WCA-DGVC: A Weight Clustering Algorithm for Decentralized Group Key Management with Variable size Cluster

Pooja Singh, Nasib Singh Gill

Abstract**:** ***Wireless Ad hoc networks are experiencing a rapid increase in its applicability as well as in security threats. The wireless communication medium makes them highly prone to security attacks. Key management plays a vital role in secured communication. Power efficient and secure key management is one of its major requirements. Group key management is a promising approach for efficient cryptographic key management for MANETs. In this paper, we proposed a weight clustering algorithm for a decentralized group key management. The whole network is divided into smaller subgroups called clusters. The cluster is locally managed by the cluster head (CH). The CHs mutually manage the security key process. All nodes have equal opportunities to take part in CH selection. The CHs are selected by a weight clustering algorithm based on the computational power and the neighbor count of the node. The elected CH selects next CH from its neighbor by comparing their computational power, neighbor nodes and their distance from it. This eliminates the need of gateway nodes for inter-cluster communications. The size of the cluster is directly proportional to the weight of the cluster head that is the cluster head with high weight will manage the large cluster. Therefore the group key management activities are proportionally divided among the cluster heads according to their power. This eliminates the risk of frequent drowning of cluster heads. The performance of our algorithm is assessed through stimulation and compare with two popular weight clustering algorithms*****.**

Index Terms**: Cluster, Decentralized group key management, weight clustering algorithm, Wireless ad hoc ntwork.**

I. INTRODUCTION

Wireless Ad hoc network is a collection of autonomous node communicating with each other without the aid of any central authority and infrastructure. In such a network, each node not only acts as a host but also as a router that forward packets to other nodes that may be multiple hops away from each other. This area arises in late 90's and suddenly a new era of wireless networking starts. No need of infrastructure and instant network built up gives new directions to the networking. Increased use of hand-held devices as well as evolution of wireless communication has spurred the applicability of ad hoc networks. Ad hoc networks are ubiquitous in nature and can be used anytime, anywhere with limited or no infrastructure [1].

Previous studies have explored many of its research areas. Some of the research areas of ad hoc networks are optimal routing, efficient security and guaranteed quality of service. The absence of fixed infrastructure, dynamic routing, and limited battery power device complicate the security process. Maintenance of security in wireless networks is much more complex than the wired networks. An ad hoc network being a wireless in nature is affected by various vulnerabilities. Absence of secure boundaries, deficiency of central management system and more importantly the hazards that may exist from compromised nodes within the network makes the ad hoc network more vulnerable than wired one. Formation of firewall and gateway for boundary security is not applicable in ad hoc network. Any node that comes in radio range of a node can penetrate the network. The freedom to join and leave the network complicates its management [2]. Cryptographic key management is one of the efficient ways to maintain security in wireless networks. Previous researches have shown that the multicast communication has benefit for battery operated devices. Group key management is based on the principles of multicast. A centralized group key management where a single node performs all key related tasks cannot completely overcome the complexity of low computation power and dynamic topology and have single point of failure. A distributed group key management where all network nodes contribute in the mechanism of key management becomes infeasible in case of large network size. A decentralized group key management where the whole network is broken into small and manageable subgroups appears to be more promising than other approaches [3]. We propose a clustering algorithm for a decentralized group key management scheme which enhances the security parameters and simultaneously consider the constraints of low computation and dynamic positioning of nodes. This paper is divided into following sections. Section 2 explains the related work in this field, section 3 explains the proposed weight clustering algorithm for a decentralized group key management with variable size cluster (WCA-DGVC), section 4 provides the experimental results and section 5 conclude our proposal.

II. RELATED WORKS

Junpei et al. [4] proposed self-stabilizing algorithms that assume a network like a vertex-weighted graph and a weight assignment technique is used to make the cluster stable by echoing the mobility of every node to its weight.

Revised Manuscript Received on 30 January 2019.
***** Correspondence Author

Pooja Singh*, Department of Computer Science, Govt. College for Women, Faridabad, India.

Nasib Singh Gill, Department of Computer Scienceand Applications, Maharshi Dayanand University, Rohtak, India.

