Security against ARP Spoofing Attacks using Bayesian Support Vector Regression

C. Divya , D Francis Xavier Christopher

Abstract: ***In computer networks, Address Resolution Protocol (ARP) discovers the MAC address associated with an IP address but ARP lacks authentication in exchanging MAC address between hosts. This creates opportunity for hackers to employ ARP spoofing to damage the data and system. Many protection systems have been developed but the requirement to modify the basic network structure and expensive tools make it difficult to utilize. This paper investigates the practical limitations and considers the problem of detecting incompletely rectified adversaries from past sessions. For resolving this problem, Bayesian Support Vector Regression based ARP (BSVR-ARP) spoofing attacks protection mechanism is proposed. This mechanism considers the host configuration changes and network transmission errors for a probability prediction algorithm to detect the attackers. The attackers from the past sessions are also detected based on past knowledge and then using these detection results, they are either rectified or discarded from the network transmission routes. The detection and prevention of ARP spoofing has been accurate and effective in BSVR-ARP. The experimental results show that the proposed mechanism overcomes the limitations of other ARP spoofing prevention schemes and has higher accuracy of detection with minimal errors.***

Index Terms: Address resolution protocol, ARP spoofing, cache poisoning, Bayesian Support Vector Regression, probability prediction algorithm

I. INTRODUCTION

Ever increasing internet usage has been credited for the rapid improvements in the networking structure which created high availability, reliability and scalability for internet markets [19]. Information retrieval and the communication are the two major aspects of internet usage. However, there are many issues faced by the internet users for experiencing best applications. Security is the foremost issue in various applications areas like networking, databases, software services, and so on [14], [17]. Specifically, the security against the ARP spoofing attacks is considered to be the immediate need as the widely employed ARP protocol has vital security vulnerabilities [13].

Revised Manuscript Received on May 06, 2019

C. Divya , Research Scholar, School of Computer Studies, Rathnavel Subramaniam College of Arts and Science, Coimbatore, Tamilnadu

Dr D Francis Xavier Christopher, Director, School of Computer Studies, Rathnavel Subramaniam College of Arts and Science, Coimbatore, Tamilnadu

The ARP is widely utilized for detecting the Media Access Control (MAC) address corresponding to the IP address of a host which are stored in the ARP cache table for effective data resolution for user queries [26]. ARP cache poisoning is the compromise of host IP-MAC address pairs that result in opening for ARP spoofing attacks like Man-in-the-middle (MITM) attack [15], Denial-of-service (DoS) attack [1] and JavaScript insertion attacks [8].

Many researchers have tried to develop effective models to resolve the ARP cache poisoning and ARP spoofing attacks in the wired or wireless LAN networks [6]. Based on the properties, the approaches are classified into dynamic and static approaches. However, the prevention techniques of ARP poisoning attacks are not easier to develop because most approaches require changes to the network infrastructure and the tools needed changing network structure are also expensive. In practical environments, the host nodes are mobile which join and leave the subnet easily and hence the ARP poisoning attack prevention techniques becomes difficult to adapt these changes. The static approaches are not appropriate for the mobile hosts [2] while the dynamic approaches are supported only for dynamic IP addresses [22]. Additionally, these approaches require manual configurations by the network managers to ensure security against ARP attacks [27]. In this paper, two major problems are considered for the development of ARP attack prevention approach. The existing ARP attack hosts from previous sessions and the false detection of attackers are primarily considered which are occurred due to the transmission errors in the network and configuration changes in host. Therefore this research focuses on developing a new mechanism that prevents ARP poisoning attacks with less infrastructure changes.

The main contributions of this paper are the development of ARP attack detection and prevention mechanism that detects the attack hosts from both the current session and previous sessions. This mechanism considers the host configuration changes and transmission errors for determining the probability of attack hosts through a probability prediction algorithm of BSVR-ARP. This probability prediction algorithm is based on the Bayesian Support Vector Regression [25] to predict the features of the attackers in the analysed hosts using the past knowledge. Once the prediction results are obtained, the specified host node are analysed using attack detection module and the recovery module repairs or replaces the attack host from the legal ARP cache table. The remainder of this article is organized as follows: In section 2, recent research techniques for security against ARP spoofing attacks are discussed. Section 3 presents the proposed BSVR-ARP mechanism to prevent

ARP poisoning attacks in detail. In section 4, the performance of the proposed mechanism is evaluated and compared with that of other schemes for ARP spoofing attack prevention while the section 5 reports the conclusions.

