A FRAMEWORK TO DESIGN SECURE EMAIL SERVICE IN THE CLOUD

Midasala Anusha Rani[1], Paparao Rapuri[2], Betam Suresh[3]

[1]Pursuing M.Tech(CSE), [2]Asst. Professor in Department of CSE , [3]HOD
Vikas Group of Institutions, Nunna, Vijayawada. Affiliated to JNTU- Kakinada, A.P, (India)

ABSTRACT

In general if we consider the user mental models may not match at any time and the process of underlying of system work and the raised problems solving are different if we compare with the humans and systems. User can think and he can create, develop both the things of calculation he can do, if we go for the system is different it can't create anything but it uses to simply solve the things and modifications only. So to demonstrate and to modify this we are proposed an attribute based and useful secure email servicing. Here in this for t maintain a trust based relationship between that we are using an email authentication, for the further process and secure purpose. In some of the cases humans are good at do the things comparing with the systems. Mainly our process will put the things of human into machines and that has to do well to perform like humans, here in this paper we are proposed that with an secure email based boundaries to make an ideal design for the process and make an early experience to test our users to perform the process in well manner.

Keywords: Email, Boundaries, Humans, System Work, Security Based.

I INTRODUCTION

In this paper we are discussed about the things of human working way and as well as the system works and this will help users to protect their data and its modifications. And different types of user interface things we are involved in this paper that is like icons, things, text boxes and dialog box etc. like this appearance will give some of the idea about the things what to implement. And as well as we work the instruction based and other researched information way to that software will work as a different manner like that of the performance of the process suggests the user to chose the options to make the selection and when ever software makes help to user to perform the actions or to make decisions based on their choices. Even though it's not performing the exact role of human to maintain the things for the process and it will provide the information related to the user matches, sometimes it may not fit the role perfectly to the users. So take this problem out from the system we are proposed a closed way to the user and to make the system to perform like the humans. Basically software's are maddened for those who use it and to implement those usages.

To implement this process we are chosen an attribute based data modification and this system will work to demolish the users and there modifications. Basically the system was developed by the users for the use of users and to work for their modification and there restoring of data implementation things, so here our mission is to

put all the thinking of human into a machine and to make it to be work as a human so that's why we are chosen this way of technology and by ding this we can overcome the failure of system and we can approach the reality of users. The main thing is trust it's a big word and it was existed in many of the ways just it make and it will based on the situations to make the things to perform. And for all the things email is the necessary thing for the work of users data and its modification of user performance of everything what he would do for this we are making an encryption and decryption of the data when ever user sending his information to other person and whenever he receives the information from the other people.

Here in this paper to address the user or to send the mail t anyone, we are implementing and using the S/MIME conditions and its rule based code solution for our application. In these to provide user friendliness and provide the digital signature of the user we are just using for an integrity purpose and to maintain the identity certificates of the sender and receiver both sides information whenever the user sends or receives any information from others and here whatever the signature was used by the user is also a public based content device and it will share that data with the remaining people whoever check and have the key of public data visibility.

The main purpose of MIME is trust based and it's divided into two types of data providing and modifying the things. That is whether user had know about the sender or not, that and we can see now in many organizations the sending of mail is very important and most related one to everyone in the companies now a days. In this main is the identity of the sender details r an organization when we send an email to anyone that we will use the identify signature that means to which organization we belong to like that an example:abc@xyz.com in this the identity is xyz.com that is the signature of that organization and it will make user to know that from where it came and to which it belongs to like that we can see as a real time example. Here in this we use PGP (Pretty Good Privacy) based emails and its related email properties which it will provide and support the users to send an email.

Fig: 1 A mock-up of GUI for presenting ABUSE attributes to user

II RELATED WORK

To address email protection and privacy concerns, so many organizations in the profitable, educational and central sectors have deployed S/MIME, a secure email typical that leverages X.509 Identity Certificates to

provide message non repudiation and reliability via digital signatures. In adding together, these signatures frequently include the sender's Identity Certificate, so contained of all information there in is available to the receiver. For example, assigned digitally message sent by his given name to the first writer would confirm that "vikas College" believes his email address to be "sudhahari@vikas.edu", a be "Sudha Rani H", and give a date after which Vikas no longer guarantees any of the above to be true. His public key contain signature would, at this time is can be used to validate the signature. Pretty Good Privacy (PGP) is another PKI-based email scheme which provides related properties, but with more random adoption.

III TRUST MANAGEMENT

At first quick look, combining a Trust Management (TM) system with S/MIME seems to be an suitable way to provide users with the extra appropriate information that we wish to provide. define *trust management* as "an approach to circulated authorization and access control, in which access control decision are based on *policy statements* made by multiple principals".

On the system dependent, digitally signed guidelines statements may be called *attributes* or *credentials*. These details must come from some party who is capable to mint them. Also, when a request is made, details must be bound to that request.TM systems normally absolutely assume some type of public key environment to give these properties. Upon reception by a TM locomotive, a request and its supporting credentials (maybe combined with other guidelines statements pulled from a local or remote credential depository) are then checked adjacent to the resource owner's trust guidelines (another set of guidelines statements). If the request satisfies the policy, authorization is granted.

