SDV: a new approach to secure distance vector routing protocols

Abdelaziz Babakhouya, Yacine Challal, Abdelmadjid Bouabdallah, Saïd Gharout

SDV: A new approach to Secure Distance Vector routing protocols

A. Babakhouya, Y. Challal, M. Bouabdallah, and S. Gharout

***Abstract*— The DV (Distance Vector) routing protocols (e.g., RIP [14]) have been widely used in the Internet. These protocols are vulnerable to a variety of attacks since they were designed without security aware. A major threat against these protocols is that a malicious router can interrupt routing operation by sending erroneous routing update. In this paper, we propose a new approach called S-DV to Secure Distance Vector Routing Protocols. In our approach, we designate some trusted routers which collaborate to detect malicious routing update with short or long distance fraud. These routers maintain also a security metric witch is used to forward data traffic through a secure route. Through our threat analysis and comparison, we show that S-DV offers a deterministic detection of malicious routing updates with reduced overhead compared to S-RIP [28].**

***Index Terms*— Security, Routing protocols, Authentication, Coherence Checking.**

I. Introduction

As all Internet-based applications rely on a dependable packet delivery service provided by the Internet routing protocols, secure routing protocols become of critical importance [21]. Routing protocols have been designed to dynamically maintain route between any pair of communicating entities in spite of changes in network topology. Consequently routing faults can jeopardize the reliability of critical applications of the Internet [4] [3] and a single malicious router can completely disrupt the routing protocols and cause a disaster.

Several DV (Distance Vector) routing protocols are used today: Routing Information Protocol (RIP) [14] is a popular example of distance vector routing protocols which is widely used in IP networks of moderate size. Distance vector routing protocols are also adapted to be used for routing within wireless ad hoc networks. DSDV [19] and AODV [18] are examples of these routing protocols.

A distance vector routing protocol finds shortest paths between nodes in the network through a distributed implementation of the classical Bellman-Ford algorithm [2]. To enable packets forwarding, each router maintains a routing table providing the distance from itself to all possible destinations within the network. A routing table entry consists of a destination IP address, the distance (usually in number of hops) and the next hop router in the path to this destination. To maintain the routing table, each router periodically transmits a routing update to each of its neighbor routers, containing its ***shortest distance to each destination***. Each node uses this information advertised by its neighbors to update its own routing table, so that its route for each destination uses as a next hop the neighbor that claimed the shortest distance to that destination.

A Distance Vector routing protocol is easy to implement and requires low resource consumption. However, it is less robust than a link state [16] routing protocol because each router has only a partial connectivity information which is the output of a (potentially faulty or malicious) neighbor router. For this reason, Perlman argues in [20] that distance vector routing protocols are poor candidates for fault detection than link state routing protocols. A major threat they are faced on is that one malicious router can interrupt routing operation by sending erroneous routing update. These erroneous routing updates are usually generated from two different entities: external and internal attackers. External attackers can inject erroneous routing messages, replay previous routing messages, or modify a valid routing message. As a result, an injected erroneous routing update would propagate throughout the network, and it might remain in use for arbitrarily long periods of time, thus deceiving more than one router. However, internal attackers can usually cause more severe damages. These are routers that have been trusted in some point of time but are not committed to their initial promises anymore or have been compromised by external attackers. These routers can also send erroneous routing update to their neighbor routers and modify their local view of the network topology to isolate them or pass their traffic through special routes. Usually, it is much harder to identify the internal attackers, since they already have some sort of credentials that everybody trusts.

To secure routing protocols, we require some security services such as: integrity, freshness, authentication, authorization and consistency of routing messages. Previous works [1, 15, 10] use public-key digital signatures or MAC (Authentication Message Code) to prevent an external attacker from modifying, deleting, or adding routing messages exchanged between routers. However, even with a digital

A. Babakhouya was with Department of Computer Science, University of Béjaïa, Algeria, while doing this work. He is now with CERIST Center of research, Algiers, Algeria. (e-mail: babakhouya@ mail.cerist.dz).

Y. Challal, and A. Bouabdallah are with Heudiasyc lab. UMR CNRS 6599, UTC, Compiègne. France. (e-mails: ychallal@hds.utc.fr, bouabdal@hds.utc.fr).

S. Gharout is with Department of Computer Science, University of Béjaïa, Algeria. (e-mail: gharout@ gmail.com).

signature or MAC, a legitimate router or a compromised legitimate router may still take incorrect actions such as advertising addresses that it does not own or reporting short or long distance information. In order to decide whether a routing update received from a legitimate neighbor router is correct or not, routers need to have information regarding the network topology beyond the immediate neighbors. Unfortunately, DV routers do not have this information. Thus, it is necessary to implement a mechanism which provides consistency check of the distance vector advertised by a legitimate router.

