

Security Solution for Mobile Ad Hoc Network of Networks (MANoN)

A. H. Al-Bayatti, H. Zedan, and A. Cau

Abstract— Our definition for Mobile Ad Hoc Network of Networks (MANoNs) is a group of large autonomous wireless nodes communicating on a peer-to-peer basis in a heterogeneous environment with no pre-defined infrastructure. In fact, each node by itself is an ad hoc network with its own management. Based on the recommendation ITU-T M.3400 security management consisting of security administration, prevention and detection of malicious nodes and containment and recovery is considered to be one of the major problems MANoN is facing. This article proposes a novel behaviour detection algorithm combined with threshold cryptography digital certificates to satisfy prevention and detection to securely manage our system. This technique will be evaluated using Network Simulator NS-2 to provide and check whether security requirements are met in a comprehensive manner.

Index Terms— Mobile Ad Hoc Network of Networks (MANoN), Mobile Ad Hoc Network (MANET), Behaviour Detectio, Digital Certificates

I. INTRODUCTION

MANoNs have various defining characteristics that differentiate them from other wired, wireless and even other ad hoc networks, because a MANoN is a combination of a Mobile Ad hoc Network (MANET) [1] and a Network of Networks (NoN) [2] [3] [4]. We define MANoNs as a number of nodes interconnected by wireless connections in a dynamic topology that lacks any infrastructure. Basically, each node is an ad hoc network in itself, with its own management and rules. In addition, MANoNs have the capability of operating under partial information, which makes them more flexible yet more configurable (evolvable) over time to networks joining and disconnecting, without affecting the main system. Fig. 1 shows different MANETs from different backgrounds and a resource communicating with each other, creating what is called a MANoN. These unique characteristics will raise nontrivial challenges for MANoNs, such as security, routing, scalability, availability, deployment considerations, media access, Quality of

Software Technology Research, Laboratory, De Montfort University, UK

Service (QoS) [5] [6]. Moreover, conflicts which might occur because of conflicting policies (for example, nodes following their own network policies and at the same time obeying different policies the new MANoN system might enforce) adopted by different entities in the MANoNs.

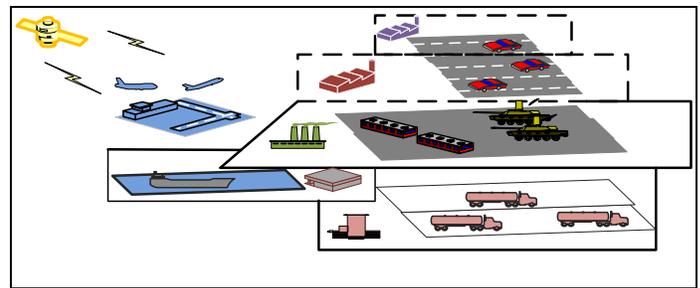


Figure 1. A MANoN Community

As a result, providing the components of a security management (prevention and detection) as defined in ITU-T M.3400 [7] is essential in order to overcome the security threats (ex. Denial of Service (DoS), host impersonation and information disclosure) our MANoN might encounter.

In this paper we propose a novel, efficient, security management framework for our MANoN. We will provide a behavioural detection algorithm combined with threshold cryptography digital certificates to provide prevention and detection to the system. Moreover, a comprehensive, end-to-end security architecture perspective for MANoNs based on the ITU-T recommendations: X.800 and X.805 [8] [9] will be proposed. The remainder of this paper will be organised as follows: Section 2 will describe our scenario, Section 3 will explain the security architecture and its components, Section 4 illustrates and evaluates MANoN implementation using NS-2, Section 5 will describe the simulation results and section 6 will set out the conclusion.

II. MOBILE AD HOC NETWORKS OF NETWORKS SENARIO

To aid the application of MANoN in any wireless environment, when required, and to achieve the services demanded by the user, we need to define MANoN as a whole object with clear syntax and semantics. In this section, we will present a framework scenario that can be applied to MANoN in different network environments, for example cellular systems, smart homes or military operations. Fig. 2

depicts four MANETs; each network is a legacy under its own management and policies coming together to create a MANoN; each MANET has the ability to perform separately which enables it to disconnect and join without affecting the main MANoN system. Networks 1, 2 and 4 are pre-defined and connected to exchange PKI information (Public keys P, Private keys pr), whereas the undefined network is obviously not. Nodes in each MANET are classified into: *General Nodes (GN)* are regular ground nodes which are typically soldiers equipped with communication and computation limited devices, and *Back-Bone Nodes (BBN)* are usually special units, such as tanks and personnel carriers, which have more extensive facilities than regular ground nodes. BBN nodes will carryout CAs (Servers CA_{se} and Combiners CA_c) duty [16].

