ADAPTIVE USER PROFILING-BASED CONTEXT-AWARE TRUST MODEL FOR DEPENDENT PEOPLE

Anas El Husseini, Abdallah M'Hamed, Bachar El Hassan

ADAPTIVE USER PROFILING-BASED CONTEXT-AWARE TRUST MODEL FOR DEPENDENT PEOPLE

Anas EL HUSSEINI[1,2], Abdallah M'HAMED[1], Bachar EL HASSAN[2]
[1]TelecomSudParis, Handicom Lab., Evry, France
[2]Lebanese University, LASTRE Lab, Tripoli, Lebanon

In smart environments, pervasive computing contributes significantly in improving daily life activities for users by providing personalized services to dependent people. Nevertheless, those environments do not guarantee a satisfactory level for protecting the user privacy and ensuring the trust between communicating entities [1].

Context-based security is an emerging approach to cope with the new security problems introduced by the high level of dynamicity and heterogeneity that characterize pervasive environments. Although the fact that pervasive environments rise the security challenges, they also bring new opportunities due to the ubiquitous technologies and ambient intelligence [2] which provide valuable contextual information about the user and its environment: user's identity, interaction history with services, location, preferences, type of requested services, time/date of service request.

1- Aims

We propose a trust evaluation model based on user past and present behavior. This model is associated to a lightweight authentication key agreement protocol (EC-SAKA). The aim is to enable the communicating entities to establish a trust level needed to succeed in a mutual authentication using a scheme suitable for low-resource devices in smart environments. Finally, we tested and implemented our scheme on Android mobile phones.

The most important challenge is to design a context-aware trust model by taking into consideration both the user profiles and context attributes [3]. For making these living/working spaces secure and trustworthy for dependent people we must put the user as a central point by considering [4]:

- the specific requirements of these people in terms of protection of their life and regardless to their capabilities and preferences,
- the strong vulnerability of some spaces like medical care environments where both reliability and security can affect timeliness and accuracy information for patient monitoring,
- the wide range of involved actors : caregivers, medical staff, nurses, emergency personnel, among which the shared information should be secure, private and anonymous,
- the variety of living spaces both private (residences) and public (hospitals, workplace, etc.), where sharing information/services between users should be controlled.

The second challenge is to protect trust models against the malicious threats that endanger the trust evaluation process by manipulating and affecting the trust values. There are other aspects in trust models that are still open challenges, like situations of blind trust model and scenarios of zero trust where the user connects to an environment in which he doesn't know neither trust anybody.

The third challenge is to improve the accuracy of trust models by using effective trust metrics which represent the right human behaviour in situations that require trust [5]. Those metrics should reflect motivational attitudes of humans like necessity, curiosity, satisfaction/dissatisfaction, knowledge/ignorance and expected utility.

2- Methods

Our proposed model concentrates on one important feature missing in the explored trust models: the compatibility with low-resource devices. Our model also combines the advantages of those trust models, by adopting some of their features like service-based trust [6][7], and even adds a new metric that makes the trust evaluation more accurate and less invincible to fraud. The main features of our proposed trust model are given below:

- It combines many of the good features presented in other models, like trust recommendations and the idea of trust distribution, and trust per service concept.
- It was essentially made for mobile devices with limited resources, which means it requires less overhead and uses less bandwidth.
- It combines the concept of trust evaluation and risk assessment in one trust model.
- It ensures preserving the privacy of the users and devices, without disclosing personal information about users, like time of usage and location.
- It introduces the new concept of Judgment.
- It deploys a new scheme to detect any abnormal behavior of the nodes in the network.
- It can be extended to take into consideration the contextual data for trust evaluation.
- It can provide different levels of trust based on requested services.