In [5] an enhanced maximum stability weighted clustering algorithm is proposed where the cluster heads are selected by considering the weight of node. The weight of each node is calculated on the basis of node degree, node energy and relative speed of a node. The minimum weight of a node promises its position in cluster head selection. This algorithm reduces the number of cluster but the inter cluster communication among the cluster heads that increase the network traffic is not considered properly. V. S. Anitha and M. P. Sebastian [6] proposed a weighted and adaptive algorithm where the energy level of node acts as the base for cluster head selection. The high energy level guarantees the CH position. This procedure executes extra computational steps that create burden on the network traffic. In [7] the cluster head is selected on the basis of number of their neighbor nodes. The nodes are assigned an ID according to its degree. The value of ID is inversely proposal to its degree. A node with smallest ID will declare itself as CH. This algorithm considers the number of neighbor nodes for CH selection to reduce the number of clusters but inconsideration of its computational power can cause repetitive CH selection. J. Sathiamoorthy and B. Ramakrishnan [8] proposed a hybrid scheme for dynamic cluster formation. Two algorithms are combined to attain the goal of stability of cluster. This algorithm consumes more energy for cluster member as well as cluster head selection. In [9] the cluster heads are selected on the basis of mobility. A node with lowest change in its mobility pattern as compare to its neighbor nodes is considered for CH selection. This algorithm improves the network stability but rapid change in mobility pattern may bring inefficiency. In [10] a weight clustering technique is implemented to reduce the risk of forwarding hello packets in cluster formation process. This algorithm considers the mobility pattern, degree and the time for which it is active to detect its battery power for weight calculation. In [11] clusters are formed according to region-clustering based on hyper sphere. This algorithm shows positive result only when the node distribution is normal. In [12] a self-organization clustering algorithm based on zone is proposed. This utilizes the bio-inspired behavior of birds flocking for the formation and maintenance of clusters. In this paper, we proposed a weight clustering algorithm for a decentralized group key management technique where the large group is subdivided into smaller subgroup called cluster. The cluster heads (CH) are elected by a procedure based on computational power, the number of neighbors and the distance between them. This technique effectively selects the cluster heads that have more computational power and more approachability to other nodes. Hence, the selected CHs will effectively management the cryptographic keys for secure communication.

III. PROPOSED MODEL OF WEIGHT CLUSTERING ALGORITHM FOR A DECENTRALIZED GROUP KEY MANAGEMENT WITH VARIABLE SIZE CLUSTER (WCA-DGVC)

We propose a weight clustering algorithm for a decentralized group key management protocol with variable size cluster (WCA-DGVC). Our scheme is design to overcome the drawback of extra computational function of cluster heads to manage the security key. In decentralized key management, the whole network is divided into small subgroups or clusters. The cluster heads performs the key related activities like key generation, distribution and re-keys distribution. These computational overloads exhaust the cluster head. WCA-DGVC opt three methods to overcome this limitation. First the selected CH are those that have high processing and energy power among the other nodes and second all CHs are in direct communication range and third the number of cluster members are according to the weight of the CH.

WCA-DGVC has advantage over other similar approaches [6] [12]. The extra computational overhead created by key management activities are handled by most capable nodes. These capable nodes that act as cluster heads are those how have high computational power and have more number of neighbor nodes among the other nodes. The size of the cluster is determined according to weight assigned to it. The cluster head with high weight value will have more number of cluster members than others and this will proportionally distribute the overall group key management activities among all the cluster heads.

A. System Parameters

The various parameters used in the system are discussed in this section. Table 1 show the various symbol used.

Table 1 Description of symbols

Parameters	Symbols	Parameters	Symbols
Total number of Nodes	n	Computational power of node	CV
Cluster Member	CM	Neighbor Count	NV
Cluster Head	CH	Distance Ratio	DR
Candidature Index	CIN	Weight Index	WIN

Computational Power (CV): The computational power of the node CV determines the strength of the node to remain alive in the network. Computational value of a node is the mean of the following parameters [13].

Processing power: CPU – It represents the available CPU capacity of the node. Memory- It represents the available memory capacity of the node.