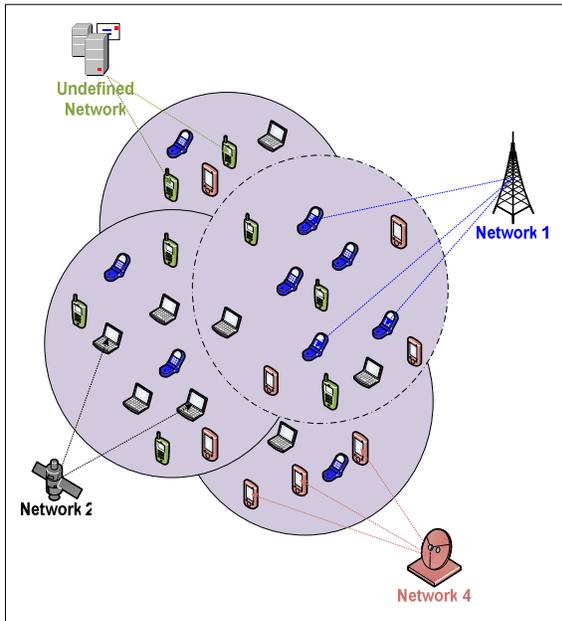


Figure 2. MANoN Scenario

It is relatively uncommon to have one node that belongs to more than one PKI, because this protocol is used either in civilian environments or military environments where the number of PKIs within a given area is limited. Before engaging into the MANoN, nodes in each MANET will receive from their MANET their digital certificates (authentication and authorisation) that are based on the ITU-T Recommendation X.509 [10] with the aim of operating in the MANoN. Authentication certificates will be used as identification (ex. passport), whereas authorisation certificates will be used as security clearance, enabling nodes to operate with distinctive permissions. With the purpose of illustrating security management, special solutions and mechanisms must be applied. Thus, as a foundation to our system, first, we build security architecture to provide an end-to-end security solution based on the ITU-T recommendations: X.800 and X.805. Second, we present behavioural detection mechanism which will provide security management for MANoN nodes, policies and services not to mention achieving the set of security

requirements any system might need to survive: Authentication, Authorisation, Availability, Data integrity, and Non repudiation.

III. SECURITY ARCHITECTURE

As we have learned from the history of security attacks [11], security cannot be considered separately after the whole system of networks has been designed; rather, security must be considered as an inseparable aspect of the development of the network. As a result, this security architecture has been created to address the global security challenges of consumers, users, services and other applications. In order to prevent any type of attack - external or internal, passive or active,- a set of requirements must be identified in our MANoNs.

A. Security Requirements

Security requirements are a set of measures used to address particular aspects of network security, which are governed, by a specific set of security policies. Seven major sets of requirements are identified [12] [13] [14]: *authentication* means that the correct identity is known to the communicating parties; *authorisation* means that only authorised nodes are able to perform in the network; *availability* means that entities, services and resources are available in the face of all kinds of attack; *non-repudiation* means that entities cannot deny the performance of a specific action; *data confidentiality* means that messages or packets are kept secure from any unauthorised entities; *data integrity* means that messages are unaltered during any communication; *privacy* means that packets and messages cannot be followed to disclose the identity or location of entities. After defining our security requirements, we must show how they can protect our system against all major security threats and how they can be applied to every part of a MANoN.

In order to provide a comprehensive, end-to-end security solution for MANoNs, we need to satisfy these security requirements to a hierarchy of network equipment, which is referred to in our architecture as security layers.

B. Security Layers

In order to provide a comprehensive solution, we divide our complex MANoNs logically into separate architectural components. This separation allows a systematic approach to the MANoNs that can be used in the planning of new security solutions for the security threats our system might face as well as for assigning security to existing MANoNs. Moreover, the success of the OSI [15] model applied in designing network protocols is a good example to follow in designing security protocols. A layered architecture can provide advantages such as modularity, simplicity, flexibility and standardisation of protocols. Fig. 3 depicts

four security layers for MANoNs, which are built on one another to provide a network-based solution. The functionality of each layer is explained below.

Trust Infrastructure: The trust infrastructure security layer represents a fundamental building block of the network, consisting of the basic relationships between the nodes. An example is given by the explanation of Zhou and Hass [16] of a well-deployed PKI environment (threshold cryptography), as there is no centralised certification authority in which public and private keys are exchanged between all nodes. The security association established in the trust infrastructure layer must serve the upper layer security mechanisms.

Communication: The communication security layer consists of the transmission facilities protected by the security requirements, as well as the security mechanisms applied to data transmitted between nodes [17].

Routing: The routing security layer consists of basic transports and connectivity as well as the individual nodes; since each node in the ad hoc network acts as host and router, our MANoN is not different from that perspective. Moreover, nodes must exchange information about their neighbours to construct the network topology in order to apply one of the ad hoc routing protocols (Proactive, Reactive and Hybrid) [18].