II. RELATED WORKS

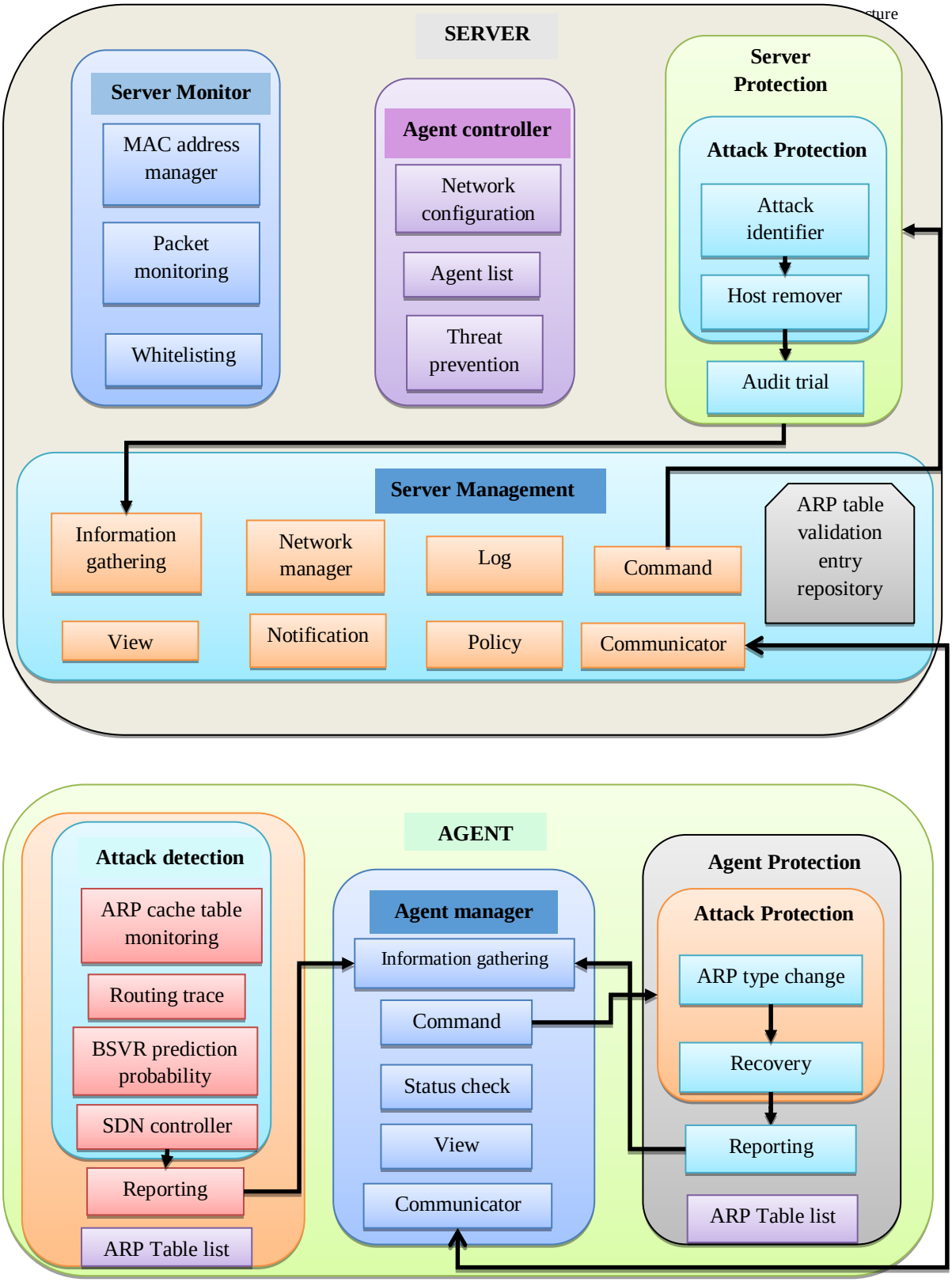
Recently, ARP cache poisoning and attack prevention has become a pivotal research interest among the system developers. The attackers utilize ARP cache poisoning to mostly initiate the man-in-the-middle attacks and causing the network performance to be downgraded with effective network service disruption. Some of the recent researches which focussed on ARP cache poisoning and attack prevention concepts are discussed in this section. Song et al, [16] presented an ARP spoofing attacks detection scheme called DS-ARP using the routing trace in the transmitted packets. This scheme utilized the trace routes to change the network and thus identifies the new nodes which are placed without authentication. Based on these traces, the malicious nodes can be eliminated from the network routing. However, this scheme is less competent when the new attack strategies are utilized instead of the already known and trained attack techniques. Moon et al, [9] also proposed a similar routing trace based system called RTNSS for detecting the ARP spoofing attacks in the networks and enhance the security. This RTNSS system installs an agent at the users and monitors the change in ARP cache tables. Based on these changes the abnormal hosts are determined and the routing traces are checked for the change from the recognized list of IP-MAC addresses. Thus detected attacker hosts are removed from the routes and this model does not require structure changes or traffic increased encryption for protection against the ARP attacks. However, RTNSS has limitations in detecting previously installed attacks and new strategies of attacks.

Srinath et al, [11] developed an ARP spoofing detection and prevention scheme by employing a centralized server. In this scheme, the drawback of voting based model is considered and in response a table based process is done by the centralized server which collects all the IP-MAC pairs of every node and maintains a legitimate host table. Based on this table of hosts, the malicious nodes performing ARP attacks are detected and prevented. However, one limitation of this scheme is that even unauthenticated host can also be registered in the network, which means the detection process at registration is not effective. Nam et al, [24] developed a new collaborative approach for detecting the MITM attacks in the co-existing wired and wireless nodes. This approach utilizes the fair voting concept for achieving uniform transmission without loss and the neighboring nodes decide whether malicious activities arise from a node through this voting process. This approach also improves the voting fairness through filtering of the voting reply messages and determining key voting related parameters. It provides ARP detection with less traffic overhead and higher accuracy. Similarly, Hijazi and Obaidat, [20] presented a new system using static entry for the detection and prevention of ARP attacks. In this system, the client/server model is equipped with IP-MAC table to detect and prevent ARP spoofing.

Ma et al, [12] presented Bayes-based ARP attack detection algorithm using the famous Bayesian theory. This detection algorithm calculates the probability of a host attacker using a prediction model to determine the normal and abnormal features of the host nodes. However this detection algorithm has the drawback of wrong detection when the attack occurs less frequently and the subsequent features gets added to the normal host features. Sakhawat et al, [10] presented an agent based scheme for ARP cache poisoning detection for detecting the MITM and DoS attacks. This approach has been installed over switch LAN system to detect and avert the insider malicious users. The detection accuracy of this scheme is high while there are only minimum effects on the system performance.

Prabadevi and Jeyanthi, [4] presented a security approach for protecting against the ARP attacks in large scale data center networks. Similarly, Prabadevi and Jeyanthi, [5] also developed Time Stamp and Counter based approach (TSCBA) for detecting the ARP cache poisoning attacks. TSCBA utilizes packet analyzer and cross layer checker for pre-processing and inspection of Ethernet header and ARP header MAC addresses. Then these addresses are checked with ARP table to detect abnormalities and the time stamp is generated along with alert messages to broadcast the entire system. This detection process is highly accurate and secured but the only drawback of using TSCBA is its cost expensive. Prabadevi and Jeyanthi, [3] also developed a similar approach for detecting and protecting against the ARP sniffing attacks through comparison of genuine IP-MAC pairs in ARP tables and Ethernet headers. These methodologies are dependent on the ARP table and Ethernet packet headers but when the packet formats are altered to resemble past session normal hosts, the detection process become complex.

