JPermit: Usable and Secure Registration of guest-phones into Enterprise VoIP network

R Vidya Lakshmi, Divya Krishnan, Parvathy S, Vishnudatha K, Jayaraj Poroor, Amit Dhar
Amrita Vishwa Vidyapeetham
Kerala, India - 690 525

***Abstract*—Providing a usable and secure mechanism for the registration of guest-phones to enterprise VoIP networks is a challenging problem. There are many security mechanisms that have been proposed to prevent the various attacks that occur during the registration of a SIP based VoIP phone to VoIP network. However, none of these mechanisms work for an enterprise environment where security and usability are the key issues to be considered. In this paper, a novel architecture called JPermit is proposed, which enables the secure entry of guest-phones to VoIP enterprise networks with minimum human intervention. The standard enterprise VoIP architecture is extended by introducing a Trust Bootstrap Gateway, which facilitates the trust bootstrapping between the guest-phone and the VoIP network. The JPermit architecture addresses the various security issues related to the registration of guest-phones to enterprise VoIP network. The prototype implementation of the proposed architecture is also presented.**

***Index Terms*—SIP registration, trust bootstrapping, registrar server.**

I. Introduction

In today's scenario the registration of guest-phones to a secure enterprise VoIP network is of significance owing to the need of providing security as well as usability. For guest-phones to get authenticated and thus get registered into the network, the authentication credentials should be transferred to the guest-phone. In order to transfer the credentials securely, a secure channel is required between the enterprise VoIP network and the guest-phone. There are a number of security threats like registration hijacking[1] that need to be considered while devising a mechanism for the registration of guest-phones to VoIP networks. Moreover, the mechanism should facilitate the guest phone to locate the valid registrar server of the network. Hence the registration of guest-phones to the VoIP enterprise network poses a classic trust bootstrap problem[2].

Minimizing human intervention would augment the usability of the system as well as enhance the security of the system. Therefore the registration process should be automated, for minimizing human intervention.

This paper presents the JPermit architecture, as an enhancement to the standard VoIP architecture[3], in order to enable the secure registration of guest-phones to enterprise VoIP network in a usable manner. A trusted gateway called Trust Bootstrap Gateway (TBG) is introduced to the VoIP architecture as shown in Fig 1, through which guest-phones can get registered into the network and establish a trusted relationship with the network, securely with minimum human intervention. The concept of Trust Bootstrap Gateway is inherited from SmartWhisper[4], which addresses a similar problem in enterprise wireless LAN's.

The TBG has a long lasting trusted relationship with the VoIP network. It is kept at secure places which are accessible to the guests hence preventing unprivileged people from registering to the network. The guest- phone connects itself to the TBG through the wireless link provided by it. The trust bootstrapping of the guest-phone and the network is performed using a protocol based on the concept of audio based location limited channel. Upon successful trust bootstrapping between the guest-phone and the TBG, the details of the phone are passed on to the registrar server via the TBG. The registrar server stores the guest-phone details in the database and transfers the authentication credentials back to the guest-phone via the TBG.

The proposed architecture has been implemented with the help of SIP communicator[5], an open source instant messenger for VoIP, Asterisk PBX[6], which is an open source PBX and MySQL database.

II. Related Works

RFC 3261[1] describes the various security attacks that can occur during the registration of SIP based VoIP phones. These attacks include registration hijacking where the attacker impersonates a valid user agent and performs registration on the user agent's behalf. In such a scenario, the attacker can block, record or manipulate the calls within the enterprise. Another attack is when an attacker impersonates the registrar server. The SIP based VoIP phone sends the registration request to the impersonated server. This allows the impersonated server to completely administer the calls placed by the VoIP phone, thereby raising a serious security concern. A proposed security mechanism to counter these attacks is to use strong authentication in the form of TLS[7]. In this paper, TLS is used to create a secure channel between

978-0-7695-4058-0/10 $26.00 © 2010 IEEE
DOI 10.1109/ACE.2010.32

the guest-phone and the TBG.

Various authentication mechanisms have been proposed in various other works. In [8], digest authentication mechanism is used to authenticate the VoIP phone to the registrar server during registration. The authentication is based on the fact that the VoIP phone and the registrar server has a key shared between them. The configuration authentication scheme described in [7] is used to solve the bootstrapping problem between the VoIP phones and the configuration server before the phone obtains the configuration information from the configuration server. It requires that the phones need to be preconfigured with the public key of the configuration server. Next it establishes TLS connection with the configuration server whose IP address is obtained through DHCP request, and the authenticity of the configuration server is verified. JPermit architecture does not impose the constraint that any preconfigured information be embedded in the guest-phones. The establishment of a trusted relationship between the guest-phone and the network is done through a trust bootstrapping protocol[4] which is based on audio based location limited channel.