Application: The application security layer concentrates upon the security of the network-based services and network protocols that perform sub-network access operations from end system to end system which, are applied in our MANoNs [19].

After dividing our security architecture into four layers, we need to handle it from different perspectives, so we consider two distinct security planes, the *Management Plane* and the *End-to-End User Plane*, which are protected by our security requirements from any threats and attacks, as shown in Fig. 3.

Management Plane: The management security plane supports FCAPS (Fault-management, Configuration, Accounting, Performance and Security) [19] [20]. Moreover, it is concerned with the protection of OAM&P (Operation, Administration, Maintenance and Provisioning) [21], functions of the nodes, services and applications.

End-to-End User Plane: The end-to-end user plane deals with end-user data flow (information flow) and security mechanisms related to the end users of the system [22].

These security planes are designed in such a way that events on one security plane are kept totally isolated from the other. At the same time, each layer will be based on each other to provide a flexible foundation to our mechanisms.

Having shown the components of our security architecture, we believe that each part has its own specific security needs, so by dividing it into layers and planes and by applying security requirements, we obtain a highly secure end-to-end security architecture.

IV. IMPLEMENTING SECURITY SOLUTIONS DEFINED IN OUR SECURITY ARCHITECTURE FOR MANoN

Various technologies can be used to satisfy the security requirements defined previously in section 3. Modern cryptography – including public key cryptography, digital signatures and digital certificates – are the most powerful tools that can be used to implement most security requirements, including authentication, authorisation, data confidentiality, data integrity and non-repudiation. The unique characteristics of MANoN make the application of these technologies a real challenge. In this paper, this issue is tackled by proposing new security mechanisms, the access control and behaviour detection mechanisms. The proposed security mechanisms will focus on the proceeds of the trust infrastructure and application layers defined in the previous section. This is because of the importance of this object in representing the main functionalities of MANoN as a wireless access networks. As shown, the security mechanisms will satisfy *authentication*, *authorisation* and help toward forcing other security requirements such as *availability*, *data confidentiality*, *data integrity* and *non-repudiation*.

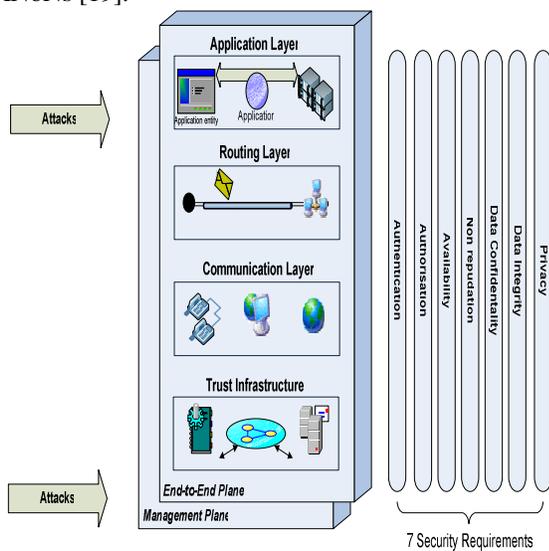


Figure 3. MANoN security architecture

As mentioned, all nodes receive their keys and certificates (Authentication, Authorisation) from their PKI (MANET);

moreover, the MANoN service has its own P/ Pr keys; all BBN (Servers CA_{se}, Combiners CA_c) will receive a share of the pr (sign certificates and perform threshold cryptography [16]) and the P in order for CA_c to validate other MANoN certificates. So, for example, if *node x* from network 1 (Network (1) defined in our MANoN) is trying to engage into our MANoN system, *node x* will broadcast his request for an authorisation certificate (perform in network (2) MANoN) attached with his own authorisation and authentication certificates that he has received from his original network. After receiving the request from *node x*, the CA_c in the correspondent network will validate the certificates by using the service public-key P_{PKI1} to which *node x* belongs. If the certificates are valid, the CA_c tries to find set of (t + 1) correct partial signatures to generate digital signature by the CA_{se} (performing threshold cryptography) in order to create an authorisation certificate, with a specific degree of security clearance, depending on the security clearance (Certificate Policies) *node x* certificate carries. After creating the authorisation certificate the combiner will forward the certificate to the new node with the intention of using it in the correspondent network.

Whereas, based on our assumption, if an undefined node (*node x*) is trying to engage into our MANoN system, a new authorisation certificate (with less security clearance) will be produced by the Local BBN (divided into two types Local BBN and Global BBN). Since, *node x* is undefined in our system, our BBN will not be able to validate its certificates as the network P of network 1 is unavailable; therefore, our BBN will carryout the duty of observing this node.

