Model of DoS Resistant Broadcast Authentication Protocol in Colored Petri Net Environment

Tomas Vanek, Matej Rohlik
Department of Telecommunication Technology
Czech Technical University in Prague
Prague, Czech Republic
[tomas.vanek,rohlimat]@fel.cvut.cz

Abstract—This paper deals with simulation of the broadcast authentication protocols using Coloured Petri Nets (CPN). CPN is a special instance of an orientated graph which enables to describe data flows and information dependencies inside of modeled systems. Protocol DREAM was taken as an example of broadcast authenticating protocol to show how Colour Petri Nets can be used to create a fully functional model of the protocol. Broadcast authentication protocols can be used in many situations where exist only one transmitter and multiple recipients such as message exchange in sensor networks routing protocols, or the process of leader election in sensors networks.

Keywords- authentication, protocol, security, DoS, Coloured Petri Nets.

I. INTRODUCTION

Currently, one of the most destructive types 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, he can overload the whole network. It is possible to avoid this incident by verifying the origin of each packet in the network. However, this approach leads to growth of time that the packets spent in the network and communication become almost impossible. In the case of broadcast communication, this issue can be solved by DREAM (DoS-Resistant Efficient Authentication Mechanism) [1] protocol.

II. DREAM PROTOCOL

DREAM protocol exploits similar techniques as distributed DoS attack. DDoS uses multiple systems as a sources of the attack that is much more powerful than a single attacker. DREAM involves analogous approach with the difference that more stations are involved in the verification process. DREAM has two basic modes of operation; normal and secure. In the secure mode, all the packets, which are incoming to the node, are authenticated. In the normal mode, the node verifies only some of the incoming packets and the remaining packets are routed to the network without authentication. This fact removes potential single point of failure in the network (the node where authentication occurs) and distributes the burden among the neighboring nodes. The advantages of DREAM protocol are mostly apparent in the network where packets must travel through a big amount of nodes. A typical example of such network could be ad-hoc wireless networks or wireless sensor networks with a mesh topology. Functionality of the protocol is mainly influenced by the following parameters:

- *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 number of the messages forwarded without authentication depends on three parameters: constants *b*, *c*, and the number of neighboring nodes – *NBR*.

III. COLOURED 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 consist 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 Coloured Petri Net [2] is an extension which consists in using the “colours" 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 coloured Petri net are not equal one to each other but they are differentiated by means of colours. [3]

IV. SIMULATION

The model of the protocol was created in Matlab and CPNTools 2.2.0 [4] [5]. Figure 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.

