Security Enhanced Trust based G-LeaSel for Grid Environment

S.Mary Vennila
Associate Professor
Department of Computer Science
Presidency College, India

ABSTRACT
Secure and trusted group communication is an active area of research. The growing importance of group communication based applications fuelled its popularity. The central research challenge is secure and efficient group key management.. The main issue in secure group communication is group dynamics and key management. A scalable secure group communication model ensures that whenever there is a membership change, the leader of the group generates and distributes a new group key to the group members with minimal computation and communication cost. G-LeaSel model adopts a random methodology for the selection of leader and does not analyze the selection of a trust worthy leader to entrust the critical task of key management. This paper explores the benefits of selecting trust based leader selection to perform the key management. The proposed mechanism proves to be more secure than leader selection methodology adopted in G-LeaSel. Also the proposed solution exhibits self-stabilization for hack attempts and improves the throughput of the network.

Keywords
Group dynamics, G-LeaSel, Trust, Multicast, Grid environment.

1. INTRODUCTION
The recent growth of applications is driving the need for secure multicast communication. The secure multicast communication ensures authenticity and preserves confidentiality of the data between registered senders and receivers [6]. Grids, formed from interconnection of clusters are capable of providing enormous computing power. Another vital reason for the prominence of Grid technology is its ability to harvest unused computing power in a distributed environment [10]. Secure Group communication is the mechanism by which nodes (computing nodes or resource nodes) can interact with each other securely. This mechanism manages the nodes or members and membership events in the cluster or group securely. Secure Group Communication in the grid is thus an important concern having wide scope for addressing the needs of a number of applications like real time audio/video streaming and computational steering. This Secure Group Communication in Grid though having a big scope of improvement only recently caught the attention of the researchers.

Due to the lack of network-level access control in grids, enforcing message confidentiality for group communication requires encryption. This requires a group key management solution to distribute and maintain cryptographic keys with registered group members [9]. Similarly, cryptographic authentication schemes are necessary to ensure that registered receivers can verify that received packets come from registered senders [7] [8].

G-LeaSel [2] is a proven secure multicast group communication model for the Grid environment. When users move in and out of a multicast session, in order to preserve confidentiality, it becomes necessary to rekey each time a user enters or leaves the multicast session or group. The G-LeaSel model creates multiple groups and authorizes selected leaders to perform key management activities within their concerned group. However, G-LeaSel does not consider selecting leaders based on trust, which is an important step in achieving security to G-LeaSel model. In this context, without human judgement, the challenge for controllers is to distinguish the peers' identities and behaviours autonomously.

Our earlier research [2] uses a random leader selection approach in which a Deputy Service Provider (DSP) can independently handle the issue of selecting the leader. Due to the dynamism of the environment and mobility of nodes, an efficient method of computing trust is required for the G-LeaSel model. This trust management scheme co-operatively collects the trust values from all the nodes directly or indirectly. It provides a mechanism of allowing neighbours to judge the trustworthiness of the node of interest to calculate its trustworthiness. The proposed model is also analysed for self-stabilization during occurrence of faults. The simulation results of the proposed Security Enhanced Trust based G-LeaSel shows results, which outperform the existing G-LeaSel, by securing the leader from hackers' without compromising the throughput of the network.

2. THE G-LEASEL OVERVIEW
The G-LeaSel [2] takes a service-oriented approach to the problem. G-LeaSel is a highly secure, dynamic, distributed sub group model, which caters to the needs of the group communication in the grid. The model aims to address issues like forward confidentiality, backward confidentiality, scalability, fault tolerance and computational efficiency. The group of 'n' nodes is divided into 'm' subgroups, based on the service-classes. The G-LeaSel architecture is as shown in figure 1.

