Secure Access Control in Distributed Environment

Peter Vilhan *
Institute of Applied Informatics
Faculty of Informatics and Information Technologies
Slovak University of Technology in Bratislava
Ilkovičova 3, 842 16 Bratislava, Slovakia
vilhan@fiit.stuba.sk

Abstract

This paper presents the designed concept to improve the public key infrastructure deployability in the mobile ad-hoc networks routed by B.A.T.M.A.N. Advanced. We have extended the B.A.T.M.A.N. Advanced routing protocol with authentication and authorization of routing updates based on X.509 certificates. Furthermore we have determined several levels of node's trustworthiness and two levels of interoperability between trusted authorities in the network. To mitigate extra load caused by renewing of certificates, we have identified critical factors affecting it and designed the computation formula for optimal amount of cross certificates issued by trusted authority. To further improve the service reachablity in highly mobile networks in earlier stages of PKI deplaoyment, we have designed the Cluster Glue. The Cluster Glue helps to connect groups of nodes from different parts of network which owns the certificates issued by the same authority. Thanks to these modifications we are able to mitigate various security risks and provide the more secure route for messages transmitting through the network. Preliminary results were verified by simulations.

Categories and Subject Descriptors

C.2.1 [**Computer-communication Networks**]: Distributed Networks; C.2.2 [**Computer-communication Networks**]: Routing Protocols; C.2.3 [**Computer-communication Networks**]: Public Networks; C.2.5 [**Computer-communication Networks**]: Access schemes; D.4.6 [**Design studies**]: Access Control; E.3 [**Data encryption**]: Public Key Cryptosystems

*Recommended by thesis supervisor: Associate Professor Ladislav Hudec. Defended at Faculty of Informatics and Information Technologies, Slovak University of Technology in Bratislava on June 17, 2014.
Work described in this paper was presented at the UKSim-AMSS seventh European modelling symposium on computer modelling and simulation and UKSim-AMSS 16th international conference on computer modelling and simulation.

Vilhan, P. Secure Access Control in Distributed Environment. Information Sciences and Technologies Bulletin of the ACM Slovakia, Vol. 7, No. 1 (2015) 1-8

Keywords

public key infrastructure, MANET, PKI, RSA, BATMAN, ad-hoc, routing, security, ClusterGLUE

1. Introduction

The MANET network is a special kind of mobile ad-hoc network, which does not rely on any fixed infrastructure. One of the main advantages is the ability to form the network in the purely ad-hoc manner without any costs spent on the infrastructure components, like access points, backbone network and others. These networks are often used at various conferences, meetings, where group of people needs to exchange data or share the connection to the Internet. In this paper we introduce our concept of Public Key Infrastructure also known as PKI designed for this kind of network. PKI consists of trusted third party - certificate or attribute authority - and clients which rely on and trust to certificates signed by this authority. The common way of how certificate authority works is binding public key to legal identity of its owner. This way certificate authority confirms the identity of network entity. If we trust this certificate authority we can safely communicate with any other node which owns certificate issued by this authority. The concept of trusted authority is based on security of its private key. This requirement can be fulfilled in infrastructure networks where certificate authorities need to fulfill strong security criteria. Despite on this from time to time we can read about incidents leading to revocation of certificates. On the other hand MANET networks are completely unmanaged and the security of node is merely based on security knowledge of its owner. One of the way how MANET tries to mitigate attacks is the network homogeneity. Every node in the network should provide the same level of functionality like routing or provide the same set of the services. This way an attack should not be targeted on the specific service or node. But the question is how can we safely distribute functionality of the certificate authority to nodes in the network without compromising the security of its private key? Several approaches have been introduced during the last years which tried to solve this problem in following ways:

1.1 Partially Distributed Certificate Authorities

Partially distributed certificate authorities introduced by Zhou and Haas [3] and Yi and Kravets [9], distributed the functionality of certificate authority to several nodes in the network. Each of these nodes generates just part of the certificate. The main drawback of this concept was the presence of special node called merger, required for the construction of the certificate this introduces the single point of failure.

1.2 Fully Distributed Certificate Authorities

Luo [5] introduced fully distributed certificate authorities where each node shares the part of private key of certificates authority private key and at least t nodes were required to provide the functionality of certificate authority. The security of this concept depends on t value. Lower t value means better service reachability but higher chance to compromise the certificate authority. On the contrary if the value of t was set too high and the node which requests service from the certificate authority does not have at least t neighbors, certificate could not be generated and the service was rendered as unreachable.

1.3 Certificate Chaining

The certificate chaining-based approach by Capkun et al. [2] was built on the chain of trust. Moreover various people have various level of security knowledge. Therefore the chain was as strong as its weakest point.

