Improving the Security of a Cyber-Physical System using Cryptography, Steganography and Digital Signatures

Laura Vegh
Faculty of Automation and Computer Science
Technical University of Cluj-Napoca
Romania
Email: Laura.Vegh@aut.utcluj.ro

Liviu Miclea
Faculty of Automation and Computer Science
Technical University of Cluj-Napoca
Romania
Liviu.Miclea@aut.utcluj.ro

Abstract—**Over the past years technology has advanced more than ever we believed would be possible, being present now in almost every aspect of our lives. In this context, securing digital information has become of crucial importance. Cyber-physical systems are gaining more and more attention in research and industry and are being used in critical applications such as transportation, water supply systems, energy systems and so on. In the present paper we propose several solutions to improve the security of a cyber-physical systems for water supply and treatment. We focus on securing the data by using methods such as cryptography, steganography and digital signatures, each method having a specific role within the security architecture.**

Keywords-security; cyber-physical systems; cryptography; steganography; digital signature; hierarchical acces

I. INTRODUCTION

A. Overview

Protecting information has always been a subject of interest and nowadays, when we use technology in most of the areas of our lives, security is more important than ever. There are many methods to protect data and more are discovered almost every day. With systems more and more complex, we strive to find complex, yet simple security architectures that protect data from unwanted access, but at the same time do not overload the system with unnecessary operations. As a result, we can see in more and more cases combined security methods, such as encryption and steganography used together and so on.

Cyber-physical systems (CPS) are new and complex systems that offer a balance between physical and computational elements as they are projected as a network of interconnected devices with physical input and output [20]. In our paper we will propose a security architecture for a cyber-physical system modeled with multi-agent systems, using cryptography, steganography and digital signatures. Moreover, we will use hierarchical access to information, to increase security and ensure confidentiality. The proposed architecture can be adapted to various cyber-physical systems, it was designed to be flexible and not tailored for a single system. We tested the results on a water supply system, where we focused on processing and securing the output data of the system.

B. Security

In a world of ever-changing technologies, security is an area that will always be needed. Software and devices they all come and go, what seems like a great program today might become outdates tomorrow. Security however, will always be needed, no matter what type of system, or software we are taking into consideration. Like all technologies, this area should be in constant research, new algorithms, new architectures should always be ready.

Probably the best known security method is cryptography. Based on modifying the form of the data in such a way that it becomes unreadable, cryptography is divided in several types of algorithms, depending on how the data is encrypted, what type of keys are needed and so on. One such type is symmetric-key cryptography. In this type of algorithm only one key is used for both encryption and decryption, which makes it suitable for securing stored data, when no communication is involved. When the secret data is going to be exchanged between users, it is preferred to use asymmetric encryption. Also called public-key encryption, the algorithms in this category use what is called a public-key to encrypt data. The receiver will use a private key for decryption purposes. The most used public-key type of algorithms are Diffie-Hellman, ElGamal, RSA and various elliptic curves techniques. We will use public-key encryption to secure data in our system, more specifically a modified version of the ElGamal algorithm, which uses elliptic curves and a divided private key used for hierarchical access.

Another method to ensure the security of a system which has made a comeback in research over the past few years is steganography. Defined as the art of hiding a message, steganography does not change the form of the message like cryptography does, it simply hides it in digital media. There are several types of steganography according to the way in which the message is hidden and especially in the cover used. To name a few, there is image steganography, where image files

are used to hide data, audio or video steganography and even linguistic steganography. Regardless of the cover file used, steganography can increase the level of security of the system, either by hiding the data in its initial form, or hiding the encrypted data or even by hiding the keys.

Another form of ensuring security, that neither hides nor changes the form of the message, is the digital signature. They are used for authentication, integrity and non-repudiation purposes. Also, they can be used to control access to information. Furthermore, they will can be used as an addition to the security of a system that uses cryptography. Once encrypted, a message cannot be read without the proper key for decryption. Someone could however alter the form of a message even if they do not understand, thus compromising the data. By signing an encrypted message, we ensure that any modifications will be immediately visible, because modifying a signed message, alters the signature.

In our paper we will use cryptography, steganography and digital signatures to create a robust and new security architecture. Each method will be used at certain moments, or by certain users and sometimes they will be combined to ensure a better security level.

C. *Cyber-physical systems*

An integration of physical and computational processes, cyber-physical systems are becoming more and more popular. They are used in a wide area of applications in which security is a critical aspect. Research in the area is ongoing, new and improved methods to secure cyber-physical systems are always needed. To note that most of the times the security methods used depend on the area of application of the CPS. Due to their complexity and the variety of applications in which they are used, it is hard to find a solution that can fit all CPS. It is safer and more efficient to adapt the security architecture to the specific needs of each CPS.

