

## AACT: Anonymous and Accountable communication topology for Wireless Mesh Networks

M.Narasimha Rao<sup>1</sup>, Shaik Jaffar<sup>2</sup>

<sup>1</sup>(M.Narasimha Rao currently doing his M.Tech in VLSI, Dept of E.C.E at Madina Engineering College, Kadapa, AP, India

<sup>2</sup>(Shaik Jaffar is currently working as Associate professor and HOD, Dept Of EIE, Madina Engineering College, Kadapa, AP, India

---

**ABSTRACT:** Recently, multihop wireless mesh networks (WMNs) have involved increasing attention and deployment as a low-cost move toward to give broadband Internet access at an urban scale. Security and privacy issues are of the major concern in pushing the success of WMNs for their broad deployment and for behind service-oriented applications. Regardless of the required, partial security research has been conducted towards privacy protection in WMNs. This motivates us to develop Anonymous and Accountable communication topology (AACT), a novel secure communication framework, tailored for WMNs. On one hand, AACT implements harsh user access control to cope with both free riders and spiteful users. On the other hand, AACT offers complicated user privacy protection beside both adversaries and a range of other network entities. AACT is accessible as a suite of authentication and key agreement protocols built upon our AACT. Our analysis demonstrates that AACT is resilient to a number of security and privacy related attacks. Additional methods were also discussed to further improve scheme efficiency.

**Keywords:** Wireless Mesh network, Anonymity, Onion ring cryptography, user security, user accountability

---

### I. INTRODUCTION

Wireless mesh networks (WMNs) have recently concerned rising attention and deployment as a promising low-cost approach to give last-mile high speed Internet access at urban scale [2], [3]. Typically, a WMN is a multihop layered wireless. The first layer consists of access points, which are high-speed wired Internet way in points. In the second layer, stationary mesh routers form a multihop spine via long-range high-speed wireless methods such as WiMAX [6]. The wireless spine connects to wired access points at some mesh routers through high speed wireless links. The third layer consists of a huge number of mobile network users. These network users access the network either by a direct wireless link or through a chain of other peer users to a nearby mesh router. WMNs correspond to a unique marriage of the ubiquitous coverage of large area cellular networks with the ease and the speed of the local area Wi-Fi networks [4]. The compensation of WMNs also contains low deployment costs, self-configuration and self maintenance, good scalability, high robustness, etc. [2]. Security and privacy issues are of mainly a concern in pushing the success of WMNs for their large deployment and for supporting service-oriented applications. Due to the essentially open and distributed nature of WMNs, it is necessary to enforce network access control to cope with both free riders and spiteful attackers. Dynamic access to WMNs should be subject to successful user authentication based on the correctly pre recognized trust among users and the network operator; otherwise, network access should be forbidden. On the other hand, it is also dangerous to provide good provisioning over user privacy as WMN communications regularly contain a vast amount of sensitive user details. The wireless standard, open network structural design, and be lacking in of physical protection over mesh routers render WMNs extremely vulnerable to different privacy-oriented attacks. These attacks range from passive eavesdropping to active message Phishing, interception, and modification, which could simply lead to the leakage of user information. Obviously, the wide deployment of WMNs can succeed only after users are assured for their capability to manage privacy risks and preserve their desired level of anonymity. Included with sensors and cameras, the WMN may also be used to gather information of interest. Perceptibly, all these communications include different kinds of sensitive user information like individual identities, actions, position information, fiscal information, transaction summaries, social/business connections, and so on. Once disclosed to the attackers, this information could negotiation any user's privacy, and when further associated together, can cause even more overwhelming consequences. Hence, securing user privacy is of paramount practical importance in WMNs. Moreover, for both billing purpose and avoiding the neglect of network resources, it is also necessary to exclude free riders and let only legitimate residents access WMNs.

Despite the need and significance, limited research has been conducted to address security mechanisms for anonymous and accountable communication in WMNs. This encourages us to propose AACT, a novel Anonymous, Accountable communication topology for WMNs. Our assistance is fourfold as follows:

*Security:* It achieves explicit mutual authentication and key organization among users and mesh routers and among users themselves. It, thus, excludes both illegitimate network access from free riders and spiteful users and Phishing attacks due to rogue mesh routers.

*Anonymity:* It concurrently enables independent anonymous authentication among users and mesh routers and bilateral anonymous authentication between any two users. It, thus, make sure user anonymity and privacy.

*Accountability:* It enables user accountability, at regulating user behaviors and defending WMNs from being harmed and attacked. Network communications can always be audited in the cases of disagreements and deceptions. It in addition allows adaptive user revocation so that spiteful users can be ejected.

*Sophisticated user privacy:* It allows users to disclose minimum information probabilistically while preserves accountability. In AACT, the user characteristics is a comprehensive information as network users as society members always interact with WMNs in different roles and contexts. Therefore, a dispute about a given communication session should only be attributed according to the role/context information about the user without disclosing his full identity information (unless necessary).

For the finest of our knowledge, AACT is the first attempt to set up an accountable security framework with a complicated privacy protection model tailored for WMNs. AACT also lays a solid background for designing other upper layer security and privacy solutions, e.g., Anonymous communication.

The rest of the paper is prearranged as follows: Section 2 is the introduction of the cryptographic knowledge entailed by AACT. Section 3 describes the problem formulation. Then, in Section 4, the details of AACT are described. We further analyze in Section 5 the security and privacy properties of AACT, as well as its presentation. Section 6 is about related work. Finally, we conclude the paper in Section 7.

## II. CRYPTOGRAPHY SPECIFICS

### 2.1 Onion ring strategy [31]

The Onion routing [31] achieves communication privacy by making communication ends as unable to link. An Onion routing network consists of a number of interconnected Onion routers (ORs); each OR has a pair of public/private keys. Each OR knows the topology of the Onion network as well as the public keys of other ORs. An end user that requires an anonymous communication will send a request to an OR that it trusts; this OR is known as the Onion Proxy (OP) for the user. The communication between an end user and its OP is protected from the adversaries. The OP determines a route that consists of a series of ORs and constructs an "Onion" using the public keys of the routers en route. The "Onion" is constructed in a way such that the most inner part is the message to the intended destination. The message is wrapped, i.e., encrypted using the public keys of the ORs in the route, in the same order as the ORs appears in the route. Once an OR receives the Onioned message, it uses its private key to peel, i.e., decrypt, the "Onion", to obtain the information such as the next hop and the session key. It then forwards the rest of the "Onion" to the next hop. This process is repeated until the "Onion" reaches the last OR, which peels the last layer of the "Onion" and obtain the exit information, i.e., the destination.

For example, if the private route is  $R_1 \rightarrow R_2 \dots \rightarrow R_n$ , where  $R_i$  is the  $i^{th}$  OR, and the last router  $R_n$  will connect to the exit funnel of the 'ORs', which will further communicate with the address requested by the session initiator; the message flow and the "Onion"(s) received at each router in the route are as follows:

$$E_{k_p R_1} \left( R_2, k_1, E_{k_p R_2} \left( \dots E_{k_p R_n} (k_n, \text{exit}) \dots \right) \right) \quad (1)$$

$$\rightarrow E_{k_p R_2} \left( \dots E_{k_p R_n} (k_n, \text{exit}) \dots \right) \dots \rightarrow E_{k_p R_n} (k_n, \text{exit}).$$

' $k_p R_i$ ' and ' $k_i$ ' are the public key and assigned session key for the  $i^{th}$  router. After the route is built up, session keys are used for constructing "Onion"s, and anonymous circuit ID (ACI) is used for routing. For the reverse path, data packet was encrypted with the session keys. The OP receives the "Onion" in the reverse path and peels it using the session keys assigned to the ORs, and sends the raw data to the end user.

For an Onion route, only the proxy knows the first and the last router. Any OR in the route only knows its previous hop and next hop. For both outside attackers and inside attackers (i.e., compromised ORs), as encryption or decryption is processed at every OR, it is difficult to link any two links (a link is a connection between two Onion routers) to the same route. Therefore, for a communication going through the Onion routers, the entry OR and exit OR are unable to link. When there are a large number of connections, it is difficult to find out the two communication ends for any connection that applies Onion routing.

To avoid that the change of "Onion" size in the route built-up stage may give adversary hints about routing information, an "Onion" has to be padded when part of its information has been read and removed, so that the length of the "Onion" keeps the same and it is difficult for an inside observer to obtain the routing information. Refer to [10], if the maximum number of Onion routers in a private route is N, the OP will construct a message of N "Onions" to build an Onion route. When a router receives the "Onion"s, it decrypts

all the "Onion"s and obtain the routing information only from the first one. It then adds a dummy packet at the end, and forward the "Onion"s further.

For example, if the maximum hop count  $N$  is 5, and the private route is as  $OP \rightarrow R_1 \rightarrow R_2 \rightarrow R_3$ , the message flow and the messages sent at each router are as follows:

$$OP \rightarrow R_1 : E_{k_p R_1} (R_2, k_1), E_{k_p R_1} (E_{k_p R_2} (R_3, k_2)), (2)$$

$$E_{k_p R_1} (E_{k_p R_2} (E_{k_p R_3} (exit, k_3))),$$

dummy,dummy

$$R_1 \otimes R_2 : E_{k_p R_2} (R_3, k_2), E_{k_p R_2} (E_{k_p R_3} (exit, k_3)),$$

dummy;dummy;dummy

$$R_2 \otimes R_3 : E_{k_p R_3} (exit, k_3),$$

dummy;dummy;dummy;dummy

## 2.2 Group Signature

Group signature schemes are a comparatively recent cryptographic concept introduced by Chaum and van Heyst in 1991 [9]. A group signature scheme is a technique for allowing a member of a group to sign a message on behalf of the group. In contrast to ordinary signatures, it gives anonymity to the signer, i.e., A verifier can only tell that a member of any group signed. However, in outstanding cases, such as a legal argument, any group signature can be "opened" by a designated group manager to make known clearly the identity of the signature's originator. Some group signature schemes support revocation, where group membership can be disabled. One of the most recent group signature schemes is the one proposed by Boneh and Shacham [8], which has an extremely short signature size that is similar to that of an RSA-1024 signature [10]. This scheme is based on the following two problems that are believed to be hard. Let  $G_1, G_2, g_1, g_2$  as defined above.

q-Strong Diffie-Hellman problem: The q-SDH problem in  $(G_1, G_2)$  is defined as follows: given a  $(q + 2)$ -tuple  $(g_1, g_2, g_2^\gamma, g_2^{(\gamma^2)}, \dots, g_2^{(\gamma^q)})$  as input, output a pair  $(g_1^{1/(\gamma+x)}, x)$ , where  $x \in Z_p^*$ .

Decision linear on  $G_1$ : Given random generators  $u, v, h$  of  $G_1$  and  $u^a, v^b, h^c \in G_1$  as input, output yes if  $a + b = c$ , and no, otherwise.

## III. PROBLEM FORMULATION AND THE SCHEME OVERVIEW

### 3.1 Network Architecture and System Assumptions

In the three-layer architecture consider a metropolitan-scale WMN under the manage of a network operator (NO). The network operator deploys a number of APs and mesh routers and forms a well-connected WMN that covers the whole area of a city and gives network services to network users, i.e., the citizens. Network users, on the other hand, subscribe to the network operator for the services and use their mobile clients to freely access the network from wherever within the city. The membership of network users may be 1) completed/renewed according to user- operator agreement in an episodic manner or 2) dynamically revoked by NO in case of argument/attack.

Similar to [4], [11], we assume that the downlink from a mesh router to all users within its reporting is one hop. However, the uplink from a user to a mesh router may be one or several hops. That is, a network user wants to transmit packets in multiple hops to a mesh router beyond his direct transmission range. In this case, network users cooperate with each other on relaying the packets to mesh routers. We further assume that all the network traffic has to go through a mesh router except the communication between two direct neighboring users. We assume so as it is probable that communications to and from a mesh router will constitute the majority of traffic in a WMN [12]. Moreover, this assumption would considerably reduce the routing complexity from the users' point of view as mesh routers will take the responsibility.

We assume that NO can always communicate with mesh routers through pre recognized secure channels, and so are mesh themselves. The WMN is assumed to be deployed with redundancy in mind so that revocation of individual mesh routers will not affect network connection. We assume the survival of an offline trusted third party (TTP), which is trusted for not disclosing the information it stores. TTP is essential only during the system setup. We further assume that there is a secure channel among TTP and each network user.

### 3.2 Threat Model and Security Requirements

Due to the open medium and spatially distributed nature, WMNs are susceptible to both passive and active attacks. The passive attacks include eavesdropping, while active attacks range from message relaying, bogus message injection, Phishing, active imitation to mesh router cooperation. Hence, for a practical threat model, we consider an adversary that is able to eavesdrop all network communications, as well as inject random fake messages. In addition, the adversary can compromise and control a small number of users and mesh routers subject to his option; it may also set up rogue mesh routers to phish user accesses. The purposes of the adversary contain 1) illegal and unaccountable network access, 2) the privacy of genuine network users, and 3) denial-of-service (DoS) attacks against service accessibility.

In light of the above threat model, the following security requirements are necessary to make sure that a WMN functions correctly and strongly as purposed.

- *User-router shared authentication and key agreement:* A mesh router and a user should equally authenticate each other to stop both unauthorized network access and Phishing attacks. The user and the mesh router should also set up a shared pairwise symmetric key for session authentication and message encryption.
- *User-user mutual authentication and key agreement:* Users should also authenticate each other by cooperation in observing to message relaying and routing. Moreover, symmetric keys should be established and efficiently maintained to give session authentication and message encryption over the equivalent traffic.
- *Sophisticated user privacy protection:* The privacy of users should be well secluded, and we distinguish user privacy against dissimilar entities such as the adversary, *NO*, and the law authority, as will be complicated in the next section.
- *User accountability:* In the cases of attacks and argument, the responsible users and/or user groups should be capable to be audited and pinpointed. On the other hand, no innocent users can be framed for disputes/attacks they are not concerned with.
- *Membership maintenance:* The network should be capable to handle membership dynamics with membership revocation, renewing, and addition.
- *DoS resilience:* The WMN should maintain service accessibility despite of DoS attacks.

## IV. AACT: ANONYMOUS, ACCOUNTABLE COMMUNICATION TOPOLOGY

When designing AACT, we find that none of the obtainable anonymous accountable cryptographic primitives, such as blind signature and group signature schemes, suits our purpose given the security and privacy requirements discussed above. Blind signature and group signature schemes can only give binding anonymously, while AACT demands user accountability, and hence, revocable anonymity. Existing group signature schemes do give revocable secrecy, but cannot support complicated user privacy. This inspiring us to tailor a group signature scheme by combining with onion ring strategy to convene all the necessities. AACT is then built on this onion ring based group signature difference by further integrating it into the authentication and key agreement protocol design.

### 4.1 AACT Key Management

The following setup operations are performed in an offline manner by all the entities in AACT, namely *NO*, a *TTP*, mesh routers, network users, and user group managers. AACT works under bilinear groups  $(G_1, G_2)$  with isomorphism  $\psi$  and respective generators  $g_1$  and  $g_2$ , as in Section 2.1. AACT also employs hash functions  $H_0$  and  $H$ , with respective ranges  $G_2^2$  and  $Z_p$ . The notation below mainly follows [8].

*NO* is responsible for the key generation operation. Specifically, *NO* proceeds as follows:

1. Select a generator  $g_2$  in  $G_2$  uniformly at random and set  $g_1 \leftarrow \psi(g_2)$ . Select  $\gamma \xleftarrow{R} Z_p^*$  and set  $w = g_2^\gamma$ .
2. Select  $grp_i \xleftarrow{R} Z_p^*$

For a registered user group  $I$ .

3. Using  $\gamma$ , generate an SDH tuple  $(A_{i,j}, grp_i, x_j)$  by selecting  $x_j \xleftarrow{R} Z_p^*$  such that  $\gamma + grp_i + x_j \neq 0$ , and setting  $A_{ij} \leftarrow g_1^{1/(\gamma + grp_i + x_j)}$ .
4. Repeat Step 3 for a prearranged number of times that are mutually agreed by *NO* and the user group manager  $GM_i$ .
5. Send  $GM_i\{[i, j], grp_i, x_j\}|\forall j\}$  via a secure channel.
6. Repeat Steps 2, 3, and 4 for every user group.

7. Send  $TTP: GM_i\{[i, j], A_{i,j} \otimes x_j\}|\forall i, j\}$  via a secure channel, where  $\otimes$  denotes bitwise exclusive OR operation.

The above operation generates the group public key  $gpk$  and a number of private keys  $gsk$ :

$$\begin{cases} gpk = (g_1, g_2, w) \\ \{gsk[i, j] = (A_{i,j}, grp_i, x_j)|\forall i, j\}. \end{cases}$$

Furthermore,  $NO$  obtains a set of revocation tokens,  $grt$ , with  $grt[i,j] = A_{i,j}$  and also keep the mapping among group id  $i$  and  $grp_i$  for all user groups. Note that  $\gamma$  is the system secret only known to  $NO$ . For the purpose of non denial,  $NO$  signs on Steps 5 and 7 under a standard digital signature scheme, such as ECDSA [13]. In AACT, we suppose that ECDSA-160 is used. For the same purpose,  $GM_i$  and  $TTP$  also sign on these messages upon receiving and send the resulted signature back to  $NO$ .

Additionally,  $NO$  prepares every mesh router  $MR_k$  a public/private key pair, denoted by  $(RPK_k, RSK_k)$ . Each mesh router also gets an accompanied public key

A certificate signed by  $NO$  to prove key authenticity. The signing key pair of  $NO$  is denoted by  $(NPK, NSK)$ . The certificate has the following fields at the minimum:

$$Cert_k = \{MR_k, RPK_k, ExpT, Sig_{NSK}\},$$

Where  $ExpT$  is the expiration time and  $Sig$ , denotes an ECDSA-160 signature signed on a given message using a private key •.

Before accessing the WMN, a network user has to validate himself to his fit in user groups. For each such user group  $i$ , a network user  $uid_j$  is assigned a casual group private key as follows:

1.  $GM_i$  sends  $uid_j(|i, j|, grp_i, x_j)$  as well as the related system parameters.
2.  $GM_i$  requests  $TTP$  to send  $uid_j(|i, j|, A_{i,j} \otimes x_j)$  by providing the index  $[i, j]$ .
3.  $uid_j$  assembles his group private key as  $gsk[i, j] = (A_{i,j}, grp_i, x_j)$ .

Note that in our setting,

- $GM_i$  only keeps the mapping of  $(uid_j(|i, j|, grp_i, x_j))$  but has no knowledge of the corresponding  $A_{i,j}$ .
- $NO$  only knows the mapping of  $(GM_i, gsk[i, j])$  but has no knowledge about to whom  $gsk [i, j]$  is assigned.
- $TTP$  has the mapping of  $(uid_j(A_{i,j} \otimes x_j, grp_i))$  as it sends  $uid_j$  this information through a safe channel among the two upon the request from  $GM_i$ . But  $TTP$  has no knowledge of the corresponding  $x_j$  or  $A_{i,j}$ .

Here, we use  $uid_j$  the user's necessary attribute information. For the purpose of non repudiation,  $uid_j$  signs on the messages it receives from  $GM_i$  and  $TTP$  under ECDSA-160, and sends back  $GM_i$  the equivalent signature.

#### 4.2 User-Router Mutual Authentication and Key Agreement

To access the WMN, a network user follows the user-router common authentication and key agreement protocol as particular below, when a mesh router is within his direct communication range.

1. The mesh router  $MR_k$  first picks a random nonce  $r_R RZ_p^*$  and a random generator  $g$  in  $G_1$  and then computes  $g^{r_R}$ .  $MR_k$  further signs on  $g g^{r_R}$ , and the current time stamp  $ts_1$ , using ECDSA-160.  $MR_k$  then broadcasts

$$g, g^{r_R} ts_1, Sig_{RSK_k}, Cert_k, CRL, URL \quad (M.1)$$

As part of *beacon message* that is periodically broadcast to declare service existence. Here,  $CRL$  and  $URL$  denote the mesh router certificate revocation list and the user revocation list, respectively. Specifically,  $URL$  contains a set of revocation tokens that corresponds to the revoked group private keys, which is a subset of  $grt$ . Both  $CRL$  and  $URL$  are signed by  $NO$ .

Upon receipt of (M.1), a network user  $uid_j$  proceeds as follows:

Check the time stamp  $ts_1$  to prevent replay attack. Examine  $Cert_k$  to confirm public key authenticity and the certificate expiration time; examine  $CRL$  and see if  $Cert_k$  has been revoked by applying NPK. Further verify the authenticity of  $Sig_{RSK}$  by applying  $RPK_k$ .

Upon positive check results,  $uid_j$  believes that  $MR_k$  is legitimate and does the following:

Pick two random nonce  $r, r_j \in \mathbb{Z}_p^*$ , compute  $g^{r_j}$ , and prepare the current timestamp  $ts_2$ . Further get two generators  $(\hat{u}, \hat{v})$  in  $G_2$  from  $H_0$  as

$$(\hat{u}, \hat{v}) \leftarrow H_0(gpk, g^{r_j}, g^{r_s} ts_2, r) \in G_2^2, \quad (1)$$

And compute their images in  $G_1$ :  $u \leftarrow \psi(\hat{u})$  and  $v \leftarrow \psi(\hat{v})$ .

Compute  $T_1 \leftarrow u^\alpha$  and  $T_2 \leftarrow A_{i,j} v^\alpha$  by selecting an exponent  $\alpha \in \mathbb{Z}_p^*$ . Set  $\delta \leftarrow (grp_i + x_j)\alpha \in \mathbb{Z}_p$ . Pick blinding values  $r_\alpha, r_x$ , and  $r_\delta \in \mathbb{Z}_p$ .