New users can join any sub-group to get the services and also users may leave the sub-group at anytime. One node, designated as the Controller (C) provides the overall multicast security service. 'M' Service providers, one from each sub group is designated namely Deputy Service Providers (DSP). DSPs provide access to all other services under them. They rank the members of the sub-group and select 'p' members of each subgroup as leaders. Then it selects one among the set of 'p' leaders as the leader of the sub-group and alternates the leaders dynamically. The Controller and the DSPs share a common group key. Each subgroup has a common subgroup key and each node has its own private key. The leader among the selected 'p' leaders is responsible for encrypting and decrypting all data within the subgroup. The identity of the leaders known only to the DSP and alternated dynamically for each membership events.

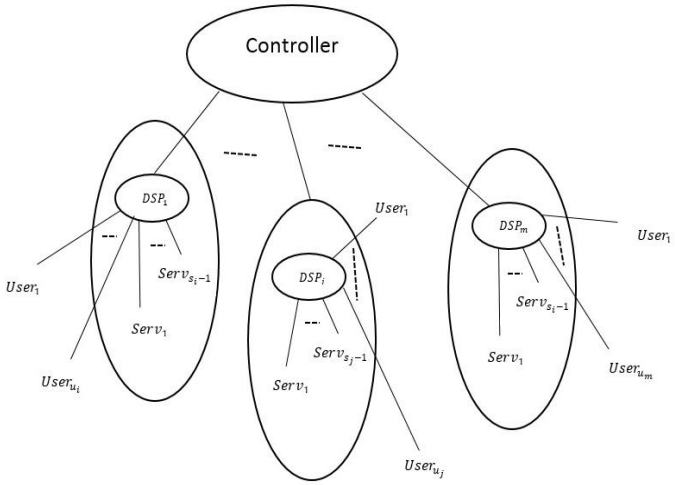

Figure 1 G-LeaSel Architecture

2.1 Group Formation Scenarios

G-LeaSel [2] embarks on a different approach to group formation. The group scenarios are chosen such that they reflect the varied needs of the multicast applications over the grid. Each scenario depicts a group of users requesting a set of services from the DSP. The scenarios for group formation are identified as follows.

Scenario 1: The user-nodes in the sub-group, request for services to the DSP. Here, the services are also available with the DSP and the message transfers are confined to the sub-group.

Scenario 2: If the user-nodes request for services that are not available under the DSP, it in turn acts as a moderator between user and another DSP that actually hosts the requested services. This scenario is typical of the grid environment, where the service is available elsewhere and an intermediate node acts as a broker to get the service. G-LeaSel handles the second scenario, splitting it into two sub-scenarios (2a, 2b).

In scenario 2a, users request services from the DSP and the services are not available with the DSP. The DSP, in turn, act as a broker and gets the required service from some other DSP, which offers the requested services. In the process, DSP becomes a member in the sub-group offering the services and remains as a part of the original sub-group containing the user-nodes.

In scenario 2b, the services requested are not available with the DSP. In cases, where the DSP is busy doing other job and cannot moderate with another DSP to get the service, it can allocate the users directly to the sub-group, which offers the requested services. The user-nodes join the multicast group of new DSP, and avail services as in Scenario 1.

Leader Selection:
In G-LeaSel [2] leaders are selected by DSP among the nodes under the sub-group and are entrusted to perform the task of key management. The revelation of the leader by observing the traffic flow becomes difficult as the leader among 'p'-leaders change for every key management event. In case of a leader compromise, a new leader can be selected immediately from the remaining 'p'-leaders. Thus in order to compromise the sub group, an intruder has to compromise all the 'p'-Leaders and thereby increasing the level of security by a factor of 'p'.

In earlier research works [1][2], during leader selection, a leader among the 'p' leaders is selected randomly. The current leader is asked to stop and the new leader is activated. The proposed trust based G-LeaSel enhances the leader selection methodology by considering the trustworthiness of nodes and hence, selecting the most trusted leader. It also dynamically collects the trust of the nodes which helps the top trusted nodes entering the 'p' leader list, at any point of time, which enhances the security of the system.

