RESEARCH ARTICLE OPEN ACCESS

Byzantine-Resilient Protected Multicast Routing In Multihop Wireless Networks

Bhima Sankaram Alladi[1], M.Venu Gopal [2], L.Vandana [3],M.Sukesh [4]
Department of Computer Science & Engineering in **St.Martin's Engineering College**, R.R Dist, AP, and India.

Abstract
Multihop wireless networks rely on node cooperation to provide multicast services. The multihop communication offers increased coverage for such services but also makes them more vulnerable to insider (or Byzantine) attacks coming from compromised nodes that behave arbitrarily to disrupt the network. In this work, we identify vulnerabilities of on-demand multicast routing protocols for multihop wireless networks and discuss the challenges encountered in designing mechanisms to defend against them. We propose BSMR, a novel secure multicast routing protocol designed to withstand insider attacks from colluding adversaries. Our protocol is a software-based solution and does not require additional or specialized hardware. We present simulation results that demonstrate that BSMR effectively mitigates the identified attacks.
Key Terms: Multihop wireless networks, secure multicast routing, Byzantine resiliency, Byzantine attacks.

I. INTRODUCTION

MULTICAST routing protocols deliver data from a source to multiple destinations organized in a multicast group. In the last few years, several protocols were proposed to provide multicast services for multihop wireless networks. A major challenge in designing protocols for wireless networks is ensuring robustness to failures and resilience to attacks. Wireless networks provide a less robust communication than wired networks due to frequent broken links and a higher error rate. Security is also more challenging in multihop wireless networks because the open medium is more susceptible to outside attacks and the multihop communication makes services more vulnerable to insider attacks coming from compromised nodes. Although an effective mechanism against outside attacks, authentication is not sufficient to protect against insider attacks because an adversary that compromised a node also gained access to the node's cryptographic keys. Insider attacks are also known as Byzantine attacks and protocols able to provide service in their presence are referred to as Byzantine-resilient protocols. Security aspects in multicast protocols relate to either routing-specific security, such as managing the routing structure and data forwarding, or application-specific security, such as data confidentiality and authenticity. In this work, we are concerned with multicast-routing-specific security. Several differences make the multicast communication model more challenging than its unicast counterpart. Designing secure multicast protocols for wireless networks requires a more complex trust model, as nodes that are members of the multicast group cannot simply organize themselves in a dissemination structure without the help of other nonmember nodes acting as routers. Unlike unicast protocols, which establish and maintain routes between two nodes, multicast protocols usually establish and maintain more complex structures such as trees or mehes. For example, protocols relying on trees require additional operations such as route activation, tree pruning, and tree merging. These actions do not have a counterpart in the unicast case and may expose the routing protocol to new vulnerabilities. Last but not least, multicast protocols deliver data from one sender to multiple receivers, making scalability a major problem when designing attack-resilient protocols. In particular, solutions that offer resiliency against Byzantine attacks for unicast are not scalable in a multicast setting. For example, multipath routing affects significantly the data dissemination efficiency, while strategies based on acknowledgments have high network overhead.

II. SYSTEM ANALYSIS

Multihop wireless networks rely on node cooperation to provide multicast services. The multihop communication offers increased coverage for such services but also makes them more vulnerable to insider (or Byzantine) attacks coming from compromised nodes that behave arbitrarily to disrupt the network.

Existing System:

Wireless networks provide a less robust communication than wired networks due to frequent broken links and a higher error rate. Security is also more challenging in multihop wireless networks

because the open medium is more susceptible to outside attacks and the multihop communication makes services more vulnerable to insider attacks coming from compromised nodes. Although an effective mechanism against outside attacks, authentication is not sufficient to protect against insider attacks because an adversary that compromised a node also gained access to the node's cryptographic keys. Insider attacks are also known as Byzantine attacks and protocols able to provide service in their presence are referred to as Byzantine-resilient protocols.

Proposed System:

We propose BSMR, a secure on-demand multicast protocol for multihop wireless networks that provides resiliency against a representative set of strong Byzantine attacks (black hole, wormhole, and flood rushing). BSMR uses a selective data forwarding mitigation mechanism based on a reliability metric that captures adversarial behavior. Nodes determine the reliability of links by comparing the perceived data rate with the one advertised by the source. Adversarial links are avoided during the route discovery phase. BSMR also prevents attacks that try to prevent or arbitrarily influence route establishment.

