Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 8, August 2013, pg.1 – 6

RESEARCH ARTICLE

Policy Enforcing and Revoking Mechanism on Trusted Ad Hoc Networks

Sujatha J[1], Manoj Challa[2]

[1]Department of Computer Science and Engineering, CMR Institute of Technology, Bangalore, India
[2]Department of Computer Science and Engineering, CMR Institute of Technology, Bangalore India

[1] Sujipatil014@gmail.com; [2] Manojreddi@cmrit.ac.in

Abstract— In ad hoc networking the polices are vulnerable to a wide range of security in network attacks. The design of enforcing and revoking policy mechanisms is a challenging task, especially in comparison to securing the ad hoc network. In this paper, the designed and implemented mechanism to provide a trusted communication for file sharing in ad-hoc network is explained, where the mechanism has been developed with the help of polices where each policy is a combination of set of parameters. Simulation result shows that the proposed mechanism can be used to provide a better trusted communication with improved latency in distributed network compare to centralised network.

Key Terms: - Ad Hoc Networking Distance Vector (AODV); Mobile Networking; Polices; Latency and Network Security

I. INTRODUCTION

The past few decades have witnessed fast development of Mobile ad hoc Networks (MANETs) technologies. However, in distinction to the massive potential and convenience enabled by MANETs, many folks are still reluctant to permit their mobile computing devices to hitch MANET and run MANET applications. one among the most reasons is that the troublesome to ensure trustiness of the MANET applications dead on remote nodes, i.e., the shortage of trustworthy applications, to ensure truthful and secure communication between multiple network nodes, i.e., the shortage of trusty communication and to authenticate network nodes, i.e., the shortage of trusty identity.

The mature development in short-range wireless technologies, increase in number of mobile computing devices and real time applications in MANET becomes attainable. As an example, two potential applications area unit traffic observation in transport networks and peer-to-peer file sharing in ad hoc networks of smart phones. The success of those applications may be a mechanism promising trusted communication and the involving entities should be properly collaborated. To realize this goal, communication policies that govern the interactions between entities ought to be made public and enforced. As associate example, throughout a traffic observation application, the policy can guarantee that an automotive endlessly forwards accident alerts to cars returning behind it. Similarly, during end-to-end applications, the policy can promise that a smart phone can post information given that the created several contributions like publication files or forwarding various queries.

Mechanisms to outline and assess security policies are well studied in ancient distributed system. Whereas these strategies offer decent communicatory power to represent policies for MANETs applications, the challenge is the way to enforce such policies in MANETs. Most of the present policy social control solutions have targeted on Internet-based systems. However, the solutions proposed are not suitable for different types that don't exist in MANETs attributable to the shortage of infrastructure. Moreover, determinative wherever to position a choke purpose in an exceedingly painter is sort of not possible as a result of the methods between nodes modification oftentimes because of quality. Second, existing ways aim to guard the servers from unauthorized shopper accesses. In MANET, this distinction doesn't exist as each node will be a server and a consumer at a similar time, and no entity are often trusty over another.

A potential answer for such a peer-to-peer surroundings is Law-Governed Interaction (LGI). LGI governs the communication between all nodes within the network by enforcing a unified cluster policy on a collection of middle ware controllers. However, the need of LGI is to controllers over a security; however it does not offer means that of creating the trust. Consequently, it will solely be applicable in controlled environments wherever the enforcers and revoking is deployed or non-appointive, like company computer network and web P2P.

