A Proposal Mechanism Against the Attacks: Cooperative Blackhole, Blackmail, Overflow and Selfish in Routing Protocol of Mobile Ad Hoc Network

Dr Karim KONATE and Abdourahime GAYE
Department of Mathematics and Computing
University Cheikh Anta DIOP, Dakar
{kkonate911, agaaye}@yahoo.fr

Abstract

The present work is dedicated to study attacks and countermeasure in MANET. After a short introduction to what MANETs are and network security we present a survey of various attacks in MANETs pertaining to fail routing protocols. We present the different tools used by these attacks and the mechanisms used by the secured routing protocols to counter them. We also study a mechanism of security, named the reputation, proposed for the MANETs and the protocol which implements it as well as its vulnerabilities. Our work ends with a proposal to fend off some of these attacks like Blackhole cooperative, Blackmail, Overflow, Selfish and an implementation of this solution on a compiler of C named Dev.-C++ in order to make comparative tests with the mechanisms already proposed.

Keywords: *Mobile Ad Hoc, Routing, Security, Attacks, Reputation, Blackhole cooperative, Blackmail, Overflow, Selfish*

1. Introduction

We have witnessed an exponential deployment of the spontaneous networks thanks to the emergence of new technologies wireless and, and also to the increasing availability of advanced and autonomous terminals (telephones, laptops…) [1]. An Ad hoc network constitutes a regrouping of a large population of portable calculating units (laptops, telephones…) inter-connected by a wireless technology, moving in an unspecified territory, forming a decentralized network, without fixed infrastructure.

This network is usually characterized by a dynamic topology, a limited bandwidth, energy constraints, the heterogeneity nodes, and a limited physical security. The applications having recourse to the ad hoc networks cover a very broad spectrum. For example in the tactical applications (fires, flood, etc.), in the soldier's field, in the monitoring systems, and the world of transport [1].

The problem of the MANET is how to find the investment of lower costs in rated capacities and reserves which ensures the routing of the nominal traffic and guarantees its reliability in the event of any breakdown of arc or node. That's why several families routing protocols emerged. Each protocol can be classified as a reactive like Ad hoc One Demand Distance Vector (AODV) and Dynamic Source Routing (DSR), proactive like Optimized Link State Protocol (OLSR), or hybrid like or Routing Protocol Zones (ZRP) [1].

In spite of the evolution ad hoc mobile networks during the last decade it still problems related security which remain unsolved. Although some solutions were proposed none of them can't satisfy all the constraints on the ad hoc networks.

2. Attacks in Routing Protocol of Mobile Ad Hoc Networks

An attack is an action which aims at compromising the security of the network. They are many and varied in these MANET.

Blackhole attack: consists in dropping some routing messages that node receives [1, 2, 3, 4, 5]. It was declined in several particularity alternatives, having different objectives, among which we can quote:

- Routing loop, which makes it possible for a node to create loops in the network;
- Gray hole, which lets pass only the packages of routing and diverts the data;
- Blackmail, which makes it possible for a node attacker to isolate another node.

Several solutions exist to counter these types of attacks, among which we name the technical estimate relation. In this mechanism the authors classified the relation between the nodes and their neighbors in three cases: Unknown (node X sent forever (received) of messages to (from) the node y and the probability of the malevolent behavior are very high), acquaintance (node X sent (received) some messages to (from) the node y and the probability of the malevolent behavior must be observed) and Friend (node X sent (received) in abundance of the messages to (from) the node y and the probability of the malevolent behavior is too small. This mechanism is implemented in the routing protocol RDSR (Relationship enhanced DSR protocol) [6].

The Threshold of sequence number consists in performing a check to find if RREP_seq_no is higher than the threshold value. The threshold value is dynamically updated in each interval of time. As the value of RREP_seq_no proves higher than the threshold value, one suspects the node to be malicious and adds it to the black list. This mechanism is implemented in the routing protocol named Detection, Prevention and Reactive AODV (DPRAODV) [21].

The Watchdog or monitoring (watchdog) is a solution which makes it possible to identify malicious nodes. The Watchdog assigns positive values with a node which successfully forwarded packages and a negative value after a threshold level of bad behavior was observed. It's implemented in the protocol called mobile Secure Watchdog for Ad hoc Network (SWAN) [14]. Pathrater which makes it possible the protocol to avoid nodes corrupted register in a black list [14].

The DRI or the data table of information's routing which is used to identify nodes of cooperative blackhole, it consists in adding two additional bits of information. These bits have as values 0 for "FALSE" and 1 for " TRUE " for intermediate nodes answering the RREQ of node source, AODV implements this mechanism [22, 23]. The Cross checking solution which consists in hoping on reliable node (nodes by which node source has forwarded the data) to transfer from the packets of data [22, 23].

The selfish attack: consists in not collaborating for the good performance of the network. We can identify two types of nodes which do not wish to take part in the network. Defective nodes i.e. do not work perfectly. Those which are malevolent, it is those which intentionally, try to tackle the system: attack on the integrity of the data, the availability of the services, the authenticity of the entities (denial-of-service, interception of messages, usurpation of identity, etc). Selfish nodes are entities economically rational whose objective is to maximize their benefit. To prevent the selfish nodes some solutions were proposed.