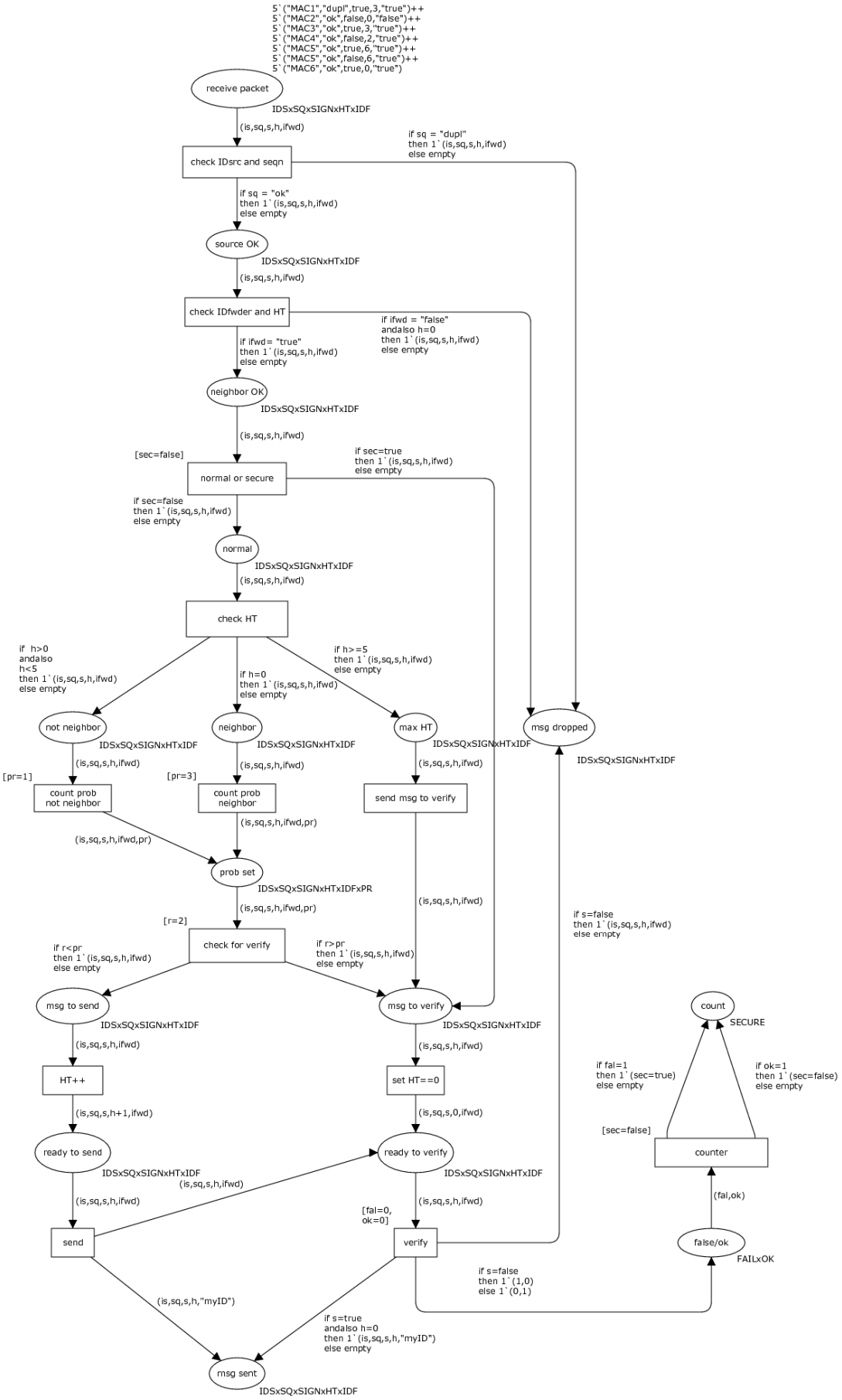

Figure 1 – The model of DREAM protocol in CPN environment

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 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* are stored only sent messages, i.e., those who 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.

We have chosen the following parameters for the simulation:

1. The network contains only one transmitter. All other nodes do verification and forwarding only.
2. Topology of the network is unchanged during the simulation, i.e., the number of neighbors does not change either.
3. All the nodes are working in the normal (not the secured) mode during the whole simulation.
4. All the nodes in the networks have the same network parameters b and c.

On the basis of these predictions, we have selected intervals for constants *b*, *c* and the possible numbers of neighbors. The protocol works differently with the messages originating directly from its neighbors and the other messages that have been already forwarded without verification. The basic decision rule, if the message will be verified or not, are the following two formulas:

$$Rand > \frac{b}{Nbr}, \qquad (1)$$

$$Rand > \frac{2 \cdot c}{Nbr}, \qquad (2)$$

where *Rand* is a random number in the range of 0 and 1 with the normal distribution.

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. The following figures show selected results of performed simulations.