3. TRUSTED LEADER SELECTION METHODLOGY

Trust is a relationship established between two entities for specific action [12]. Trust is multi-faceted, even in the same context. Nodes need to develop differentiated trust in different aspects of other nodes' behaviors [16]. In the leader selection methodology, 'p' trustworthy leaders are selected from the group of nodes under the control of DSP and this list of 'p' leaders is updated when their reputation and credentials change due to the dynamism of the environment. An integrated reputation and credential based trust model[13][14][15][16] to provide a more secure and reliable trust management scheme, is used to find the leaders which supports the multicast key management. Using these trust computation, the DSP's for different group scenarios gather the trust parameters and hence the trustworthy of the nodes of the group. This helps to identify the trust based 'p' leaders at any given environment. This also reflects the changes of trustworthiness of different nodes with different environmental situation and keep 'p' trustful leaders at any given situation. The trust calculation methodology is shown in figure 2 and this helps to improve the security of the group communication.

This paper proposes a non-linear reputation computation mechanism [15] [16] in which the increase or decrease in the most recent trust value of the node or provider are non-linear. The variation of trust values happens in small increments if nodes recent past change of trust values are higher and in moderate increments if change of trust value is moderate. A sharp increase in trust value is awarded to nodes, which performs decently if their values are close to the threshold.

However, if a node performs poorly, the trust value is decreases by normal decrements and if the trust value further goes below the threshold, the nodes gets eventually excluded from the leader list maintained by the DSP. The DSP and nodes exchange nodes' credentials or attributes between them. First, the sending node requests the DSP whether the receiving node which is going to provide the service is certified by the DSP

After an affirmative response the transmitting node registers itself with the DSP and the transaction begins. Here the DSP acts as a controller which transmits the request - response between the sender and receiving nodes, so as to involve in the credential exchange.

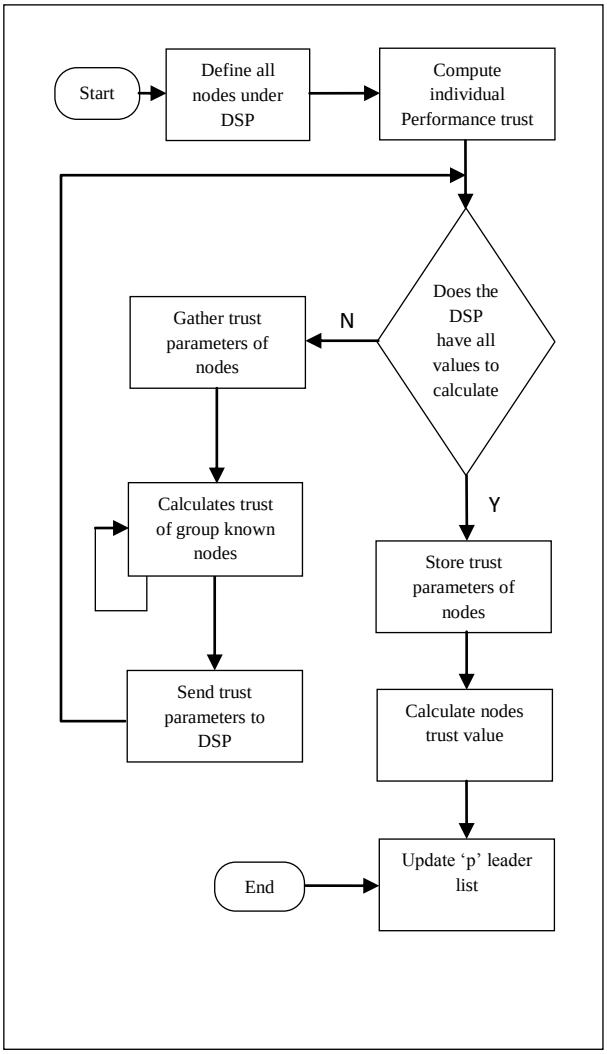

Figure 2 Trust Calculation Methodology

3.1 Parameters used in Trust Calculation

The trust can be determined [4] using the weight factors such as

- Rate trustworthiness.($w1$)
- Age of the rating.($w2$)
- Distance between ratings.($w3$)
- Current score.($w4$)