This paper presents the planning and implementation of a policy enforcing and revoking mechanism supported a kernel-level trusty execution monitor. Below this mechanism, every MANET application or protocol has its own policy one. All nodes supporting a definite application and implementing its policy kind a sure application network. Since an application could depend upon different applications, our policy enforcing and revoking mechanism creates a trusty multi-tier network. The member nodes in such a network should enforce and revoking the policies related to these applications still. As an example, a peer-to-peer file sharing application could depend upon an on-demand routing protocol. During this case, the mechanism creates a two-tier trusty file sharing network. It initial establishes a trustworthy routing tier, and therefore a trustworthy network for routing, combining of all the nodes that enforcing and revoking the routing policy. On prime of this tier, it then creates a file sharing tier, enforcing the file sharing policy. In our policy enforcing and revoking mechanism, nodes are members of multiple networks at the same time. As an example, allow us to take into account that a vehicle traffic observance application uses a similar routing algorithmic rule with the file sharing application. Nodes within the a fore mentioned file sharing network can even establish a traffic observance network by making, on high of the routing tier, a separate trustworthy tier enforcing the traffic watching policy.

II. EXISTING SYSTEM

Most of the prevailing policy enforcement solutions have targeted solely on the Internet-based systems. Our work leverages previous analysis on trusty computing and distributed policy enforcement. Distribute Policy enforcement. The concept of trustworthy policy enforcement on every network node will retrospect to our earlier add. There in paper, we have a tendency to develop a Satem- primarily based methodology to implement network access management in ad hoc networks.

Unfortunately, these solutions do not seem to be suitable for MANET i.e. for 2 reasons. First, they enforce policies on trusted “choke points” (e.g., firewall or proxy), that do not exist in MANET as a result of the shortage of infrastructure. What is more, crucial wherever to position a choke purpose during a Manet is nearly not possible as a result of the methods between nodes amendment often as a result of quality. Second, existing strategies aim to shield the servers from unauthorized shopper accesses. In MANET, this distinction does n0t exist as each node are often a servers and a shopper at constant time, and no entity are often sure over another. A possible resolution for such a peer-to-peer atmosphere is Law-Governed Interaction (LGI). LGI governs the communication between all nodes within the network by implementing a unified cluster policy on a group of middle ware controllers. However, LGI needs the controllers to be sure; however does not offer suggest that of building the trust. Consequently, in apply, it will solely be applicable in controlled environments wherever the enforcers are often deployed or elective , like company computer network ,and net P2P .McCun advanced another step by developing a shared sure reference monitor across a coalition of nodes exploitation remote attestation. Enforces communication policies at the virtual machine level and needs that every node runs multiple virtual machines (one for every application), which cannot be sensible for a mobile devices. In addition, does not offer enough flexibility to compose applications and policies. When the application depends on others, then there will be conjunction with their policies should be isolated in one virtual machine.

This paper presents the planning and implementation of a policy enforcing mechanism supported a kernel-level sure execution monitor. Below this mechanism, every Manet application or protocol has its own policy one. All nodes supporting a precise application and enforcing its policy type a sure application centrical network. Application could rely on different applications, so that policy enforcing mechanism creates a sure multi-tier network. The member nodes in such a network should enforce the policies related to these applications likewise. For example, a peer-to-peer file sharing application could rely upon on-demand routing protocol. During this case, the mechanism creates a two-tier trustworthy file sharing network. It initial establishes a trustworthy routing tier, and therefore network for routing are trustworthy hence, comprising of all the nodes that are enforced in the routing policy. On prime of this tier, it then enforces a file sharing policy and creates the file sharing tier.

We enforced a model of the policy enforcing mechanism in Linux tested it over an IEEE 802.11-based wireless ad hoc network that's composed of TPM-enabled laptops. We have a tendency to conjointly run NS-2 simulations to judge the performance in MANETs in large scale. The results after experimental is demonstrated at lows prices in application execution and network communication despite high one-time initial price in network institution. The simulation results reveal that nodes will be part of the trusty tiers with high chance even though the underlying MANETs square measure extremely volatile. the general communication overhead over long network ways increases however still remains at low levels: but 100% in networks with rare property loss and regarding 2 hundredth in high quality networks wherever property among nodes is unstable.