In Talking to strangers[9], the concept of location limited channel is used to exchange a small amount of cryptographic information to perform authentication to access the network. It introduces the concept of demonstrative identification, that is, the identity of the device is checked by exchanging cryptographic information through the location limited channel. Location limited channel reduces human intervention. The location limited channel is implemented using IrDA. Seeing is believing[10] proposes visual channel for authentication. It uses camera on the devices to capture the printed or displayed bar-codes of public-key hashes from other devices, and to verify authenticity using the wireless link. JPermit architecture uses the concept of audio based location limited channel thus overcoming the disadvantages of visual based location limited channel like precise positioning and image clarity. Since all the VoIP phones have audio capability, use of audio based location limited channel is not an overhead.

Signaling is the process of creating and managing VoIP calls between end points. Many methods have been proposed in the past for the authentication of end users during signaling like the method proposed in Authentication of Signaling in VoIP Applications [11]. Wang and Verma[12], introduce a network-based authentication mechanism during SIP signaling. Unlike their scheme, JPermit architecture introduces the concept of authentication of a guest-phone during its registration into the enterprise VoIP network. Since, registration is the process by which new VoIP phones become part of the enterprise VoIP network, the proposed scheme enhances the security of the network by ensuring that only privileged and trusted phones are entering and registering with the network.

The need for usable authentication has been emphasized in [13], where the author mentions that systems introducing authentication techniques need to bring a balance between both security and usability. The JPermit architecture implements an authentication scheme which ensures both security and usability. In the JPermit architecture, utmost usability to the user is ensured by minimizing human intervention in the authentication process. The only thing that the user needs to perform is to hold the VoIP phone near the TBG.

III. System Architecture

The system architecture is shown in Fig 1.

Figure 1. System Architecture

The registration of guest-phone to enterprise VoIP network requires that

1) The guest-phone locates the valid registrar server
2) Only privileged guest-phones get registered to the network

This problem of trust bootstrapping is solved in JPermit architecture using the trust bootstrapping protocol[4] shown in fig 2.

The two main steps involved in the registration process are described below:

1) Trust bootstrapping between the guest-phone and the TBG using the trust bootstrapping protocol

The TBG executes the server code which handles the requests from the guest-phones and performs the trust bootstrapping between the TBG and the guest-phone. The TBG obtains a certificate issued and signed by an Enterprise Certification Authority (CA) by sending a Certificate Signing Request (CSR) to the Certification Authority. The signed certificate establishes the identity of the TBG. The client side of the JPermit architecture is implemented as a signed applet because applets can be run on java enabled browsers on VoIP phones. Moreover implementing the JPermit client as an applet doesn't require additional software to be installed in the phone. The applet is signed using the TBG's certificate.

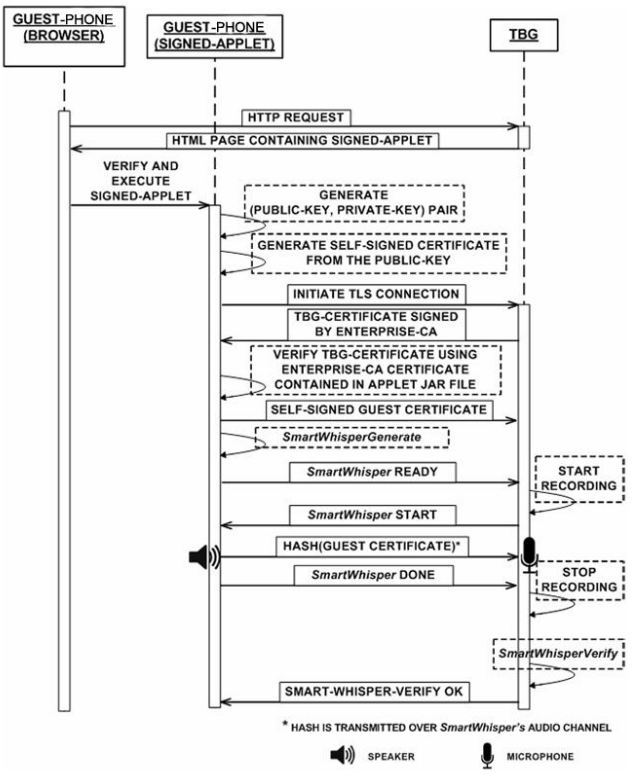


Figure 2. Trust bootstrapping Protocol

The guest-phone's browser loads the signed applet from TBG through a HTTP request.

