ISSN:0975-9646

A Self-Configured Secure Protocol for the Management of Wireless Ad Hoc Networks

K.Srinivas[1], G.Balu NarasimhaRao[2], Dr.Sai Satyanarayana Reddy[3]

[1]M.Tech, CSE, LBRCE, Mylavaram.
[2]Assistant Professor, CSE, LBRCE, Mylavaram, India.
[3] Professor, CSE, LBRCE, Mylavaram, India.

Abstract— My paper presents a self-configured secure protocol for wireless ad hoc networks which uses an hybrid symmetric/asymmetric scheme. My proposal is a complete self-configured secure protocol that is able to create the network within the region and share secure services. The protocol consists of several functions to create network and to share the among the users in the network. The management of network will be done in a secure way that provides the trust between users. In these days Ad hoc networks plays main role in the sharing of data with limited resources. The network will be created spontaneously for the completion of particular task that includes data sharing in a secure environment. The protocol will provide the secure environment for the users in the network. After completion of the task, all users will be disconnected from the network.

1 Introduction

In these days Ad hoc networks plays main role in the sharing of data with limited resources. The network will be created spontaneously for the completion of particular task that includes data sharing in a secure environment. The exponential growth in the development and acceptance of mobile communications in recent years is especially observed in the fields of wireless local area networks, mobile systems, and ubiquitous computing. This growth is mainly due to the mobility offered to users, providing access to information anywhere, user friendliness, and easy deployment. Furthermore, the scalability and flexibility of mobile communications increase users' productivity and efficiency. Spontaneous ad hoc networks are formed by a set of mobile terminals placed in a close location that communicate with each other, sharing resources, services or computing time during a limited period of time and in a limited space, following human interaction pattern. People are attached to a group of people for a while, and then leave. Network management should be transparent to the user. A spontaneous network is a special case of ad hoc networks. They usually have little or no dependence on a centralized administration. Spontaneous networks can be wired or wireless. We consider only wireless spontaneous networks in this paper. Their objective is the integration of services and devices in the same environment, enabling the user to have instant service without any external infrastructure. Because these networks are implemented in devices such as laptops, PDAs or mobile phones, with limited capacities, they must use a lightweight protocol, and new methods to control, manage, and integrate them. Configuration services in spontaneous networks depend significantly on network size, the nature of the participating nodes and running applications. Spontaneous networks imitate human relations while having adaptability to new conditions and fault tolerance (the failure of a device or service should not damage the functionality). Methods based on imitating the behaviour of human relations facilitate secure integration of services in spontaneous networks. Furthermore, cooperation among the nodes and quality of service for all shared network services should be provided. Spontaneous ad hoc networks require well defined, efficient and user-friendly security mechanisms. Tasks to be performed include: user identification, their authorization, address assignment, name service, operation, and safety.

2 Secure Network

2.1 Network Overview

My protocol allows the creation and management of distributed and decentralized spontaneous networks with little intervention from the user, and the integration of different devices (PDAs, cell phones, laptops, etc.). Cooperation between devices allows provision and access to different services, such as group communication, collaboration in program delivery, security, etc. The network members and services may vary because devices are free to join or leave the network. Spontaneous network should complete the following steps in order to be created. The network will be secured by using this protocol functions.

2.2 Nodes in the Network

The first node creates the network and generates a random session key, which will be exchanged with new nodes after the authentication phase. The phases of a node joining the network: node authentication and authorization, agreement on session key, transmission protocol and speed, and IP address and routing. When node B wants to join an existing network, it must choose a node within communication range to authenticate with (e.g., node A). A will send its public key. Then, B will send its Identity Card signed by A's public key. Next, A validates the received data and verifies the hash of the message in order to check that the data has not been modified. In this step, A establishes the trust level of B by looking physically at B (they are physically close), depending on whether A knows B or not. Finally, A will send its Identity Card data to B (it may do so even if it decides not to trust B). This data will be signed by B's public key (which has been received on B's Identity Card). B will validate A's Identity Card and will establish the trust and validity in A only by integrity verification and authentication. If A does not reply to the joining request, B must select another network node (if one exists). After the authentication, B can access data, services, and other nodes certificates by a route involving other nodes in network.