The parameters for trust computation that are considered and reported by the members/nodes are:

- Number of previous compromises.($s1$)
- Service Availability.($s2$)
- Service Stability.($s3$)
- Computational capability. ($s4$)
- Communication Capacity (s5)

The score of a node A given by ith node is given by,

$$ScoreAi = (s2+s3+s4+s5) / s1$$

By the above formula, it can be inferred that a node with frequent history of compromises will have the lowest score. Depending on the situation, $s1$ can also take absolute values like, 1 if it has no history of compromises and 0 otherwise. This will eliminate that particular node from being considered (since, its score will be 0, if s1 equals 0).The trust rating of a node A given by the ith node is then calculated by the sum as given below

$$\text{Trust } RatingAi = (w1+ w2 + w3 + w4) * \text{Score } Ai$$

The trust rating of the node A is then sum of all the trust ratings calculated.

$$\text{Trust } RatingA = \sum \text{Trust } RatingAi$$

3.2 Leader Selection Algorithm

In this proposed work, each DSP maintains a list of all untrustworthy nodes it had met with in the past and had dismissed as a quack. This mechanism allows a node to store a quack's identifier after it had been snubbed for its poor service. Thus, when the untrustworthy node tries to form relationship with the DSP again it is refused. In addition, the list of untrustworthy node can be exchanged by the nodes when they are idle. This leads to faster identification of malicious nodes in a group. These untrustworthy nodes are also excluded during the leader selection.

The algorithm for selecting a set of 'p' leaders is given below:

Step 1: Define all nodes under DSP for trust computation.

Step 2: Re-examine the quack list maintained by DSP and exclude the untrustworthy nodes from trust computation.

Step 3 : Compute the individual trust of the nodes(Direct) using available performance parameters with DSP.

Step 4: Collect the scores of the node indirectly from the neighboring nodes of the subgroup.

Step 5 : Repeat step 4 until the DSP gathers the group trust parameters of all the nodes of the subgroup.

Step 6: Calculate the trust rating for every node of its subgroup as described in the previous section using direct and indirect trust parameters.

Step 7 : Sorts the trust ratings by ranking the highest rated member as first rank and others subsequently.

Step 8: Select the first 'p' members from the sorted list based on a heuristic threshold value. The first 'p' members form the set of 'p' leaders for the session.

Step 9: Select a member from the 'p' list as a leader for the current membership events.

Steps 10 : Alternate the leaders for different membership events.

Steps 11 : Update the 'p' leader list for frequent intervals by repeating steps 1 through 8.

Whenever DSP initiates membership transaction, it performs the leader selection algorithm and updates the set of 'p' leaders as required. By monitoring the credentials of the members continuously, the DSP updates the set of 'p' leaders as required. The threshold value for the leader selection algorithm is decided based on the application and the level of security requirement.

4. INFERENCES

The G-LeaSel[2] model with the proposed trust based leader selection was analysed on a test bed built using ns-2 for the following parameters – System throughput, Self-stabilization and Average time taken for Hacking and the results obtained

are presented below. Throughput of the model refers to the total amount of data transferred in a given unit of time. It is affected by communication overheads within the system. Average time taken for Hacking is the average time needed by the hacker to disrupt multicast services, by carrying out various kinds of attacks.

4.1 Application Throughput

The G-LeaSel model was simulated using the proposed trust based leader selection and the existing randomized leader selection methodology to check for its performance. Simulations were done considering a group of 500 nodes and the results were obtained. In the simulation, arbitrary hackers were introduced into the model[5] and the throughput of the system was measured. The simulation results thus obtained are compared between G-LeaSel model for trust based leader section methodology and randomized leader selection methodology and plotted for analysis.