There are several ways to model CPS. The one that we will be using is via multi-agent systems. Agents are autonomous components with decision making capabilities. One of the most important properties of multi-agent systems is that data is decentralized, meaning that no agent has all the information, each one has only the data it needs to complete its tasks. This is an important aspect when modelling complex systems such as CPS and more so when developing a security architecture. It is a property that aids limiting access to information and defining roles within a system.

D. *Water supply system*

The system used as a case study is the Cluj-Napoca city water supply system. This is a SCADA (supervisory control and data acquisition) system. Such systems contain applications and hardware that perform functions in order to provide various services such as energy, water, electricity and so on. A water supply system will control various processes such as water treatment, distributing water to consumers or treating used water.

Using CPS when working with such systems can aid with remote control and with monitoring the states and the operating conditions of the equipment.

All the processes within a water supply system are dynamic and they can be subjected to changes at various times. Such processes include water treatment, water chlorination or distribution. SCADA systems have as main functions monitoring and controlling, system surveillance and so on. They are complex systems and securing them is still a challenge. Due to the numerous components and processes most of the time the choice is to secure certain aspects of the system, certain processes. Our case study will be conducted on the module for data analysis regarding water flow and pressure. The findings are details and discussed in Section 4.

II. RELATED WORK

Security is an area in constant research. Cryptography has been used in information technology security for a rather long time. However, steganography, a method overlooked for a long period of time, has made a comeback in the attention of researchers. As such, the authors of paper [7] present a new LSB method for steganography which enhances security in the embedding and extraction phases. Paper [8] offers an approach on securing data through video steganography, while the authors of paper [11] focus on text steganography. In paper [9] steganography is used to secure biometrics, while in paper [12] we can see an extensive study on the usage of image steganography to secure communications.

The fact that steganography has returned in attention does not mean cryptography is used less. For example, in [19] we can see an address based cryptography scheme for mobile ad-hoc networks. The described approach is based on a combination of ad-hoc node address and public-key cryptography. In [16] the author proposes the usage a public-key infrastructure that supports both certificate-based and identity-based cryptography. Also in the area of identity-based cryptography is [3] where we can see a review of the main techniques and applications of this type of cryptography. Finally, in [14] the authors present the usage of hierarchy to secure data.

A relatively new approach, used also in the present paper is to combine steganography with cryptography. Literature provides us with a few interesting such works. For example [1] presents a secure electronic prescription system using steganography with encryption key implementation. Paper [5] presents the possibility to use the AES encryption standard, with linguistic steganography – more specifically the word shifting protocol method – to secure transmitted messages. Other papers studying and proving the benefits of combining cryptography with steganography are [4][15]. Here the authors use the two techniques to secure data features of communications.

In the area of CPS security, research is ongoing, there is no fixed proposed architecture. Most approaches are based on the area of application of the system, or even more on certain parts of it. Papers [20] and [21] presents a state of the art of the

challenges found in securing cyber-physical systems. Paper [6] focuses on risk management for power grid, while the authors of paper [10] focus on the security of cyber-physical energy systems.

In the area of access control, authentication and digital signatures, literature also provides us with some interesting work. Paper [22] describes the challenges of access control in cyber-physical systems. Also on the subject of access control we find [17]. In the area of digital signatures is [18] where the authors offer a new scheme for digital signatures for an application of document review in a hierarchical organization.

III. PROPOSED MODEL

The proposed security architecture is composed of three modules: one for cryptography, one for steganography and finally a module for digital signatures. Each module has its specific usage within the system, either used on its own, or combined with another module. We will discuss each module in the following paragraphs.

A. Cryptography module

We use cryptography in our systems for two purposes. The first is the classical use of encryption – to secure information by changing its form so that it becomes unreadable to any third party. We use a public-key type of algorithm, where the sender of a message is the one who performs encryption using the public key and the receiver performs decryption using the private key. What is different in our approach is that the private key is not unique for all users. We have what is called a divided private key, each user in the system will thus have its own private key. Each key will allow a user to decrypt only certain data. This will be the second use of cryptography in our system – that to restrict access to information for each user, thus creating a hierarchy in the system. For this we use an extension of the ElGamal algorithm called ElGamal with (k+1) degrees of access [13]. The algorithm uses the basic principles of the classic ElGamal algorithm and it requires using large numbers of up to 1024 bits, its security relying on the discrete logarithm problem.

Even though it is proven to be a strong secure algorithm, even for cyber-physical systems [23], the original ElGamal with (k+1) degrees of access algorithm can slow down the system in the case of multiple operations being performed at the same time due to the very large numbers used in calculations. Elliptic curve cryptography is said to eliminate these issues that arise in classical cryptography. We have thus modified the algorithm to use elliptic curves instead of large numbers, thus reducing computation times and increasing the level of security.