Hong et al, [21] presented a protection model against the ARP attacks through AES and RSA data encryption schemes. This method averts the attackers by authenticating the data and does not require expensive equipment or protocol modifications. However this model is lighter application and only used under constrained environments. Likewise, Singh et al, [23] presented a two-phase validation scheme for detecting the ARP attacks and averting them. This model works on the basis of validation of the new binding acknowledged by each host for sensing pair of ICMP packets to old and new binding of ARP cache while new hosts are validated through the ARP packets from the claiming hosts. This model also prevents the flooding attacks but its limitation is the complexity in validating a new host. Based on the insights from these models in literature, the problem of detecting the presence of existing attacker hosts of past misbehaviours is considered to be resolved in this research using BSVR-ARP scheme.



III. ARP SPOOFING ATTACK DETECTION BASED ON BSVR PROBABILITY PREDICTION

Detecting an ARP attack can be effectively performed using the mapped IP-MAC address in the cache but the major limitation in this process is the inability to detect existing attacks from previous sessions. The major reason for this limitation is that the ARP cache based detection needs full system cooperation for fetching the previous session information. In general, the ARP packet frames are of two types: request and response packets. The IP-MAC address of requesting host and the IP of responding host are stored in the ARP response packets. This information obtained from each response packet is stored in the global ARP cache and can be used for determining attack features. But as said before, the limitation in accessing previous sessions must be resolved first. The Software defined networks (SDN) used a controller which can avert this limitation. Hence the SDN controller is employed to collect the ARP packets to create a global knowledge

base of all user sessions which can be utilized for attack detection. But the other problems of transmission errors and host configuration changes may alter the IP-MAC mapping cache and results in false detection. These problems can be overcome by using the proposed BSVR based probability prediction algorithm which iteratively predicts the attack features in each host. A similar mechanism of probability prediction for ARP spoofing detection has been developed in [12], but that approach has limitations when low frequent attack occurs in which case the abnormal behaviour is registered as normal features. The proposed BSVR based probability prediction is developed in such a way it mitigates this scenario.

A) Architecture of BSVR-ARP scheme

The proposed BSVR-ARP scheme consists of two main processes: BSVR prediction probability algorithm and SDN based attack verification and recovery. The architecture of the proposed BSVR-ARP is illustrated in Fig. 1 based on the ARP cache monitoring process. It has been modelled based on the basic server agent configuration in networks with ARP routing similar to that in RTNSS.

The BSVR-ARP architecture is composed of server and agent with separate management for each. The agent acts as the protector against the ARP spoofing while the server manages the agent information and acts as the decision maker in determining the ARP attackers based on information obtained through agents. The agent includes the agent detection, manager and protection modules. The agent detection module is the main module in this research where the major process of attack detection is carried out. This module includes the BSVR prediction probability algorithm and the subsequent routing trace based attack verification. This also includes the attacker host recovery process. The detection module initially compares the IP-MAC pair from the routing trace packets using the prediction probability on ARP cache list and global ARP cache table. This information is sent to the agent manager which confirms the attacker occurrence and informs the agent protection module to change the ARP type after which the recovery process is initiated. The agent manager also performs information gathering, command forwarding, self-status checking and communicating with the server manager.

The server includes the server monitor, server protection, agent controller and the server manager. Server monitor performs the MAC addresses monitoring of hosts, packet monitoring, duplication checking and whitelisting of attacker nodes. The agent controller maintains all the information about the agents like agent status, network control and network configurations and prevents from the threats. The server protection performs the decision making process on attackers and initiates the host removal from the network. The server manager has similar but greater processes than the agent manager. It collects the information about agents, hosts and server protection modules and manages network. It maintains log, network policies and ARP cache entry repository. It is the module through which the server communicates with the agent. The attack detection mechanism in the agent detection, agent protection and server protection are elaborated in this paper using the BSVR probability prediction and SDN controller based attack verification and recovery.

B) BSVR based attacker probability prediction

The global ARP cache and the network information from each host can be employed for detecting the attacks through general IP-MAC mismatches or duplication of MAC addresses. To achieve this, initially the network structure must be modelled and the subsequent host configurations are determined. Let the set $N = \{N_i, 1 \leq i \leq$ denote the network host with m host features which are placed in the network. These host set contain both normal as well as attackers. Based on past knowledge, the initial probability of the host being an attacker is set as $P($ while the probability of being non-attacker is $P(\bar{}$. As per the probability theory, the total probability is one, i.e.

$$P(H_j) + P(\overline{H_j}) =$$

(1)

Where $P($ and $P(\bar{}$ are the probabilities of host as attacker and normal host, respectively whose sum is always unity. When the SDN identifies a host feature , the posterior probability $P(H_j|N$ and is computed using the Bayesian theory as given by

$$P(H_j|N_x) = \frac{P(H_j)P(N_x|H_j)}{P(H_j)P(N_x|H_j)+P(\overline{H_j})P(N_x|}$$

(2)

$$P(\overline{H_j}|N_x) = \frac{P(\overline{H_j})P(N_x|\overline{H_j})}{P(H_j)P(N_x|H_j)+P(\overline{H_j})P(N_x|}$$

(3)

Where is the host feature that can either be attacker or normal feature.

The decision of SDN to categorize the host as attacker depends on the probability value of $P(H_j|N$, which means high value of $P(H_j|N$ denotes the host as attacker and hence this value has significant impact. But depending upon only this value is hypothetical as this prediction can also be false and wrong identification is quite possible. Hence modelling the Eq. (1) modelled based on Eq. (2) and (3) becomes hypothetical.