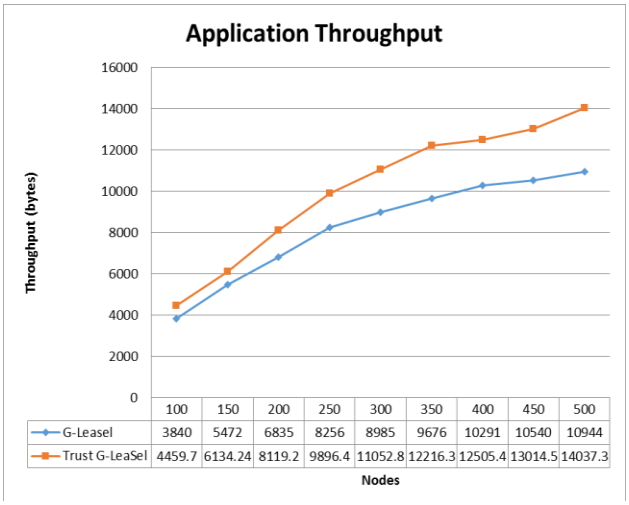

Figure 3: Throughput of G-LeaSel vs. Trust based G-LeaSel.

Figure 3 shows the comparison of the system throughput with nodes and it can be observed that Trust based G-LeaSel performs relatively better than G-LeaSel model in terms of system Throughput. This proves the adaptability of the trust based G-LeaSel model and the improved performance.

4.2 Self-Stabilization

To find the fault tolerant behavior of the different models, the entity that performs the key generation and distribution is simulated to be attacked by malicious attackers and the fault tolerant behavior and the stabilizing effect is analyzed.

A fault is defined to be a failure of the active leader in a sub-group due to physical node related problems (like connectivity, power outages) or being compromised. Self-stabilization is the capability of the system to recover from such faults with graceful degradation of performance. The time taken for such self-stabilization can significantly affect the system performance. In this simulation the packet delivery rate is 50 per second and 10 % of hackers introduced for comparison. The results shown in Figure 4 were obtained by introducing a fixed number of faults into the system and the system's self-stabilizing performance was analysed for increasing number of nodes. Therefore, from figure 4, it can be inferred that Trust based G-LeaSel takes less time for self-stabilization.

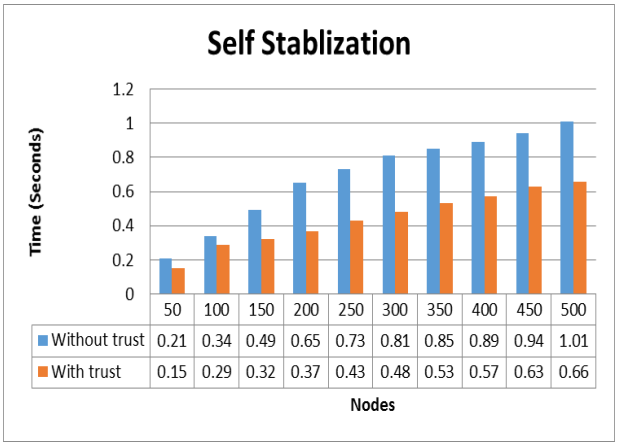

Figure 4: Time to stabilize under fault conditions

4.3 Security

The proposed trust based leader selection methodology on G-LeaSel, was investigated in terms of security. Keeping in mind the adverse influence of hackers on Internet[5], the security level of the G-LeaSel multicast model with the proposed trust based leader selection methodology was analyzed. A multicast session is simulated using the network simulator (ns) with varying number of nodes in the subgroups subject to a maximum of 500. Hacker refers to a type of computer hacker who exploits systems or gain-unauthorized access through skills, tactics and detailed knowledge [2].

10 % members randomly selected from a subgroup of N members/nodes, designated as hackers were introduced into the model and the average time to compromise the leader was found out. The hacker randomly makes attempts to search for the leader by randomly generating packets to other members in the group till it finds the leader. Once the leader is hacked the information with the leader will be under stake.