Among these we have a solution based on the Negative Selection Algorithm (NSA). It's based on the principles of the discrimination of self or no self in the immune system (to define it to oneself like a collection S of elements in a characteristic space X, a collection which needs to be supervised) [21].

The detection of anomaly aims at distinguishing a new model like part of self or no-self, given a model of system of self [21]. Structured Gene Activation (SGA) is a type of evolutionary algorithm which incorporates the redundant genetic material, which is controlled by a mechanism. It uses the multi-layer genomic structures for its chromosome i.e. all the genetic material (expressed or not) “is structured” in a hierarchical chromosome. The activation and deactivates mechanism these coded genes. This solution is implemented in AODV [21].

A solution based on the reputation named Collaborative Reputation (CORE) and Cooperation Of Nodes and Fairness In Dynamic Ad-hoc Network (CONFIDANT) which consists in collecting information on an old behavior of the tested entity by others [8, 9, 10]. A solution based on the payment (Nuglets) which requires with nodes which benefit from the resources of the network (transmitters and/or receivers) to pay "service providers" (intermediate nodes)[9,10] and a solution based on the localization (directional antennas).

Overflow routing tables: consists of malicious nodes to cause the overflow routing tables of nodes being used as relay [4]. To fend off this attack the named solution Trust evaluation was proposed. It's based on the evaluation of confidence to ensure a secure routing in MANETs. The success of a communication through a node will increase the index of confidence of this node and the failure by this node will decrease the index of confidence. If this value reaches zero this node is registered in a blacklist and we inform the other neighbors. Trust-based Routing Protocol (TRP) implements this solution [20].

3. Cooperative Mechanism

The basic mechanisms of security prove to be effectively ensured the traditional security functionalities which are the confidentiality, the integrity and above all the authentication. They thus ensure to prevent many attacks which disrupt the process of routing. On the other hand, they do not prove to be adapted to resolve the problem of the selfish nodes. Indeed, the cryptographic mechanisms, so effective they are don't ensure a node takes part in the process of routing by relaying all the packets.

However, in the context of the ad hoc networks, it's a primordial functionality as far as this type of network is based on the cooperation between the nodes. That's why some protocols aim at more specifically for the incitement to cooperate. Among these solutions, we set those which are based on a reputation nodes elaborated in the course of time according to the observations [1]. Among the protocols which are based on the reputation we can cite CORE which will be the subject of our contribution article.

3.1 An existing CORE Mechanism

The mechanism of Collaborative Reputation (CORE) [1, 9, 10, 11, and 14] is used to impose the cooperation of the nodes. In CORE each entity of the network encourages the collaboration of other entities by using metric cooperation called reputation. This metric is calculated while being based on the local data for each node and can be based optionally on the information provided by other nodes of the network implicated in the interchange messages with the supervised nodes.

This reputation is based on the analysis of the behavior (Watchdog) associated each node. A Boolean vector represents a good (with one 1) or a bad (with one 0) behavior. A punishment mechanism is adopted as solution to prevent a selfish behavior for gradually refusing the communication services to the entities which have bad behavior. This punishment is applied if the metric of reputation (Pathrater) reached a threshold and in this case we declare that the selfish nodes constitute a denial of service and they will be put in the blacklist. Thus the legitimate nodes (which cooperate) reach to save energy. Figure 1 illustrates the existing operation of CORE.