1.4 Mobility Based Certificate Authorities

There were several other approaches as Mobility based by Capkun et al. [1], benefiting from the fact that node canmove close to the certificate authority and optionally utilize secondary communication channel. The requirement to move into direct communication area of authority was far from practical and hardly usable in real life scenario.

1.5 Cluster Based Certificate Authorities

Cluster based approach by Hahn et al. [6] divided the network into clusters and elects the only node in the cluster responsible for the inter-cluster communication. This solution suffered from the heterogeneity it brings. An attack can be targeted against the node responsible for the inter- cluster communication. This can have various impacts and can result into denial of service or man in the middle attack.

1.6 Identity Based Certificate Authorities

Xia, Wu and Chen introduced Identity based [4] authority utilizing the public key identity optimised for the Optimised Link State Routing Protocol. This solution was acceptable only for the partially independent networks and requires the existing PKI.

Various other approaches like grid based or virtual authority based by Shukat and Holohan [6] were introduced but none of them provides us with the successful solution usable in regular conditions.

The majority of introduced solutions suffer from the nonexistent PKI infrastructure in the moment of deployment or from the absence of routing protocol which could be used for the safe communication during the process of PKI deployment. Routing is a critical part of the network, especially in MANET, where the various attacks like Man-In-The-Middle, BlackHole attack or Sybil attack can be effectively performed. There are several modifications of well-known MANET routing protocols like Optimized Link State Routing (OLSR) and Ad-hoc On Demand Vector (AODV) utilizing PKI. However at most of the cases the formation of certificate authority and distribution of keying material were required prior the network can operate. On the other side distribution of keying material could not be completed without functional routing protocol. This is also known as chicken&egg phenomenon.

2. Proposed solution

2.1 Critical Parts of PKI Security

After the completion of analysis we were able to identify critical parts of PKI security concept:

- *Preserve the maximum possible level of homogeneity*
  Homogeneity in the network is required to harden the targeting of attacks.
- *Grant permissions with higher level of granularity*
  Design the way of how to grant permissions to entities with higher level of granularity. The majority of introduced solutions provide the nodes with "all or nothing" level of permissions while accessing network resources. This way nodes joining the network were either unable to establish connections and access services or got verified status prematurely which could cause security problems.
- *PKI architecture with respect to weaker MANET security*
  Design the PKI architecture with respect to weaker MANET security. In the MANET we cannot guarantee overall and consistent security level over the network. Each node is owned by different user with different security knowledge. Furthermore the nodes are mobile so can be lost, stolen or compromised in a shorter time. Being able to quickly detect anomalies in the node's behaviour is therefore very important. This can be done with the help of distributed intrusion detection system (DIDS). To further mitigate security problems and chicken&egg phenomenon the routing protocols should be PKI aware.

2.2 Introduction to Proposed Concept

The proposed concept consists of following parts:

2.2.1 Attribute certificates

Due to significantly shorter time between security incidents in MANET, we have to use certificates with shorter validity period. Furthermore to be able to grant permissions to network resources with higher level of granularity, we have opted for the use of attributes certificates. Attributes certificates are issued by attributed authority and have a shorter validity period than general certificates issued by certificate authorities. In case of really short validity period we do not need to establish nor manage the certificate revocation list (CRL). The certificates will time-out sooner, than CRL could be fully distributed through the network.

2.2.2 Identities

Node can acquire several identities. Various attribute authorities can issue certificates for the same node. This way node can limit the negative effect when the issuer of its certificate was compromised and certificates have to be revoked. The main idea behind this concept is to let the node obtain its digital identity and then build its reputation upon it.

2.2.3 Gradual privilege escalation

As a solution to "all or nothing" problem we have created following predefined levels of permissions which node have to gain before it becomes the fully integrated part of the network. We can think about this as about kind of "accession talks":

L1 - Endpoint node, this is the basic permission level and digital identity that node gets with the new certificate. With L1 permission node cannot participate on routing processes inside the network but can still access network services, defined by network security policy.

L2 - If there is sufficient communication history between the node and the hosts providing services in the network and its identity is not listed on the Distributed IDS(further reffered as DIDS) blacklist, node can ask the issuer to elevate permissions of its certificate. With L2 permissions level the node participates on the network routing, mediates certificate, provides routing for the incoming nodes and runs various services like distributed certificate storage and DIDS. Distributed storage is used to load-balance the load caused by storing of certificates on network nodes. Both of these services are implemented via distributed hash tables.

L3 - The highest level of permissions, node transforms itself into attribute authority and cross certificates between both authorities are created.

