Analysis of Broadcast Authentication Mechanism in Selected Network Topologies

Tomas VANEK, Matej ROHLIK

Dept. of Telecommunication Engineering, Czech Technical University in Prague,
Technicka 2, 166 27, Prague 6, Czech Republic

tomas.vanek@fel.cvut.cz, rohlimat@fel.cvut.cz

Abstract. *This paper deals with simulation of the broadcast authentication protocols using Colored Petri Nets and further optimizations in Matlab environment. Typical application of broadcast authentication protocols can be configurations where only one transmitter with multiple recipients exists (such as message exchange in sensor networks routing protocols, or the leader election process in sensors network). Authentication of every packet seems to be very effective way to mitigate an attack, however resulting in increase of end-to-end delay. To mitigate this drawback, the broadcast authentication protocols have been proposed. Concept of optimization of the broadcast authentication protocol DREAM parameters in a special case of fully N-ary tree and general random topology containing the same amount of nodes with regard to delay and energy consumption minimization is showed in the paper. Protocol DREAM was taken as an example of broadcast authenticating protocol to show how Color Petri Nets can be used to create a fully functional model of the protocol.*

Keywords

Authentication, protocol, security, DoS, DREAM, N-ary tree.

1. Introduction

Currently, one of the most destructive kinds of attacks is Denial of Service (DoS), primarily its distributed form – DDoS (Distributed DoS). For the attacker, it is quite easy to overload selected server or whole network. Vulnerability to DoS attacks is much more striking in broadcast communication when each packet should be delivered to all nodes in the network. If the attacker is able to generate a sufficient number of packets, the whole network can be overloaded. It is possible to avoid such incident by verifying the origin of each packet in the network before its processing in the node. However, in the network whose nodes have limited computing power and memory (typical for sensor networks), each authentication process using traditional public key cryptographic techniques is quite burdensome. A big number of these calculations lead to growth of time that the packets spent in the network and communication among nodes becomes almost impossible. In the case of broadcast communication, this issue can be efficiently solved by DREAM (DoS-Resistant Efficient Authentication Mechanism) [1] protocol.

2. DREAM Protocol

DDoS is so powerful since it uses multiple systems as resources of the attack and therefore, it is much stronger than a single sourced attack. DREAM mitigates the DDoS impact by involving an analogous approach which the DDoS uses itself. The only difference is that more stations are involved in the process of verification. The DREAM protocol can operate in two modes: normal and secure. Every incoming message is authenticated by the network node before being sent to the outgoing interface in the secure mode. In the normal mode, some of the messages are sent directly to the outgoing interface without being authenticated. This approach mitigates the potential single point of failure in the whole network since there is not a single node where authentication occurs. The verification process is distributed among the neighboring nodes. The protocol functionality is influenced by the following parameters [2]:

- *NBR* – number of neighbors.
- *HT* – number of nodes that message passed without authentication. For such each node, the parameter is incremented by one. When the packet is authenticated *HT* is set to zero.
- *K* – maximum number of nodes, that can message pass without authentication.
- *b* – expected number of neighbors in unity distance from the source.
- *c* – expected number of neighbors in unity distance from the last node that forwards the message.

The amount of messages to be sent or verified before sending out the interface is defined by the following decision rules (formulas) [2]:

$$Rand < \frac{b}{NBR}, \tag{1}$$

$$Rand < \frac{2c}{NBR} \tag{2}$$

where *Rand* is a random number generated independently by every node for every message in the range of 0 and 1 with the continuous uniform distribution [1]. The first formula is used when the message comes directly from a neighbor, a neighbor has been verified, or the parameter $HT = 0$. The second formula is used if the message did not come from a neighbor, or a neighbor has not been verified, or the parameter $HT > 0$.

3. Model of DREAM Mechanism

Two very different tools have been used to create a model of DREAM protocol and simulation of the process of authentication. The finite state machine using of the protocol was done in Colored Petri Nets (CPN) while simulation of spreading the messages through network was done in Matlab.