Compute helper values  $R_1, R_2$ , and  $R_3$ :

$R_1 \leftarrow u^{r_\alpha}, R_2 \leftarrow e(T_2, g_2)^{r_x} \cdot e(v, w)^{-r_\alpha} \cdot e(v, g_2)^{-r_\delta}$ , and  $R_3 \leftarrow T_1^{r_x} \cdot u^{-r_\alpha}$ . Compute a challenge value  $c \in \mathbb{Z}_p$  using  $H$ :

$$c \leftarrow H(gpk, g^{r_j}, g^{r_s}, ts_2, r, T_1, T_2, R_1, R_2, R_3) \in \mathbb{Z}_p.$$

Compute  $s_\alpha = r_\alpha + c\alpha, s_x = r_x + c(grp_i + x_j)$  and  $s_\delta = r_\delta + c\delta \in \mathbb{Z}_p$ . Obtain the group signature on  $\{g^{r_j}, g^{r_s}, ts_2\}$  as

$$SIG_{gsk[i,j]} \leftarrow (r, T_1, T_2, c, s_\alpha, s_x, s_\delta).$$

Compute the shared symmetric key with  $MR_k$ :

$$K_{k,j} = (g^{r_s})^{r_j}.$$

Unicast back to  $MR_k$

$$g^{r_j}, g^{r_s}, ts_2, SIG_{gsk[i,j]}. \quad (M.2)$$

Upon receipt of (M.2),  $MR_k$  carries out the following to authenticate  $uid_j$ :

Check  $g^{r_s}$  and  $ts_2$  make sure the freshness of (M.2).

Check that  $SIG_{gsk[i,j]}$  is a valid signature by applying the group public key  $gpk$  as follows:

Compute  $\hat{u}$  and  $\hat{v}$  using (1), and their images

$u$  and  $v$  in  $G_1$ :  $u \leftarrow \psi(\hat{u})$  and  $v \leftarrow \psi(\hat{v})$ .

Retrieve  $R_1, R_2$  and  $R_3$  as:

$$\tilde{R}_1 \leftarrow u^{s_\alpha} / T_1^c$$

$$\tilde{R}_2 \leftarrow e(T_2, g_2)^{s_x} \cdot e(v, w)^{-s_\delta} \cdot (e(T_2, w) / e(g_1, g_2))^c,$$

And  $\tilde{R}_3 \leftarrow T_1^{s_x} \cdot u^{-s_\delta}$ .

Check that the challenge  $c$  is correct:

$$c \stackrel{?}{=} H(gpk, g^{r_j}, g^{r_s}, ts_2, r, T_1, T_2, \tilde{R}_1, \tilde{R}_2, \tilde{R}_3). \quad (2)$$

For each revocation token  $A \in URL$ , check whether  $A$  is encoded in  $(T_1, T_2)$  by checking if

$$e(T_2 / A, \hat{u}) \stackrel{?}{=} e(T_1, \hat{v}). \quad (3)$$

If no revocation token of the URL is encoded in  $(T_1, T_2)$ , then the signer of  $SIG_{gsk[i,j]}$  has not been revoked.

If all the above checks succeed,  $MR_k$  is now assured that the current user is a legitimate network user, although  $MR_k$  does not know which particular user this is. Note that  $uid_j$  is never disclosed or transmitted during protocol execution.

a.  $MR_k$  Further computes the shared symmetric key as  $K_{k,j} = (g^{r_j})^{r_k}$  and sends back  $uid_j$  :

$$g^{r_j}, g^{r_k}, E_{K_{k,j}}(MR_k, g^{r_j}, g^{r_k}), \quad (M.3)$$

Where E denotes the symmetric encryption of the given message within the brackets using key •.

The above protocol allows explicit mutual authentication among a mesh router and a genuine network user; it also enables unilateral anonymous authentication for the network user. Upon successful completion of the protocol, the mesh router and the user also create a shared symmetric key used for the succeeding communication session. And this session is uniquely identified through  $(g^{r_j}, g^{r_k})$ .

**Remarks**

Equation (2) holds because

$$\begin{aligned} \tilde{R}_1 &= u^{s_\alpha} / T_1^c = u^{r_\alpha + c\alpha} / (u^\alpha) = u^\alpha = R_1. \\ \tilde{R}_2 &= e(T_2, g_2)^{s_\alpha} \cdot e(v, w)^{-s_\alpha} \cdot e(v, g_2)^{s_\alpha} \cdot \left( \frac{e(T_2, w)}{e(g_1, g_2)} \right)^c = (e(T_2, g_2)^{r_\alpha} \cdot e(v, w)^{-r_\alpha} \cdot e(v, g_2)^{s_\alpha}) \cdot (e(T_2, g_2)^{gr_{p_i} + x_j} \cdot e(v, w)^{-\alpha} \cdot e(v, g_2)^{-(gr_{p_i} + x_j)\alpha}) \cdot \frac{e(T_2, w)^c}{e(g_1, g_2)^c} \\ &= R_2 \cdot \left( \frac{e(T_2 v^{-\alpha}, w g_2^{gr_{p_i} + x_j})}{e(g_1, g_2)} \right)^c = R_2 \cdot \left( \frac{e(A_{i,j}, w g_2^{gr_{p_i} + x_j})}{e(g_1, g_2)} \right)^c = R_2 \cdot \left( \frac{e(g_1, g_2)}{e(g_1, g_2)} \right)^c = R_2. \\ \tilde{R}_3 &= T_1^{s_z} u^{-s_\delta} = (u^\alpha)^{r_z + c(gr_{p_i} + x_j)} \cdot u^{-r_\delta - c\alpha(gr_{p_i} + x_j)} = (u^\alpha)^{r_z} \cdot u^{-r_\delta} = T_1^{r_z} \cdot u^{r_\delta} = R_3. \end{aligned}$$

Equation (3) holds when there is an element A of URL encoded in  $(T_1, T_2)$  because of the following.

We know that  $\psi : G_2 \rightarrow G_1$  is an isomorphism such that  $\psi(g_2) = g_1$ . According to the definition of isomorphism, we have  $\psi(PQ) = \psi(P)\psi(Q)$  for any P, Q  $\in G_2$ . Using this property and mathematical induction, it is easy to know the following fact: For any natural number  $m \in N, \psi(g_2^m) = g_1^m$ .

Hence, if a group private key  $(A_{i,j}, gr_{p_i}, x_j)$  with  $A_{i,j} \in URL$  signed the group signature  $\sigma$ . For simplicity, let  $\hat{u} = g_2^a$  and  $\hat{v} = g_2^b$  for some integers a and b. On one hand,

$$e(T_2 / A_{i,j}, \hat{u}) = e(A_{i,j} v^\alpha / A_{i,j}, \hat{u}) = e(v^\alpha, \hat{u}) = e((\psi(\hat{v}))^\alpha, \hat{u}) = e((\psi(g_2^b))^\alpha, \hat{u}) = e((g_1^b)^\alpha, g_2^a) = e(g_1, g_2)^{ab\alpha}.$$

On the other hand,

$$e(T_1, \hat{v}) = e(u^\alpha, \hat{v}) = e((\psi(\hat{u}))^\alpha, \hat{v}) = e((\psi(g_2^a))^\alpha, \hat{v}) = e((g_1^a)^\alpha, g_2^b) = e(g_1, g_2)^{ab\alpha}.$$

Therefore,  $e(T_2 / A_{i,j}, \hat{u}) = e(T_1, \hat{v})$ .

**4.3 User-User Mutual Authentication and Key Agreement In AACT**

Adjacent genuine network users may help to relay each other’s traffic. To this end, two network users within each other’s direct communication range first authenticate each other and create shared secret pairwise key as follows:

1.  $uid_j$  picks a random nonce  $r_j RZ_p^*$  and computes where  $g^{r_j}$  is obtained from the inspirational messages broadcasted by the current service mesh router.  $uid_j$  further signs on  $g, g^{r_j}$ , and current time stamp  $ts_1$ , using his group private key  $gsk[i,j]$  following Steps 2b(i) to 2b(iv), as in Section 4.2.  $uid_j$  Then locally broadcasts

$$g, g^{r_j}, ts_1, SIG_{gsk[i,j]}. \quad (M.1)$$

2. Upon receipt of  $(\tilde{M}.1)$ ,  $uid_i$  checks the time stamp and verifies the authenticity of  $SIG_{gsk[i,j]}$  by applying the group key  $gpk$  following Step 3b, as in Section 4.2.  $uid_i$  further checks if the signature is generated from a revoked group private key following Step 3c, as in Section 4.2. Note that URL can always be obtained from the *beacon messages*.

If all checks succeed,  $uid_i$  is assured that the current user it communicates with is legitimate.  $uid_i$  proceeds to pick a random nonce  $r_i RZ_p^*$  and computes  $g^{r_i}$ .  $uid_i$  further signs on  $g^{r_j}, g^{r_i}$ , and current time stamp  $ts_2$ ,

using an appropriate group private key  $gsk[t, I]$  of his.  $uid_I$  also computes the shared pairwise session key as  $K_{r_j, r_i} = (g^{r_j})^{r_i}$ . then replies  $uid_I$

$$g^{r_j}, g^{r_i}, ts_2, SIG_{gsk[t, I]} \quad (\tilde{M}.2)$$

3. Upon receipt of  $(\tilde{M}.2)$ ,  $uid_j$  first delay window.  $uid_j$  checks whether  $ts_2 - ts_1$  is within the acceptable delay window.  $uid_j$  also examines  $SIG_{gsk[i, j]}$  and  $URL$  as  $uid_j$  did above. If all checks succeed,  $uid_j$  is also assured that its communicating counterpart is legitimate.  $uid_j$  Computes the shared pairwise session key as  $K_{r_j, r_i} = (g^{r_i})^{r_j}$ .  $uid_j$  Finally replies  $uid_I$

$$g^{r_j}, g^{r_i}, E_{K_{r_j, r_i}} = (g^{r_i}, g^{r_j}, ts_1, ts_2). \quad (\tilde{M}.3)$$

Upon receipt of  $(\tilde{M}.3)$  and successful decryption of  $E_{K_{r_j, r_i}} = (g^{r_i}, g^{r_j}, ts_1, ts_2)$ .  $uid_I$  is assured that  $uid_j$  has successfully completed the authentication protocol and recognized the shared key for their subsequent communication session, which is uniquely identified through  $(g^{r_j}, g^{r_i})$ .

This design of AACT protects user privacy in a complicated manner, while still maintaining user accountability. *User Anonymity against the Adversary, the User Groups, and TTP*

In AACT, a user only authenticates himself as a genuine service subscriber without disclosing any of his identifying information by make use of the group signature method. Neither the adversary nor the user group managers can tell which meticulous user generates a given signature. The adversary, even by compromising mesh routers and other network users, that is, knowing a number of group private keys in addition to the group public key, still cannot infer any information concerning the meticulous group private key used for signature generation. This is due to the rigidity of the underlying q-SDH problem, where q is a 1,020-bit prime number. Due to the similar reason, neither a user group manager can distinguish whether or not one of his group members has signed a meticulous signature as he has no knowledge of the corresponding  $A_{i, j}$ s nor can he compute them. The same termination also holds for TTP as TTP can compute neither  $x_j$  nor  $A_{i, j}$  given  $A_{i, j} \otimes x_j$ . Furthermore, each data session in AACT is identified only through pairs of fresh random numbers, which again discloses nothing concerning the user identity information. In addition, AACT needs a network user to refresh session identifiers and the shared symmetric keys for each different session. This further eliminates the ability to link among any two sessions initiated by the same network user. We note that even with the help of compromised mesh routers and other network users, the opponent still cannot judge whether two communication sessions are from the similar user. This is because, basically, none of them can tell whether two signatures are from the same user, given q-SDH problem and decision linear on G problem are hard.

*User Privacy against NO and User Accountability:* Since NO knows  $grt$ , it can always tell which  $gsk[i, j]$  produces a given signature. However, NO has no knowledge about to whom  $gsk[i, j]$  is assigned as AACT allows a late compulsory among group private keys and network users. Furthermore, it is user group managers' sole responsibility to assign group private keys to every network user without any participation of NO. Therefore, NO could only map  $gsk[i, j]$  to the user group  $i$  based on  $grp_i$ . Because no other entities except NO and the key holder himself has the knowledge of the corresponding  $A_{i, j}$ , and can therefore, generate the given signature, the key holder must be a member of the user group  $i$ . This audit result serves us both necessities. On one hand, the result only discloses partial nonessential attribute information of the user and still protects user privacy to an extent. On the other hand, the result is adequate for user accountability purposes for NO.

When NO (on behalf of mesh routers) finds a certain communication session disputable or suspicion, it conducts the following protocol to audit the responsible entity:

1. Given the link and the session identifier, find the equivalent authentication session message  $(M.2) = g^{r_j}, g^{r_i}, ts_2, SIG_{gsk[i, j]}$  from the network log file.
2. For each revocation token  $A_{i, j} \in grt$ , check whether  $e(T_2 / A_{i, j}, \hat{u}) \stackrel{?}{=} e(T_1, \hat{v})$ . Output the first element  $A_{i, j} \in grt$  such that  $e(T_2 / A_{i, j}, \hat{u}) \stackrel{?}{=} e(T_1, \hat{v})$ .
3. For the found revocation token  $A_{i, j}$ , output the corresponding mapping between  $A_{i, j}$  and  $grp_i$ .



Since  $grp_i$  maps to a particular user group  $i$ , now a responsible entity has been found from the perspective of NO.

From the user's perspective, only part of his unneeded attribute information is disclosed from the audit. But such unneeded attribute information will not reveal his necessary attribute information. For example, the above audit may find that the dependable user is a member of Company XYZ but cannot reveal any other information about the user. Yet NO still has adequate proof to prove to Company XYZ that one of his members violates certain network access rule so that Company XYZ should take the corresponding responsibility specified in their service contribution agreement.

*Revocable User Anonymity against Law Authority:* When law authority decides to track the meticulous attacker that is responsible for a certain communication session, the following procedure is taken: NO reports to the law authority  $(A_{i,j}, grp_i)$  by executing the above protocol against the session in audit.  $(A_{i,j}, grp_i)$  is then further forwarded to  $GM_i$ .  $GM_i$  Checks its local record, finds out the mapping between  $(grp_i, andx_i)$ , and hence, the corresponding user uniqueness information  $uid_j$ , to whom  $gsk[i,j]$  is assigned during the system setup.  $GM_i$  then replies  $uid_j$  to the law authority. At this point, law authority and only law authority get to know about which particular user is conscientious for the communication session in the audit. We point out that this tracing procedure has the non denial property because 1)  $GM_i$  signed on all  $gsk$ s that are assigned from NO as the proof of receipt; 2)  $uid_j$  also signed on the messages when obtaining  $gsk[i, j]$  from  $GM_i$  and TTP as the proof of receipt. AACT also not able to frame because no one else knows  $gsk[i, j]$  except NO and  $uid_j$  or is able to forge a signature on behalf of  $uid_j$ .

## V. PERFORMANCE ANALYSIS OF AACT

### 5.1 System Security Analysis

As its basic security functionality, AACT enforces network access control. Hence, we are the majority concerned with the following three different types of attacks, i.e., Bogus data injection attacks, data Phishing attacks, and DoS attacks.

*Bogus data injection attacks:* In such attacks, the opponent needs to inject bogus data to the WMN aimed at using the network service for free. The sources of the bogus data could be outsiders, revoked users, or revoked mesh routers.

However, such bogus data traffic will be all instantly filtered in AACT. First, with respect to outsiders, they do not know any group private keys. Thus, they cannot produce correct message signatures, when attempting to initialize a communication session with NO and/or other network users. They also cannot bypass the authentication procedure and straightly send out bogus data to others as they do not possess any shared symmetric session keys with them, and thus, cannot produce correct MACs. Then, regarding revoked users, there are two situations: 1) they do not have any group private key at present in use due to group public key update or 2) the corresponding group private keys owned by them are previously revoked and are published in the URL in beacon messages. Obviously, the revoked users cannot increase network access in neither cases. Finally, for revoking mesh routers, they are no longer valid members of the WMN. By checking CRL, no genuine mesh routers will accept/relay data traffic from revoking mesh routers. Also, since the downlink from a mesh router to its service range is only one hop, network users never require to and will not relay data traffic for mesh routers in AACT.

*Data phishing attacks:* In such attacks, the opponent may set up bogus mesh routers and try to phish user connections to such routers. In this way, the opponent could control network connection and analyze users' data traffic for their benefits. The Phishing mesh routers can be either completely new mesh routers or revoked mesh routers both at the adversary's control. In the former case, the mesh router will not be capable to authenticate itself to the network user. Therefore, no network user will set up any session with such a mesh router. Even if the mesh router could stop the network traffic among a network user and a genuine mesh router, it will not be able to decrypt the message and obtain any useful information. In the latter case, a newly revoked mesh router, however, will possibly be capable to authenticate itself to a network user, if such a user does not possess the most recent version of CRL. The network user may be deceived in this case but only for up to (inverse of the update frequency—(current time—last periodically update time)) time period. This is because the revoked mesh router will not be capable to give a legal CRL update at the next periodical CRL update time point.

*DoS attacks:* In such attacks, the opponent may flood a huge number of illegal access request messages to mesh routers. The purpose is to exhaust their resources and render them less capable of serving legitimate users. In AACT, for every access request message (M.2), the corresponding mesh router has to confirm a group

signature and check the validity of the signer. Both operations involve costly pairing operations, which, hence, can simply be exploited by the opponent. To deal with this issue, we assume the same client- puzzle approach as adopted in [18]. The idea of this approach is as follows: When there is no proof of the attack, a mesh router process (M.2) usually. But, when under a suspected DoS attack, the mesh router will attach a cryptographic puzzle to every (M.1) and need the solution to the puzzle be attached to every (M.2). The mesh router commits resources to process (M.2) only when the solution is correct. Typically, solving a client puzzle needs a brute-force search in the solution space, while the solution conformation is trivial [18].

Therefore, the opponent must have abundant resources to be capable to promptly compute a huge efficient number of puzzle solutions in line with his sending rate of bogus access request (M.2). In contrast, although puzzles slightly increase genuine users' computational load when the mesh router is under attack, they are still able to obtain network accesses despite the subsistence of the attack. We refer the readers to [18] for the complete design.

## 5.2 User Privacy and Accountability Analysis

AACT protects user privacy in a complicated manner, while still maintain user's responsibility. First, AACT enables user anonymity against the opponent, the user group managers, and TTP. In AACT, a network user only authenticates himself as a genuine service subscriber without disclosing any of his identity information by using the group signature method. Neither the opponent nor the user group managers can tell which meticulous user generates a given signature. The adversary, even by compromising mesh routers and other network users, that is, knowing a number of group private keys in addition to the group public key, still cannot deduce any information about the particular group private key used for signature generation. This is due to the rigidity of the underlying q-SDH problem, where q is a 1,020-bit prime number. Due to the same reason, a user group manager also cannot differentiate whether or not one of his group members has signed a particular signature as he has no knowledge of the corresponding  $A_{i,j}$ s nor can he compute them. The same finish also holds for TTP as TTP can compute neither  $X_j$  nor  $A_{i,j}$  given  $A_{i,j} \otimes x_j$ . Furthermore, every data session in AACT is recognized only through pairs of fresh random numbers, which again discloses nothing about user identity information. In addition, AACT requires a network user to refresh session identifiers and the shared symmetric keys for every different session. This further eliminates the linkage among any two sessions originated from the same network user. We note that even with the help of compromised mesh routers and other network users, the adversary still cannot judge whether two communication sessions are from the same user. This is because, basically, none of them can tell whether two signatures are from the same user, given q-SDH problem and decision linear problems on  $G_1$  are hard.

Second, AACT gives adequate user privacy protection against NO while maintaining user accountability. Since NO knows grt, it can always tell which  $gsk[i, j]$  produces a given signature. However, NO has no knowledge about to whom  $gsk[i, j]$  is assigned as AACT allows a late binding among group private keys and network users. Furthermore, it is the user group managers' sole liability to assign group private keys to each network user without any participation of NO. Therefore, NO could only map  $gsk[i, j]$  to the user group i based on  $grp_i$ . Because no other entities except NO and the key holder himself has the knowledge of the corresponding  $A_{i,j}$ , and can therefore, generate the given signature, the key holder has to be a member of the user group i. This audit result serves us both necessities. On one hand, the result only reveals partial unneeded attribute information of the user and still protects user privacy to an extent. On the other hand, the result is adequate for user accountability purposes for NO.

Finally, AACT gives revocable user anonymity against the law authority. As discussed in Section 4.5, the law authority could track any particular user through the cooperation from both NO and the corresponding user group manager.

## 5.3 Performance Analysis

*Communication overhead:* In AACT, Both authentication and key agreement protocols need only three-way communication among mesh routers and network users and among network users. This is the minimal communication rounds essential to achieve mutual authentication, and therefore, AACT incurs a compact authentication delay. Furthermore, by design, AACT poses minimum additional communication overhead on network users as they may carry their mobile clients such as PDAs and smart phones other than laptops to access the WMN. These mobile clients are much less powerful as evaluate to mesh routers with regard to their communication ability. In messages (M.1), ( $\tilde{M}.1$ ), and ( $\tilde{M}.2$ ), a network user only needs to broadcast a group signature to accomplish the authentication function. As we base our group signature difference in the scheme proposed in [8], the signature comprises two elements of  $G_1$  and five elements of  $G_1$ .

When using the curves described in [19], one can take  $p$  to be a 170-bit prime and as a group  $G_1$ , where each element is 171 bits. Thus, the total group signature length is 1,192 bits or 149 bytes. With these parameters, security is about the same as a standard 1,024-bit RSA signature, which is 128 bytes [8]. That is, the length of the group signature is almost the same as that of a standard RSA-1024 signature.

*Computational overhead:* In AACT, the most computationally expensive operations are the signature generation and verification. Signature generation requires two applications of the isomorphism  $\psi$ . Computing the isomorphism takes roughly the similar time as an exponentiation in  $G_1$  (using fast computations of the trace map) [8]. Thus, signature generation needs about eight exponentiations (or multi exponentiations) and two bilinear map computations. Signature verification takes six exponentiations and  $3 + 2|URL|$  computations of the bilinear map. By design, AACT adopts an asymmetric-symmetric hybrid approach for session authentication to decrease computational cost. Network entities (both mesh routers and network users) execute exclusive group signature operation to authenticate each other only when establishing a new session; all subsequent data exchanging of the same session is authenticated through a highly efficient MAC-based approach.

More specifically, AACT requires a network user executing exactly one signature generation and one signature verification when performing mutual authentication for establishing a new session. It can be seen that the actual computational cost of signature verification depends on the size of the URL, while signature generation cost is fixed. AACT can proactively control the size of the URL. Moreover, a far more efficient revocation checks algorithm, whose running time is independent of  $|URL|$  can be adopted as described in [8] with a little bit sacrifice on user privacy. This technique could further bring the total cost of signature verification to six exponentiations and five bilinear map computations. On the other hand, AACT requires a mesh router to perform mutual authentication with every network user within its coverage for each different session and sign on every beacon message being periodically broadcasted.