2.2.4 Level of trust between authorities

We have determined two levels of trust between authorities:

L1 - Authority's cross-sign the certificate of each other. This way nodes owning certificate issued by one of these authorities can verify the certificate of each other.

L2 - Distributed certificate store and DIDS knowledge base of both authorities are merged.

This way each attribute authority can create a group of nodes owning certificate issued by this authority and the list of trusted authorities in the network. The number of authorities inside the network is not limited. Each node made its own decision to provide or not to provide trusted authority services which depends on the amount of its free resources. Every node can transfer itself into attribute authority and issue certificates.

However the usefulness of these certificates will depend on the amount of cross certificates between the issuer and another attributes authorities. A lot of the cross certificates means higher chance to verify certificates issued by another authority. Similarly the node can own as many certificates as it wish but it must participate on various services like distributed certificate storage or DIDS resulting from its privilege level. Owning of more certificates with higher privilege level, means lower effect perceived on authority breakdown.

Thanks to the cross certification between authorities the maximum length of the certificate chain to verify is extremely short.

2.2.5 PKI aware routing protocol

We have analysed various protocols and have decided to use the B.A.T.M.A.N. Advanced, thanks to its easy setup, open-code nature and ability to provide connection to the network even to nodes, which do not participate on routing processes inside the network.

2.2.6 Optimal amount of cross certificates issued by authority

As we have stated before, cross certificates are required in case of communication should be established between nodes, which own certificates issued by different authorities. In the extreme case the number of cross certificates per node can reach values like *n.(n-1)*, where n is the number of nodes in the network. This amount of cross-certificates guarantees that node will be able to verify certificate issued by any other node in the network. But imagine that we have to regenerate our private key. Doing this it will cause the need to recreate all of our cross-certificates. To be able to mitigate the effect of recertification we have identified the critical factors affecting it, designed the computation formula for optimal amount of issued cross certificates and verified it by simulation.

We have identified the following critical decision factors:

On the side of local authority:

- The amount of received currently valid cross-certificates - this indicates the level of usage of authority distributed certificate storage
- The amount of issued currently valid certificates with permission level of L2 - this indicates the capacity of authority distributed certificate storage
- The amount of issued currently valid certificates with permission level of L1 or L2 - this indicates the theoretical upper level of capacity or growth potential of distributed certificate storage
- Is the remote authority on the list of TOP3 most occurred authorities, which identity cannot be verified?

On the side of remote authority:

- The amount of issued currently valid certificates - indicates the number of nodes served by remote authority.

We can proceed with the cross-certification providing, that computed value of penalization between local and remote authority is lower than maximum value of penalization as defined by local authority:

$$p_c < p_m \tag{1}$$

where:
p_c - computed value of penalization between local and remote authority
p_m - maximum value of penalization as defined by local authority at the time
The maximum value of penalization at the time is proposed to prevent the overload of distributed certificate storage and to limit the amount of cross-certificates to further mitigate the impact of recertification, defined as:

$$p_m = f_1(k_l).f_2(k_r) \tag{2}$$

where:
k_1 - utilization rate of distributed storage at the time,which will be introduced later

f_1 - is the probability density function of exponential distribution:

$$f_1(x) = \left\{ \begin{array}{ll} \lambda e^{-\lambda x} & ;\mathrm{x} \geq 0 \\ 0 & ;\mathrm{x} < 0 \end{array} \right\}, \text{ with } \lambda = 2,5 \quad (3)$$

The utilization rate is computed as:

$$k_l = \frac{P_x}{k_u} \quad (4)$$

where:
P_x the amount of received currently valid cross- certificates
k_u - the capacity of distributed certificate storage computed as:

$$k_u = \sum_{1}^{n} k_j \quad (5)$$

Where n is the amount of nodes which owns certificate issued by local authority with permission level of L2 and k_j is indicated storage capacity on node j. The indicated storage capacity is a part of L2 certificate request.
Continue in equation (2):
k_r - growth potential of distributed certificate storage
f_2 - is a function of cumulative distribution function

$$f_2(x) = \left\{ \begin{array}{ll} 1 - e^{-\lambda x} & ;\mathrm{x} \geq 0 \\ 0 & ;\mathrm{x} < 0 \end{array} \right\}, \text{ with } \lambda = 10 \quad (6)$$