In this paper, we present a new approach called S-DV (Secure Distance Vector routing protocols) which provides both protections from internal and external attackers. The main idea is to designate some trusted routers, which we called S-DV routers that collaborate in consistency checking of routing update messages. This is an alternative mechanism, less expensive that offers a deterministic detection of distance fraud than the approach proposed in S-RIP [28]. Moreover, S-DV routers use a new metric that we designate by *Security Indicator*, to prefer a choice of a secure route than a shortest one which has been subject to frequent attacks.

The remainder of the paper is organized as follows. Section II presents DV routing protocols vulnerabilities. We review the related work in Section III. We present generality and details of our approach in Sections IV and V. Sections VI discusses threat analysis. We measure the overhead of our approach and we give comparison between S-DV and S-RIP in Section VII. We end up our paper with some conclusions in Section VIII.

II. Distance vector routing protocols' vulnerabilities

In DV routing protocols, as in most routing protocols, nodes (routers) exchange routing messages about their neighborhood and construct a virtual view of the network topology so that they can route the data packets to the desired destinations. Such routing messages could be target of any malicious adversary who intends to disrupt functionality of the network. It has been noted long ago that abusing routing protocols may be the easiest way for launching attacks [21]. Since a routing update message contains a vector of pairs (destination distance), DV routing protocols are vulnerable to:

A. Router impersonation:

Called also router masquerading. This attack occurs when an external entity successfully imitates legitimate router's identity (this can be accomplished using the IP spoofing) to generate, modify or replay routing messages exchanged between legitimate routers. This is due to the lack of mechanisms which provide authentication, integrity and freshness of routing messages exchanged between neighbor routers. However, a legitimate router can also generate an authentic routing update which contains erroneous routes such as: claiming to be directly attached to a sub-net it does not own (Prefix impersonation) or advertising a short or a long distance than the real distance to some destination in the network (Distance fraud).

B. Prefix Impersonation

A malicious or a compromised router can claim a zero distance to a non-directly connected subnet (prefix) or a nonexistent one. Without authorization mechanisms, the neighbor router which receives this routing update is unable to detect such malicious behaviors.

C. Distance fraud

A malicious or compromised router can claim a distance shorter or longer than the real distance to a specific destination. A short distance fraud is used to attract traffic to some destinations. This is commonly called: black hole attack. Whereas, long distance fraud can avoid traffic and preserve its resources and energy. This is due to the lack of mechanisms which provide consistency check of routing update.

In order to counter these vulnerabilities, it is necessary to propose a new approach which provides authentication, integrity, freshness, authorization and consistency check of a routing update. Current version of RIP [14] uses only a clear-text password for authenticating peers. This is vulnerable to a traffic analysis. An external entity can sniff the password and overcome the network. Thereafter, Kayed-MD5 [BAL97] has been proposed to replace clear-text password authentication. This mechanism uses a MAC (see section IV.B) to provide authentication, integrity and freshness of a routing updates. However, a MAC does not provide any guarantee on the coherence of routing updates. For example, a malicious router can advertise an authentic routing update (with a valid MAC) which contains erroneous routes. In the following section, we present the different solutions proposed in the literature which deal with consistency check to secure distance vector routing protocols.

III. Related work

Several works have been done to secure Internet routing protocols (RIP [14], OSPF [16] and BGP [25]). [6, 21, 23] survey these efforts and give a comparison between them. The actual works accepted as standards [1, 15, 10] use public-key digital signatures or MAC to provide authenticity, freshness and integrity of routing messages. These mechanisms are efficient to protect routing messages from external attacks. However, they do not protect routing messages against malicious intermediate routers as well as compromised routers.

In this paper we focus our study on the approaches proposed in the literature that protect Distance Vector routing protocols from external and internal attackers. We can classify these approaches in two categories: Semantic approaches and cryptographic approaches (see Fig. 1).

A. Semantic Approaches

In semantic approaches, the solutions proposed in the literature [17, 22] use the semantic of the protocols to detect routing messages anomalies.

