A Security Scheme for Home Networked Appliances

Mazhar Ul Hassan, David Llewellyn-Jones, Madjid Merabti
School of Computer and Mathematical Sciences
Liverpool John Moores University
Byrom Street, Liverpool L3 3AF, UK
Email: mazhar77@gmail.com, {D.Llewellyn-Jones, M.Merabti}@ljmu.ac.uk

***Abstract*-The term peer-to-peer refers to the concept that in a network of equals (peers) using appropriate information and communication systems, two or more individuals are able to spontaneously collaborate without centralized coordination. A Network Appliance is defined as a dedicated function consumer device with an embedded processor and a network connection. Security for distributed peer-to-peer devices in distributed network environments presents many challenges, and remains a largely unresolved issue. Considering security, various schemes have been proposed, however research shows various weaknesses within the reported solutions. We have proposed and implemented Home Networked Appliances Security Scheme (HNASS) to secure communication among peers utilizing the services of home networked appliances. The key feature of this scheme is utilization of a simple distributed architecture able to protect peers both within and outside the network in a flexible way. We believe this scheme can be developed into a more generalized security standard for protecting interacting networked appliances.**

I. INTRODUCTION

Peer-to-peer (P2P) has become one of the most widely discussed terms in networking technology in recent years. The term peer-to-peer refers to the concept that in a network of equals (peers) using appropriate information and communication systems, two or more individuals are able to spontaneously collaborate in the absence of central coordination [1].

In a P2P network, peers can join or leave the system without intervention from a centralized server, which facilitates seamless integration of new nodes (peers) into existing systems. Understandably, the decentralized nature of P2P networks facilitates scalability since there is no limit as to how many devices can connect with in established network. P2P systems, beginning with KaZaA [2], Napster [3], Gnutella [4], and several other related systems, have become immensely popular in the past few years, primarily because they offered a way for people to get music without paying for it [5]. For example, in the case of KaZaA, which is used mainly for music file sharing, users can search for a particular song and download it. Although P2P networks are interesting in their own right, in this paper we consider them as a means to facilitate the deployment of networked appliances within the home. The characteristics of P2P networks make them ideal for this task and to explain this we must further consider the concept of Networked Appliances as they relate to P2P networks.

When we consider the vision of how computing devices are likely to merge with their surroundings, it is clear that Networked Appliances provide a platform on which to build such a future. A Networked Appliance (NA) is defined as “a dedicated function consumer device with an embedded processor and a network connection” [6].

Security is an active research field within P2P Home Networked Appliances. Our concern is for the security for P2P NAs, providing a means to secure our network from a variety of security threats. Security is not just about keeping people out of your network. Security between interacting Networked Appliances is also an important aspect to be considered. For instance in situations when there are multiple NAs interacting simultaneously to provide a particular service. However, security must also provide access to the NA services in a way that minimises restrictions, allowing different network appliances to work together as long as security isn’t compromised. The tighter your security controls are, the greater the level of access that you can safely provide to trusted external networked appliances. Clearly security is an important issue. We have therefore proposed a novel scheme known as the Home Networked Appliances Security Scheme (HNASS). This scheme has been designed to secure all service requests besides taking measures to protect against attacks posing threats to the peers utilizing one of the provided services. One of such measures is a use of broadcast ID which is a part of message send by the peers to the scanner. This broadcast ID is generated by the peer for every time it broadcasts a message. An example of such threats could be a Denial of service. To explain further consider a scenario where an intruder tries to steel information belonging to a legitimate recognized peer. In this case broadcast ID will play a crucial role to prevent such attempts. With no doubt scanner would already be aware of the previous broadcast ID of the registered peer. Since it cannot be known to the intruder, chances are that any bogus broadcast ID will be recognized easily. Thus HNASS can easily prevent Denial of Service attacks. It was crucial to design a format which enables HNASS to provide security. It is cleared that ad-hoc network or peer-to-peer network does not support any centralized structure, however selected nodes could be assigned additional role as reported in some of the existing literature. In view of this explanation the role of scanner, analyzer and DTP are assigned to the selected nodes or distributed among the participating nodes.

