Creating Spontaneous Wireless Ad Hoc Networks by using Secure Protocol

Shaik Rafi[1], V. PremaLatha Williams[2]

[1]M.Tech (CS), Nimra College of Engineering and Technology, A.P., India.

[2]Head of the Department, Dept. of Computer Science & Engineering, Nimra College of Engineering and Technology, A.P., India.

Abstract—Our paper presents a secure protocol for spontaneous wireless ad hoc networks which uses an hybrid symmetric/ asymmetric scheme and the trust between users in order to exchange the initial data and to exchange the secret keys that will be used to encrypt the data. Trust is based on the first visual contact between users. Our proposal is a complete self-configured secure protocol that is able to create the network and share secure services without any infrastructure. The network allows sharing resources and offering new services among users in a secure environment. The protocol includes all functions needed to operate without any external support. We have designed and developed it in devices with limited resources. Network creation stages are detailed and the communication, protocol messages, and network management are explained. Our proposal has been implemented in order to test the protocol procedure and performance. Finally, we compare the protocol with other spontaneous ad hoc network protocols in order to highlight its features and we provide a security analysis of the system.

Keywords — Distributed protocol, secure protocol, spontaneous network, wireless ad hoc networks

I. Introduction

In recent years there has been an exponential growth in the development and acceptance of mobile communications and it 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 [1], [2]. 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 behavior of human relations facilitate secure integration of services in spontaneous networks [3]. Furthermore, cooperation among the nodes and quality of service for all shared network services should be provided [4]. 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. Generally, wireless networks with infrastructure use Certificate Authority (CA) servers to manage node authentication and trust [5], [6]. Although these systems have been used in wireless ad hoc and sensor networks [7], they are not practical because a CA node has to be online (or is an external node) all the time. Moreover, CA node must have higher computing capacity.

Security should be based on the required confidentiality, node cooperation, anonymity, and privacy. Exchanging photos between friends requires less security than exchanging confidential documents between enterprise managers. Moreover, all nodes may not be able to execute routing and/or security protocols. Energy constraints, node variability, error rate, and bandwidth limitations mandate the design and use of adaptive routing and security mechanisms, for any type of devices and scenarios.

Dynamic networks with flexible memberships, group signatures, and distributed signatures are difficult to manage. To achieve a reliable communication and node authorization in mobile ad hoc networks, key exchange mechanisms for node authorization and user authentication are needed.

The related literature shows several security methods such as predistribution key algorithms, symmetric and asymmetric algorithms, intermediate node-based methods and hybrid methods. But these methods are not enough for spontaneous networks because they need an initial configuration (i.e., network configuration) or external authorities (for example, central certification authorities).

None of the existing papers propose a secure spontaneous network protocol based on user trust that provides node authenticity, integrity checking, and privacy. The network and protocol proposed in this paper can establish a secure self-configured environment for data distribution and resources and services sharing among users. Security is established based on the service required by the users, by building a trust network to obtain a distributed certification authority. A user is able to join the network because he/she knows someone that belongs to it. Thus, the certification authority is distributed between the users that trust the new user. The network management is also distributed, which allows the network to have a distributed name service. We apply asymmetric cryptography, where each device has a public-private key pair for device identification and symmetric cryptography to exchange session keys between nodes. There are no anonymous users, because confidentiality and validity are based on user identification.

Preliminary versions of this paper appeared in [8], [9], [10]. In 2003, we presented the basis to setup a secure spontaneous network [8]. To solve mentioned security issues, we used an authentication phase and a trust phase [10]. Moreover, we presented a mechanism to allow nodes to check the authenticity of their IP addresses while not generating duplicated IP addresses. The mechanism helps nodes to authenticate by using their IP addresses. We have used this mechanism in the secure protocol presented in this paper, but it can be replaced by any other IP address assignment mechanism.

II. Related Work

In [11], Latvakoski et al. explain a communication architecture concept for spontaneous systems, integrating application-level spontaneous group communication, and ad hoc networking together. A set of methods to enable plug and play, addressing and mobility, peer to peer connectivity and the use of services are also provided.

Liu et al. [12] show how networked nodes can autonomously support and cooperate with each other in a peer-to-peer (P2P) manner to quickly discover and self-configure any services available on the disaster area and deliver a real-time capability by self-organizing themselves in spontaneous groups to provide higher flexibility and adaptability for disaster monitoring and relief.