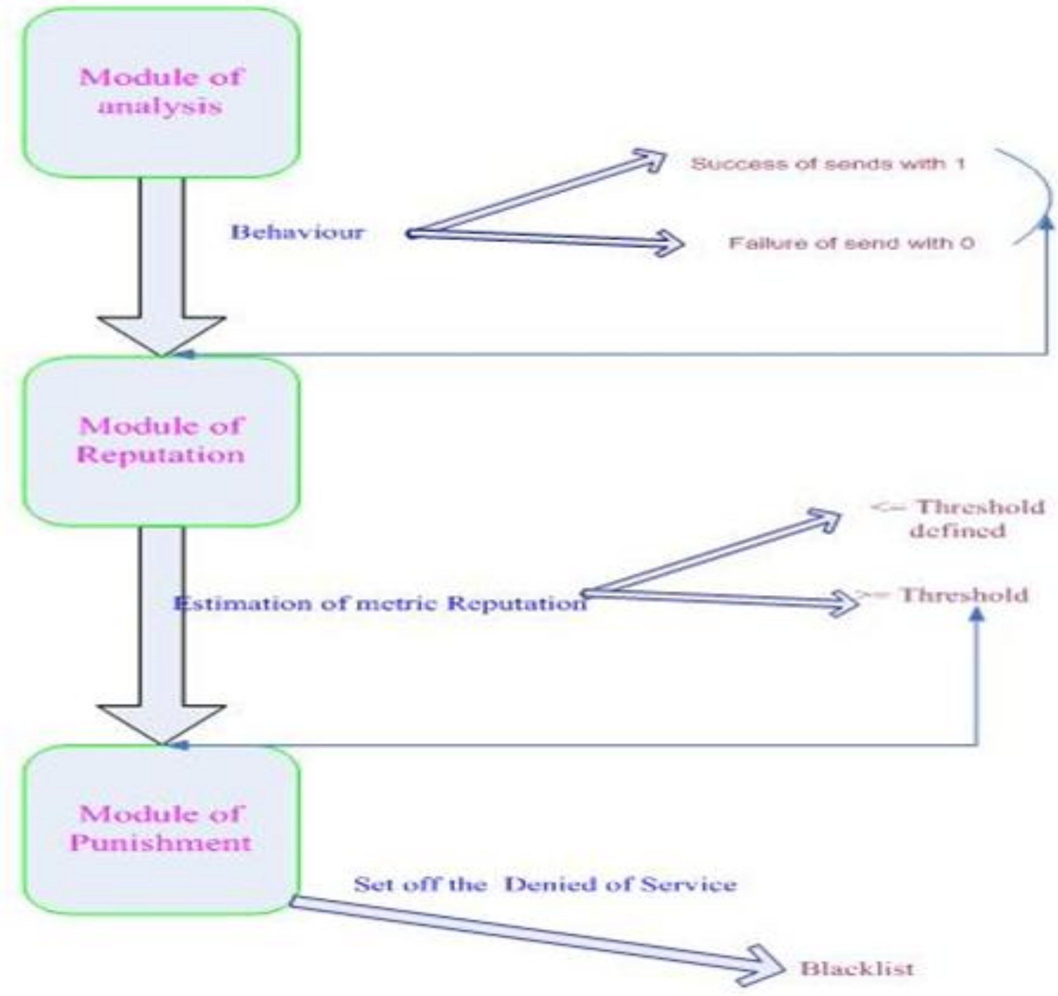


Figure 1: Existing Functioning of CORE

In [09, 11, 15, 16, 17] the authors use a mechanism called CORE to counter the selfish behavior (no cooperation) of the nodes of MANET. They are based on analytical modeling for using mathematical tools like the game theory to do the analysis of CORE. They are based on very simple models, called Dilemma of the Prisoner [9,11,15,16,17] which is a model analysis behaviors, to represent the conflict of interest which each node faces up to make at time a decision (in particular to forward a packet or not to cooperate).

3.2 Vulnerabilities of CORE

CORE suffers unfortunately from important defects. First, it doesn't really resolve the problem of selfish [1]. Immediately, all the selfish nodes see their packets systematically rejected and in this, the protocol is effective. But on the other hand, a quantity of data remains lost, reducing significantly the efficiency of the network. The protocol is based on assumptions (secure routing, single and nonusurpable addresses) which still remain to make a reality. It's a common disadvantage to all the reputation protocols. Indeed, this one is based on the information observed for the nodes and consequently requires an authentication mechanism in order to affect the marks to the legitimate which could store nonexistent links thus causing the Overflow attack [1].

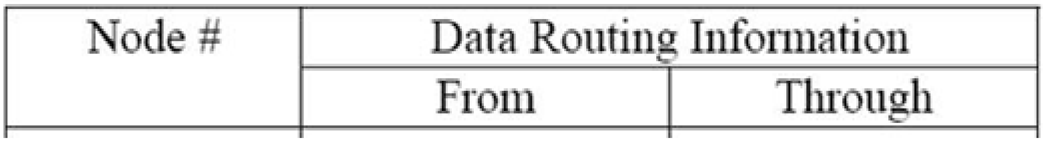
In addition, it's difficult to avoid the fictitious denunciation (Blackmail) [1] in which a malicious node generates false messages to put up the legitimate nodes on the blacklist. The mechanism of the reputation is potentially vulnerable face up to the cooperative nodes (Blackhole Cooperative) [1] which agree between them to assign good marks and to allocate in the other hand, bad marks the legitimate nodes.

Moreover, in that case the nodes couldn't make the distinction between the useful and the useless messages, and will be obliged to forward all the messages which come through them for having their good reputation. This could generate a waste of energy (sleep deprivation) [10, 12] and moreover the constant monitoring nodes would engender a network overload causing a reduction in the bandwidth.

In our algorithm we try to counter the four vulnerabilities cited for endowing CORE with a mechanism called DRI table [22, 23].

3.3 Operation of DRI Table

The DRI or the data table of routing information which is used to identify nodes of cooperative black hole, it consists in adding two additional bits of information. These bits have as values 0 for "FALSE" and 1 for" TRUE "for intermediate nodes answering the RREQ of node source. Each node updates an additional table of information of data routing (DRI) [22, 23]. The following figure represents the structure of the table.