2.1 *Resource Minimization*

Our trust model is designed to serve smart environments where the hardware equipment has small processors and limited memories. Some trust models used a mesh-like approach in calculating trust values [8]. It means that each node of the network has to compute trust values of all neighbor nodes, even if it has no current need for exchanging data with those nodes. It also tries to use as minimal memory resources as possible. Essentially, each node in the network, independently of its nature, will need two matrices. The node will store trust values of nodes communicating with it only. Because of the memory limitation, the node will get rid of trust data related to nodes that had left the neighborhood or went dead in the network.

2.2 *Trustworthiness*

Trustworthiness is used to refer to the level of trust of an entity B in respect to another entity A. The calculation of a trustworthiness value is divided into two parts: the calculation of direct trust and the calculation of indirect trust. Direct trust is what is commonly called “Risk Assessment”. It is used for dealing with newcomers which the entity has not yet any records of trust evaluation. In case where trust is service-dependent, we added a multiplicative factor to the number of negative actions. This factor is called the Security Action Coefficient (SAC). This coefficient refers to the security level of a service. Direct trust is obtained using [9]:

$$DT = \frac{\Sigma PA_i}{\Sigma PA_i + SAC \times \Sigma NA_i} .$$

where PA_i represents the number of positive actions done by the given node and noticed by node i. NA_i refers to the number of negative actions, and SAC is the Security Action Coefficient related to the security level of the service.
The indirect trust is given by:

$$IT = \frac{\Sigma Tw_i \times J_i}{n} .$$

where Tw_i and J_i are the trustworthiness and judgment values corresponding to the node i.
The value of the net trustworthiness is a combination of direct and indirect trust values:

$$Tw = \alpha_{DT} \times DT + \alpha_{IT} \times IT .$$

where α_{IT} the indirect trust coefficient is:

$$\alpha_{IT} = \frac{TS_{self}}{TS_{self} + \Sigma \frac{TS_i}{n_{reccomm}}} \times \frac{\Sigma J_i}{n_{tot}} .$$

and α_{DT} the direct trust coefficient is:

$$\alpha_{DT} = 1 - \alpha_{IT} .$$

where *TS* self refers to the timestamp of the trust value of the node itself, while TS_i denotes the timestamp of the trust value of the node *i*. n_{tot} is the total number of nodes in the subnetwork, whereas n_{recomm} is the number of nodes that responded with recommendations.

2.3 Judgment

Judgment is one of the new features introduced that aims to imitate the human behavior in a technical approach [9]. As mentioned before, the judgment ability is represented by the overall experience of dealing with the node in question. That experience includes both the total number of signaling messages exchanged and the total number of actions classified positively or negatively.

The judgment related to the number of actions is equal to the total number of actions over the maximum number of actions. The judgment related to the number of messages exchanged. At last, the overall judgment value the multiplication of the two judgment values.

2.4 Control Messages

In order to control the aspect of trust evaluation and share the trust data, short control messages are used for that purpose. Most of these messages belong to the trigger-based type, but only two are of the periodical type, for network and security monitoring as explained below. The messages used in our proposed trust model are:

1. Recommendation messages: trigger-based messages used as request for trust recommendations about certain node. The addressed nodes will reply, if possible, with a recommendation reply that contains their trustworthiness value of the node.
2. Experience messages: similar to recommendation messages in type and purpose. However, their replies contains information about messages exchanged and positive and negative actions. Such information will be used in the calculation of Judgment and Direct Trust.
3. Hello messages: periodic messages that are issued to inform the neighbors that the node is still alive and within the network range.
4. Consistency messages: periodic messages that aim to test the consistency behavior of a certain node. This scenario is used to prevent any suspicious behavior that tends to maliciously affect the trust evaluation.
5. Knowledge Migration messages: issued only when a node is about to pass out or leave the network. The message is a notification of the availability of trust data that is going to be lost. Interested nodes will reply to get the migrated experience.