The rest of this paper is organized as follows. In the existing literature there are various schemes that have been proposed for the implementation of NAs, but research shows there are weaknesses with these schemes. Several schemes are discussed in Section 2. In Section 3 HNASS is described and explained as a solution to the problem of P2P networked appliance security. A discussion about the evaluation of our scheme is provided in Section 4, with conclusions and future work covered in Section 5.

II. RELATED WORK

Research initiatives such as Universal Plug and Play (UPnP) [7], The Open Services Gateway Initiative (OSGi) [8] and Home Audio and Video Interoperability (HAVi) [9] can be used for integrating home NAs. In some cases the user would be required to configure their devices, whereas for other solutions devices are managed via a centralised provider. Services are usually discovered and composed using middleware protocols and interoperability issues are addressed using agreed standards in the above mentioned schemes [10]. We consider a number of these research initiatives in particular: OSGi, UPnP, ePerSpace and NASUF. A brief introduction to each of these standards follows.

Open Services Gateway Initiative (OSGi): OSGi is a well known middleware standard used to realise the digital home [11]. The standard was found in March 1999 [8] and was specifically designed for the delivery of a wide range of services to end users. OSGi deploys services over wide area networks to local networks and devices. This is achieved using a complete end-to-end solution architecture from the service provider, who actually operates the service through the local networks and devices that deliver it to the end user. This scenario is also potentially applicable to residential gateways, in vehicles and for mobile phone environments, among many others. Various services run on the OSGi framework. The framework is a service-oriented architecture, and responsible for the management of various services it contains. OSGi consists of a gateway between the Internet and the home network.

OSGi framework is controlled using centralized service providers. In addition, the configuration is also done in the same manner. Proprietary communication is a key factor in driving service discovery and composition. This complex structure clearly posed limitation to the distributed and computing service models. Moreover with very less doubt such structures could be more complex to handle as such devices and services becoming more heterogeneous in nature [12]. The security of OSGi is based on Java security rules and the Java programming model. The OSGi specification adopts an access control list to control the relationship between services. OSGi uses a static access control file which stores policies to manage admission control [13]. The access control list concept of OSGi is not fully suitable for P2P networking due to its lack of flexibility; for example it makes it difficult to allow anonymous but secure access to a service.

Universal Plug and Play (UPnP): UPnP is a technology framework which is somewhat simpler than OSGi because its sole purpose is to automatically interconnect, discover and control devices within the local home network. UPnP is also designed to work with many different types of networked devices and operating systems. Many home network routers offer UPnP support [7, 14].

UPnP does not provide mechanisms that allow devices to automatically discover and compose services. Similarly services cannot be provided without human intervention. Furthermore devices can only be used that match to the specification. This makes it difficult for user to access. It is somewhat limited and may segregate a large number of other networked appliances using different standards. Therefore the current version of UPnP, on its own, only provides controlled interoperability which is restrictive and leaves little room for improvement [12]. UPnP adopts a complex security structure which makes it difficult for users to access their NA services even once they have authenticated.

ePerSpace: ePerSpace is a project under the EU 6th Framework program for the development of personalized communication services within home networks [15]. The ePerSpace framework provides Global Network Integration and Interoperability which allows interconnecting audio and video to exchange its content between distributed services in a secure manner.