Energy: Battery- It represents the available battery level of the node. The parameter values are computed in the following way:

$P_{CPU}(t) = 1 - load_{CPU}(t);$ (1)

where $load_{CPU}(t)$ is the actual fraction (between 0 and 1)of the node's CPU load. The CPU parameter value is time dependent and nodes with higher CPU load have lower value of this parameter.

$P_{mem}(t) = 1 - \frac{loadmem\,(t)}{MAXmem};$ (2)

where $MAX_{mem}$is the maximum memory capacity, while $load_{mem}(t)$ is the actual memory load in MByte on the node, respectively. The memory parameter value is also time dependent and higher memory load on the node results in lower parameter value.

$P_{bat}(t) = 1 - load_{bat}(t);$ (3)

where $load_{bat}(t)$ is the actual fraction (between 0 and 1) of the node's battery load. The battery parameter value is time dependent and higher battery load results in lower parameter value giving an indication about the remaining battery power. The computational value of a node is the average of equation (1), (2) and (3).

$CV = [P_{CPU}(t) + P_{mem}(t) + P_{bat}(t)]/3$

The CV falls in the range between 0 and 1. The value of CV determine its capability to participate in key management activities while value very close to 0 means that the node has not enough power to remain alive in the network.

Neighbour Count (NV): The neighbor count, NV, of a node is the ratio of the number of its neighbor nodes to the total number of nodes in the network.

Candidature Index (CIN): The capacity of a node or its candidature to become a subgroup or cluster head is the average of its computational value (CV) and its neighbor count (NV). All members calculate their computational value CV and the neighbor count NV and propagate their CIN to all approachable nodes. Each node maintains a candidature table as shown in Table 2.

Table 2 Candidature Table.

Node	1	2	------	n
CIN			------	
DR			-----	
WIN			-----	

Where distance ratio (DR) is the ration of the distance between the nodes and its transmission range. Weight index (WIN) is the average of CIN and distance ratio. After receiving CIN from neighbor nodes, all nodes update their Candidature table. And then again propagate the updated values to their neighbor nodes. These rounds of propagation will continue until candidature table is not getting completed. The number of message exchange for the completion of candidature table has lower bound 1 and upper bound approx. log(n).

Proposition 1: Number of rounds to complete Candidature table has lower bound 1 and upper bound ≈ log(n).

Proof: To prove above proposition, we consider the two extreme situations with an ad hoc wireless network having 8 nodes. The first extreme situation when all nodes are in direct range of all other nodes i.e. every node has n-1 neighbor nodes (Fig. 1). In this situation only one round of CIN propagation will complete the candidature table. Thus, the lower bound is 1.

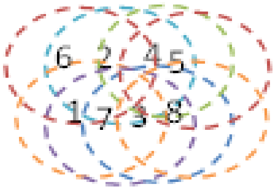


Fig. 1: All nodes are in direct range of remaining nodes.

Secondly, when each node is in the range of only two neighbor node making a circular chain of connectivity(fig. 2(a)) or an open chain with two end nodes having only one neighbor node(fig. 2(b)).

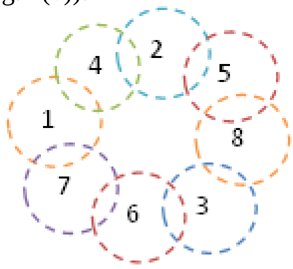


Fig. 2(a): All nodes are in range of only two nodes.

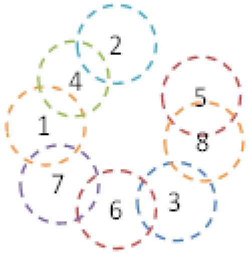


Fig. 2(b): All nodes are in direct range of two nodes except for two nodes that makes the ends of the connectivity chain.

The number of propagations required for CIN exchange to complete candidature table is increased linearly by the function f(n) where f(n)=2*f(n-1)-1 with the increase in the number of nodes as depicted in Table 3. The situation depicted in fig. 2(a) & 2(b) require approximately log(n) number of propagations round to complete the candidature table. Hence, the upper bound for above proposition is ≈log(n).

Table 3: Number of propagation rounds required for completion of candidature table.