2.3 Providing Services

A user can ask other devices in order to know the available services. It has an agreement to allow access to its services and to access the services offered by other nodes. Services have a large number of parameters which are not transparent to the user and require manual configuration. One issue is to manage the automatic integration tasks and use, for example, service agents. Other is to manage secure access to the services offered by the nodes in the network. The fault tolerance of the network is based on the routing protocol used to send information between users. Services provided by B are available only if there is a path to B, and disappear when B leaves the network.

2.4 Trust between Users

There are only two trust levels in the system. Node A either trusts or does not trust another node B. The software application installed in the device asks B to trust A when it receives the validated IDC from B. Trust relationship can be asymmetric. If node A did not establish trust level with node B directly, it can be established through trusted chains, e.g., if A trusts C and C trusts B, then A may trust B. Trust level can change over time depending on the node's behaviour. Thus, node A may decide not to trust node B although A still trusts C and C trusts B anymore. It can also stop trusting if it discovers that chain not exist anymore.

3 Network Management

In the network formation, nodes perform an initial exchange of configuration information and security using the mechanism of authentication. This mechanism avoids the need for a central server, making the tasks of building the network and adding new members very easy. The network is created using the information provided by users, thus, each node is identified by an IP address. Services are shared using TCP connections. The network is built using IEEE 802.11b/g technology which has high data rates to share resources. We have reserved the short-range technology (Bluetooth) to allow authentication of nodes when they join the network. After the authentication process, each node learns the identity card of other known nodes, a public key and a LID. This information will be updated and completed throughout the network nodes. This structure provides an authenticated service that verifies the integrity of the data from each node because there is a distributed CA. Each node requests the services from all the nodes that it trusts, or from all known nodes in the network, depending on the type of service. A request to multiple nodes is made through diffusion processes. The protocol prioritizes access to information through trusted nodes. When the information cannot be obtained through these nodes, it can then ask other nodes.

3.1 Network Creation

The first node in the network will be responsible for setting the global settings of the spontaneous network. However, each node must configure its own data (including the first node): IP, port, data security, and user data. This information will allow the node to become part of the network. After this data are set in the first node, it changes to standby mode.

3.2 Protocol Functionalities

Once the validation/registration process of the user in the device has been done, he/she must determine whether to create a new network or participate in an existing one. If he/she decides to create a new network. First, a session key will be generated. Then, the node will start its services (including the network and authentication services). Finally, it will wait for requests from other devices that want to join the network. If the user wants to become part of an existing network to find a device that will give trust to it, save corresponding data and will able to begin communications. The node that belongs to the network, and is responsible for validating the new node's data, will perform a diffusion process to the nodes that are within its communication range. These nodes will forward the received packets to their neighbours until the data reach all nodes in the network.

3.3 Implementation of Protocol

In this there are 16 packets for the running of the protocol. When a device wants to join a spontaneous network it has to start the process by sending a Discovery request packet (01), which contains the Logical Identity of the user in order to let the destinations know the sender device. The receivers will reply with the Discovery reply packet (02) with their Logical Identity, their IP address, and network mask. This information is then used to learn the selected device to authenticate and to propose an IP inside that network IP range. The authentication request packet (03) is used for the new device authentication. The authentication reply packet (04) confirms that the proposed IP and the email are unique in the network, so the new device is officially authenticated. In case of duplication, an error packet is sent. The IP and e-mail checking packet (05) is used by the authenticator device to verify that no one in the network has the same email or IP address as the one proposed by the new device. The IP and e-mail checking reply packet (06) is sent to the authenticator device in order to verify that the IP and e-mail are unique. If the IP is duplicated, the device must restart the authentication process after the generation of a new IP. The update request to one node (07) is used to request information to a specific known node and the update reply from one node (08) is used to reply with the information requested by the update request packet to one node. Unknown information can be requested from all nodes in the network by sending the update request to all network nodes packet by flooding (09).

The reply with the information requested is called update reply to all network nodes packet (10). The Certificate request to trusted nodes (11) and the Certificate request to known nodes (12) are used to request the certificate from all trusted and all known nodes, respectively. Both packets are replied to by the certificate reply packet (13). Data are sent using the Packet for sending data (14). This packet is sent when the user decides to communicate with one or more nodes. These data could be sent in plain or encrypted text. The error packet (15) can be sent to indicate that this operation is not possible, because the authentication has failed, or because the node does not have the required data. The acknowledge packet (16) is used to confirm to the sender that the packet has arrived at its destination correctly.

