Trust and risk analysis in Jini Service Design

M.Kavitha
Department of Information Technology
SRM University
kavitha.mu@ktr.srmuniv.ac.in

D.Hemavathi
Department of Information Technology
SRM Univeristy
Hemavathi.d@ktr.srmuniv.ac.in

Sivasankari.S
Department of Information Technology
SRM Univeristy
sivasankari.s@ktr.srmuniv.ac.in

Abstract--Currently, scores of pervasive computing systems have not yet taken trust based on service discovery, due to the scarcity of a suitable sustaining infrastructure. In this paper we propose an adapted Jini service architecture for Pervasive Computing. This architecture processes the service discoveries which also preserve ACID properties. The infrastructure is built on both the Jini service architecture and the Kerberos technology. Such a infrastructure each system can discover all service and can utilize all possible services.

Keywords: ACID, Jini, service discovery, pervasive computing, Kerberos.

I INTRODUCTION

As computing becomes more pervasive, users expect to access services associated information at anytime and anywhere. Trust management not only improves security and Privacy for devices in pervasive computing environments, but also mends the efficiency and quality of communications among users. In this paper we present a modified structure for a previously proposed Jini service architecture scheme in a pervasive computing environment [5]. This paper's main offerings are Service discovery based on trust scheme's construction and the laying down of the foundations for the theoretical concepts of pervasive computing.

Jini takes care of service discovery in the pervasive environment. After a service requestor achieves service object provided by service from lookup service, it can communicate with service directly.

The rest of the paper is organized as follows: Section 2 presents existing related work. Section 3 presents the proposed modified Jini service architecture. Section 4 provides the performance evaluation. Finally, section 5 discusses the conclusion and ideas for future work.

II RELATED WORK

The long-established centralized security mechanism scheme is not directly applicable and the existing user authentication and access control schemes are inadequate to ensure security in pervasive computing [1]. Calculation of Trust value is in a distributed fashion and the exchange of trust information is carried out on demand to reduce communication overhead.

Predicament comes when a service providers and users in a service detection session has not been well addressed [2]. The Users and service providers interchange partial and encoded forms of their identities and service evidence. Some times many numbers of encrypted messages are transferred to maintain trust. Still service discovery is a major problem [2]. When an unknown device tries to build a trust relationships problem occurs. Existing model [3] does not address security issues. There are also models that support security with the assistance of additional hardware. But these models don't take risk into account. Adaptive Trust model and Risk Model Evaluation reduce the overhead of encrypting messages each time a device requests a service. When there is no prior information or history of service available, building a trust relationship is very difficult. Problem [3] comes when we integrate multiple resources and sharing.

2.1. Existing Jini Architecture

Figure 2.1 shows the existing Jini architecture which is briefed as follows:

- Lookup Service:

The lookup service is the dominant component for Jini and provides key point of interaction between service and client [5]. The lookup service is also responsible for keeping track of all the services that are currently available within a Jini arrangement, also accountable for providing the proxies to transfer with the services.

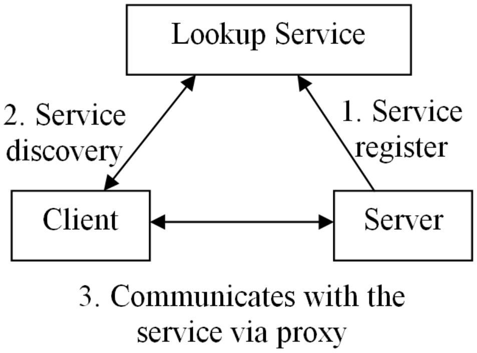

Figure 2.1. Existing Jini Architecture

- Service Provider (Server)

It based on Java Remote Method Invocation (RMI), which offers mechanisms to find and use the service.

- Service Requestor (Client)

Service requestor can search the Jini services by service ID, Service Interface or some specific Service Attributes. Then the user of the Jini service downloads the service proxy object (stub) from the Lookup service and access the service via the stub which is liable to handle the communication with the remote service.