As stated in [13], the elliptic curve is defined as the set of all points $(x, y) \in (Z_p \times Z_p)$ satisfying the equation:

$$y^3 = x^3 + a \cdot x + b \bmod p \quad (1)$$

Where $a, b \in Z_p$ such that:

$$4a^3 + 27b^2 \neq 0 \bmod p \quad (2)$$

In terms of implementation, when working with elliptic curves, a challenge is represented by implementing the operations. One should pay special attention to the addition and multiplication of two points. These operations will be more or less the same with every elliptic curve algorithm, but they are essential because all the operations required for key generation, encryption and decryption respectively, will use them. Using Java as the programming language, we have implemented these operations in a separate class to allow to them access from all the other classes and to all users within the final system.

Looking back at our algorithm, once an elliptic curve is defined we need to choose a point belonging to the curve, Q, called the generator point. In classical elliptic curve cryptography, the private key is a random number, chosen by the user itself. In our case, we will calculate the private keys following the steps of the original ElGamal with (k+1) degrees of access. The main difference will be that we will not work with large numbers of 1024 bits, but with smaller numbers (the double type in Java), like the elliptic curve cryptography requires. The formulas to compute the private keys can be viewed in [13].

Probably the most important and challenging aspect of the present algorithm is key generation and distribution. We have to perform these operations in a way that is efficient but that keeps the keys a secret – if the keys would be found at the moment of distribution, the security of the system would be compromised. On the other hand, we need to pay special attention to the generation step as well, because we do not just compute keys, we decide the hierarchy of the system. This hierarchy can be seen as a tree structure in which the root is the user with the highest degree of access, being able to decrypt all and any messages, while the leaves of the system have the lowest degree of access. A unique aspect in our system is the leaves are the users who perform encryption. Therefore, in their case we are not interested in the messages they can decrypt, because they perform the encryption. Another aspect to note is that in order for a user to be able to decrypt a message there has to be a direct link between that user and the one who encrypted the message. Figure 1 presents such a tree structure.

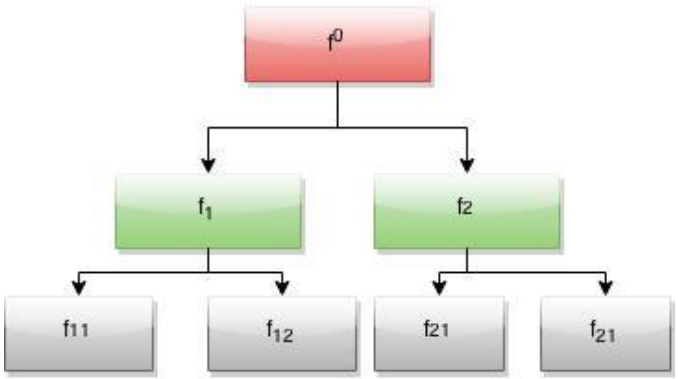

Figure 1. Tree structure example

If we look at the implementation, we are using multi-agent systems. Each user in the hierarchy is going to be represented by an agent. To increase security, we decided the key generation and distribution should be performed by an agent that will not be a part of the system itself. We will call this agent the 'key manager' for further reference. In order to generate the hierarchy, the key manager needs to know the exact number of levels – the value of k – the number of leaves and of course the total number of users. In the present case, this user is also the one who chooses the elliptic curve. Once all the data is generated and the keys are computed, the next step is to distribute the keys to their owners. The leaves will have the public keys, while the rest of the users will each receive a private key. After all the users have confirmed the receipt of their keys, the key manager agent shuts down. This will add to the security level of the system, as no agent will have knowledge of all the keys.

Encrypting and decrypting a message are performed according to the requirements and formulas of classic elliptic curve cryptography. The particularity in our system is that data is perceived as a set of messages. This means that when certain data is received, we have a user again outside of the tree structure, one who does not have any keys who will divide the data into small messages. The number of message equals the number of leaves in the system. Thus, each leaf receives a message and then proceeds to encrypting the data. We wanted the system to be as efficient as possible, therefore we built it in such a way that each leaf will only send encrypted data to its ascendants. However, because the cost of a failed decryption could be too big in a complex system, we verify that a user receiving an encrypted message does indeed have the right to decrypt it. Messages are not necessarily sent to all the users in the system. Certain data can have a chosen receiver.

While the encryption algorithm used is based on strong secure structures and we tried to implement the generation and distribution of the keys in the safest manner possible, there were still minuses to the algorithm. For this reason we added a steganography module, to increase the level of security and eliminate the weak points of the system.