3.4 Session Key Revocation

The user certificate has an expiration time. After this time, the user must authenticate with the device. Otherwise, the device is blocked. Session key has an expiration time, so it is revoked periodically. A node that leaves the spontaneous network will keep the session key until it expires. It will let the user return to the network if it has joined previously (the spontaneous network is usually set up for a limited period of time, which is usually not very long). However, if a node is disconnected from the network during the period of time when the session key has been renewed, it will not be able to access the network until it is authenticated again with someone from the network.

When the system detects that a node is compromised, trust is removed and the session key is regenerated following the aforementioned procedure. If a neighbour is compromised, a new session key is sent using asymmetric encryption. Moreover, authenticity can be guaranteed if the node that generated the new session key signs it. To process the received request, the node checks if it can reply to the request, if not, the node sends the search to other nodes (that it trusts or known nodes). Then, the node has to validate the certificate and sends it to the requesting node. When the server process receives the packet, it processes the packet in order to take the certificate and checks its validity access to the certificate data.

To send data encrypted with the public key to a node, the user has to select the remote node and write the data. Then, the message is encrypted using the remote node's public key. By this only the user can receive the response from the server and the task will completed.

4 Performance Analysis

The protocol has been developed using Java programming, whose mobility, interoperability, and multiplatform features are very useful to deploy the protocol. Given that the protocol may work on devices with limited resources, we have used a Java variant called Java 2 Platform, Micro Edition (J2ME). It also has a small and fast virtual machine (KVM) that allows us to run the software without overloading the device. This platform is especially designed for personal use and embedded devices. The configuration package selected was the connected limited device configuration (CLDC). It is a set of classes for a family of mobile devices, defining the type and amount of memory or processor type. Devices running CLDC must have a minimum of 160 KB memory to store the Java technology stack. It can run when there are computing and process limitations, and for low-power devices. It allows the implementation of communication protocols over both Wi-Fi and Bluetooth technologies. We used Mobile Information Devi Profile (MIDP), suitable for portable devices with limited screen, required for persistent storage, and network communications capabilities. We used a third party manufacturer to introduce security in the protocol. We selected Crypto, a Bouncy Castle Lightweight API [38] solution, since it provides a lightweight cryptographic open source API, which can be used in any environment.