3- Results

In order to validate the proposed scheme composed of an authentication module and a trust module, we have implemented it using Java language and Eclipse IDE platform. A server in our architecture is the device that provides the service to other nodes. It can be a computer, an RFID reader, or even a sensor. The server part of the implementation contains no graphical interface, since it only calculates trust and authentication data and communicates them to other nodes. On the other hand, the client part includes a graphical interface and can run on almost any mobile device that supports Java. We choose to test our implementation on Android mobile phones using Android SDK tools in Java.

Our implementation participates in a project called “Cohabit”, a smart environment project for dependent people. The project takes place in a particular residence for disabled people called ADEP. The services provided in that smart environment are daily services needed by dependent people, such as opening the door, turning on/off the light, closing the curtains, etc. The security modules we have developed evaluate the trust between the users and their

environment before giving them access to use those services.

For the sake of tracking the steps of the trust evaluation and authentication process, many virtual nodes were created on a simulator and connected to the network where other nodes exist. When the user of the Android phone chooses a service, he will be directed to another window that lists the security steps needed to activate that service. When clicking on 'Evaluate Trust' button, the trustworthiness value of the service provider will be calculated, and the threshold value set by the administrator will be displayed. For more flexibility, the procedure was divided to several steps, to allow the user to use if the trust value is below the threshold. The next button uses the EC-SAKA protocol, used to generate a secret shared key and enable the two nodes to verify each other's identities. If the user sees two checkmarks next to the two buttons, he can now execute the service with content. Some services might not need trust evaluation and authentications, such as date/time and weather-forecast services.

4- Conclusion

After emphasizing on the necessity and importance of security, privacy and trust in smart environments, we have proposed and developed a new trust model that respects the limitation of mobile devices in terms of resources and bandwidth. Our trust model contains two new features that enhance the process of trust evaluation. The first one is a new trust metric introduced called the ability of judgment. This value tends to imitate the human rational thinking in trust. The second feature is a lightweight security monitoring ability to defend against security threats and trust forging. Our next goal is to include both *Context Awareness* and *User Profiling* to our Trust model.

References

[1] Langheinrich M. (2001) Privacy by Design-Principles of Privacy-Aware Ubiquitous Systems. In: Proceeding of the 3rd International Conference on Ubiquitous Computing (UbiComp 2001), Springer-Velag LNCS 2201. pp. 273-291.

[2] Stajano F. (2002) Security for Ubiquitous Computing. In: Halsted Press.

[3] Martin Modahl, Bikash Agarwalla, T. Scott Saponas, Gregory Abowd and Umakishore Ramachandran (2006) UbiqStack: a taxonomy for a ubiquitous computing software stack. In: Personal and Ubiquitous Computing Journal, Volume 10, Number 1, pp 21-27.

[4] Colin English, Sotirios Terzis and Paddy Nixon (2005) Towards self-protecting ubiquitous systems: monitoring trust-based interactions. In: Personal and Ubiquitous Computing Journal, Volume 10, Number 1, pp 50-54.

[5] Yin Shuxin, Ray Indrakshi (2006) A Trust Model for Pervasive Computing Environments. In: International Conference on Collaborative Computing: Networking, Applications.

[6] Taherian Mohsen, Jalili Rasool, Amini Morteza (2008) PTO: A Trust Ontology for Pervasive Environments. In: 22nd International Conference on Advanced Information Networking and Applications - Workshops, IEEE.

[7] Cheng Heng Seng, Zhang Daqing, Tan Joo Geok (2005) Protection of Privacy in Pervasive Computing Environments. In: Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05).

[8] Campbell R., Al-Muhtadi J. (2002) Towards Security and Privacy for Pervasive Computing. In: Proceedings of ISSS, Tokyo, Japan, 2002, pp 1-15.

[9] A. El Husseini, A. Mhamed, B. El Hassan, M. Mokhtari (2011) A Novel Trust-Based Authentication Scheme for Low-Resource Devices in Smart Environments. In: The second International Conference on Ambient Systems, Networks and Technologies (ANT-2011).