Values of Lambdas were set empirically after series of tests. The growth potential k_r is computed the following way:

$$k_r = 1 - \left(\frac{P_{L2}.u_r}{P_{L1} + P_{L2}} \right) \quad (7)$$

where:
P_{L2} - is the amount of issued, currently valid certificates, with permission level L2
P_{L1} - is the amount of issued, currently valid certificates, with permission level L1
u_r - is reachability index, computed as a ratio between P L2 and number of entries in Originator table. The Originator table contains entries about nodes reachable at that time.
Continuing in equation (1), computed value of penalization between local and remote authority is defined as:

$$p_c = f_3(S_B) + M_B \quad (8)$$

where:
S_B - ratio between the dimensions of local and remote authority ecosystems:

$$S_B = \frac{P_A}{P_B} \quad (9)$$

f_3 from equation (8) is the probability density function of exponential distribution:

$$f_3(x) = \left\{ \begin{array}{ll} \lambda e^{-\lambda x} & ;\mathrm{x} \geq 0 \\ 0 & ;\mathrm{x} < 0 \end{array} \right\}, \text{ with } \lambda = 1,8 \quad (10)$$

M_B - penalization constant with value experimental value of 0.4, used in case that remote authority is not in list *TOP3*, containing the three most frequently seen authorities.

The next section describes the changes we have made to B.A.T.M.A.N. Advanced required enabling PKI support.

3. Modifications to B.AT.M.A.N. Advanced

The B.A.T.M.A.N. Advanced is Layer 2 routing protocol supported in Linux kernel since the version 2.6.38 onwards. MAC addresses are used for the routing so it is fully independent of the upper layer protocol - network agnostic - so we can use IP, IPv6, IPX, or any other protocol on top of it.

The B.A.T.M.A.N. Advanced network is created from the nodes in partial mesh. Each node knows about the existence of other node in the network and knows the direction (MAC address of the neighbour) to which it should forward the message. Unlike link state routing algorithm the node is not aware about the overall network topology. This on the other hand, greatly reduces the computational requirements so B.A.T.M.A.N. Advanced can be used on embedded devices, too. Its primary goal is the simplicity of configuration. Disadvantage of B.A.T.M.A.N. Advanced comes from its simplicity. B.A.T.M.A.N. is missing any form of routing updates authentication and overall security is let on upper layer protocols. This is where our solution comes in.

Following changes were made to the B.A.T.M.A.N. Advanced concept to mitigate the impact of following security problems:

- Man In The Middle attacks - through the exploiting of certification issue process
- Denial of Service attacks - over-helming the certificate authority with tons of requests
- Various routing attacks like Black Hole attack or Sybil attack - by routing the traffic through unknown, unreliable or compromised nodes

3.1 Added authentication support to routing updates

B.A.T.M.A.N. Advanced exchanges routing data through OGM messages. We have modified B.A.T.M.A.N. in the way that every sent OGM packet has to be PKI signed and verified upon reception before it is processed. The verification process consists of validation of OGM message signature. Nodes with the certificate from the same issuer or their attribute authorities are cross certified can verify OGM message signature and check if the peers permission are at least on L2 level. As a precaution against various DoS attacks OGM messages that cannot be verified are dropped. We have secured the following types of messages:

- OGM messages - contains information about neighbours in range
- Translation table request/response messages - spreading information about L1 and local devices through the network
- Roaming advertisement messages - used when L1 client roam to another node

Security was achieved by encapsulation of listed messages into new message called OGMS message which consists of following:

- Packet type (1Byte) - identifies the secured version of encapsulated message type

- Certificate hash (8Byte) - hash of certificate created from the value of Subject field
- Issuer hash (8Byte) - hash of certificate computed from the value of Issuer field
- Signature length (2Byte) - length of the signature
- PKI Signature - signature, size varies according to selected cryptographic algorithm

Each of secured message is signed with one of the certificates that node owns. Certificates used for signing process are changed in a round-robin manner with each OGMS message. Thanks to the omni-directional Wi-Fi transmits profile this is not a problem and each of neighbours receives the proper update in a finite time.

3.2 Rules for Communication and Exchange of Data

As we have told before one way of how to mitigate the impact in case the authority was compromised is to obtain digital identity from several authorities. The level of permission is tied to digital identity and is not movable. This way node has to build its reputation synchronously among several authorities. This could be time consuming process and in case of large network only marginally usable. Furthermore node which owns certificates from several authorities can be called multi-homed node. Now we see that we need to develop rules for the exchange of routing data.

3.2.1 Sharing of route data

The routing protocol was modified in the way that routing data are forwarded to the network only if transmitting node is able to verify the validity and non-repudiation of routed data. The node can verify routing data only if OGMS message was signed with certificate issued by the same issuer or there is a cross-certificate between both authorities. If OGMS signature was verified, routing data are extracted and added to the queue containing data from the same trusted anchor. In case of cross-certificate with permission level of L1, data are aggregated to the queue, containing data from remote authority. If the permission level is L2 data are extracted and merged with queue containing local trusted anchor. As we can see, with help of data aggregation into queues, based on trusted anchor, we can safely isolate data from various ecosystems.

