Impact on the inclusion of security in the UPnP protocol within the Smart Home

Alberto Alonso Fernández, Alejandro Álvarez Vázquez, M.P. Almudena García Fuente, Ignacio González Alonso
Computer Science Department University of Oviedo
Oviedo, Spain
alonsoalberto@uniovi.es, alvarezvalejandro@uniovi.es, agarciaf@uniovi.es, gonzalezaloignacio@uniovi.es

***Abstract* — This paper describes the impact caused by an encryption security system on a protocol for interoperability between robots and home automation. DHCompliant is an open source interoperability protocol supported by the UPnP standard. Until today, UPnP does not provide mechanisms for secure communications, since messages are transmitted over the network unencrypted and anyone can intercept and read its contents. The proposed security system is intended to provide DHCompliant with a dual security mechanism based on RSA and AES algorithms. The use of these algorithms can influence the performance of the protocol and the present work is focused on describing the real impact of the inclusion of such security mechanisms. Our results show that hiding information in a Smart Home interoperability protocol by the inclusion of a security system is viable and does not imply great consequences in CPU memory consumption.**

Keywords* – *DHCompliant; Security; Data Encryption; UPnP.

I. INTRODUCTION

Security and interoperability are key issues in computer systems. In a system designed for the Digital Home, in which several technologies coexist handling data from devices as well as from the users, it is needed to cover the security of them as well as the interoperability of the whole system.

A. Security in the Digital Home

Having smart devices in the Digital Home is very useful. Once all the devices in a home are automated and connected through a network, it is important to consider security issues, authentication and access control [1]. There is a need for each device and each user to be authenticated in the system at the same time in order to interact. Regarding the interoperability protocols into the Smart Home, information related to its inhabitants and its habits are managed. This information is confidential and mechanisms, which make it inaccessible and/or illegible for entities from outside the Home, must be developed. At the same time, the devices that compose the system must be validated and accomplish a group of requisites in order to be part of the web, avoiding malignant devices to take control of the installation or allow a leak of information. The information managed in these environments includes all the values gathered by all the Smart Home sensors, as well as behavior patterns of the inhabitants (e.g., daily tasks, timetables and other personal information).

Without the existence of security in the Smart Home, its inhabitants' personal life information is exposed. It is necessary that this situation does not occur in order to extend the concept of Smart Digital Home in the society, this way the users will trust a system with a high level of reliability, which does not allow situations in which information and devices can be compromised.

B. Interoperability and security

The development of software systems incorporating heterogeneous components has a great potential, reducing costs and increasing productivity and flexibility to future changes, but on the other hand it is prone to suffer threats in non functional aspects of the system [2]. One of the problems identified is how to build a secure system from components, which may or not be safe by themselves. In this study, an example of components can be robotic adapters developed in different programming languages and executed in different platforms, the OpenID identity supplier or the software component, which administer the control and events of the home automation installation inside the Digital Home. The security of all the system cannot fall on an only component and the interoperability in the security of integrated systems is not a trivial problem [3]. It is possible that each component can implement different policies and security mechanisms, which may not be interoperable among them. This is the reason why it is highlighted the need of providing these systems with security mechanisms common to all components in order to preserve interoperability among them with a security guarantee.

Another aspect to be considered is the quality of the service provided (QoS) [4]. The main concern is the delay that may occur to access, transmit and display the information, which is exchanged in the Digital Home environment. In order to guarantee all the aspects previously stated, in the present study different options regarding security issues were evaluated. The principal aim was to choose a group of security mechanisms and algorithms already proven that endow a domo-robotic interoperability protocol with the security needed for preserving communications and confidential information that can circulate through the network.

C. Digital Home Compliant (DHCompliant)

DHCompliant [5] project aims to integrate home automation and robotics in the digital home and media communications network based on the Universal Plug and Play (UPnP) technology [6]. DHCompliant proposes a solution to develop collaborative tasks between robots

 ISBN: 978-1-61208-165-6

taking into account the information that home automation devices can provide, such as lighting conditions, humidity parameters or presence detection. All the information is handled to perform tasks managed by UPnP. From the automatic discovery of devices to remote invocations of robot actions are controlled by the UPnP protocol.

