Trust Based Service Discovery in Mobile Ad-Hoc Networks

Gohil Bhumika, Mukesh A. Zaveri, and Hemant Kumar Rath

***Abstract*—Wireless mobile ad-hoc network (MANET) is a self-configuring network, which is composed of several movable devices connected with each other without wires. MANET is highly dynamic, multi-hop, and infrastructure-less in nature. In this dynamic environment, different nodes offering different services may enter and leave the network at any time. For better utilization of shared resources within the network, efficient and timely service discovery is required. Not only that, trustworthiness of the service provider is also an important thing to be considered, as all the devices are unknown to each other and network is formed on the fly basis. There is a need for a mechanism, which without creating much overhead in the system identifies the malicious behavior of a device. In our proposed approach service discovery mechanism is developed that efficiently discovers all the services within the network with minimum packet overhead as possible and trust model is applied on top of it for service selection. We have also conducted extensive simulations through QualNet and analyzed the performance of our proposed scheme.**

***Index Terms*—Ad-hoc networks, cross layer, service discovery, trust.**

I. Introduction

Mobile Ad-hoc Networks (MANETs) are networks consisting of mobile nodes such as laptops, cell phones, personal digital assistants, portable computers, etc. They are all equipped with wireless interfaces and communicate with each other without any reliance on infrastructure. Each device can behave as a client, a server and a router. These devices are moving at high or low speeds or they can remain stationary even at times; they enter of leave the system at any time. These networks are dynamic in nature because of mobility of nodes, adverse conditions of wireless channels, and the energy limitations of mobile nodes, all of which lead to frequent disconnections and/or node failures [1].

However, it is not sufficient to solve the problem of connectivity alone for the adoption of MANETs. Because their basic role is to allow mobile users to exchange data and use each other's services, there also is a need for architectures, mechanisms and protocols for service discovery. To benefit from these services, a device must be able to both locate them in the network and invoke them [2]. A service in the network is any tangible or intangible facility, it can be a computation, storage, a communication channel to other user or software a device provides that can be useful for any other device [3]. The process to locate a service within a specified environment that provides capabilities matching given non-functional criteria is referred to as discovery. In general, the purpose of service discovery comprises locating services that can satisfy given requirements, choosing the best between these service candidates [4].

To get acquaint with more clear idea about service discovery. Let us consider the scenario of a large shopping mall or suburbs in big cities where large number of people with different kinds of hand-held mobile devices are gathered. These devices may be diversely equipped with services such as music, GPS receiver or navigation, etc. A device which needs service, i.e., service like music should be able to locate the device which owns the required service and also willing to communicate with other devices in network if they want its service.

In case of wireless networks challenges such as node mobility, which affects service availability, frequent disconnections of the server or the client or intermediate nodes breaking or changing the path and the parameters of service selection, variations in channel, which leads to significant communication characteristics variations such as data rate, delay, etc may arise.

The existing strategies based on centralized service discovery do not work well with MANETs because of several reasons. First, in such networks a centralized server is difficult to maintain as the nodes can join or leave the network at random. Second, because of the dynamic nature of the network, every time a service leaves or joins the network it has to inform the centralized server about its activity [3]. When a client looks for a service and there are large sets of services present in the network offered by different providers, in addition to functionality, the reputation-based trust is also a key factor for service selection. It is also a critical task for the client to identify and maintain the list of reputable and trustworthy services and service providers [5].

In our proposed approach when a client interacts with the server in order to request service, after getting replies from all the available servers it first calculates the trust value for every service provider and establishes its connection with the server which contains highest possible trust value out of all the available responses.

This paper is organized as follows. In Section II, we discuss related work, and in Section III, we proposed our protocol. In Section IV, we conducted performance analysis and conclude the paper in Section V.

Manuscript received June 15, 2014; revised August 15, 2014.

Bhumika Gohil is with the Sardar Vallabhbhai National Institute of Technology, Surat, India (e-mail: bhumika18188@gmail.com).

Mukesh A. Zaveri is with the Computer Engineering Department, Sardar Vallabhbhai National Institute of Technology, Surat, India (e-mail: mazaveri@coed.svnit.ac.in).

Hemant Kumar Rath is with CTO Networks Lab, Tata Consultancy Services, Bangalore, India (e-mail: hemant.rath@tcs.com).