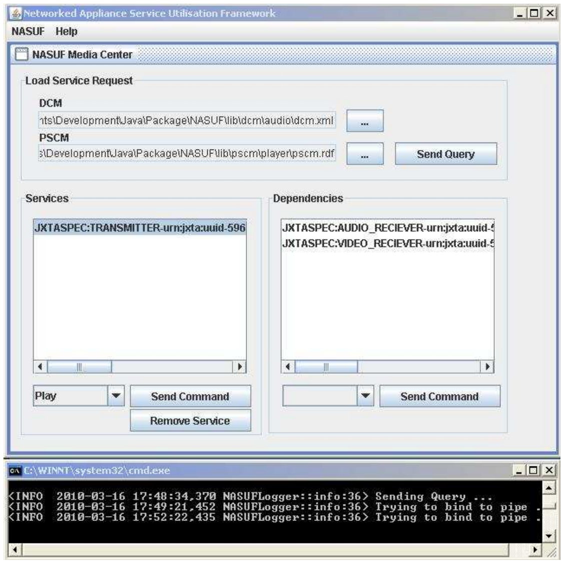
Primarily this standard is used to build a dynamic personalized network within a home network. This framework helps home and personal devices to build a personal environment that can be controlled using tools provided by Rich Media Object Management standard. This standard attempts to move one step further than the standards discussed above, by adding a level of intelligence that provides context adaptation mechanisms based on user profiles. It is difficult to implement this standard in pervasive ad-hoc environments as it is a choreographed solution. New devices, standards or services have to conform to the ePerSpace specifications in order to integrate within the environment [10]. In ePerSpace users are recognized and then accepted in the ePerSpace framework. Authentication in ePerSpace guarantees security and privacy in managing user private information. As users need authentication to join a network in ePerSpace, this standard does not fit well with our P2P NA environment, where peers must be able to join and leave the network easily and dynamically.

In the light of the above discussion this is clear that existing approaches do not provide mechanisms to detect conflicts and change configurations accordingly. P2P home networked appliances require a platform where devices are combined to produce value added function and to assist in zero configuration. This allows devices to adopt the environmental changes and to maintain composition. We have found the Networked Appliance Service Utilization Framework (NASUF) capable of meeting the above mentioned requirements.

Networked Appliance Service Utilization Framework (NASUF): In a NASUF service enabled network, appliances offer their services to other appliances when needed. These services are dynamically discovered and composed within a P2P network without the need for centralization [16]. In a P2P home network each device with its own services will have NASUF as well as application specific services that disperse the functions devices provide as independent services within the network.

In order to achieve device discovery automatically NASUF uses the JXTA (Juxtapose) architecture [17]. The JXTA protocols enable device to discover and communicate. In addition it also support mechanisms for interoperability in between with each other and provide mechanisms to perform interoperability between devices. NASUF provides much of the desired functionality of a P2P networked appliance system. However, an area that is left for future work is that of security. Trustworthiness is a particular issue in ad-hoc environments, and as pointed out by Fergus *et al.* "The middleware must ensure that the content received from a service is authenticated and that data streams are not intercepted and altered during transmission" [16]. By authenticating and encrypting data streams they claim that trust between network entities can be maintained. In ad-hoc environment trust become more suitable because of its de-centralized nature.

As an experiment audio, video and player and controller objects are implemented on different machines. It is important to mention implementation of these devices represents the secondary services which comprise NASUF. For NASUF it could act on its own or could be used remotely in a network. Controller device in NASUF is a device which is used to discover, control devices and maintain services. This could also be used to start, stop and invoke devices. It also allows individual service of a device to be stopped and started [16]. Figure 1 shows how the audio and video devices are discovered by the controller.



DCM
ıts\Development\Java\Package\NASUF\lib\dcm\audio\dcm.xml

PSCM
:\Development\Java\Package\NASUF\lib\pscm\player\pscm.rdf

JXTASPEC:TRANSMITTER-urn:jxta:uuid-596

JXTASPEC:AUDIO_RECIEVER-urn:jxta:uuid-5
JXTASPEC:VIDEO_RECIEVER-urn:jxta:uuid-5

Play

```
<INFO  2010-03-16 17:48:34,370 NASUFLogger::info:36> Sending Query ...
<INFO  2010-03-16 17:49:21,452 NASUFLogger::info:36> Trying to bind to pipe .
<INFO  2010-03-16 17:52:22,435 NASUFLogger::info:36> Trying to bind to pipe .
```

Fig. 1 NASUF Controller

Unfortunately NASUF does not provide valid security for P2P home networked appliances as mentioned above. This shows that there is a lack of security standard in the above mentioned schemes, in particular within NASUF, thus require an effective security mechanisms to secure P2P communication in NASUF.