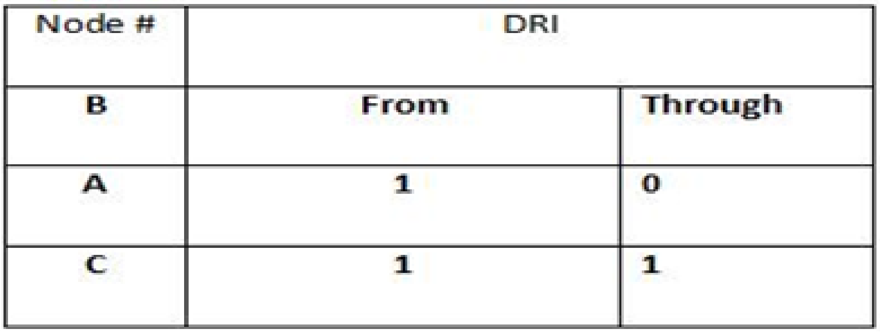


Node #	Data Routing Information	
	From	Through

Figure 2: The Structure of the DRI Table

In the DRI table, the first bit named “From” represents the information on the packet of the node data routing (the node from which the packets comes) while the second bit “Through” represents the information on the packet by the node of data routing (the node through which its forwards the packets). For example the entry “1, 0” for node A means that the node B forwards the packets data coming from A but it doesn't forward any packet of data through A. The entry “1, 1” for the node C means that the node B forwards the packets data coming from C and the packets of data through C. This example is represented in table 1.

Table 1: Example of DRI Table Utilization



Node #	DRI	
B	From	Through
A	1	0
C	1	1

To discover a route towards the destination node the source node (SN) broadcasts a RREQ message. The intermediate node (IN) which produces a RREP must provide the hop of the next node (NHN) and its DRI entry. According to the RREP message from the intermediate node, the source node will control its own DRI table to see if the intermediate node will a trustworthy node. If the source node used IN before the new route discovery for routing the data, then IN is a reliable node and the source node begins to forward data towards IN. This obliges the attacking nodes to cooperate and to relay messages until the destination to appear in the DRI of its neighbor. This solution can be also adapted to counter the attacks like Overflow, Blackmail and also Selfish.

4. A Proposal Solution Against the Attacks: Cooperative Blackhole, Blackmail, Overflow and Selfish

The Reputation and Punishment concepts, or Payment, can encourage the nodes to fully play their role not to lose their good behavior but these solutions cannot counter some attacks in MANETs as the above attacks. That's why in our algorithm we try to counter the vulnerabilities quoted for endowing CORE with a mechanism named DRI table [22, 23].

4.1 Description of XCORE

In the existing CORE, we include DRI table and we estimate the table if we receive a routing packet. To making this estimation, we calculate the times that the node has forwarded the packets coming from another node and the times that the node has forwarded the packets through another node. If the Rate_Send_Reception rate of the DRI is equal to [0, 0] we declare that this link is fictitious (it's an Overflow attack). Else when a node sends a routing message, we estimate this message. If it's a route error, we will check its validity by looking at the DRI. If Rate_send_Reception is [0, 0] then we confirm that it's a defective node else we consider that it's an invalid message (if it is a Blackmail attack) and in this case we continue to estimate the reputation. If the reputation is < 0 we consider that it's a denied of service node (a Selfish node) else we declare that it's a cooperating node.

4.2 A Proposal Mechanism: XCORE

Figure 3 illustrates the operation of XCORE proposed.