Gallo et al. [13] pursued two targets in spontaneous networks: to maximize responsiveness given some constraints on the energy cost and to minimize the energy cost given certain requirements on the responsiveness.

Backstrom and Nadjm-Tehrani [14] developed the first real spontaneous network that offers services dynamically using the Jini technology. They explain the architectural design of the contact service and its implementation. The prototype demonstrates how major design criteria, flexibility, dependability, efficiency, and transparency, affect the design and services of a dynamic network of devices.

In [15], Untz et al. propose a lightweight and efficient interconnection protocol suitable for spontaneous edge networks. They design and implement Lilith, a prototype of an interconnection node for spontaneous edge networks. It uses MPLS and allows different communication paths on a per flow basis, provides seamless switching between operational and back-up paths, and makes available information on destination reachability.

Feeney et al. [16] presented Spontnet, a prototype implementation of a simple ad hoc network configuration utility based on the main ideas of spontaneous networks. Spontnet allows users (using face-to-face authentication and short-range link with easily identifiable endpoints) to distribute a group session key without previous shared context and to establish shared namespace. Two applications, a simple web server and a shared whiteboard, are provided as examples of collaborative applications. They use IPSec protocol (used for Virtual Private Networks), applied though internet. Spotnet therefore uses both wired and wireless links and corresponding protocols.

Danzeisen et al. [17] apply WEP, the regular security mechanism used in Wireless LANs, available by default in the IEEE 802.11 wireless protocol. Other proposals that did not discuss security aspects could also apply this default solution. Although it was available to us, we did not use it because WEP is vulnerable to hacking attacks, and better solutions, e.g., WPA, WPA2 should be considered instead.

Rekimoto introduced the concept of synchronous user operation in [18], and described a user interface SyncTap technique for spontaneously establishing network connections between digital devices. This method can deal with multiple overlapping connection requests by detecting “collision” situations, and can also ensure secure network communication by exchanging public key information upon establishing a connection. Shared session key for secure communication is created by piggybacking Diffie-Hellman public keys (generated by each device) on multicast packets. These public keys are used to calculate a shared secret session key for encrypted communication. In this case, the authors do not propose any secure protocol. They have just added an existing security mechanism in their authentication phase. It is similar to the one used by us when a new node joins our network, but we have added other security mechanisms in order to create a complete secure protocol for spontaneous networks.

This paper does not tackle routing issues in spontaneous ad hoc wireless networks. A paper that presents a security protocol for routing purposes, based on trust. It presents two secure and energy-saving spontaneous ad hoc protocols for wireless mesh client networks where two different security levels (weak and strong) are taken into account in the path when information is transmitted between users.

III. Proposed Work

A. *Network Overview*

Our 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 [1].

Step1: Joining Procedure

This step enables devices to communicate, including the automatic configuration of logical and physical parameters. The system is based on the use of an IDentity Card (IDC) and a certificate. The IDC contains public and private components. The public component contains a Logical Identity (LID), which is unique for each user and allows nodes to identify it. It may include information such as name, photograph or other type of user identification. This idea has been used in other systems such as in vehicular ad hoc networks. It also contains the user's public key (K_i), the creation and expiration dates, an IP proposed by the user, and the user signature. The user signature is generated using the Secure Hash Algorithm (SHA-1) on the previous data to obtain the data summary. Then, the data summary is signed with the user's private key. The private component contains the private key (k_i). The user introduces its personal data (LID) the first time he/she uses the system because the security information is generated then. Security data are stored persistently in the device for future use.

Certificate C_{ij} of the user i consists of a validated IDC, signed by a user j that gives its validity. To obtain IDC signature of user i, the summary function obtained by SHA-1 is signed with j's private key. No central certification authority is used to validate IDC. Validation of integrity and authentication is done automatically in each node. The certification authority for a node could be any of the trusted nodes. This system enables us to build a distributed certification authority between trusted nodes. When node A wants to communicate with another node B and it does not have the certificate for B, it requests it from its trusted nodes. After obtaining this certificate the system will validate the data; if correct then it will sign this node as a valid node. All nodes can be both clients and servers, can request or serve requests for information or authentication from other nodes.