To resolve this drawback, the subsequent features of must be monitored and a feature list must be maintained. The feature list is built as $FS = \{FS_i, 1 \leq i \leq n + a; FS_i \in$. with $n +$ features; where is the attack features and is the normal features. On simplifying the posterior probability in iterative manner, the host can be verified either as attacker or normal host. For this iterative process, the BSVR is applied and the posterior probability of can be modelled as

$$P(FS|N_x) = \frac{P(FS)P(N_x|}{P(N_x)}$$

(4)

Here is represented as the hyper parameter features of Support vector regression for formulation and the prior distribution on these features are required to compute the probability $P($. For the estimation of $P($, the flat distribution is assumed such that its value is greatly insensitive to the and

hence alternative values of can be assigned to validate $P(N_x|F$. By the Bayesian nature of BSVR, the $P(N_x|F$ can be derived after an integral over the host in sampling function f based on the Taylor expansion. It can be written as

$$P(N_x|FS) = \int_0^{n+\alpha} P(N_x|f,FS)\ P(f,FS),$$

(5)

By employing the prior probability, the $P(N_x|F$ can be rewritten as

$$P(N_x|FS) = Z_f^{-1} Z_{FS}^{-1} \int_0^{n+\alpha} \exp(-FS(f)).$$

(6)

Here Z denotes the noise function associated with the transferred frames. When considering the Taylor expansion of the $FS($, the probability becomes expanded for F.

$$P(N_x|FS_i) = Z_f^{-1} Z_{FS}^{-1} \int_0^{n+\alpha} \exp(-FS_i(f) \dots\dots FS_{n+\alpha}(f)).\, df$$

(7)

Based on these probability obtained from the hyper parameters of BSVR, the probability of the attack prediction can be modelled as in Eq. (8)

$$P(H_j|FS_i \dots\dots FS_{n+\alpha}) + P(\overline{H_j}|FS_i \dots\dots FS_{n+\alpha}) = \frac{P(H_j)P(FS_i \dots\dots FS_{n+\alpha}|H_j)}{P(H_j)P(FS_i \dots\dots FS_{n+\alpha}|H_j)+P(\overline{H_j})P(FS_i \dots\dots FS_{n+\alpha}|\overline{H_j})} + \frac{P(\overline{H_j})P(FS_i \dots\dots FS_{n+\alpha}|\overline{H_j})}{P(H_j)P(FS_i \dots\dots FS_{n+\alpha}|H_j)+P(\overline{H_j})P(FS_i \dots\dots FS_{n+\alpha}|\overline{H_j})}$$

(8)

This can be simplified to form the BSVR final iterative prediction probability by retaining the conditional probability based on the feature list and host configurations.

$$P(H_j^i|FS_i(N_x)) = \frac{P(H_j^i)P(FS_i(N_x)|H_j^i)}{P(H_j^i)P(FS_i(N_x)|H_j^i)+P(\overline{H_j^i})P(FS_i(N_x)|\overline{H_j^i})}$$

(9)

Applying limit to this equation can help in determining whether the host is attacker or non-attacker. Based on the results obtained from taking limit of Eq.(9), the following three cases are inferred for iterative hosts:

Case 1: If $P(H$ increases with the occurrence of feature, then $P(FS_i(N_x)|H_j^i) > P(FS_i(N_x)|\overline{H}$, i.e. the probability of being an attacker is greater than that of a non-attacker.

Case 2: If $P(H$ value is close to 1 when the feature occurs, then the $P(FS_i(N_x)|H$ will also be close to 1; i.e. host should be an attacker in this case.

Case 3: If $P(H$ value is close to 0 when the feature occurs, then the $P(FS_i(N_x)|H$ will most probably becomes zero or closer to zero indicating the host is a non-attacker.

C) Attacker verification and response phase

Though the BSVR based prediction algorithm helps in identifying the attacker host effectively, the verification is carried out to avoid even the minor mistakes. In this verification phase, the SDN controller refreshes $P(H$ at each initialization to confirm the closing of previous sessions.

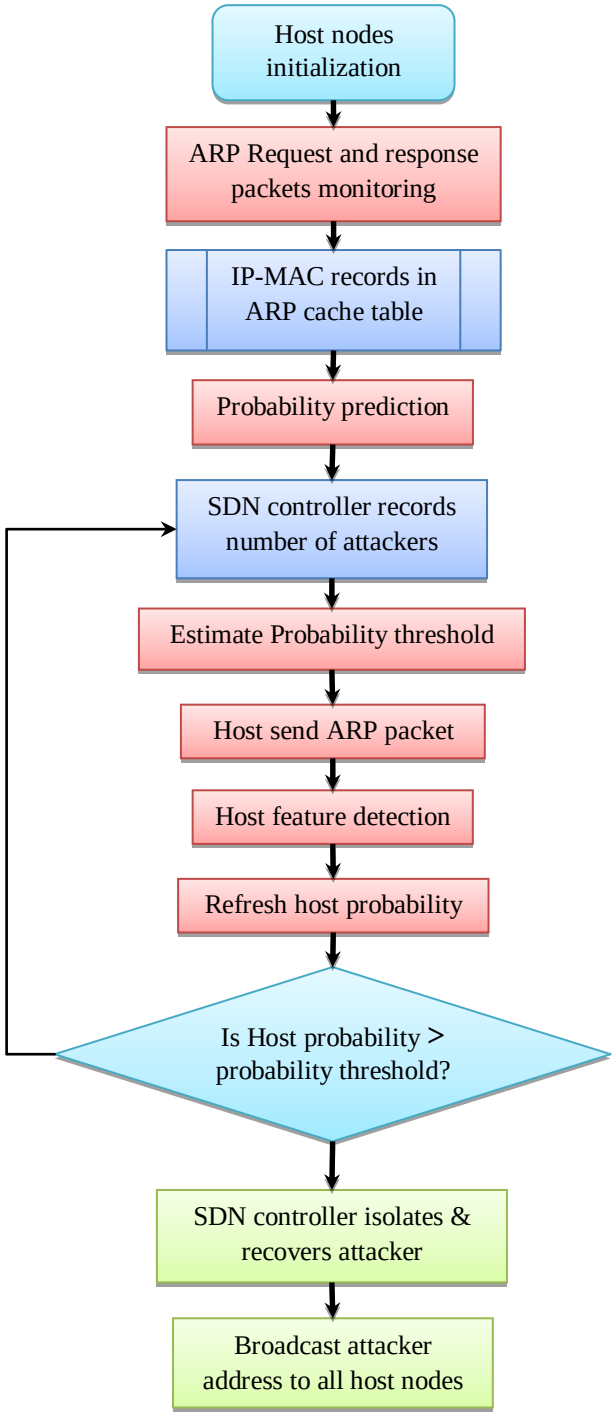


Fig.2. BSVR-ARP attack detection process

Only when the previous session is logged out, the session information will be moved to permanent memory. Based on the three cases obtained from applying limit on Eq. (9), the host is concluded as attacker if the following condition is satisfied.