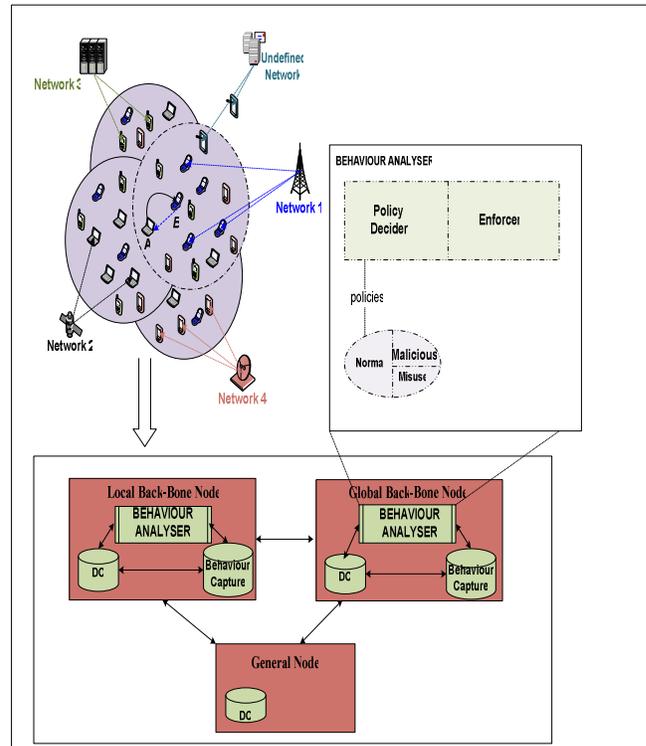
Consequently, when *node x* from network (1) (Network (1) is undefined in our MANoN) is trying to engage and communicate with node y from network (2) (Network (2) defined into our MANoN), first of all, *node x* will broadcast his request for an authorisation certificate (to perform in MANoN) attached with his own authorisation and authentication certificates that he had received from his original network. In the second place, after *node x* receives its authorisation certificate (provisional), our LBBN and GBBN will carryout the responsibility of observing it. This security clearance may evolve or be revoked based on *node x* activity in our MANoN system. As we are dealing with an infrastructure-less MANoN, *node x* will be observed based on his Routing Information Table (RIT); this RIT will show a history of all activities performed by *node x*. before defining the observation process we need to show the architecture components of the behaviour detection algorithm. Fig. 4 illustrates the behaviour detection architecture.

The components of our behaviour detection architecture are:

- **General Node (GN):** Regular ground nodes, e.g. typically soldiers equipped with communication

and computation limited devices (Level 1). Its duty is to collect data and transfer them to the BBN.

- **Local Back-Bone Node (LBBN):** They are usually special units located within the same MANET, e.g. tanks and personnel carriers, which have more extensive facilities than regular ground nodes. LBBN can establish direct wireless links for communication amongst themselves (Level 2). Its responsibility is to collect data and observe nodes entering the MANoN systems.



- **Global Back-Bone Node (GBBN):** They are usually special units from external networks, e.g. tanks and personnel carriers, which have more extensive facilities than regular ground nodes. GBBN can establish direct wireless links for communication amongst themselves (Level 2). Its duty is to collect data and observe nodes entering the MANoN system.
- **Data Collector (DC):** The main buffer of collected data located in both GN and BBN, enables the behaviour analyser to analyse all available data the system has collected. The data collector will be separated from other components to permit the data collector to operate simultaneously by collecting data from different recourses and at the same time enable the behaviour analyser to process the transferred information [23].

- **Behaviour Analyser:** Behaviour of the observed node will be abstracted, so it will check whether the behaviour of the node is malicious (anomaly, misuse) or normal. The behaviour analyser comprises a Policy Decider and Enforcer. The *Policy Decider* contains set of policies which are a set of rules which change dynamically over time and events. The *Enforcer* is a dynamic mechanism that enforces those policies.
- **Behaviour Capture:** The behaviour capture stores the history of behaviours that nodes might have (normal, anomaly or malicious) during a specific period of time; this capture is constantly updated, depending on the observed node actions, despite the fact that saving all behaviours is impossible. Nevertheless, a reasonable number of behaviours must be stored.

A. Syntax Expression

Before demonstrating categories of the behavioural detection architecture and implementation of the behaviour analyser, we need to explain our syntax and the variables of our behaviour detection algorithm:

- P : is a property; a property P is a formula written in underlying logic (e.g. using ITL)
- h_i : is the behaviours i , $i \geq 1$;
- σ_i : is a states i , $i \geq 1$;
- T : is a trace which is a portion of h_i ;
- Var : is a set of interesting variables;
- Val : is a set of semantic values.