The paper is structured as follows: Section number two breaks down the current state of the art in the field of security and it is exposed the main motivation for this work. Section three describes the methodology used and Section four describes the experiments that have been included. Finally, Section five presents the results obtained and Section six assess all these results to draw conclusions and propose several future works.

II. MOTIVATION AND STATE OF THE ART

The main motivation of this study is to assess the impact of the proposed security system for protection of communications in the digital home. Due to the lack of security in the UPnP protocol, it has been studied the mechanisms and security encryption algorithms to choose an optimal solution to provide the required security system to safeguard the privacy of users. Today the latest specifications of the UPnP protocol does not provide any security mechanism for messages transmitted over the network or to authenticate users on the network as well as concepts of privacy.

One of the goals of this study is to provide a safety mechanism for interoperability protocol DHCompliant based on UPnP. Another goal is to evaluate how it affects the security system on the overall performance of the protocol.

In the present section, the main data encryption systems will be presented, as well as the DHCompliant protocol.

A. Data encryption

1) RSA (Rivest, Shamir y Adleman)

It is a public key cryptographic system developed in 1977. The safety of this algorithm lies in the problem of factoring integers. Sent messages are represented by numbers, and the operation is based on the product of two random large prime numbers in a secret way.

When you want to send a message, the speaker looks for the recipient's public key, encrypts the message with that key, and once the encrypted message reaches the receiver, it´s decrypted using its private key.

RSA was believed to be safe until it was not known the quick ways to decompose a large number of prime products. Quantum computing could provide a solution to this problem of factoring.

RSA is used in multiple applications including electronic cash, secret broadcasting, secret balloting systems, various banking and payment protocols, smart cards, and biometrics [7].

2) AES

Advanced Encryption Standard (AES), also known as Rijndael is a schematic block cipher adopted as an encryption standard by the U.S. government.

AES has a fixed block size of 128 bits and key sizes 128, 192 or 256 bits. Rijndael is a block cipher with both a variable block length and a variable key length. It would be possible to define versions of Rijndael with a higher block length or key length, but currently there is no need for it [8]. By design, the DES and TDES are slow algorithms. AES can be up to 6 times faster and, besides, not vulnerable [9].

AES has multiple libraries for the development of secure applications in several programming environments as C, C++, Java, C# o Python. Among all its uses, file compression, disk encryption, security in local networks (LAN) or as part of other applications as GPL [10] o Pidgin [11] are highlighted.

3) DES and 3DES

Data Encryption Standard (DES) is a method for encrypting information, chosen as FIPS in the United States in 1976, its use has spread widely throughout the world, [12].

Today, DES is considered insecure for many applications. This is mainly because the key size of 56 bits is short. DES keys have been broken in less than 24 hours. There are also analytical results, which demonstrate theoretical weaknesses in the cipher, although they are unworkable in practice. It is believed that the algorithm is safe in practice as a variant of Triple DES, although there are theoretical attacks.

Triple DES is also known as TDES or 3DES, was developed by IBM in 1998 [13]. The Triple DES is slowly disappearing, being replaced by the AES algorithm. However, most credit cards and other electronic payments have as standard Triple DES algorithm (previously used the DES) [14]. By design, the DES and TDES algorithms are slow.

4) BLOWFISH

Is a public domain symmetric block encoder, designed by Bruce Schneier [15] in 1993 and included a large number of sets of encoders and encryption products. While no analyzed Blowfish cipher has been found effective today, it has been given more attention than decoding blocks with larger blocks, like AES.

Blowfish was designed as a general purpose algorithm, which attempted to replace DES and avoid the problems associated with other algorithms for use in performance-constrained environments such as embedded systems [16].

5) IDEA

Is a block cipher designed by Lai and James L. Xuejia Massey of the Federal Polytechnic School in Zurich and was first described in 1991 [17]. An algorithm was proposed as a replacement for DES.

The designers analyze IDEA to measure its strength against differential cryptanalysis and concluded that it is immune under certain assumptions. Non successful linear or algebraic weaknesses have been reported. One of the most popular uses is within the framework of PGP [18].