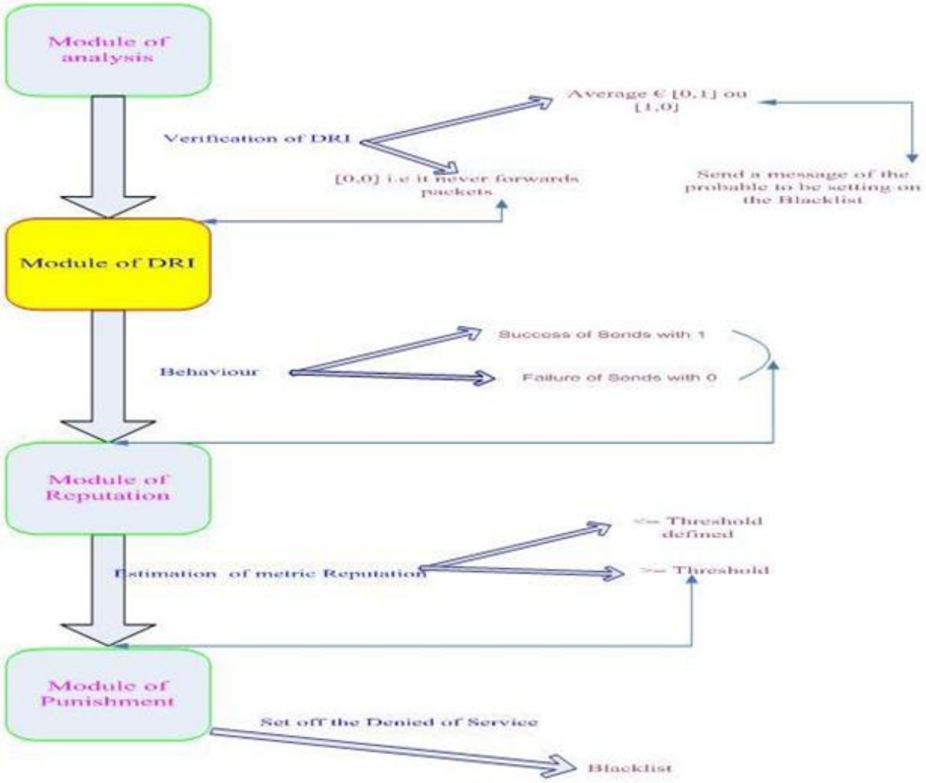


Figure 3: Functioning of XCORE

4.3 Algorithm of XCORE

Begin

- Verification of DRI before transmission;
- If Rate_send_Reception is equal to [0, 0] then;
- We put the node on the blacklist because it's a fictitious link;
- Else when a node sends a route message, we estimate the message;
- If it's a route error, we will check its validity by looking the DRI;
- If the Rate_send_Reception is equal to [0, 0] then we confirm that this node is defective;
- Else we considered that this message is invalid (it's a Blackmail attack);
- Else it cooperates for the first iteration and it sends the message by monitoring the node;

- In each iteration of period T, it observes the behavior of the opposing node and it builds a vector V= (V1, V2, …. VT) which element Vi is shown by 1 for a good behavior and 0 for a bad behavior;
- To assess the reputation during this period;
- Reputation= (1/T) * sum of Vi;
- If Reputation $\succ0$ then the node is cooperating node;
- Else the node is a denied of service node.

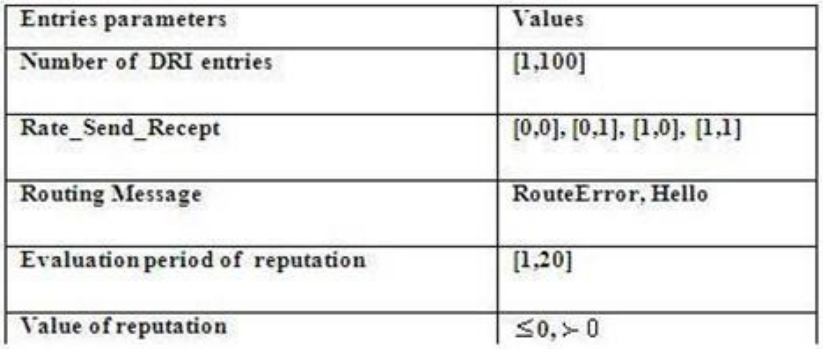
End

The absence of simulators which take into account the protocol CORE and also the complexity of protocol CORE the majority of the authors use other means like the MATLAB software to make their CORE simulations [9].

5. Implementation and Test of CORE and XCORE

For the tests we made our program on the C software named Dev.-C++. To test our proposal, we gave in entry the iteration count of the DRI, If the Rate_send_Reception rate of the DRI is equal to [0, 0] we declare that this link is fictitious (it is a Overflow attack). Else when a node sends a route message, we estimate this message. If it's a route error, we will check its validity by looking at the DRI. If Rate_send_Reception is [0, 0] then we confirm that it's a defective node else we consider that it's an invalid message (if it is a Blackmail attack) and in this case we continue to estimate the reputation. If the reputation is < 0 we consider that it's a denied of service node (a Selfish node) else we declare that it's a cooperating node. The evaluation parameters are represented in table 2.

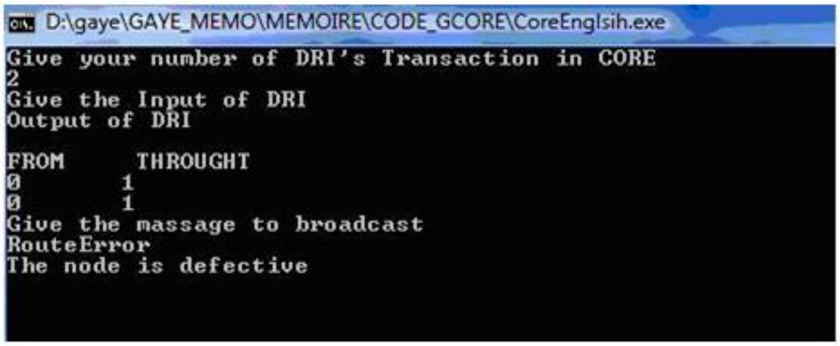
Table 2: The Test Parameters of CORE and XCORE



Entries parameters	Values
Number of DRI entries	[1,100]
Rate_Send_Recept	[0,0], [0,1], [1,0], [1,1]
Routing Message	RouteError, Hello
Evaluation period of reputation	[1,20]
Value of reputation	≤0, ≻ 0

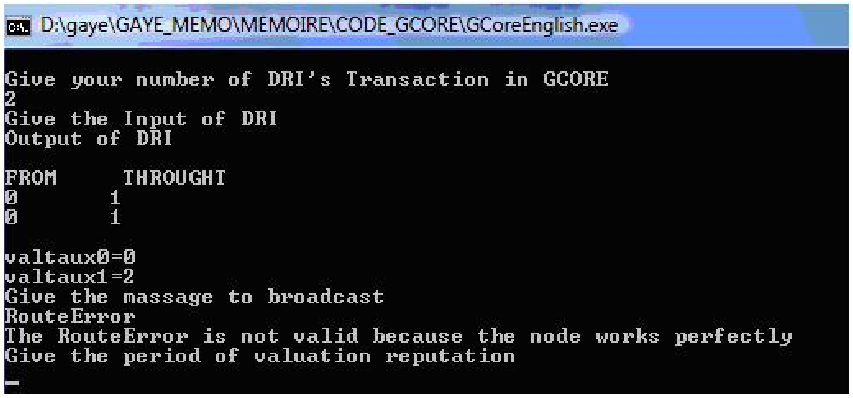
5.1 The Comparative Tests of Attacks on CORE and XCORE