*Storage overhead:* In AACT, network users may carry resource-constrained persistent devices such as PDAs and smart phones to access the WMN. Therefore, storage overhead for each network user should be reasonable to modern pervasive devices. As is shown in our scheme description, each network user in AACT needs to store two pieces of information: his group private key and the related system parameters. The group private key for each user just contains 1 group element of  $G_1$  and 2 elements of  $Z_p^*$ . If we choose  $p$  to be a 170-bit prime and as a group  $G_1$  with each group element of 171 bits, the group private key for every user just consumes 511-bit memory, which is insignificant for modern pervasive devices. The most memory-consuming parts are the system parameters, which may contain codes to describe the bilinear groups ( $G_1$  and  $G_2$ ), the bilinear pairing function ( $e$ ), the isomorphism  $\psi$ , the hash functions ( $H_0$  and  $H_1$ ), and the signing function ECDSA-160. Fortunately, the needed code size for each part could be in the magnitude of kilobytes as is studied in prior work such as [20]. Therefore, it should be affordable to most of the modern pervasive devices.

## VI. RELATED WORK

Security study in WMNs is still in its early stage, particularly with respect to user privacy protection. Ben Salem and Hubaux [21] discussed specifics of WMNs and identified basic network operations that needed to be secured. Siddiqui and Hong [22] surveyed the threats and vulnerabilities faced by WMNs and also recognized a number of security goals. Cheikhrouhou and Chaouchi [23] discussed a security architecture for WMNs based on IEEE 802.1X. [5] And Zhang and Fang [4] discussed how to support secure user roaming in a number of WMNs belonging to dissimilar domains. Wu and Li [24] presented an anonymous routing scheme for static WMNs. Wan et al. [25] proposed two privacy-preserving routing schemes to give anonymity, unlinkability, and security for WMNs. The authors of [26], [27] presented an authentication scheme for WMNs, which is resilient against mesh router compromise. Other general privacy-aware authentication methods are described in [28], [29], [30].

## VII. CONCLUSION

In this paper, we proposed AACT, which, to the most excellent of our knowledge, is the first attempt to set up an liable security framework with a complicated user privacy protection model tailored WMNs. We tailored group signature scheme[8] that combined with onion ring strategy [31]. We then built AACT on this new model by further integrating it into the authentication and key agreement protocol design. On one hand, AACT enforces strict user access control to cope with both free riders and spiteful users. On the other hand, AACT offers complicated user privacy protection against both adversaries and different other network entities. Our analysis showed that AACT is elastic to a number of security and privacy related attacks. Additional methods were also discussed to further improve the scheme efficiency.

## REFERENCES

- [1] K. Ren and W. Lou, "A Sophisticated Privacy-Enhanced Yet Accountable Security Framework for Wireless Mesh Networks," Proc. 28th Int'l Conf. Distributed Computing Systems (ICDCS '08), June 2008.
- [2] I.F. Akyildiz, X. Wang, and W. Wang, "Wireless Mesh Networks: A Survey," Computer Networks, vol. 47, no. 4, pp. 445-487, Mar.2005.
- [3] "Self Organizing Neighborhood Wireless Mesh Networks," <http://www.research.microsoft.com/mesh/>, 2009.
- [4] Y. Zhang and Y. Fang, "A Secure Authentication and Billing Architecture for Wireless Mesh Networks," ACM Wireless Networks, to be published.
- [5] Y. Zhang and Y. Fang, "ARSA: An Attack-Resilient Security Architecture for Multi-Hop Wireless Mesh Networks," IEEE J. Selected Areas in Comm., vol. 24, no. 10, pp. 1916-1928, Oct.2006.
- [6] "The Wimax Forum,"<http://www.wimaxforum.org>, 2009.
- [7] "Boston Suburb Secures Metro-Scale Wireless Mesh Network with Bluesocket," <http://www.tmcnet.com/usubmit/2006/09/27/1936581.htm>, Sept. 2006.
- [8] D. Boneh and H. Shacham, "Group Signatures with Verifier-Local Revocation," Proc. ACM Conf. Computer and Comm. Security (CCS), pp. 168-177, 2004.
- [9] D. Chaum and E. van Heyst, "Group Signatures," Proc. Conf. Eurocrypt, pp. 257-265, 1991.
- [10] R. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Comm. ACM, vol. 21, no. 2, pp. 120-126, 1978.
- [11] M. Jakobsson, J. Hubaux, and L. Buttyan, "A Charging and Rewarding Scheme for Packet Forwarding in Multi-Hop Cellular Networks," Proc. Seventh Int'l Conf. Financial Cryptography (FC), 2003.
- [12] N. Salem, L. Buttyan, J. Hubaux, and M. Jakobsson, "A MicroPayment Scheme Encouraging Collaboration in Multi-Hop Cellular Networks," Proc. ACM MobiHoc, 2003.
- [13] D. Hankerson, A. Menezes, and S. Vanstone, Guide to Elliptic Curve Cryptography. Springer-Verlag, 2004.
- [14] Y. Zhang, W. Liu, and W. Lou, "Anonymous Communications in Mobile Ad Hoc Networks," Proc. IEEE INFOCOM, Mar. 2005.
- [15] Y. Zhang, W. Liu, W. Lou, and Y. Fang, "MASK: Anonymous On- Demand Routing in Mobile Ad Hoc Networks," IEEE Trans. Wireless Comm., vol. 5, no. 9, pp. 2376-2385, Sept. 2006.
- [16] J. Kong, X. Hong, and M. Gerla, "An Identity-Free and On- Demand Routing Scheme against Anonymity Threats in Mobile Ad Hoc Networks," IEEE Trans. Mobile Computing, vol. 6, no. 8, pp. 888-902, Aug. 2007.
- [17] B. Zhu, Z. Wan, M.S. Kankanhalli, F. Bao, and R.H. Deng, "Anonymous Secure Routing in Mobile Ad-Hoc Networks," Proc. 29th Ann. IEEE Int'l Conf. Local Computer Networks (LCN '04), pp. 102-108, Nov. 2004.
- [18] A. Juels and J. Brainard, "Client Puzzles: A Cryptographic Countermeasure against Connection Depletion Attacks," Proc. Sixth Network and Distributed System Security Symp. (NDSS), 1999.
- [19] D. Boneh, H. Shacham, and B. Lynn, "Short Signatures from the Weil Pairing," J. Cryptology, vol. 17, no. 4, pp. 297-319, 2004.
- [20] TinyECC Library, <http://discovery.csc.ncsu.edu/software/TinyECC/index.html>, 2009.
- [21] N. Ben Salem and J.-P. Hubaux, "Securing Wireless Mesh Networks," IEEE Wireless Comm., vol. 13, no. 2, pp. 50-55, Apr.2006.
- [22] M. Siddiqui and C. Hong, "Security Issues in Wireless Mesh Networks," Proc. IEEE Int'l Conf. Multimedia and Ubiquitous Eng.,2007.
- [23] A. Cheikhrouhou and H. Chaouchi, "Security Architecture in a Multi-Hop Mesh Network," Proc. Fifth Conf. Security Architecture Research, 2006.
- [24] X. Wu and N. Li, "Achieving Privacy in Mesh Networks," Proc. ACM Workshop Security of Ad Hoc and Sensor Networks (SASN),2006.
- [25] Z. Wan, K. Ren, B. Zhu, B. Preneel, and M. Gu, "Anonymous User Communication for Privacy Protection in Wireless Metropolitan Mesh Networks," Proc. ACM Symp. Information, Computer and Comm. Security (AsiaCCS), 2009.
- [26] X. Lin, R. Lu, P.-H. Ho, X. Shen, and Z. Cao, "Tua: A Novel Compromise-Resilient Authentication Architecture for Wireless Mesh Networks," IEEE Wireless Comm., vol. 7, no. 4, pp. 1389-1399, Apr. 2008.
- [27] X. Lin, X. Ling, H. Zhu, P.-H. Ho, and X. Shen, "A Novel Localized Authentication Scheme in IEEE 802.11 Based Wireless Mesh Networks," Int'l J. Security and Networks, vol. 3, no. 2, pp. 122-132, 2008.
- [28] K. Ren, W. Lou, K. Kim, and R. Deng, "A Novel Privacy Preserving Authentication and Access Control Scheme for Pervasive Computing Environment" IEEE Trans. Vehicular Technology, vol. 55, no. 4, pp. 1373-1384, July 2006.
- [29] K. Ren and W. Lou, "Privacy-Enhanced, Attack-Resilient Access Control in Pervasive Computing Environments with Optional Context Authentication Capability," ACM Mobile Networks and Applications (MONET) (special issue on wireless broadband access), vol. 12, pp. 79-92, 2007.
- [30] Y. Zhang and K. Ren, "On Address Privacy in Mobile Ad Hoc Networks," ACM/Springer Mobile Networks and Applications (MONET), vol. 14, no. 2, pp. 188-197, Apr. 2009.
- [31] M. Reed, P. Syverson, and D. Goldschlag, Anonymous Connections and Onion Routing, IEEE Journal on Selected Areas in Communication Special Issue on Copyright and Privacy Protection, 1998.

## Challenges of Technology Infrastructure Availability in E-Governance Program Implementations: A Cloud based Solution.

<sup>1</sup>Sameer Goel, <sup>2</sup>Manoj Manuja, <sup>3</sup>Rajeev Dwivedi, <sup>4</sup>A.M. Sherry,

<sup>1</sup>Institute of Management Technology, Ghaziabad

<sup>2</sup>Manager, Education and Research, Infosys, Ltd.

<sup>3,4</sup>Institute of Management Technology (CDL), Ghaziabad

---

**Abstract:** Implementation of large program in a governmental scenario is always a challenging work. There are many constraints which a government needs to address before offering value-added services to its citizens and other stake holders in a seamless environment. With the explosion of information over web in recent past, governments across the world face a major challenge in keeping a pace with ever changing technologies and offer an efficient, effective and transparent way of offering its services. Last couple of years has observed cloud computing taking a center stage for offering various services in a cost efficient and automated model. In this article, we discuss various challenges being faced by government in providing its services through e-governance model to all its stake holders. A cloud computing solution is proposed to address some of the e-governance challenges being faced by government. A fine grained model is discussed towards the end to highlight various benefits this solution brings along with it.

**Keywords:** Cloud Computing, E-governance, Technology Infrastructure

---

### I. Introduction

In today's time, governance across the globe is getting sturdy and challenging, be it public, business or corporate governance. It is a crucial process of decision making with lot of dependency on how these decisions are implemented. We can evaluate successful or unsuccessful economies across the world just by comparing their governance indicators with each other which are published by World Bank on a regular basis [1].

Information technology (I.T.) has experienced an exceptional growth over the recent couple of decades. We can easily observe the influence of I.T. in governance of either a big corporate or a country. This electronic governance, popularly known as e-governance, is becoming the backbone of any country's growing economy in today's world of internet enabled systems and processes. The word "electronic" primarily indicates the usage of technology in all matters of governance. This includes Government-to-citizens (G2C), Government-to-Business (G2B), Government-to-employees (G2E), Government-to-Government (G2G) as well as interactions and processes happening at the back office system levels within the entire government frame work. Government tries to leverage technology to make all its services available to its different stake holders namely citizens, businesses and government itself in an efficient, effective and transparent manner.

Today, many governments across the globe are using technology to leverage the huge reach of e-governance model in their respective countries [2]. There are many ways of implementation being followed to provide a trouble-free, proficient access to most of the required information and services for the citizens. One of the most accepted and popular ways of offering e-governance services is to launch one-stop unified portal which can be accessed by the people for various governmental services.

It is observed that the developed countries spend a record budget in supporting its e-government initiatives [3]. The trend of enhanced spending on e-governance programs is catching up in developing countries and Indian government has committed large part of budget to e-governance programs [4]. It is evident after careful analysis of this expenditure that considerable part of the budget is incurred towards building hardware infrastructure. The hardware infrastructure such created struggles to sustain itself in view of the technology advancement and lot of tax payers' money is spent without making full use of this. To implement e-governance in large developing countries like India, it is important that optimum usage of resources is made and expenditure made on technology infrastructure is efficiently used. This paper explores the challenges in technology infrastructure availability and a cloud based solution for the same.

### II. Methodology

A review of literature has been carried out for better understanding of the challenges and issues involved in e-governance. Based on the literature review the issues in availability of technology Infrastructure are collated. Focused group discussions were held with the experts in the area of e-governance and technology innovations. Based on the focused group discussions with these experts, a cloud based framework has been proposed for e-governance programs.

### **III. Literature Review**

Development of e-governments is directly proportional to the IT infrastructure that is capable of supporting and enabling the execution of e-government. An e-government infrastructure in general comprises network infrastructure, security infrastructure, application server environment, data and content management tools, application development tools, hardware and operating systems, and systems management platform [5]. However, large parts of India do not have the infrastructure necessary to deploy e-government services throughout its territory.

Apart from the availability of the infrastructure, the other key challenge on this front is to design interoperable systems. Lex van Velsen [6] describe that e-Government services often span several organizations or departments. Such a service can be offered to the citizen via one single website, supported by an interoperable system. This website should present the service as a coherent and logical whole for the user.

Subhasis Ray and Amitava Mukherjee [7] in their study on development of a framework towards successful implementation of e-governance initiatives in health sector in India identify that lack of standardization of system components and services such as health information systems, health messages, electronic health record architecture, and patient identifying services may be a hindrance for Interoperability of e-healthcare systems.

Jaijit Bhattacharya and Sushant Vashistha [8] discuss the challenges of not following a standardized architecture for e-governance programs. These challenges are:

- It will lead to a large pile-up of heterogeneous IT infrastructure.
- Difficulty in provisioning complete IT infrastructure for an e- Governance initiative in every department, region and state. It is going to exert a lot of cost pressures on the governments if the spikes in computing requirement are to be met in-house without creating a common back-end that smoothens out these spikes and hence maximizes the utilization of the IT infrastructure and public money.
- Interoperability issue with the different systems being required to talk with each other.
- Adapting to rapidly changing IT software and hardware technologies and “Technology Obsolescence”.
- Under-utilization of the IT capacity, hence leading to wastage and ever accruing maintenance cost.

The above issues are even more critical for the developing economies where the resources are limited and hence require prudent expenditure on the part of the government.

Interoperability especially in case of G2G project should have an in-built system for seamless operations across different platforms. This is especially important in cases where different departments operating on different platforms are sought to be integrated to ensure seamless flow of information within the government.[9, 10]

### **IV. Analysis and Discussion**

Based on the inputs from literature review, focused group discussions were held with the technology and e-governance experts. The challenges faced by governments in providing e-governance solutions on technology infrastructure front, which were highlighted in these focused group discussions were:

1. **Infrastructure complexity:** It is a gigantic task to put proper infrastructure in place so that all public services are made available to citizen online. We need to install hardware at different places in villages, cities, districts and state to cater to different stake holders. Linking all the installed infrastructure over a network is indeed a tedious work. Moreover, a regular updation and maintenance of hardware and software consumes lot of money on a regular basis.
2. **System Management:** Once the infrastructure is in place, management of the system is next critical and complex activity. There are very rapid changes and advancements in the technology which makes our hardware and software obsolete. This is a big challenge because the budgets are directly linked with this activity and huge cost is involved in replacing or upgrading the hardware / software.
3. **Human resource management:** The complete system and infrastructure is installed, commissioned and maintained by highly paid information technology professionals. Government is required to recruit such people and support the IT dept with a high salary cost. Also, the movement of IT professionals within the industry is very common; hence, it becomes very difficult to retain them.
4. **Scalability:** E-governance systems have huge variations in the demands during peak and off peak hours with demand growing manifold over the time. The systems designed most of the time are not scalable to meet the requirements and eventually customer services suffer due to this issue.

### **V. Cloud Computing Based E-Governance Solution**

Cloud computing is like a service processing mechanism which is web-based, and the customer is almost unaware of the source of the solution he is getting from. He is just accessing a web based service available on cloud and getting the solutions or replies to his queries. Cloud computing is termed as a Web-based processing, whereby shared resources of software and data are provided to computers and other related devices (such as smart-phones, PDAs and other mobile devices) on demand over the network.

### Challenges of Technology Infrastructure Availability in E-Governance Program Implementations: A

Cloud computing is a versatile technology that uses the internet based web services and remotely available central servers to maintain various applications and related databases. Cloud computing allows consumers who may be end-users or any business organization to use various remote applications without even a single installation being done at their premise. It also helps to access their business as well as personal files at any computer situated anywhere in the globe with available internet access. This resourceful, adaptable and flexible technology allows the users for much more capable computing by centralizing storage space, proper memory management, fine-tuned processing and properly utilized bandwidth [11]. Figure – 1 shows five layers of type cloud computing implementation.

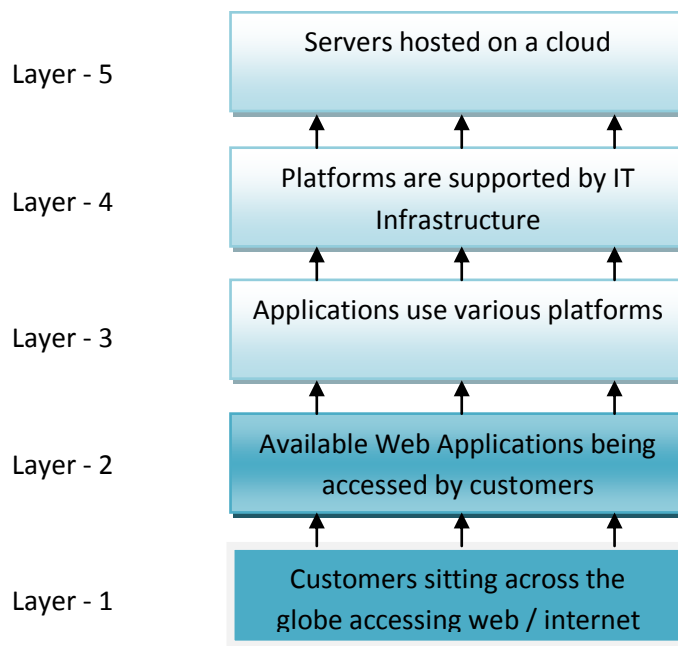
**Layer – 1** - Customers / end users: This layer consists of customers sitting across the globe accessing various applications through web / internet.

**Layer – 2** – Web Applications: Cloud application services are available on internet eliminating the need to have the application installed and being run and managed on the customer's own computer infrastructure. It primarily simplifies the required maintenance and support. This can be categorized as SaaS (Software as a Service).

**Layer – 3** – Platform services: The said layer provides a computing platform with or without solution stack as a service. It often consuming cloud infrastructure and sustaining *cloud applications*. This can be categorized as PaaS (Platform as a Service).

**Layer – 4** – Infrastructure Services: This layer primarily provides computer infrastructure for creating virtualization environment which may be termed as a platform service. This can be categorized as “Infrastructure as a Service” or popularly known as IaaS.

**Layer – 5** – Servers: This layer is the amalgamation of computer hardware and software that are exclusively designed for the delivery of services on the cloud. This infrastructure includes multi-core processors, cloud-specific operating systems and combined offerings.



**FIGURE – 1**

#### **Cloud Deployment Models:**

Different models of cloud deployment are available in the global arena [12]:

1. **Public Cloud:** It is also termed as external cloud which describes cloud computing in the conventional mainstream sense. The resources are dynamically provisioned on a fine-grained, self-service model over the Internet, through web applications and/or web services, by a third-party service provider who bills the customers on a fine-grained utility computing model.
2. **Private Cloud:** It is also termed as internal cloud which describes cloud computing as a model, where resources are dynamically provisioned on a fine-grained, private-service basis over the Intranet through web applications and/or web services, from privately managed self provider who is also the owner of the deployment. This infrastructure is being made available solely to the users within an organization. It may be managed by the company itself or it can hire a third party to host and operate it on premise or off premise.
3. **Community Cloud:** This type of cloud infrastructure is being shared by several businesses or organizations for a shared purpose. It supports a specific community that has shared concerns and common goals like mission, particular security requirements, general policies, and their compliance considerations. It may be

## Challenges of Technology Infrastructure Availability in E-Governance Program Implementations: A

managed by the organization itself or a third party can be entrusted to operate the same on premise or off premise.

4. **Hybrid Cloud:** This type of cloud infrastructure is primarily composed of two or more clouds which may be private, community, or public. It is primarily being used when we need to bind together standardized or proprietary technology which enables data and application portability as shown in figure - 2.

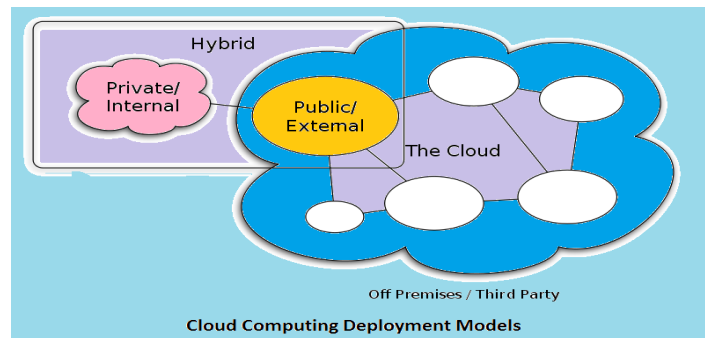


FIGURE - 2

### **Solution Framework for E-Governance Programs**

A framework is suggested for a governmental e-governance.

There are mainly two types of users for this cloud based e-governance service:

- a. **Government Officials:** This segment of users is primarily government officials who will be accessing the web applications available on cloud to manage and maintain the complete governance infrastructure and offer public focused services. This segment can be sub divided into three parts:

- I. Village / City level Govt officials
- II. District level Govt officials
- III. State level Govt officials

- b. **General public end users:** This segment of users is primarily end users who will be accessing the web applications available on cloud to avail all the services being offered by the government through this model. Again, this segment can be sub divided into three parts:

- I. Village / City level end users
- II. District level end users
- III. State level end users

**Cloud Infrastructure:** As shown in figure – 3, all users will be accessing the cloud through a common cloud infrastructure which offers common services.

The complete e-governance has two major stake holders namely service providers and service consumers. Therefore, the suggested framework offers a hybrid solution for the cloud implementation. The framework offers two clouds namely:

- A. **Public Cloud:** This cloud will be accessed by general end users who want to use an available web application to avail some governmental service. This cloud will be available on public domain where anybody and everybody can access the available services.
- B. **Private Cloud:** This cloud will be accessed by governmental officials who want to access the web application to manage and maintain the offered governmental services. This cloud will be available as a private cloud where secured login will be required by authorized officials to access available web applications. These officials will be the people who do add / delete / update operations on the e-governance database which is being accessed by general end users through public cloud.