Number of nodes (n)	**Number of propagation rounds**	**Number of nodes (n)**	**Number of propagation rounds**	**Number of nodes (n)**	**Number of propagation rounds**
1	1	7	3	13	4
2	1	8	3	14	4
3	1	9	3	15	4
4	2	10	4	16	4
5	2	11	4	17	4
6	3	12	4	…	…

B. Cluster formation

After the completion of candidature table, the node with highest CIN starts the process of selection of cluster head. The process of CH selection and cluster formation is summarized in Algorithm1 and Algorithm2.

Algorithm1

1. Nodes exchange their candidature Index CIN with the neighbour nodes and update the candidature table.
2. The process is repeated until the candidature table is not got completed.
3. On completion of candidature table, the node with highest CIN declares itself as cluster head and

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

initiates the formation of cluster.

4. The elected CH forms the cluster and declares its neighbours as its cluster members.
5. The elected CH check the condition that if the number of its cluster members is n-1 then the process is completed otherwise algorithm2 is initiated.

Algorithm2

1. The CH selects the highest weighted (WIN) node among its neighbours. Declares it the next CH and exclude it from its cluster membership.
2. The selected CH forms a new cluster and declares its neighbours as its cluster members. The CH will not include any node that is previously selected either as CH or CM.
3. The process is completed when the union of all CH's and CM's is equal to the total number of nodes in the network otherwise algorithm2 is repeated.

C. New node joins the cluster

When a new node joins the group, one of the two situations may arise. First the new node comes in the range of any cluster head then only rekeying is required for backward secrecy. Second, the new node doesn't come in the range of any CH. In this case the CH selection process is restarted.

D. Node leaves the cluster

When a node leaves the network, one of the two situations may arise. First the leaving node is not a CH. Then only rekeying of the respective cluster is performed for forward secrecy. Second if the leaving node is a CH, then the whole process of CH selection is repeated and a new key is generated for forward secrecy.

E. Reselection process

The key management activities like key generation, distribution and re-distribution of secure keys create extra computational burden on CH. The reselection process circulates this burden. After even interval of time, the reselection process restarts the process of cluster formation so that the cluster maintenance workload is shifted to more capable nodes at that particular time. This eliminates the risk of frequent drowning of CH and increase the life span of the network.

F. Inter cluster communication

The first selected CH chooses the next CH according to algorithm2 from its neighbor nodes. Similarly, the next CH is selected. Therefore all CH's come in transmission range of previous and next selected CH that eliminate the need of gateway nodes during the inter cluster communication.

IV. EXPERIMENTAL STUDIES

A customized simulation in MATLAB is used to evaluate the performance of WCA-DGVC. The parameters used for the simulation are listed in Table 4.

Table 4 Simulation parameters

Simulation Parameters	Value
Simulation area	100m x 100m
Number of nodes	40,60,80,120
Transmission range	20m,30m,40m,50m
Mobility model	Random walk model
Simulation time	500 seconds

The simulation results are compared with two recent similar clustering algorithms SOCZBM [12] and DSCAM [6]. Fig. 3 shows the number of clusters formed for network size of 40, 60, 80 and 120 nodes. The number of clusters formed is decrease with increase in transmission range of nodes. When the transmission range increases, the cluster size is also increased and thereby reduces the number of clusters formed. In fig. 4, fig. 5, fig. 6 and fig. 7, a comparison of results of SOCZBM, DSCAM and our algorithm WCA-DGVC is shown. For different network sizes of 40, 60, 80 and 120 nodes under different transmission range is simulated for these three algorithms. The results of WCA-DGVC are better than the SOCZBM and DSCAM. The number of cluster formed with WCA-DGVC algorithm is lesser then the other algorithms.