3.1 Colored Petri Nets

A Petri net is a formalism that can be used to describe systems or protocols. It has some differences to the classical finite state machine. The most significant difference is the ability to describe concurrency among processes. A Petri net consists of places and transitions which are connected by unidirectional curves. The Petri net formalism is very powerful in describing the behavior of systems where the actions are performed in sequences. The Colored Petri Net [2] is an extension which consists in using the "colors" in the Petri nets. This is a very helpful approach to describe most of the details which should be shown when modeling some complex systems. The tokens in a colored Petri net are not equal one to each other but they are differentiated by means of colors [3].

The model of the protocol was created in CPN Tools 2.2.0 [2], [3]. Fig. 1 depicts the protocol described as a finite state machine. The model itself can be divided into six main parts. In the first one, the initial check is done to find whether a packet was received in the past or not. This is done by checking the sequence number of the packet and the *ID* of the source. The second part analyses the *ID* of the last node that accomplished the verification. Only messages from trusted neighbors are accepted. In the next step, normal or secure mode is selected. In the fourth part, the decision making process is based on the number of nodes passed without verification (*HT*). Messages that have undergone the maximum number of nodes without verification are immediately sent to the queue for verification. In the next step, the reports are divided into two sets, the first with *HT* equal to zero and the second with *HT* greater than zero and less than the maximum permissible value. The model further branches into two parts. The left part of the messages will be sent prior to authentication. In these messages, the variable *HT* is increased by one and they are sent afterwards. For the messages in the right part of the model, namely those for which there is first verification and then they are sent, the variable *HT* is set to zero and are then sent to a queue for verification.

The whole model has thus three exit points: *dropped msg*, *msg sent,* and *counter count*. The *msg dropped* contains all the messages that have been discarded, whether on the basis of duplicate sequence, unknown neighbor, or a false signature. In the *msg sent* only sent messages are stored, i.e., those which were sent without verification and sent with the verification. The *counter* contains information about the number of true and false reports from the certification module.

The general DREAM model has been reduced according to considered assumptions which are explained in section 3.2. Slightly modified protocol was used in N-ary tree topology. Since this topology is loop-free we dropped detection module of repeated messages. Another assumption was that the protocol worked only in normal mode, so the detection of falsified messages was removed.

3.2 Analysis of the Protocol Parameters

A time unit for authenticated and sent message has been determined as 100 % and the time unit for message sent without verification estimated as 20 % of the determined time unit. Since the duplicate message detection process checks only the received packet ID and a time unit of such procedure has been estimated as 1 – 2 %. The simulation considers the following assumptions: if a duplicate packet is received, it is immediately dropped by the system. Such time is negligible compared to the verification and decision based times. Since a packet is being verified in every single node, a small amount of time is saved only by sending the packet before verification is executed.

Such assumptions enable to change the general network topology into a loop free network scheme – into a tree and to omit the packet duplicity detection. Consideration that every single node of the tree topology is connected to the same number of neighboring nodes results into an N-ary tree topology.

For the first simulation, the following expectations have been considered:

- The network contains only one transmitter. All other nodes are engaged into verification and forwarding processes only.
- The topology of the network is unchanged during the simulation, i.e., the number of neighbors does not change either.
- All the nodes are working in the normal (not the secured) mode during the whole simulation.
- Every node has exactly the same number of neighbors so the network topology represents a fully N-ary tree and resulting into the loop-free topology

