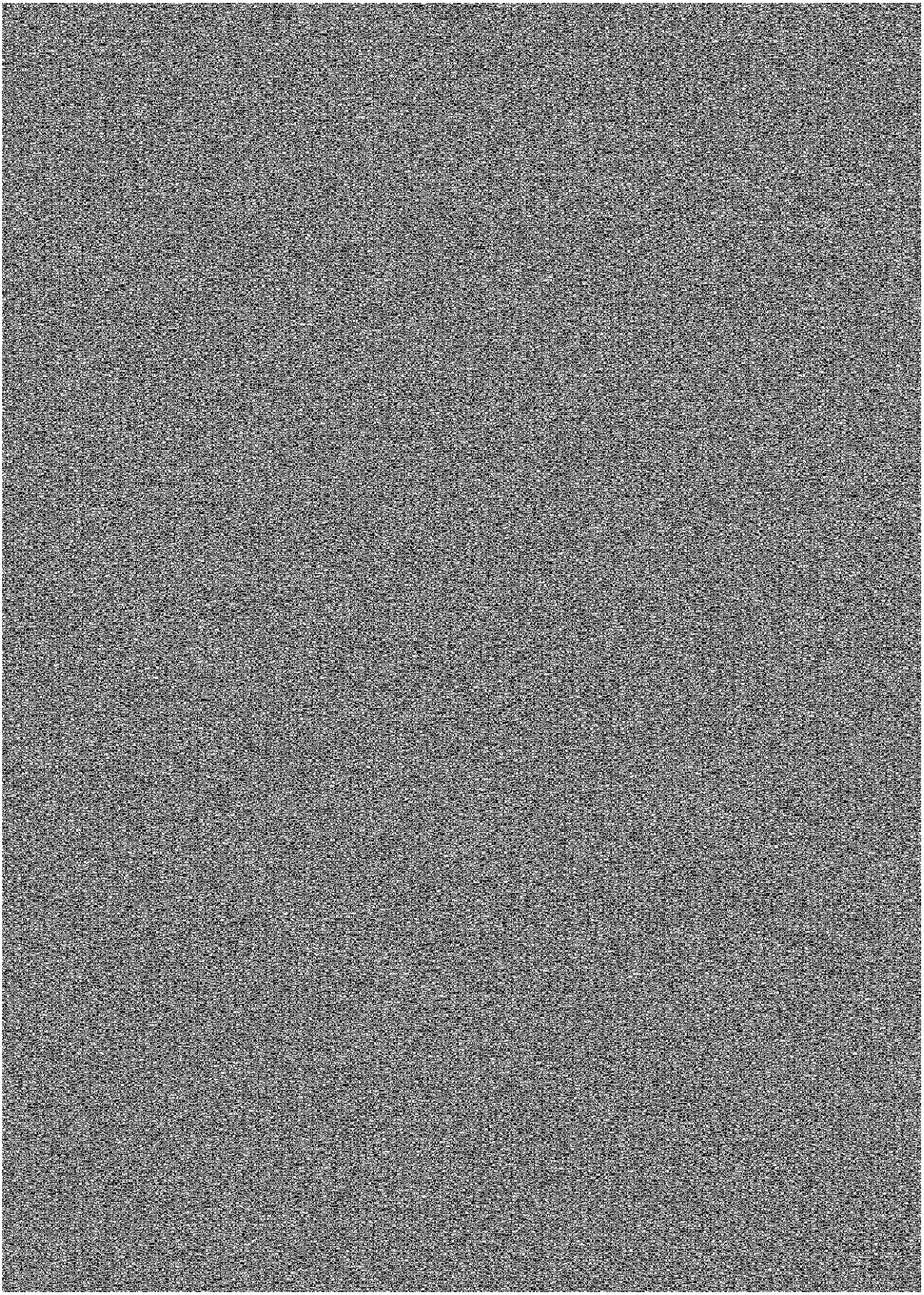

Figure 8. Share 2

As we can see, each share is larger than the original image; the reason behind this is that each pixel of the original image is replaced with 4 pixels block.
The both shares look similar by human's eyes, but in the matter of fact, these 2 shares are different in order to camouflage the attacker from detecting the hidden data.