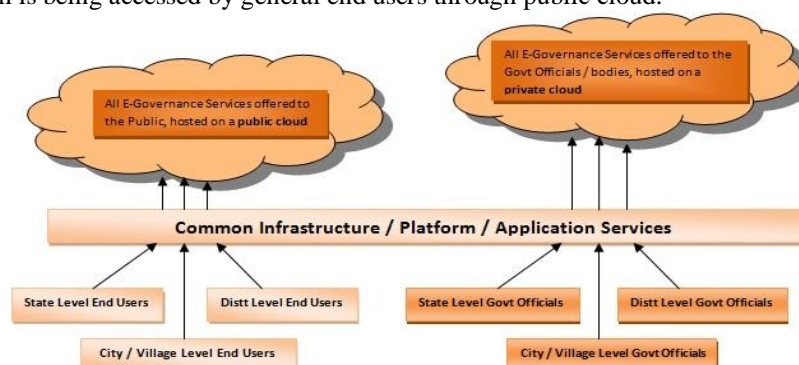


FIGURE - 3



### **Advantages of Suggested Framework**

1. **24x7 Service Availability:** The service in the pre-cloud era has been restricted to only office timings. The government officials are available between 9am till 5pm and all the work is done with personal intervention of the governmental officials. In post-cloud era, all services are available 24x7 with minimal human intervention.
2. **Cost effective solution:** The suggested model helps reduce the cost to provide governance services to the masses. In a non-cloud scenario, state government was forced to install so much of hardware at different places in villages, cities, districts and state to cater to different stake holders. In cloud environment, complete hardware and software support is provided by the service provider who build, install, commission and maintain the whole set-up. Government pays to service provider on mutually agreed terms and conditions.
3. **Easy system administration:** We can quickly obtain the benefits of the colossal infrastructure with no hassles to implement, control and administer it directly. This also allows the users to access multiple database centers situated anywhere on the globe. It also signifies that the companies can add additional services and infrastructure as and when required based upon the customers' need. It also helps the organizations to save on the cost of additional hardware in this scenario where expenditure is borne by the cloud computing vendor.
4. **Services on Mobile:** Cloud computing helps Government to offer all administrative services to its users through mobile devices. Employees can access work-related information from anywhere. All stake holders can access relevant data and various services wherever they are, rather than having to remain at their desk.
5. **Scalability:** The systems such designed are scalable and additional infrastructure can be added as the demand grows and the department pays for the infrastructure only when it has requirement for the same.
6. **High performance:** Systems developed on this environment are very good in terms of availability, accessibility and performance. The cloud service provider ensures that enough redundancy is built in the system, so that it can provide uptime to the tune of 99.99% and desired performance response at all times.

## **VI. Conclusion**

The success of any e-governance model depends upon various factors. The challenges primarily focus on system availability, infrastructure complexity, application management and people management. These challenges amount to various complexity during the implementation of e-governance model by any government agency. Cloud computing has offered a lot of opportunities to corporate as well as governments across the globe. E-governance implemented on a cloud offers enormous opportunities to the government as well as end users i.e. citizens. Government scores in terms of providing 24x7 services to its citizens, effortless maintenance as the cloud systems are managed by third parties in general. Citizens get the seamless services any where any time which is a great advantage. In nut shell, cloud is the future of e-governance model which is scalable, easy to operate and maintain.

## **References**

- [1] Subhajt Basu, "E-Government and Developing Countries: An Overview", Pages 109–132, International Review of Law Computers & Technology, Vol. 18, No. 1, 2004
- [2] Sunny Marche, James D. Mcniven, "e-government and e-governance: The future isn't what it used to be", Page 74-86, Canadian Journal of Administrative Science, 2003
- [3] Lourdes Torres, Vicente Pina, Basilio Acerete, "E-Governance Developments in European Union Cities: Reshaping Government's Relationship with Citizens", Page 277-302, Governance: An International Journal of Policy, Administration, and Institutions Vol. 19, No 2, April 2006
- [4] Rahul De, "E-Government Systems in Developing Countries: Issues and Concerns – Discussion" Page 377-388, IIMB Management Review, December 2006
- [5] Subhajt Basu, "E-Government and Developing Countries: An Overview", Pages 109–132, International Review of Law Computers & Technology, Vol. 18, No. 1, 2004
- [6] Velsen Lex van, Geest Thea van der, Hedde Marc ter, Derks Wijnand, (2009) Requirements engineering for e-Government services: A citizen-centric approach and case study, Government Information Quarterly 26, 477–486
- [7] Ray Subhasis, Mukherjee Amitava, (2007) Development of a framework towards successful implementation of e-governance initiatives in health sector in India, International Journal of Health Care Quality Assurance Vol. 20 No. 6, 464-483
- [8] Bhattacharya Jaijit, Vashista Sushant, (2008) Utility Computing-Based Framework for e-Governance, ICEGOV2008, Cairo, Egypt
- [9] Gregory D. Streib, Katherine G. Willoughby, "e-Governments: Meeting the Implementation Challenge", Page 78-112, PAQ Spring, 2005
- [10] Subhash C. Bhatnagar, Nupur Singh, "Assessing the Impact of E-Government: A Study of Projects in India", Page 109–127, Information Technologies & International Development Vol. 6, No 2, 2010
- [11] Manish Pokharel, Jong Sou Park, "Cloud computing: future solution for e-governance", Proceedings of the 3rd international conference on Theory and practice of electronic governance, 2009.
- [12] Ashish Rastogi, A model based approach to implement Cloud Computing in E-Governance, International Journal of Computer Applications, ISSN – 0975-8887, Volume – 9 , No. 7, November 2010

## Modeling and Containment of Uniform Scanning Worms

Namratha M<sup>1</sup>, Pradeep<sup>2</sup>

<sup>1,2</sup> (Dept. of information science, PESIT/visvesvaraya technological university, India)

---

**Abstract:** Self-propagating codes called worms such as Code Red, Nimda, and Slammer have drawn significant attention due to their enormously adverse impact on the Internet. Thus, there is great interest in the research community in modeling the spread of worms and in providing adequate defense mechanisms against them. In this model, we present a (stochastic) branching process model for characterizing the propagation of Internet worms. The model is developed for uniform scanning worms and then extended to preference scanning worms. This model leads to the development of an automatic worm containment strategy that prevents the spread of a worm beyond its early stage. Specifically, for uniform scanning worms, we are able to determine whether the worm spread will eventually stop. We then extend our results to contain uniform scanning worms. Our automatic worm containment schemes effectively contain both uniform scanning worms and it is validated through simulations.

The Internet has become critically important to the financial viability of the national and the global economy. Meanwhile, we are witnessing an upsurge in the incidents of malicious code in the form of computer viruses and worms. One class of such malicious code, known as random scanning worms, spreads itself without human intervention by using a scanning strategy to find vulnerable hosts to infect

**Keywords:** Branching process model, Broadcast, Containment, Intrahost spreading, Scanning, Socket Communication, Worm

---

### I. Introduction

A computer worm[1] is a self-replicating computer program. It uses a network to send copies of itself to other nodes and it may do so without any user intervention. The name **worm** comes from *The Shockwave Rider*, a science fiction novel published in 1975 by John Brunner.

A computer virus[2] is a computer program that can copy itself and infect a computer in some form of executable code. Spreads by human action, sent over a network or the Internet, carried it on a removable medium such as a floppy disk, CD, DVD, or USB drive. Unlike a virus, it does not need to attach itself to an existing program. Worms almost always cause harm to the network, if only by consuming bandwidth, whereas viruses almost always corrupt or modify files on a targeted computer.

Worms[3] spread by exploiting vulnerabilities in operating systems. All vendors supply regular security updates and if these are installed to a machine then the majority of worms are unable to spread to it. If a vendor acknowledges a vulnerability but has yet to release a security update to patch it, a zero day exploit is possible. However, these are relatively rare. Users need to be wary of opening unexpected email, and should not run attached files or programs, or visit web sites that are linked to such emails. However, as with the ILOVEYOU worm, and with the increased growth and efficiency of phishing attacks, it remains possible to trick the end-user into running a malicious code. Anti-virus and anti-spyware software are helpful, but must be kept up-to-date with new pattern files at least every few days. The use of a firewall is also recommended. Computer worms -- malicious, self propagating programs -- represent a substantial threat to large networks. Since these threats can propagate more rapidly than human response [, automated defenses are critical for detecting and responding to infections. One of the key defenses against scanning worms which spread throughout an enterprise is *containment*. Worm containment, also known as virus throttling, works by detecting that a worm is operating in the network and then blocking the infected machines from contacting further hosts. Currently, such containment mechanisms only work against *scanning* worms because they leverage the anomaly of a local host attempting to connect to multiple other hosts as the means of detecting an infectee. Within an enterprise, containment operates by breaking the network into many small pieces, or *cells*. Within each cell (which might encompass just a single machine), a worm can spread unimpeded. But between cells, containment attempts to limit further infections by blocking outgoing connections from infected cells.

A key problem in containment[4] of scanning worms is efficiently detecting and suppressing the scanning. Since containment *blocks* suspicious machines, it is critical that the false positive rate be very low. Additionally, since a successful infection could potentially subvert any software protections put on the host machine, containment is best effected inside the network rather than on the end-hosts. We have developed a scan detection and containment algorithm using java platform. The make our algorithm suitable for both hardware and software implementation. Evaluating the algorithm on traces from a large (6,000 host) enterprise,

we find that with a total memory usage of 5 MB we obtain good detection precision while staying within a processing budget of at most 4 memory accesses (to two independent banks) per packet.

Worm containment systems have an epidemic threshold: if the number of vulnerable machines is few enough relative to a particular containment deployment, then containment will almost completely stop the worm. However, if there are more vulnerable machines, then the worm will still spread exponentially.

Finally, in general malicious attacks on worm containment systems[5],[6],[7]: what is necessary for an attacker to create either false negatives (a worm which evades detection) or false positives (triggering a response when a worm did not exist), assessing this for general worm containment, cooperative containment, and our particular proposed system. We specifically designed our system to resist some of these attacks. Worm containment is designed to halt the spread of a worm in an enterprise by detecting infected machines and preventing them from contacting further systems. Current approaches to containment are based on detecting the scanning activity associated with scanning worms, as is our new algorithm. The key component for today's containment techniques is *scanning*: responding to detected *portscans* by blocking future scanning attempts. Portscans--probe attempts to determine if a service is operating at a target IP address--are used by both human attackers and worms to discover new victims. Portscans have two basic types: *horizontal* scans, which search for an identical service on a large number of machines, and *vertical* scans, which examine an individual machine to discover all running services. The goal of scan suppression is often expressed in terms of preventing scans coming from "outside" inbound to the "inside". If "outside" is defined as the external Internet, scan suppression can thwart naive attackers. But it can't prevent infection from external worms because during the early portion of a worm outbreak an inbound-scan detector may only observe a few (perhaps only single) scans from any individual source. Thus, unless the suppression device halts all new activity on the target port it will be unable to decide, based on a single request from a previously unseen source, whether that request is benign or an infection attempt. For worm *containment*, however, we turn the scan suppressor around: "inside" becomes the enterprise's larger internal network, to be protected from the "outside" local area network. Now any scanning worm will be quickly detected and stopped, because (nearly) *all* of the infector's traffic will be seen by the detector.

## II. Design Considerations

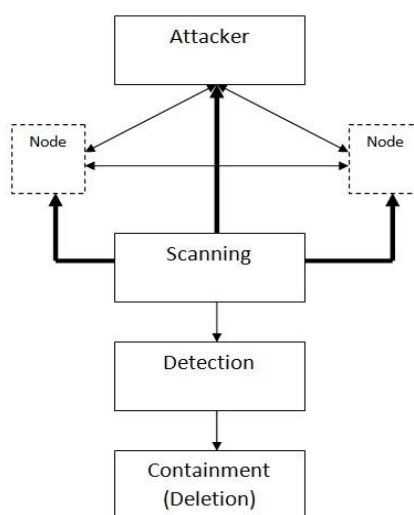


Fig 1: High level design

Functionality:

- Step 1: The attacker sends the worm to each of the other host
- Step 2: Each host intern sends it to each of the other host
- Step 3: This process continues until stop spreading is pressed.
- Step 4: Simultaneously in each host, scan the specified path.
- Step 5: Delete all the detected worm files.

### 2.1 Branching Process Model[8]:

To the problem of combating worms[9], we have developed an inter and intra branching process model to characterize the propagation of Internet worms. Unlike deterministic epidemic models present earlier, this model allows us to characterize the early phase of worm propagation.

Inter host spreading: This module is responsible for modeling the spread of uniform scanning worms over a network, i.e., from one host to another.

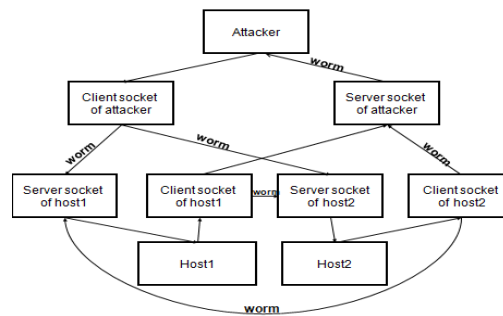


Fig 2: Flow diagram of inter host spreading

Functionality:

The sequence of steps is as follows:

Step 1: Create server socket for each of the host.

Step 2: On the primary attacker select the type of the worm to spread in the network.

Step 3: Create client socket on the primary attacker node

Step 4: Send the worm through the client socket to listening server socket of each of the other nodes in the network

Step 5: Now each node sends the worm through its client socket to the server socket of the other node.

Step 6: repeat step 5 until stop spreading is pressed.

Intra host spreading: This module is responsible for modeling the spread of worms[10] through a replication mechanism within a node in the network.

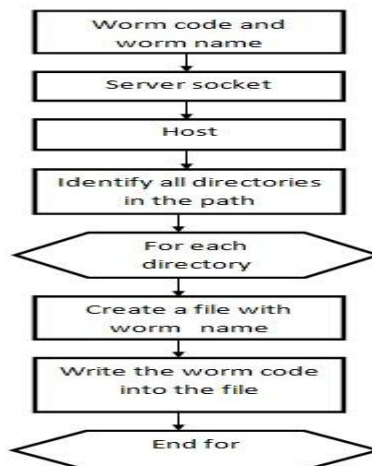


Fig 3: Flow diagram of intrahost spreading

## 2.2 Scanning:

Our strategy is based on limiting the number of scans to the specified path in the system. Our worm containment schemes effectively contain uniform scanning worms and it is validated through simulations and real trace data to be non-intrusive. This module is responsible for scanning the specified path in the system for the presence of the specific worm.

Step 1: Start.

Step 2: t1:=start time

Step 3:=for each directory

Step 4: dir\_count++

Step 5: for each directory in the file

Step 6: file\_count++

Step 7: Display current\_file being scanned

Step 8: if file\_name != worm\_name then goto step 12.

Step 9: detections\_count++

Step 10: Worm containment

Step 11: Display path of detected worm file.

Step 12: End for

- Step 13: End for
- Step 14: t2=stop time
- Step 15: t:= t2-t1
- Step 16: Display detections\_count, dir\_count, file\_count, time\_to\_scan;
- Step 17: Stop

2.3 Containment of worms:

This module leads to the development of an automatic worm containment strategy that prevents the spread of a worm, specifically for uniform scanning worms. This module is responsible for the deletion of the detected files.

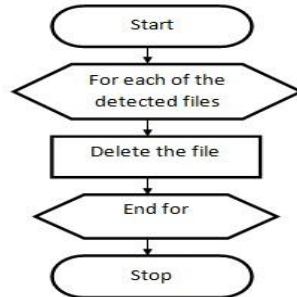


Fig 4: Flow diagram for containment of worms

2.4 Client-Server Framework:

The client-server framework[11] has been used for the modeling of worm spread across a network. The client sends the worm and the server receives the worm. Hence the client is associated with the attacker and the server is associated with the node being attacked in the network. The client and server communicate with each other by creating sockets at specific port numbers and creating Input/Output DataStreams to pass request messages and responses to enable communication.

III. Experiments And Results

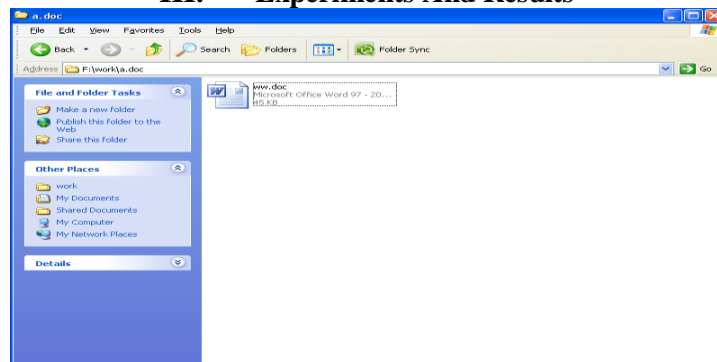


Fig 5: Before spreading there is no worm file in the folder a.doc

```

Host1.java - Notepad
File Edit Format View Help

/*          Host1          */
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.net.*;
import java.io.*;
import java.lang.String;

public class Host1 extends JFrame
{
    // Variables declaration
    public JLabel logLabel1;
    public JLabel wormCollageLabel;
    public ImageIcon logo;
    public ImageIcon wormCollage;
    public JLabel logText1;
    public JLabel logText2;

    private JLabel jLabel1;
    private JLabel jLabel2;
    private JComboBox jComboBox1;
    private JButton jButton1;
    private JButton jButton2;
    private JButton jButton3;
    private JButton jButton4;
    private JPanel contentPane;
    ServerSocket hst;
    socket hstcp, hstc;
    public static String wrm, incwrm, add, host, host2, host3, host4;
    public static int ch, ptno, j, k, d;
    public int status;
    public String drive="F:\\work";
    public static String Fien, filename;
    public static FileWriter fw;
    public static String ad="", wname="worm32.dll";
    public static String spl[] = new String[500];
    // end of variables declaration
  
```

Fig 6: The path in each host has been set to F:\work. Therefore a worm file will be created in each directory in the path, F:\work.

```

C:\WINDOWS\system32\cmd.exe
F:\8th Sem Project\5. Modeling & Automated Containment of Worms full complete w
ith ppt .doc, run. ieee ref\SOURCE CODE\Host 1>set path = C:\Program Files\Java\j
dk1.6.0_06\bin;

F:\8th Sem Project\5. Modeling & Automated Containment of Worms full complete w
ith ppt .doc, run. ieee ref\SOURCE CODE\Host 1>javac *.java
Note: Containment.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

F:\8th Sem Project\5. Modeling & Automated Containment of Worms full complete w
ith ppt .doc, run. ieee ref\SOURCE CODE\Host 1>java Host1

Worm Selected
>>MyDoom.xff is selected.
MyDoom.xff df
-
    
```

Fig 7: The worm MyDoom.xff is selected for spreading at host1+

<pre> C:\WINDOWS\system32\cmd.exe Start Spreading Worm Sent Received Worm From Host2 F:\work\*.doc Worm Sent Host 2 Worm Sent Host 3 Worm Sent Host 4 Received Worm From Host4 F:\work\movies Worm Sent Host 2 Worm Sent Host 3 Worm Sent Host 4 Received Worm From Host3 F:\work\*.doc Worm Sent Host 2 Worm Sent Host 3 Worm Sent Host 4 Received Worm From Host2 F:\work\movies Worm Sent Host 2 Worm Sent Host 3 Worm Sent Host 4 Received Worm From Host4 F:\work\*.doc                 </pre>	<pre> C:\WINDOWS\system32\cmd.exe S=1 Received Worm From Host1 F:\work\movies Worm Sent to Host 3 Worm Sent to Host 4 Worm Sent to Host 1 S=1 Received Worm From Host4 F:\work\*.doc Worm Sent to Host 3 Worm Sent to Host 4 Worm Sent to Host 1 S=1 Received Worm From Host3 F:\work\movies Worm Sent to Host 3 Worm Sent to Host 4 Worm Sent to Host 1 S=1 Received Worm From Host1 F:\work\*.doc Worm Sent to Host 3 Worm Sent to Host 4 Worm Sent to Host 1                 </pre>
<pre> C:\WINDOWS\system32\cmd.exe Worm Sent to Host 2 S=1 Received Worm From Host1 F:\work\movies Worm Sent to Host 4 Worm Sent to Host 1 Worm Sent to Host 2 S=1 Received Worm From Host2 F:\work\*.doc Worm Sent to Host 4 Worm Sent to Host 1 Worm Sent to Host 2 S=1 Received Worm From Host4 F:\work\movies Worm Sent to Host 4 Worm Sent to Host 1 Worm Sent to Host 2 S=1 Received Worm From Host1 F:\work\*.doc Worm Sent to Host 4 Worm Sent to Host 1                 </pre>	<pre> C:\WINDOWS\system32\cmd.exe Worm Sent to Host 2 Worm Sent to Host 3 S=1 Received Worm From Host3 F:\work\movies Worm Sent to Host 1 Worm Sent to Host 2 Worm Sent to Host 3 S=1 Received Worm From Host1 F:\work\*.doc Worm Sent to Host 1 Worm Sent to Host 2 Worm Sent to Host 3 S=1 Received Worm From Host2 F:\work\movies Worm Sent to Host 1 Worm Sent to Host 2 Worm Sent to Host 3 S=1 Received Worm From Host3 F:\work\*.doc Worm Sent to Host 1                 </pre>

Fig 8: Host1 sends the worm to host2, host3, host4 and receives the worm from host2, host3, host4. Meanwhile the spreading within the path , F:\work is taking place.

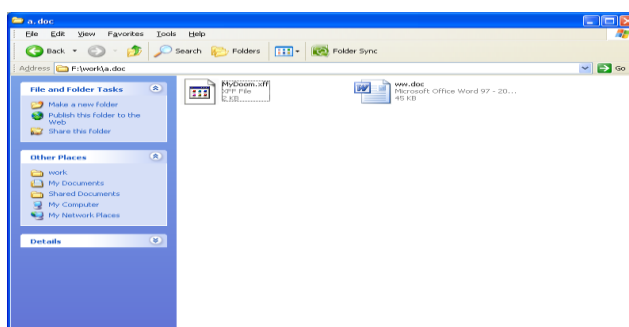


Fig 9: The worm has been created in F:\work\\*.doc, which is in the specified path.