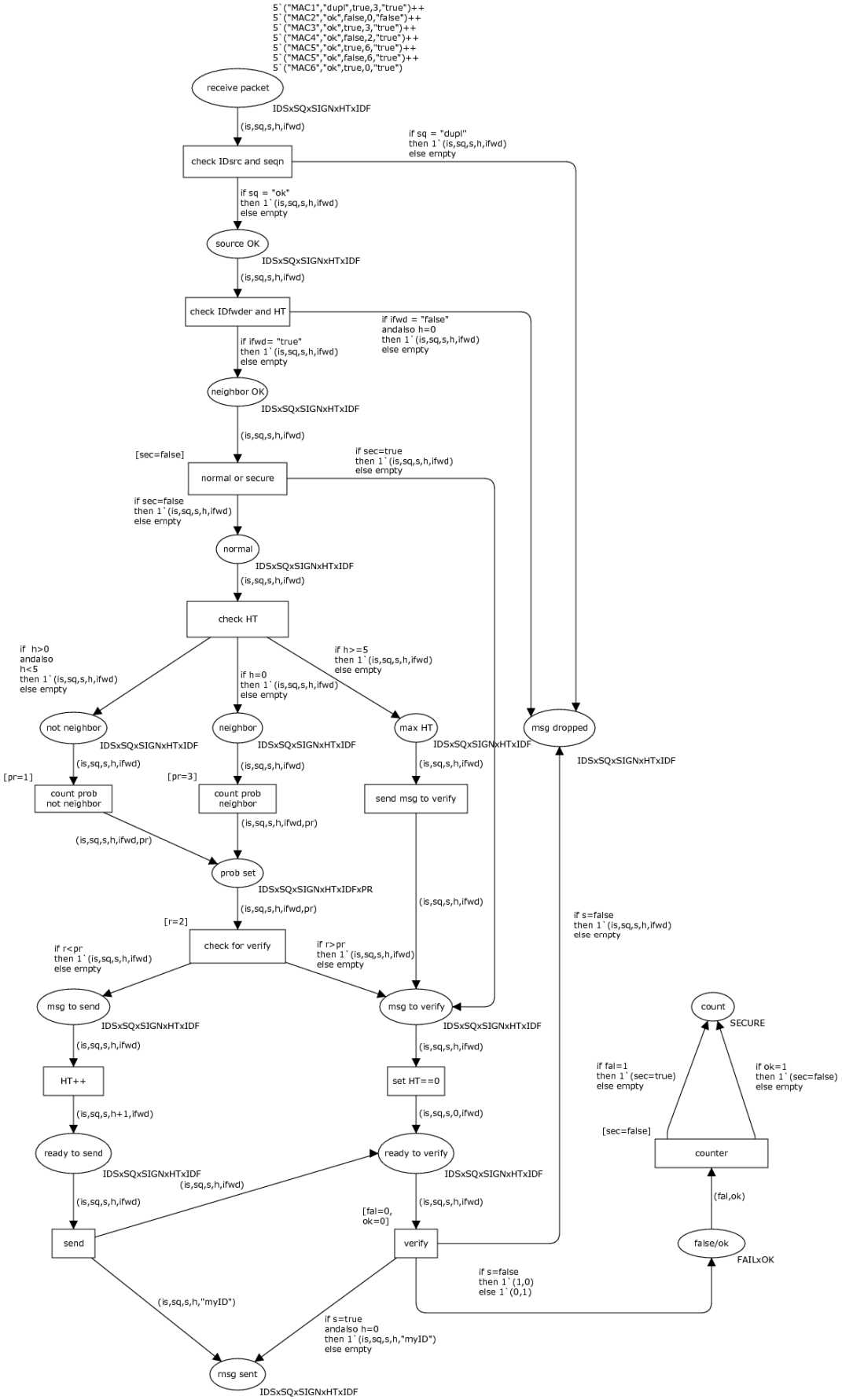

Fig. 1. The model of complete DREAM protocol in CPN environment.

Values c or b lesser that zero cause the model to enforce each node to work in the secure mode resulting in verification of all incoming messages before sending them. On the other hand, values $c \geq NBR/2$ and $b \geq NBR$ cause that each node sends the message without being verified. Therefore the parameters b and c have been selected with respect to (1) and (2) to ensure the random model behavior. This means that the parameters b and c must confirm the following conditions [4], [5]:

$$0 < b < NBR., \tag{3}$$

$$0 < c < \frac{NBR}{2}. \tag{4}$$

To limit the network topology and to respect the practical application of the simulation, a *length* variable has been determined and defined as follows: the *length* is the maximum number of nodes in a row to be passed by the message from the transmitter to the last node (a leaf of the tree). Since the total number of all nodes depends on *NBR* and *length* parameters, it can be counted as (5):

$$\sum_{i=0}^{length} \frac{NBR^i}{NBR-1} = \frac{NBR^{length+1}-1}{NBR-1} \tag{5}$$

where $NBR \neq 1$. To reflect the practical usage of this simulation, the *length* variable has been selected with respect to the *NBR* variable to generate at the most one million of nodes in total.