III. ADAPTED JINI SERVICE ARCHITECUTURE

Figure 3.1 shows the adapted Jini architecture which is concise as follows:

3.1 Lookup Service

Lookup service is a program which runs in server side and is responsible for service discovery based on trust management scheme. It is responsible for initiating the authentication server and service server. Lookup service installs the lookup proxy in the client side, through which the client can proceed for further communication. Consider the client is using one service and request for additional service the lookup service it check the trust of previous service and provide the additional service based on the trust of previous service. The main advantage of the lookup service is with the help of single ticket the user can access multiple services. Lookup service takes care of lifetime of the ticket and control the access the service.

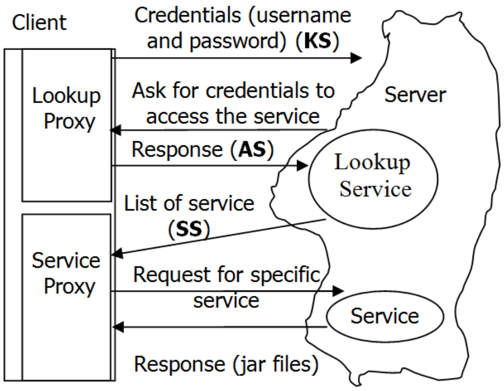

Figure 2.1. Adapted Jini Architecture

3.2 Servers used

- **Kerberos Server(KS)**

When basic credentials like username and password are provided Kerberos server is invoked. It just checks the username and password. For more security and privacy purpose Lookup service will ask for more credentials to access the service and by installing the lookup proxy in the client.

- **Authentication Server(AS)**

When additional credentials are provided the authentication server is invoked. This server checks for the user is the valid person or not for the pervasive environment to access the service.

- **Service Server (SS)**

After the work of Authentication server is over it gives the control to service server which is handled by lookup service. It then lists all the services that are available to the user by installing the service proxy in client side. With the help of service proxy the user communicate with the specific service. The service then response to the client by installing the jar files of the specified client. Jar files are nothing but the collection of class files.

IV PERFORMANCE EVALUATION

In order to evaluate our proposed adapted Jini service architecture, we ran a series of simulation experiments on an e-learning service discovery environment. This pervasive computing environment is made up of multiple nodes and service running in different nodes. For simplicity we assume that each system provide multiple service and the service are not repeated and at a particular point of time the user will not log on to the system simultaneously from multiple nodes. In our paper we considered three services like file service, chat service and virtual classroom service.

4.1. Performance metrics

We have used two performance metrics, average throughput and average packet loss ratio. Each performance metric is investigated by varying the network's size and traffic load. The two performance metrics are described below.

Network Size: We define network size as number of system connected in the network work in pervasive environment. We evaluate our proposed scheme's performance by comparing network size with time using Jini and without Jini in simulations.

Average Throughput: Throughput is the quantity of packets that an middle node has effectively delivered during a set period.. As Network size increase packet loss will also be increased.

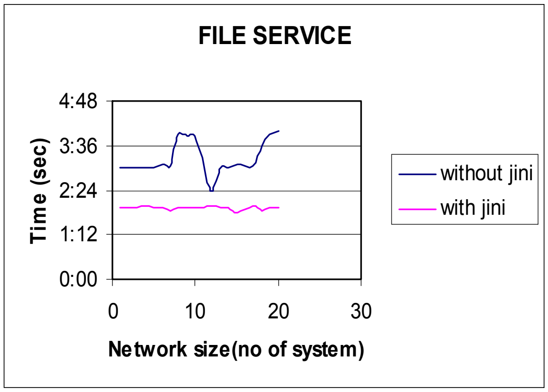

Figure 4.1 Time taken for file service with and without Jini

Figure 4.1 shows the time taken for file service with Jini architecture. When we draw graph between Network Size and Time, When network size increases (i.e Number of systems connected in the network) time increases without Jini technology, the time taken is above 2sec as show above. But with Jini the time taken is below 2sec because the time taken depends upon the size of the file transferred.

