Light weight Authentication System and resource Monitoirng using MAS

Abhilasha Sharma  
RKDF Institute of Science Tech.  
Bhopal, India

Rajdeep Singh  
RKDF Institute of Science Tech.  
Bhopal, India

Jitendra S Rathore  
Technocrats Institute of Tech  
Bhopal, India

Abstract—Application, resource and Network supervision and trust management is a significant issue due to today’s speedily development of computer and communication environment specially in Local Area Network (LAN). Client-server based network management approach suffer from problems such as insufficient scalability, interoperability, reliability, and flexibility, as networks become more geographically distributed [1]. Another big issue is trust management. RSA, DES and Kerberos is another good methods to achieve authentication but require but high computation is a big deal for LAN, another issue is availability. In this paper, we have proposed a new (novel) light weight approach for resource and trust management using the concept of Multi agent system. Proposed method used the concept of certificate authority (AS of Kerberos) for authenticating the users in LAN or peer to peer network. Availability and minimum delay are the key factor of any authentication scheme, in this paper we proposed a fresh new concept for authenticity and supervision of resources. Our Mobile agent based solution will work same as Kerberos with better throughput and with high availability due to distributed and roaming features of MAS system. Proposed method provides good solution for trust management as well as supervision for network resource and application. We have used SPADE for development of MAS.

Keywords—Authentication, kerberos, MAS, Resource management, SPADE

I. INTRODUCTION

Application, resource and Network supervision is a significant issue due to today’s speedily development of computer and communication environment. Client-server based network management approach suffer from problems such as insufficient scalability, interoperability, reliability, and flexibility as networks become more geographically distributed [1]. A framework for an intelligent Multi Agents System (MAS) architecture is proposed using agents to achieve distributed management. The policies that govern the mobile agents’ operation are specified by the management entity. The MAS architecture diminish the complexity of management (application, resource or network) at the managing entity by entrusting part of the management responsibility to the managed network entities. Adding mobility and intelligence to an agent provides many advantages such as extensibility and portability. The intelligence of mobile agents helped to make dynamic decisions.

Management of network and other resources is a distributed activity by nature follows the widely used client-server model [1]. The well known developed application protocol is Simple Network Management Protocol (SNMP) [4]. Most of the essential functions of network and other resources management are well realized in this client-server model, and the network entities with limited computation power follow SNMP’s philosophy of simple and passive agent structures. However, this approach has several technical confines like scalability, reliability, performance degradation, and more complicated as well as networks are expanding and more distributed [5]. Distributed management with authorization is another alternate to centralized management. In distributed management system there must be authenticate applications that concurrently worked as managing as well as managed agents (or hosts to agents). The distributed management architecture was developed to trim down the centralized management system computation yoke at the managing entity, to reduce and localize the network traffic by decreasing overhead due to polling [6].

Efficient distributed management architecture must deal with the reliability, flexibility, consistency, and scalability [7]. Managing of public and private keys in a large organization is a big challenge. Software agents can be an adaptive and reactive method for administration and authenticate users trying to connect to network resources. The advantages are that the agents can query multiple information sources to select the level of trust to entrust to a user [8]. The task of validating legitimate users over distributed network and services remains a tricky practice [9].

Due to recent advances in web services, Quality of Service (QoS) becomes a key factor [2] to distinguish service providers. Since current web service and technologies standards are ill with of QoS. Software agents have been recognized as a promising technology for organizing network and web services. Using FIPA [3] compliant Multi Agents we were able to propose a Multi Agents based web service QoS Management Architecture.

In this paper we have give the solution of two problems first one is authentication of the users to use network services and second is the supervision and management of network resources. Our proposed scheme used the core concept of Multi Agent System (MAS). For developing of agents we have used the Smart Python Based Agent Development...
Environment (SPADE) [10] version 2 on UBUNTU 11.10 environment.

Rest of the paper organized as follow, section 2 give the brief overview of basic terminology and background to understands idea clearly , section 3 insight on related work of authentication and resource supervision in distributed environment, section 4 discuss the proposed method and finally section 5 conclude the paper.

II. BASIC TERMINOLOGY AND BACKGROUND

A. Authentication

Authentication ensures that the identity of particular cannot be ambiguated or misrepresented. In addition, interactions between entities maybe anonymous and still require that the participants be authenticated; some information about the entity is known and is adequate for interaction [11]. Authentication is the method allows users (sender or receiver) of in sequence to validation. If the communications party has not validated each other, there is no faith in the activities supplied by either party. Lots of research based on Authentication have been used some of them are highly complex and secure methods or a simplest one. The simplest form of authentication is the transmission of a shared password between entities wishing to authenticate each other.