Fig 10: User Interface for containment in host1

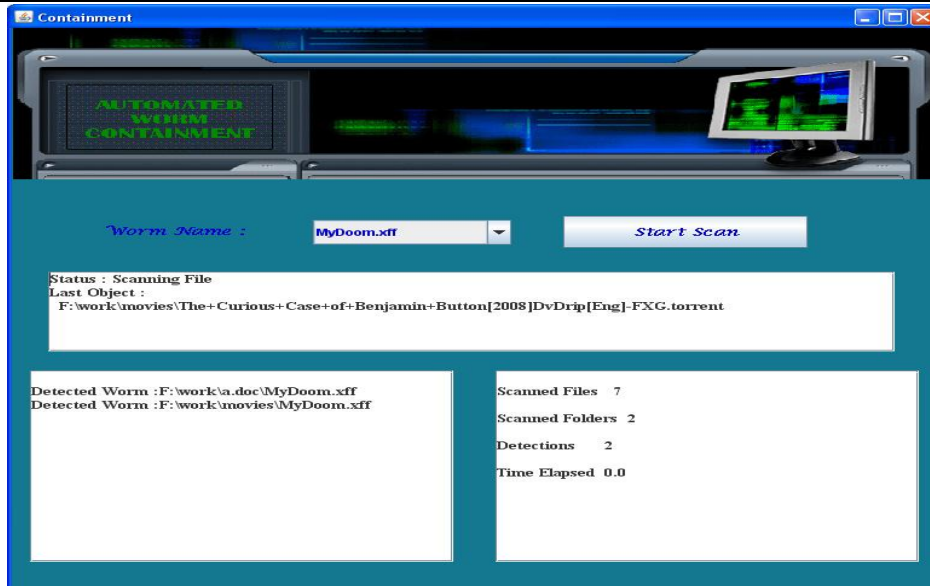


Fig 11: Scanning performed

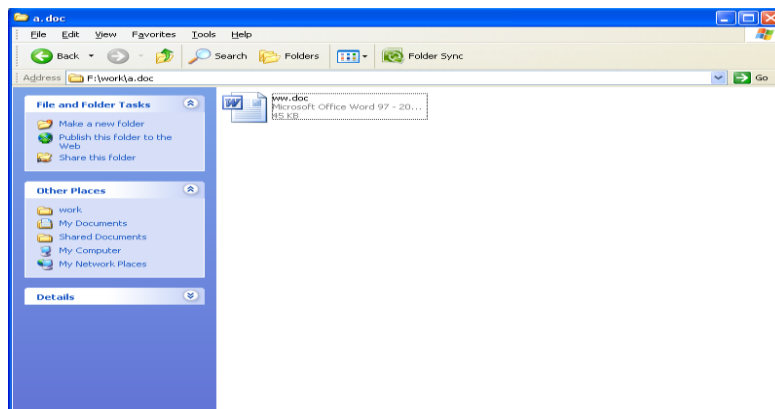


Fig 12: After the scan it can be seen that the worm file has been removed

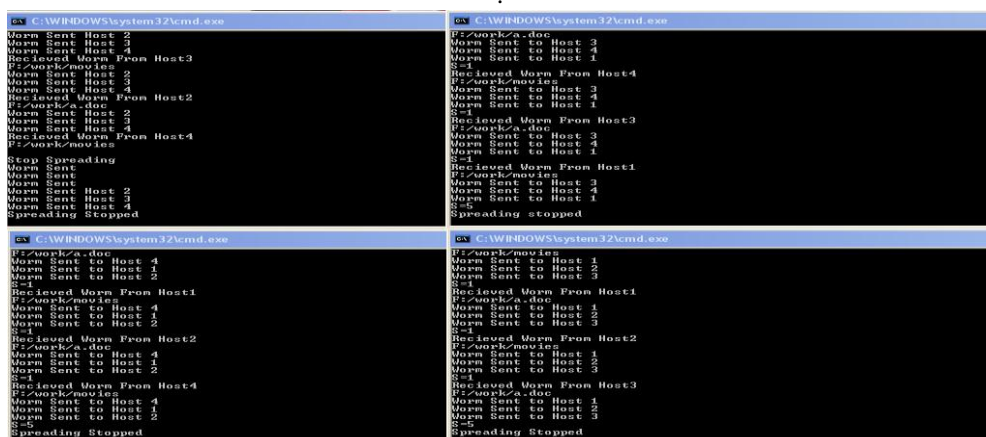


Fig 13: 'Stop Spreading' button has been pressed in host1 and the spreading stops in the network.

#### IV. CONCLUSION AND FUTURE WORK

In this Paper we have modeled the problem of combating Internet worms. We have developed a branching process model to characterize the propagation this model allows us to characterize the early phase of worm propagation on of Internet worms. The insights gained from analyzing this model also allow us to develop an effective and automatic worm containment strategy that does not let the worm propagate beyond the early stages of infection, by detecting worms in the nodes and deleting them. Thus we have demonstrated a way in which we can prevent the large scale destruction caused by worms, which in the worst case could not only cause

the system to crash but also other systems connected to it in the network, worms are not given as much importance as viruses but which are still deadly since a worm doesn't require a user to take action to compromise the computer.

For further work, we would like to propose a statistical model for the spread of topology-aware worms and subsequently design mechanisms for automatic containment of such worms. Topology aware worms are more intelligent and adaptive to network topologies than other worms and thus are difficult to control.

### References

- [1] <http://virusall.com/computer%20worms/worms.php>
- [2] [http://www.us-cert.gov/reading\\_room/virus.html](http://www.us-cert.gov/reading_room/virus.html)
- [3] <http://webcache.googleusercontent.com/search?q=cache:http://madchat.awired.net/vxdevl/papers/avers/taxonomy.pdf>
- [4] <http://www1.icsi.berkeley.edu/~nweaver/containment/>
- [5] <http://ants.iis.sinica.edu.tw/3BkMJ9lTeWXTSrrvNoKNFDxRm3zFwRR/17/04483668.pdf>
- [6] [http://ieeexplore.ieee.org/xpl/freeabs\\_all.jsp?arnumber=4358715&count=3&index=3&login=3&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs\\_all.jsp%3Farnumber%3D4358715](http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4358715&count=3&index=3&login=3&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D4358715)
- [7] <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=4358715&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F8858%2F4509574%2F04358715.pdf%3Farnumber%3D4358715>
- [8] <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=5342174&contentType=Conference+Publications>
- [9] <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=4358715&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F8858%2F4509574%2F04358715.pdf%3Farnumber%3D4358715>
- [10] <http://www.computer.org/csdl/trans/tq/2007/02/q0119-abs.html>
- [11] <http://java.sun.com/j2se/1.5.0/docs/api/java/util/Scanner.html>



## Privacy Preserving Clustering on Distorted data

Thanveer Jahan<sup>1</sup>, Dr.G.Narasimha<sup>2</sup>, Dr.C.V.Guru Rao<sup>3</sup>

<sup>1</sup>(M.Tech, (Ph d(CSE)), JNTU, Kukatpally ,Hyderabad, A.P,India)

<sup>2</sup>(Assistant Professor (CSE), JNTU, Kukatpally, Hyderabad, A.P, India)

<sup>3</sup>(HOD (CSE), SR Engineering College, Warangal, A.P, India)

---

**Abstract:** In designing various security and privacy related data mining applications, privacy preserving has become a major concern. Protecting sensitive or confidential information in data mining is an important long term goal. An increased data disclosure risks may encounter when it is released. Various data distortion techniques are widely used to protect sensitive data; these approaches protect data by adding noise or by different matrix decomposition methods. In this paper we primarily focus, data distortion methods such as singular value decomposition (SVD) and sparsified singular value decomposition (SSVD). Various privacy metrics have been proved to measure the difference between original dataset and distorted dataset and degree of privacy protection. The data mining utility k-means clustering is used on these distorted datasets. Our experimental results use a real world dataset. An efficient solution is achieved using sparsified singular value decomposition and singular value decomposition, meeting privacy requirements. The accuracy while using the distorted data is almost equal to that of the original dataset.

**Keywords-** Privacy Preserving, Data Distortion, Singular Value Decomposition (SVD), Sparsified Singular Value Decomposition (SSVD), k-means clustering.

---

### I. Introduction

The rapid increase of applications in data mining has raised a major concern in corporations. Private information of innocent people is collected and used for data mining purpose. In this modern technology, data of various kinds are collected and exchanged at an unprecedented speed and scale. A latest and fruitful direction for future research is to efficiently discover valuable information from large datasets and to develop techniques that incorporate privacy concerns. Now a day's data is an important asset of companies, governments, and research institutions [10] and is used for various public and private interests. Data is sensitive to privacy issues. Defense applications, financial transactions, healthcare records and network communication traffic are few of examples. Preserving Privacy in sensitive domains has become a major concern in data mining applications. Many data mining applications would not be acceptable, without an acceptable level of privacy of sensitive information. Data can be collected at centralized or distributed location. In centralized location, major concern is to shield the exact values of the attributes from the data analysts, where as in distributed locations data storage patterns are different i.e. they are horizontally distributed or vertically distributed [1, 8]. There has been much research on privacy preserving data mining (PPDM) based on data perturbation or data distortion, randomization and secure multi party computations. The goal of privacy-preserving data mining techniques is used to hide sensitive or confidential data values from an unauthorized user and preserve data patterns. These patterns and semantics are used to build a valid decision model on distorted data sets. Different data mining techniques such as classification, clustering etc are proposed for privacy protection in data processing. The best scenario is to construct a data pattern model on distorted data equivalent to or better than an original data.

There are two approaches in this case i.e. to distort the data so that the analysts are unaware of original data and the second approach is to modify the data mining algorithms. In this paper we propose the first approach the analysts uses distorted dataset transformed into data matrix  $\bar{D}$ , not the original dataset  $D$ . The matrix  $\bar{D}$  cannot be used to reconstruct the original matrix  $D$ , without knowing the error part  $E = D - \bar{D}$ . The analysts are unable to know attributes (columns) of original attributes and apply data mining algorithms on it. In this way data privacy preservation is premised on the maintenance of data analytical values. We transformed original dataset into distorted dataset to protect privacy. Among the widely used approach is Singular value decomposition (SVD) and its derivative Sparsified Singular value Decomposition (SSVD) are the one most popular techniques to address issues. SSVD was first introduced by Gao and Zhang in [4] to reduce cost and enhance performance of SVD in text retrieval applications. Xu et al. applied SVD and SSVD methods in terrorist analysis system [15,16]. SSVD was further studied in [5] in which structural partition strategies proposed to partition data into submatrices. In Ref. [11] privacy preserving clustering in singular value decomposition (SVD) was proposed and the results proved that accuracy of original and distorted dataset are equivalent. In our work, we take a closer look to perform data distortion by singular value decomposition and

sparsified singular value decomposition. Thus, data mining techniques k-means clustering is applied on the distorted dataset to attain inherent property of privacy protection.

The remaining part of the paper is organized as follows; Section 2 briefly introduces related work on data analysis system and data distortion methods: SVD and sparsified SVD and k-means clustering. Section 3 discusses the various data perturbation metrics. The experiments are carried out and the results are presented and discussed in Section 4. We finally sum up this paper and bring our future plans in Section 5.

## II. Related Work

### 2.1 Privacy preserving data mining

There has been a raising concern for disclosure of security and privacy, as the data mining techniques gain popularity and widely used in business and research. Two parties having private data wish to work in collaboration by to other party. Indeed, neither party shares their private data. In such cases privacy preserving data mining (PPDM) have major significance. PPDM develops algorithms for modifying the original data in a way that data and knowledge remain private even after mining process [12]. Common techniques include data perturbation, blocking feature values, swapping tuples etc. PPDM scheme should able to maximize the degree of data modification to retain the maximum data utility level.

### 2.2 Analysis system and data distortion

A simplified model of data analysis system consists of two parts, the data manipulation and the data analysis as illustrated in Fig 1. The original data is completely manipulated by the authorized user's or data owner using data distortion process i.e. matrix decomposition method. Data distortion is one of the important parts in many privacy preserving data mining tasks. The distorted methods must preserve data privacy and at the same time must keep the utility of the data after distortion. The data distorted or perturbed data is collected by analysts to perform all actions such as clustering etc. The protected data maintains privacy as analysts is unknown with actual data values. The classical data distortion methods are based on random value perturbation and are applied [8]. Singular value decomposition (SVD) is a popular method in data mining and information retrieval

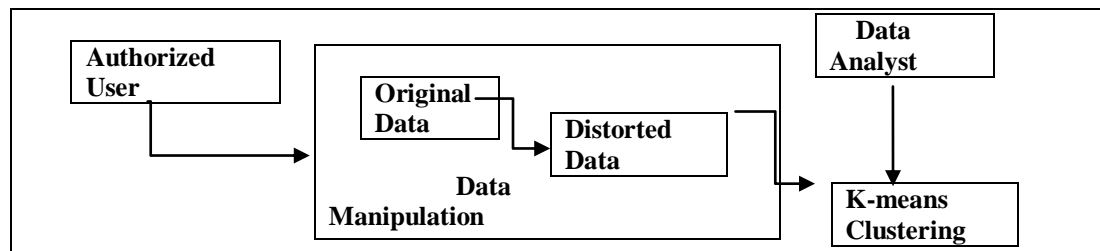


Figure 1. A Data Analysis system for Clustering

[9].SVD has numerous applications in data mining, information retrieval and image compression in which it is often used to approximate a given a matrix by a lower rank matrix with minimum distance between them. SVD is used to reduce dimensionality of the original dataset  $D$ . A sparse matrix  $D$  of dimension  $p \times q$  represents the original dataset. The rows and columns are the data objects and attributes. The singular value decomposition of the matrix  $D$  is [3].

$$D = U S V^T$$

Where  $U$  is an  $p \times p$  orthonormal matrix,  $S = \text{diag} [\sigma_1, \sigma_2, \dots, \sigma_s]$  ( $s = \min\{p, q\}$ ) is an  $p \times q$  diagonal matrix whose nonnegative diagonal entries are in a descending order, and  $V^T$  is an  $q \times q$  orthonormal matrix. The number of nonzero diagonals of  $S$  is equal to the rank of the matrix  $D$ . The singular values in the matrix  $S$  are arranged in a descending order. The SVD transformation has property that the maximal variation among objects is captured in the first dimension, as  $\sigma_1 \geq \sigma_i$  for  $i \geq 2$ . The remaining variations are captured similarly in the second dimension and so on. Thus, a transformed matrix with a lower dimension can be constructed to represent the original matrix i.e.

$$D_r = U_r S_r V_r^T$$

Where  $U_r$  contains the first  $r$  columns of  $U$ ,  $S_r$  contains the first  $r$  nonzero diagonals of  $S$  and  $V_r^T$  contains the first  $r$  rows of  $V^T$ . The rank of the matrix  $D_r$  is  $r$  and with  $r$  being small, the dimensionality of the dataset has been gradually reduced from  $\min\{p, q\}$  to  $r$  (assuming all attributes are linearly dependent).  $D_r$  is proved to be best  $r$  dimensional approximation of  $D$  in the sense of Frobenius norm. In data mining applications the use of  $D_r$  to represent  $D$  has important function. The removed error part  $E_r = D - D_r$  can be considered as the noise in the original dataset  $D$  [8]. Mining on reduced dataset  $D_r$  yield better results than on original dataset  $D$ . Thus, the distorted data  $D_r$  can provide effective protection for data privacy. Sparsified SVD is a data distortion method

better than a SVD in preserving privacy. After reducing rank of the SVD matrices, we set small size entries which are smaller than a certain threshold  $\epsilon$  in  $U_r$  and  $V_r^T$  to zero. This operation is referred as a dropping operation [4]. Thus, drop  $u_{ij}$  in  $U_r$ , if  $|u_{ij}| < \epsilon$  and  $v_{ij}$  in  $V_r^T$ , if  $|v_{ij}| < \epsilon$ . Let  $\overline{U}_r$  denote  $U_r$  with dropped elements and  $\overline{V}_r^T$  denote  $V_r^T$  with dropped elements, the distorted data matrix  $D_r$  is represented as

$$D_r = \overline{U}_r S \overline{V}_r^T,$$

The sparsified SVD method is equivalent to further distorting the dataset  $D_r$ . Denote

$$E_c = D_r - \overline{D}_r,$$

$$D = \overline{D}_r + E_r + E_c,$$

The data provided to analysts is  $\overline{D}_r$  which is twice distorted in the sparsified SVD method. The sparsified SVD was proposed by Gao and Zhang in [4] for reducing storage cost and enhancing performance of SVD in text retrieval applications.

### 2.3 K-means Clustering

Clustering is a well-known problem in statistics and engineering, namely, how to arrange a set of vectors (measurements) into a number of groups (clusters). Clustering is an important area of application for a variety of fields including data mining, statistical data analysis and vector quantization [6]. The problem has been formulated in various ways in the machine learning, pattern recognition optimization and statistics literature. The fundamental clustering problem is that of grouping together (clustering) data items that are similar to each other. Given a set of data items, clustering algorithms group similar items together. Clustering has many applications, such as customer behavior analysis, targeted marketing, forensics, and bioinformatics.

## III. Data Perturbation Metrics

In literature privacy metrics have been proposed in [2, 10]. In Ref. [2] the metrics are incomplete and is proved in Ref. [8]. It is important to know the density function of each attribute a priori, which may be difficult to obtain for the real world datasets. We propose some privacy measures which depend on the original matrix  $D$  and its distorted matrix  $D$ .

### 3.1 Value difference (VD)

The elements of data matrix change after distortion. The value difference (VD) of the datasets is represented by relative value difference in the Forbenius norm. VD is the ratio of the Forbenius norm of the difference of  $D$  and  $|D|$  to the Forbenius form of  $D$ .

$$VD = \frac{\|D - |D|\|_F}{\|D\|_F}.$$

### 3.2 Position difference

several metrics are used to measure position difference of the data elements. RP is used to denote average change of order of all attributes. The order of the element changes after distortion. Dataset  $D$  has  $n$  data objects and  $m$  attributes.  $Ord_j^i$  denotes the ascending order of the  $j$ th element in attributes  $i$ , and  $\overline{Ord}_j^i$  denotes the ascending order of the distorted element  $D_{ij}$ . Then,  $RP = (\sum_{i=1}^m \sum_{j=1}^n |Ord_j^i - \overline{Ord}_j^i|) / (m * n)$

RK represents the percentage of elements that keep their orders of value in each column after the distortion.

$$RK = (\sum_{i=1}^m \sum_{j=1}^n RK_j^i) / (m * n),$$

$$RK_j^i = \begin{cases} 1, & \text{if } Ord_j^i = \overline{Ord}_j^i, \\ 0, & \text{otherwise.} \end{cases}$$

The metric CP is used to define the change of order of the average value of attributes:

$$CP = (\sum_{i=1}^m |OrdDV_i - \overline{OrdDV}_i|) / m,$$

where  $OrdDV_i$  is the ascending order the average value of attribute  $i$  while  $\overline{OrdDV}_i$  denotes ascending order after distortion. Like RK, we define CK to measure the percentage of the attribute that keeps their order of average value after distortion.

$$CK = (\sum_{i=1}^m CK^i) / m,$$

$$\text{where } CK^i = \begin{cases} 1, & \text{if } OrdDV_i = \overline{OrdDV}_i, \\ 0, & \text{otherwise} \end{cases}$$

The higher the value of RP and CP and the lower the value of RK and CK, the more privacy can be preserved. In order to be fair for a dataset, privacy metrics are calculated as shown in the Table 1. The value of VD, RP, CP is more in SSVD than in SVD. Among the four distortion methods SSVD is better than SVD to preserve privacy as shown in Fig 4.

Table 1. Comparison of Privacy Metrics for distortion methods

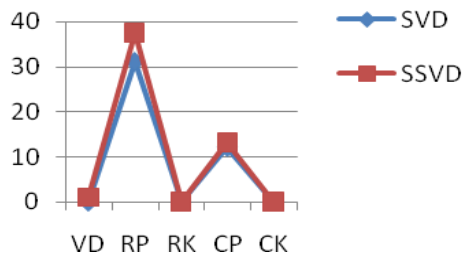


Figure 2. Performance of privacy metrics

		Privacy Metrics			
DATA	VD	RP	RK	CP	CK
Org	—	—	—	—	—
SVD	0.0525	31.2	0.0251	12.2	0.12
SSVD	1.0422	37.5	0.0066	13.1	0.05

IV. Experiments And Results

We conduct experiments on real data set having 100 data points. For a real world dataset, we downloaded information about 100 terrorists (q), 42 attributes (p) such as age, place, relationship etc. The original matrix is of dimension 100x42.

4.1 Proposed Algorithm

**Input:** Data matrix D, No of clusters k,

**Output:** Distorted Data matrix  $\overline{D}_r$ , Clusters

**Step 1:** Finding sensitive or confidential attributes (p<sub>i</sub>) i= 0,1,.....41 in D.

**Step 2:** Form the matrix C. C = [p<sub>0</sub>,p<sub>1</sub>,p<sub>2</sub>,...p<sub>41</sub>]

**Step 3:** Apply SVD to the matrix C.

$$SVD(C) = U S V^T,$$

Then distorted matrix  $C_r = U_r S_r V_r^T$

**Step 4:** Then, apply SSVD to matrix C<sub>r</sub>

Choosing the rank r and dropping

Threshold ε as 10<sup>-3</sup>

Then Distorted matrix SSVD (C<sub>r</sub>) =  $\overline{U}_r S_r \overline{V}_r^T$

**Step 4:** Update  $\overline{C}_r$  in D, gives  $\overline{D}_r$

**Step 5:** Generate Clusters for sensitive attribute in  $\overline{D}_r$ .

Table 2: Data objects and Clusters

Clusters(k)	Original data		SVD				SSVD		
	2	3	4	2	3	4	2	3	4
Data objects(points)	2	1	2	2	1	2	2	2	1
	1	3	3	1	3	1	2	2	2
	1	3	3	1	3	1	2	3	2
	1	3	1	1	3	1	2	3	2
	1	3	1	1	3	1	2	3	2
	2	1	2	2	1	2	2	2	2
	1	3	1	1	3	1	2	3	2
	1	3	1	1	3	1	2	3	2
	1	3	1	1	3	1	2	3	2
	1	3	1	1	3	1	2	3	2

The illustration of the above method is represented for 10 data objects is shown in the Table 2. we analyzed a specific number of clusters ranging from 2 to 4 clusters. The effectiveness is measured in terms of the proportion of the points that are grouped in the same clusters after we apply a transformation on data, such points as legitimate points. Considering the transformation attributes as relationship with terrorist group. k denotes the number of clusters to group the data objects. For the clusters k=2 and k=3 the data objects grouped in original dataset and in SVD dataset are exactly same. In SSVD data objects are effectively grouped when, compared to original and SVD data set for k=2,3,4.