B. DHCompliant architecture

DHCompliant protocol is divided into a number of subsystems that can meet existing needs in a home automation environment. It is a protocol set up over UPnP and it includes the following subsystems: Groups, Localization, Intelligence, Energy, and Security & Privacy.

 ISBN: 978-1-61208-165-6

- **DHC-Groups:** Is the service that manages the collaborative tasks. It transmits to the connected robots the task information to be executed and responds to requests that they are later made to form a hive of robots capable of performing a particular task.
- **DHC-Localization:** Allows obtaining the position of the robots in the house. The robots take the coordinates of the location system to navigate to the point where the task is performed.
- **DHC-Energy:** Enables power profiles management to perform collaborative tasks and calculations of costs and fees for expenditure control.
- **DHC-Intelligence:** Here are included semantic tagging capabilities, building and testing user-defined rules and machine learning. In this module is the Machine Learning [19] technology that provides the system with learning capabilities for making decisions in a more autonomous way.
- **DHC-Security&Privacy:** Allows the encryption of communications in the DHCompliant UPnP network protocol established [20]. Through the RSA asymmetric encryption algorithm is sent to all devices on the network a system password to be used by the AES encryption as its symmetric key. In the next section you can see the process in a more detailed reflection in a SysML diagram of sequence (Figure 2).

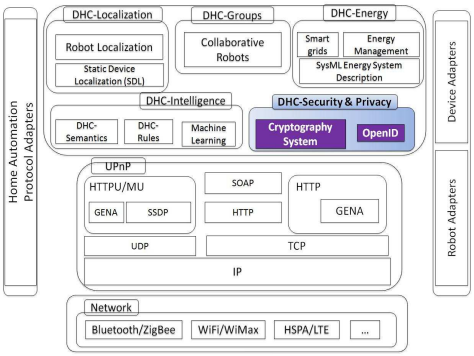

Figure 1. DHCompliant Architecture.

III. METHODOLOGY

This section describes the tools and elements required for including a security system into the DHCompliant protocol, as well as for performing the experiments.

A. Tools

To carry out the tests several tools have been used. A simple and effective technique has been used for measuring execution times for the .NET platform. It consists of the use of the basic classes and methods to measure time like TimeSpan and the attribute Ticks.

The method consists of the introduction of a Date .Now instruction in the source code at the beginning of what it is wanted to measure and a statement at the end of the method or code section. The two times are subtracted to get how many milliseconds.

To analyze the performance of the system it has been chosen a profiler for the NET platform, the YourKit Profiler [21]. It provides zero-overhead profiling for your .NET applications and makes code profiling and memory usage optimization simple and fast. The remote option has been used in all the experiments because it does not interfere with measurements. Measuring time and resources usage remotely is needed to obtain the better results.

B. Items

It can be distinguished two types of elements in consideration in conducting the experiments, hardware items and software items.

1) Hardware items

The following table (Table 1) describes the characteristics of the equipment used to perform all experiments.

TABLE I. LIST OF HARDWARE COMPONENTS USED FOR EXPERIMENTS

Computer	1	2
OS	Windows XP Profesional 32 bits	Windows Vista Business 64 bits
CPU	AMD Athlon X2 4000+ 2,11GHz	AMD Athlon X2 4000+ 2,11GHz
RAM	3GB DDR2	2GB DDR2
HD	Western Digital 7.200 rpm	Western Digital 7.200 rpm

The computer number one is the machine that contains all the DHCompliant system. The computer number two is responsible for running an instance of the YourKit profiler to run tests remotely.

2) Software items. DHCompliant protocol.

DHCompliant protocol modules involved throughout all the tests are the following:

- **GUI:** It is the user interface from where the tasks are created and launched to be performed by the robots. The interface consists primarily of a form in which the user enters data for the task as the task name, the number of robots that are to be used, the target room and other necessary parameters for the job. It also allows the creation of user rules, selection of the energy profile for the task and the cancellation of tasks.
- **DHC:** Is the main part of DHCompliant. It contains all the protocol services: DHC-Groups, DHC-Localization, DHC-Energy, DHC-Intelligence and DHC-Security.
- **Adapters:** It is the software component that acts as a link between the physical robot and the DHCompliant protocol. It implements all the protocol functionality to perform the tasks sent by the user. It communicates with the API of each robot to use its features [22]

The experiments described in the following sections have been carried out to demonstrate what is the real impact of the inclusion of security and privacy in an interoperability protocol. The goal is to demonstrate what is the time penalty and performance when compared with the same protocol without restraint.