III. IMPLEMENTATION

The first step in developing anything is to state the requirements. This applies just as much to leading edge research as to simple programs and to personal programs, as well as to large team efforts. Being vague about your objective only postpones decisions to a later stage where changes are much more costly.

The problem statement should state what is to be done and not how it is to be done. It should be a statement of needs, not a proposal for a solution. A user manual for the desired system is a good problem statement. The requestor should indicate which features are mandatory and which are optional, to avoid overly constraining design decisions. The requestor should avoid describing system internals, as this restricts implementation flexibility. Performance specifications and protocols for interaction with external systems are legitimate requirements. Software engineering standards, such as modular construction, design for testability, and provision for future extensions, are also proper. Many problems statements, from individuals, companies, and government agencies, mixture requirements with design decisions. There may sometimes be a compelling reason to require a particular computer or language; there is rarely justification to specify the use of a particular algorithm. The analyst must separate the true requirements from design and implementation decisions disguised as requirements. The analyst should challenge such pseudo requirements, as they restrict flexibility. There may be politics or organizational reasons for the pseudo requirements, but at least the analyst should recognize that these externally imposed design decisions are not essential features of the problem domain.

A problem statement may have more or less detail. A requirement for a conventional product, such as a payroll program or a billing system, may have considerable detail. A requirement for a research effort in a new area may lack many details, but presumably the research has some objective, which should be clearly stated.

Most problem statements are ambiguous, incomplete, or even inconsistent. Some requirements are just plain wrong. Some requirements, although precisely stated, have unpleasant consequences on the system behavior or impose unreasonable implementation costs. Some requirements seem reasonable at first but do not work out as well as the request or thought. The problem statement is just a starting point for understanding the problem, not an immutable document. The purpose of the subsequent analysis is to fully understand the problem and its implications. There is no reasons to expect that a problem statement prepared without a fully analysis will be correct.

The analyst must work with the requestor to refine the requirements so they represent the requestor's true intent. This involves challenging the requirements and probing for missing information. The psychological, organizational, and political considerations of doing this are beyond the scope of this book, except for the following piece of advice: If you do exactly what the customer asked for, but the result does not meet the customer's real needs, you will probably be blamed anyway.

Modules:

1. Route Discovery
2. Multicast Route Activation
3. Multicast Tree Maintenance
4. Selective Data Forwarding Mitigation

1. Route Discovery

BSMR's route discovery allows a node that wants to join a group to find a route to the multicast tree. The protocol follows the RREQ/RREP procedure used by on-demand routing protocols, with several differences. To prevent outsiders from interfering, all route discovery messages are authenticated using the public key corresponding to the network certificate. Only group-authenticated nodes can initiate RREQs, and the s is required in

each request. Tree nodes use the tree token to prove their tree status.

2. Multicast Route Activation

The requester signs and unicasts on the selected route a multicast activation (MACT) message that includes its identifier, the group identifier, and the sequence number used in the RREQ phase. The MACT message also includes a one-way function applied on the tree token extracted from RREP which will be checked by the tree node that sent the RREP message to verify that the node that activated the route is the same as the initial requester. An intermediate node on the route checks if the signature on MACT is valid and if MACT contains the same sequence number as the one in the original RREQ. The node then adds to its list of tree neighbors the previous node and the next node on the route as downstream and upstream neighbors, respectively, and sends MACT along the forward route. During the propagation of the MACT message, tree neighbors use their public keys to establish pairwise shared keys, which will be used to securely exchange messages between tree neighbors.

3. Multicast Tree Maintenance

Routing messages exchanged by tree neighbors are authenticated using the pairwise keys shared between tree neighbors. If a malicious node prunes itself even if it has a subtree below it, the link repair procedure is initiated by nodes that detect a broken link and is similar to route discovery. The group leader periodically broadcasts in the entire network a signed GroupHello message that contains the current group sequence number, the tree token authenticator, and the hop count anchor. A signed GroupHello message containing a special flag also ensures that when two disconnected trees are merging, one of the group leaders is suppressed.

4. Selective Data Forwarding Mitigation

The source periodically signs and sends in the tree an MRATE message that contains its data transmission rate. As this message propagates in the multicast tree, nodes may add their perceived transmission rate to it. Each tree node keeps a copy of the last heard MRATE packet. The information in the MRATE message allows nodes to detect if tree ancestors perform selective data forwarding attacks. Depending on whether their perceived rate is within acceptable limits of the rate in the MRATE message, nodes alternate between two states. The initial state of a node is disconnected; after it joins the multicast group and becomes aware of its expected receiving data rate, the node switches to the connected state. Upon detecting a selective data forwarding attack, the node switches back to the disconnected state.