III. Proposed System

We leveraged static root of trust to determine trust on the trustworthy agent. In observe, this approach is understood to be prone to variety of attacks because of bugs in implementations of boot loader, BIOS and TPM. These vulnerabilities are also alleviated by the dynamic root of trust feature of latest processors. Another limitation is that Satem solely measures and protects the code that the applying depends on. As realized in, the trustiness of the applying conjointly depends on the dynamic information it uses. Roti provides an answer to the current drawback. Satem solely ensures that a protected service cannot load international organisation trustworthy code from the disk. it\'s unable to tackle attacks, like buffer overflow, that may cause the protected service to run absolute code while not ever-changing its disk image. Satem solely mitigates the matter in 2 aspects. First, Satem could reveal the code that has familiar buffer overflow vulnerabilities by attesting it to the user. Hence, the user will avoid trusting the vulnerable code. Second, within the case of a prospering buffer overflow attack, the assaulter runs her own code on the service stack while not being caught by Satem. However because of the restricted size of the stack, the attacker’s code usually needs to decision different native programs on the service supplier to form the attack purposeful. Satem restricts the attacker’s capability of launching absolute native code (i.e. any code launched by the protected service should be outlined within the commitment).

The tier keys square measure protected in memory. Absolutely addressing this vulnerability could need because arts changes to DRAM to form it lose memory quicker. Satem kernel code is not modularized attributable to the necessity of inserting integrity check points at numerous places within the kernel. This makes the code troublesome to port and modify. We tend to square measure exploring alternative strategies like Linux Security Module for improvement. We tend to attempt to implement the supporter by exhausting writing the policy implementing operate within the application ASCII text file. This is often inflexible since ever-changing the policy could need modifying the applying. Within the future, we tend to commit to implement a standalone supporter because the clear application proxy. During this method, the applying request is redirected to its native supporter that communicates with the applying on the remote node. A technique to realize this is often to ascertain the mapping between the applying and its supporter once the supporter registers with the tier manager.

IV. System Work

In this section, we have an inclination to introduce the node style of our technique. As shown in Fig. 1, it consists of a reliable agent (Satem) a tier manager and type of enforcers, each of that enforces and revokes a tier policy. We have an inclination to then discuss in details the two protocols: be a region of and MERGE, followed by the analysis of their correctness. The service provider uses static or run time analysis to figure out the code base. Commitments for an application usually include several Dozens of code hashes. The system and repair commitments altogether for the P2P file sharing application .The code hashes which generates the commitment certificate as service provided.