DOI: 10.7763/LNSE.2015.V3.210

II. Related Work

Mobile ad-hoc networks are formed by diverse unknown devices within nearby vicinity, with willingness to server the services which they own and want to acquire the service they do not have but can be offered by some device in network. These devices have limited battery power and mobile in nature. But still these users want wired kind luxury of sharing data and applications, which we termed as services. Service is any tangible hardware or software entity of a network. Service Discovery is a process of automatically finding services and their providers within the network in a timely manner. The open access nature of network makes the network vulnerable to attacks such as black hole attack, grey hole attack, selfish behavior and some other kind of malicious activity.

MANETs are usually deployed in harsh or uncontrolled environment, which increases the probability of compromises and malfunctioning as there is no centralized control unit to monitor the node operations. This characteristic makes a node cautious while communicating with other nodes because of environmental changes. Accordingly behavior of node may change apparently. There are myriad of approaches proposed in the literature for discovering service in a trusted manner. Konark-service discovery and delivery protocol which is mainly designed for ad-hoc networks proposed by [6], is a middleware based on a peer-to-peer model with each participating device having the capabilities to store its local services, query the network for other available services, deliver its own services using a resident micro HTTP server, and use the services it discovered in the network. Each device will have a local repository that will maintain the local services being offered by that device. It supports both advertisement and discovery of services. Since each device functions as an independent entity, events such as any participating device leaving or joining a network do not affect the functioning of the entire or a part of a network. But they have not considered security while designing the protocol.

In [7] authors have proposed a service discovery protocol with security features, the Secure Pervasive Discovery Protocol (SPDP). SPDP is a fully distributed protocol in which services offered by devices, without a central server. It is based on an anarchy trust model, which provides location of trusted services, as well as protection of confidential information, secure communications, or access control.

In [8] Authors proposed an ant-based resource discovery and mobility aware trust management for mobile grid systems. Initially, the super-grid nodes are selected in the network using ant colony optimization based on the parameters such as distance, CPU speed, available bandwidth and residual battery power. In order to maintain strong security with mobility management system, a proficient trust reputation collection method has been adopted.

A trust-based dynamic secure service discovery model for pervasive computing presented in [9] gives an idea of level wise bifurcation of devices based on quality of service and privacy they provide. A device is selected which has reasonably good processing capability, network capability, and storage capacity as a user agent of devices belongs to one service provider. The services of the same user register on the directory, so that the services of a same agent domain can be requested directly. The services are classified into three levels depending on the privacy of them. Different service levels apply to different service discovery methods. To avoid malicious users the trust management unit is introduced. A dynamic service discovery model is used to realize the trade-offs between discovery efficacy and privacy.

A trust based security mechanism for service discovery is proposed in [10] for MANET. Nodes using Swarm Intelligence Based Service Discovery Architecture discover the services. This mechanism computes confidence or trust value for each node considering its interaction and behavior in the network. The trust model designates different protection level for services. Based on this protection level of the services, secure communication is offered; the secure communication channel is established among nodes by sharing secret keys.

An ad-hoc network is a set of limited range wireless nodes that function in a cooperative manner so as to increase the overall range of the network. Each node in the network pledges to help its neighbors by passing packets to and fro, in return of a similar assurance from them. Like Routing protocols, data, battery power and bandwidth are the common targets of the attacks, which can range from passive eavesdropping to vicious battery draining attacks. In [11], authors utilize the information that can be gathered by analyzing packets forwarded, received or transferred by each node by using routing protocols such as AODV (Ad-hoc On Demand distance Vector routing protocol) and DSR (Dynamic Source Routing) and then based on this events trust can be computed. The information from these events is classified into one or more trust categories. Trust categories signify the specific aspect of trust that is relevant to a particular relationship and are used to compute trust in other nodes in specific situations. The events recorded during the trust derivation process are quantized and assigned weights to compute the situational trust values for different nodes. These trust values are then normalized. The situational trust values from all trust categories are then combined according to their assigned weights, to compute an aggregate trust level for a particular node. The aggregate and situational trust values are maintained and updated for each node based upon the frequency of events and severity of the situation.

III. Proposed Trust Based Service Discovery Approach

In this section we discuss our proposed approach and how efficiently it can deal with issues, which come across due to mobility and open access nature of Ad-hoc networks.