Following factor affect the authentication-
1. What you know – password.
2. What you have – Smart card or token.
3. What you are – Fingerprint, handprint, retina pattern, voice and keystroke pattern etc.

Network authentication used authentication protocol like, digital signature, username/password and smart card. Some well known authentication protocols are Kerberos, CHAP and Microsoft CHAP.

Authentication is one of the major concerns of information security especially in distributed environments [12]. Marcel Waldvogel [13], address the necessity of additional features for distributed environment: Quality of Service and resource reservation issues [14] [15]. Reliable transmission of data and concurrency oath is usually measured to be application-specific, if overhead is to be minimal [16], [17]. But currently the prerequisite of confidentiality and authenticity for group members is still missing. Existing methods often necessitate human intervention (manual keying is common), or limit the dynamics provided by multicasting and required by many applications.

B. Kerberos

Kerberos is used as an authentication protocol, allows communication between hosts over non-secure networks. It used client-server model. Kerberos used mutual authentication. Client and server identify each other.

a) Kerberos method perform following steps for authentication-
- Every service request needs a ticket.

- Tickets come from the TGS (except the ticket for the TGS!).
- Workstations cannot understand tickets; they are encrypted using the server key.
- Every ticket has an associated session key.
- Tickets are reusable.
- Tickets have a finite lifetime.
- Authenticators are only used once (new connection to a server).
- Authenticators expire fast.
- Server maintains list of authenticators (prevent stolen authenticators).

b) Ticket Contents:
- Client name (user login name)
- Server name
- Client Host network address
- Session Key for Client/Server
- Ticket lifetime
- Creation timestamp

Primer designer of Kerberos was Steve Miller and Clifford Neuman.

C. Multi Agent System (MAS)

Agent-based computing permits proficient utilization of resources and amortizes communication delay in a distributed environment. In a dynamically and heterogeneous environment like the Internet, no assumptions can be made about execution environments of agents [11]. Agents have many characteristics like [18]; they are social, mobility and migration. Interested readers will refer Russell and Norvig [19] for agent characteristics. Agents communicate with other agents through message passing KQML [20] and FIPA-ACL [21] is two well known languages used agents for communication. Agents also negotiate with other agents this process called “searching for an agreement” [22]. The function that maps input to an agent act is called Agent Function or Behavior Agent Architecture [19]. Many different multi-agent frameworks have been proposed [23], [24], [25], [26], [27].

D. SPADE

Simply put, SPADE[10] is an agent platform based on the XMPP/Jabber technology. This technology offers by itself many features and facilities that ease the construction of MAS, such as an existing communication channel, the concepts of users (agents) and servers (platforms) and an extensible communication protocol based on XML, just like FIPA-ACL. Many other agent platforms exist, but SPADE is the first to base its roots on the XMPP [28] technology.

The SPADE Agent Platform does not require (but strongly recommends) the operation of agents made with the SPADE Agent Library (see next section). The platform itself uses the library to empower its internals, but aside from that, you can develop your own agents in the programming language...
of your choice and use them with SPADE. The only requirement those agents must fulfill is to be able to communicate through the XMPP protocol [28]. The FIPA-ACL messages will be embedded in XMPP messages. Be warned, however, that some features of the whole SPADE experience may not be available if you do not use the SPADE Agent Library to build your agents.

SPADE is written in the Python programming language. In order to fully understand and use SPADE, a bit of knowledge about Python is required.

III. RELATED WORK

Marcel Waldvogel [29], address the necessity of additional features for distributed environment: Quality of Service and resource reservation issues [30] [31]. Reliable transmission of data and concurrency oaths is usually measured to be application-specific, if overhead is to be minimal [32], [33]. But currently the prerequisite of confidentiality and authenticity for group members is still missing. Existing methods often necessitate human intervention (manual keying is common), or limit the dynamics provided by multicasting and required by many applications.

With the rapid growth and development of Computer Networks, avail services located from remote places is easy. Sometimes these services use the personal data of users like on-line account passwords while doing on-line transactions, thus need of security become prime importance. To provide quicker and safe communication services to users, various authentication protocols which offer both, authorization and authentication and integrity and secrecy of messages have been utilized. Authentication protocols are good security mechanism whereby each party is assured its identity to one another. One of such well known authentication protocol which is commonly used is Kerberos. Kerberos was developed in the Athena Project at the Massachusetts Institute of Technology (MIT) [1], is a network authentication protocol, which allow communication over a non-secure network using secure manner. It is based on client/server model and it provides mutual authentication [34]. Kerberos is the most standard single sign-on protocols. Presently Kerberos is widely used for providing security on networks, but has several potential security vulnerabilities in it. One of them is the clock synchronization of authentication code in network; the attacker breach the wall using replay attack by amending the host time. Other one is guessing of password through the password dictionary due to weak password used by users. Improved Kerberos [35] has been improved the shortcomings in the previous Kerberos, but replay and password attack still remain.