7. Mininet

SDN have different simulation equipment like "Fs-SDN" [15], "NS-3" [16], and "EstiNet" [17], however, mini-net is the most broadly used net. "Emulator" and "simulator". That makes utilize of "Operating System-level" virtualization abilities to offer expedient simulation quickly, so it has qualities that give the meaning of flexibility. Which absolutely back up the "OpenFlow" protocol and deals with different type of (SDN) "open-sourced" projects and SDN equipment. These benefits have a guide to the usage of mini-net with the most studies of SDN. But, this net which used to simulate the real numerous scenarios and environment have a lack of version variety. Mini-net model "2.2.2" gives the best 4 models which can be (SDN switch, hosts, legacy router, and legacy switch) models. Best itemized parameters of loss probability, delay, and bandwidth may be provided by the link model. likewise, this model lacks to action analysis function and its own visitor's version creator. Instead, site visitors can be created with the aid of an external visitor creation tool that includes, "Iperf", "Ping"[18], and "D-ITG" [19]. Hence, it's distant, not adequate to utilize Mininet as a simulation apparatus in a test to affirm the stableness and unwavering quality of (SDN) and for tests in a genuine net environment. A few consider were carried out the utilization of Mini-net for the take a look at traffical networks [20-21]. Tests with Mininet have been completed for energetic rules control in combined naval force systems in [22] the utilize of "OpenDaylight" (ODL) controller, so also, (SDN) procedures have been given in energetic martial situations [23] wherein topology, assets and activity conditions alter. Mininet is used to design a 2d tree topology installed in a Ubuntu Linux based as an OS platform [24].
The scenarios used in this system is showed below:

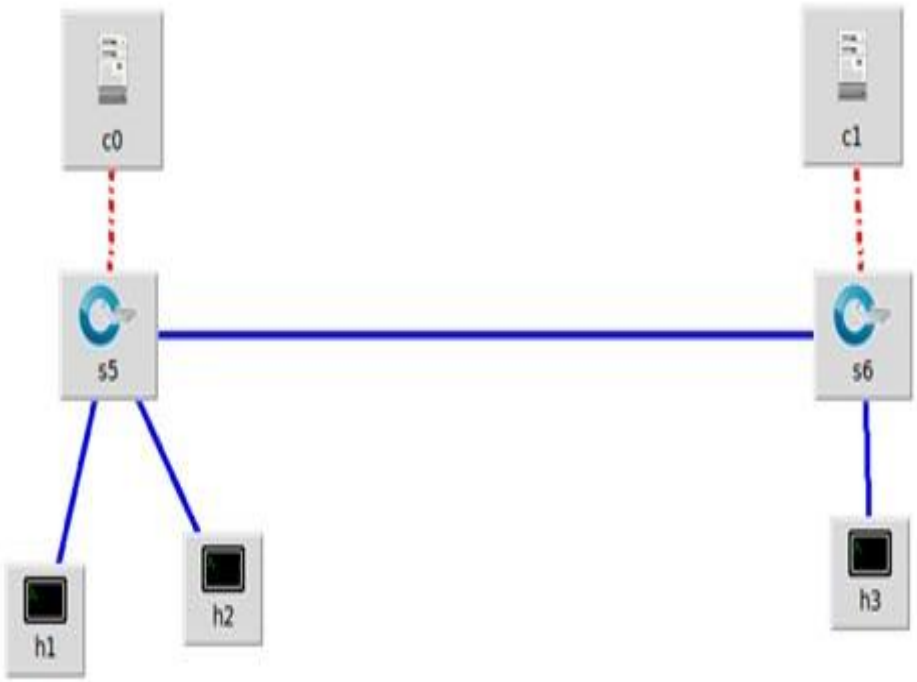

Figure 9. Scenario 1

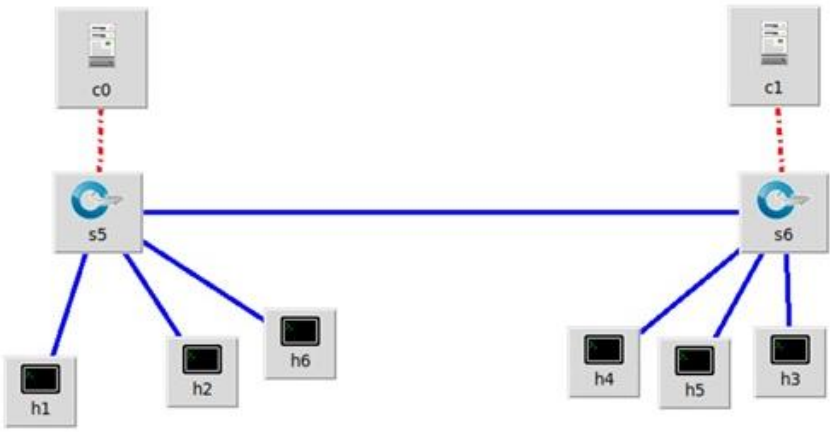

Figure 10. Scenario 2

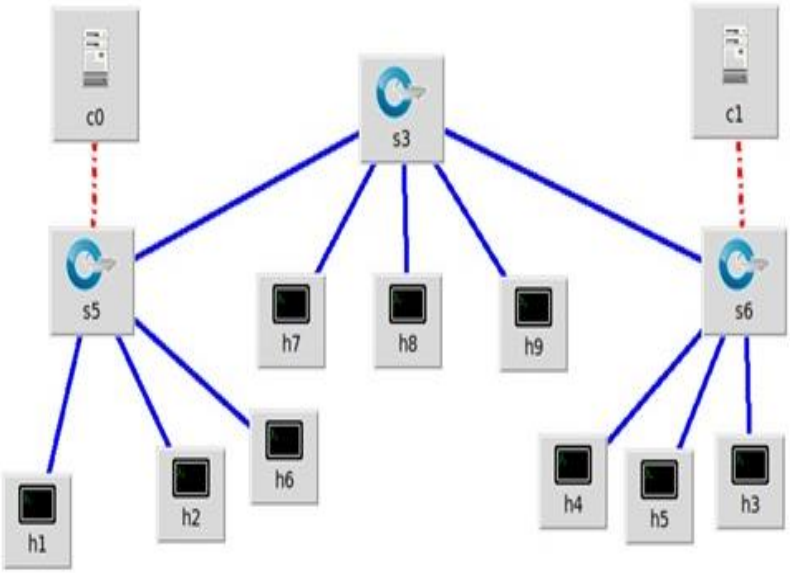

Figure 11. Scenario 3

In the above scenario, an OpenFlow protocol-based controller used which is more flexible [25]. The time performance of the transaction of the data is performed to evaluate how fast the system is. The following figure (Figure 12) shows the time of each scenario:

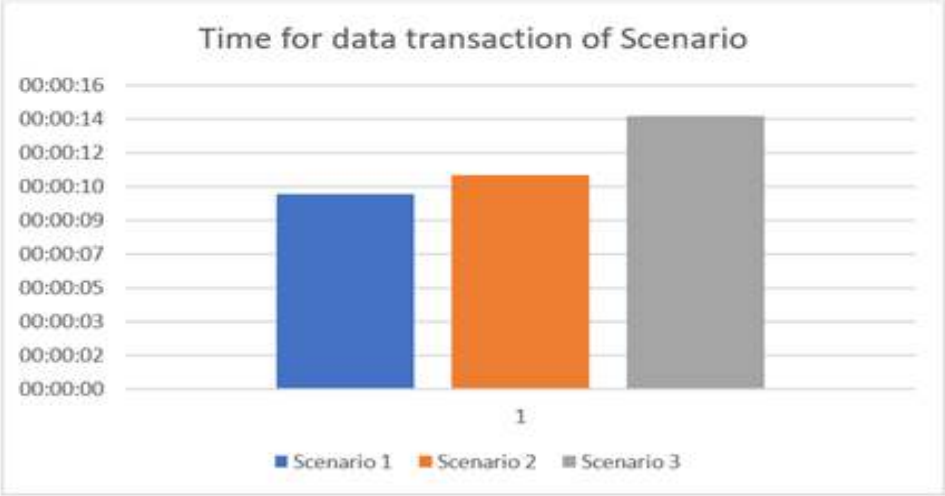

Figure 12. Time for data Transaction of Scenario

After applying data compression on the resulted shares, less time consuming is displayed in our evaluation:

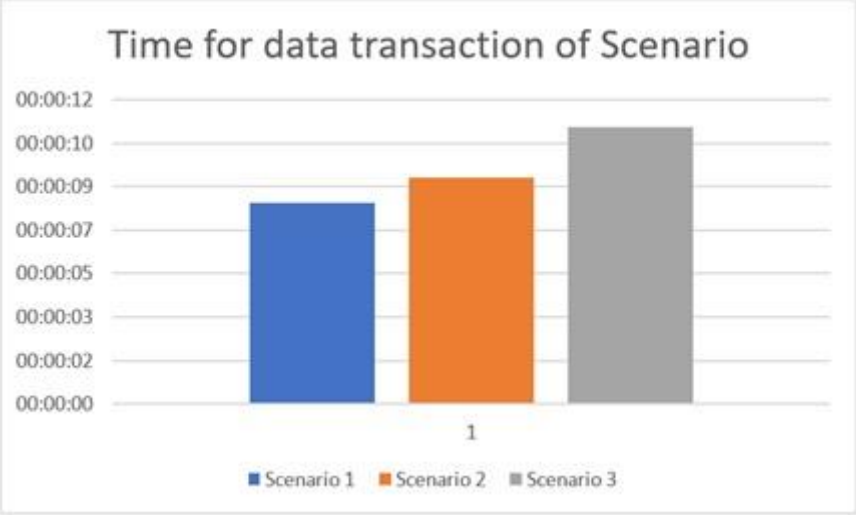

Figure 13. Time for Data Transaction of Scenario after Applying Data Compression

8. Conclusion

The multiple security layers show an excellent increase in SDN security. The random generation with LSB can disguise the attacker from concluding the message, and in addition, Simple Visual Cryptography will increase the difficulty of the attacker for cracking the messages.