4.2 Measuring Accuracy

The efficiency is measured on the number of data points those are legitimate and are grouped in the original and distorted datasets. k- means clustering do not consider noise.

A Misclassification Error is used to concentrate on a potential problem where the data point from a cluster migrates to a different cluster.

$$ME = 1/ N * \sum_{i=1}^k (|Cluster_i(D)| - |Cluster_i(D_r)| - |Cluster_i(D_r)|)$$

Table 3. Results of Misclassification Error

Data objects(points)	Original data set			Distorted dataset-SVD			Distorted dataset-SSVD		
	K=2	K=3	K=4	K=2	K=3	K=4	K=2	K=3	K=4
10	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00
100	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00

Misclassification error must be 0% where N represents the number of point in the original dataset, k is the number of clusters under analysis and |Cluster<sub>i</sub>(D)| represents the number of data points legitimate in the ith cluster in dataset D. The results are tabulated in the Table 3. The cluster analysis yields good results for the original and distorted datasets using SVD and SSVD distortion techniques. The results suggest that our techniques perform well to achieve feasible solution. The accuracy of distorted data is same as original data. Thus, a complete privacy can be obtained in k-means cluster analysis and is also proved in privacy metrics.

V. Conclusion

We propose a better approach for a data analysis system to use data distortion techniques: singular value decomposition (SVD) and Sparsified SVD to preserve privacy. We have presented privacy preserving data mining application which distorts original dataset to meet privacy requirements. Experimental results show the effectiveness by measuring accuracy of original data and distorted data. It has proved that high degree of data distortion can maintain high level of data utility using k-means clustering. Future work may address other scenarios to protect data along with different data mining algorithms.

References

- [1] B.Gillburd, A. Schuster and R.Wolff. "K- TTP: A new Privacy model for large scaled distributed environments". In *Proceedings of the 10<sup>th</sup> ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (kdd'04)*, Seattle, WA, USA, 2004.
- [2] D.Agrawal and C .C.Agarwal "On the and quantification of privacy –Preserving data minng algorithms". In *Proceedings of the 20th ACM SIGACT-SIGMOD*.
- [3] G.H.Golub and C.F.van Loan. *Matrix Computation*, John Hopkins Univ., 3rd ED., 1996.
- [4] J.Gao, J. Zhang. "Sparsification strategies in latent semantic indexing", In *Proceedings of the 2003 Text Mining Workshop, M.W. Berry and W.M. Pottenger*, (ed.), pp. 93-103, San Francisco, CA, May 3, 2003.
- [5] J.Wang, W.Zhong,S. Xu,J.Zhang. "Selective Data Distortion via Structural Partition and SSVD for privacy Presevation",In *Proceedings of the 2006 International Conference on Information & knowledge Engineering*,pp:114-120,CSREA Press,Las Vegas.
- [6] J. Han, M. Kamber. *Concepts and Techniques*. Morgan Kaufmann Publishers, 2001
- [7] N. Maheswari & K. Duraiswamy CLUST- SVD: Privacy preserving clustering in singular valued decomposition In *World journal of modeling and simulation vol 4(2008) No4*,pp250-256.
- [8] R.Agrawal, A.Evfimieski and R.srikanth."Information sharing across private databases". In *proceedinds of the 2003 ACM SIGMOD International Conference on management of data*, pp.86-97,san Diego,CA,2003
- [9] R.Agrawal and R.srikant"Privacy–Preserving data mining". In *proceedings of the 2000 ACM SIGMOD International Conference on management of data*, pp86-97, San Diego, CA, 2003.
- [10] S.Deewester, S.Dumais, et al."Indexing by latent semantic analysis", *J. Amer. Soc.Info.Sci*, 41:391–407, 1990.
- [11] V. S. Verykios, E. Bertino, I. Fovino, L. Provenza, Y. Saygin, and Y. Theodoridis. "State-of-the-art I privacy preserving data mining", *SIGMOD Record*, 33(1):50-57, 2004
- [12] V. S. Verykios, E. Bertino, I. Fovino, L. Provenza, Y. Saygin, and Y. Theodoridis. "State-of-the-art in privacy preserving data mining", *SIGMOD Record*, 33(1):50-57, 2004
- [13] W.Frakes and R.Baeza-Yates. *Information Retrieval: Data Structures and Algorithms*. Prentice-Hall, Englewood Cliffs,NJ,19192
- [14] V.Estwill–Castro, L Brankovic and D.L.Dowe. "Privacy in data mining". Australian Computer Society NSW Branch, Australia. Available at [www.acs.org.au/nsw/articles/199082.html](http://www.acs.org.au/nsw/articles/199082.html)
- [15] S. Xu, J. Zhang, D. Han, J.Wang. Data distortion for privacy protection in a terrorist analysis system. in: *Proceedings of the 2005 IEEE International Conference on Intelligence and Security Informatics*, 2005, 459–464.
- [16] S. Xu, J. Zhang, D. Han, J. Wang. Singular value decomposition based data distortion strategy for privacy protection. *Knowledge and Information Systems*, 2006, 10(3): 383–397.

## Data Allocation Strategies for Leakage Detection

Sridhar Gade<sup>1</sup>, Kiran Kumar Munde<sup>2</sup>, Krishnaiah.R.V.<sup>3</sup>

<sup>1</sup>Department of CSE, DRK Institute of Science & Technology, Ranga Reddy, Andhra Pradesh, India

<sup>2</sup>Department of CSE, DRK College of Engineering & Technology, Ranga Reddy, Andhra Pradesh, India

<sup>3</sup>Principal Department of CSE, DRK Institute of Science & Technology, Ranga Reddy, Andhra Pradesh, India

---

**Abstract:** Data plays a pivotal role in IT systems. Especially when sensitive data has to be sent to other places through trusted agents, it is very challenging and important to detect leakage when they deliberately leak it to others. The scenario where a distributor gives sensitive data to his trusted agents and the data is intentionally leaked to others. The distributor should identify or detect this leakage and its means that is who leaked it as well. This is the problem this paper intended to solve. Towards this we propose new data allocation strategies for improving the probability of detecting leakages accurately. The system should detect leakage correctly and the means as well as against to the leakage by other means. The proposed methods do not rely on the alterations of released data. It is also possible to inject “looks genuine but fake” data in order to improve the probability of detecting leakage and tracing the party who actually leaked it.

**Index Terms** – Data leakage, leakage detection model, data allocation strategies, fake records

---

### I. Introduction

In business applications data can be transmitted securely through network. Due the emergence of many cryptographic algorithms, end to end security methods, it is possible to send data across the machines with full security. However, there is possibility for online attacks. The security in this case depends on the strength of cryptographic algorithms. This is one side of the coin. The other side of the coin is that in business scenarios people need to send information through trusted parties. In this case the distributor of data is fully aware that the data leakage may happen. However, the distributor has trust over the agents who carry his data to other destinations to which the distributed is associated for business purposes. Provided this scenario, the distributor can only hope genuine behavior from his trusted agents. What if the agents behave quite opposite to the belief of distributor? is the important question answered by this paper. When data is leaked by trusted agents, there should be some way to identify it and prove it. Unfortunately this is the job difficult to achieve. Other scenarios where data has to be distributed through trusted agents include patients records may be given by hospital; sharing of data is required among companies with partnerships; an enterprise may decide to outsource its data process and hence need to handover the valuable data to other party. In all these scenarios, the provider of sensitive data is considered as distributor.

The aim of this paper is to detect leakage no matter who is involved in leakage and proving that data has been leaked. One naïve technique is to modify and make it “less sensitive” before actually giving to trusted agents. The alterations may be done by introducing noise in the data or replace certain values and remember them [1]. But it is not good practice to modify original data. To ensure this data leakage detection is done using watermarking traditionally. A unique symbol is embedded into each copy that has been distributed. When such symbol is found with any unauthorized person, it is the proof that data leakage has been occurred. Watermarking is effective in leakage detection. However, it involves modification of original data. There is security problem with this. When receiver is malicious, it can be destroyed. This paper proposes a novel technique that ensures that data leakage is detected without actually modifying data. It is achieved like this. When data is given trusted agents and found that leaked to other parties who are not authorized, the proposed system can identify the leakage and also identify the means of leakage. The distributor can find out the likelihood of data leakage and means of leakage. This is achieved by using algorithms for distributing objects to trusted agents in such a way that it improves possibility of data leakage detection. The algorithms also consider adding “fake” objects to the set of distributed objects in order to improve the possibilities of detecting leakage. The fake objects are not at all related to real objects but appear so in the eyes of agents. The fake objects are indirectly acting as watermark in this case. When any agent finds fake objects in somewhere, he can suspect that particular agent to be guilty of leakage.

### II. Problem Statement

We take a hypothetical problem in which a distributor owns a set of objects. He wants to share those objects with a group of humans known as trusted agents. The distributor does not want the objects shared with agents to be leaked to third parties. The objects may be of any type of any size. They could be records in

---

relational tables or files in file system. Agents get some or all of the objects based on the requirement. The trusted agents are believed to be trusted. However, when they involve in any fraud activities the data gets leaked. This is the problem addressed in this paper. Towards this a guilt model is proposed. After receiving objects from distributor the trusted agents may misbehave and leak the data objects to some third party. When the leaked objects are viewed by distributor through any means, he can suspect that those objects are given by one of his trusted agents. The aim of this paper is to prove that the data is leaked by so and so agent by proposing data allocation strategies.

### **III. Related Work**

Data leakage detection has been around with respect to IT systems. Security threats like impersonation, hacking, intrusion, eaves dropping and VIRUS can be prevented using security software available. All forms of electronic exchange of data have security mechanisms in place. However, guilt detection in a scenario where data is handed over to trusted agents (humans) and expect them to transfer data to intended recipients is a challenging task. The following review establishes facts in line with this problem. Data provenance problem has been around and it is related to data origin and the originality of data. In [2] a data provenance problem is discussed which is relevant to the guilt detection problem presented in this paper. By tracing the origin of given objects does mean that tracing the probability of guilt. Further research in this field is presented in a tutorial [3] which reviews all possible causes and probability of proving data provenance problem. The solutions in this area are domain specific and they are pertaining to data warehouses [4] assuming to have prior know how on data sources and the way data is created. In this paper, our problem formulation is simple and general and does not alter the original objects to be distributed. When a set of objects are to be distributed through trusted agents, we formulate objects that are not changed as opposed to watermarking. Lineage tracing is performed without using watermarking here. Watermarking technology has been around to protect intellectual property of people that is in electronic format. However, it needs the object that needs to be protected to be modified in order to embed some sort of watermark for security reasons. When watermarked image is tampered that is made well known to the distributor thus establishing the fraud taken place. Watermarking can be used with images [4], audio [5] and video [6]. These media's digital data has redundancy. Relational data can also be protected using something similar to watermarking. This is achieved by inserting some marks into the data for security reasons. This kind of research is reviewed in [7], [8], and [9].

Our approach in this paper and watermarking are similar in the sense of providing identification of information for originality. However, they are totally different as our approach does not need to alter objects to be distributed as opposed to watermarking. There are other research works that focused on enabling IT systems to ensure that only intended receivers will receive data. It is achieved access control policies proposed in [10] and [11]. These policies help in protecting data when it is transferred and detect leakage of data as well. However, they are very restrictive in nature and it is impossible for them to satisfy requests from agents.

### **IV. Agent Guilt Model**

Probability of guilt  $\Pr \{G_i|S\}$  can be computed by estimating the probability that the a target can guess objects in "S". The proposed guilt model makes two assumptions. The first assumption is that the source of a leaked object can be of any agent. The second assumption is that An object which is part of set of objects distributed can only be obtained from one of the agents or through other means. With these assumptions the probability of guilt is computed as

$$P_r\{U_i \text{ leaked } t \text{ to } S\} = \begin{cases} \frac{1-p}{|V_t|}, & \text{if } U_i \in V_t \\ 0, & \text{otherwise} \end{cases}$$

### **V. Analysis Of Guilt Model**

In this section our guilt modeling is analyzed to see whether it works correctly. Two simple scenarios we take and in each case all distributed objects are obtained by target i.e.,  $T=S$ . Assuming that T has 16 objects. Out of them only 8 are given to U2 and all of them are given to U1. Probability of guild for both the users and agents is calculated. The results are as given in fig. 1, 2, 3 and 4.

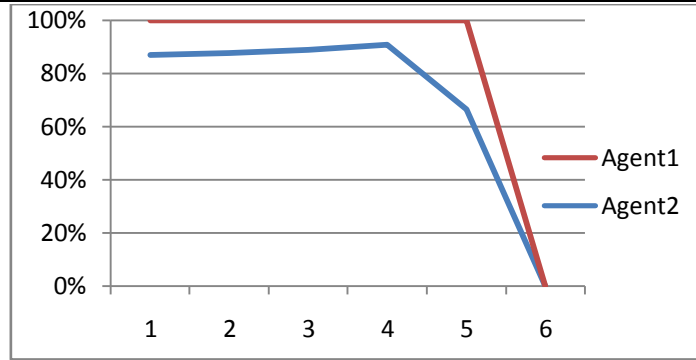


Fig. 1 – Guilt probability as a function with  $p = 0.5$

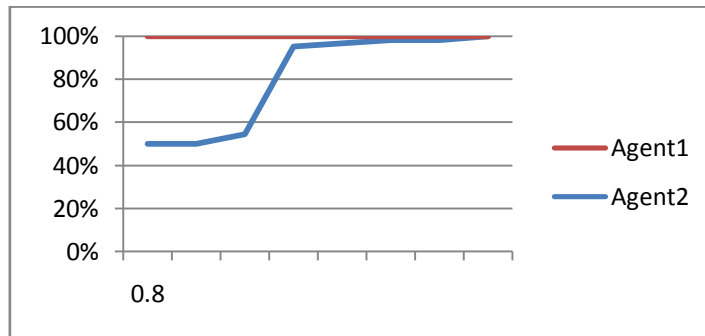


Fig. 2 – Guilt probability as a function with  $p = 0.2$  (Overlap b/w S and R2)

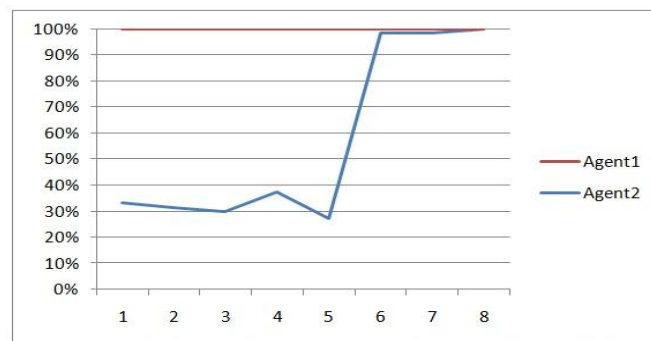


Fig. 3 – Guilt probability as a function with  $p = 0.5$  (Overlap b/w S and R2)

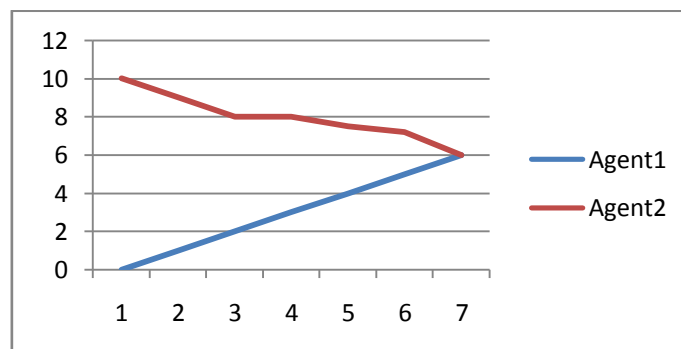


Fig. 4 – Guilt probability as a function with  $p = 0.9$  (Overlap b/w S and R2)

As can be seen in above figures when  $p$  value is 0 it is not likely that all 16 objects are guessed by target. Each agent has some leaked and approaches 1. The probability that U2 is guilty decreases when  $p$  value increases. However, the probability of U2's guilt remains more and close to 1 as agent 2 has 8 values that are not known to other agent. When  $p$  value approaches 1, the agent's probability of guilt becomes zero.

### VI. Data Allocation Problem Description

Data allocation is the main focus in this paper. Distributor is supposed to allocate data objects to trusted agents intelligently. Two types of requests are handled namely sample and explicit. While distributing objects,



fake objects that mimic real objects are created and given to agents along with real objects. The fake objects are created in such a way that the agent's information who carries the objects is kept in those objects. The intention of creating fake objects is to maximize the probabilities of detecting guilt agents. The fake objects created are given to trusted agents along with the actual and real objects. The fake objects are somehow associated with information of agent who carries them. The whole thing is transparent to agents as they can't distinguish between the fake and real objects. The process of creating fake objects has to be done carefully and intelligently. While creating fake objects, the distributor can also specify certain limit for fake objects so as to ensure that the agents do not suspect some of the objects as fake. The fake objects look like real objects and the agents have no knowledge as to how to distinguish between fake and real objects.

In order to optimize the data allocation process a distributor must have a constraint and also an objective. The constraint is that distributor has to send objects required by agents. Objective of distributor is having the ability to detect an agent when objects are leaked. The distributor's objective is calculated as  $\Delta(i,j) = P_r\{G_i|R_i\} - P_r\{G_j|R_j\}$   $i,j = 1, \dots, n$

### VII. Data Allocation Techniques

The data allocation strategies used to solve the problem of data distribution as discussed in previous sections exactly or approximately are provided in the form of various algorithms. The algorithms are provided here.

---

Algorithm 1 Allocation for Explicit Data Requests (EF)

---

```

Input:  $R_1, \dots, R_n, \text{cond}_1, \dots, \text{cond}_n, b_1, \dots, b_n, B$ 
Output:  $R_1, \dots, R_n, F_1, \dots, F_n$ 
1:  $R \leftarrow \phi$   $\triangleright$  Agents that can receive fake objects
2: for  $i=1, \dots, n$  do
3: if  $b_i > 0$  then
4:  $R \leftarrow R \cup \{i\}$ 
5:  $F_i \leftarrow \phi$ 
6: while  $B > 0$  do
7:  $i \leftarrow \text{SELECTAGENT}(R, R_1, \dots, R_n)$ 
8:  $f \leftarrow \text{CREATEFAKEOBJECT}(R_i, F_i, \text{cond}_i)$ 
9:  $R_i \leftarrow R_i \cup \{f\}$ 
10:  $F_i \leftarrow F_i \cup \{f\}$ 
11:  $b_i \leftarrow b_i - 1$ 
12: if  $b_i = 0$  then
13:  $R \leftarrow R \setminus \{R_i\}$ 
14:  $B \leftarrow B - 1$ 
    
```

---

Fig. 5 – Allocation for explicit data requests

It is a general algorithm that is used by other algorithms.

---

Algorithm 2 Agent Selection for e-random

---

```

1: function SELECTAGENT( $R, R_1, \dots, R_n$ )
2:  $i \leftarrow$  select at random an agent from  $R$ 
3: return  $i$ 
    
```

---

Fig. 6 – Agent selection for e-random

This algorithm actually performs random selection of objects.

---

Algorithm 3 Agent selection for e-optimal

---

```

1: function SELECTAGENT ( $R, R_1, \dots, R_n$ )
2:  $i \leftarrow \text{argmax}_{i: R_i \in R} \left( \frac{1}{|R_i|} - \frac{1}{|R_i| + 1} \right) \sum_j |R_i \cap R_j|$ 
3: return  $i$ 
    
```

---

Fig. 7 – Agent selection for e-optimal

This algorithm is meant for making a greedy choice of choosing an agent that causes improvement in the sum-objective.

---

Algorithm 4: Allocation for Sample Data Requests(SF)

---

```

Input:  $m_1, \dots, m_n, |T|$   $\triangleright$  Assuming  $m_i \leq |T|$ 
Output:  $R_1, \dots, R_n$ 
1:  $a \leftarrow 0_{|T|}$   $\triangleright$   $a[k]$ : number of agents who have received object  $t_k$ 
2:  $R_1 \leftarrow \phi, \dots, R_n \leftarrow \phi$ 
3: remaining  $\leftarrow \sum_{i=1}^n m_i$ 
4: while remaining  $> 0$  do
5: for all  $i=1, \dots, n: |R_i| < m_i$  do
6:  $k \leftarrow \text{SELECTOBJECT}(i, R_i)$   $\triangleright$  May also use additional parameters
7:  $R_i \leftarrow R_i \cup \{t_k\}$ 
8:  $a[k] \leftarrow a[k] + 1$ 
9: remaining  $\leftarrow$  remaining-1
    
```

---

Fig. 8 – Allocation for sample data requests

This is a general algorithm that is required by other algorithms.

Algorithm 5: Object Selection for s-random

```

1: function SELECTOBJECT (i, Ri)
2: k ← select at random an element from set {k' | tk ∈ Ri}
3: return k
    
```

Fig. 9 – Object selection for s-random

This algorithm is meant for random selection of objects.

Algorithm 6: Object Selection of s-overlap

```

1: function SELECTOBJECT (i, Ri, a)
2: K ← { k | k = argmin a[k'] }
3: k ← select at random an element from set {k' | k' ∈ K ∧ tk ∈ Ri}
4: return k
    
```

Fig. 10 – Object Selection of s-overlap

This algorithm is meant for selection of objects in s-overlap fashion.

Algorithm 7 Object Selection for s-max

```

1: function SELECTOBJECT (i, R1, ..., Rn, m1, ..., mn)
2: min_overlap ← 1 ▷ the minimum out of the maximum relative overlaps that the allocations
of different objects to Ui yield
3: for k ∈ { k' | tk ∈ Ri } do
4: max_rel_ov ← 0 ▷ the maximum relative overlap between Ri and any set Rj that the
allocation of tk to Ui yields
5: for all j=1, ..., n: j ≠ i and tk ∈ Rj do
6: abs_ov ← |Ri ∩ Rj| + 1
7: rel_ov ← abs_ov / min(mi, mj)
8: max_rel_ov ← MAX(max_rel_ov, rel_ov)
9: if max_rel_ov ≤ min_overlap then
10: min_overlap ← max_rel_ov
11: ret_k ← k
12: return ret_k
    
```

Fig. 11 – Object selection for s-max

This algorithm defines a new SELECTOBJECT() procedure, used to select objects to achieve minimum increase of maximum relative overlap among agents.