IV. PROPOSED SOLUTION

Our proposed solution is to use a distributed SPADE [10] agent-based application to deal with the process of user authentication and supervision of user credentials. The agent architecture provides a number of security services with the goal of automating the process of user authentication and trust management. In particular, the agents handle all password, encryption keys and certificate management [9]. Our proposed prototype agent architecture offer two function first one is Light weight solution to the authentication problem and second objective is monitoring and supervision of network resource and applications. We are using SPADE [10] as our multi-agent framework, a XMPP [28] server is the foundation for our communication and provide interface to agents. Each area of monitoring should have one or more agent which will decide what to do with the information receive like: communicate to the customer, negotiate with others and check which information is valid and correct.

The SPADE framework acts as a XMPP server and where all agents connect to and are responsible to manage communication through XMPP protocol. Our proposed method has divided into two sections, first is Authentication of users using MAS and second is supervision and management of resources.

A. Authentication

Our main key agenda for proposed work is its simplicity. This is very new concept we are going to proposed in the field of cryptography. Our first work is, to test on PEER TO PEER network, then for Internet afterward in wireless environment.

In this paper, we will expand the idea from CA (Certificate Authority) and KDC of Kerberos with Mobile agent System (MAS) for doing same (Key exchange for authentication). Key distribution is the major function of cryptography; we used the concept of Mobile agent for efficient key management. Agents have mobility property that allows an agent to move and migrate from one host to another on a network. Mobility is the core concept we are using for key management. In traditional cryptography the function of key distribution was handle by certification authority (CA) in asymmetric and in case of symmetric, KDC (Key Distribution Center) was used. Other methods (like DH Kerberos) of key distribution requires high computation that slow- downs the CPU performance, as well as there is chances of compromising.

In this paper we suggested a new and efficient scheme for key management using mobile agent. The key idea behind this scheme, we have designed agents that reside on a host and move to network, when any host wants to send message then request to CA-Agent that stores public and private key pairs for source and destination. After completing, registration (for a new arrival host), and validation process the CA-Agent issues the secret key to that host and he (host) can able to send data securely. For this task, we will design 3 types of agents, Reg-Agent for registration of users for issuing private and public key. Second is Valid-Agent that checks authenticity of a user
(host) and third agent CA-Agent which issues the session key for secure communication like SSL.

Advantage of this scheme is that, for all type of authentication (Registration, verification and Session), we make different agent that reduces the computation due to autonomous and social property of agent, and the probability of compromising of an agent (CA-Agent) is less. If an agent is destroy or comprising then other agents can easily identified. This solution also gives high response due to mobility of an agent. And security is more because an agent is an intelligent system, that can cloned itself.

Figure 1 show the internal architecture of Certificate authority agent using SPADE 2. The entire three agents will be run on SPADE using XMPP protocol.

![Figure 1 Proposed Authentication system using SPADE](image)

To test the validity and performance of our agent based authentication system, we will compare the performance of our proposed system with Kerberos 5, on ubuntu 11.10 machine.

**B. Management and supervision of Resources using MAS**

Developing a MAS application means follow the standards. The SPADE platform was developed in Python language, is FIPA compliant and offers to developers a simple API which can be used to communicate, create conferences between agents and even bring out services on a Directory Facilitator (DF). SPADE agents have behaviors like Periodic, Time Out, Event, Finite State Machine, One Shot and Cyclic by extending default classes to your needs. Following services and supervision performed by our MAS based system-

- **a) Request a service**
- **b) Calculate response-time**
- **c) Send messages**
- **d) Register the information**
- **e) Communicate with its superiors (managers)**
- **f) Check the log (being serviced used by users)**

Our proposed method provides the solution to network management with maintaining authenticity for peer to peer and distributed environment using the concept of agent system. Our methods requires less computation and fast access with high availability due MAS features, our authentication system has light weight because does not require high computation.

**V. CONCLUSION**

In this paper we have proposed a light weight authentication system especially in peer to peer network using the concept of multi agent system technology. We will implement our scheme using SPADE2 agent tool. It uses the python and XMPP protocol. Primarily results show the satisfactory results as compared to Kerberos. Our method offers fast solution with high availability.

**REFERENCES**