State σ_i is considered to be a (in)finite sequence of states, where State σ_i is a function from the set of variables to the set of values.

$$\sigma_i : Var \rightarrow Val$$

Behaviour h_i is a sequence of States σ_i ,

$$h_1 \triangleq \sigma_1 \sigma_2 \sigma_3 \dots$$

Trace T is a portion of Behaviours h_i

We are seeking to prove that a given observation (Trace) will satisfy a Property.

$$T \text{ sat } P \dots (1)$$

To show the satisfaction, we consider (1) as a semantic level.

$[P] \triangleq$ Set of all possible behaviours, each satisfying the Property P

B. Behaviour Analyser

As depicted in Fig. 4, each BBN will embrace a behaviour analyser. The structure of the behaviour analyser consisted of the policy decider and the enforcer. The policy decider consists of a set of policies (properties); consequently, at the beginning we define all policies as normal policies (indicating good behaviours), and during the process of the system those policies will be dynamically changed into malicious policies (indicating anomaly, misuse behaviours) based on specific actions and events the system might go through. Moreover, normal policies will not stay good, as is mandatory, as those policies are dynamic; a different situation might change them into malicious. Meanwhile, the enforcer is a mechanism that enforces those policies. The mechanism of enforcing those policies is dynamically changed based on specific actions and events. (For example, when security alerts in airports are elevated in any country, different procedures implemented, according to the seriousness of the situation). After defining the structure of our behaviour analyser we will use the policy categories as a comparison model to check the activity of the new nodes, whether those activities are normal or not. Hence, the question now is: how does our behaviour detection operate? When *node x* (one of the GNs) tries to operate in our system, other GNs and BBNs will audit the data from the RIT of *node x*. Basically, these audit data will create the state of *node x*; in other words, our audit data is originally the function from the set of variables to the set of values. As a result, a new state of *node x* will be created when ever there are new data in the RIT. Fig. 5 shows how new states are created using RIT data.

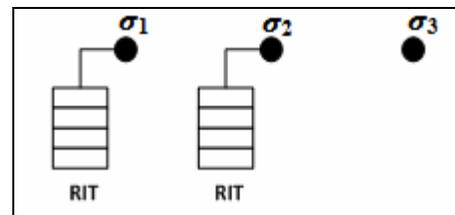


Figure 5. Creation of States

The sequence of these states will create the behaviour, and since there is an infinite number of behaviours, it is impossible to observe all type of behaviours; therefore, if our observers try to make an observation on *node x* activity, they will slice a trace of *node x* behaviour and try to satisfy it with our properties to find if *node x* activity is legitimate or not, and which policy category it belongs to.

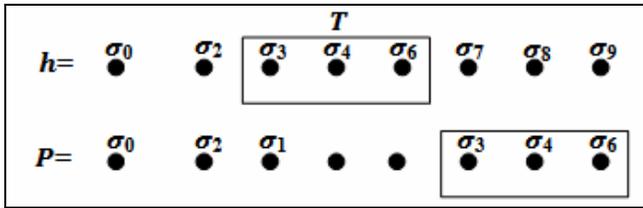


Figure 6. Trace Satisfy a Property

As can be seen in Fig. 6, h represents the behaviour of

node x while T is a slice of node x behaviour so that a comparison with our P can be made to find whether or not this behaviour is found identical to our policy category; if not, our BBN will try to analyse the manoeuvre of node x behaviour in a specific duration of time by collecting more data, and will eventually show whether the behaviour is going towards a diversion or normal act. For example, if node x was found to be attempting to enter a specific area, he is not allowed to enter or endeavour to access a specific log file for which he does not have the authorisation to do; this behaviour will be considered as malicious or misuse contingent on the auditing data.

After a specific period of time and based upon the BBN observation, a decision will be taken whether to upgrade or revoke node certificates.

C. Coping with Misbehaviour Users

In the behaviour detection algorithm, it is easier for malicious nodes to make other nodes accept false certificates. This is because some certificates are issued by undefined CAs, networks or even certificates signed by the nodes themselves. In these cases, and as has been shown, our BBN will carryout the observation task to identify the malicious from the normal nodes; the following, are some malicious examples which have been recognised by our behaviour detection algorithm:

- Forging Sequence Number (SN)

This is a single attack type that can be launched from an insider node on AODV routing protocol. The SN refers the freshness of route to the associated node [25]. Whenever the SN is forged by a high number from an attacker, the route will be changed towards the higher SN route. For example, when employing AODV routing protocol in our MANoN scenario (see Fig. 7), if source A tried to connect destination D, the normal route

will be {A,B,C,D} but if M sends RREP m_2 to B with SN.Des equal to 100 (>50 normal replay), it will take precedence over c_2 ; with the same method to B, M can control the route between A and D (creating man in the middle attack). This attack can be detected and dealt with by our BBN. According to our BBN the forwarding table in BBN SN.Dst=50. If BBN detects any packet having a SN that is larger than it should be and that packet is not sent by the owner of SN (IP address is not equal to the source), then the BBN will treat it as an attack.