In the following section, we present a novel scheme to provide security for NASUF.

III. Home Networked Appliances Security Scheme (HNASS)

The Home Networked Appliances Security Scheme (HNASS) takes an intermediate approach between the existing schemes and some of the new concepts which have been developed. In HNASS all peers go through a combination of security checks before being able to utilize available services. Scanner, analyzer and a Decision Taking Peer (DTP) work together to provide the necessary security.

HNASS specification contains definition of various functions. These functions assist the above mentioned components to perform their routine task.

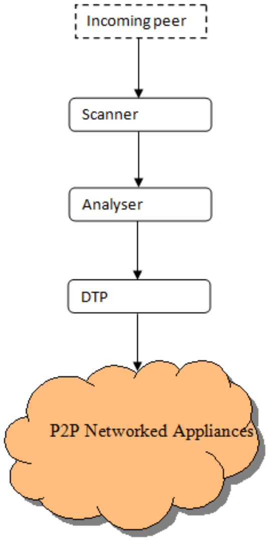


Fig. 2. Interactions between the HNASS security services.

In Figure 2 a scanner scans all incoming peers and views their unique IDs (UIDs), which are provided by JXTA. In addition, it collects relevant security properties of the peer, as well as details of connections with other services, forwarding the results on to the analyzer. The analyzer needs these three types of information from the scanner, and uses them to decide whether to allow or deny the incoming peer access. In fact, the analyzer cannot take a direct action, other than making decisions about incoming peers. The result is therefore sent to the DTP (Decision Taking Peer) which will either allow or deny a peer based on the analyzer's decision. If the scanner detects security issues with the incoming peer, it reports a connection error to the incoming peer and prevents it from connecting with services in the network. To understand better how this takes place, we will consider the required functions of these services in more detail.

A. Scanner

The scanner performs three functions. It collects the UID and security properties from a peer, as well as details about other peers it’s connected to. The scanner scans all incoming peers and then forwards these details to the analyzer if they can be verified; otherwise the incoming peer will be rejected from accessing the network.

A 1 Check UID Function

The Check User Identification Function (CUIDF) is called by the scanner to check the identification of an incoming peer. This function scans through the UID and broadcast ID to verify both the peer and the freshness of the received request.

A 1.1 Forward Check UID Function (FCUIDF)

This will be called by the scanner after the UID of the incoming peer is checked and verified by the CUID function. After UID verification the FCUIDF will pass the information to the next step.

A 1.2 Reverse Check UID Function (RCUIDF)

This will be called by the scanner if the UID of the incoming peer is checked but not verified by the CUID function. If the UID of a peer is not verified the RCUIDF will discard the peer and send it backs the sender using the previously gathered information.

A 2 Check Properties Function

The Check Properties Function (CPF) is called by the scanner to check relevant properties of an incoming peer. The CPF will analyse the same information as mentioned in the CUIF to check the properties. However the difference lies in the fact that it scans the received message against the list of known viruses.

A 2.1 Forward Check Properties Function (FCPF)

This is the last scanning function of the scanner. If all of the properties of the incoming peer have been verified by the CPF, then the FCPF will allow the system to move on to the next step: that of analyzing the data collected.

A 2.2 Reverse Check Properties Function (RCPF)

If there are problems with properties of the incoming peer, such as if the file is infected or the message doesn't contain request of an action. The RCPF will be called by the scanner in order to discard the peer and send it back to the sender using the previously gathered information.

B. Analyzer

To analyze an incoming peer, the analyzer needs three types of information from the scanner: the UID of the peer, its properties and the connections it has to other peers. On the basis of this information the analyzer will make a decision as to whether or not to allow the incoming peer access to the other services on the network. In fact the analyzer can only make a decision about the incoming peer – it's unable to act on it. Therefore a peer will be send to the Decision Taking Peer (DTP) for the decision made to be executed.