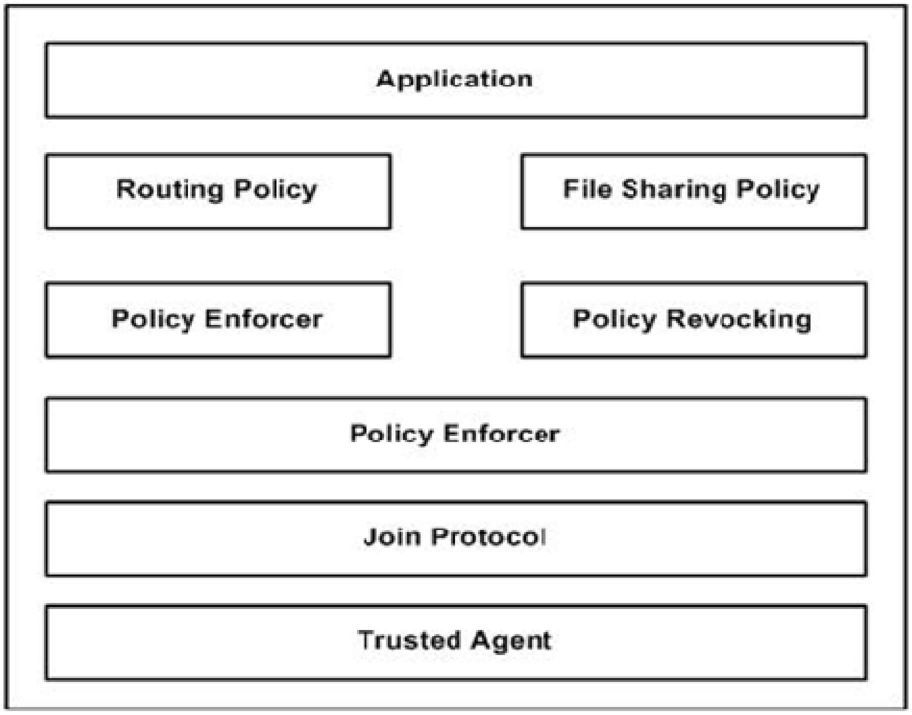
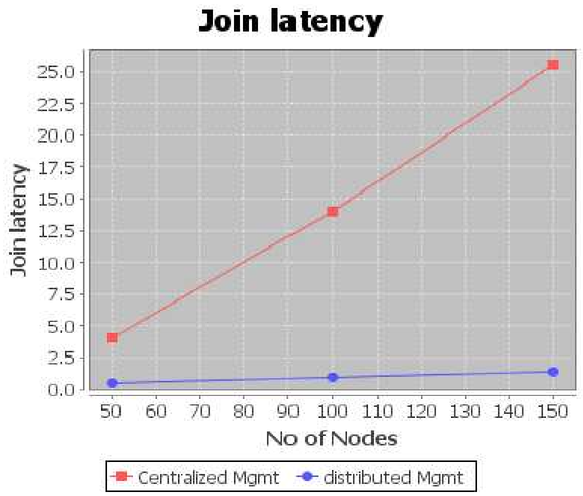


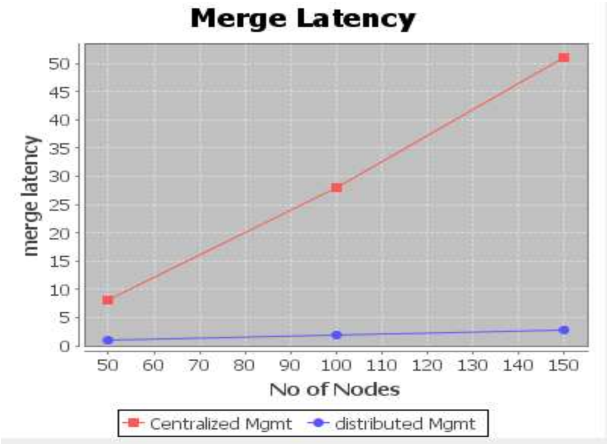
Fig 1 Node architecture for trusted multiple tier

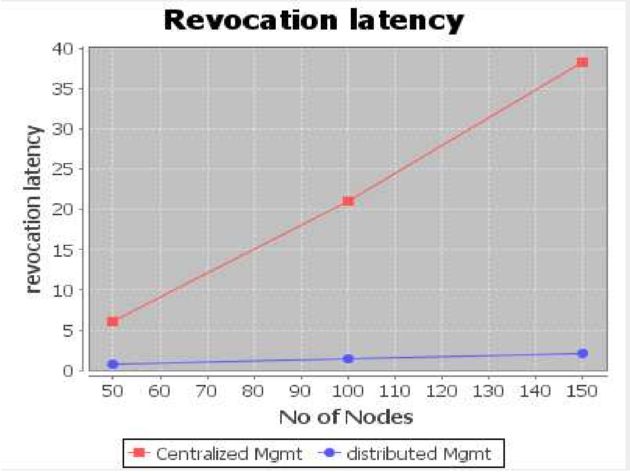
1) Request code certificates. The service supplier requests each merchant to get a vendor-signed code certificate within the same format because the commitment for its code.