IV. DESIGN OF THE EXPERIMENT

The experiment was performed to study the impact of the security system consisted in executing a video surveillance task within the protocol DHCompliant. The objective is to perform a task from the user interface to be carried out by a

 ISBN: 978-1-61208-165-6

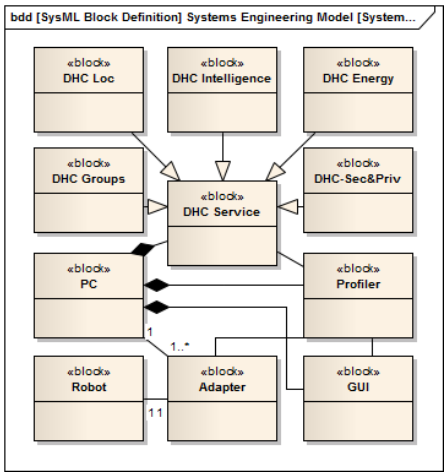

Figure 2. SysML package diagram of the experiment components.

robot. The DHC module is located between the robot and the user interface, and is responsible for the tasks management (choose appropriate robot, location service, energy service ...).

Because the encryption is included in each of the entities involved in the flow of execution of a task, it was decided to divide the experiment into three stages to obtain more accurate results and more data for analysis. One stage was chosen for the flow of execution in the graphic user interface, another for the DHC device, and the last one for the robot adapter.

First, the GUI generates the internal system, which will be the future symmetric key for the AES encryption algorithm to encrypt all protocol communications. This key must be shared with other devices in a safe way, so it is sent encrypted using the RSA algorithm.

Once the devices (DHC and adapters) are subscribed to the UPnP security service of the GUI, they perform an invocation to obtain the system key. The devices also implement the RSA algorithm so in the previous involution the GUI sends your public key. Next, the GUI key system encrypts the public key of each device. The value returned by the invocation is the key encrypted with the public key of each device that relies on the security action interface. After receiving the key, the device decrypts with its private key and initializes the AES symmetric cipher with the key obtained.

Once the processing is completed, the devices can subscribe to other services of the encrypted communications system.

V. RESULTS

This section describes the results obtained with and without the inclusion of a security system in DHCompliant.

A. *Time measurements*

Measure ranges were the following:

- In the interface, time was measured since the launching of a task until the last change of state variable (including the specification of the energy profile).
- In the DHCompliant central system, time was measured since the detection of the first state variable change until the last change of variable.
- In the robot adapter, time was measured since the change in the TaskID variable until it receives the response from the first request for coordinates to the location system.

The tables (Tables II and III) show the measurements obtained, with a system in which the data is unencrypted and another system in which data is encrypted.

TABLE II. TIME MEASUREMENT WITHOUT DATA ENCRYPTION

Iteration Number	Average Time (ms)
Interface	49,99696
DHC	462,29
Adapter	17946,77
Total	**18459,06**

TABLE III. TIME MEASUREMENTE WITH DATA ENCRYPTION

Iteration Number	Average Time (ms)
Interface	112,492
DHC	1118,722
Adapter	18444,75
Total	**19675,96**

B. *CPU and memory consumption*

With the profiler it has been taken samples from memory and CPU consumption during the course of carrying out the experiment described above. To obtain more reliable results without interferences, tests have been carried out with the remote profiler option.