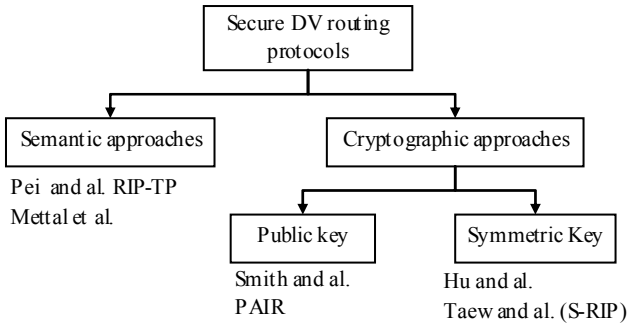

Fig 1. Secure Distance Vector routing protocols approach

Mittal et al. [17] detect faults in a RIP network using sensors which are placed on some (or all) of the links and each sensor is given the whole network topology as well as the positions of all other sensors. A sensor computes all the possible paths from each router to each subnet by essentially running a link state protocol on the manually configured topology. A sensor then analyzes the routing updates on its links and the updates' semantics (i.e. distances) are checked against the sensor's set of all possible distances. If a distance is not in the legitimate range, an alarm is raised. Otherwise, a query is sent to all the sensors along the possible path(s) that have this distance in order to verify the distance. This has major drawbacks for practical deployment since it implicitly requires static network topology, static sensor placements, and each sensor has to compute all the possible paths for each router to each destination.

RIP with Triangle theorem checking and Probing messages (RIP-TIP) [22] uses the routing update to check a simple triangle theorem. The theorem states that given a set of 3 nodes in a shortest path protocol, the distance between one pair of nodes should be always less than the sum of the distance of the other two pairs. However, message losses or update message delays may cause a temporary violation of the triangle theorem. To distinguish temporary delays from faults, probing messages are sent to the destination to verify the suspicious routing update. One disadvantage is that probing messages may be manipulated. A node advertising an invalid route can convince a receiver that a route is valid by manipulating the TTL value in a probing message or sending back an ICMP message (port unreachable) on behalf of the destination.

B. Cryptographic approaches

Several solutions which use cryptographic mechanisms to detect malicious routing messages have been proposed in the literature. These solutions can be classified in two categories: public-key approaches and symmetric key approaches.

Smith et al. [26] add a new attribute called the predecessor (second router before the destination) and message a sequence number to route updates. This predecessor and the originating UPDATE sequence number are signed by the private key of the origin router. Given the signed predecessor, one can then use a loop free path finding algorithm [8] to reconstruct and verify the route to the destination. This approach protects a route update from distance fraud by the intermediate routers. Unfortunately, generation and recursive verification of digital signatures are very resource and time consuming. In addition, this approach can not prevent from predecessor fraud. Manimaran and al [7] discuss these drawbacks and propose an improvement of this mechanism using Pivot Based Algorithm for Inconsistency Recovery (PAIR).

Hu and Perrig [12] use efficient cryptographic mechanisms, including one-way hash chains and authentication trees for authenticating sequence numbers and distances of advertised routes. This mechanism can prevent from short distance fraud but it does not prevent from long distance fraud.

S-RIP [28] can prevent from short and long distance fraud using request/reply messages to confirm the consistency of an advertised route. In S-RIP, a router confirms the consistency of an advertised route with those routers that have propagated that route. To support this mechanism, S-RIP adds a next hop attribute to a routing update exchanged between neighbor routers. A reputation-based framework is proposed for determining how many routers should be consulted to flexibly balancing security and efficiency. The major drawback of this mechanism is the non deterministic detection of faulty routes. Indeed, a malicious advertised route may be accepted if routers who forward this route are colluding. In addition to this, S-RIP generates an important overhead due to consistency check massages transmission in the network.

All these solutions offer a better security level but they are expensive in terms of control messages overhead, CPU time in generation and verification of signatures and consumption of memory resources in routers. Besides, these approaches do not maintain any history of detected malicious routing updates. In our approach, S-DV routers maintain neighbor routers behavior metric to prefer a choice of a secure route than a shortest one which has been subject to frequent attacks. Moreover, we introduced an efficient Distance Request Distance Reply (DR) mechanism for route consistency checking. Our DR mechanism reduces considerably the number of messages sent in the network to check route consistency compared to S-RIP. Besides, our approach offers a deterministic detection of distance fraud as we will see in the subsequent sections.

IV. S-DV GENERALITY

To prevent Distance Vector routing vulnerabilities, we propose a new approach which we call S-DV (Secure Distance Vector routing protocols). Our approach reduces considerably the overhead generated by route consistency checking messages and is deterministic with respect to malicious routing updates detection. The main idea is to designate some trusted routers (nodes), which we call S-DV routers, which collaborate to provide the consistency checking through our DR (Distance Request and Distance Reply) mechanism. These routers reject malicious routing update advertised by one of their neighbors and maintain variables

which measure the *Security Indicator* associated to a route which passes through this neighbor. These variables are used as a metric to prefer a choice of a secure route than another. We use a symmetric key cryptography to authenticate routing information exchanged between neighbor routers. Symmetric key cryptography is also used to authenticate DR messages exchanged between S-DV routers.