Mitchell and Li defined a useful framework for discuss trust management systems, which "consists of three aspects: language, infrastructure and deduction." A TM language, according to, has a instrument for identifying principals, a syntax for specifying queries and guidelines statements, and a semantic relation that determines whether a query is true given a set of guidelines statements.

The deduction locomotive of this semantics in TM system implements, while the environment provides support of the transport, creation and maintenance of guidelines statements. Several of the TM systems discussed here leverage the "logical programming" model In all cases, parties involved in the TM system can make assertions about the attributes possessed by other parties, as was touched on above. Resource owners, then, can express policies not only in terms of principals, For instance, my policy could state that I trust the "Dartmouth" principal to grant a "student" credential to all currently enrolled students, and that anyone presenting such a credential issued by "Dartmouth" can access my "fun stuff to do in Hanover" files.

To implement a TM system atop email, we would need to

1) Choose a policy language,

2) Attach some credentials to digitally signed messages,

3) Build some kind of policy-checking engine into an email client

4) Convince users to specify policies that accurately capture their trust behavior when reading email.

Before we can intelligently discuss such an implementation, we must first survey existing TM systems in greater detail.

3.1 Logic-Based Approaches

TM systems which leverage the "logical programming" model either use some Prolog-like language to specify guidelines statements or design a new guidelines language that can be reduced to Prolog. We will demonstrate this portion of the space by evaluating, the Delegation Logic (DL), Trust Policy Language (TPL) and Role-based Trust-management (RT) framework , SD3.

DL, SD3 and RT are all TM systems based on Data log, a form of Prolog. They leverage the logical-programming model to prove that a request meets a given policy along with appropriate details. SD3 and RT both provide some facility for retrieving credentials from nonlocal repositories, while DL does not. To use these systems, a user would have to specify his trust policy in one of these logical programming languages.

TPL is an XML-based TM language that can be reduced to Prolog. Like SD3 and RT, it also provides some ability for remote credential retrieval. Guidelines generation is once again done by hand. Though XML is a more accessible language than Data log, a user would still need to be comfortable with programming to use TPL.

ATTRIBUTE-BASED, USEFULLY SECURE EMAIL

For the dual purposes of demonstrating our design philosophy

and helping users manage trust in secure email, we introduce the Attribute-Based, Usefully Secure Email (ABUSE) system. Rather than attempt to automatically make trust decisions for users, the system is designed to help them make more informed trust decisions about email that they receive. We do this by allowing users to create useful metadata about each other, access the store of data about themselves, and attach selected attributes to outgoing messages. Then we present this information to recipients in an understandable fashion. Our design goals are to

1) Enable users to bind appropriate trustworthy assertions about themselves to outgoing email,
2) Enable users to understand trustworthy assertions about senders of incoming email,
3) Avoid push-back from users without ABUSE-savvy clients,
4) minimize the administrative burden on everyone involved,
5) Avoid the need for an organization - wide Attribute Administrator.
6) Avoid limiting the attribute space (i.e. avoid predefining a set of attributes and relationships),
7) Leverage existing PKI and S/MIME infrastructure, and
8) Provide some support for attributes belonging to users at outside organizations.

In the case of the email "from the Dean" discussed earlier, the Dean's assistant could have cryptographically bound an attribute given to him by the Dean herself to his message stating his relationship to her. Recipients would then have been able understand the situation without having to keep track of who works for the Dean.

IV PROPOSED WORK

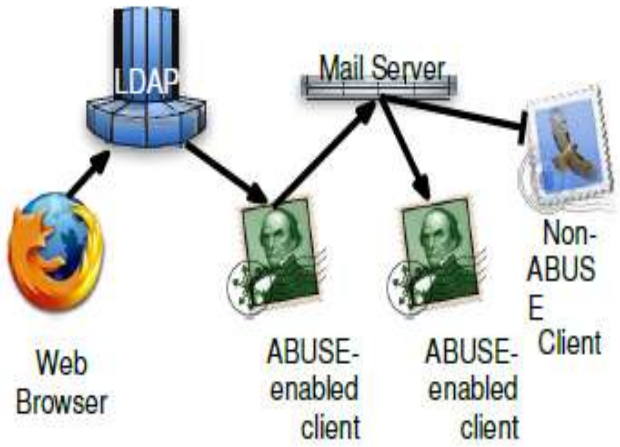
There are two portions of this project that are still pending: evaluation of a single-institution ABUSE system, and expanding ABUSE beyond the borders of one organization.

4.1 Evaluating ABUSE

In addition to making heavy use of user studies during the user interface design portions of building ABUSE, we also plan to deploy our prototype to a large community of users and collect feedback on their usage patterns. As mentioned earlier, we are building ABUSE on top of blitz mail, due to its prevalence at the College. We believe this will both allow us to conduct broader user studies, as well as providing us with clearer results, due to user familiarity with the basic user interface.