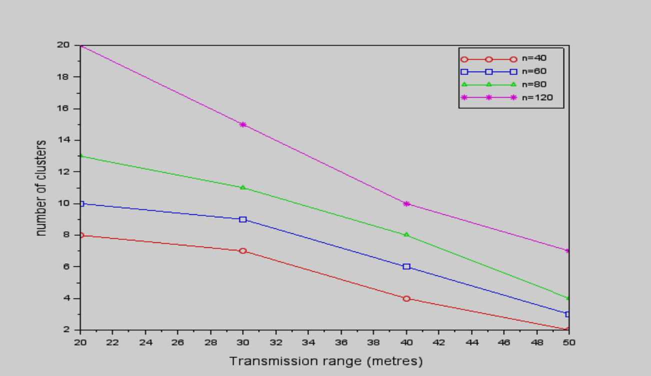


Fig. 3: Transmission range against number of clusters (WCA-DGVC)

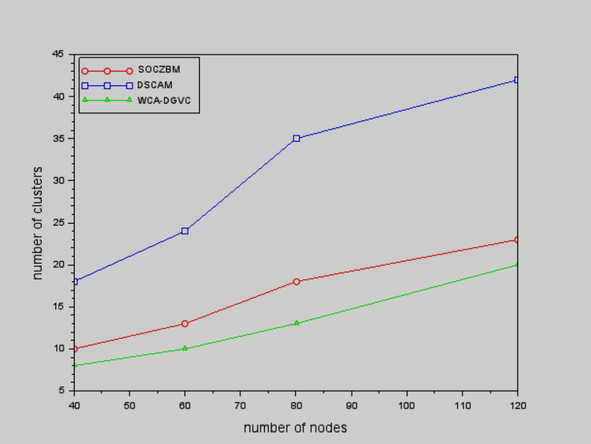


Fig. 4: Comparison of number of clusters in SOCZBM, DSCAM and WCA-DGVC with transmission range of 20m.

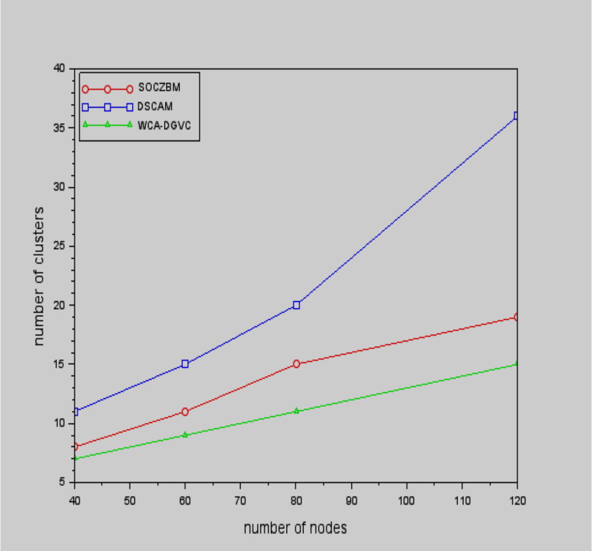


Fig. 5: Comparison of number of clusters in SOCZBM, DSCAM and WCA-DGVC with transmission range of 30m.

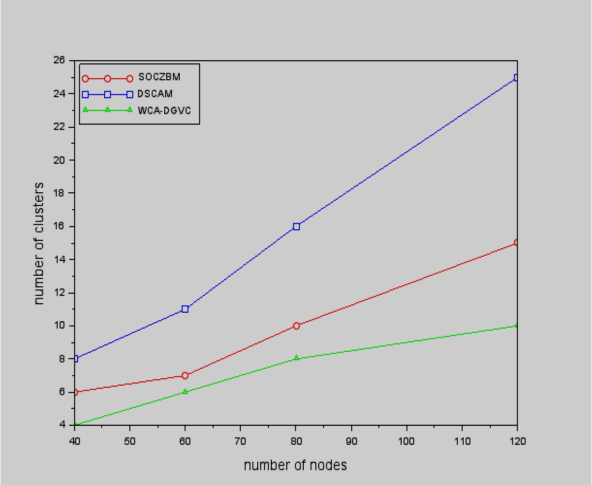


Fig. 6: Comparison of number of clusters in SOCZBM, DSCAM and WCA-DGVC with transmission range of 40m.