In the applet, the trust bootstrapping process is initiated by the creation of cryptographic keys and signed certificate for the guest-phone. Next, a TLS (Transport Layer Security) connection is established between the TBG and the guest-phone through TLS handshake. The establishment of a TLS connection creates a secure channel between the guest-phone and the TBG. TLS handshake is a series of ten steps, wherein the guest-phone and the TBG exchange their certificates. The guest-phone is able to verify the TBG's certificate and hence the identity of the TBG using the TBG's certificate signed by the Enterprise CA contained in the applet jar file. Now, the TBG has to verify whether the guest-phone which has loaded the applet and established TLS connection is privileged to access the network, by checking whether the device is near the TBG. The verification is done based on the assumption that the TBG is accessible only to the privileged guests whereas unprivileged users can't access the TBG.

Audio based location limited channel is used to verify the identity of the guest-phone. The JPermit client first generates a set of audio sample values based on the cryptographic hash of the JPermit client's signed digital certificate, by executing SMARTWHISPER-GENERATE algorithm[4]. The cryptographic hash is calculated using RSA algorithm. Hamming bits are added to the hash for performing single bit error correction, followed by Manchester encoding of the hash bits. Then a carrier wave of 16kHz frequency is modulated according to the hash bits using Amplitude Shift Keying and is played through the guest-phone's speaker.

The TBG records the played audio sample and then executes SMARTWHISPER-VERIFY[4] algorithm. The recorded sample which is in time domain is converted to frequency domain using Fast Fourier Transform. Unwanted noise signals are removed using a Butterworth filter. Then it is demodulated and decoded to recover the hash. The decoded hash is compared with the hash of the guest-phone's certificate received through the TLS connection and is verified. If the two hashes match, verification succeeds, thereby confirming the authenticity of the guest-phone.

2) Registration of the guest-phone to the VoIP network by the Registrar server via TBG

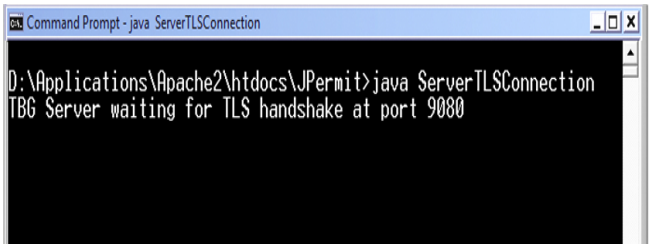
The creation of secure channel between the guest-phone and the TBG results in the establishment of a temporary secure channel between the guest-phone and the registrar server of the VoIP network. After authenticating the guest-phone, the credentials of the guest-phone is obtained by the TBG through the established secure channel and are sent to the registrar server. The registrar server creates an account for the phone in the VoIP network and sends back the account credentials to the guest-phone via the TBG. Henceforth, the guest-phones can access the services offered by the enterprise VoIP network.

Hence, JPermit prevents the registration attacks like registration hijacking and server impersonation through the trust bootstrapping protocol based on audio based location limited channel, thereby enabling secure registration of guest-phones to VoIP enterprise network. JPermit allows the entire process of guest-phone registration in an automated manner, thereby reducing human intervention and maximising usability.

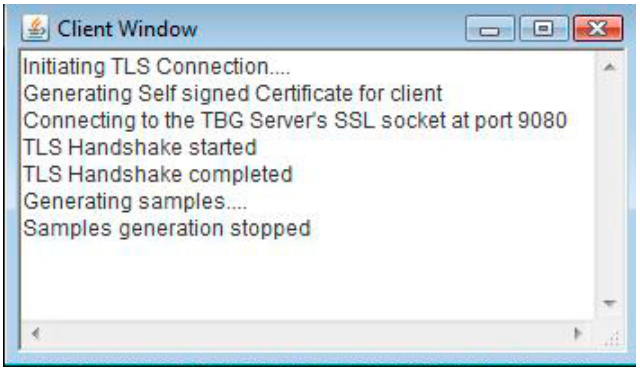
IV. Implementation

A working prototype of the JPermit architecture was implemented in Java language[14]. Java was opted because it contained packages like Java Sound API and JSSE for implementing the different components of the proposed architecture. Moreover it has been planned to test the prototype using JVM enabled VoIP phones[15] in future work. For the current prototype, a laptop installed with SIP communicator[5] would act as the guest-phone. A laptop installed with Windows Vista and Apache server[16] served as the TBG. A small enterprise network was set up with Asterisk PBX[6], acting as the registrar server and MySQL, serving as the database for storing the SIP account details. Asterisk is a widely used open source PBX, suitable for small enterprise VoIP network and MySQL is a widely used Open Source database.