References

[1] Scott-Hayward, Sandra, Gemma O'Callaghan, and Sakir Sezer. "SDN security: A survey." 2013 IEEE SDN For Future Networks and Services (SDN4FNS). IEEE, 2013.
[2] Aziz, Normaziah A., Teddy Mantoro, and M. Aiman Khairudin. "Software Defined Networking (SDN) and its Security Issues." 2018 International Conference on Computing, Engineering, and Design (ICCED). IEEE, 2018.
[3] Shi, Yue, Fangfang Dai, and Zhiguo Ye. "An enhanced security framework of software defined network based on attribute-based encryption." 2017 4th International Conference on Systems and Informatics (ICSAI). IEEE, 2017.
[4] Meghrajani, Yogesh K., and Himanshu S. Mazumdar. "Hiding secret message using visual cryptography in steganography." *2015 Annual IEEE India Conference (INDICON)*. IEEE, 2015.
[5] Tariq, Zaid Bin, Naveed Arshad, and Muhammad Nabeel. "Enhanced LZMA and BZIP2 for improved energy data compression." *2015 International Conference on Smart Cities and Green ICT Systems (SMARTGREENS)*. IEEE, 2015.
[6] Chhabra, Neha. "Visual cryptographic steganography in Images." International Journal of Computer Science and Network Security (IJCSNS) 12.4 (2012): 126.
[7] Ajish, S., and R. Rajasree. "Secure mail using visual cryptography (SMVC)." Fifth International Conference on Computing, Communications and Networking Technologies (ICCCNT). IEEE, 2014.
[8] Akhtar, Nadeem, Vasim Ahamad, and Hira Javed. "A compressed LSB steganography method." 2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT). IEEE, 2017.
[9] Jena, Debasish, and Sanjay Kumar Jena. "A novel visual cryptography scheme." *2009 International Conference on Advanced Computer Control*. IEEE, 2009.
[10] Sabitha, S. "User authentication using visual cryptography." *2015 International Conference on Control Communication & Computing India (ICCC)*. IEEE, 2015.
[11] Neeta, Deshpande, Kamalapur Snehal, and Daisy Jacobs. "Implementation of LSB steganography and its evaluation for various bits." 2006 1st International Conference on Digital Information Management. IEEE, 2006.

[12] Qiao, Weikang, et al. "An FPGA-Based BWT Accelerator for Bzip2 Data Compression." 2019 IEEE 27th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). IEEE, 2019.
[13] Van der Weken, Dietrich, Mike Nachtegael, and Etienne E. Kerre. "Image quality evaluation." 6th International Conference on Signal Processing, 2002. Vol. 1. IEEE, 2002.
[14] Cui, Ziguan, et al. "Simple and effective image quality assessment based on edge enhanced mean square error." 2014 Sixth International Conference on Wireless Communications and Signal Processing (WCSP). IEEE, 2014.
[15] M. Gupta, J. Sommers, P. Barford, "Fast Accurate Simulation for SDN Prototyping," 2nd ACM SIGCOMM workshop on Hot topics in software defined networking, pp. 31-36, 2013.
[16] T. Henderson, M. Lacage, G. Riley, M. Watrous, G. Carneiro, T. Pecorella, "Network Simulator 3," Available: https://www.nsnam.org.
[17] S. Wang, C. Chou, C. Yang, "EstiNet OpenFlow network simulator and emulator," IEEE Communications Magazine, vol. 51, no. 9, pp. 110-117, 2013.
[18] A. Botta, A. Dainotti, A. Pescapè, "A Tool for the Generation of Realistic Network Workload for Emerging Networking Scenarios," Computer Networks, Vol. 56, No. 15, pp. 3531-3547, 2012.
[19] E. Sorensen, "SDN used for policy enforcement in a federated military network," Master thesis, Institutt for Telematikk, 2014.
[20] H. F. Skappel, "Traffic Policing in Dynamic Military Networks Using Software Defined Networking," Master thesis, NTNU, 2016.
[21] D. Anderson, "An Investigation into the Use of Software Defined Networking Controllers in Aerial Networks," IEEE MILCOM 2017, 2017.
[22] P. Du, E. Pang, T. Braun, M. Gerla, C. Hoffmann, J. Kim, "Traffic Optimization in Software Defined Naval Network for Satellite Communications," IEEE MILCOM 2017, 2017.
[23] I. Elgendi, K. S. Munasinghe, B. Mcgrath, "A Heterogeneous Software Defined Networking Architecture for the Tactical Edge," in 2016 Military Communications and Information Systems Conference (MilCIS), pp. 1–7. 2016.
[24] Zaw, Hnin Thiri, and Aung Htein Maw. "Traffic management with elephant flow detection in software defined networks (SDN)." International Journal of Electrical & Computer Engineering (2088-8708) 9 (2019).
[25] Miano, Sebastiano, and Fulvio Risso. "Transforming a traditional home gateway into a hardware-accelerated SDN switch." International Journal of Electrical and Computer Engineering 10.3 (2020): 2668.