$$P(H_j^i) > P_t, \qquad 0 < P_t <$$ (10)

Here is the threshold which has greater impact in misdetection and probability of exclusion. The value of must be dynamically adjusted to avoid the adverse effects. can be adjusted by monitoring the number of

attackers , in a fixed time period t with minimum probability given as P_m. Based on expected attackers and step size , the value is fixed between $P_{min} < P_t <$.

As the probability value for non-attacker must be close to zero, the P_m value is determined. When the P_m value is less than the threshold, P_m is taken as new threshold while if the threshold is almost equal to 1, 1 is selected as new threshold. In other cases, the combination of $P_t + (\alpha - A_n$ is used to fix the threshold. As mentioned before, the transmission errors and host configuration changes will increase the doubts on the host behaviour though the probability prediction results indicate it as non-attacker. Hence if the value of $A_n >$, the threshold must be minimized and if $A_n <$, then must be increased to avoid misdetection and increase detection accuracy. The attack detection process of BSVR-ARP based on the prediction probability and the threshold based attack verification is demonstrated in Fig. 2.

Once the attackers are verified, the SDN controller isolates the host from the network and initiates the host recovery process. The network starts to operate with new routes or with a new host in the place of attacker. In the recovery process, if the damage is less, the host can be recovered and included again in the network after prior tests. But if the damage is severe and irreversible, the host is not recommended to be included to the network in the foreseeable future.

IV. PERFORMANCE EVALUATION

A) Simulation environment

The performance of the proposed BSVR-ARP scheme is evaluated in a prototype simulation model developed using MATLAB. The model most similar to BSVR-ARP is the RTNSS which utilizes routing trace as the only source to detect the attacker host. BSVR-ARP framework utilizes the ARP cache data along with the routing trace to detect the attacker based on attacker probability prediction. The experimental environment for BSVR-ARP is given in Table.1. The network is setup based on these parameters with capacity to include 100 hosts which can be extended based on requirements.

TABLE 1
SIMULATION ENVIRONMENT

OS	Windows 8, 32bit
Processor	Intel core i5 3470 3.2 GHz
RAM	4GB DDR3
Storage	500GB Intel SSD SC2CT060A3 ATA device
Network bandwidth	1 Gbps
Simulation tool	MATLAB v.2013a
Simulation time	120 seconds
Network area	1000x1000 m
Packet size	80 bytes

B) Simulation results

The ARP spoofing detection performance of the proposed BSVR-ARP scheme is evaluated and compared with that of the schemes of KNN, SVM, Naïve Bayes and RTNSS for spoofing detection. As said above, the BSVR-ARP and the other schemes are tested in a framework similar to that of RTNSS. The detection schemes are compared in terms of detection time, attack accuracy, detection error, false detection probability and packet drop rate.

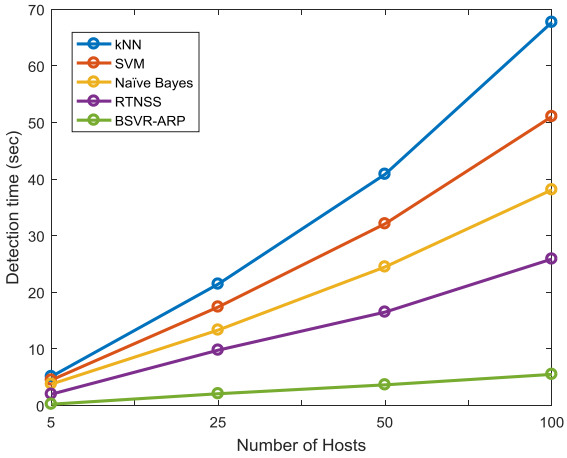


Fig.3. Detection time comparison

The detection time is the time taken by the proposed BSVR-ARP scheme to analyze and detect the presence of attacker in the network. The graph presented in Fig. 3 shows the detection time taken by the proposed BSVR-ARP and other existing ARP attack detection schemes. It can be seen that the proposed BSVR-ARP scheme takes less time to detect the ARP attacks than the other detection schemes. Compared with RTNSS in 100 host scenario, the detection time in BSVR-ARP is reduced by 60%. The utilization of local ARP cache table and global ARP cache list reduces the detection time in the proposed BSVR-ARP.

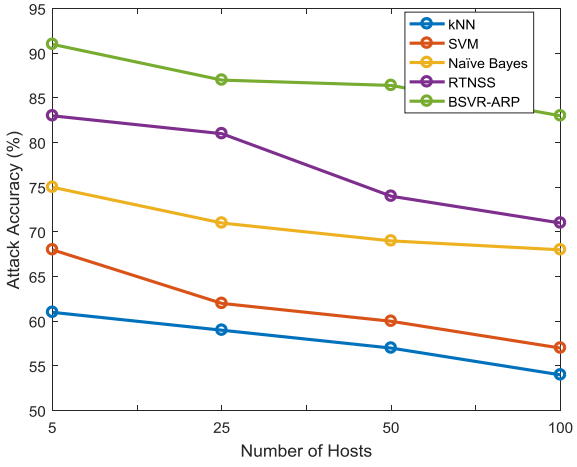


Fig.4. Attack accuracy comparison

Fig. 4 shows the comparison of the detection schemes in terms of attack detection accuracy. For this comparison, four attackers have been initialized when the number of host nodes is greater than 5 while 2 attackers are initialized for 5 host nodes. In this evaluation, the proposed model has detected higher percentage of attacks than the other compared schemes.