B1 Check Analyser Function (CAF)

To make a decision about an incoming peer, the CAF will be called by the analyzer. The CAF will analyze a peer and make a decision based on the analysis.

B 1.1 Forward Check Analyser Function (FCAF)

The FCAF is called by the analyzer once the name and properties of a peer have been verified by the CAF. The FCAF will pass the information of a peer to the DTP, where it will take the decision made by the analyzer.

B 1.2 Reverse Check Analyser Function (RCAF)

The RCAF is a function that will be called by the analyzer if information regarding a peer's name *i.e.* it's UID and its properties are not verified by the CAF. In this case the RCAF will send it back to the sender using the previously gathered information.

C. Decision Taking Peer (DTP)

The DTP receives all of the information about the incoming peer from the analyzer. As mentioned earlier the analyzer makes a decision about whether to allow or deny a peer, whereas the DTP takes that decision and will allow or deny the incoming peer based on the decision made by the analyzer.

C1 Allow Decision Taking Peer Function (ADTPF)

To take a final decision about the incoming peer and allow it to access a P2P NA network so that it can use or offer services, the ADTPF will be called by the DTP. This will be done only if a decision about the incoming peer has been made by the analyzer.

C2 Deny Decision Taking Peer Function (DDTPF)

If a peer is not verified and the analyzer makes a decision to discard it, the DDTPF will be called to deny the peer and send it back to the sender using the previously gathered information.

It is also important to mention that the scheme isn't limited to just NASUF, and with slight modifications can be extended for use in other networks i.e. general ad-hoc networks. For instance consider a situation where few mobile nodes establish an ad-hoc network in an emergency situation like earthquake or flood etc. In this case if an intruder tries to enter into the established network scanner could be modified to accept request only from the peers who registered at the time network formation. Therefore it is definite that any illegitimate attempt will be discarded.

IV. EVALUATION

We have performed an initial evaluation of our model in order to give us an idea about the efficiency of our proposed work, as well as to highlight possible areas for future work. We considered a number of scenarios intended to suit and reflect the working processes of HNASS. There are many situations that could be used to measure the performance of our scheme; however the scenarios described within this section were carefully selected to obtain the concrete observations.

We set up an experiment using three NAUF peers and a controller installed in our lab on three separate machines. The three peers included a video player, audio player and video/audio transmitter (player) peer. In addition, the HNASS scanner software was set up on a fourth machine in order to monitor the peers and their connections.

On invoking the video player peer using the NASUF controller, the player automatically connected to the audio and video output peers. The invoked peers and the consequent connections were detected by the HNASS scanner software as shown in Figure 3. This shows the peers labelled with their JXTA peer IDs, along with their connections. The scanner was able to build the ID and topology structure in real time based on the connections created between the peers.

Although this experiment does not currently include the complete property scanning, analysis or DTP capabilities from our design, it demonstrates the feasibility of measuring some of the essential peer and topology information required in order to perform a more detailed security analysis. As we can see from the experiments, we are able to successfully monitor peer IDs, and our ongoing work involves extending the scanner to determine other security properties. We have also developed an analysis peer which we intend to test using the output generated by the scanner as shown here.

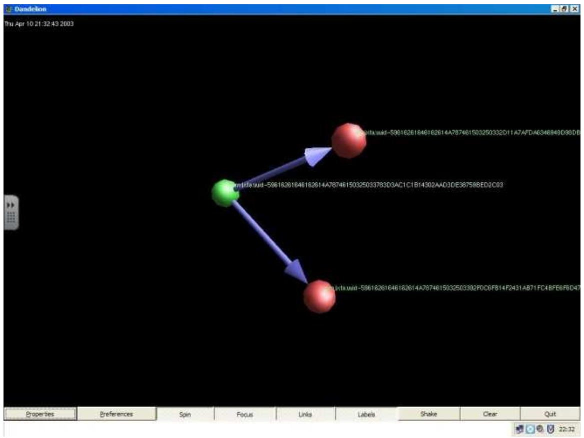

Dandelion
Thu Apr 10 21:32:43 2003
urn:jxta:uuid-59616261646162614A787461503250332D11A7AFDA6346949D98DB
urn:jxta:uuid-59616261646162614A78746150325033783D3AC1C1B14302AAD3DE38759BED2C03
urn:jxta:uuid-59616261646162614A78746150325033392F0C6F814F2431AB71FC4BFE6F6D47
Properties
Preferences
Spin
Focus
Links
Labels
Shake
Clear
Quit
22:32