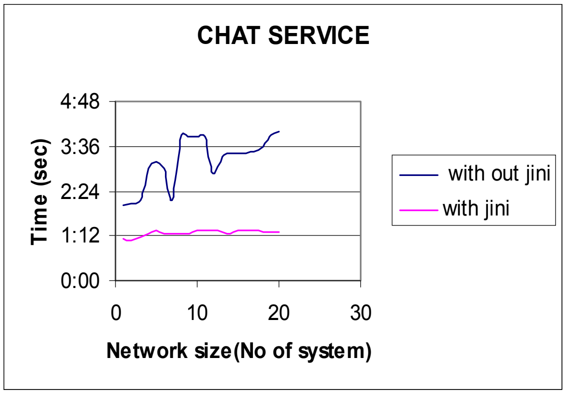

Figure 4.2 Time taken for chat service with and without Jini

Figure 4.2 shows the time taken for chat service with Jini architecture. When we draw graph between Network Size and Time, When network size increases (i.e Number of systems connected in the network) time increases without Jini technology, the time taken is above 1:50 sec as show above. But with Jini the time taken is around 1sec because chat service is just the transfer of simple text messages.

Figure 4.3 shows the time taken for virtual classroom service with Jini architecture. When we draw graph between Network Size and Time, When network size increases (i.e Number of systems connected in the network) time increases without Jini technology, the time taken is above 4:50 seconds as show above. But with Jini the time taken is around 3 seconds because it is similar to live telecast.

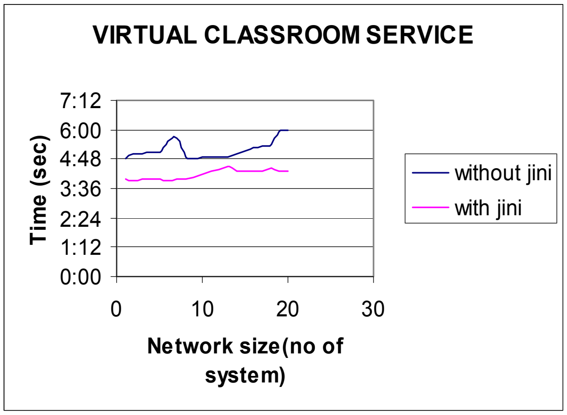

Figure 4.3 Time taken for virtual classroom service with and without Jini

V CONCLUSION AND FUTURE WORK

This paper has discussed several methodical issues elaborate in extending the Jini service architecture via improving Jini service lookup, location and dilemma mechanism. The main benefits of the work is a Jini service can be accessible from open-air of a Jini community and will be invoked in the similar way as any additional Web services. It is worth citing that we have not reformed the basic Jini philosophy. Integrating Kerberos process for progressive approach to generate ticket and provide privacy date transfer. Here single sign-in process is also included i.e. with the help of single ticket more service can be offered to the user.

In future a separate log file can be maintained (All login process are recorded). This log file contains details about the login, access right; access denial etc. with the help of this log file the entire risk process is performed. Risk analysis and Risk avoidance is done in next phase. Service Discovery can be performed in mobile network with the connection of PDAs.

6. References

[1] Tao Sun Mieso.K.Denko, “A Distributed Trust Management Scheme in the Pervasive Computing Environment”, Electrical and Computer Engineering, 2007.

CCECE 2007. Canadian Conference on April 2007. pp. 1219-1222.

[2] Feng Zhu , wei zhu, matt W.Mutka "Private and Secure Service Discovery via Progressive and Probabilistic Exposure" IEEE Transactions on Parallel and Distributed systems, Vol 18. No. 11 Nov 2007. pp. 1565-1577.

[3] Sheikh I. Ahamed, Moushumi Sharmin, and Shameem Ahmed "A Risk-aware Trust Based Secure Resource Discovery (RTSRD) Model for Pervasive Computing", 6th Annual IEEE International conference on pervasive computing and communication. 2008. pp. 590-595.

[4] Swaroop kalasapur "Dynamic Service Composition in pervasive Computing", IEEE transaction on parallel and distributed systems, Vol 18,No.7 july 2007. pp. 907-918.

[5] Baomin Xu1, Qiang Gao2 , Xiaoping Yang1 "Extensions to Jini Service Architecture for Pervasive Computing " 2006 1st International Symposium on Pervasive Computing and Applications, pp. 90-94

[6] Salutation Consortium, .Salutation Architecture Specification Version 2.0c . Part 1 The Salutation Consortium, June 1, 1999.