Compared with RTNSS in 100 host scenario, the attack detection accuracy in BSVR-ARP is increased by around 12%. The major reason for this improvement is the inclusion of BSVR based probability prediction algorithm that predicts the attackers based on features with higher precision.

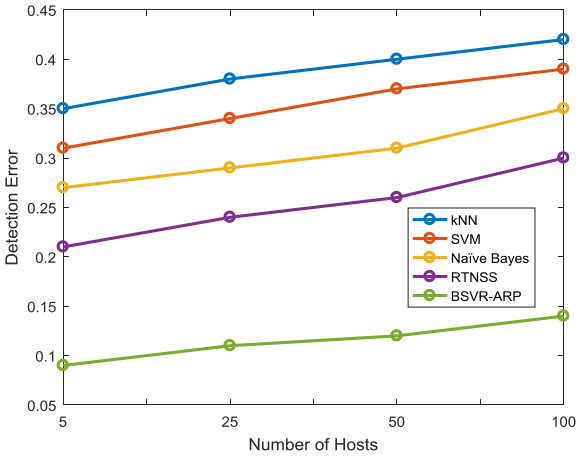


Fig.5. Detection error comparison

Attack detection error of the detection schemes is compared in the graph shown in Fig. 5. It is a known fact that due to increased host number, the traffic increases and the detection modules fluctuate to determine the normal and attack features based on ARP packets. Yet, the proposed BSVR-ARP has minimal effects of traffic in the overall performance that can be understood by this comparison of detection error. The BSVR-ARP scheme has very less detection error which is mostly near the negligible range than the other detection schemes. Compared with RTNSS in 100 host scenario, the detection error in BSVR-ARP is reduced by 50%.

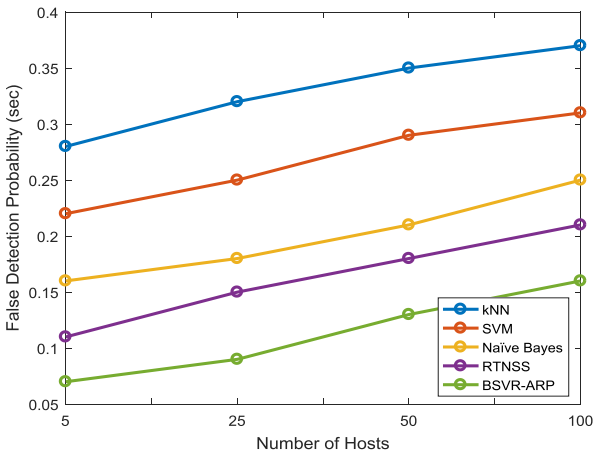


Fig.6. False detection probability comparison

The comparison of detection schemes in terms of false detection probability is shown in Fig. 6. False detection probability of a detection scheme is mostly dependent on the independent features that do not have any link with either of normal or attacker host. Literally, false detection is achieved when the number of ARP packets from a host increases due to various factors in such a manner, the detection scheme fails to identify the differing feature between attacker and normal host. In the above figure, it can be seen that the BSVR-ARP has less false detection probability than the other models. Compared with RTNSS in 100 host scenario, the false detection probability in BSVR-ARP is reduced by 23%.

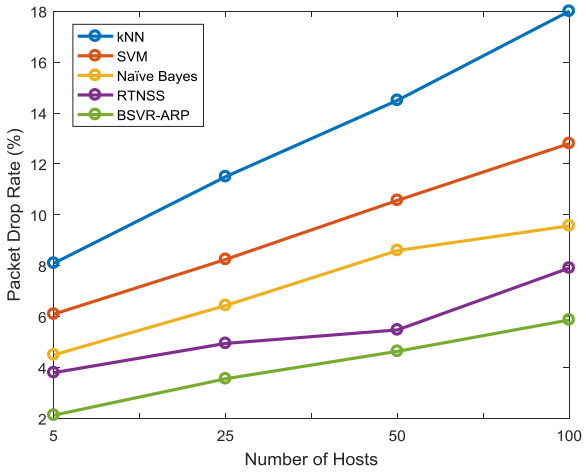


Fig.7. Packet drop rate comparison

Fig. 7 shows the comparison of the detection schemes in terms of packet drop rate. Packet drop rate is directly dependent on number of host ARP packets and traffic congestion. When the number of hosts in a network increases, the number of request and response ARP packets increases and causes traffic congestion and delay. This is the ideal situation for packet loss as well as vulnerable attacks. The above figure shows that the BSVR-ARP has less packet drop rate than the other schemes which is mainly due to prior attacker detection and uninterrupted host to server communication. Compared with RTNSS in 100 host scenario, the packet drop rate in BSVR-ARP is reduced by 2%.

C) Comparative analysis of BSVR-ARP

The simulation results illustrate the efficient performance of the proposed BSVR-ARP scheme in detecting and protecting against the ARP spoofing attacks. The comparative analysis of BSVR-ARP with other detection schemes can further justify this claim. For this comparison, the attacks are carried out in four scenarios where the number of hosts is increased in each scenario. The number of hosts considered each scenario is 5, 25, 50 and 100 respectively. 4 attackers are created in scenarios with 25, 50 and 100 hosts while 2 attackers are designated for 5 host scenario. The attacker detection from two scenarios i.e. 5 hosts and 50 hosts are shown along with the corresponding recovery process simulation output.