### VIII. Empirical Results

The environment used for the experiments include Windows XP OS, Java programming language, Eclipse IDE. A prototype application has been built in order to simulate the data leakage detection process. The results showed in fig. 5 and 6 show the results with respect to e-optional, e-random and no fake algorithms.

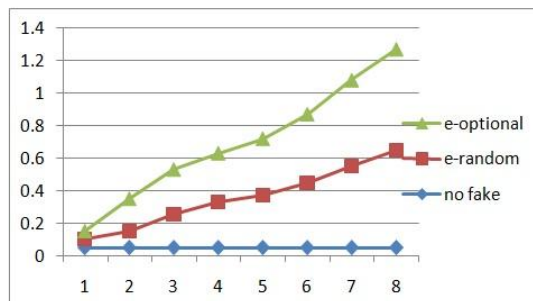


Fig. 5 – Evaluation of Explicit Data Request Algorithms (Average Metric)

Fig. 5 – Evaluation of Explicit Data Request Algorithms (Average Metric)

As can be seen in fig. 5, it shows average metric is affected by allocation of fake objects. The straight line in the graph represents that object allocation is done without fake objects.

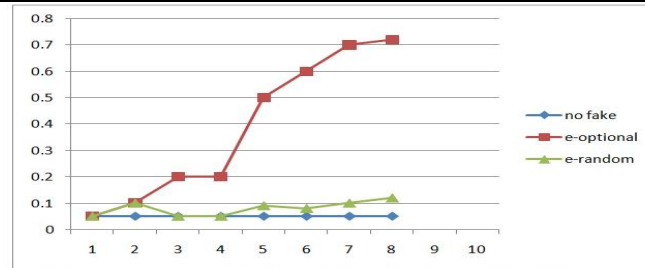


Fig. 6 – Evaluation of Explicit Data Request Algorithms (Average Min Metric)




As can be seen in fig. 6, it shows average metric is affected by allocation of fake objects. The straight line in the graph represents that object allocation is done without fake objects.

### IX. Conclusion

When the world is not perfect in conduct and behavior and you need to send sensitive data to intended recipient through trusted agents, it is essential to have monitoring on the distribution process. When sensitive data has to be sent through electronic means, there are many security systems that can protect the data when it is on transit and also ensure that it reaches only to the intended recipient in original format. This paper addresses a different problem where data transmission takes place through human beings known as trusted agents. Detecting probability of data leakage has paramount importance especially when the data is confidential and sensitive in nature. We considered a scenario where a distributor is supposed to send sensitive data through his trusted agents and needs to detect when data is leaked by trusted agents in any fashion. The establishment of the probability of leakage and identifying the agent who leaked it is a challenging task. To address this problem, we proposed data allocation strategies that are personalized in such a way that when leaked data is found somewhere, it is possible to identify the agent who leaked it as agent’s information is embedded somewhere as part of the strategies. Unlike watermarking which modifies original objects before being transmitted for security reasons, our system does not need any modification of original objects. Instead we introduce fake objects that are personalized in terms of agents who carry them. The fake objects are given to agents along with real objects that are transparent to trusted agents. When they leak the data for any reason and when distributor finds the leaked data, the proposed system helps the distributor to identify the agent who caused leakage. We implemented various algorithms that are having different data allocation strategies meant for enhancing the probabilities of distributor in identifying the leaker. In future we work on the agent guilt models that are not discussed in this paper and also enhance the distribution strategies further to make it more robust to data leakage.

### References

- [1] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression, 2002.
- [2] P. Buneman, S. Khanna, and W. C. Tan. Why and where: A characterization of data provenance. In J. V. den Bussche and V. Vianu, editors, Database Theory - ICDDT 2001, 8th International Conference, London, UK, January 4-6, 2001, Proceedings, volume 1973 of Lecture Notes in Computer Science, pages 316–330. Springer, 2001.
- [3] P. Buneman and W.-C.Tan. Provenance in databases. In SIGMOD '07: Proceedings of the 2007 ACM SIGMOD international conference on Management of data, pages 1171–1173, New York, NY, USA, 2007. ACM.
- [4] J. J. K. O. Ruanaidh, W. J. Dowling, and F. M. Boland. Watermarking digital images for copyright protection. *I.E.E. Proceedings on Vision, Signal and Image Processing*, 143(4):250–256, 1996.
- [5] S. Czerwinski, R. Fromm, and T. Hodes. Digital music distribution and audio watermarking.
- [6] F. Hartung and B. Girod. Watermarking of uncompressed and compressed video. *Signal Processing*, 66(3):283–301, 1998.
- [7] R. Agrawal and J. Kiernan. Watermarking relational databases. In *VLDB '02: Proceedings of the 28th international conference on Very Large Data Bases*, pages 155–166. VLDB Endowment, 2002.
- [8] F. Guo, J. Wang, Z. Zhang, X. Ye, and D. Li. *Information Security Applications*, pages 138–149. Springer, Berlin / Heidelberg, 2006. An Improved Algorithm to Watermark Numeric Relational Data
- [9] Y. Li, V. Swarup, and S. Jajodia. Fingerprinting relational databases: Schemes and specialties. *IEEE Transactions on Dependable and Secure Computing*, 02(1):34–45, 2005.
- [10] P. Bonatti, S. D. C. di Vimercati, and P. Samarati. An algebra for composing access control policies. *ACM Trans. Inf. Syst. Secur.*, 5(1):1–35, 2002.
- [11] S. Jajodia, P. Samarati, M. L. Sapino, and V. S. Subrahmanian. Flexible support for multiple access control policies. *ACM Trans. Database Syst.*, 26(2):214–260, 2001.

	Sridhar Gade is a student of DRK Institute of science and Technology, Ranga Reddy, Andhra Pradesh, India. He has received M.Sc degree in Computer Science and M.Tech Degree in Computer Science and Engineering. His main research interest includes Data Mining ,Networking
	Kiran Kumar Mundeis is a student of DRK College of Engineering & Technology, Ranga Reddy, Andhra Pradesh, India. He has received M.C.A and M.Tech Degree in Computer Science and Engineering. His main research interest includes Data Mining ,Software Engineering.
	Dr. R.V. Krishnaiah (Ph.D) is working as Principal at DRK INSTITUTE OF SCIENCE & TECHNOLOGY, Hyderabad, AP, INDIA. He has received M.Tech Degree (EIE & CSE). His main research interest includes Data Mining, Software Engineering.

## Securing Group Communication in Partially Distributed Systems

<sup>1</sup>Pankesh Bamotra, <sup>2</sup>Prashant Dwivedi, <sup>3</sup>Nishant Gupta, <sup>4</sup>Rajat Pandey

(<sup>1,2,3,4</sup> Computer Science & Engineering, Vellore Institute of Technology, India)

---

**Abstract :** This paper deals with symmetric key exchange in partially distributed systems. Unlike the traditional distributed approaches for key exchange utilized when we have a centralized KDC (Key Distribution Center), which is a trusted server responsible for key exchange between the users involved in group communication or a KDC for each node, we divide the group of nodes in regions with each having its own KDC. Each user can communicate securely with members of its own region as well as with those belonging to other regions. We use a hierarchical approach to represent the partial distributed structure of the distributed system using key graphs. The outcome is a secure group communication providing authenticity, confidentiality and integrity of messages delivered between groups. Each secure group is represented in form of a triplet  $(U, N, R)$  where  $U$  is the set of users,  $N$  is the set of keys held by the users and  $R$  represents a relation on  $U$  and  $R$ . This approach has been developed keeping in view the scalability issues of the distributed systems where number of group members may increase or decrease with time. For the same reason we use rekeying strategies to redistribute the keys every time a user joins or leaves the group. This approach is irrespective of frequency of joins and leaves.

**Keywords** – Group communication, multicast, partially distributed systems, rekeying, security

---

### I. Introduction

In many distributed applications like time server, *group communication* plays an important role in multicasting the information from one authentic user to many target users. Consider one such case in which the communication network of a business firm is divided into a distributed network with one branch in New Delhi, second in Calcutta and third at Hyderabad. There is a project on which employees from various branches are working together on. If we use group communication approach then the contribution of one team member can be made available to other colleagues remotely without incurring much cost. Now suppose that a new employee or anyone who wants to join the project but on the condition that he shall not have access to previous transactions of the already existing members. We may have the protocol as follows. Whenever a person joins the group the access mechanism (we shall use keys) is dynamically changed so that the new member can't access the previous message exchanged between the other members. Similarly whenever a member leaves the group the access mechanism to the data is again dynamically changed so the leaving member can't access any future transactions which take place between other members of the group in the near future.

Now talking about the partially distributed systems we shall first describe distributed systems formally. Distributed systems are an abstraction of multiple network systems connected together locally or located geographically far apart but behave as if a single system exist. There is abstraction of resources viz. the user is unaware of the location, name, number of copies of that resource. Now with respect to our protocol we discuss about the partially distributed systems. In this case various nodes (collection of systems) are segregated as *regions* as described in the figure (Fig. 1) below. Here the communication inside the region follows a usual protocol in which the sender needs to know about the *public key* of the receiver who decrypts the message using his *private key*. However cross-region communication involves both public and private-key cryptosystems and *symmetric key* as well.

In the initial setup all the users of all the nodes have to authenticate themselves with a trusted server, the KDC here. In this process each of the users is allotted an ID and a public-private key pair which is used for further communications. This is what is referred to as *user-key*. For the group communication the KDC also sends to each user/ member a group key to be shared by all members of the group. This gives an extension to the above example where suppose that different departments of different branches contribute to the project and only members of that project and now of that department can access their respective contributions. For example say that developer group of the project would like only developers to see the project design reports while coding team may want only coders to have access to the java modules. This seems to give a communication tradeoff in linear proportion with number of users 'n'. However the one-to-one initial setup cost is incurred only once i.e. at the beginning. Only changing the keys for each departure or entry comprises the real communication cost.

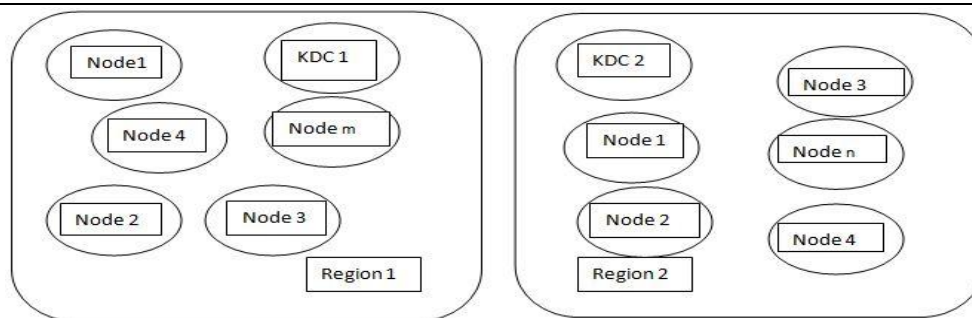


Fig. 1 A simple partially distributed architecture

Here, the KDC is responsible for rekeying after every departure and entry happens anywhere in that region. As a result a new *region-key*, *node-key*, *individual key* and an *ID* is given to the new member (refer [1]) while the already existing members are broadcasted a message about the new *region-key* encrypted using the previous group key along-with new node key to existing members of that node.

## II. APPROACH

The approach that this protocol uses has been discussed in the paper published by Wong et. al. [2]. The notion here slightly differs in that instead of subgroups we consider nodes and regions to be analogous to groups and thus we have a notion of node-key and region-key. To represent a secure group/region we use a triplet (U, N, R), where U represents number of users, N denotes the key set and R represents a relation between U and N which in mathematical terms is given as  $R \subset U \times N$ . This representation can be made clear from the following illustration. Let us consider the former example of a company project. Let there be an ongoing project in company in which all the teams including the analysis team, design team and implementation team are currently working on. Thus they form the three nodes in our case. The trusted server distributes them the key in following way. In the initial setup phase the individual keys are given to each user along-with their node key and the region key. Now suppose that in the design team one of the members is transferred from this project then the procedure is as follows. The node key for the designing team has to be recomputed by the KDC/ trusted server and multi-casted to the remaining members.

## III. Tree Key Graphs

Key graphs are a diagrammatic representation Fig. 2 of KDC along with the nodes and the users at the lower levels. It is basically a directed acyclic graph i.e. it has no cycles. There are two kinds of nodes in the graph – U-nodes represented by a square and K-nodes representing the keys. Following are some properties about key graphs:-

- 1) There is a bijection between set U and the set of u-nodes in graph.
- 2) There is a bijection between set N and the set of k-nodes in graph.
- 3)  $(u,k) \in R$  iff there exists a directed path from that u-node to k-node.

We use two functions associated with each secure group.

$$\text{USERSET}(k) = \{u \mid (u, k) \in R\}$$

$$\text{KEYSET}(u) = \{k \mid (u, k) \in R\}$$

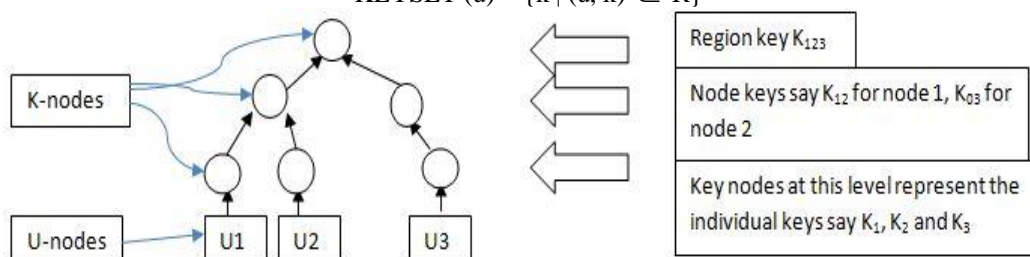


Fig.2 Various types of nodes in Tree key graph

The tree key graphs are a special case of this key graphs in which there is only a single root. Every such tree is has following two parameters:-

- 1) Height of the tree - It refers to the longest path in terms of edges in a key graph.
- 2) Degree of the tree – It refers to maximum number of edges incident on a node.

It is quite obvious from the figure itself that each path ends with a user node and hence each user has number of keys equal to the height of the tree at most.

3.1 Rekeying strategy using group-oriented protocol

Now we discuss about the rekeying concept which is the core point of this paper. Before going to the rekeying strategy let's talk about the role of rekeying and some useful notations. After the initial setup each of the existing users in a region has his set of keys. Now suppose a new user wants to join a node, he sends a request message to the KDC requesting for the new region and node key (refer [1]). To achieve this, the KDC sends rekey messages having both the newly generated region key/node key to the respective users. Following notation has been used to represent this context:-

$$A \rightarrow B (Z)$$

Where

1. If B is a single user then representation means sending the message from Z from A to B.
2. If B is a set of users in a node then it stands for a multicast /unicast message from A to B.

3.2 Rekeying during joining in a tree key representation

After the KDC receives the joining request from the user 'r' in the node 'n' (say) the new keys have to be distributed to the joining user as well as the existing users so the new user doesn't have access to previous transactions. To do so the KDC shall create a new 'k' node,  $k_u$  for the new user and finds an existing 'k' node representing a node (to which 'r' belongs) in the region and chooses it to be its parent node which Wong. et. al[2]. refers to as the *joining point*. After this the rekeying messages are constructed, encrypted and sent to the respective nodes. As represented in the Fig.3 below, the parent node of the new user's  $k_u$  node will get a new *node key* and also the *region key* will change.

As shown in the Fig. 3, user  $U_4$  joins the group by sending joining request to the KDC server it is assigned an individual key  $K_4$ . The *joining node* here is  $K_{03}$ . We see that the region key changes from  $K_{1-3}$  to  $K_{1-4}$  and the *node key* at  $K_{03}$  changes to  $K_{34}$ . We also see that users  $U_1, U_2$  need to have only the new region key  $K_{1-4}$  because there is no change in its *node key* while users  $U_3, U_4$  needs new region key  $K_{1-4}$  as well as new node key  $K_{34}$ .

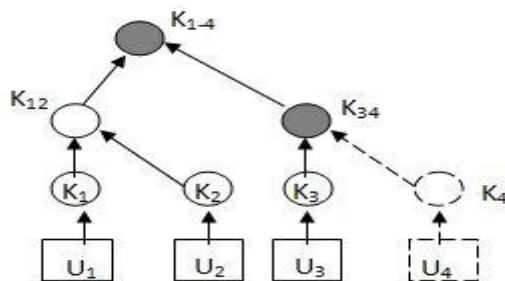


Fig.3 Joining of a user in tree key graph

To distribute the new keys to the respective users the server constructs the rekey messages which is a collection of encrypted keys which is decrypted by the user using appropriate keys. We use group oriented protocol to achieve this rekeying. We now discuss on this approach in the event of the user joining the group. One of the key points while discussing distributed systems is the communication traffic. Thus we choose to use such a protocol which tries to minimize the total number of intra- region messages. In group-oriented protocol the KDC frames a single rekeying message whose length is  $O(\log_d(n))$  for a region with 'n' nodes and key tree degree 'd'. Thus there is an obvious advantage in terms of network traffic as only one multicast message is needed to be sent to all the existing users in the region and a single unicast message to the joining user. In our example also there would be one multicast message and one unicast message depicted as following:-

1.  $KDC \rightarrow \{U_1, U_2, U_3\}: (K_{1-4})_{K_{1-3}}, (K_{34})_{K_3}$
2.  $KDC \rightarrow \{U_4\}: (K_{1-4}, K_{34})_{K_4}$ , here the subscripts in the message refers to encryption key

Though we have reduced the number of messages but the encryption cost remains to  $2(h-1)$ , where 'h' is height of the tree.

3.3 Rekeying during leaving in a tree key graph representation

The user node  $K_u$  in Fig. 4 sends the request for leaving request to the KDC which finds the parent of the  $K_u$  and removes the corresponding K node and U node for the user 'u'. The parent of  $K_u$  is referred to as the *leaving point*. Now to prevent the leaving user from accessing the future transactions the rekeying is again performed and new keys are assigned. There is a new node key for the members of the node of which user 'u' was the part and also the region key is generated and distributed. In our example the rekeying follows group oriented protocol so following messages are transmitted to respective users.

1.  $M_1 = (K_{1-3})_{K_{12}}, (K_{1-3})_{K_3}$
2.  $M_2 = (K_{03})_{K_3}$

3.  $KDC \rightarrow \{U_1, U_2, U_3\}: M_1, M_2$

We see that this message is 'd' times bigger than join message and the encryption cost is  $d(h-1)$ .

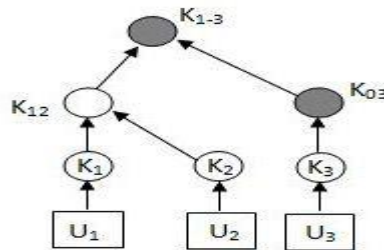


Fig.4 Leaving by a user in tree key graph

#### IV. Inter And Intra Region Communication

In this section we discuss how inter-region and intra-region communication takes place. To begin with we first discuss basic concepts of cryptography. First is symmetric cryptography which is the cryptography protocol that we use in our model of partial distributed systems for inter-region communication while for intra-region communication we use the well-known concept of public-key cryptography. In symmetric cryptography there is a shared secret key known only to the communicating parties and no one else in between. While in public-key cryptography (E.g. RSA) each communicating party has a pair of keys, the public key which is known to all and the private key which is kept secret. Till now we have discussed only the intra-node scenario of rekeying which helps established a security amongst the users inside a particular region that new users can't have access to past transactions of the existing users and leaving users can't have access to future transactions. Now we discuss how the communication takes place after the things are established.

The intra-node communication is not much of an importance so we discuss it in brief. When the users belonging to the same region want to communicate with each other they first contact the KDC to know the node key and the public key of the other user with whom it wants to communicate with. Then it encrypts the message using the two keys which is decrypted by the other user using its prior knowledge of its node key and its secret key. Thus the communication remains secure. The inter-node communication is a more likely and interesting scenarios which involves parties or the users belonging to different regions. The phenomenon of symmetric key distribution is depicted below in the Fig. 5.

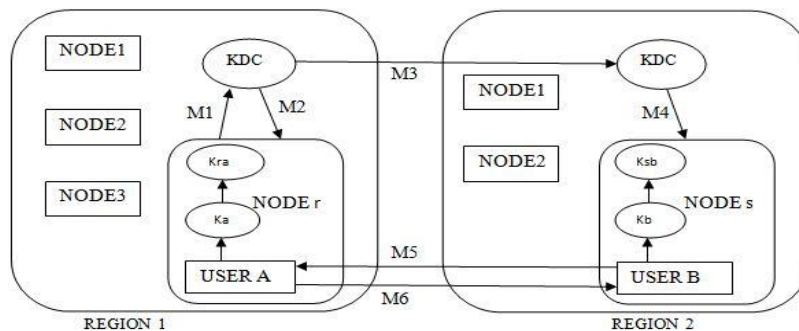


Fig.5 Key exchange in partially distributed systems for group communication

- $M1 = (Req_a, ID_a, ID_b)$       **Req<sub>a</sub>**: request code by user A , **ID<sub>a</sub>**, **ID<sub>b</sub>**: Identifiers of user A and user B
- $M2 = E((Req_a, ID_a, K_{ab}), (K_{ra}, K_a))$       **K<sub>ra</sub>**: node key of user A , **K<sub>a</sub>**: individual key of user A , **K<sub>ab</sub>**: shared key
- $M3 = E((K_{ab}, K_{ra}, ID_a, ID_b), K_2)$       **K<sub>2</sub>**: secret key of KDC at region
- $M4 = E((K_{ab}, K_{ra}, ID_a), (K_{sb}, K_b))$       **K<sub>sb</sub>**: node key of user B , **K<sub>b</sub>**: individual key of user B
- $M5 = E((N_{rand}, K_{sb}), K_{ab})$       **N<sub>rand</sub>**: random number generated by user B , **K<sub>sb</sub>**: node key of user A
- $M6 = E(N_t, K_{ab})$       **N<sub>t</sub>**=**f(N<sub>rand</sub>)**: f is a previously decided function

Now we discuss the phenomena happening in the figure above. The distribution of the secret depends on at what locations the user A and user B exists. If they lie in the same region then we can use a fairly simple centralized approach but when both the users lie in different regions the phenomena is as above in the figure. It is assumed that the KDC is a trusted party that distributes the keys to all the users and holds a table containing information about the individual/ secret keys of the users in that region as well as the node keys and the secret keys of all other KDCs too. The process of key exchange happens as below :-

1. User A sends the request to KDC of its own region consisting of code request, its identifier and the identifier of the user (B) it wants to communicate with in the other region.