Blackmail attack on CORE and XCORE: in these tests we estimate the Blackmail attack. If we receive a “route error” message, we look at the DRI table, if the rate is in [0, 0], we consider that this node is indeed defective, else we consider that this message was sent by an attacking node and we reject this message i.e. we will not let the Blackmail attack passed. These tests are illustrated by figures 4 and 5.



```
Give your number of DRI's Transaction in CORE
2
Give the Input of DRI
Output of DRI

FROM     THROUGHT
0       1
0       1
Give the massage to broadcast
RouteError
The node is defective
```

Figure 4: Blackmail on CORE

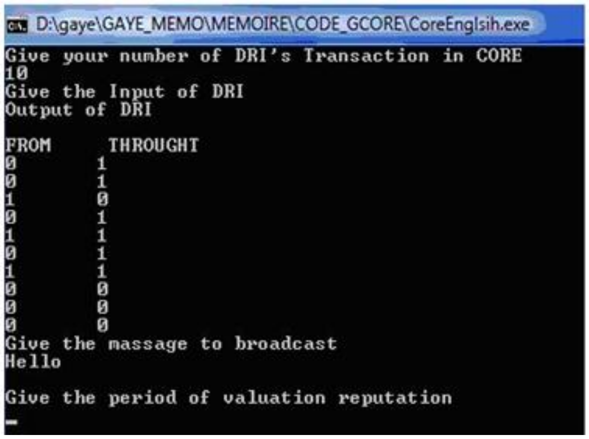


```
Give your number of DRI's Transaction in GCORE
2
Give the Input of DRI
Output of DRI

FROM     THROUGHT
0       1
0       1

valtaux0=0
valtaux1=2
Give the massage to broadcast
RouteError
The RouteError is not valid because the node works perfectly
Give the period of valuation reputation
_
```

Figure 5: Blackmail on XCORE

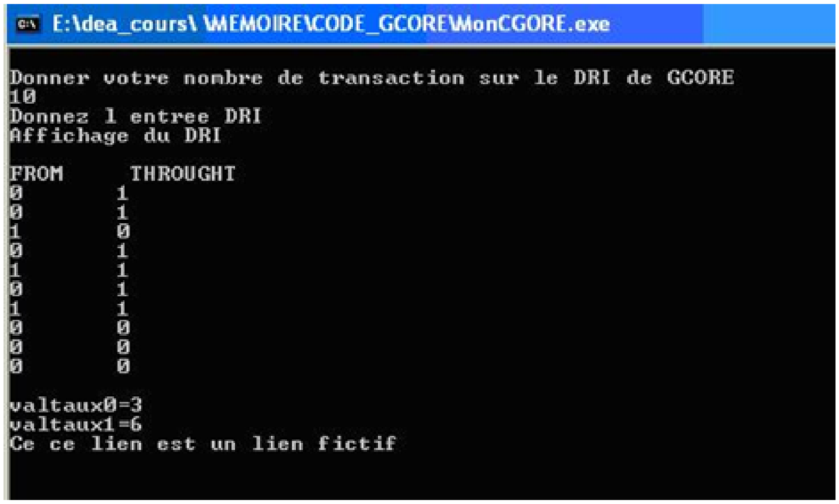
These tests show indeed that if an attacking node tries to make a Blackmail attack on a legitimate node, the latest is fended off with our model (figure 4) while on the existing model the attack isn't blocked (figure 5).

Overflow attack on CORE and XCORE: to test our proposal, we gave in entry the iteration count of the DRI, if the Rate_send_Reception rate of the DRI is equal to [0, 0] we declare that this link is fictitious i.e. an attacking node stored nonexistent links to implement the Overflow attack.



```
Give your number of DRI's Transaction in CORE
10
Give the Input of DRI
Output of DRI

FROM     THROUGHT
0       1
0       1
1       0
0       1
1       1
0       1
1       1
0       0
0       0
0       0
Give the massage to broadcast
Hello

Give the period of valuation reputation
_
```