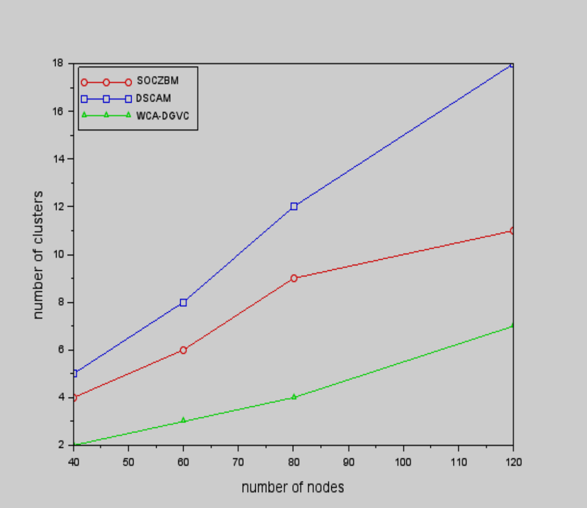


Fig. 7: Comparison of number of clusters in SOCZBM, DSCAM and WCA-DGVC with transmission range of 50m.

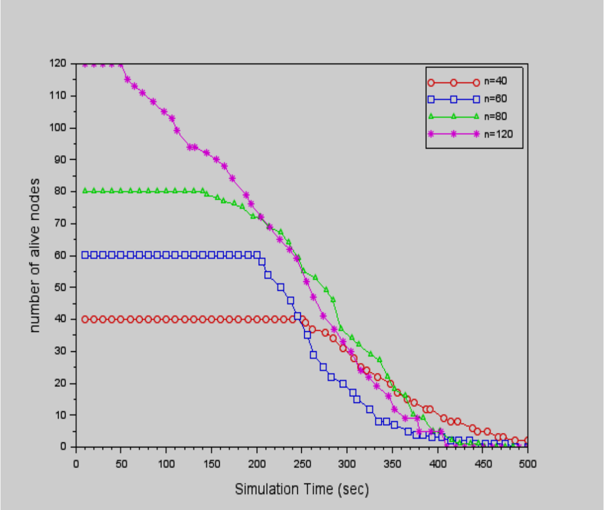


Fig. 8: Number of alive nodes under different network size.

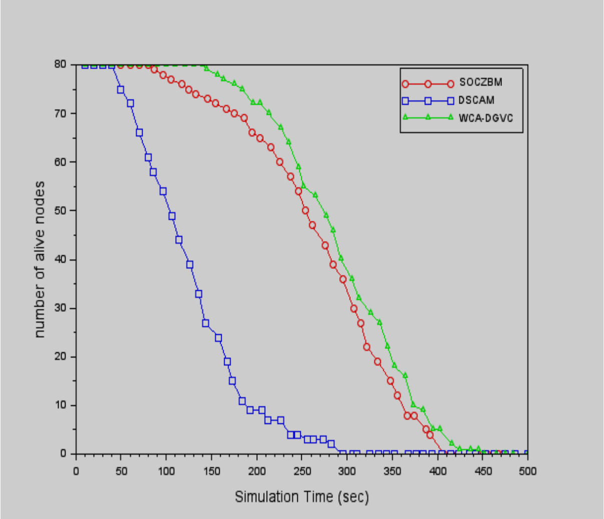


Fig. 9: Comparison of number of alive node under SOCZBM, DSCAM and WCA-DGVC schemes with network size of 80 nodes.

The life span of network with different number of nodes under WCA-DGVC algorithm is depicted in fig. 8. The simulation results show the smooth decline in the number of alive nodes. WCA-DGVC algorithm selects the most capable nodes as CH and according to their weight the cluster size is determine. The process is repeated after a threshold time. This decreases the number of exhausted nodes. Fig. 9 shows a comparative result of network life span of SOCZBM, DSCAM and WCA-DGVC. The number of alive nodes under WCA-DGVC algorithm is higher than the other two algorithms.

V.CONCLUSION

WCA-DGVC is a weighted clustering algorithm for a decentralized group key management protocol with variable size cluster (WCA-DGVC). It is based on idea of assigning the computational work of secure key management to those nodes which are more capable among the other nodes.

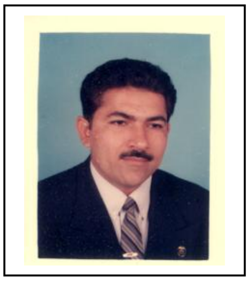
SOCZBM and DSCAM clustering algorithm also consider the computational power and number of neighbor nodes as a deciding factor while selecting CH but the interlinking of all CHs in WCA-DGVC eliminate the need of gateway node and that simplifies the extra burden on CHs. The reselection process of WCA-DGVC re-allocates the work to most capable nodes after a certain period of time. This increases the life span of network. The simulation results show the better performance of WCA-DGVC over SOCZBM and DSCAM.