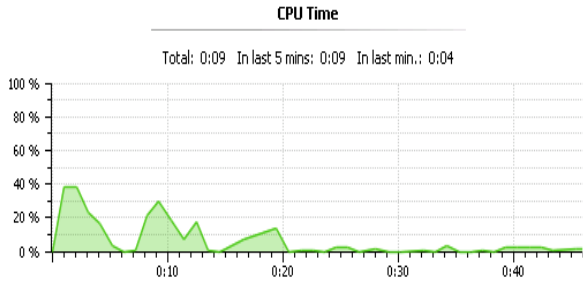

Figure 3. CPU time consumed in the adapter without encryption

As it has been described in previous sections, the task was divided in three sections, one for every device involved and tests have been performed on the system with and without the encryption system. Samples from memory and CPU consumption have been taken with the profiler during the course of carrying out the experiment described above. To obtain more reliable results without interferences, tests have been carried out with the remote profiler option.

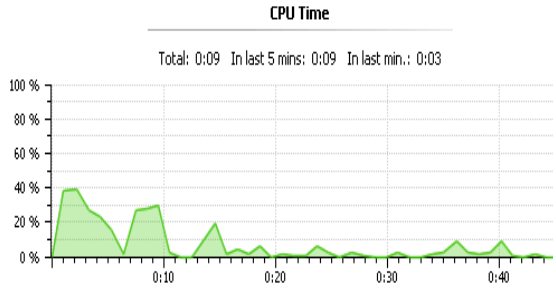

Figure 4. CPU time consumed in the adapter with encryption

The pictures above show the most significant results obtained. Figure 3 and 4 illustrate the task execution flow in

the adapter in each case. This flow starts when the adapter is started until it receives the last task parameter from the DHC device.

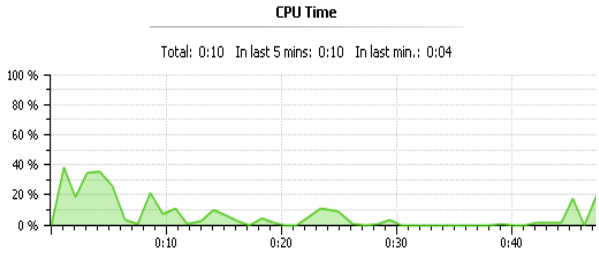

Figure 5. CPU time consumed in DHC without encryption

At the beginning there is no difference between two systems in terms of CPU load, but in the final moments it is noticed a small increase in the adapter with encryption system due to the obtained data from the task. The adapter ask DHC device for the task parameters so DHC answers the adapter with those parameters encrypted and the adapter must decrypt them. This process has a little increase of about 5% in the CPU load.

Figure 5 and Figure 6 show the results in the DHC device with and without the encryption system, respectively. In the first 15 seconds the adapter receives all the task parameters from the user interface and this information has to be decrypted. In this case, a peak in the encrypted system can be seen. This is because DHC receives parameters encrypted and it has to decrypt and encrypt them again to send them to the adapter.

At about the twentieth third second a small increase in CPU load occurs. This stage corresponds to the stage when the adapter asks DHC for task information.

Figure 7 and Figure 8 shows the memory consumption in the user interface device with and without encryption system. All memory generations are shown. The encryption system consumes 1 MB of memory more than the non-encrypted.

VI. CONCLUSIONS AND FUTURE WORKS

In this paper it is shown the most widely used encryption

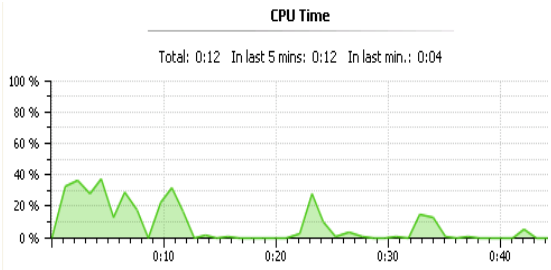

Figure 6. CPU time consumed in DHC with encryption

algorithms applied to a Smart Home protocol. Each system has its advantages and disadvantages but it has been decided to use RSA and AES systems for several reasons.