Figure 6: Overflow on CORE

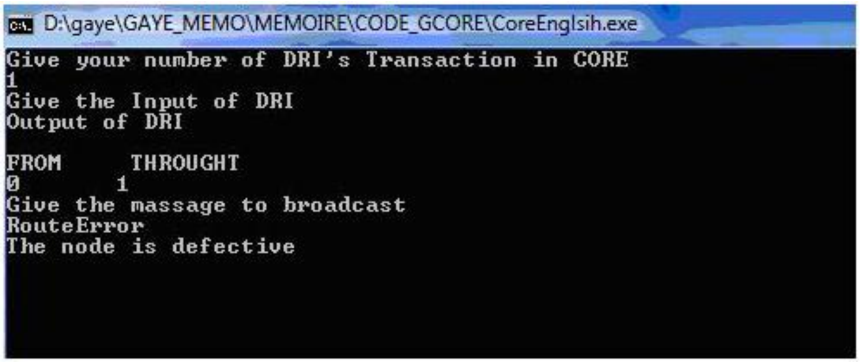


```
Donner votre nombre de transaction sur le DRI de GCORE
10
Donnez 1 entree DRI
Affichage du DRI

FROM     THROUGHT
0        1
0        1
1        0
0        1
1        1
0        1
1        1
0        0
0        0
0        0

valtaux0=3
valtaux1=6
Ce ce lien est un lien fictif
```

Figure 7: Overflow on XCORE

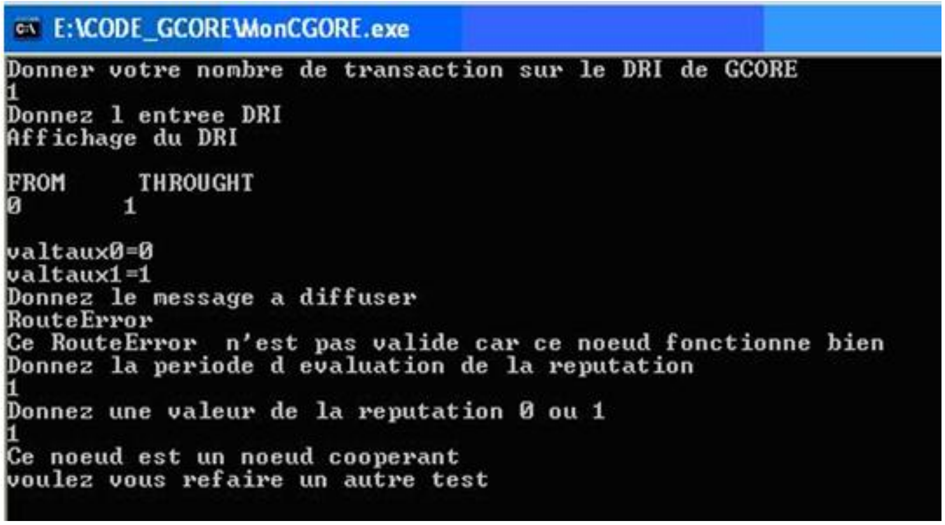
The tests above show the making Overflow attack by an attacking node which stored fictitious links. The attack is countered by the model of XCORE (figure 7) whereas this attack passes in CORE (figure 6).

Selfish attack on CORE and XCORE: this test enables us to see whether a node is selfishious or not. That is based on the reputation evaluation. A node can receive a false “route error” message and it will remove this link in its table whereas this node works perfectly. If we receive a message we estimate if it is valid or not and if it is valid, we estimate the reputation now. If the reputation is < 0 we consider that this node is a denied node (a Selfish node) else we declare that this node is a cooperating node.



```
Give your number of DRI's Transaction in CORE
1
Give the Input of DRI
Output of DRI

FROM     THROUGHT
0        1
Give the massage to broadcast
RouteError
The node is defective
```

Figure 8: Selfish attack of CORE



```
Donner votre nombre de transaction sur le DRI de GCORE
1
Donnez 1 entree DRI
Affichage du DRI

FROM      THROUGHT
0        1

valtaux0=0
valtaux1=1
Donnez le message a diffuser
RouteError
Ce RouteError  n'est pas valide car ce noeud fonctionne bien
Donnez la periode d evaluation de la reputation
1
Donnez une valeur de la reputation 0 ou 1
1
Ce noeud est un noeud cooperant
voulez vous refaire un autre test
```

Figure 9: Selfish attack on XCORE

These tests show that if an attacking node tries to eliminate a legitimate node i.e. a node which cooperates for the perfect work of the network, this attack is detected in XCORE (figure 11) by estimating its reputation whereas CORE (figure 10) lets pass this attack.

6. Conclusion

In our work we have presented the specificities of the MANET as well as the problems of the security routing protocols in these types of network. We presented some attacks met in MANETs, their functioning mode thus the mechanisms used and the protocols which implement them to counter these attacks. We analyzed the functioning mode of CORE and brought out some of its vulnerabilities, and then we proposed a new algorithm, named XCORE, which improves the basic CORE. This algorithm ensures to resist the attacks Blackhole cooperative, Blackmail, Overflow and Selfish. We implemented CORE and XCORE in Dev.-C++ in order to carry out comparative tests and these tests show that the above mentioned attacks do not pass any more with XCORE that it's validates our proposed solution.

References

[1] Wiley John: Security for Wireless ad hoc networks. Eyrolles, book 2007, pages 247.

[2] ADJIDO Idjiwa,BENAMARA Radhouane,BENZIMRA Rebecca,GIRAUD Laurent: Protocol of secure routing ad hoc in a clusterized architecture. University Pierre and Marie Curia(Paris VI),FRANCE,November 2005,pages 4.