3.2.2 Routing of data between ecosystems

The routing protocol routes data based on information contained in Originators table. Originators table contains data about all nodes in the network accessible at that time. To be able to distinct between ecosystems we have to modify Originators table following way:

- Extends the Originators table with ca_crt_hash column, containing hash of trusted anchor's certificate.
- Modify the B.A.T.M.A.N. Advanced to differ between various ca_crt_hash values, to separate data from different authorities.

3.3 Cluster Glue

As we have stated before, the number of authorities in the network is not limited and every node is able to provide this type of service. The probability of successfull verification of peer certificate depends on the existence of cross certificate between both authorities. It is suitable to limit the number of cross certificates to prevent authority overload, on the other side.

After we have run several sets of simulations we have detected several cases of limited ability to verify peer certificate. This could happen in a highly mobile networks, like Vehicular Ad-Hoc networks, or in the areas with obstacles preventing the transmission of a radio waves. Special case happens when set of nodes moves out of the zone where their authority provides its services. This situation leads to the formation of clusters which owns the certificate issued by the same authority. All of the clusters except the one which contains the authority itself are unable to reach authority services. Clusters which do not contain their authority dissapears, when the certificates on nodes expires. As we have stated before routing information data which cannot be verified are dropped. This happens on the border node of cluster - the first node which cannot verify routing data. To further mitigate the presense of isolated clusters we have created the concept of Cluster glue. On the fig.1, we can see the network consisting of several nodes which are divided into clusters.

The idea is as follows: If the node is a member of at least two clusters and owns the certificates issued by the authorities of both clusters with the permission level of at least L2, it represents the Edge node. The Edge node should announce its presence so other nodes should be able to locate it. After that the Edge nodes can tunnel the data through the transmitting cluster. Moreover they are reponsible for tunneling of traffic between clusters. To prevent the negative performance impact of Edge nodes, information about their presence is broadcasted only after following requirements were fullfiled:

- The remote cluster contains at least two nodes
- The remote cluster contains just one node and this owns certificate with permission level of L1
- The remote cluster contains the node providing authority services to cluster.

If these requirements are not fullfiled the information about the presence of edge node is transmitted in reduces manner, only in every tenth OGM message. The B.A.T.M.A.N. Advanced works the way where each node announce every other node listed in its *Originators* table to the network. We have designed the Cluster glue as an extension to OGM message structure. This means that information about the presence of Edge nodes have to be attached to the OGM message, too. To make this procedure effective we have designed the Edge node matrix, memory structure that stores the data about advertised clusters.

The following example describes how the Cluster glue works in a network on Figure 1. [8]

- Nodes B and D represents the Edge nodes

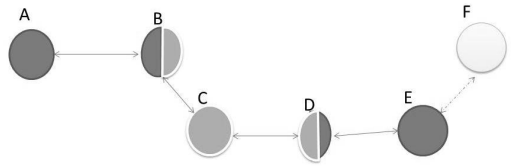

Figure 1: Example of clusters in the network

- The color of node represents its cluster's membership
- Node A and B provides authority services
- Node F owns certificate issued by authority A with the level pof permissions L1

As we can see there are two clusters of nodes with certificate issued by node A, separated by nodes B,C and D. Cluster glue works the following way: Extended OGM message contains information about the presence of Edge nodes. The challenge is the exchange of this information between the nodes B and D. The OGM extension consist of field *cgcount* (1B) containing the number of clusters behing the Edge node followed by the thumbprints of certificates (8B/entry).

After the node C receives the OGM message from node D, the message is verified with certificate issued by authority B. Routing data are processed, stored and prepared to be transmitted to the network.

Node B receives the message transmitted by node C, validates and process the routing data. Node B extract data about the presence of node D as an Edge node, too. Information about the presence of node D is stored in Edge node table. Edge node table is another part of our extension.

Similar situation happens in opposite way. The data from the Edge node tables are used to established tunnelled connection between the both Edge nodes and consequently clusters. This point-to-point or point-to-multipoint connection can be built upon arbitrary technology like GRE, etc. and is used to transfer routing and general data. To futher prevent overload caused by Extended OGM messages, the cluster glue is designed to work across only one intermediate cluster. We can think about Cluster glue as a temporary substitute to cross certification or a way how to provide communication between clusters if cross certification is not yet possible. It is important to note that this solution consume the resources of intermediate cluster and its security policy could deny the use of Cluster glue functionality.

4. Verification and results

This section contains the results of several performance and reachability tests.