In the case of RSA, the main advantage of public key cryptography is an increased security and comfort, as the private key is not sent to any network device. In a secret key, however, the secret keys must be transmitted (either manually or through a communication channel), because the same key is used for encryption and decryption. A major problem is that there may be a possibility that an intruder can discover the secret key during transmission. This is why the private key is transmitted using the RSA system. The function of using a system based on public/private key

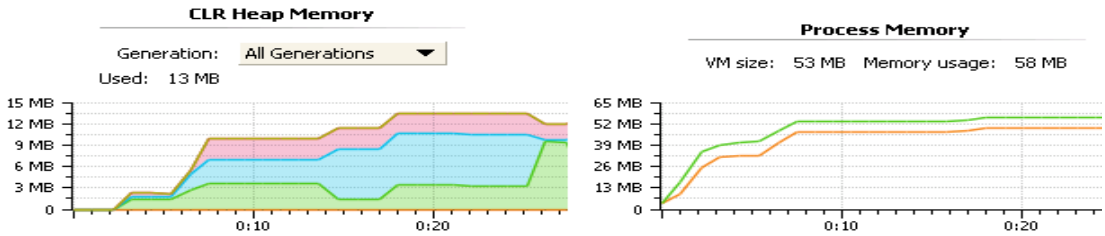

Figure 7. Memory usage in the GUI device with encryption system

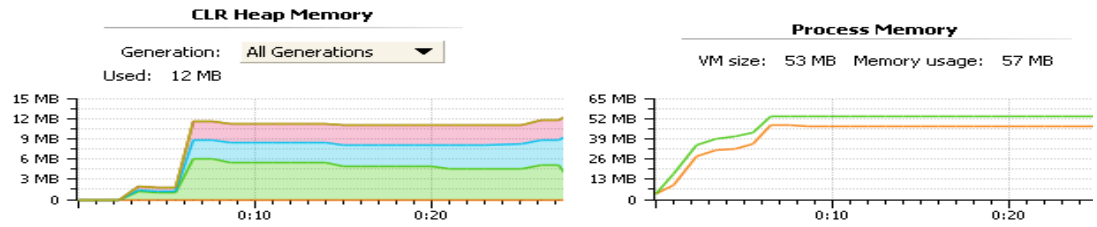

Figure 8. Memory usage in the GUI device without encryption system

encryption is to guarantee transmission of the AES key used to encrypt communications.

In the case of AES, the National Institute of Standards and Technology (NIST) with the joint work of Belgian researchers Vincent Rijmen and Joan Daemen selected Rijndael in October as a basis for AES. Rijndael was selected from among five finalists in a process that took more than three years [23]. Compared with other AES encryption algorithms, Rijndael had more elegant mathematical formulas behind, and only requires one pass to

 ISBN: 978-1-61208-165-6

encrypt the data. AES has been designed from scratch to be faster, unbreakable and capable of supporting smaller computing devices imaginable. The big differentiators between AES and other systems are safety, superior performance and better use of resources. Another reason to choose AES is that it provides strong encryption and has been selected by NIST as Federal Information Processing Standard in November 2001. In June 2003 the U.S. Government (NSA) announced that AES is secure enough to protect classified information up to TOP SECRET level, which is the highest level of security over the information, and which disclosure to the public would cause exceptionally damage to national security.

The experiments performed in this paper show that the inclusion of an encryption system in a protocol of interoperability provokes only a slight increase in consumption of RAM and CPU. Taking this into account, it can be concluded that the inclusion of a security system in the interoperability protocol in the Smart Home hides information is viable.

As future work, it would be interesting to implement other encryption systems for the DHCompliant protocol and compare them with the proposed solution of RSA + AES in order to get real data on the performance of each of the alternative algorithms. It is advisable to extend the encryption system to not only to encrypt the contents of the variables that contain information of the tasks within the digital home, also to encrypt the names of these variables.

Another aspect is to consider in the future the implementation of policies and recommendations on privacy issues. For products made in the European Union, the system proposed must comply with the Data Protection Directive 95/46/EC (European Union, 1995) and Regulation (EC) 45/2001 (European Union, 2001) and according to the instructions of the European Data Protection Supervisor (European Data Protection Supervisor, 2010). For products made in USA, it must comply with the Guide to Protecting the Confidentiality of Personally Identifiable Information (PII) (NIST (National Institute of Standards & Technology), 2010) [23]. Finally, Adapter, Robot or DHC service manufacturer MUST comply with the ISO / IEC 27002 (ISO / IEC - International Standard Organization, 2009) for information security.