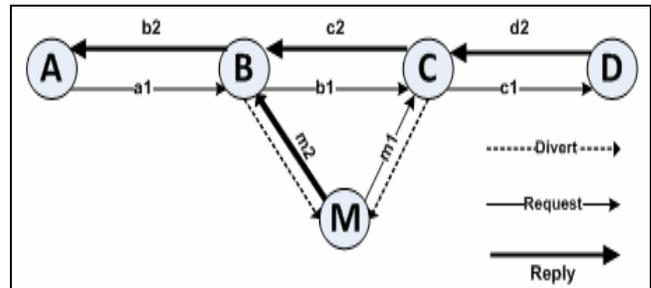


Figure 7. Man in the Middle Attack

- Wormhole Attack (Tunnelling)

This is a cooperating attack done by two malicious nodes; an attacker receives packets at one location in the network and tunnels them to another location in the network, where packets resent into the network (creating a traffic route through them) [26]. As shown in Fig. 8, if A wanted to connect E the shortest path will be {A,B,C,D,E}; instead, X will pretend to have a direct connection to Y, creating a false short path {A,X,Y,E} which will enable A from choosing the wrong path which is actually {A,X,B,C,D,Y,E}, preventing A to choose the really short path {A,B,C,D,E}. Based on our behaviour detection, the solution is simple when ever our BBN checks the RIT, it will find that the RREP is not from X; therefore, our detection algorithm will detect that the route between X and Y is actually fake.

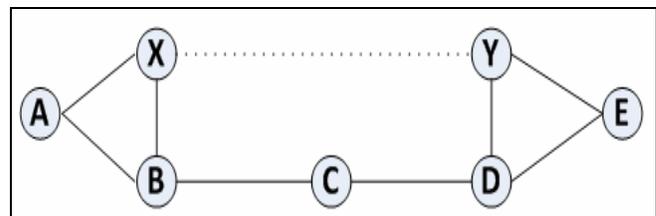


Figure 8. Tunnelling Attack

V. SIMULATION RESULTS

This section will show the results of providing Authentication and Authorisation certificates to the nodes of our MANoN system. NS-2 simulations have been carried out to evaluate the performance of the proposed scheme in the pre-defined scenario. The parameters used for simulation are depicted in Table 1.

Table 1. Ns-2 Simulation Parameters

Total no. of nodes	10, 20, 40, 60
No. of BBN (CA)	4, 10
Network area	1000 m * 1000 m
Total simulation time	500 s
Type of Routing	AODV
Radio range	250 m
Max node speed	1, 10, 15, 20, 25, 30
Pause Time	0, 10, 40, 60, 100
Antenna model	Omni Antenna

It can be argued, that the success ratio is one of the most significant factors that measures the number of successful certificate authentication and authorisation requests to the total number of certificate authentication requests that take place during the simulation time. As an assumption, each node will make at least one authentication request. Therefore, the total number of authentication requests made during the simulation time is equal to the number of nodes trying to enter the MANET.

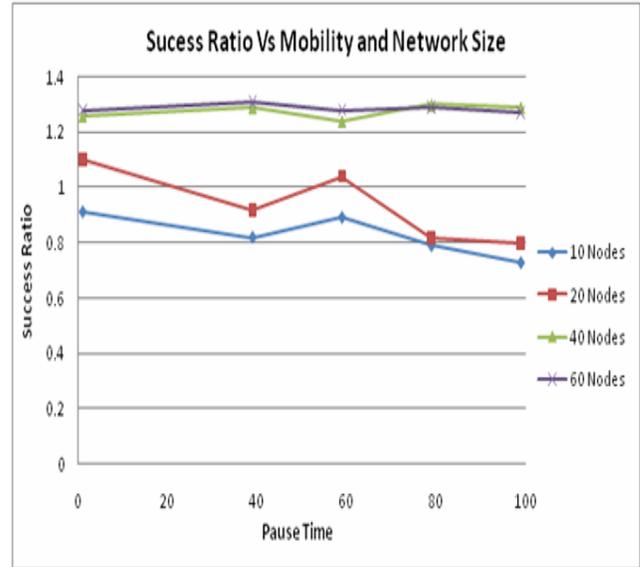


Figure 9. Success Ratio versus Mobility and Network size

Fig. 9 shows the success ratio against mobility and network size. Mobility is most often a big issue in developing ad hoc protocols. As can be seen, our MANoN is not much affected by mobility. In general, the success ratio increases with high mobility situations and large network sizes. The effect of mobility is more noticeable with a small number of nodes. This is because of the number of neighbour nodes. The number of neighbour nodes based on transmission range and simulation area can be calculated using the following formula:

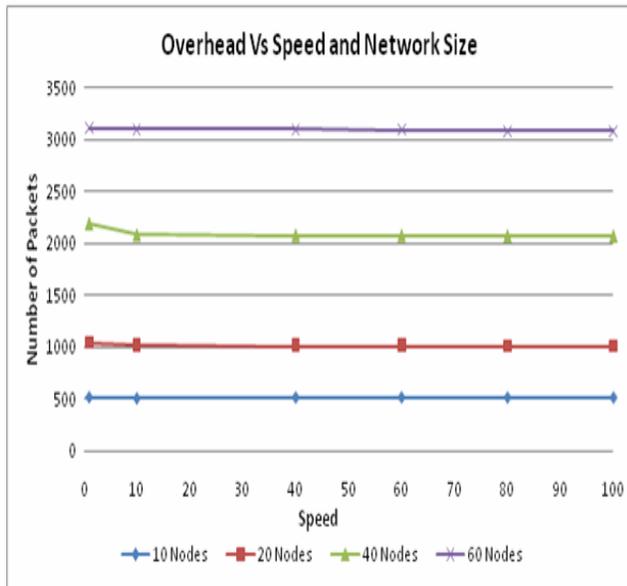
$$\frac{(\pi \times r^2)}{\left[\frac{W \times h}{n}\right]}$$

Where w = area width, h = area height, r = transmission range, n = number of nodes. For example, when the network size is 10, the number of neighbours is around 1.96, but when the network size is 30 the number of neighbours is more than 6. Therefore, the effect of mobility increases with a smaller number of nodes, because high mobility reduces the effect of fewer neighbourhoods.

Similarly to success ratio, overhead is considered essential to any system. Overhead is the number of packets generated by this security protocol. There are three types of packets in our MANoN algorithm: certificate packets, request packets and reply packets.

For a MANoN with N nodes, the total number of generated packets is equal to the number of certificate packets, the number of request packets (Max(N)) and the number of reply packets (Max(N)). This explains why the overhead is almost unchanged for the same number of nodes. The overhead has been calculated against speed and network size. As the node speed increases, it can be observed that the overhead remains almost unchanged; nevertheless, overhead

slightly decreases when the speed increases. Fig. 10 illustrates the increase in overhead caused by increasing network size.



VI. CONCLUSION

In this paper, we have dealt with a specific type of networks called mobile ad hoc network of networks (MANoN). We have focused on designing a comprehensive end-to-end security architecture based on the ITU-T recommendation X.800 and X.805 as foundation to our system. Furthermore, this article provides a novel security mechanism that provides prevention based on digital certificates evaluated using NS-2 and detection based upon node behaviours. We expect that, this security solution could be used to satisfy security for any wireless systems that have the same properties as our system.

REFERENCES

- [1] C.-K Toh, *Ad Hoc Mobile Wireless Networks: Protocols and Systems*, Prentice Hall, New Jersey, pp: 34-37, 2007.
- [2] L. Stotts, S. Seidel, T. Krout, and P. Kolodzy. *MANET Gateways: Radio Interoperability via the Internet, Not the Radio*. *IEEE Communications Magazine*, June 2008, Vol 46. No. 6, PP.51- 60.
- [3] Lt Col R J B Spencer, and J. Ironside. *Network Centric Warfare Operation in an Expeditionary Context*. Military Information & communications, Symposium of South Africa (MICSSA), July 2007.
- [4] A. Cau. *Trust Management System for Network of Networks (NoN)*. <http://www.cse.dmu.ac.uk/STRL/research/nato/index.html>, visited June 28, 2009.
- [5] C. Siva Ram Murthy, and B.S. Manoj. *Ad Hoc Wireless Networks: Architectures and Protocols*. Prentice Hall communications engineering and emerging technologies series Upper Saddle River, New Jersey, 2004.
- [6] Mohammed Ilyas. *The Handbook of Ad Hoc Wireless Networks*, CRC Press, New York, 2003.
- [7] Recommendation M.3400 (02/2000), Telecommunication management network. Technical report, International