A. Problem Description

Let's assume that there is a network, composed of D devices, some devices offer S services, and expects to remain available in this network for T seconds. This time T is previously configured in the device, depending on its mobility characteristics. And there are some clients devices request services offered by servers. Each device has a service discovery mechanism enabled within it, which a process is working on the user's behalf to search information about

services offered in the network.

B. Service Discovery at Network Layer

Traditional Service discovery mechanisms mainly are designed for static environment and topologies, while in Mobile Ad-hoc networks topology and environment changes are very frequent so an efficient mechanism is required which can deal with these constraints of ad-hoc networks. Not only that it should be able to compete with topology changes and frequent disconnections, but also it should produce as minimum overhead as possible as transmission medium is wireless and mobile devices have limited battery power. Moreover, network layer statistics are also not available at application layer. To keep track of network traffic and also when multiple mappings are available for the same service, to select the best service out of all available, need to determine which one is having minimum response time or which one is more reliable to communicate with.

We extend service discovery approach used in [12], [13] in order to discover services within the network. Internet draft by Koodli and Perkins [14] outlined some general ideas for extending ad-hoc on-demand routing protocols to support service discovery.

AODV is one of the most prominent on-demand routing protocols for MANETs, and based on some ideas sketched by Koodli and Perkins in an Internet draft [14], we have extended the AODV protocol to perform service discovery.

In order to perform service discovery with AODV, we have extended the formats of the RREQ and RREP messages; not only new fields are defined, but also actions that nodes along the network need to perform when receiving these extended messages.

As service discovery will take place along with route discovery a client will bind the service name with RREQ packet, which we call as SREQ and it will get broadcast within the network with RREQ packet. Fig. 1 shows the extension we have proposed in RREQ message. An extension to the RREP message includes the following fields: a Type to identify SREP messages, the Length of the extensions, a service lifetime, the location information of a server (IP address of a server), as binding and service discovery happens at the same time in our case and Trust parameters as shown in Fig. 2.

Type (8 bits)	J (1 bit)	R (1 bit)	G (1 bit)	D (1 bit)	U (1 bit)	Reserved (11 bits)	Hop Count (8 bits)
RREQ ID (32 bits)							
Destination IP Address (32 bits)							
Destination Sequence Number (32 bits)							
Originator IP Address (32 bits)							
Originator Sequence Number (32 bits)							
Service Name (32 bits)							

Fig. 1. Extended route request packet.

A service table is maintained to store information about services provided by not only the current node but also by other nodes, whenever a node requires a service, it performs a lookup in its table. The information about the services offered by the current node is set when the node is initialized, while the information about the services offered by other nodes is acquired when the current node participates in a service discovery process. Each row in a services table contains the service identifier (a string that uniquely identifies the service), Node Id, a lifetime, service name, response time and trust value associated with every server.

Type (8 bits)	R (1 bit)	A (1Bit)	Reserved (9 bits)	Prefix Size (5 bits)	Hop Counts (8 bits)
Destination IP Address (32 bits)					
Destination Sequence Number (32 bits)					
Originator IP Address (32 bits)					
Lifetime (32 bits)					
Service Lifetime (32 bits)					
Service Name (32 bits)					
Server IP Address (32 bits)					
Trust Parameters (32 bits)					

Fig. 2. Extended route reply packet.

Service ID	IP Address	LifeTime	Service_ Name	Response Time	Trust

Fig. 3. Service table.

Lifetime is used to have a soft state and keep information up to date, which is mandatory in ad-hoc networks where there are frequent changes in topology. Every time when new reply is received from server, its lifetime field is updated.

Response time is one of the important metric considered to calculate trustworthiness of a node so it is maintained in the service table. Format of service table is shown Fig. 3.

C. Service Discovery Process

Wireless ad-hoc on-demand distance vector routing protocol is reactive routing protocol with route discovery and route maintenance. We have selected AODV as a base because of its reactive nature and also it creates fewer packets overhead as compared to other routing protocols. In order to achieve service discovery in AODV routing process, each node need to maintain a service table to record the service information. Due to reactive nature of the protocol we have used the "Pull based" service discovery approach where client floods the service request whenever it wants some service. The overall service discovery process takes place as follow.