The selection of the parameter K value higher than the *length* value does not affect the simulation. The packet cannot get further than the maximum network length:

$$0 \leq K \leq length. \tag{6}$$

4. Simulations

In the following section, the general network of a large amount of mutually interconnected nodes is considered. The goal of the simulation is to analyze the protocol parameter influence on the unit time delay caused by the authentication process. The following two topologies are considered – a loop-free topology with a constant number of neighbors in each node and a random topology (containing loops and having a random number of neighbors).

The authentication process is considered as general as well as the network. No other delays caused by specific higher level authentication methods or media access protocols are assumed, thus do not affect the simulation process. The fact is that the total delay of the message each node will be also influenced by lower layers, e.g., specific MAC mechanism. However, there are other procedures that can cause the delay as well, e.g. low computational power of the nodes or small amount of memory. All these effects are specific for each model application. The model itself is general and independent on the lower layers. Therefore, this research was accomplished in the same manner.

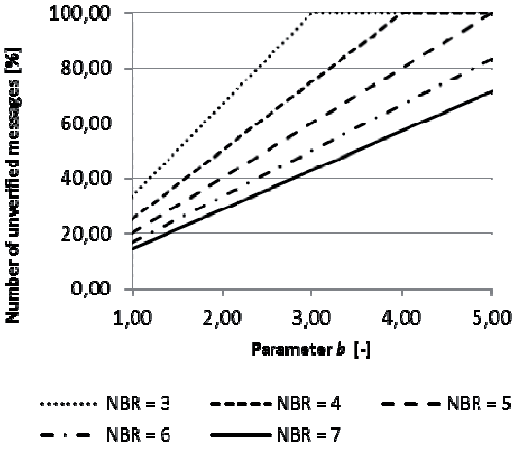

Fig. 2. Number of unauthenticated messages depending on the number of neighbors.

In Fig. 2, the percentage of unverified messages is shown in dependence on the number of neighbors and the parameter b. Remaining figures display the results of other simulations which were accomplished with $NBR = 5$.

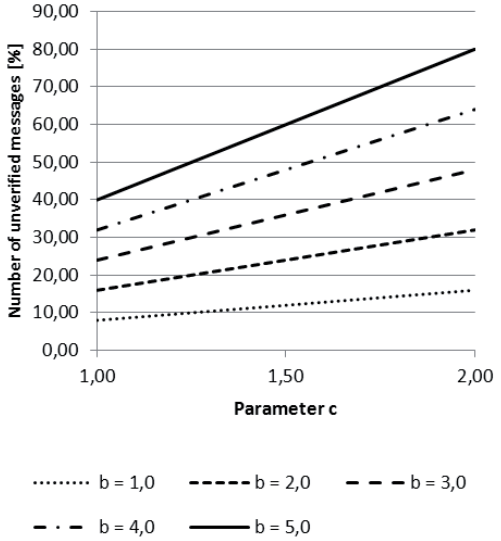

Fig. 3. Number of unverified messages after passing the 2nd node.

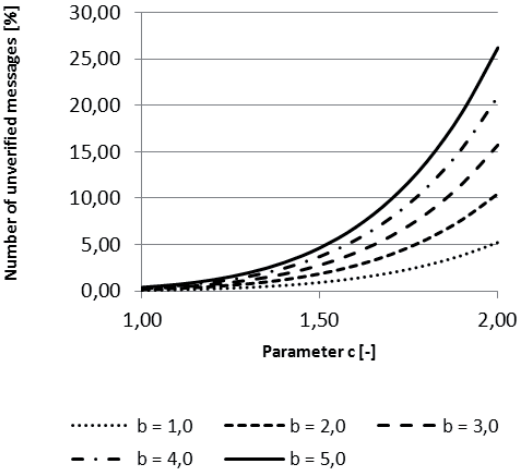

Fig. 4. Number of unverified messages after passing the 7th node.