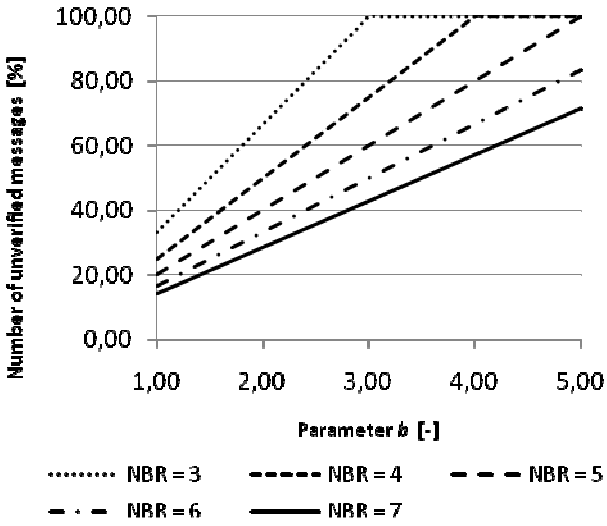

Figure 2 - Number of unauthenticated messages depending on the number of neighbors

In the figure 1, 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.

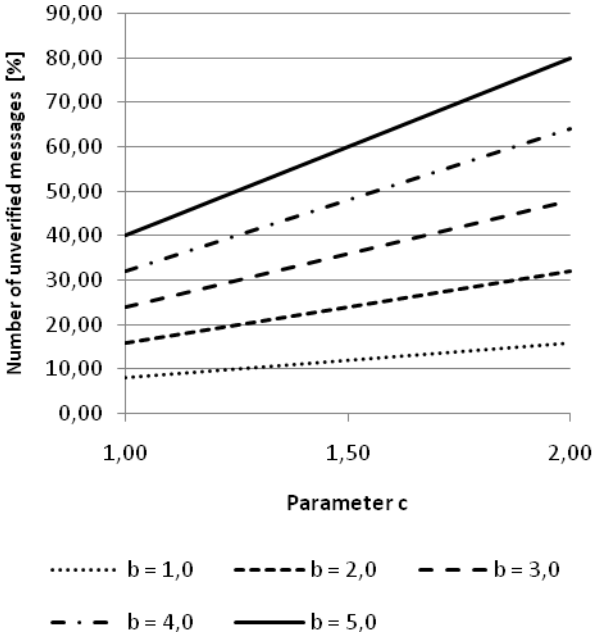

Figure 3 - Number of unverified messages after passing the 2nd node

Figures 2, 3 and 4 show the total number of unverified messages after passing two, four and seven nodes. From the fourth node depicts the dependence between the numbers of unverified messages and the parameter *c* changes from linear to exponential. This is due to multiplication of probability of message verification. It

can also be seen that the number of unverified messages is influenced mainly more by parameter *b* than *c*.