Step 1: When client wants some service from the networks it first checks its own service table, if it has the service entry whose lifetime is not expired yet. If it has, then it will check whether it has route entry for that node. If it is also there then it will generate request and unicast it to the server. In case it doesn't have route entry, then it broadcasts request within the network. If it also does not have an entry for the required service in its service table then also it will flood request message within the network to acquire information, whether there is any server providing the required service available in network.

Step 2: Upon receiving a message from a neighbor, a node will first check whether it has route information available for the requested node or not. If it has, then it will generate reply containing route information of destination node. If it does not have that information and it is not the destination as well (destination in terms of whether a node provides the service requested by client) then it would simply replay the request message to the neighboring node.

Step 3: If the node has the service requested by a client, it will initiate reply to the client. It will bind the service information into the service reply packet such as lifetime of

the service provided by it, its own IP address, battery and mobility information. It will set the destination address to its own address and will unicast the reply message to the client node.

Step 4: Upon receiving the reply, node will first check whether this packet has the destination address same as its own address (if it is the source of the request or not). If it is not the client node then it will relay the packet to the next node. If the node itself is a source of the request then it will check whether its service table contains the same server information, if it already has the service information then it will update the lifetime of the service. If it does not have an entry of that service provider it will create an entry into the service table.

There is a possibility that same service is provided by multiple servers within the network. In that case client will receive multiple replies from different servers. Client node will use trust model to calculate trust value of that server and will select the server with the highest trust value. Fig. 4 shows the flowchart of the service discovery process carried out at network layer.

As we embed service discovery with routing mechanism our main emphasis is on service discovery Hence our approach modifies the basic AODV protocol to satisfy the need of service discovery and core idea is to find services within the network and then their routes.

D. Trust Based Service Discovery Model

Despite the advances in the area of wireless networks, service selection is still a challenging problem for service oriented computing. Several approaches have been developed to support the selection of services based on one, or a combination of functional, behavioral, quality, and contextual aspects presented in [15]. However the use of QoS (Quality of Service) information supplied by service providers, or even behavioral information, is not enough to distinguish between good and bad services during the selection process.

We consider the trust management schemes presented in [11], [16], [17] for establishing trust in our system. This is efficient in not only giving trust information of each and every server but also efficiently discover the malicious behavior of a server. When the node, which wants to communicate with server node and it, does not have any prior knowledge about it, it broadcasts service request within the network. Here with service reply it not only gets the service information, but also some metrics of the server. Once service discovery process gets over, client node calculates trust values based on the metrics information received from server. We consider the following matrices to evaluate trust. Weights are assigned to this metrics in order to calculate trust.

1) *Mobility of a node*: while binding reply to client server adds its mobility information into the service reply packet. More the mobility of a server, higher the probability that server may leave network, which lessens the possibility that server will be selected for communication.
2) *Battery*: This is also an important aspect to be considered, as mobile devices have limited battery power, it is necessary that server with maximum battery should be considered for selection.
3) *Response Time*: This is another important criterion of service selection, we do not select service based on the minimum hop count information but the time it takes to give response to the service request gives us information about the congestion of a network and can come to know how busy the server is.

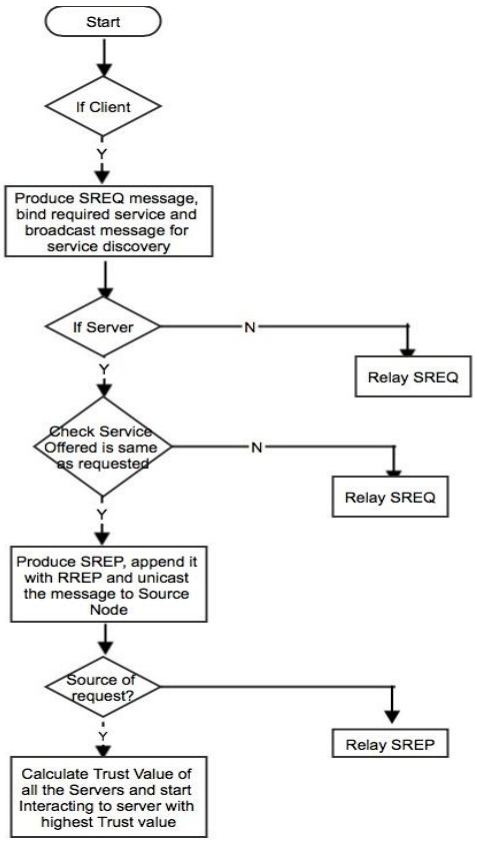

Fig. 4. Flowchart of service discovery process.