B. Steganography module

Steganography is the art of hiding a message. Unlike cryptography, with steganography the data is not changed. In the present paper we use image steganography, more specifically the least significant bit (LSB) method. This method is a simple, yet effective way to hide data inside an image file. As the name suggests it, one will modify the LSB of an image, by replacing them with the data to hide. It is probably the most used method in image steganography and for this reason some might argue it is not very secure. However when used in addition to another method, cryptography in this case, the method is efficient. It provides an easy way to hide data, adding to the security layer but without slowing down the system – LSB is proven to be a fast method.

After the initial implementation of the system, we found that the key distribution step was rather unsafe. Keys were being sent by the key manager as simple messages between agents. This was an obvious minus as anyone could have intercepted those messages and read the keys. Therefore we added steganography to this step by embedding each key in an image before sending it. In order for this to be possible in multi-agent system, we first change each key to its binary form. This is a request of the LSB method. Once this operation is performed, the key is hidden inside the image. In terms of implementation, in order to send an image from one agent to another, that file needs to be changed into a byte array, operation performed simply with the command *SetByteSequenceContent()* available when using the JADE environment to model multi-agent systems. By using steganography at this stage we ensure that the keys are not immediately visible to any third party, all they would see are images being sent. Also, by modifying only the least significant bit, changes made to the original image are not visible to the naked eye.

Another way in which steganography can be used is to hide the encrypted data. Even though the encryption algorithm is strong, some intercepting that communication could notice the data is encrypted. To add to the security level one can hide the encrypted message in an image by following the exact same steps used to hide the keys. However, this operation could overload the system, especially when there is a large flow of data being sent. Therefore, we made embedding the encrypted data an optional step. The necessity of this step is decided by the same user to divide the data into messages for the leaves. Figure 2 provides an example of the main operations performed by the system at this stage. Once the system is "ready" it can receive data to be encrypted than forwarded to the users on the upper levels who will decrypt it and analyze it. We will describe this functionality in Section 4 of the present paper.

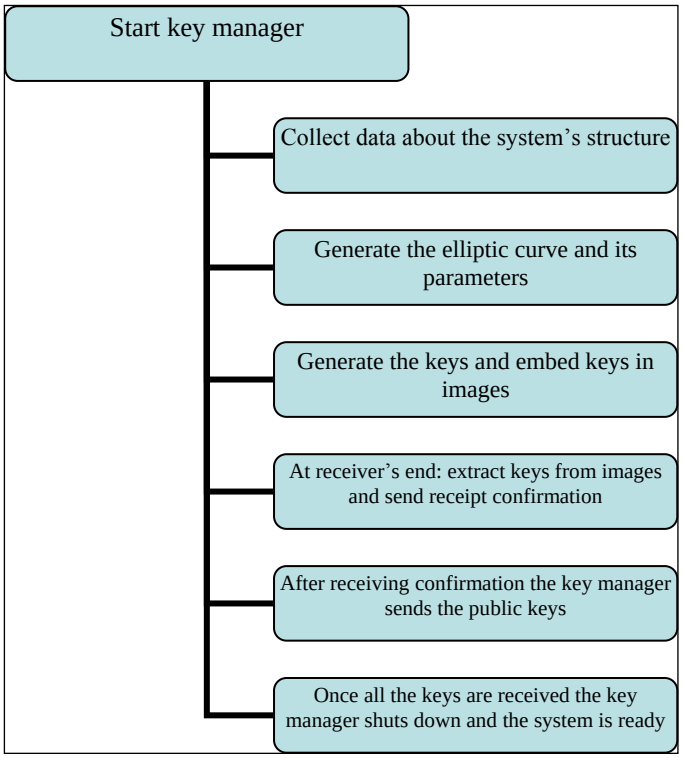

Figure 2.Main operations to start the system

C. *Digital signature module*

Digital signatures can be used for authentication, integrity or non-repudiation purposes. A cyber-physical system can benefit from signatures in all of these areas. In our case we use digital signatures for authentication and integrity purposes. In general, a signature can be applied on both plain text or encrypted data. We will only apply to encrypted data, as we do not work with unencrypted messages within the system.

Like their on-paper counterpart, digital signatures usually imply a document signed by one person, one user. Because the system described in the present paper is based on hierarchy, the messages will be signed by more users. More specifically, they should be signed by all the users on the same level. For this we use the signature generated by the algorithm ElGamal with divided private key [13]. The phases of this algorithm are the same as with any digital signature algorithm: generating the keys, signing the message and verifying the signature. If usually both the signing and the verification phase respectively are performed by one user each, in our case multiple user will sign the message, while only one will verify it.