Figure 5 Security Improvement Graph for trust based G-LeaSel

If a leader is hacked, the key generation and distribution process gets some malicious treatment. The G-Leasel[2], as already pointed out, the DSP changes the leader for every transaction. Figure 5 shows the average time to hack the leader with increasing number of nodes, in G-LeaSel with trust based leader selection and randomized leader selection. It can be very well seen that the effectiveness of trust based

G-LeaSel is sustained even in the presence of hackers. In addition, various security threat issues like Snooping, Denial of Service and Information Disclosure attacks were studied. The successful hack attempt is used as a metric for testing the security of trust based G-LeaSel model.

4.3.1 *Information Disclosure:*

Information Disclosure attack is defined as an attack on confidentiality of the packet carrying the session key during a re-keying operation. When the confidentiality is compromised then the information contained in the re-key packet is revealed enabling an external member to decrypt multicast messages sent only to the subscribers.

The simulation is carried out for a sub-group and then generalized for the whole system. The sub-group is simulated to have 50% partially armed hackers and 50% of fully armed hackers out of the malicious nodes in a sub-group under consideration. The partially and fully armed hackers are made to launch attacks in such a way that the identity of the leader is disclosed. A probabilistic security approach is used to simulate hack attempts. An attempt is said to be successful if the random number generated and the sequence number of the packet are same.

This reasoning is justifiable as a partially or fully armed hacker can compromise and also there are no white or black boxes for security testing. Any vulnerability in a system is disclosed only after a successful hack attempt. Observations are made for number of successful attempts for given different percentages of malicious nodes among the member nodes. The observations are made with and without the trust based leader selection implemented on G-LeaSel.

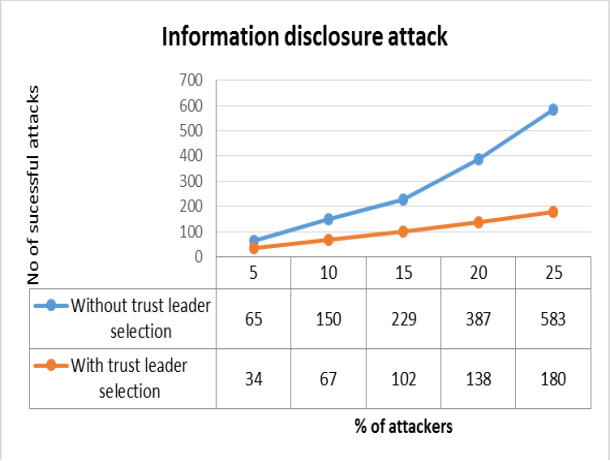

Figure 6 Information Disclosure Attack

The observations obtained from analysis are plotted in figure 6. From figure 6, it can be seen that the number of successful hack attempts are significantly more for different percentages of malicious node among the nodes. The successful hack attempts increases as the number of malicious nodes increases. The reason for such an observation depends on two things one the cryptographic algorithm used and the quack list maintained by the DSP. The use of a good cryptographic algorithm ensures confidentiality for messages and so compromise of confidentiality becomes less. In addition to that, the quack list prevents the reentry of the malicious nodes into the system again.

4.3.2 *Denial of Service:*

Denial of service is attack on the availability of the multicast service. The simulation is carried out for a sub-group and then generalized for the whole system. The sub-groups is simulated to have 50% partially armed hackers and 50% of fully armed hackers out of the malicious nodes in a sub-group under consideration. The partially and fully armed hackers are made to launch attacks in such a way that the service becomes unavailable to the subscribers or members of the sub-group. The malicious node is made to flood the DSP with join and leave requests. The DSP gets engaged in servicing the counterfeit requests rather providing the multicast service to the members so that the service becomes unavailable. Observations are made for number of successful DoS attempts for given different percentages of malicious nodes among the member nodes. The observations are made with and without the trust based leader selection implemented on G-LeaSel.