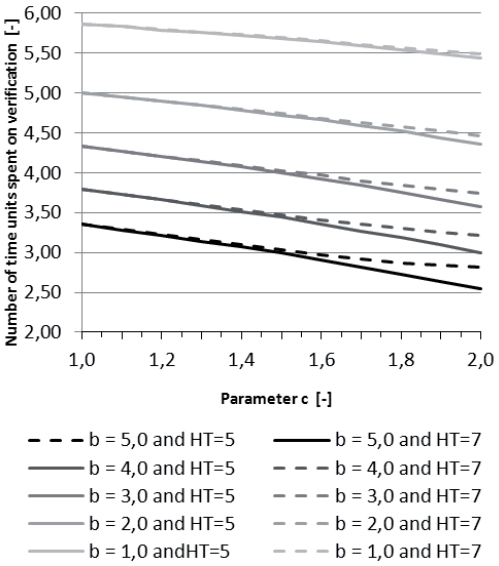

Fig. 5. The average delay of all messages after passing through the 7th node for NBR = 5.

From the presented figures 2 – 5 it can be seen that the difference between lines with the same parameter *c* and different values of parameter *HT* is much more significant only for a higher value of the parameter *c*. This is due to a higher number of messages that have not been verified and achieved maximum value for the parameter *HT* that caused a forced authentication.

4.1 Simulation of N-ary Topology

The Matlab simulation of DREAM in N-ary topology verified the importance of process selection of particular parameters. Incorrect choice of *b* and *c* parameters values can significantly affect the protocol behavior.

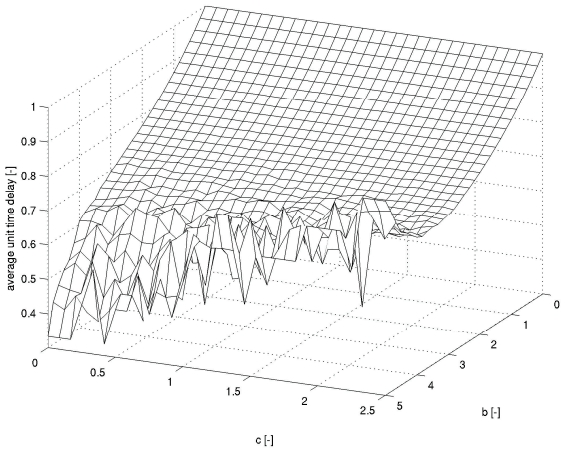

Fig. 6. Average unit time delay, $K = 1$, $NBR = 5$, $length = 8$.

To ensure the model works correctly and retains the random character, these two parameters must be chosen with respect to the conditions (3) and (4). Similar recommendation applies to the parameter *K* which does not have sense to select otherwise than with respect to the condition (6).

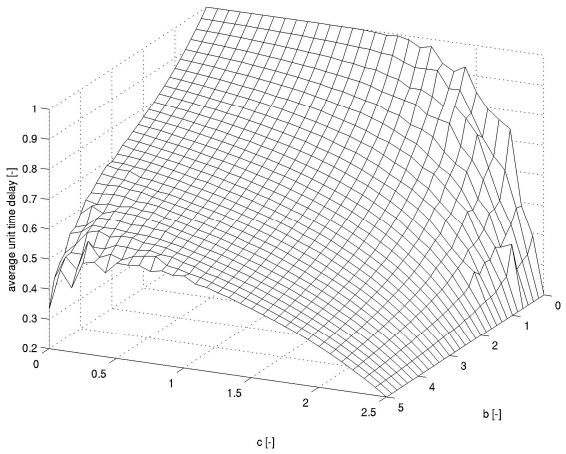

Fig. 7. Average unit time delay, $K = 3$, $NBR = 5$, $length = 8$.

Simulating the same network topology (equal *NBR* and *length* values), the average unit time delay tends to the value of unit time delay of model when no verification occurs in the nodes (Fig. 6 – 9). The difference is only for $K = 0$ (verification in every node is enforced) and $K = 1$ when the verification process proceeds very often when compared to the total number of nodes (Fig. 6). This behavior results into DREAM model independency on the *K* value (higher than 1) in the N-ary tree network topology.