In this section, we present how our approach can prevent from Distance Vector routing vulnerabilities. Firstly, we present our assumptions and give some definition of the basic concepts of our approach. In the following sections, the word node is used to designate a router and the word prefix to indicate the IP address of a sub-net.

A. Assumptions:

As any other secure routing protocol, our approach requires the existence of key establishment mechanisms such as: pair-wise shared key or a Public Key Infrastructure (PKI) [11]. Other key establishment mechanisms can be used. For simplicity we assume that:

*(A1) Every node of the network shares a different secret key with every neighbor node***.** This secret key is used to authenticate routing messages exchanged between each pair of neighbor nodes.

*(A2) Every S-DV router shares a different secret key with every other S-DV router in the network***.** This second secret key is used to authenticate special messages (DR messages) exchanged between each pair of S-DV nodes.

These assumptions are less strong than those of S-RIP [28] where the authors suppose that every node of the network shares a different secret key with every other node of the network. This increases the number of keys to maintain in each node and complicates the key management mechanism.

To check if a node is authorized to advertise a route to a sub-net (prefix) which claims to be directly attached to, we suppose that:

(A3): Every node of the network knows which prefixes are directly attached to every one of its neighbor nodes. Such knowledge, router-prefix mapping, are securely distributed to each router, e.g., it can be pre-configured on each router since in an AS (Autonomous System) network configurations are administratively controlled by a single authority. We note that this assumption is less strong than S-RIP [28] assumption where the authors suppose that each node of the network knows which prefixes are attached to every other node of the network.

Finally, we propose to add a new attribute to Distance Vector routing update messages which we call ***predecessor***. This attribute is used to support consistency check mechanism of an advertised route. We define this attribute as follow:

Definition 1: the predecessor of a route is the last S-DV router that advertises or forwards this route.

Thus, an advertised route contains three fields (destination, distance and predecessor) which are maintained by each node in its routing table. More discussions about the update of this field are given in section (IV.D).

B. Preventing router impersonation

We use a shared secret key authentication mechanism to authenticate routing messages exchanged between neighbor nodes. In this mechanism, when a node v_i sends routing message to its neighbor v_j, it adds:

1) A sequence number to protect against replay of previous routing massages,
2) MAC (Message Authentication Code) which is the result of a hash function, e.g. Keyed MD5 [24] applied to the routing message, the sequence number and the shared secret key (A1) with the recipient neighbour.

When v_i receives the routing message from its neighbor v_j it verifies the MAC of the received message. It also checks the sequence number. If all these checks are valid, this routing message is accepted. Otherwise, it is rejected.

With this mechanism, it is more difficult for an unauthorized node to impersonate a legitimate node if its keying materials are not disclosed. This is why, [5] argue that the efficiency of data origin authentication mechanisms rely on the key management mechanism in use.

After checking the authenticity of a routing message, we need also to check the consistency of a prefix and the distance for each advertised route in this routing message.

C. Preventing Prefix impersonation

Several solutions have been proposed in the literature to prevent against prefix impersonation. Especially in BGP [25] where the problem becomes more difficult since its administration is distributed between several autonomous systems. These solutions require a public key infrastructure PKI [11] which distributes the certificates (a signature which binds the prefix to routers attached to). Among other solutions using this concept, one can site: Address Attestation in S-BGP [13] and the certificates of authorization in soBGP [27]. In RIP [14], prefix impersonation occurs when a node v_i receives from its neighbor v_j an advertised route to a prefix with a distance equal to **zero,** v_i checks if the announced prefix belongs to this neighbour or not using router-prefix knowledge (A3). Depending on this verification, it decides whether to accept or reject the advertised route.

D. DR mechanism to prevent Distance fraud

In Distance Vector routing protocols, distance frauds are difficult to prevent since routers have no information regarding the network topology beyond the immediate neighbors and routing updates received by a node are computational results of short path computations by other nodes. In Fig. 2, we illustrate how to use the DR (Distance Request Distance Reply) mechanism to check the consistency of a distance advertisement.