2) Sign the commitment. The requester forwards all the code certificates and also the commitment to a third-party trusty Certificate Authority (CA). The CA must verify the signatures of all code certificates and compare the code hashes within the commitment against the certificates. The CA signs the commitment if and providing it verifies all code certificates and code hashes within the commitment. Satem solely guarantees the integrity and therefore the credibility of the code, however not its correctness. The requester should have a local neighbourhood trust policy that governs that kernel and services are trusty. It takes 2 steps to verify whether a service is trusty. First, it authenticates the kernel and repair commitment certificates and learns the identities of the kernel, its modules, and therefore the service second, it verifies the kernel and also the service against the trust policy.

V. Test and Results







VI. CONCLUSION

In this paper conferred a mechanism for MANETs to enforce application communication policies. Below this mechanism, nodes supporting an equivalent set of applications enforcing equivalent policies construct a trusty multi-tier application-centric network. The policy enforcing and revoking is associated with each tier in the network has its application. The appliance of the higher tier depends on the applications of the lower tiers to speak. Solely trustworthy nodes are allowed to hitch the network furthermore; communication between them is regulated by the policies at each tier. To make sure trusty policy social control, we augment every node with a trusty kernel agent supported the TPM.

ACKNOWLEDGEMENT

The authors thank the management of CMR Institute of Technology for support carrying out this work.

REFERENCES

[1] P. Papadimitratos, “Secure and Fault-Tolerant Communication in Mobile Ad Hoc Networks,” PhD Dissertation, Cornell University,January 2005

[2] P. Papadimitratos and Z.J. Haas, “Secure Data Communication in Ad Hoc Networks,” IEEE Journal on Selected Areas in Communication,2nd quarter, 2006 (to appear)

[3] H. Luo, P. Zerfos, J. Kong, S. Lu, and L. Zhang.Self-securing Ad Hoc Wireless Networks. In proc.Seventh International Symposium on Computers and Communications (ISCC'02), July 1-4 2002.

[4] G. Montenegro and C. Castelluccia. Statistically Unique and Cryptographically Veri_able (SUCV) Identi_ers and Addresses. In proc. Network and Distributed System Security Symposium (NDSS'02),February 2002.

[5] G. Montenegro and C. Castelluccia. Crypto-based Identi_ers (CBIDs): Concepts and Applications. ACM Transactions on Information and System Security, 7(1):97{127, 2004.

[6] G. O'Shea and M. Roe. Child-proof authentication for MIPv6 (CAM). ACM SIGCOMM Computer Communication Review, 31(2):4{8, 2001.

[7] Keith I. Farkas, John Heidemann, and Liviu Iftode. Intelligent transportation and pervasive computing. In IEEE Pervasive Computing Magazine, number 4, pages 18–19, 2006.

[8] A. Mayer, A. Wool, and E. Ziskind. Fang: A firewall analysis engine. In Proceed-ings of IEEE Symposium on Security and Privacy (S&P’00), 2000.

[9] F. Stajano. The Resurrecting Duckling –What Next? In The 8th Int. Workshop on Security Protocols, 2000.

[10] G. Xu, C. Borcea, and L. Iftode, “Satem: A Service-aware Attestation Method Toward Trusted Service Transaction,” in the Proceedings of IEEE Symposium on Reliable Distributed Systems (SRDS), October 2006, pp. 321–336.

Authors Bibliography

Ms. Sujatha .J is currently pursuing M.Tech. Degree in Computer Network and Engineering at CMR Institute of Technology. He received the B. Tech. Degree in Information Science and Engineering from RYMEC Institute of Technology, India in 2011.

Mr. Manoj Challa is pursuing Ph.D (CSE) in S.V. University, Tirupati, India. He completed his M.E (CSE) from Hindustan College of Engineering, Tamil Nadu in 2003. He is presently working as Associate Professor, CMR Institute of Technology, Bangalore. He presented nearly 18 papers in national and international conferences. His research areas include Artificial intelligence and computer networks.