1) *Theoretical aspects*

The first phase of the algorithm is generating the keys. Since there are several users signing, we will need as many keys as users involved in the algorithm. We begin by choosing a random number *q*, prime, for which the discrete logarithm problem is difficult and a number *g* called the generator. From the set *{1,…q-1}* the private keys are chosen. Each user has its own private key and the keys should be distinct and prime, if possible. The public key is computed in two steps. Firstly, each user computes equation (3), where x_i is the user's private key:

$$h_i = g^{x_i} \bmod q \tag{3}$$

Once all users compute their h_i, they will calculate together:

$$h = (h_1 \cdot h_3 \cdot \ldots \cdot h_{2n+1}) \cdot (h_2 \cdot h_4 \cdot \ldots \cdot h_{2n})^{-1} (\bmod q) \tag{4}$$

Note that *2n+1* is the total number of users – the algorithm specifically requests for an odd number of users to sign a message.

Within the signing phase, again each user has to compute a part of the signature. With these parts, they will compute together the final signature. For this, the users need the hash function of the messages, denoted with H(m). They will also need to choose a random number *y, 0<y<q-1*, prime with *q-1*. Each user will then compute:

$$r = g^y \bmod q \tag{5}$$

$$s_i = \left(\frac{1}{2n+1} H(m) - x_i r\right) \cdot y^{-1} (\bmod q-1) \text{ with } i = 1 \to 2n+1 \tag{6}$$

If any of the values s_i is 0 a new *y* has to be chosen and all computations performed with the new value. Once all the values are correct, the final signature will be (r,s), with s:

$$s = s_1 - s_2 + s_3 - s_4 + \ldots + s_{2n+1} (\bmod q-1) \tag{7}$$

The final phase of the algorithm is verifying the signature. From the mathematical point of view this means computing the equality:

$$g^{H(m)(\bmod q)} = h^r \cdot r^s \tag{8}$$

The verifier needs to know only the public key, the hash function of the message and of course, the signature. The proof of correctness of the algorithm can be viewed in paper [13].

2) *Algorithm implementation and usage*

Our goal regarding the algorithm was to implement it in a way that is flexible and allows for the signature to be usable with any system and not just the hierarchical structure we are working on. For this we used again the Java programming language and implemented the three phases of the algorithm each in its own class. These classes were later included in a package that can be imported and used in any Java program.

The signature sequence, independent from the hierarchical structure, is described in Figure 3.

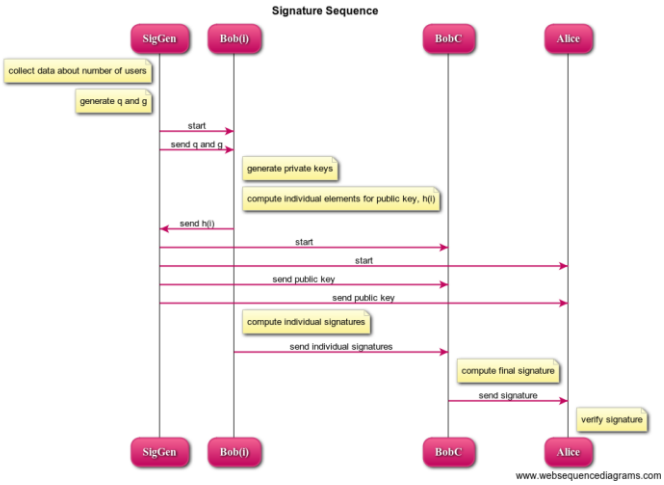

Figure 3.Signature sequence

The 'SigGen' from the above figure represents the 'signature generator' user. As we did with the hierarchy generation – using a key manager, an entity outside of the final structure – here we have an entity that collects the initial data. It needs the number of the users that will sign a message and it generates *q* and *g*. Bob(i) is a representation of the users that will sign a message, of the operations they perform individually – generate the private keys and their own part of the public key. This piece of the public key is sent back to the signature generator who in turn computes the final value of the public key and sends it to all the users. Once all the keys are sent, the signature generator is no longer needed. Further on, we observe in Figure 3 an entity called BobC, which will be used as the common ground for all users when calculating the signature. That is, each user computes its part of the digital signature, using the formulas (5) and (6). The results are sent to BobC. Here is where the final signature is calculated, based on equation (7). Finally, Alice is the signature verifier.

When integrating the algorithm within the hierarchical system, we focused on both efficiency and security. In a system that uses both encryption and steganography, digital signatures are not always necessary. Therefore, the operation is optional and requesting it is done before encryption, by the sender of the message. The request can be made either because the data is sensitive or there is suspicion an attempt to alter it will be made or to verify the integrity of certain users within the system. To note that we do not allow for both steganography and digital signature to be applied to an encrypted message. Allowing it would of course increase even more the level of security. However applying three security methods to the same methods seems both unnecessary and inefficient.

When deciding between the two methods to apply to the encrypted message one should take into consideration certain factors. Firstly, we should consider the amount of data to be secured. The greater the amount, the harder it will become to hide it in an image and the greater the load on the system. Thus, digitally signing that data might be a better idea. However, if we want to make sure it will not be visible to the naked eye that we are communicating secret data, steganography will be a better choice. Finally, if we want this extra layer of security to help us also verify the integrity of the users in the system, digital signature is the choice to make. Of course, we do not need to choose one of the two, we can simply choose to send the message in its simple, encrypted form.