REFERENCES

[1] J. A.-M. Manish Anand and R. C. M. Dennis Mickunas, «Secure Smart Homes using' Jini and UIUC SESAME», *ACSAC '00 Proceedings of the 16th Annual Computer Security Applications Conference*, 2000.

[2] E. A. O. Lawrence Chung, «Analyzing Security Interoperability duringComponent Integration», *IEEE/ACIS International Conference on Computer and Information Science*, 2006.

[3] K. M. K. J. Han,, «Composing security-aware software», *IEEE Software*, vol. 19, pág. 34–41, Feb. 2002.

[4] C. L. Samuel Pierre, «Security, Interoperability, and Quality of ServiceAspects in Designing a TelecommunicationsPlatform for Virtual Laboratories», *IEEE Electrical and Computer Engineering*, 2000.

[5] Infobotica Research Group, «DHCompliant web site», *dhcompliant.com*, 2010. [Online]. Available: http://dhcompliant.com/. [Accessed: 04-Mar-2011].

[6] «UPnP Forum», *http://www.upnp.org/*, Oct-2010. [Online]. Available: http://www.upnp.org/. [Accessed: 10-Nov-2010].

[7] Richard A. Mollin, *RSA and Public-Key Cryptography*. Chapman & Hall/CRC, 2002.

[8] Joan Daemen, *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer, 2002.

[9] A. A. H. Abul Ahsan and Md. Mahmudul Haque, «A Comparative Study of the Performance and Security Issues of AES and RSACryptography», presented at the Third 2008 International Conference on Convergence and Hybrid Information Technology, 2008.

[10] gnupg.org, «The GNU Privacy Guard - GnuPG.org», *The GNU Privacy Guard - GnuPG.org*. [Online]. Available: http://www.gnupg.org/. [Accessed: 26-May-2011].

[11] pidgin.im, «Pidgin, the universal chat client», *Pidgin, the universal chat client*. [Online]. Available: http://www.pidgin.im/. [Accessed: 26-May-2011].

[12] Mikael J. Simovits, *The Des: An Extensive Documentation and Evaluation of the Data Encryption Standard*. Aegean Park Pr, 1996.

[13] «IBM Press room - 1998-02-23 IBM Offers S/390 Customers Wider Safety Net to Conduct e-business - United States», *IBM Offers S/390 Customers Wider Safety Net to Conduct e-business*, 1998. [Online]. Available: http://www-03.ibm.com/press/us/en/pressrelease/2780.wss. [Accessed: 26-May-2011].

[14] William C.Barker, «Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher». NIST, 2008.

[15] «Schneier on Security». [Online]. Available: http://www.schneier.com/. [Accessed: 09-Mar-2011].

[16] Bill Gatliff, «Encrypting data with the Blowfish algorithm», Ago-2003.

[17] José M. Granado, Miguel A. Vega-Rodríguez, Juan M. Sánchez-Pérez, and Juan A. Gómez-Pulido, «IDEA and AES, two cryptographic algorithms implemented using partial and dynamic reconfiguration», Microelectronics Journal, Jul-2009.

[18] «The International PGP Home Page». [Online]. Available: http://www.pgpi.org/. [Accessed: 27-May-2011].

[19] Nils J. Nilsson, «Introduction to Machine Larning». Stanford University, Nov-1998.

[20] Infobotica Research Group, «Draft specification for data protection, user data privacy and access restriction». Dic-2010

[21] «.NET Profiler - Java Profiler - The profilers for .NET and Java professionals». [Online]. Available: http://www.yourkit.com/.net/profiler/index.jsp. [Accessed: 03-Mar-2011].

[22] Alejandro A. Vázquez, Ignacio G. Alonso, and M.P. Almudena García Fuente, «UPnP adapter for collaborative tasks development over the open protocol DHCompliant», presented at the INTERA 2011, Oviedo, Spain, 2011.

[23] «Goodbye DES, Hello AES», Jul-2001. [Online. Available: http://www.networkworld.com/research/2001/0730feat2.html. [Accessed: 07-Mar-2011].

 ISBN: 978-1-61208-165-6