- Telecommunication Union- Telecommunication Standardisation sector (ITU-T), (02/2000)
- [8] ITU-T Recommendation X.800 (1991), Security architecture for Open Systems Interconnection for CCITT applications, 1991.
- [9] ITU-T Recommendation X.805 (2003), Security architecture for Systems providing end-to-end communications, 2003.
- [10] ITU-T Recommendation X.509, Public-key and attribute certificate framework. August 2005.
- [11] P. Chandra. *Bullet Proof Wireless Security GSM, UMTS, 802.11 and Ad Hoc Security*, Elsevier, Oxford, 2005.
- [12] J. Al-Jaroodi. *Security Issues In Wireless Mobile Ad Hoc Networks (MANET)*. Technical Report TR02-10-07, University of Nebraska-Lincoln, 2002.
- [13] W. Stallings. *Cryptography and Network Security: Principles and Practices*. 3rd Edition, Prentice Hall 2003.
- [14] K. Fokine. *Key Management in Ad Hoc Networks*. Master Thesis, Linkping University, 2002. <http://www.liu.se/>.
- [15] E. Carrieri, C.A. Rocchini, A. Fioretti, and A.J. Haylett. An OSI compatible architecture for integrated multichannel metropolitan and regional networks. *Integrating Research, Industry and Education in Energy and Communication Engineering*, MELECON '89. Mediterranean, 11-13 April, 1989, pp. 639 – 643.
- [16] L. Zhou, and Z. J. Haas. *Securing ad hoc networks*. *IEEE Magazine*, 1999, pp. 24-30.
- [17] ITU-T Recommendation M.3010, “Principles for a Telecommunications management network”, February 2000.
- [18] C.E Perkins., E.M. Royer. *Ad-hoc on-demand distance vector routing Mobile Computing Systems and Applications*. *Proceedings. WMCSA'99*. Second IEEE Workshop on, 25-26 Feb. 1999, pp: 90 – 100.
- [19] ITU-T Recommendation M.3400, TMN Management Functions. February 2000.
- [20] R. Boutaba, and A. Polyakis. *Projecting FCAPS to Active Networks*. *Enterprise Networking, Applications and Services Conference Proceedings*, 2001, pp. 97 – 104.
- [21] S. Hayes. *A standard for the OAM&P of PCS systems*. *personal Communications*, 1994, pp.24.
- [22] ITU-T Recommendation X.701, Information technology – open Systems Interconnections – Systems Management Overview, August, 1997.
- [23] R. Puttini, M. Percher, L. Me, r. de Sousa. *A fully distributed IDS for MANET*. *Computers and Communications*, 2004. *Proceedings. ISCC 2004*. Ninth International Symposium. Publication Date: 28 June-1 July 2004 Volume: 1, On page(s): 331- 338.
- [24] K. Kim, S. Keun. *Cognitive Ad-hoc Networks under a Cellular Network with an Interference Temperature Limit*. *Advanced Communication Technology*, 2008. *ICACT 2008*. 10th International Conference on Volume 2, 17-20 Feb. 2008 Page(s):879 – 882.
- [25] C. E. Perkins and E. M. Royer, “Ad Hoc On-Demand Distance Vector Routing”, *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications 1999*, pp. 90-100. February 1999.
- [26] Y. Hu, A. Perring, and D. B. Johnson, “Packet Lashes: A Defence Against Wormhole Attacks in Wireless Ad Hoc Networks,” *Proceedings of IEEE INFOCOM 2003*, vol. 3, pp. 1976-1986, April 2003.

A. H. Al-Bayatti received his Ph.D from the school of technology at De Montfort University, Leicester – United Kingdom, in 2009. He obtained his B.Sc in Computer Engineering and Information Technology at the Technology University, Baghdad – Iraq, in 2006. His research interests include Security Wireless Networks, Ad hoc Networks, Sensor Networks, Mesh Networks, Trust, Grid Networks and Security management in wireless networks. He is currently a member of the IEEE, ICNS Networking and Services committee.

H. Zedan is a Professor of Software Engineering at De Montfort University (UK) and Director of the Software Technology Research Laboratory, with over 20 years experience as an academic and practitioner in Computer Science and IT industry. He is also the Technical Director of the University Technology Centre (UTC) in Software Evolution, which is funded by Software Migration Ltd. This is in addition to being on the executive board of the Institute for Creative Technology (IOCT), an interdisciplinary centre involving Computing, Art and Design and Humanities. Hussein's work on the formal specification, verification, validation and analysis of computing and Information Systems, in particular those used in (time-, safety, mission-, financial-) critical applications, has gained him an excellent national and international reputation in the field. He has published 4 books and over 100 technical papers and articles in highly reputable journals and international conferences.

A. Cau gained his MSc in Computer Science from Eindhoven University of Technology (The Netherlands). He then joined Christian Albrechts University of Kiel (Germany) as a junior lecturer where he was awarded his PhD under the supervision of Prof. W.-P. de Roever. He subsequently worked as a Research Associate at De Montfort University on the EPSRC project 'A Compositional Approach to the Specification of Systems using ITL and Tempura'. He is currently a University Senior Research Fellow at the Software Technology Research Laboratory, De Montfort University. Dr. Cau's main interests are compositionality theory, especially the assumption/commitment style of reasoning, refinement, both the algebraic style of refinement as well as refinement mappings, temporal logics such as interval temporal logic and dense time temporal logic, and theorem provers and model checkers.