Fig. 3 Visual view of Peers after scanning process

V. Conclusion and Future Work

In this paper we have presented a brief introduction and evaluation study of HNASS. HNASS utilizes a combination of three components to provide secure communication between the peers that we have described. Here we have successfully scanned audio, video and player devices on the basis of their UIDs and properties. We understand that due to the nature of this conference it was necessary to contribute our research findings since our scheme evolved around the concept of layer formation and combination. In future we will be conducting further research experiments to monitor our proposed scheme for performance in the analysis and decision of incoming peers on the basis of the information gathered by the scanner. We aim to share our research findings with the ongoing research in this area.

References

[1] D. Schoder, K. Fischbach, and C. Schmitt, "Core Concepts in Peer-to-Peer Networking," in *Peer-to-Peer Computing: The Evolution of a Distruptive Technology*: IGI Global, 2005, pp. 308.

[2] P. Sanderson, "Identifying an existing file via KaZaA artefacts," *Digital Investigation*, vol. 3, pp. 174-180, 2006.

[3] E. Palomar, J. M. Estevez-Tapiador, J. C. Hernandez-Castro, and A. Ribagorda, "Security in P2P networks: survey and research directions," Seoul, South Korea, 2006.

[4] Gnutella, "The Gnutella Protocol Specification v0.4," vol. 2009, 2009.

[5] D. S. Wallach, "A survey of peer-to-peer security issues," Berlin, Germany, 2003.

[6] S. Moyer, D. Marples, S. Tsang, and A. Ghosh, "Service portability of networked appliances," Piscataway, NJ, USA, 2001.

[7] Microsoft, "Understanding Universal Plug and Play," vol. 2008: Microsoft, 2004.

[8] D. Marples and P. Kriens, "The open services gateway initiative: An introductory overview," *IEEE Communications Magazine*, vol. 39, pp. 110-114, 2001.

[9] HAVi, "The HAVi Specification," vol. 2006: HAVi, 2004.

[10] A. Muhammad, M. Merabti, and B. Askwith, "An Ad hoc Gateway Service for Discovering and Composing Networked Appliances," presented at sixth annual postgraduate symposium on the convergence of telecommunications, networking and

broadcasting (PGNet 2005), Liverpool John Moores University,UK, 2005.

[11] O. Forum, "The OSGi Service Platform - Dynamic services for networked devices," vol. 2005: OSGi Forum, 2005.

[12] P. Fergus, "A Framework for Self-Adaptive Networked Appliances," in *School of Computing and Mathematical Sciences*, vol. PhD. Liverpool: Liverpool John Moores University, 2005, pp. 233.

[13] H. Chi-Chih, W. Pang-Chieh, and H. Ting-Wei, "Advanced OSGi security layer," Piscataway, NJ, USA, 2007.

[14] B. A. Miller, T. Nixon, C. Tai, and M. D. Wood, "Home networking with Universal Plug and Play," *IEEE Communications Magazine*, vol. 39, pp. 104-109, 2001.

[15] ePerSpace, "Towards the era of personal services at home and everywhere," vol. 2005: ePerSpace, 2005.

[16] P. Fergus, A.Taleb-Bandiab., A.Mingkhwan., M. Merabti, and M. Hanneghan, "A semantic Framework for self-adaptive networked appliances," presented at IEEE Consumer Communications and Networking Conference(CCNC'05), Las Vegas, Nevada, USA, 2005.

[17] S. Microsystems, "JXTA v2.3.x: Java Programmer's Guide," vol. 2008, 2.3 ed: Sun Microsystems, 2005.