If a signature is requested a few steps need to be taken. Firstly, an agent with the role of signature generator is started. It will have the exact same tasks as previously described. Bob(i) represent now the users in the tree structure that will sign the message, they are the ones with whom the signature generator will communicate. Finally, we start two more agents, BobC and Alice, with the roles already described.

Because we want all the users on a level to sign a message, we had to address the request of the algorithm: that a message can only be signed by an off number of users. It is obvious we cannot always have an odd number of users on a level. Therefore, when the number is even, the root of the tree will participate in the signature. The root is the user with the highest degree of access and using it to have an odd number of users, when necessary, is a useful operation. Firstly because this user will have access to any messages, therefore we do not need to worry about access rights. Secondly, because the algorithm also states a message cannot be signed by only one user. Therefore we could never request only the root to sign, so we would not be able to verify its integrity.

IV. Results and Discussion

The proposed security architecture can be used on several types of cyber-physical systems. It is built in flexible way, there can be any number of users as long as the tree structure is respected. As our case study we used the Cluj-Napoca city SCADA system for water supply. The first test was conducted on a smaller part of the system, more specifically the part that deals with data analysis of flow and water pressure. The simulation of the physical components was done using LabView, a software created by National Instruments. We will not go into detail regarding the LabView simulation of the water supply system, as we are not interested in how that is done, but on the output data. This data is collected in files that are later taken for analysis by the hierarchical system.

Our interest in this case study was not so much on the hardware part of the system but on the data analysis. Periodically the system will output data such as flow and pressure of water. This data will is collected and at certain amounts of times it will be sent for analysis. The system that analyses the data will be modeled as a hierarchical system. Here hierarchy will not be focused on who has a greater degree of access, it will be focused on who has access to what. This means that a level will have the role of gathering and studying the data regarding abnormal flow, another level deals with abnormal pressure and so on. Figure 4 presents a scheme of the data analyzing system.

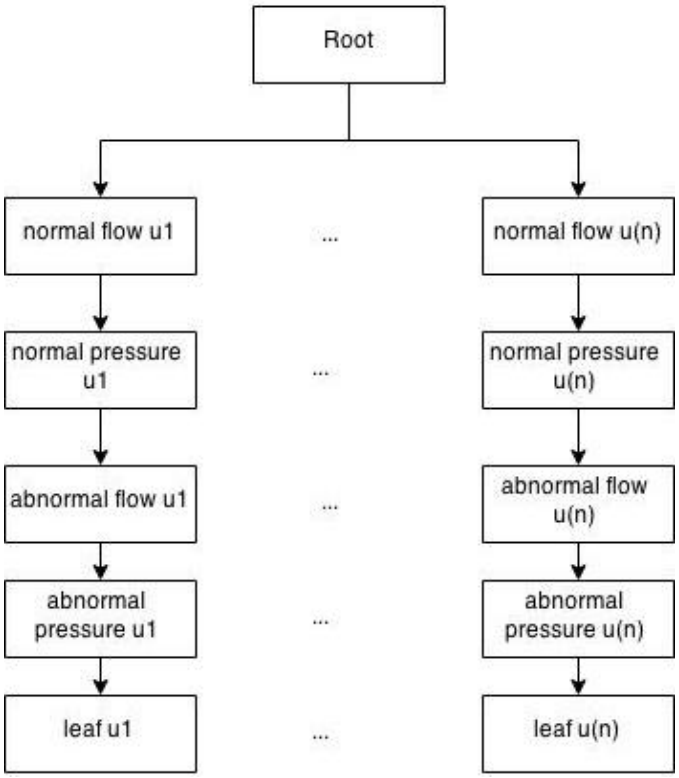

Figure 4. Pressure and flow analysis system scheme

We conducted the case study on a simple tree structure in which every node had two children. Figure 4 presents the role of each level. Thus, counting from top to bottom, with the root on level 0, the users on level 1 will analyze data when the water flow is considered normal. In the same way, level 2 users will analyze data when the water pressure is normal. Levels 3 and 4 analyze the abnormal values of flow and pressure respectively – both too high and too low values. Finally, level 5 is represented by the leaves.

We found that using a hierarchy added to the efficiency of the system. Dividing operations with the aid of a private key that limits access to data, knowing exactly which level needs to receive certain data is helpful in increasing efficiency of the system. An interesting aspect was choosing which levels should receive the data about abnormal results and which should receive the rest. We decided that we need more users analyzing the abnormal result, therefore, due to the tree structure, we would have more users on the lower levels, in this case levels 3 and 4. This is not because we expect to get more abnormal results with flow and pressure, but because this data would need a more careful analysis. As previously described, the system divided the information in sets of messages. The lower the level in the hierarchy, the smaller the message will be. Ideally, each user analysis at most two or three abnormal results per cycle, to ensure that work is done properly. Normal results will not need as much analysis, therefore each user on levels 1 and 2 respectively can receiver larger amount of data.