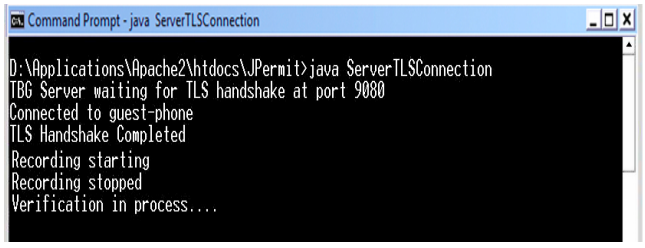
Fig 3 shows the screenshots of the client side and server side implementation.



(a) TBG Server waiting for TLS handshake



(b) Applet in guest-phone during the trust bootstrapping protocol



(c) TBG Server during the trust bootstrapping protocol

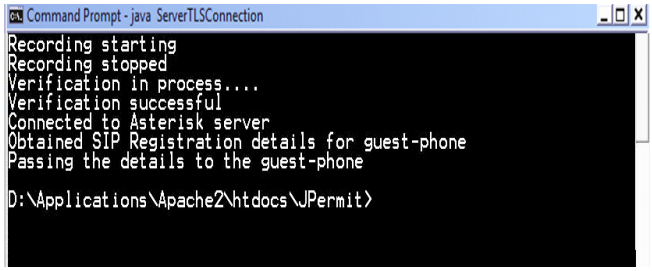
Figure 3. Screenshot depicting prototype implementation of JPermit

In the current prototype, the cryptographic key pair required for the generation of the TBG's certificate and the signed certificate were generated using a Java based tool called keytool[17].

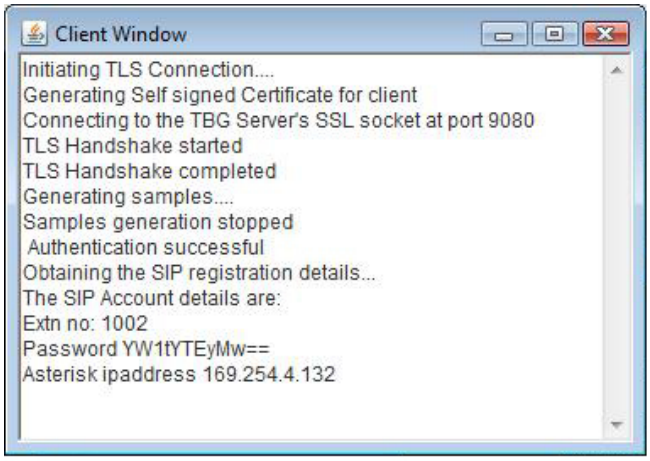
The java.security package and sun.security package were used for the creation of cryptographic keys and signed certificate for the JPermit client. TLS connection was implemented using JSSE (Java Secure Socket Extension) package which contains the implementation of the java version of the TLS protocol.

Java threads were used to synchronize the working of various processes like the TLS handshake and audio sample capturing.

On successful authentication of the guest-phone by the



(d) TBG Server after successful registration



(e) Applet in guest-phone after successful registration

Figure 3. Screenshot depicting prototype implementation of JPermit

TBG, the guest-phone details were send by the TBG to the Asterisk PBX through the already existing connection with it. The Asterisk PBX then created a SIP account consisting of a unique extension number and a random password and these details were stored in the MySQL database. Next, the configuration information consisting of the extension number, password and IP address of the Asterisk server was sent to the TBG, and then to the SIP communicator which acted as the guest-phone. The information was used to configure the SIP communicator. This was done by modifying the sip-communicator.xml file of the SIP communicator using Java API for XML processing. The password in the sip communicator.xml file was stored in base 64 encoding. Through this SIP account created for the SIP communicator, calls could be placed with other VoIP phones registered with the Asterisk.

V. Performance Analysis

For the performance analysis of the registration process, the rate of successful registrations were tested in different environments at varied noise levels. Noise was introduced in the form of sound of fan, music, generator, AC, chats etc. Based on the above tests, it was inferred that the success rate mainly depends on the frequency as well as the amplitude of the noise. Further study revealed that the frequency dependency is more than the amplitude dependency. The success rate was 100%

up to a noise of 55dB. Then it decreased to 50% when the noise level was in the range of 55 to 62 dB. Then gradually the rate increased but after 70dB the success rate decreased. The success rate was zero after the noise crossed 82 dB. An analysis of these results is showed in Fig 4.