REFERENCES

1. Basagni, S., Conti, M., Giordano, S., Stojmenovic, I.: Mobile Ad Hoc Networking: Cutting Edge Directions, Second Edition, Chap-1, John Wiley and Sons (2013)
2. Zhang, Y., Lee, W.: Security in Mobile Ad-Hoc Networks. In: Ad Hoc Networks Techologies and Protocols, Springer (2005)
3. Rafaeli, S., Hutchison, D.: A survey of Key Management for Secure Group Communication, ACM Computing Surveys, pp. 309-329, Vol. 35, No. 3, September (2003)
4. Kuroiwa, J.,Yamauchi,Y., Sun,W., Ito,M.: A self-stabilizing algorithm for stable clustering in mobile ad-hoc networks. IEEE (2011)
5. Tao, Y., Wang, J., Wang, Y. L., Sun, T.,: An enhanced maximum stability weighted clustering algorithm in ad hoc network. In: Proc. 4th Int. Conf. Wireless Commun. Netw. Mobile Comput. Pp. 1-4 (2008)
6. Anitha, V. S., Seastian, M. P.,: (k,r)-dominating set-based, weighted and adaptive clustering algorithms for mobile ad hoc networks, IET Commun., vol. 5, no. 13, pp. 1836-1853 (2011)
7. Wang, X., Cheng, H., Huang, H.,: Constructing a MANET based on clusters. Wireless Pers. Commun., vol. 75, no. 2, pp. 1489-1510 (2014)
8. Sathiamoorthy, J., Ramakrishnan, B.,: Energy and delay efficient dynamic cluster formation using hybrid AGA with FACO in EAACK MANETs. Wireless Netw., vol. 23, no. 2, pp. 371-385 (2017)
9. Cai, M., Rui, L., Liu, D., Huang, H., Qiu, X.,: Group mobility based clustering algorithm for mobile ad hoc networks. In: proc. APNOMS, pp. 340-343, August (2015)
10. Maragatham, T., Karthik, S., Bhavadharini, R. M.,: TCACWCA: transmission and collusion aware clustering with enhanced weight clustering algorithm for mobile ad hoc networks, Cluster Computing, https://doi.org/10.1007/s10586-017-1574-0 (2018)
11. Salma, B. U., Lawrence, A. A.,: Improved group key management region based cluster protocol in cloud. Cluster Computing. https://doi.org/10.1007/s10586-017-1455-6 (2017)
12. Aftab, F., Zhang, Z., Ahmad, A.,: Self-Organization Based Clustering in MANETs Using Zone Based Group Mobility. IEEE Access https://doi.org/10.1109/ACCESS.2017.2778019 (2017)
13. Farkas, K., Hossmann, T., Plattner, B., Ruf, L.,: NWC: node weight computation in MANETs. In: Int. Conf. on Computer Commun. And Netw, pp. 1059-1064 (2007)

AUTHORS PROFILE

Pooja Singh was born in New Delhi, India. She has completed her Master of Computer Application in 2002. She is presently working as Assistant professor in Govt. College for Women, Faridabad, Haryana. She has 12 years of teaching experience. Her research interests include Information Security, Cryptography and Network Security, Security in Ad Hoc Networks. She has published papers in Conferences and Journals.
Email:poojasinghpooja77@gmail.com



Prof. Nasib Singh Gill is presently working in the *Department of Computer Science & Applications, M. D. University, Rohtak, Haryana (India)*. Professor Gill earned his Doctorate in Computer Science in year 1996. He carried out his Post-Doctoral research at Brunel University, West London during 2001-2002. He has published more than 150 research papers in National & International Journals, Conference Proceedings, Bulletins, Books, and Newspapers. Professor Gill is presently working and guiding researchers in the areas - *Measurement of Component-based Systems, Complexity of Software Systems, Component-based Testing, Computer Networks & Security, Data mining & Data warehousing, and NLP.*