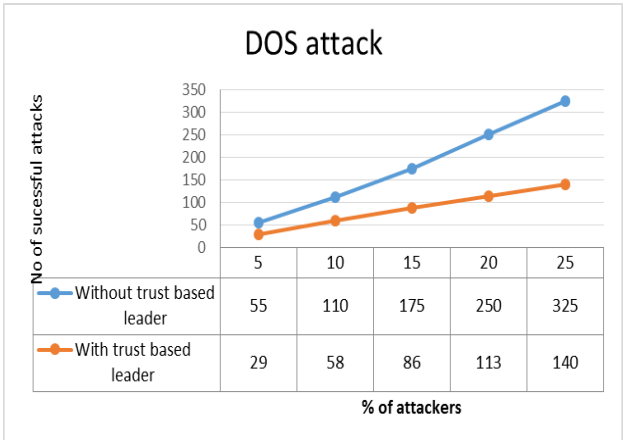

Figure 7 Denial of Service Attack

The observations are plotted in figure 7 and analyzed. From the graph it can be seen that the number of successful DoS attempts are significantly more for percentages of malicious node among member nodes. The difference increases rapidly (almost exponentially) for higher percentages of nodes. The reason for this performance is the quack list wherein, the malicious nodes are not allowed to reenter. Moreover, nodes behaving malevolently are expelled from the system. The trust computation sub-system monitors for malevolent behaviors like denial of service attacks and such nodes are expelled.

4.3.3 *Snooping*

Snooping Attack is defined as the attack on the identity of the leader. If the information about the identity of the leader is disclosed then an armed hacker can launch attacks to compromise the leader. The identity can be found by snooping the traffic flowing out of member nodes as the leader will be transmitting many packets relatively (due to rekeying).The simulation is carried out for a sub-group and then generalized for the whole system. The sub-groups is simulated to have 50% partially armed hackers and 50% of fully armed hackers out of the malicious nodes in a sub-group under consideration. The partially and fully armed hackers are made to launch attacks in such a way that the identity of the leader is disclosed. A probabilistic security approach is again used to simulate successful snoop attempts. An attempt is said to be successful if the random number generated and the sequence number of the packet are same. Observations are made for number of successful attempts for given different percentages of malicious nodes among the member nodes. The

observations are made with and without the trust based leader selection implemented on G-LeaSel.

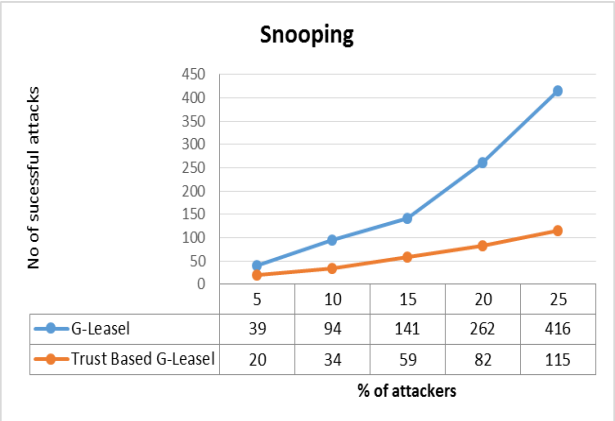

Figure 8 Packet Snooping

The observations are plotted in figure 8 and analyzed. From the graph it can be seen that the number of successful hack attempts are significantly more for different percentages of malicious node among member nodes. The difference increases rapidly (almost exponentially) for higher percentages of nodes increases. The reason for this performance is twofold one is the leader selection procedure and the other is the quack list maintained by the DSP. The leader selection sub-system changes the leader very often such that the information about the identity of the leader is disguised. The quack list prevents the reentry of the malicious nodes in to the multicast session.