4) *Life Time*: This information helps in a way that if the server lifetime is about to get over it will no longer be able to give response to the requested service so better to select the service with greater lifetime in the network.
5) *Packet Drops*: If a node which claims to be a server but continuously dropping packets is not an appropriate choice to communicate. So, negative weight is associated with this as it is considered as a malicious activity.

By assigning weights to this metrics the Trust value of a node is calculated using as equation below:

$$T(X_c) = \sum_{i=0}^{n} W_i \times P_i \quad (1)$$

where:

$T(X_c)$: trust value of a server X calculated for a client c,

W_i: weight assigned to i^{th} parameter metrics,

P_i: value of i^{th} parameter metrics.

While intermediate nodes relay service reply, they also cache the service information and calculate the trust value of that server and maintain it in their service tables. This way

trust information of a server can propagate within the network.

Once client receives replies from all the servers within the network it selects one of them with the highest trust value and starts communicating with that server. Client node periodically keeps calculating trust value of all the servers which are providing the same services and when it discovers a more trustworthy and reliable server than the current server it is communicating with is available in the network, it stoops its communication with the current server and starts new session with the new server which has higher trust value. This way trust values are calculated dynamically and accordingly servers are selected in our proposed scheme.

IV. Experimental Evaluation

We have conducted experiments using Qualnet wireless network simulator. Each experiment has been run 5 times with different simulation intervals. We used 802.11 MAC at MAC layer transmission range is 350 m. QualNet [18] simulator provides model for AODV protocol, which we have modified according to our need. To simulate real traffic Constant Bit Rate (CBR) is used at application layer. This generates data packets at the interval of 5 sec. We employed our nodes in the area of 1500 × 1500 m dimensions. We experimented with mobile and static scenarios. To provide mobility Random Way Point model is used. We set minimum speed to 0 m/s and maximum speed to 10 m/s and pause time is set to 30 sec. We consider the battery of a node as an important criterion so that linear battery model is applied.

Experiments were carried out with simulations time of 30, 60, 900, 1800 seconds, with 20, 40, 60, and 80 numbers of nodes, 4, 8, 12, 16 numbers of servers with static and Mobile scenarios.

A. Performance Metrics

Here we are going to discuss performance results according the metrics we chose to evaluate are analyzed and comparisons are made between basic AODV protocol and our proposed Trust based service discovery model.

Service Discovery Time: Response time which is also considered as service discovery time in our proposed scheme is shown in Fig. 5, which depicts that the time our mechanism takes to discover services within the network gradually increases as we increase the number of nodes in the network. It can be derived that density of the network affects the service discovery time.

B. Control Packet Overhead

As service discovery is incorporated at network layer with routing protocol AODV, control overhead produced by our mechanism is analyzed in terms of the RREQ packets generated by the client node in order to find service providers in the network. Fig. 6 shows that in static scenario, our proposed scheme do not produce much overhead but when mobile scenario is considered, frequent disconnections are possible. In that case client floods request packets into the network to find route to reach the destination server, at the same time it also discovers new service, which may have entered in the network.

Comparison is made between basic AODV and our modified AODV, which is termed as SD-AODV with both the mobile and static scenarios. Increasing the simulation time does not affect much in static scenarios but when mobility is added, increase in simulation time results in increased number of request packets sent for service discovery.

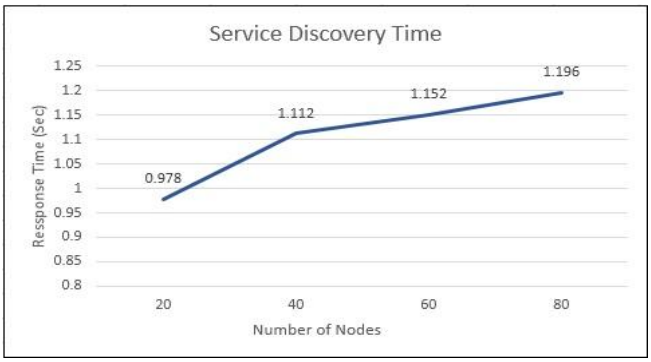

Fig. 5. Service discovery time.