4.1 Comparison of Computational Overhead Caused by Selected Cryptographic Algorithms

We have completed the series of tests where we have measured the bandwidth and computational load caused by Elliptic curve and RSA algorithm. The tests have been completed on following devices:

Table 1: Performance comparison of cryptographic algorithms

Algorithm	Dev No.1		Dev No.2		Dev No.3	
	Sign	Verify	Sign	Verify	Sign	Verify
RSA 2048	354	11529	67	2240	6	224
ECDSA 256	4921	1134	1827	397	-	-

- HP ProBook 4310s, CPU Intel Core2Duo T6670, 6GB DDR3 800Mhz RAM, 4389 Bogomips, Ubuntu 13.04 64bit, Dev. No. 1
- HP Micro Server N40L, CPU AMD Turion II Neo 1,5 Ghz, 2GB Ram ECC DDR3 1066MHz, 2995 Bogomips, Ubuntu 12.04 LTS 32bit, Dev. No. 2
- Apple iPhone 4, CPU Apple A4 800MHz (ARM Cortex-A8), 512 MB DRAM, Dev. No. 3

As a benchmark tool we choose the OpenSSL 1.0.1c. In Table 1 we can see performance results, represented as a number of operations measured in second.
The second objective was to measure the bandwidth overhead caused by the PKI signature. In case of RSA 2048bit algorithm the signature length was 340 bytes. In case of ECDSA 256bit the signature length was 90 bytes. After the further analysis we have conducted the following statement:

- The node needs to sign in each period at least the same amount of messages, as is the number of certificates it owns.
- The node needs to verify in each period at least the same number of messages, as is the number of its direct neighbours.

As we can see there is a high probability that we will receive far more messages from our neighbours than we will send. This means that RSA2048 is despite of its size better option. Standard OGM header has size of approx. 27 bytes and OGM message containing encapsulated entry has approx. 52 bytes. The overhead caused by PKI signature is rather big but since B.A.T.M.A.N. Advanced version 2010.0, OGM aggregation is enabled by default. So we are able to send more than one entry in the OGM message. Preliminary testing shows that PKI overhead could be acceptable.

4.2 Performance Tests

Furthermore we tested the overall usability of introduced concept, by testing the convergence of networks as it grown. Next we have measured the performance impact and network throughput before and after the PKI security layer was enabled. Tests were run in the virtual environment which consists of the following components:

- HP MicroServer N40L, CPU AMD Turion II Neo 1,5 Ghz, 2GB Ram ECC DDR3 1066MHz, 2995 Bogomips, HDD 2x250GB 7200rpm Seagate, RAID1

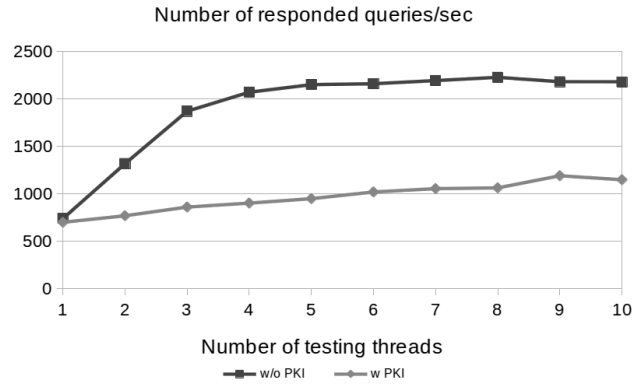

Figure 2: Measuring the performance impact caused by PKI enabled routing protocol

- Qemu-kvm 1.1.2, VDE_switch 2.2.3, Wirefilter for the simulation of packet loss between nodes
- Virtual Guests - x86 architecture, 32MB RAM, squashfs, 2 x Virtual NIC
- Topology of the network - network with dimensions 150x150m, 3 clusters, contained 35 nodes, 5 of them moved diagonally and the rest of nodes moved in random direction. Thanks to the Wirefilter we were able to simulate changes in the network topology - nodes movement - during test.

Performance test was done with help of Apache benchmark utility. Requested page was simple "It Works!" example page. The test parameters were as follows:

- Number of queries: 30 000
- Number of testing threads: 1-10
- Keep-alive: yes

As we can see on Figure 2, the highest number of served responses (2229 per second) was achieved with 8 parallel testing threads. After the PKI was deployed, the highest number of served responses (1192 per second) was achieved with 9 testing threads. This means that performance after deployment of PKI is 54,38% of former value. The use of visualized environment could be partially responsible for this effect.