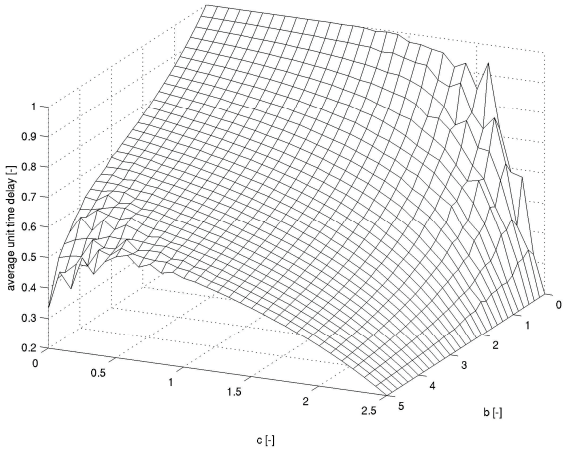

Fig. 8. Average unit time delay, $K = 5$, $NBR = 5$, $length = 8$.

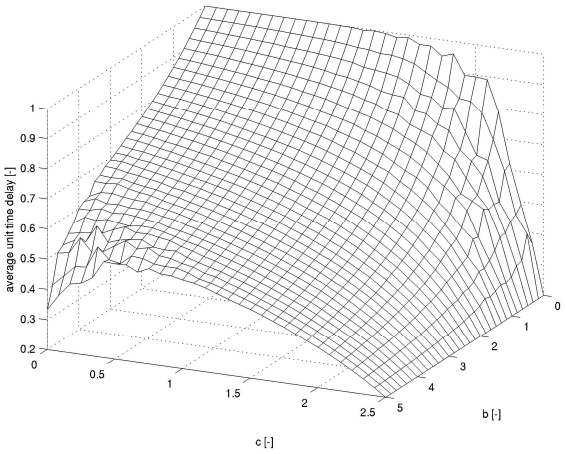

Fig. 9. Average unit time delay, $K = 7$, $NBR = 5$, $length = 8$.

4.2 Random Topology

Simulation of the random topology is a little bit tricky. The issue can occur when the randomness topology is not used correctly during the simulation. Therefore, at the beginning of the whole process, a random topology with random number of nodes is generated and such topology is used for further analysis and dependency of investigated parameters on the whole protocol behavior.

To be able to compare the random and N-ary tree topology, the total number of nodes in the whole topology preserved with respect to the practical application should be one million. The maximum number of nodes was chosen with the same respect to the first simulation and also to the practical application.

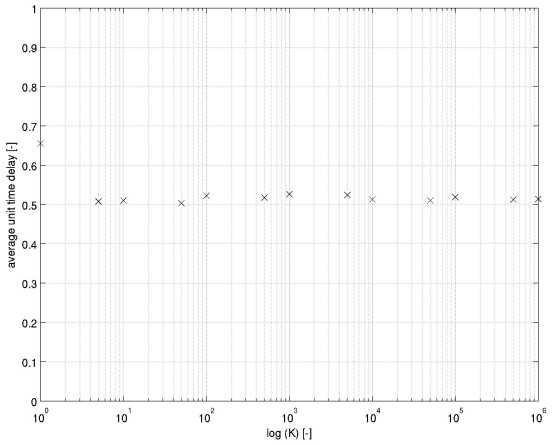

Fig. 10. Dependency of average unit time delay on *K* parameter

The average unit time delay does not depend on the *K* parameter and is depicted in Fig. 10. In further simulations, only the following values were considered:

$$0 < K < 5\,. \tag{7}$$

Since the *b* and *c* parameters should be chosen with respect to the *NBR* parameter which is being randomly generated at the beginning of the simulation, the interval was set from 0 to the maximum number of neighbors and the half of the maximum number of nodes respectively. This creates a little inaccuracy since the model forces the nodes to skip the verification process. However, this enables the simulations to be smoothly comparable.

In the following figures (Fig. 11 – 14) it can be seen that the trend of *K* parameter value in a random topology is similar to the trend in an N-ary tree topology which is independency of the *K* value parameter.