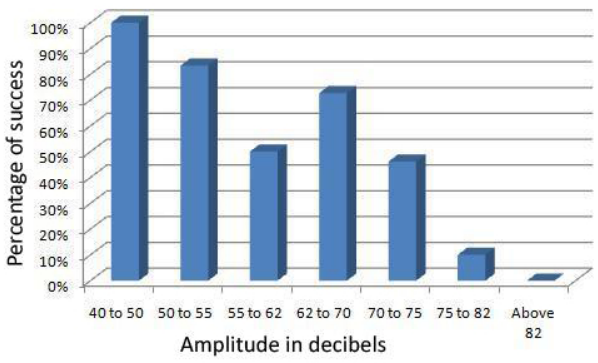


Figure 4. Performance Analysis based on amplitude of noise

The time from the loading of the applet till the successful authentication of the guest-phone was recorded. It was noted that it took on an average 34 seconds to successfully authenticate the phone.

The maximum level of noise for which the device was authenticated was recorded using a Sound Level Meter. It was found that it successfully authenticated the guest-phone while the surrounding noise levels were in the range of 60-70 db.

The rate of success based on multiple consecutive executions of the authentication was studied and it was found that on an average, 8 out of 10 times, the verification was successful.

VI. Conclusion

Through the JPermit architecture the problem of providing a usable and secure mechanism for the registration of guest-phones to enterprise VoIP networks was solved. Authentication of guest-phones by using TLS handshake and audio based location limited channel added an extra layer of security to the VoIP network. The problem of trust bootstrapping between the guest-phone and registrar server was solved through the TBG. The JPermit architecture provides maximum usability to the user. Also, JPermit can be easily implemented in any enterprise VoIP network. JPermit can be extended to include access control, which will set privileges for the guest-phones to access the enterprise resources.

References

[1] J. Rosenberg *et al.*, "SIP: Session Initiation Protocol," RFC 3261, June 2002.

[2] F. Stajano and R. Anderson, "The Resurrecting Duckling: Security Issues for Ubiquitous Computing," *IEEE Computer*, vol. 35, pp. 22–26, April 2002.

[3] S. Zeadally and F. Siddiqui, "Design and Implementation of a SIP-based VoIP Architecture," in *Proc. 18th International Conference on Advanced Information Networking and Application (AINA'04).*

[4] J. Poroor and A. Dhar, "SmartWhisper: Automated collaborative authentication with minimal human intervention in Secure Wireless Enterprise 802.11 Networks," in *IEEE Workshop on Collaborative Security Technologies (CoSec'09)*, Bangalore, 9 December 2009.

[5] SIP Communicator. [Online]. Available: http://sip-communicator.org/

[6] Asterisk. [Online]. Available: http://www.asterisk.org/

[7] D. Butcher, X. Li, and J. Guo, "Security Challenge and Defense in VoIP Infrastructures," *IEEE Transactions on Systems, Man, and Cybernetics PART C: Applications and Reviews*, vol. 37, no. 6, November 2007.

[8] A. M. Hagalisletto and L. Strand, "Formal modeling of authentication in SIP registration," in *Proc. The Second International Conference on Emerging Security Information, Systems and Technologies (SECURWARE '08)*, Cap Esterel, France, 25-31 August 2008, pp. 16–21.

[9] D. Balfanz, D. K. Smetters, P. Stewart, and H. C. Wong, "Talking to Strangers: Authentication in Ad-hoc Wireless Networks," in *Proc. Network and Distributed System Security Symposium (NDSS '02)*, San Diego, California, USA, February 2002.

[10] J. M. McCune, A. Perrig, and M. K. Reiter, "Seeing-Is-Believing: Using Camera Phones for Human-Verifiable Authentication," in *Proc. 2005 IEEE Symposium on Security and Privacy*, Washington, DC, 8-11 May 2005.

[11] R. Srinivasan *et al.*, "Authentication of Signaling in VoIP Applications," in *Asia-Pacific Conference on Communications*, Perth, Western Australia, 3-5 October 2005.

[12] L. Wang and P. K. Verma, "A Network Based Authentication Scheme for VoIP," in *International Conference on Communication Technology (ICCT'06)*, 27-30 November 2006.

[13] S. Chiasson, "Usable Authentication and click-based graphical passwords," Ph.D. dissertation, Carleton University, Ottawa, Ontario, December 2008.

[14] Java. [Online]. Available: http://java.sun.com/

[15] G. Lawton, "Moving Java into Mobile Phones," *IEEE Computer*, vol. 35, no. 6, June 2002.

[16] Apache HTTP server. [Online]. Available: http://httpd.apache.org/

[17] Keytool. [Online]. Available: http://java.sun.com/javase/6/docs/technotes/tools/windows/keytool.html