5. CONCLUSION

The modified G-LeaSel model suits secure multicast in the Grid environments while incorporating a trust based leader selection methodology. The proposed model was designed and analysed though simulations in-terms of throughput, self-stabilization and security for multicast events in Grid environment. The Trust based G-LeaSel proves to be potential choice for a secure multicast security model for grid. This is a big stride forward towards solving the security problem for a wide class of applications. Thus, G-LeaSel with trust based leader selection algorithm is exhibits better throughput, self-stabilization without degrading the throughput, enhanced security, and proves to be a potential choice for a secure multicast security model for grid. A good extension of this work in future is the performance optimization of the trust based G-LeaSel in terms of computational complexity and load balancing which is indispensable in the grid environment. Moreover, in future the proposed model may be adapted for cloud environments as well.

6. REFERENCES

[1] Mary Vennila S, Sankaranarayanan V,(2008) , " P-LeaSel for Grid Environment", IJCSNS : International Journal of Computer Science and Network Security, Vol. 8, No. 4, pp. 55-64.

[2] Mary Vennila S, Sankararanarayanan V, et al (2007), " G-Leasel : A secure Multicast Model for Grid", IEEE Xplore, Communication Systems Software and Middleware (COMSWARE)

[3] Mary Vennila S, Sankaranarayanan.V,(2007), "Kerberized Leasel Model for Grid", IJCSNS International Journal of Computer Science and network security, Vol.6, No.9A,206 pp.154-160.

[4] Mary Vennila S,(2014), " Trust based Leader Selection Methodology for P-LeaSel, a Multicast Group Communication Model", International journal of Computer application, Vol. 91, No.9, pp 40-45.

[5] Mary Vennila S, Sankaranarayanan V,(2008), "Threat Analysis for P-LeaSel, a multicast group communication model", Asian Journal of Information Technology, Vol.7, pp. 64-68.

[6] Mittra S,(1997), " IOLUS : A frame work for scalable secure multicasting", Procedings of ACM SIGCOMM, pp. 277-288.

[7] Ballardie T, Crowcroft J,(1995) "Multicast specific Secutity Threats and counter measures", Proc. Symposium on Networks and Distributed System Security, San Diego, California, pp. 216-225.

[8] Butler D, Engert D, Foster I, KesselmanC, Tuecke S, Volmer J, Welch V,(2000), "A national Scale authentication infrastructure" , IEEE Computer, Vol.33, No.12, pp.60-66.

[9] Liu Jing, Zhou Mingtian,(2003), " Secure group communication for large dynamic multicast group", Journal of Electronics, Vol. 20, No.4, pp. 418-422

[10] Sivakami Priya S, Sumathi G,(2013), " In improved Security and Trusting Model for Computational Grid", International Journal of Grid and Distributed Computing, Vol. 4, No.1, pp. 57-63.

[11] Broadfoot P. and Martin A.,(2003) ,"Critical Survey of Grid Security Requirements and Technologies", Technical Report PRG-RR-03-15, Oxford University Computing Laboratory.

[12] Ch. Statya Keerthi.N.V.L, et.al, (2012), "Behaviour based Trust Management using geometric mean approach for Wireless Sensor Networks", International Journal of Computer Trends and Technology, Vol.3, Issue.2, pp 229-234.

[13] Jaydip Sen, (2010), "A Distributed Trust Management Framework for detecting malicious packet droping nodes in a mobile adhoc network", International Journal of Network security & Its Application (IJNSA), Vol.2, No.4, pp 92-104.

[14] Robson de Oliveira Albuquerque, Luis Javier García Villalba, and Tai-Hoon Kim, (2014), "GTrust: Group Extension for Trust Models in Distributed Systems", International Journal of Distributed Sensor Networks, Article ID 872842.

[15] Valarmathi J,et.all, (2011), "A Novel Trust management scheme in Pervasive Healthcare", IEEE-International Conference on Recent Trends in Information Technology, ICRTIT 2011, INDIA. June 3-5, 2011, proceedings pp.503-508.

[16] Valarmathi. J, et. all.,(2013), "An integrated approach for trust management based on policy, community adherence and reputation" Int. J. of Ad Hoc and Ubiquitous Computing. Vol. 13(2), pp.132 – 139.