C. Service Discovery Success Rate

It needs to be analyzed that how efficiently our proposed approach discovers the service providers in the network. Fig. 7 shows that our mechanism successfully discovers all the available services, which match with our requirement in the network. In this figure we have shown the graph of number of servers which offers requested service and number of services discovered by our mechanism. It can be said that success rate of our service discovery mechanism is 100%.

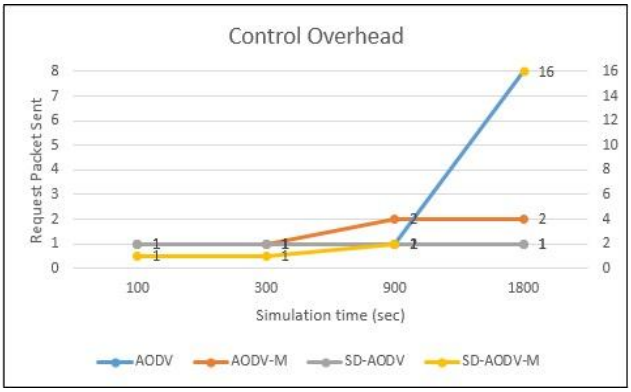

Fig. 6. Control overhead produced by trusted service discovery.

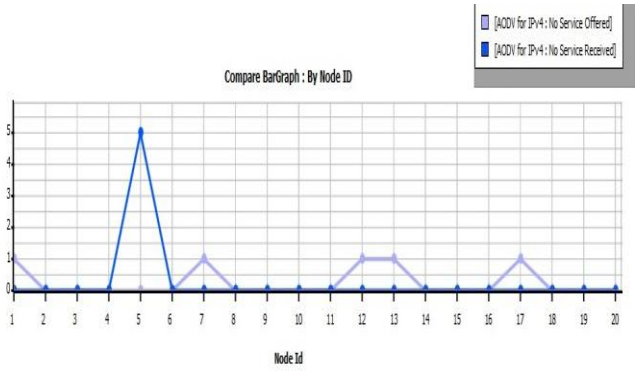

Fig. 7. Service discovery success rate.

V. Conclusion

We have proposed trust based service discovery scheme, which not only efficiently discovers all the available service

providers within the network but also finds their trust value and communicates with the server with highest trust value. As trust dynamics changes, client dynamically calculates the trust value of a server and communicates with the server with higher trust value.

Due to resource constraint environment of wireless networks it is important to utilize resources as efficiently as possible. Here therefore we have selected routing based cross layer service discovery approach and also verifying trustworthiness of a node while communicating. This is another important issue which has been addressed. As energy is an important issue in mobile environment incorporating trust to substitute traditional security mechanism for secure communication is an idea presented in the proposed scheme. Implementing these techniques in real network is a challenge and appropriate methods should be discussed, which can be considered as future work direction of this paper.

REFERENCES

[1] C. N. Ververidis and G. C. Polyzos, "Service discovery for mobile Ad Hoc networks: A survey of issues and techniques," *Communications Surveys & Tutorials*, vol. 10, no. 3, pp. 30-45, 2008.

[2] A. N. Baldoni and R. Beraldi, "A survey of service discovery protocols in multihop mobile Ad Hoc networks," *Pervasive Computing*, vol. 8, no. 1, pp. 66-74, March 2009.

[3] M. Uday, C. A. Kevin, and M. Elizabeth, "Scalable service discovery in mobile Ad Hoc networks," in *Proc. Networking 2004*, 2004, vol. 3042, pp. 137-149.

[4] D. Ding, L. Liu, and H. Schmeck, "Service discovery in self-organizing service-oriented environments," in *Proc. 2010 IEEE Asia-Pacific on Services Computing Conference (APSCC)*, Dec. 2010, pp. 717-724.

[5] L. Baresi, C. H. Chi, and J. Suzuki, "Trust-oriented composite service selection and discovery," *LNCS*, pp. 50–67, 2009.

[6] S. Helal, N. Desai, V. Verma, and L. Choonhwa, "Konark - A service discovery and delivery protocol for ad-hoc networks," *Wireless Communications and Networking*, vol. 3, pp. 2107-2113, March 2003.