Attacks Against Multicast Routing Adversarial Model

Nodes may exhibit Byzantine behavior, either alone or colluding with other nodes. Examples of such behavior include not forwarding packets, injecting, modifying, or replaying packets, rushing packets, and creating wormholes. We refer to any arbitrary action by authenticated nodes resulting in the disruption of the routing service as Byzantine behavior and to such an adversary as a Byzantine adversary. Adversaries do not have control over the physical and MAC layers. We consider a three-level trust model that captures the interactions between nodes in a wireless multicast setting and defines a node's privileges: the first level consists of the source, which must be continually available and assumed not to be compromised (an unavailable or untrusted source makes the multicast service useless); the second level consists of the multicast group member nodes, which are allowed to initiate requests for joining multicast groups; and the third level consists of nonmember nodes, which participate in the routing but cannot initiate group join requests. In order to cope with Byzantine attacks, even group members are not fully trusted.

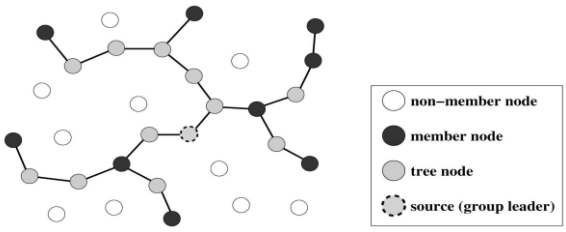


Fig. 1. Types of nodes in a multicast setting for ad hoc wireless networks.

Attacks in Multicast in Multihop Wireless Networks

An adversary can attack the control messages corresponding to the route discovery, route activation, and tree management operations or can attack data messages. The route discovery phase can be disrupted by outside attackers creating undesired results by injecting, replaying, or modifying control packets. Nodes that are not in the tree can mislead other nodes into believing that they found and are connected to the tree. Nodes can flood the network with bogus requests for joining multicast groups. A Byzantine adversary can prevent a route from being established by dropping the request and/or response or can influence the route selection by using wireless-specific attacks such as wormhole and flood rushing. A Byzantine adversary can also modify the packets carrying the route selection metric such as hop count or node identifiers. An attacker can prevent a path from being activated by injecting bogus route activation messages or by dropping correct route activation messages. A node authorized to join a

multicast group can initiate route activation packets to more than one tree node, which may result in unnecessary branches being grafted to the multicast tree. Nodes can maliciously report that other links are broken or generate incorrect pruning messages, resulting in correct nodes being disconnected from the network or tree partitioning. In the absence of authentication, any node can pretend to be the group leader. Although many routing protocols do not describe how to select a new group leader when needed, we note that the leader election protocol can also be influenced by attackers. Attacks against data messages consist of eavesdropping, modifying, replaying, injecting data, or selectively forwarding data after being selected on a route. A special form of packet delivery disruption is a denial-of-service attack, in which the attacker overwhelms the computational, sending or receiving capabilities of a node. In general, data source authentication, integrity, and encryption can solve the first attacks and are usually considered application-specific security. Defending against selective data forwarding and denial of service cannot be done exclusively by using cryptographic mechanisms. Because external attacks can be prevented using the authentication framework described, we focus on the following three Byzantine attacks:

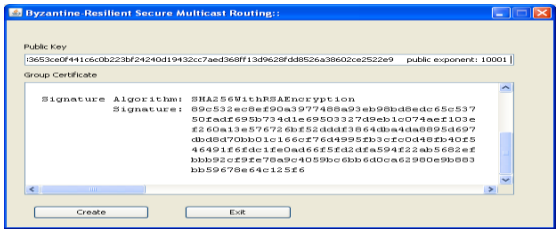
1. Black hole attack. One or several adversaries forward only routing control packets, while dropping all data packets.
2. Wormhole attack. Two colluding adversaries tunnel packets between each other in order to create a shortcut in the network. The adversaries use the low cost appearance of the wormhole to increase the probability of being selected on paths; once selected on a path, they attempt to disrupt data delivery by executing a black hole attack.
3. Flood rushing attack. One or several adversaries rush an authenticated flood through the network before the flood traveling through a legitimate route. This allows the adversaries to control many paths. Flood rushing can be used to increase the effectiveness of a black hole or wormhole attack.