4.3 Network Throughput Tests

Network throughput tests shown on Figure 3 were realized with help of iPerf tool. We have measured the network throughput of TCP and UDP protocol connections before and after the deployment of PKI. Measurements were done bidirectional between remote nodes. This way routing protocol overhead can be measured. Measurements before PKI deployment were quite stable reaching (UDP 38,5/39,12 Mbit/s; TCP 32,39/33,54 Mbit/s) throughput with maximum deviation of 2,1% from average value. After that we have fully deployed PKI the situation rapidly changed. We could see a lot of software interrupts as well as high amount of IO-APIC-fasteoi interrupt on a network interface. Host CPU was heavy loaded, too. This caused increase of deviation to value of 36,36% and decrease of network throughput to (UDP 11,96/11,96 Mbit/s; TCP 12,98/13,98 Mbit/s).

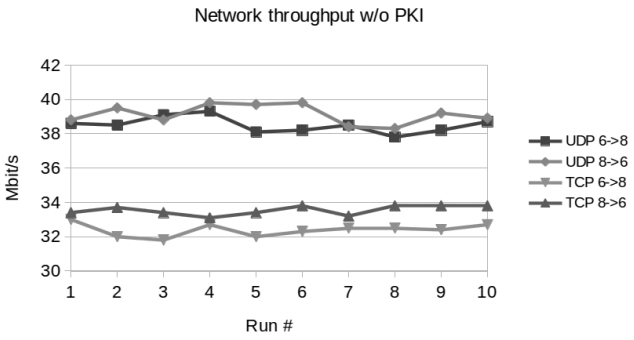

Figure 3: Measuring the network throughput impact caused by PKI enabled routing protocol

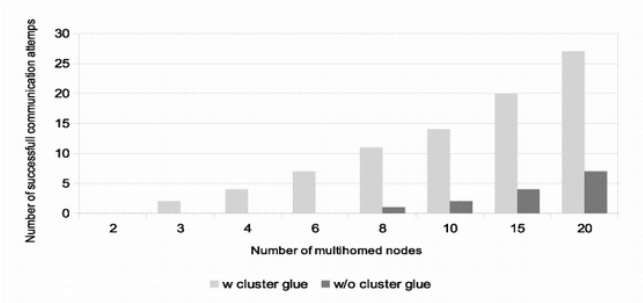

Figure 4: Number of successfull communication attempts in 30 runs

4.4 The effect of ClusterGLUE

The first measurement we have done was to measure the effect of Cluster glue. We have started with two multihomed nodes in the network and continued up to total of 20 multihomed nodes. In each run we have made 30 attempts to establish communication between nodes in remote parts of the cluster. The results were recorded and visualised on Figure 4. As we can see there is major improvement in node reachability compared to our previous solution without Cluster glue. Next we have tried to measure the computational and bandwidth overhead caused by the Cluster glue compared to the numbers of Edge nodes in the intermediate cluster. In this simple scenario only two authorities were used, so Edge node will advertise one another cluster. On the other side, every other advertised cluster means another 8 bytes added to OGM message, but this is negligle compared to 340 bytes added by RSA2048 signature [7]. On the other side, advertising

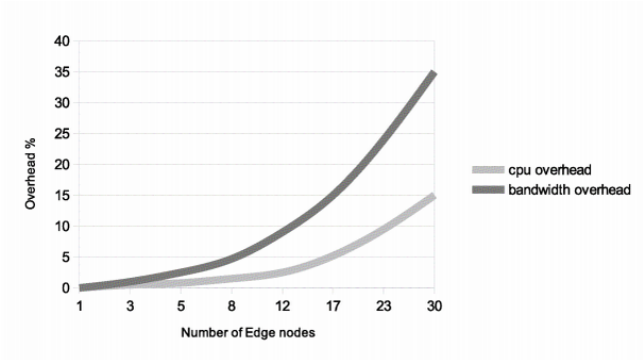

Figure 5: Computational and bandwidth overhead

more than 50 Edge nodes in one cluster could presents higher requirements on bandwidth and increased computational costs caused mainly by tunnelled traffic. Figure 5 displays the results of computational and bandwidth overhead test.

5. Conclusions

We have presented the concept of public key infrastructure on top of B.A.T.M.A.N. Advanced routed network. The idea was to gradually raise nodes permissions to mitigate the effect of MITM, Black Hole, Sybil Attack and compensate the fact that each authority is located on a single node. On the other hand each node can create its own authority and build its own ecosystem of client nodes. Furthermore we have designed the extension to OGM messages which brings PKI authentication support to B.A.T.M.A.N. Advanced and introduced the concept of privilege level escalation and cross certification. Next we have designed the computational formula for optimum amount of cross certificates. Another challenge was to improve the service reachability in highly mobile network, where the clusters can appear. Our solution is targeted against the situation, where remote clusters are separated by the intermediate cluster and cross certification is not possible. As our measurements proved, the computational overhead of ClusterGLUE is almost neglible and the level of bandwidth overhead is acceptable. On the other side, more certificates node owns, the higher will be bandwidth overhead and this special case will require Edge node selection algorithm to be developed. Since we have made substantial changes to B.A.T.M.A.N. Advanced we have extended Wireshark dissector plugin to keep it fully compatible with our changes. Our measurements proved that this concept is usable and can be deployed even to mobile devices.