[7] C. Campo, F. A. Arez, D. Danial, C. G. Rubio, and A. M. Lopez, "Secure service discovery based on trust management for ad-hoc networks," *Journal of Universal Computer Science*, vol. 12, no. 3, pp. 340-256, 2006.

[8] A. Singh and P. Chakrabarti, "Ant based resource discovery and mobility aware trust management for Mobile Grid systems," *2013 IEEE 3rd International Conference on Advance Computing Conference (IACC)*, Feb. 2013, pp.637-644.

[9] F. Shen, Q. Pei, and S. Bu, "A trust-based dynamic secure service discovery model for pervasive computing," in *Proc. Seventh International Conference on Computational Intelligence and Security*, 2011, pp. 630-634.

[10] S. Pariselvam and R. M. S Parvathi, "Trust based security mechanism for service discovery in MANET," *Journal of Theoretical and Applied Information Technology*, vol. 56, no. 2, pp. 226-234, 2013.

[11] A. A. Pirzada and C. McDonald, "Trust establishment in pure Ad-hoc networks," *Wireless Personal Communications*, vol. 37, issue 1-2, pp. 139-168, 2006.

[12] J. A. Garcia-Macias and D. A. Torres, "Service discovery in Ad-hoc networks: better at network layer?" in *Proc. International Conference on Parallel Processing*, 2005, pp. 452-457.

[13] J. Zhong, S. Geng, L. Weng, and X. Li, "A cross layer service discovery protocol for MANET," *Journal of Computational Information Systems*, vol. 8, no. 12, pp. 5085-5092, 2012.

[14] R. Koodli and C. Perkins, "Service discovery in on-demand Ad hoc networks," *IETF draft*, 2002.

[15] S. E. Athanaileas, C. N. Ververidis, and G. C. Polyzos, "Optimized service selection for MANETs using an AODV-based service discovery protocol," presented at Annual Mediterranean Ad Hoc Networking Workshop Corfu, 2007.

[16] I. D. Silva and A. Zisman, "A framework for trusted service," *Service-Oriented Computing*, Heidelberg, Berlin: Springer, 2012, pp. 328-343.

[17] I. R. Chen, J. Guo, F. Bao, and J. H. Cho, "Trust management in mobile ah hoc networks for bias minimization and application performance maximization," *Ad Hoc Networks*, vol. 19, 2014, pp. 59-74.

[18] QualNet. (Dec. 6, 2013). Scalable Network Technology. [Online]. Available: http://web.scalable-networks.com/content/qualnet

Bhumika B. Gohil was born at Gandhidham, Gujarat on January 18, 1988. She has completed her bacholer's degree in computer engineering from GEC, Gandhinagar, India in 2009. Then she served as a lecturer at Babaria Institute of Technology, vernama, Gujarat. She pursued her M.Tech from S.V. National Institute of Technology, Surat, Gujarat, India in 2014. Her research interests are wireless networks, operating system and embedded systems.

Mukesh A. Zaveri is serving as an associate professor at Computer Engineering Department, Sardar Vallabhbhai National Institute of Technology, Surat, Gujarat, India since 1993.

Mukesh obtained his PhD from Indian Institute of Technology, Bombay in 2005. His research interests are signal processing, wireless network, mobile computing, image processing, machine learning.

Hemant Kumar Rath is a senior research scientist at Network Lab, Tata Consultancy Services, Banglore, India, where he is working since Dec. 2010.

Hemant is a senior member IEEE, member IARCS. He held his MTech and PhD from IIT Bombay, India both in Communication Engineering and BE in EL&TCE from VSSUT Burla (formerly UCE Burla), Sambalpur, Odisha. He has almost 15 years of experience in academics, research and industry.

His current research interests include SDN, WiFi offloading, QoS in networks, LTE/WiMAX scheduling, self-optimization, propagation model design, speech processing, M2M communication, cloud computing, modelling of social network traffic etc. He has published many research papers and presented many talks in national and international conferences/seminars such as IEEE Globecom, ICC, PIMRC, COMSWARE, COMSNETS, NCC, ITU-T, BWCI-COAI Workshop etc., and has filled several patents through TCS. He is also participating in national and international standardization activities (IoT – GISFI, IoT – ITU-T, DOSTI, TSDSI-3GPP) in the areas of networking and communication.