2. The KDC of A's region then acknowledges this message by encrypting the secret shared key (generated by the KDC) for secure communication between A and B with the node key and individual key of the user A.
3. The  $KDC_A$  then sends a message containing the shared key  $K_{AB}$ , node key of A and identifiers of A and B encrypted with the secret of  $KDC_B$  which is also a trusted party.
4.  $KDC_B$  then extracts the information from the message sent by  $KDC_A$  and sends the shared key and A's identifier to user B encrypted with its node key and individual key.
5. To authenticate/ validate the genuineness of user a user B sends a dummy message containing a random number and its node key encrypted with the shared key  $K_{AB}$ . If A is the genuine person who wants to communicate with B then he is also the owner of this shared key at the first place. A then finds a functional value ( $f(N_{rand})$ ) of the random number and sends back a message to B encrypted by  $K_{AB}$ . Now the user B is sure of the authenticity of user A and both have the shared keys too. Thus A and b can communicate securely using this shared key.

## V. Conclusion

This paper is an extension to work done on hierarchical group communication. We have basically provided the application of hierarchical group communication with respect to the partially distributed systems which divide the users in terms of regions and nodes. The aim is to provide secure communication to the users in the system so that each time a user joins or leaves the system he doesn't have access the previous transactions or the future transactions. We used the concept of trees to achieve this hierarchical representation of users in the system. In this paper we have kept in mind the limitations of the distributed systems which may under perform during high network traffic so we used the group-oriented protocol for rekeying. Inter-region secure communication was achieved by using the concepts of public-key cryptography. This paper can serve as a platform for developing partially distributed system for practical scenarios as discussed in the introduction of this paper. Systems which involve group communication and need its communication to be secured on each leave and join in the system can use this paper as its baseline.

## References

### Books:

- [1] Pradeep K. Sinha, "Distributed operating systems concepts and design", PHI learning private limited, 2007

### Journal Papers:

- [2] Chung Kei Wong, *Member, IEEE*, Mohamed Gouda, and Simon S. Lam, *Fellow, IEEE*, "Secure Group Communications Using Key Graphs" *IEEE/ACM TRANSACTIONS ON NETWORKING*, VOL. 8, NO. 1, FEBRUARY 2000



## Comparative and Behavioral Study on VANET Routing Protocols

S.Sujatha<sup>1</sup>, P.Soundeswari<sup>2</sup>

<sup>1</sup>(Associate Professor, Computer Science, Dr.G.R.D College of Science, India)

<sup>2</sup>(Scholar, Computer Science, Dr.G.R.D College of Science, India)

---

**Abstract :** Vehicular ad hoc networks (VANETs) without any support from fixed infrastructure offer a large number of applications. A Complete understanding of the communications Channel between vehicles is necessary for realistic modeling of VANETs and the development of related technologies and applications. This paper gives a brief summary of different routing algorithms in VANET and also discusses the major classifications. The comparative study is mainly done on the essential characteristics and behavior of the protocols and based on the comparison results, the position based protocol is considered as the best for handling issues in packet delay, traffic congestion and better throughput in Vehicular ad hoc networks.

**Keywords:** VANET, CAR, CGSR, DSR, DV-CAST

---

### I. Introduction

VANET stands for vehicular ad hoc network provide better communication between moving vehicles and fixed equipments, VANET is a new standard that integrates Wi-Fi, Bluetooth and other mobile connectivity protocols. The essential requirement of VANET is that it should be able to communicate in any environment irrespective of traffic densities and vehicle locations. Vehicular communications are made in fluctuating environment and should work both in urban and rural areas.

The primary factors to be considered in designing a VANET is security, location and the maintenance of the services. As an open network VANET is always targeted to malicious communication, the safety and security factors of the messages should not harm the traffic and driver. MANET routing protocol is not suitable for VANET because MANET routing protocol has difficulties from finding stable routing paths in VANET environments. Six types of routing protocols in VANETs are topology based, position based, geocast based, cluster based, broadcast based and infrastructure based.

The rest of the paper is organized as follows. Section 2 deals topology based protocols, section 3 on position based protocols, section 4 on broadcast protocols, section 5 on performance study and comparison results on VANET Protocols, section 6 concludes the paper.

### II. Topology Based Protocols

These protocols discover the route and maintain it in a table before the sender starts transmitting data. They are further divided into reactive, proactive and hybrid protocols.

#### 2.1 Proactive protocols

The proactive protocol is also known as table driven routing protocol. These protocols work by periodically exchanging the knowledge of topology among all the nodes of the work. The proactive protocols do not have initial route discovery delay but consumes lot of bandwidth for periodic updates of topology. There are several routing protocols that fall under this category

##### 2.1.1 Source-Tree Adaptive Routing (STAR)

Source-Tree Adaptive Routing (STAR) is another link State protocol. It reduces overhead on the network by eliminating periodic updates. This protocol can be suitable for large scale networks but it needs large memory and processing because it has to maintain large trees for whole network. The Each node maintains a source tree. Each node builds a partial topology graph using aggregates of neighbor information learnt using an underlying neighbor discovery protocol and source trees reported by the neighbors [1].

##### 2.1.2 Clusterhead Gateway Switch Routing (CGSR)

CGSR is a clustered multihop mobile wireless network with several heuristic routing schemes. Advantages of CGSR Better bandwidth utilization reduce the size of distance vector table because the routing is performed only over cluster head. Disadvantages of CGSR More time is spend in selection of cluster heads and gateways if the mobile node uses CDMA/TDMA then it can take some time to get permission to send packets Changes in the cluster-head, may result in multiple path breaks [2].

##### 2.1.3 Destination Sequenced Distance Vector Routing (DSDV)

Each DSDV node maintains two routing tables: one for forwarding packets and one for advertising incremental routing packets. Advantages of DSDV It is quite suitable for creating ad-hoc networks with small number of nodes Solve the Routing Loop problem Count to infinity problem is reduced DSDV maintains only the best path instead of maintaining multiple paths to every destination . Disadvantages of DSDV, DSDV requires a regular update of its routing tables, which uses up battery power and a small amount of bandwidth even when the network is idle whenever the topology of the network changes, a new sequence number is necessary DSDV is not suitable for highly dynamic networks [3].

## 2.2 Reactive protocols

Reactive routing opens a route only when it is necessary for a node to communicate with another node. It maintains only the routes that are currently in use, thereby reducing the burden on the network.

### 2.2.1 Dynamic source routing (DSR)

This protocol consists of two operations “*Route Discovery*” and “*Route Maintenance*” that makes it self-configuring and self-organizing. Another important property of DSR routing protocol is network type flexibility. Disadvantages of DSR The route maintenance mechanism does not locally repair a broken link The connection setup delay is higher than in table-driven protocols This routing overhead is directly proportional to the path length [3].

### 2.2.2 Dynamic MANET On demand (DYMO)

DYMO is another reactive routing protocol that works in multi hop wireless networks. DYMO has a simple design and is easy to implement. The basic operations of DYMO protocol are route discovery and route Maintenance was studied extensively along with comparison of two on demand routing protocols [4].

## III. Position Based Routing Protocols

Position based routing consists of class of routing algorithm. Position based routing is beneficial since no global route from source node to destination node need to be created and maintained. Position Based routing is broadly divided in two types: Position based greedy V2V protocols, Delay Tolerant Protocols.

### 3.1 Greedy forwarding protocol

#### 3.1.1 Geographic Source Routing (GSR)

GSR use “*switch back to greedy*” method for local recovery. After a packet reach to its local maximum, it switch back to greedy forwarding. The sender node reaches the destination by using the road topology map and the above information. In other words in GSR the source node finds the shortest path to destination on the graph using simple graph algorithms and marks the packet with destination’s location. In this the packet travels through junctions to reach the destination. In this algorithm, each node maintains a Neighbor list, a Topology table, a Next Hop table and a Distance table [1].

#### 3.1.2 Anchor-Based Street and Traffic Aware Routing (A-STAR)

A-STAR is traffic aware: the traffic on the road determines whether the anchor points of the road will be considered in the shortest path. A-STAR routes based on two kinds of overlaid maps: a statically rated map and a dynamically rated map. A statistically rated map is a graph that displays bus routes that typically imply stable amount of traffic. The development of A-STAR was inconsideration with city environment. A-STAR also use traffic information and street awareness in path finding [5].

#### 3.1.3 Greedy Perimeter coordinator Routing (GPCR)

GPCR (Greedy Perimeter coordinator Routing) modified to adapt to city scenario. Here, a restrictive greedy algorithm is simply followed when nodes are in street and an actual routing decision is taken when at the junction of streets. Here the packet is forwarded to a node in the junction rather sending it across the junction. GPCR traverses the junctions by a restricted greedy forwarding procedure [6].

#### 3.1.4 Vehicle-Assisted Data Delivery Routing Protocol (VADD)

VADD protocol adopted the idea of carry-and-forward for data delivery from a moving vehicle to a static destination. Use predictable traffic pattern and vehicle mobility to assist efficient data delivery. A vehicle makes a decision at a junction and selects the next forwarding path with the smallest packet delivery delay. A path is simply a branched road from an intersection. It is suitable for multi-hop data delivery [5].

#### 3.1.5 Connectivity Aware Routing Protocols (CAR)

CAR is designed specifically for inter-vehicle communication in a city and/or highway environment. CAR integrates locating destinations with finding connected paths between source and destination. CAR ensures to find the shortest connected path because CAR has higher packet delivery ratio than GPSR and GPSR+AGF. It cannot adjust with different sub-path when traffic environment changes [7].

#### 3.1.6 Diagonal-Intersection-Based Routing Protocol (DIR)

DIR protocol constructs a series of diagonal intersections between the source and destination vehicle. DIR vehicle is auto adjustable, Auto adjustability means that one sub path with low data packet delay between two neighboring diagonal intersections, which is dynamically selected to forward data packets. For given a pair of neighboring diagonal intersections, two or more disjoint sub-paths exist between them. To reduce the data packet delay, the route is automatically re-routed by the selected sub-path with lowest delay [8].

### 3.1.7 Border-node based most forward within radius routing protocol (B-MFR)

Border-node based Most Forward within Radius routing (B-MFR) which uses the concept of border-node within the sender's communication range to minimize the number of hops between source and destination. The B-MFR utilizes the border-node to avoid using interior nodes within the transmission range for further transmitting the packet. Next-hop forwarding method like greedy forwarding scheme for linear network does not support well in highly mobile ad hoc network such as vehicular ad hoc network [9].

### 3.1.8 GVGrid

GVGrid is designed not for sparse regions with high-speed vehicles such as highways, but for dense regions with low-speed vehicles such as cities. It also reconstructs the route when it is broken by the movement of vehicles. GVGrid divides the geographical area into uniform-size squares called grids. Then the intermediate grids between source and destination are recorded in the routing table. An appropriate vehicle which has the fewest number of disconnections in each grid is chosen to forward packets to next grid. GVGrid consists of two processes, a route discovery process and a route maintenance process [10].

### 3.1.9 Contention Based Forwarding (CBF)

The contention-based forwarding (CBF) algorithm is a greedy position-based forwarding algorithm that does not require the proactive transmission of beacon messages. In CBF, the next hop is selected through a distributed contention process based on the actual positions of all of the current neighbors. In this contention process, CBF makes use of biased timers. To avoid packet duplication, the first node that is selected suppresses the selection of further nodes using an area-based suppression algorithm [11].

### 3.1.10 Directional Greedy Routing Protocol (DGRP)

DGRP is a position based greedy routing protocol, which uses the location, speed and direction of motion of their neighbors to select the most appropriate next forwarding node. . It predicts the position of nodes within the beacon interval whenever it needs to forward a data packet. This prediction can be done using previous known position, speed, and direction of motion of node [12].

#### 3.1.11 Predictive Directional Greedy Routing Protocol (PDGRP)

Jiayu Gong proposed PDGRP, in which the weighted score is calculated from two strategies namely, position first forwarding and direction first forwarding. Here next hop selection is done on prediction and it is not reliable at all situations. It doesn't guarantee the delivery of packet to the node present in the edge of the Transmission range of forwarding node, which is considered as most suitable next hop, due to high dynamics of vehicles. This will lead to low packet delivery ratio, high end to end delay and increased routing overhead [13].

### 3.1.12 Hierarchical Clustering Based Greedy Routing (HCBGR)

HCBGR is a unicast position based greedy routing algorithm designed for sending messages from any node to any other node in VANETs. The HCBGR Algorithm has six basic functional units. The first is Neighbor Node Identification(NNI), the second is Distance Calculation(DC), the third is Direction of Motion Identification(DMI), the fourth is Reckoning Link Stability(RLS), the fifth is Weighted score calculation(WS) and the sixth is Potential Node Selection(PNS) [13].

#### 3.1.13 Reliable Directional Greedy Routing (RDGR)

Reliable Directional Greedy Routing (RDGR) is a reliable position-based greedy routing approach which uses the position, speed, direction of motion, and link stability of neighbors to select the most appropriate next forwarding node. The packet sender or forwarder node, selects neighbor nodes which have forward progress towards destination node using velocity vector, and checks link stability of those nodes [14].

## **IV. Broadcasting Based Routing Protocols**

Broadcast routing is frequently used in VANET for sharing, traffic, weather and emergency, road Conditions among vehicles and delivering advertisements and announcements. Broadcasting is used when message needs to be disseminated to the vehicle beyond the transmission range i.e. multi hops are used.

4.1 Distributed vehicular broadcast protocol (DV-CAST)

DV-CAST for a multi-hop broadcast routing protocol in VANETs and indicates three traffic scenarios for a vehicular broadcasting; dense traffic scenario, sparse traffic scenario, and regular traffic scenario. This protocol causes high control overhead and delay in end to end data transfer [10].

4.2 Preferred group broadcast (PGB)

PGB is not a reliable broadcasting protocol but it is a solution to prevent broadcast storm problem from route request broadcasting. Each node in PGB will sense the level of signal strength from neighbor broadcasting. The signal strength is used for waiting timeout calculation. Nodes in the edge of circulated broadcast will set shorter waiting timeout. Only node with shortest timeout will rebroadcast the message. But there exists a problem on low density area [15].

5. Performance Study on VANET Protocols

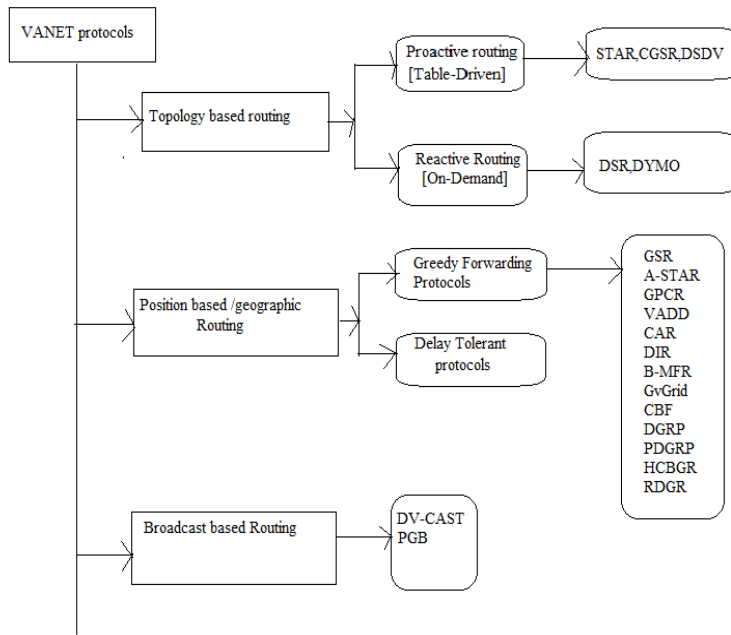


Fig. 1 VANET protocols

TABLE 1 Comparison Results of routing protocols in VANETs

Protocol	Scenario	Routing maintenance	Routing type	Strategy	Simulator
STAR	Urban	Proactive	Unicast	Greedy forwarding	NS2
CGSR	Urban	Proactive	Unicast	Multihop	NS2
DSDV	Urban	Proactive	Unicast	Multihop	NS3
DSR	Urban	Reactive	Unicast	Multihop	NS2
DYMO	Urban	Reactive	Unicast	Multihop	QUALNET
GSR	Urban	Reactive	Unicast	Greedy forwarding	NS2
A-STAR	Urban	Reactive	Unicast	Greedy forwarding	NS2
GPCR	Urban	Reactive	Unicast	Greedy forwarding	NS2
VADD	Urban	Reactive	Unicast	multihop	NS2
CAR	Urban	Reactive	Unicast	Greedy forwarding	NS2
DIR	Urban	Reactive	Unicast	Greedy forwarding	NS2
B-MFR	Urban	Reactive	Unicast	Greedy forwarding	NS2
GVGrid	Urban	Reactive	Unicast	Greedy forwarding	NS2
CBF	Urban	Reactive	Unicast	Greedy forwarding	NS2
DGRP	Urban	Reactive	Unicast	Greedy forwarding	NS2
PDGRP	Urban	Reactive	Unicast	Greedy forwarding	NS2
HCBGR	Urban	Reactive	Unicast	Greedy forwarding	NS2
RDGR	Urban	Reactive	Unicast	Greedy forwarding	NS2
DV-CAST	Highway	Proactive	Broadcast	Multihop	NS2
PGB	Urban	Reactive	Unicast	Multihop	NS2

This table gives a comparison of these protocols and discusses forward strategy, routing type, protocol scenario, routing maintenance and network simulator of the different VANET routing protocols. Among this scenario the position based routing protocol works best in open space scenario with evenly distributed nodes. The absence of fewer obstacles in highway scenario is attributed to its good performance. From forward strategies position based routing technique employs the awareness of vehicle about the position of other vehicle to develop routing strategy.

## V. Conclusion

In this paper we have discussed several VANET protocols. Position of the vehicle is one of the most important data for vehicles. Position based routing protocols need the information about the physical location of the participating vehicles to be made available. After analyzing the survey of protocols, it is found that the position based routing has better performance because there is no creation and maintenance of global route from source node to destination node. In the position based routing protocol, all the packets are received with small average delay, better throughput, and effective utilization and also helps to prevent the accidents on the road effectively. In future these protocols can also be used for further research in VANET.

## References

- [1] Bilal Mustafa, Umar Waqas Raja, "Issues of Routing in VANET", Master Thesis, Computer Science, C m p t S i , and Thesis no: MCS-2010-20 Jun 2010.
- [2] Natarajan Meghanathan<sup>1</sup> "Survey and taxonomy of unicast routing protocols for mobile ad hoc networks", The International Journal on Applications of Graph Theory in Wireless Ad hoc Networks and Sensor Networks (GRAPH-HOC), Vol.1, No.1, December 2009.
- [3] Pankaj barodia, "routing protocols in wireless network", Deenbandhu Chhotu Ram University of Science and Technology Murthal, Sonipat (Haryana), June 2011.
- [4] Manish Sharma<sup>1</sup> & Gurpadam Singh<sup>2</sup> "Evaluation of Proactive, Reactive and Hybrid Ad hoc Routing Protocol for various Battery models in VANET using Qualnet"
- [5] Uma Nagaraj, <sup>2</sup>Dr. M. U. Kharat, <sup>3</sup>Poonam Dhamal, "Study of Various Routing Protocols in VANET", IJCST Vol. 4, ISSue 4, oCT.-DeC. 2011.
- [6] Amab Kumar Banik, "Routing Protocol with prediction based mobility model in vehicular ad hoc network (VANET)", April 2010.
- [7] Yuh-Shyan Chen, "Routing in Vehicular Ad Hoc Networks (VANETs)", NTPU, Department of Computer Science and Information Engineering.
- [8] Rakesh Kumar, <sup>2</sup>Mayank Dave, " A Comparative Study of Various Routing Protocols in VANET", IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 1, July 2011.
- [9] Monika, Sanjay Batish and Amardeep Singh, " Border-node based Movement Aware Routing Protocol", International Journal of Computer Science and Informatics ISSN (PRINT): 2231 -5292, Vol-1, Iss-4, 2012.
- [10] YUN-WEI LIN<sup>1</sup>, YUH-SHYAN CHEN<sup>2</sup>, AND SING-LING LEE,<sup>3</sup> " Routing Protocols in Vehicular Ad Hoc Networks: Survey and Future Perspectives".
- [11] Si-Ho Cha<sup>1</sup>, Min-Woo Ryu<sup>2</sup> and Kuk-Hyun Cho<sup>2</sup> "A Survey of Greedy Routing Protocols for Vehicular Ad Hoc Networks", Smart Computing Review, Vol. 2, no. 2, April 2012|25.
- [12] K. Jayasudha, " Hierarchical Clustering Based Greedy Routing in Vehicular Ad Hoc Networks", European Journal of Scientific Research ISSN 1450-216X Vol.67 No.4 (2012), pp. 580-594 © Euro Journals Publishing, Inc. 2012
- [13] K. Lakshmi<sup>1</sup>, K.Thilagam<sup>2</sup>, K. Rama<sup>3</sup>, A.Jeevarathinam<sup>4</sup>, S. Manju Priya<sup>5</sup>, " Comparison of Three Greedy Routing Algorithms for Efficient Packet Forwarding in VANET", IJCTA|JAN-FEB 2012.
- [14] K.Prasanth<sup>1</sup> Dr. K.Duraiswamy<sup>2</sup> K.Jayasudha<sup>3</sup> and Dr.C.Chandrasekar<sup>4</sup> "I m p r o v e d P a c k e t F o r w a r d i n g A p p r o a c h i n V A N E T u s i n g R D G R a l g o r i t h m " , International Journal of Next Generation Network (IJNGN), Vol.2, No.1, March 2010.
- [15] Uma Nagaraj, Poonam Dhamal, "Broadcasting Routing Protocols in VANET", ISSN 2224-610X (Paper) ISSN 2225-0603 (Online) Vol 1, No.2, 2011.