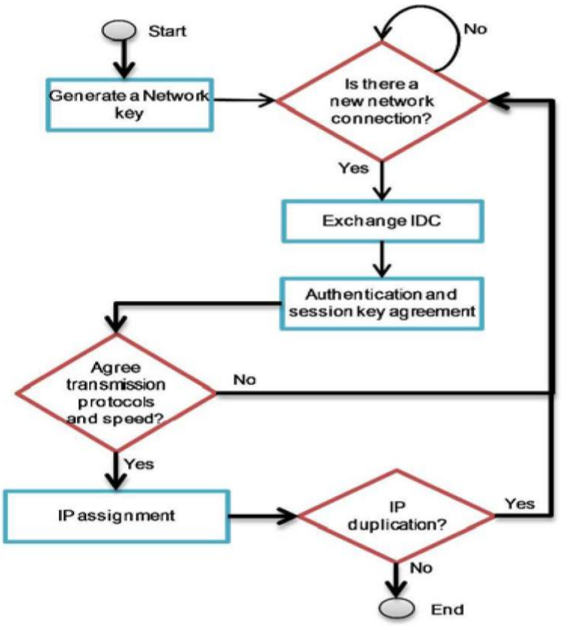
Start
Generate a Network key
Is there a new network connection?
No
Yes
Exchange IDC
Authentication and session key agreement
Agree transmission protocols and speed?
No
Yes
IP assignment
IP duplication?
Yes
No
End

Figure 1 Flowchart for joining a new node

The first node creates the spontaneous network and generates a random session key, which will be exchanged with new nodes after the authentication phase. Fig. 1 shows 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 IDC 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 IDC 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 IDC). B will validate A's IDC 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.

Security management in the network is based on the Public Key Infrastructure and the symmetric key encryption scheme. Symmetric key is used as a session key to cipher the confidential messages between trust nodes. It has less energy requirements [27], [28], [29] than the asymmetric key. We have used the Advanced Encryption Standard (AES) algorithm for the symmetric encryption scheme [30]. It offers high security because its design structure removes sub-key symmetry. Moreover, execution times and energy consumption in cryptography processes are adequate for low-power devices. The asymmetric key encryption scheme is used for distribution of the session key and for the user authentication process. We used two types of asymmetric encryption schemes: Elliptic Curve Cryptosystem (ECC), because of its high performance, and the Rivest, Shamir & Adleman cryptographic algorithm (RSA). After the mutual authentication, A will encrypt the session key with B's public key and will send it to B. Then, they will agree the transmission protocols and the wireless connection speed.

Finally, B will configure IP address and routing information. Secure routing protocol is borrowed from. B generates an IP address which has a fixed part in the first two bytes and the rest is formed by a random number which depends on the user's data. Then, B will send the data to process the routing information to A. A will check whether the IP is duplicated in the network. When B sends data to other network nodes, e.g., node C, these data will be validated by C (using hashing and authentication methods). Afterwards, C will establish the trust level with B, by looking physically. If no trust level is established, it will be done afterwards by using trusted chains.

Step 2: Services Discovery

B asks for the available services. Services can be discovered using Web Services Description Language (WSDL). Our model is based on [12], but in our spontaneous network we don't use a central server. Moreover, other service discovery services can be implemented in our system. 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.

Step 3: Establishing Trusted Chain and Changing Trust Level

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 behavior. Thus, node A may decide not to trust node B although A still trusts C and C trusts B. It can also stop trusting if it discovers that previous trust chain does not exist anymore.

B. Protocol And Network Management

In the network formation, nodes perform an initial exchange of configuration information and security using the mechanism of authentication or greeting based on the works shown in [35], [36]. 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.

IV. CONCLUSION

In this paper, we show the design of a protocol that allows the creation and management of a spontaneous wireless ad hoc network. It is based on a social network imitating the behavior of human relationships. Thus, each user will work to maintain the network, improve the services offered, and provide information to other network users. We have 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. We have 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).

We have performed several tests to validate the protocol operation. They showed us the benefits of using this self-configuring ad hoc spontaneous network. 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...).

We 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. WirelessComm. 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] 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.

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

[10] 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.

[11] 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.

[12] 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.

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

[14] 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.

[15] 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.

[16] 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.

[17] 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.

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