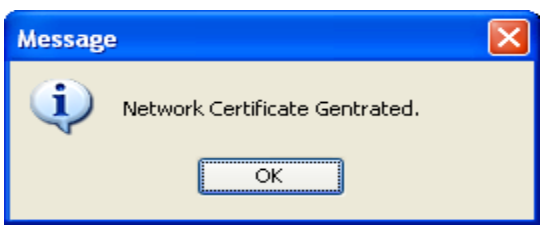
SECURE MULTICAST ROUTING PROTOCOL

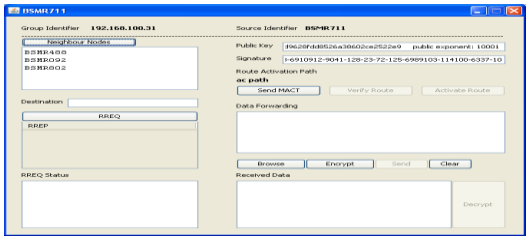
BSMR Overview

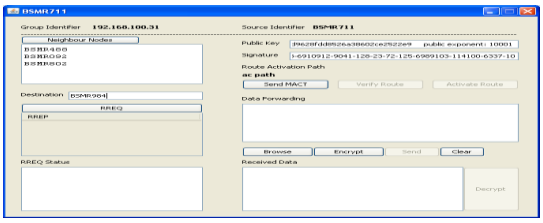
Our protocol, BSMR, ensures that multicast data is delivered from the source to the members of the multicast group, even in the presence of Byzantine attackers, as long as the group members are reachable through non adversarial paths and a no adversarial path exists between a new member and a node in the multicast tree. To achieve this strong guarantee, BSMR builds on the basic operation of the tree-based on-demand protocol presented. To eliminate a large class of outside attacks, we use an authentication framework that ensures that only authorized nodes can perform certain operations (e.g., only tree nodes can perform tree operations, and only nodes that possess valid group certificates can connect to the group multicast tree). For example, only member nodes can send RREQ and route activation messages, and only tree nodes can reply to route activation messages. BSMR mitigates inside attacks that try to prevent a node from establishing a route to the multicast tree by flooding both RREQ and RREP and by using a time-out-based mechanism that ensures that a path is established even if route activation messages are dropped. If an adversarial free route exists, BSMR guarantees that a route is established. BSMR provides resilience to selective data forwarding attacks by using a reliability metric that captures adversarial behavior. The metric consists of a list of link weights in which high weights correspond to low reliability. Each node maintains its own weight list and includes it in each RREQ to ensure that a new route to the tree avoids adversarial links. A link's reliability is determined based on the number of packets successfully delivered on that link. Tree nodes monitor the rate of receiving data packets and compare it with the transmission rate indicated by the source in the form of a multicast rate (MRATE) message. If the perceived transmission rate falls below the rate indicated in the MRATE message by more than a threshold, an honest node that is a direct descendant of an adversarial node updates its weight list by penalizing the link to its parent and then tries to discover a new route to the tree. Only weights corresponding to penalized links are included in RREQs. All no faulty links have a default weight of one. Note that links can also be penalized due to natural losses. We do not differentiate between losses caused by adversarial behavior and natural losses because lossy links should be avoided just as well.