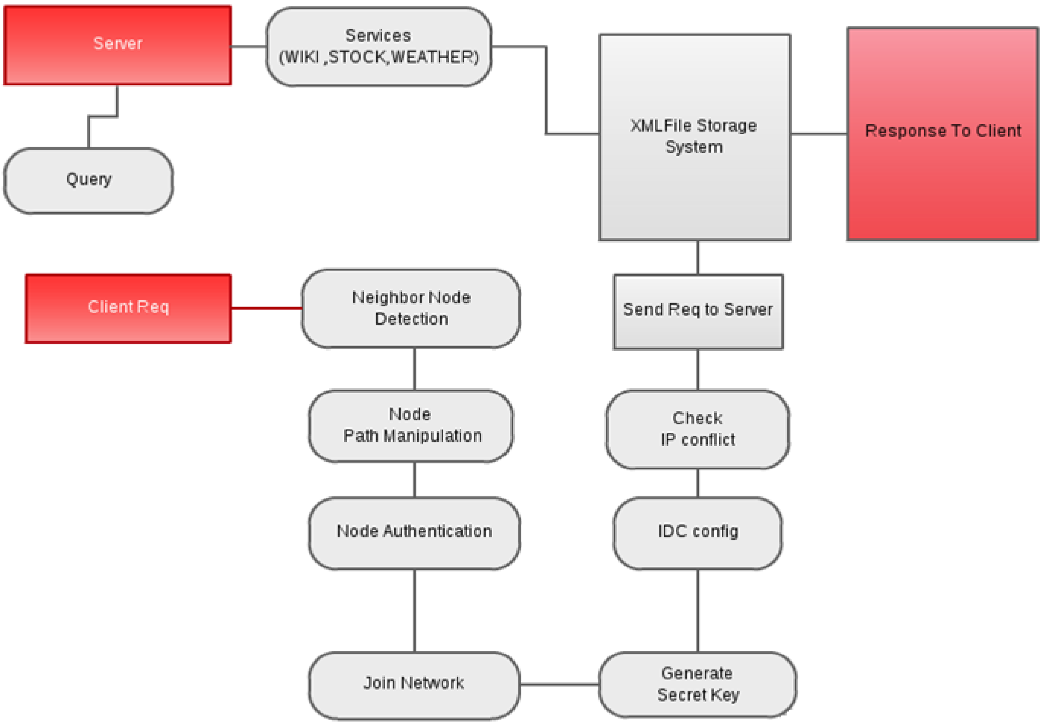

Fig 1. System design process

5 Comparing Network Features

Some of them have not included any security system, but others have included some systems that exist by default in the used technology, such as wired Equivalent Privacy (WEP), IPSec, and Diffie-Hellman. But no one propose a complete security protocol, which is the main purpose of this paper. Moreover, the security explanations in these papers are few and they do not tackle security issues in detail. They only describe how new nodes join the network securely; neither explain how the information is secured nor show its security performance. In this comparison, we have included the main features of a spontaneous network: need for user intervention, self configuration, and security. We have also included network purpose (what is it created for), the programming language used to create it and if there is a real prototype in existence. Other protocols only include a security mechanism for accessing new nodes is only included, not a complete security protocol to perform secure communications in spontaneous networks. The protocol that includes several functionalities used create a secure environment for the users in order to share the data among them. It can gives trust between users.

6 Conclusion

The creation and management of a spontaneous wireless ad hoc network is described here in this paper. It is based on a social network imitating the behaviour of human relationships. Thus, each user will work to maintain the network, improve the services offered, and provide information to other network users. The protocol provided some procedures for self-configuration: a unique IP address is assigned to each device, the DNS can be managed efficiently and the services can be discovered automatically. It also created a user-friendly application that has minimal interaction with the user. A user without advanced technical knowledge can set up and participate in a spontaneous network. The security schemes included in the protocol allow secure communication between end users (bearing in mind the resource, processing, and energy limitations of ad hoc devices). The response times obtained are suitable for use in real environments, even when devices have limited resources. Storage and volatile memory needs are quite low and the protocol can be used in regular resource-constrained devices (cell phones, PDAs...).

I intend to add some new features to the user application (such as sharing other types of resources, etc.) and to the protocol, such as an intrusion detection mechanism and a distributed Domain Name Service by using the LID and IP of the nodes. Now, we are working on adding other types of nodes that are able to share their services in the spontaneous network. The new nodes will not depend on a user, but on an entity such as a shop, a restaurant, or other types of services.

References

[1] L.M. Feeney, B. Ahlgren, and A. Westerlund, “Spontaneous Networking: An Application-Oriented Approach to Ad-hoc Networking,” IEEE Comm. Magazine, vol. 39, no. 6, pp. 176-181, June 2001.

[2] J. Lloret, L. Shu, R. Lacuesta, and M. Chen, “User-Oriented and Service-Oriented Spontaneous Ad Hoc and Sensor Wireless Networks,” Ad Hoc and Sensor Wireless Networks, vol. 14, nos. 1/ 2, pp. 1-8, 2012.

[3] S. Preuß and C.H. Cap, “Overview of Spontaneous Networking - Evolving Concepts and Technologies,” Rostocker Informatik- Berichte, vol. 24, pp. 113-123, 2000.

[4] R. Lacuesta, J. Lloret, M. Garcia, and L. Pen˜ alver, “A Spontaneous Ad-Hoc Network to Share WWW Access,” EURASIP J. Wireless Comm. and Networking, vol. 2010, article 18, 2010.

[5] Y. Xiao, V.K. Rayi, B. Sun, X. Du, F. Hu, and M. Galloway, “A Survey of Key Management Schemes in Wireless Sensor Networks,” Computer Comm., vol. 30, nos. 11/12, pp. 2314-2341, Sept. 2007.

[6] V. Kumar and M.L. Das, “Securing Wireless Sensor Networks with Public Key Techniques,” Ad Hoc and Sensor Wireless Networks, vol. 5, nos. 3/4, pp. 189-201, 2008.

[7] S. Zhu, S. Xu, S. Setia, and S. Jajodia, “LHAP: A Lightweight Hopby-Hop Authentication Protocol For Ad-Hoc Networks,” Ad Hoc Networks J., vol. 4, no. 5, pp. 567-585, Sept. 2006.

[8] A. Noack and S. Spitz, “Dynamic Threshold Cryptosystem without Group Manager,” Network Protocols and Algorithms, vol. 1, no. 1, Oct. 2009.