One can say that *K* is invariant according to the topology of the network. It can be seen as a good property of the protocol, because, no special optimizations are required according to the physical network topology of the real network.

In Fig. 15 the independency of *K* parameter on *b* and *c* parameters is depicted as for N-ary topology as for random topology. For each value of *K*, a simulation with randomly chosen parameters *b* and *c* has been computed. The parameters *b* and *c* have been randomly chosen with continuous uniform distribution from intervals respecting the conditions (1) and (2). The independence of parameters *b* and *c* has been demonstrated for a large range of values of *K*.

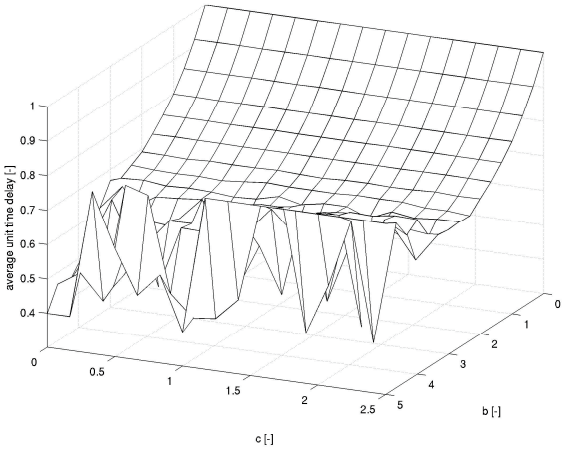

Fig. 11. Average delay of the message in the node, *K*= 1, random topology.

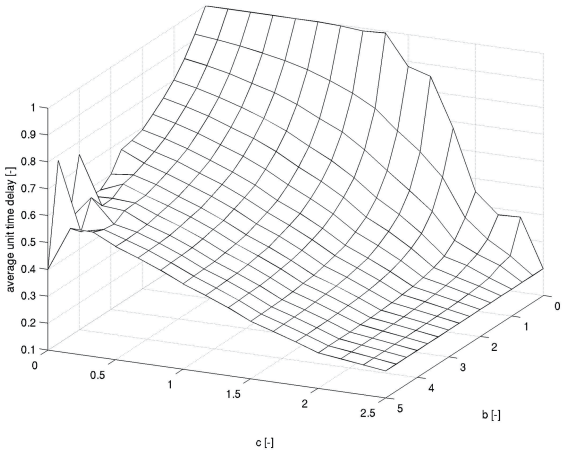

Fig. 12. Average delay of the message in the node, *K* = 3, random topology.

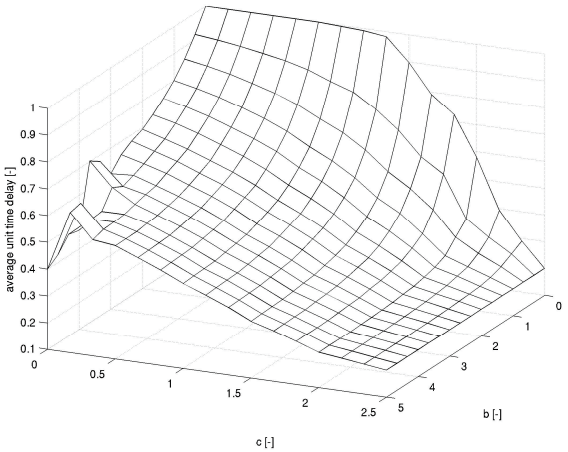

Fig. 13. Average delay of the message in the node, *K* = 5, random topology.

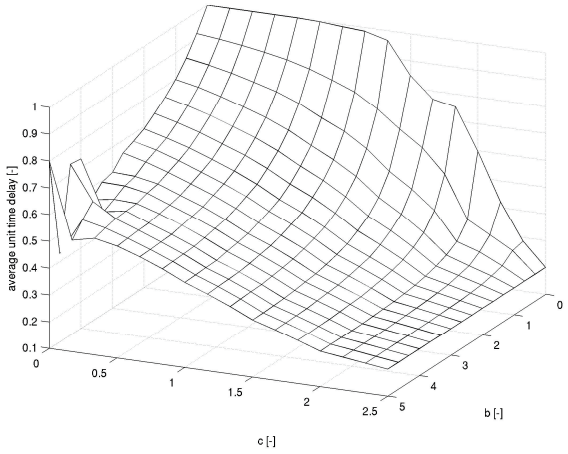

Fig. 14. Average delay of the message in the node, $K = 8$, random topology.