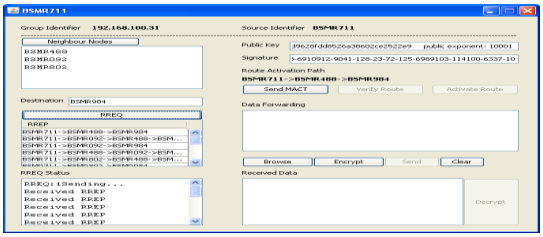
IV. SCREENS

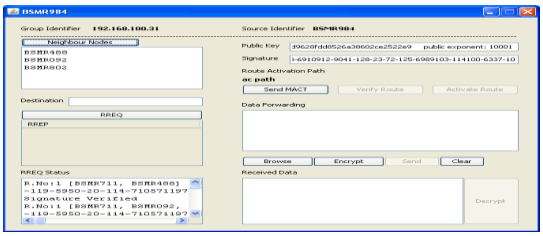


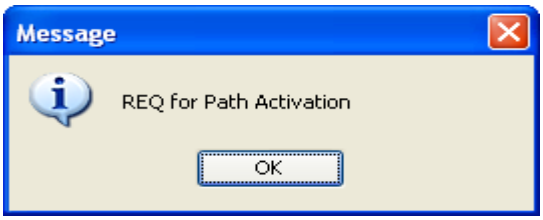


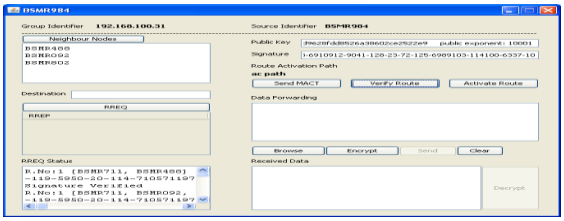


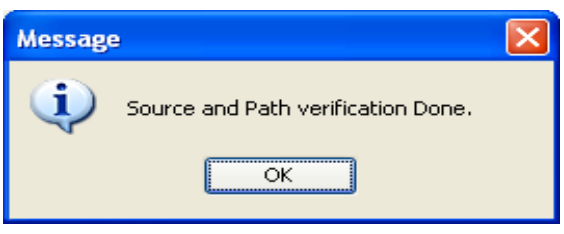


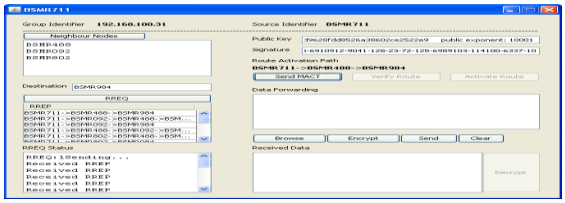


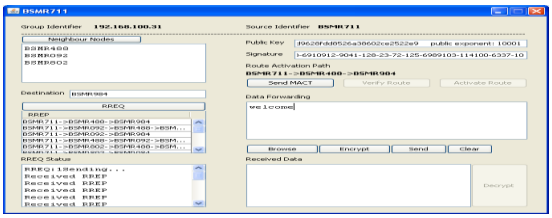


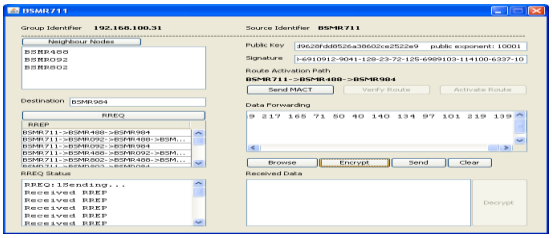


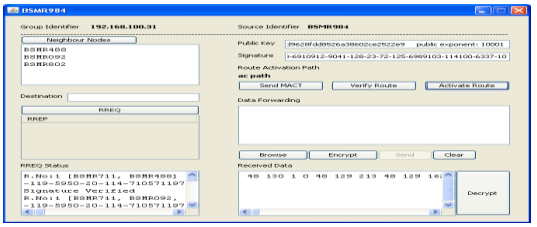


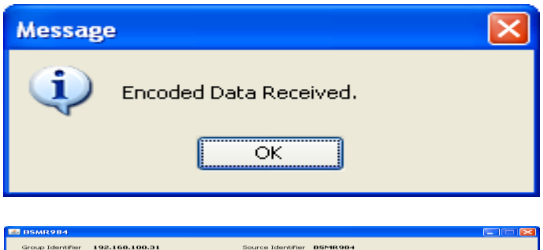


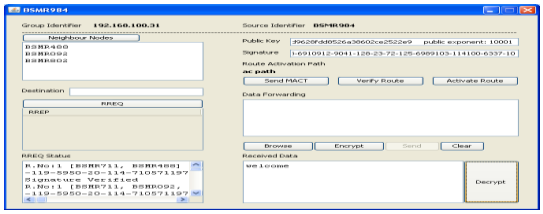












V. CONCLUSION

In this paper, we have discussed several aspects that make designing attack-resilient multicast routing protocols for multihop wireless networks

more challenging than their unicast counterpart we have proposed BSMR, a routing protocol that relies on novel general mechanisms to mitigate Byzantine attacks. BSMR identifies and avoids adversarial links based on a reliability metric associated with each link and capturing adversarial behavior. Our results show that BSMR's strategy is effective against strong insider attacks such as wormholes, black holes, and flood rushing.

REFERENCES

[1] R. Curtmola and C. Nita-Rotaru, "BSMR: Byzantine-Resilient Secure Multicast Routing in Multi-Hop Wireless Networks," Proc.Fourth Ann. IEEE Conf. Sensor, Mesh and Ad Hoc Comm. And Networks (SECON '07), 2007.