4.2 ABUSE across Organizations

To take ABUSE beyond a single institution, we must address both the issue of verifying attribute chains from foreign sources and also that of mapping unfamiliar attributes into a locally sensitive context. Bridge Certificate Authorities provide a method of joining disparate hierarchical X.509-based PKIs in a non-hierarchical way. By making both S/MIME digital signatures and ABUSE attributes on ABUSE-enabled emails "bridge aware", we can not only address the problem of verifying foreign attributes, but also begin exploring the possibilities and pitfalls of bridged PKIs. The Higher Education Bridge Certification Authority (HEBCA) has been set up



.

Fig: 2 a web browser and publish them to an LDAP. Using an ABUSE-enabled email client, senders of email can retrieve their attributes, attach them to outgoing email, and send them along.

In addition to the TM work discussed at length above, there are two other groups of related work: systems directly related to trust in email, and general work on usability in secure email.

4.3 Trust in Email

Both S/MIME, upon which ABUSE is built, and PGP/MIME can be considered work in this space. Digital signatures can, in many cases, provide users with enough contexts to decide whether or not to trust an incoming message. However, as we have discussed, there are cases that are not addressed by S/MIME and work done by other researchers has shown that

PGP/MIME clients are not usable by average users. To our knowledge, only Role-Based Messaging has attempted to address the same portion of the problem space as our work.

Role Based Messaging (RBM) is a system that creates role-based mail accounts. Users who have appropriate credentials (where "appropriate" is defined by policy on a per-role basis) can log into those accounts to read mail sent to that role and also to send signed and encrypted mail from that role. Mail may be encrypted to a role, not simply to a specific user. Role membership is controlled by a PERMIS back end, in which X.509 ACs are used to store role membership information. Policies can be added to messages to further control what recipients can do with them. A policy governs who can assign roles, though this could be set up to allow any user to grant roles to others. Also, these "role managers" can create new roles within their organization, but they will not be recognized by the system.

V CONCLUSION

We have introduced Attribute-Based, Usefully Secure Email (ABUSE), which we are building to exemplify our principle of leveraging humans in the design of secure systems. We chose to address secure email in particular due to several concerns about the expressiveness of email technology, including cases in which names lack specificity, properties mean different things in different contexts and properties not names effect trust decisions. ABUSE addresses these first two concerns by enabling users to delegate trustworthy attributes to each other, and then bind them to messages sent over email. Humans are leveraged at both ends of the process: humans hand out attributes to each other, and humans decide whether the attributes bound to a message are enough to build trust in the displayed content. The third concern is addressed by bridging across distinct PKIs and by mapping foreign attributes into a local context. Through development and testing of ABUSE, we hope to answer two long-term questions: whether issuing credentials in distributed way will actually work, and also whether users will actually understand these distributed credentials, enabling them to make more accurate trust judgments about incoming messages from unfamiliar senders.

REFERENCES

[1] A. Whitten and J. Tygar, "Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0." in *8th USENIX Security Symposium*, 1999.

[2] A. Whitten, "Making security usable," Ph.D. dissertation, Carnegie Mellon University School of Computer Science, 2003.

[3] S. Garfinkel, "Design principles and patterns for computer systems that are simultaneously secure and usable," Ph.D. dissertation, Massachusetts Institute of Technology, 2005.

[4] L. G. Zucker, "Production of trust: Institutional sources of economic structure, 1840–1920," in *Research in Organizational Behavior*. JAI Press Inc., 1986, vol. 8, pp. 53–111.

[5] Y.-H. Chu, J. Feigenbaum, B. LaMacchia, P. Resnick, and M. Strauss, "REFEREE: Trust management for Web applications," *Computer Networks and ISDN Systems*, vol. 29, no. 8–13, pp. 953–964, 1997.

[6] A. Schutz, "On multiple realities," in *Collected papers 1: the problem of social reality*, M. Natanson, Ed. The Hague: Martinus Nijhoff, 1962, pp. 207–259.

[7] H. Garfinkel, "A conception of and experiments with "trust" as a condition of stable concerted actions," in *Motivation and social interaction: Cognitive determinants*, O. Harvey, Ed. New York: Ronald Press, 1963, pp. 187–239.

AUTHORS PROFILE

Midasala Anusha Rani, pursuing M.Tech(CSE) from Vikas Group of Institutions, Nunna, Vijayawada. Affiliated to JNTU-Kakinada, A.P., India

Paparao Rapuri, working as an Asst. Professor of CSE department at Vikas Group of Institutions, Nunna, Vijayawada, Affiliated to JNTU-Kakinada, A.P., India

Betam Suresh, is working as an HOD, Department of Computer science Engineering at Vikas Group of Institutions, Nunna, Vijayawada, Affiliated to JNTU-Kakinada, A.P., India