[9] J. Yan, J. Ma, F. Li, and S.J. Moon, “Key Pre-distribution Scheme with Node Revocation for Wireless Sensor Networks,” Ad Hoc and Sensor Wireless Networks, vol. 10, nos. 2/3, pp. 235-251, 2010.

[10] M. Mukesh and K.R. Rishi, “Security Aspects in Mobile Ad Hoc Network (MANETs): Technical Review,” Int’l J. Computer Applications, vol. 12, no. 2, pp. 37-43, Dec. 2010.

[11] K. Sahadevaiah and P.V.G.D. Prasad Reddy, “Impact of Security Attacks on a New Security Protocol for Mobile Ad Hoc Networks,” Network Protocols and Algorithms, vol 3, no. 4, pp. 122-140, 2011.

[12] L. Herrero and R. Lacuesta, “A Security Architecture Proposal for Spontaneous Networks,” Proc. Int’l Conf. Advances in the Internet Processing System and Interdisciplinary Research, Oct. 2003.

[13] R. Lacuesta and L. Pen˜ aver, “IP Addresses Configuration in Spontaneous Networks,” Proc. Ninth WSEAS Int’l Conf. Computers (ICCOMP ’05), July 2005.

[14] R. Lacuesta and L. Pen˜ alver, “Automatic Configuration of Ad-Hoc Networks: Establishing Unique IP Link-Local Addresses,” Proc. Int’l Conf. Emerging Security Information, Systems and Technologies (SECURWARE ’07), 2007.

[15] J. Latvakoski, D. Pakkala, and P. Paakkonen, “A Communication Architecture for Spontaneous Systems,” IEEE Wireless Comm., vol. 11, no. 3, pp. 36-42, June 2004.

[16] L. Liu, J. Xu, N. Antonopoulos, J. Li, and K. Wu, “Adaptive Service Discovery on Service-Oriented and Spontaneous Sensor Systems,” Ad Hoc and Sensor Wireless Networks, vol. 14, nos. 1/2, pp. 107-132, 2012.

[17] S. Gallo, L. Galluccio, G. Morabito, and S. Palazzo, “Rapid and Energy Efficient Neighbour Discovery for Spontaneous Networks,” Proc. Seventh ACM Int’l Symp. Modelling, Analysis and Simulation of Wireless and Mobile Systems, Oct. 2004.

[18] J. Ba¨ckstro¨m and S. Nadjm-Tehrani, “Design of a Contact Service in a Jini-Based Spontaneous Network,” Proc. Int’l Conf. and Exhibits on the Convergence of IT and Comm., Aug. 2001.

[19] V. Untz, M. Heusse, F. Rousseau, and A. Duda, “Lilith: an Interconnection Architecture Based on Label Switching for Spontaneous Edge Networks,” Proc. First Ann. Int’l Conf. Mobile and Ubiquitous Systems: Networking and Services (Mobiquitous ’04), Aug. 2004.

[20] L.M. Feeney, B. Ahlgren, A. Westerlund, and A. Dunkels, “Spontnet: Experiences in Configuring and Securing Small Ad Hoc Networks,” Proc. Fifth Int’l Workshop Network Appliances, Oct. 2002.

[21] M. Danzeisen, T. Braun, S. Winiker, D. Rodellar, “Implementation of a Cellular Framework for Spontaneous Network Establishment,” Proc. IEEE Wireless Comm. and Networking Conf. (WCNC ’05), Mar. 2005.

[22] J. Rekimoto, “SyncTap: Synchronous User Operation for Spontaneous Network Connection,” Personal and Ubiquitous Computing, vol. 8, no. 2, pp. 126-134, May 2004.

[23] C. Cornelius, A. Kapadia, D. Kotz, D. Peebles, M. Shin, and N. Triandopoulos, “Anonysense: Privacy-Aware People-Centric Sensing,” Proc. Sixth Int’l Conf. Mobile Systems, Applications, and Services (MobiSys ’08), pp. 17-20, June 2008.

[24] R. Lacuesta, J. Lloret, M. Garcia, and L. Pen˜ alver, “Two Secure and Energy-Saving Spontaneous Ad-Hoc Protocol for Wireless Mesh Client Networks,” J. Network and Computer Applications, vol. 34, no. 2, pp. 492-505, Mar. 2011