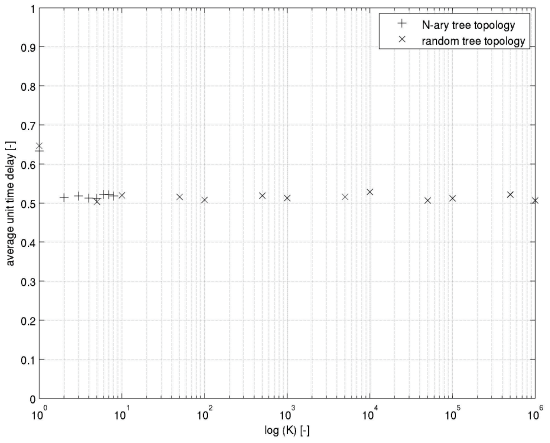

Fig. 15. - Dependency of K parameter on b and c parameters for N-ary and random topologies.

5. Conclusion

The simulations verified the importance of process selection of particular parameters. Incorrect choice of b and c parameters' values can affect the protocol behavior. To ensure the model works correctly and retains the random character, these two parameters must be chosen with respect to the conditions (3) and (4).

The analysis shows that K is independent of network topology and also parameters b and c. To minimize delays resulting from the authentication process, it is advisable to choose K close to the diameter of the network. Another conclusion is that secure mode of the protocol is useless because it greatly increases the computational demands on the edge nodes. Better solution is to keep nodes in normal mode even in during a DoS attack and let the rest of the network to participate on authentication. Another possible solution is implementation of some kind of dropping algorithm, e.g., WRED (Weighted Random Early Detection) to drop incoming messages regardless if they were genuine or forged.

Acknowledgements

This research work was supported by MSMT under the project no. MSM 6840770038.

References

[1] HUANG, Y., HE, W., NAHRSTEDT, K.; LEE, W.C. *DoS-Resistant Broadcast Authentication Protocol with Low End-to-end Delay*. [Online] Cited 2008. Available at: http://www.ideals.uiuc.edu/handle/2142/11432

[2] ALY, S., MUSTAFA, K. *Protocol Verification and Analysis Using Colored Petri Nets.* [Online] Cited 2003. Available at: http://facweb.cs.depaul.edu/research/TechReports/TR04-003.pdf

[3] JENSEN K. *Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use*. Vol. 1, Springer, 1996.

[4] VANEK, T., ROHLIK, M. Model of DoS resistant broadcast authentication protocol in colored petri net environment. In *IWSSIP 2010 Proceedings* [CD-ROM]. Rio de Janeiro: EdUFF – Editora da Universidade Federal Fluminense, 2010, p. 264-267. ISBN 978-85-228-0565-5

[5] ROHLIK, M., VANEK, T. Broadcast authentication mechanism optimization in fully N-ary tree topology protocol. In *Proceedings of the Xth Conference KTTO 2010*. Ostrava (Czech Rep.), FEI VSB-TU, p. 111-114, ISBN 978-80-248-2330-0

About Authors ...

Tomas VANEK was born in 1976. He received his M.Sc. degree in Telecommunication Engineering from the Czech Technical University in Prague in 2000. In 2008 he received the Ph.D. degree in Applied Cryptography from CTU in Prague. Currently, he works as an assistant professor at the Department of Telecommunication Engineering, CTU in Prague. His research interests include advanced network protocols and VoIP security.

Matej ROHLIK was born in 1980. He received his engineering degree in Telecommunication Engineering from the Czech Technical University in Prague in 2008. In 2008 he received the bachelor degree in specialization in pedagogy from CTU in Prague, Masaryk Institute of Advanced Studies. His Ph.D. research is focused on femtocell security simulation and optimization. He is actively involved in projects focused on femtocell security and high speed mobile transmission optimization.