Acknowledgements. This work was supported by VEGA 1/0649/09, 1/0722/12 grants.

References

[1] C. S. CAGALJ, M. and J. HUBAUX. Key agreement in peer-to-peer wireless networks. In *Special Issue on Cryptography and Security*, pages 467–478. IEEE, 2006.

[2] H. J. CAPKUN, S. and L. BUTTYAN. Mobility helps peer-to-peer security. *IEEE Trans. Mobile Comput.*, 5(1):43–51, 2006.

[3] L. Caromel, ZHOU and Z. J. HAAS. Securing ad hoc networks. *IEEE Netw. Special Issue on Network Security*, 13(6):24–30, 1999.

[4] M. HOLOHAN, Edmond; SCHUKAT. Authentication using virtual certificate authorities: a new security paradigm for wireless sensor networks. *9th IEEE International Symposium*, 9(1):92–99, 2010.

[5] Z. P. K. J. L. S. LUO, H. and L. ZHANG. Self-securing ad hoc wireless networks. In *Seventh International Symposium on Computers and Communications*. IEEE, 2002.

[6] K. W. X. C. Pengrui Xia, Meng Wu. Identity-based fully distributed certificate authority in an olsr manet. *WiCOM'08. 4th International Conference on.*, 4(1), 2008.

[7] L. VILHAN, P.; HUDEC. Building public key infrastructure for manet with help of b.a.t.m.a.n. advanced. In *UKSim-AMSS seventh European modelling symposium on computer modelling and simulation,*, pages 530–535. IEEE Computer Society, 2013.

[8] L. VILHAN, P.; HUDEC. Cluster glue - improving service reachability in pki enabled manet. In *UKSim-AMSS 16th international conference on computer modelling and simulation.*, pages 493–498. IEEE Computer Society, 2014.

[9] S. YI and R. KRAVETS. Moca: Mobile certificate authority for wireless ad hoc networks. In *2nd Annual PKI Research Workshop*, pages 84–89. IEEE, 2003.

Selected Papers by the Author

P. Vilhan, J. Gajdos. ADEUS: Tool for Rapid Acceleration of Network Simulation in OMNeT++. In *14th International Conference on Modelling and Simulation,* Proceedings; Cambridge, United Kingdom; 28-30 March 2012. - Los Alamitos : ISBN 978-0-7695-4682-7, IEEE Computer Society, 2012.

P. Vilhan, L.Hudec. Building public key infrastructure for MANET with help of B.A.T.M.A.N. Advanced. In: EMS 2013 : proceedings. *UKSim-AMSS seventh European modelling symposium on computer modelling and simulation,* 20-22 November 2013, Manchester, United Kingdom. - Los Alamitos : ISBN 978-1-4799-2578-0. - pp. 530-535, IEEE Computer Society, 2013.

P. Vilhan, L.Hudec. Cluster glue - improving service reachability in PKI enabled MANET. In: UKSim-AMSS 2014 : proceedings *UKSim-AMSS 16th international conference on computer modelling and simulation*, 26-28 March 2014, Cambridge, United Kingdom. - Los Alamitos : ISBN 978-1-4799-4923-6. - pp. 493-498, IEEE Computer Society, 2014

I. Grellneth, P. Vilhan, M.Hruby. Emulating Cisco Network Laboratory Topologies in the Cloud. In: *ICETA 2011 : 9th IEEE International Conference on Emerging eLearning Technologies and Applications*, October 27-28, 2011. Stará Lesná, The High Tatras, Slovakia. - Piscataway : ISBN 978-1-4577-0050-7. - pp. 67-92, IEEE, 2011.

P. Vilhan, P. Marko, I.Grellneth. Efficient detection of malicious nodes based on DNS and statistical methods. In: *SAMI 2012 : IEEE 10th Jubilee International Symposium on Aplied Machine Intelligence and Informatics*, Herľany, Slovakia, January 26-28, 2012. - Budapest : ISBN 978-1-4577-0195-5, Obuda University, 2012.

P. Vilhan, L.Hudec. Improving Deployability of PKI in MANET Networks Routed by B.A.T.M.A.N. Advanced. In: *IIT.SRC 2013: 9th Student Research Conference in Informatics and Information Technologies*, Bratislava, April 23, 2013. Post-conference proceedings. ISBN 978-80-227-4111-8, pp. 389-396 - Bratislava : Nakladatel'stvo STU, 2013.