[2] Y.B. Ko and N.H. Vaidya, "Flooding-Based Geocasting Protocols for Mobile Ad Hoc Networks," Mobile Networks and Applications,vol. 7, no. 6, 2002.

[3] R. Chandra, V. Ramasubramanian, and K. Birman, "Anonymous Gossip: Improving Multicast Reliability in Mobile Ad-Hoc Networks,"Proc. 21st Int'l Conf. Distributed Computing Systems(ICDCS '01), 2001.

[4] Y.-B. Ko and N.H. Vaidya, "GeoTORA: A Protocol for Geocasting in Mobile Ad Hoc Networks," Proc. Eighth Ann. Int'l Conf. Network Protocols (ICNP '00), p. 240, 2000.

[5] E.L. Madruga and J.J. Garcia-Luna-Aceves, "Scalable Multicasting:The Core-Assisted Mesh Protocol," Mobile Networks and Applications,vol. 6, no. 2, 2001.

[6]. J.G. Jetcheva and D.B. Johnson, "Adaptive Demand-Driven Multicast Routing in Multi-Hop Wireless Ad Hoc Networks," Proc. ACM MobiHoc, 2001.

[7]. S.J. Lee, W. Su, and M. Gerla, "On-Demand Multicast Routing Protocol in Multihop Wireless Mobile Networks," Mobile Networks and Applications, vol. 7, 2002.

[8]. L. Zhou and Z. Haas, "Securing Ad Hoc Networks," IEEE NetworkMagazine, vol. 13, no. 6, 1999.

[9]. P. Papadimitratos and Z. Haas, "Secure Data Transmission in Mobile Ad Hoc Networks," Proc. ACM Workshop Wireless Security (WiSe '03), pp. 41-50, 2003.

[10] B. Awerbuch, D. Holmer, C. Nita-Rotaru, and H. Rubens, "An On-Demand Secure Routing Protocol Resilient to Byzantine Failures," Proc. ACM Workshop Wireless Security (WiSe '02), 2002.

[11] B. Awerbuch, R. Curtmola, D. Holmer, C. Nita-Rotaru, and H. Rubens, "ODSBR: An On-Demand Secure Byzantine Resilient Routing Protocol for Wireless Ad Hoc Networks," ACM Trans. Information and System Security, vol. 10, no. 4, 2007.

Author's Profile

Mr. Bhima Sankaram Alladi, Post Graduated in Computer Science & Technology (**M.Tech**) From **Andhra University**, Visakhapatnam in 2010 and Graduated in Information Technology (B.Tech) form **JNTU**, Hyderabad, 2008. He is working as an Assistant Professor in Department of Computer Science & Engineering in **St.Martin's Engineering College**, Dhulapally , R.R Dist, AP, India. He has 3 years of Teaching Experience. His Research Interests Include Network Security, Cloud Computing & Data Warehousing and Data Mining.

Mr M.Venu Gopal, Post Graduated in Computer Science & Engineering (M.Tech) , Bharath University, Chennai in 2008 and Graduated in Computer Science Engineering (B.Tech) from JNTUH, in 2005. He is working as an Assistant Professor in Department of Computer Science & Engineering in **St.Martin's Engineering College**, R.R Dist, AP, and India. He has 5+ years of Teaching Experience. His Research Interests Include Network Security, Cloud Computing & Data Warehousing and Data Mining.

Mrs. L.Vandana, Post Graduated in Computer Science & Engineering (M.Tech) , JNTUH , Hyderabad in 2012. She is working as an Assistant Professor in Department of Computer Science & Engineering in **Nalla Narashima Reddy Education society's group of Institutions**, R.R Dist, AP, and India. He has 9+ years of Teaching Experience. Her Research Interests Include Network Security, Cloud Computing & Data Warehousing and Data Mining.

Mr. Manyam Sukesh, Post Graduated in Computer Science & Engineering (M.Tech) , JNTUH , Hyderabad in 2012 and Graduated in Computer Science Engineering (B.Tech) from JNTUH, in 2010. He is working as an Assistant Professor in Department of Computer Science & Engineering in **Vaagdevi College Of Engineering**, R.R Dist, AP, and India. He has 1+ years of Teaching Experience. His Research Interests Include Network Security, Cloud Computing & Data Warehousing and Data Mining.