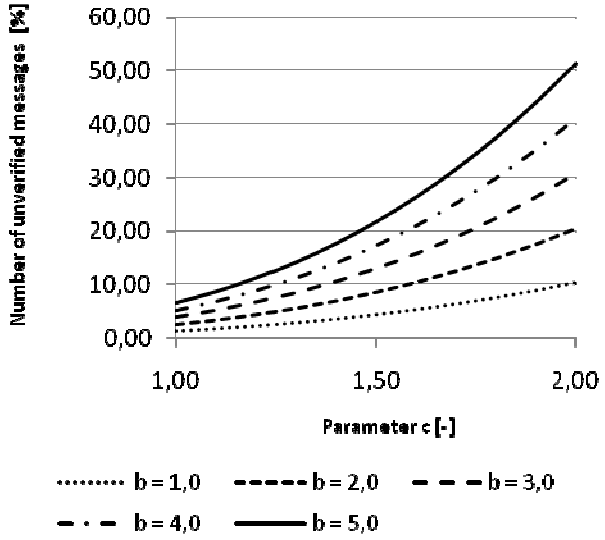

Figure 4 - Number of unverified messages after passing the 4th node

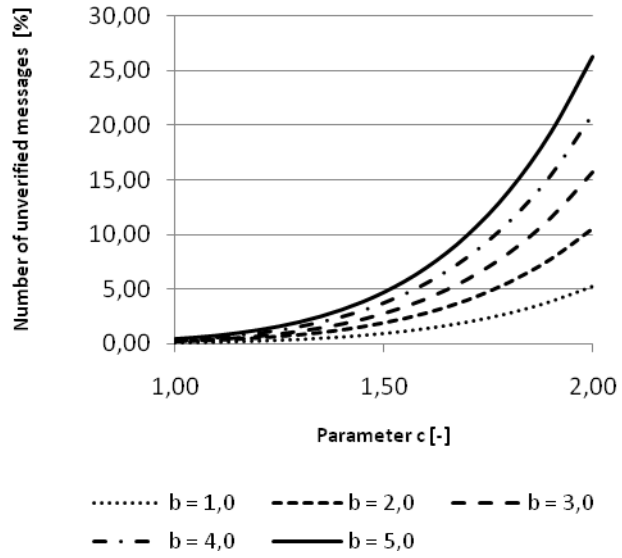

Figure 5 - Number of unverified messages after passing the 7th node

Since the time of verification of the message is unknown, it was set that one verification takes a one time unit, so the Figure 6 represents the average delay of all messages in the network at a distance seven from the source.

From the presented figures 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.

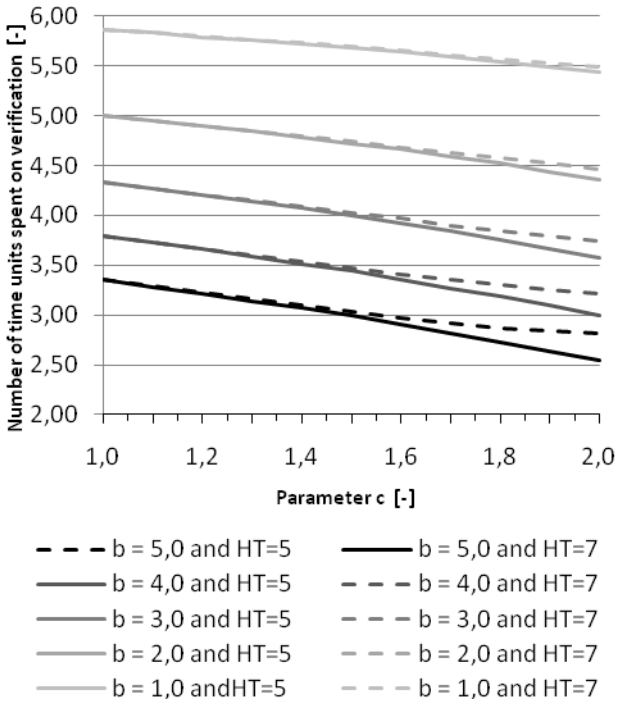

Figure 6 – Average delay of messages after passing through the 7th node for *Nbr*=5

V. CONCLUSION

In this paper, the usage of Coloured Petri Nets for broadcast authentication protocols simulation was accomplished. On the example of DREAM protocol, it was presented that CPN environment is able to simulate protocol behavior. An analysis of the impact of the most important parameters that affect the ability of the protocol to resist DoS attacks was simulated in Matlab. The simulations of the protocols continue with the analysis of the behavior of the protocol in networks with heterogeneous parameters (*b, c*) for individual nodes and networks with variable topology (mobility of nodes).

ACKNOWLEDGMENT

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

REFERENCES

[1] HUANG, Ying, HE, Wenbo, KLARA, Nahrstedt. DoS-Resistant Broadcast Authentication Protocol with Low End-to-end Delay [online], 2008, <https://www.ideals.uiuc.edu/handle/2142/11432>

[2] K. Jensen, "Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use", Volume 1, Springer, 1996.

[3] ALY, Salah, MUSTAFA, Khaled. Protocol Verification And Analysis Using Colored Petrinets [online], 2003 <http://facweb.cs.depaul.edu/research/TechReports/TR04-003.pdf >

[4] CPN Tools, <http://wiki.daimi.au.dk/cpntools/cpntools.wiki>

[5] CPN ML - Language for declarations and net inscriptions, <http://wiki.daimi.au.dk/cpntools-help/cpn_ml.wiki>