[3] Curtmola Reza. Security of Routing Protocols in MANET. 600.647-Advanced Topics in Wireless Networks,February 2007,pages 26.

[4] Bing Wu,Jianmin Chen,Jie Wu,Mihaela Cardei. A Survey of Attacks and Countermeasures in MANET. Department of Computer Science and Engineering Florida Atlantic University,Decembre 2005

[5] Chen Ruiliang,Snow Michael,Park Jung-Min,M. Refaei Tamer,Eltoweissy Mohamed. Defense against Routing Disruption Denial-of-Service Attacks in MANET. Department of Electrical and Computer Engineering Virginia Polytechnic Institute and State University Blacksburg,VA,USA,November 2005,pages 15.

[6] A.Rajaram,Dr. S. Palaniswami. The Trust-Based MAC-Layer Security Protocol for Mobile Ad hoc Networks.(IJCSE) International Journal on Computer Science and Engineering Vol.02,No.02,2010,400-408. Anna University Coimbatore,India,March 2010,pages 9.

[7] T.V.P.Sundararajan et Dr.A.Shanmugam. Behavior Based Anomaly Detection Technique to Mitigate the Routing Misbehavior in MANET. Sathyamangalm-638401,Tamilnadu,India,May 2009,pages 14.

[8] Kevin Hoffman,David Zage,and Cristina Nita-Rotaru. A Survey of Attack and Defense Techniques for Reputation Systems. Department of Computer Science and CERIAS Purdue University. April 2008,pages 19.

[9] Pietro Michiardi: Cooperation in the ad hoc networks: Application of the evolution and game theory within the framework of imperfect observability. Institute Eurecom 2229,road of the Peaks LP 19306904 Sophia-Antipolis,France,July 2006,pages 17.

[10] Michiardi Pietro and Molva Refik: CORE: A Collaborative Reputation Mechanism to enforce node cooperation in MANET. European Wireless Conference, November 2003,pages 15.

[11] Hu Jiangyi: Cooperation in Mobile Ad Hoc Networks. Computer Science Department Florida State University,January 11,2005,pages 23.

[12] Buttyan Levente and Hubaux Jean-Pierre: Nuglets: a virtual Currency to Stimule Cooperation in Self-Organized Mobile Ad Hoc Networks. Institute for Computer Communications and Applications Department of Communication Systems Swiss Federal Institute of Technology Lausanne,18 January 2001,pages 15.

[13] Yan Zheng, Zhang Peng,Virtanen Teemupekka. Trust Evaluation Based Security Solution in Ad Hoc Networks. Helsinki University of Technology,Finland, December 2003, pages 14.

[14] Xue Xiaoyun. Security mechanisms for ad hoc routing protocols. Computer Science and Network Department,ENST,thesis September 2006,pages 234.

[15] Pietro Michiardi and Refik Molva. Analysis of Coalition Formation and Cooperation Strategies in MANET. Institut Eurecom May 2004,pages 28.

[16] Levente Buttyan and Jean-Pierre Hubaux. Report on a Working Session on Security in Wireless Ad Hoc Networks. Laboratory for Computer Communications and Applications Swiss Federal Institute of Technology-Lausanne(EPFL),Switzerland,September 2002,pages 17.

[17] Pietro Michiardi - Refik Molva. Game theoretic analysis of security in mobile ad hoc networks. Institut Eurécom Research Report N°RR-02-070,juin 2002,pages 10.

[18] Hu Yih-Chun,Perrig Adrian,Johnson David B.: Packet Leashes: A Defense against Wormhole Attacks in Wireless Networks,INFOCOM 2003,pages 11.

[19] Emmanouil A. Panaousis, Levon Nazaryan, Christos Politis. Securing AODV Against Wormhole Attacks in Emergency MANET Multimedia Communications,Wireless Multimedia and Networking(WMN) Research Group Kingston University London. July 2009,pages 7.

[20] Shang-Ming Jen 1,Chi-Sung Laih 1 and Wen-Chung Kuo. A Hop-Count Analysis Scheme for Avoiding Wormhole Attacks in MANET.

[21] Payal N. Raj,Prashant B. Swadas. DPRAODV: A DYANAMIC LEARNING SYSTEM AGAINST BLACKHOLE ATTACK IN AODV BASED MANET,IJCSI International Journal of Computer Science Issues,Vol.2,Computer Engineering Department,SVMIT Bharuch,Gujarat,India,September 2009,pages 6.

[22] Ramaswamy Sanjay,Fu Huirong, Sreekantaradhya Manohar,Dixon John and Nygard Kendall: Prevention of Cooperative BlackHole Attack in MANET. Department of Computer Science,IACC 258 North Dakota State University,Fargo,ND 58105,March 2003,pages 7.

[23] Hesiri Weerasinghe and Huirong Fu. Preventing Cooperative BlackHole Attacks in Mobile Ad Hoc Networks: Simulation Implementation and Evaluation;International Journal of Software Engineering and Its Application Vol.2,No.3. Oakland University Rochester MI 48309 USA,June 2008,page 16.