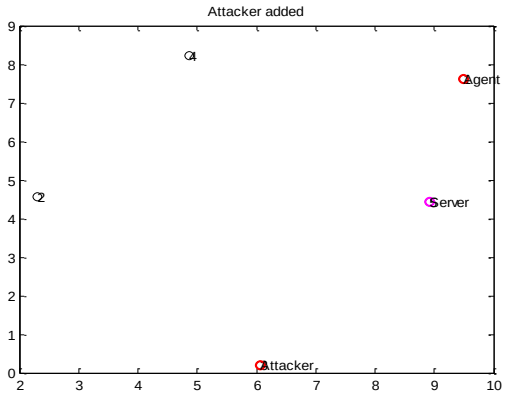


Fig.8. a) Attacker detection, 5 hosts

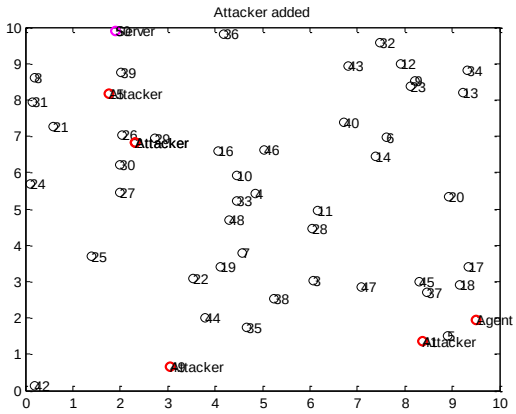


Fig.8. b) Attacker detection, 50 hosts

Fig 8 a) and b) shows the attacker detection output in 5 hosts and 50 hosts scenarios respectively. Fig 9 a) and b) shows the final process of isolating and recovering the attacker host for 5 hosts and 50 hosts scenarios respectively. BSVR has detected the attackers with higher accuracy and less time. The SDN controller utilized in BSVR-ARP has effectively isolated the infected hosts and creates new paths for transmissions while also initiating the process to recover the attacker hosts.

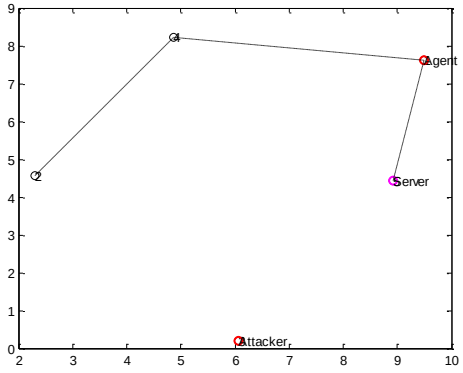


Fig.9. a) Attacker isolation and recovery, 5 hosts

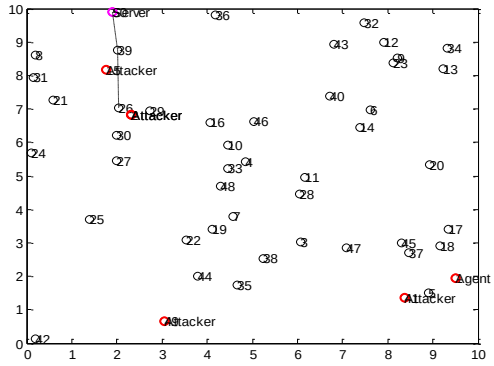


Fig.9. b) Attacker isolation and recovery, 50 hosts

In order to further evaluate the BSVR-ARP, a comparison with the most popular ARP spoofing detection schemes is performed. Table 2 shows the comparison of existing and proposed schemes based on efficiency. TSCBA [5], S-ARP [7], P-ARP [18], DS-ARP [16] and RTNSS [9] are compared with the proposed BSVR-ARP.

TABLE 2
COMPARISON OF ARP SPOOFING DETECTION SCHEMES

Features	**TSCBA**	**S-ARP**	**P-ARP**	**DS-ARP**	**RTNSS**	**BSVR-ARP**
Host cost minimization	L	L	L	L	M	H
Impact of warning and detection	L	L	L	M	H	M
Simplicity	L	L	M	H	M	H
Hardware cost minimization	M	M	L	M	H	M
ARP compatibility	M	H	M	H	H	H
ARP speed	L	M	L	M	M	M
Network load	M	M	L	L	L	L
Security	M	M	M	H	M	H

*L=Low; M=Medium; H=High

From Table 2, it can be seen that the proposed BSVR-ARP has high performance in terms of all features considered. Though in some cases like impact of warning and detection, RTNSS has better performance, comparatively BSVR-ARP scheme has best overall feature performance.

V. CONCLUSION

A novel ARP spoofing attack detection and protection scheme called BSVR-ARP is proposed in this paper using the BSVR algorithm. This includes the development of BSVR based attack probability prediction and SDN controller based attack verification and recovery. As this approach utilized the routing trace and ARP cache table information, the detection of the attacker is highly accurate. Additionally, this scheme has also identified the presence of existing attackers from previous sessions more effectively using the log data. The proposed scheme is tested in a prototype simulation

environment and the evaluation results are compared with existing schemes to determine the efficiency of BSVR-ARP. The results concluded that the proposed BSVR-ARP has better performance in detecting the ARP attackers with high accuracy and less time complexity. In future, the packet forwarding based attacks and ability to cope with the new attack strategies of the proposed model will be examined.