Figure 5 presents the flow chart of the operations performed to secure data.

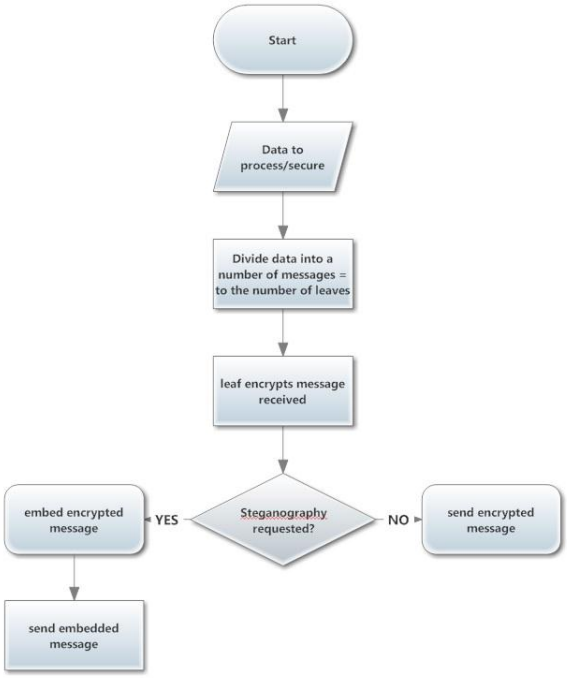

Figure 5. Securing data

Note that the flow chart in Figure 5 presents only for either encrypting or hiding data. The same steps would be followed in case we would want to verify is a digital signature has been requested.

Figure 6 described the operations performed by the users on the upper levels in order to retrieve secure data.

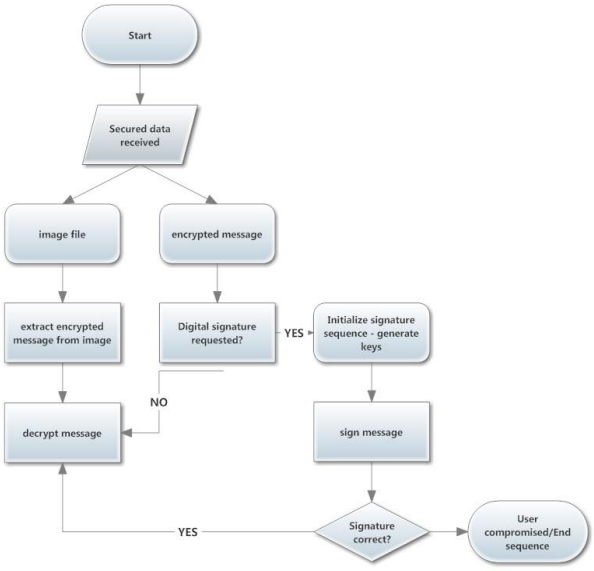

Figure 6. Steps to decrypt a message

The operations presented in Figure 6 repeat every time an encrypted message is received. The first step is to verify if the received message is in image or encrypted format. If it is an image, the receiver will proceed to extract the data and then decrypt it. Otherwise, there might be a request for a digital signature to verify the integrity of the level. If such a request exists, the steps presented in the previous chapter to sign a message are taken. If the signature is correct the message can be decrypted. Otherwise, we have the certainty one of the users has been compromised and decryption is not allowed. If the request for a signature does not exist, the user simply decrypts the message.

V. CONLUSIONS

The work described in the present paper focuses on building a security architecture for cyber-physical systems. We use a new approach by combining three types of security methods: encryption, steganography and digital signatures while also using hierarchical access. We conducted our case study on the Cluj-Napoca SCADA water supply system, more specifically on the module that deals with analyzing water pressure and flow. The results concluded that the data analyzing process was more efficient and more secure. Future work will focus on testing the algorithm on other modules of the water supply system and eventually on integrating within the hardware components.

ACKNOWLEDGMENT

This paper is supported by the Sectorial Operational Program Human Resources Development (SOP HRD), ID/134378 financed from the European Social Fund and by the Romanian Government.