REFERENCES

1. A. Chonka, Y. Xiang, W. Zhou and A. Bonti, "Cloud security defence to protect cloud computing against HTTP-DoS and XML-DoS attacks," Journal of Network and Computer Applications, vol. 34, no. 4, pp. 1097-1107, 2011.
2. A. M. AbdelSalam, W. S. Elkilani and K. M. Amin, "An automated approach for preventing ARP spoofing attack using static ARP entries," International Journal of Advanced Computer Science and Applications, vol. 5, no. 1, pp. 105-112, 2014.
3. B. Prabadevi and N. Jeyanthi, "A framework to mitigate ARP Sniffing attacks by Cache Poisoning," International Journal of Advanced Intelligence Paradigms, vol. 10, no. 1-2, pp. 146-159, 2018.
4. B. Prabadevi and N. Jeyanthi, "Security Solution for ARP Cache Poisoning Attacks in Large Data Centre Networks," Cybernetics and Information Technologies, vol. 17, no. 4, pp. 69-86, 2017.
5. B. Prabadevi and N. Jeyanthi, "TSCBA-A Mitigation System for ARP Cache Poisoning Attacks," Cybernetics and Information Technologies, vol. 18, no. 4, pp. 75-93, 2018.
6. C. Modi, D. Patel, B. Borisaniya, H. Patel, A. Patel and M. Rajarajan, "A survey of intrusion detection techniques in cloud," Journal of network and computer applications, vol. 36, no. 1, pp. 42-57, 2013.
7. D. Bruschi, A. Ornaghi and E. Rosti, "S-ARP: a secure address resolution protocol," In Proceedings 19th Annual Computer Security Applications Conference, IEEE, pp. 66-74, 2003.
8. D. Gruss, C. Maurice and S. Mangard, "Rowhammer. js: A remote software-induced fault attack in javascript. In International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, Springer, Cham, pp. 300-321, 2016.
9. D. Moon, J. D. Lee, Y. S. Jeong and J. H. Park, "RTNSS: a routing trace-based network security system for preventing ARP spoofing attacks," The Journal of Supercomputing, vol. 72, no. 5, pp. 1740-1756, 2016.
10. D. Sakhawat, A. N. Khan, M. Aslam and A. T. Chronopoulos, "Agent-based ARP cache poisoning detection in switched LAN environments," IET Networks, vol. 8, no. 1, pp. 67-73, 2018.
11. D. Srinath, S. Panimalar, A. J. Simla and J. Deepa, "Detection and Prevention of ARP spoofing using Centralized Server," International Journal of Computer Applications, vol. 113, no. 19, pp. 26-30, 2015.
12. H. Ma, H. Ding, Y. Yang, Z. Mi, J. Y. Yang and Z. Xiong, "Bayes-based ARP attack detection algorithm for cloud centers," Tsinghua Science and Technology, vol. 21, no. 1, pp. 17-28, 2016.
13. H. S. Kang, J. H. Son and C. S. Hong, "Defense technique against spoofing attacks using reliable ARP table in cloud computing environment," In 2015 17th Asia-Pacific Network Operations and Management Symposium (APNOMS), IEEE, pp. 592-595, 2015.
14. K. Ren, C. Wang and Q. Wang, "Security challenges for the public cloud," IEEE Internet Computing, vol. 16, no. 1, pp. 69-73, 2012.
15. M. Conti, N. Dragoni and V. Lesyk, "A survey of man in the middle attacks," IEEE Communications Surveys & Tutorials, vol. 18, no. 3, pp. 2027-2051, 2016.
16. M. S. Song, J. D. Lee, Y. S. Jeong, H. Y. Jeong and J. H. Park, "DS-ARP: a new detection scheme for ARP spoofing attacks based on routing trace for ubiquitous environments," The Scientific World Journal, vol. 2014, 2014.
17. M. T. Khorshed, A. S. Ali and S. A. Wasimi, "A survey on gaps, threat remediation challenges and some thoughts for proactive attack detection in cloud computing," Future Generation computer systems, vol. 28, no. 6, pp. 833-851, 2012.
18. P. Limmaneewichid and W. Lilakiatsakun, "P-ARP: A novel enhanced authentication scheme for securing ARP," In Proc. 2011 Int. Conf. on Telecommunication Technology and Applications, pp. 83-87, 2011.
19. R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," Future Generation computer systems, vol. 25, no. 6, pp. 599-616, 2009.
20. S. Hijazi and M. S. Obaidat, "A New Detection and Prevention System for ARP Attacks Using Static Entry," IEEE Systems Journal, pp. 1-7, 2018.
21. S Hong, M. Oh and S. Lee, "Design and implementation of an efficient defense mechanism against ARP spoofing attacks using AES and RSA," Mathematical and Computer Modelling, vol. 58, no. 1-2, pp. 254-260, 2013.
22. S. Khurana, "A security approach to prevent ARP poisoning and defensive tools," International Journal of Computer and Communication System Engineering, vol. 2, no. 3, pp. 431-437, 2015.
23. S. Singh, D. Singh and A. M. Tripathi, "Two-Phase Validation Scheme for Detection and Prevention of ARP Cache Poisoning," In Progress in Advanced Computing and Intelligent Engineering, Springer, Singapore, pp. 303-315, 2019.
24. S. Y. Nam, S. Djuraev and M. Park, "Collaborative approach to mitigating ARP poisoning-based Man-in-the-Middle attacks," Computer Networks, vol. 57, no. 18, pp. 3866-3884, 2013.
25. W. Chu, S. S. Keerthi and C. J. Ong, "Bayesian support vector regression using a unified loss functions," IEEE transactions on neural networks, vol. 15, no. 1, pp. 29-44, 2004.
26. Z. Trabelsi and W. El-Hajj, "ARP spoofing: a comparative study for education purposes," In 2009 Information Security Curriculum Development Conference, ACM, pp. 60-66, 2009.
27. Z. Trabelsi and W. El-Hajj, "On investigating ARP spoofing security solutions," International Journal of Internet Protocol Technology, vol. 5, no. 1, pp. 92, 2010.

BIOGRAPHY

Ms. C. Divya is currently pursuing Ph.D in Computer Science in Rathnavel Subramaniam College of Arts and Science, Coimbatore under Bharathiar University. She has obtained her M.Phil (Computer Science) in the area of Network Security and Cryptography from Bharathidasan University. Her research areas include Network Security and Cryptography. She has published paper in International Conferences and Journals.

Dr.D.Francis Xavier Christopher received his Ph.D., in the area of Networking from Bharathiar University, Coimbatore in 2014 from Bharathiar University, Coimbatore. He obtained his M.Phil, in the area of Networking from Bharathiar University, Coimbatore in 2002. At present he is working as a Director, School of Computer Studies in Rathnavel Subramaniam College of Arts and Science, Coimbatore. His research interest lies in the area of Networking and Software Engineering. He has published 27 research papers in various reputed journals ranking with international standard. He served as a key note speaker for various research conferences country wide.