REFERENCES

[1] Adebayo Omotosho, Omotanwa Adegbola, Olaniyi Olayemi Mikail, Justice Emuovibofarhe, “A Secure Electronic Prescription System Using Steganography with Encryption Key Implementation”, International Journal of Computer and Information Technology, Volume 03-Issue 05, September 2014

[2] Devotha Nyambo, Zaipuna O. Yonah, Charles Tarimo, “Review of Security Frameworks in the Converged Web and Mobile Applications”, International Journal of Computer and Information Technology, Volume 03-Issue 04, July 2014

[3] Anand, D.; Khemchandani, V.; Sharma, R.K., "Identity-Based Cryptography Techniques and Applications (A Review)," Computational Intelligence and Communication Networks (CICN), 2013 5th International Conference on , vol., no., pp.343,348, 27-29 Sept. 2013

[4] Arun Kumar Shakar, ”Enhancing the Data Security Features of Communication by Means of Media Files through Improvising the Cryptographic and Steganographic Techniques”, ASM’s International E-Journal of Ongoing Research in Management and IT, 2013

[5] Abdelraham Altigani, Bazara Barry, “A hybrid approach to secure transmitted messages using advanced encryption standard (AES) and word shifting protocol”, International Conference on Computing, Electrical and Electronic Engineering (ICCEEE), 2013

[6] Riadh W. Y. Habash, Voicu Groza, Kevin Burr, “Risk Management for Power Grid Cyber-Physical Security”, British Journal of Applied Science & Technology, Volume 3, Issue 4, July 2013.

[7] Mamta Juneja, Parvinder Sandhu, ”An improved LSB based Steganography with Enhanced Security and Embedding/Extraction”, 3rd International Conference on Intelligent Computational Systems, Hong Kong, China, January 2013.

[8] Vipula Madhukar Wajgade, Dr. Suresh Kumar, ”Enhancing Data Security Using Video Steganography”, International Journal of Emerging Technology and Advanced Engineering, Aprilie 2013.

[9] Chander Kant, Rajender Nath, Sheetal Chaudhary, ”Biometrics Security using Steganography”, International Journal of Security, Volume 2, Issue 1

[10] K. Usman, S. Aleksandar, “Security in cyber-physical energy systems”, Workshop on Modelling and Simulation of Cyber-Physical systems (MSCPES), 20-23 May 2013.

[11] M. Grace Venice, Prof. Tv. Rao, ”Hiding the Text Iinformation using Steganography”, International Journal of Engineering, Research and Application, Vol. 2, Ian-Feb 2012, pp. 126-131.

[12] Tayana Morkel, ”Image Steganography Applications for Secure Communication”, Dissertation Thesis, University of Pretoria, May 2012

[13] S. Flonta, V. V. Patriciu, L. C. Miclea, Metode criptografice pentru sisteme structurate (Cryptographic methods for structured systems), U.T. Press, Cluj-Napoca, Romania 2011

[14] V. Valli Kumari, D.V. NagaRaju, K. Soumya, K.V.S.V.N. Raju , “Secure Group Key Distribution Using Hybrid Cryptosystem”, Machine Learning and Computing, pp. 188-192, February 2010.

[15] Dipti Kapoor Sarmah, Neha Bajpai, ”Proposed System for Data Hiding using Cryptography and Steganography”, International Journal of Computer Applications, 2010

[16] Byoungcheon Lee, "Unified Public Key Infrastructure Supporting Both Certificate-Based and ID-Based Cryptography," Availability, Reliability, and Security, 2010. ARES '10 International Conference on , vol., no., pp.54,61, 15-18 Feb. 2010

[17] Soon M. Chung, “Role-Based Access Control for Cyber-Physical Systems Using Shibboleth”, Washington University, June 2009

[18] Iuon-Chang Lin , Chin-Chen Chang, “A Novel Digital Signature Scheme for Application of Document Review in a Linearly Hierarchical Organization”, 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing, August 2008, Proceedings pp. 1367-1370.

[19] Zhenfei Zhang; Susilo, W.; Raad, R., "Mobile ad-hoc network key management with certificateless cryptography," Signal Processing and Communication Systems, 2008. ICSPCS 2008. 2nd International Conference on , vol., no., pp.1,10, 15-17 Dec. 2008

[20] Partha Pal, Rich Shantz, Kurt Ruhloff, Joseph Loyall, “Cyber-Physical Systems Security – Challenges and Research”, BBN Technologies, Cambridge, Available at: http://cimic.rutgers.edu/positionPapers/CPSS_BBN.pdf

[21] Dr. Clifford Neuman, ”Challanges in Security for Cyber-Physical Systems”, Available at: http://cimic.rutgers.edu/positionPapers/CPS-Neuman.pdf

[22] Indrakshi Rai, Indrajit Ray, “Access control for cyber-physical systems”, Colorado State University, Available at: http://cimic.rutgers.edu/positionPapers/paper Indrakshiray.pdf

[23] Vegh, L.; Miclea, L., "A new approach towards increased security in cyber-physical systems," Systems, Signals and Image Processing (IWSSIP), 2014 International Conference on , vol., no., pp.175,178, 12-15 May 2014