Fig. 2 shows the DV routing update sent from v_j to v_i having 3 fields: [*dest,* $D(v_i, v_j)$, $pred(v_i, v_j)$] to designate respectively: the destination, distance and predecessor of this route. Whenever v_i receives this routing update, it checks its consistency by sending Distance Request message to the predecessor v_k (cf Definition 1) of this route, queering its local

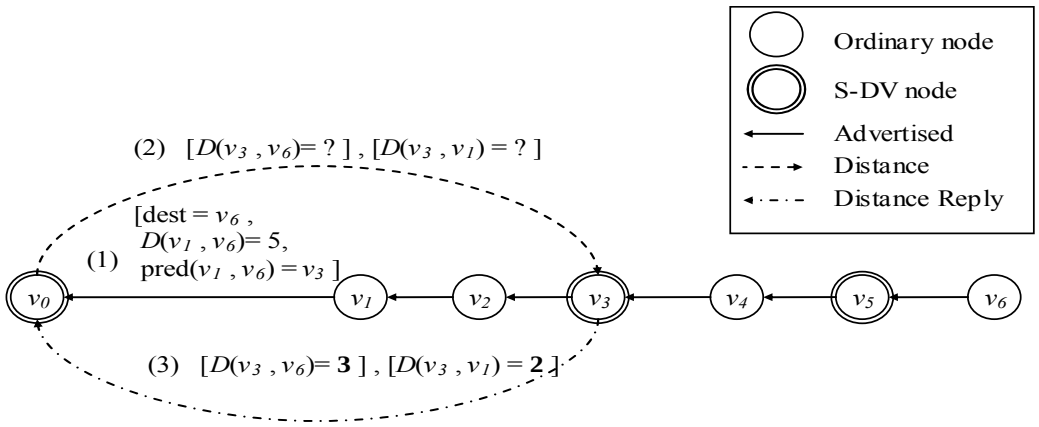

Fig. 2 Consistency Check mechanism of an advertised route

distances $D(v_k, dest)$ and $D(v_k, v_j)$. This predecessor v_k consults its routing table and reply to this query through a Distance Reply message. When v_i receives this message, it checks if the sum of the received distances is equal to the announced distance (formula 1). If it is the case, this advertised route is accepted. Otherwise, it is rejected.

$$D(v_j, dest) = D(v_k, dest) + D(v_k, v_j) \ldots\ldots (1)$$

For example, in Fig. 2, let v0, v3, v5 be S-DV routers, the rest v_1, v_2, v_4, v_6 are DV routers. When v_1 advertises to v_0 a 5 hop route to v_6 with v_5 as a predecessor, v_0 queries v_3's distance for v_6 and v_1. Since $D(v_1, v_6)$ must be the sum of $D(v_3,v_6)$ and $D(v_3,v_1)$, v_1 distance fraud can be easily detected. If the advertised route is consistent, v_0 update its routing table and forwards this route to its neighbor with 1 hop more distance than v_1's distance to that destination and v_0 as a predecessor.

Every S-DV node maintains a temporary routing table (*TmporaryTab*) which contains all current routes in validation process. A consistent route that succeeds in validation is moved to the regular routing table, and can be used for routing data traffic. Otherwise, an inconsistency route or an expired route (Time-out of consistency check process of this route), is rejected

Finally, we use the same authentication scheme as describe in section IV.B to authenticate DR messages exchanged between S-DV peers

V. Details of S-DV

In this section we present the details of our approach S-DV. For an advertised route [destination, distance, predecessor] we use v_i, v_j and *dest* to represent the recipient, the advertiser and the ultimate destination respectively. A destination can be either an IP address of a router noted v_n or a sub-net noted P_n. To be more specific, we use $D(v_j, dest)$ and $pred(v_j, dest)$ to represent the distance and the predecessor respectively from v_j to *dest* for this route.

In the following sections, we present the routing information in both DV and S-DV nodes, the *Security Indicator* used as a metric by S-DV nodes and the details of S-DV nodes process.

A. Common Routing Information

Every node v_i maintains the following information for each neighbor router v_j:

- A shared secret key with this neighbor (cf. A1),
- A sequence number of the last routing update received from this neighbor. This provides the freshness of routing update exchanged between neighbors,
- Router prefix mapping: the set of prefixes (sub-nets) P_j directly attached to this neighbor.
- A routing table (TABLE I) which contains an entry for each destination. This route entry contains the following fields.

TABLE I
ROUTING TABLE

Field	Signification
dest	Identifier of a destination
D(vi, dest)	The distance of this destination
next hop(vi, dest)	The next hop of this destination
pred(vi, dest)	The predecessor of this destination

B. Routing Information in S-DV nodes

In addition to the previous information, every S-DV node, maintains also:

- A shared secret key with every other S-DV node (A2), and a sequence number used in the authentication scheme of DR messages.
- A neighbor table (*neighborTab*) (TABLEI II) which maintains the number of authentic and consistent route advertised from its neighbors
- A temporary routing table (*TemporaryTab*) which maintains the routes in consistency check process.

We note that $\alpha(v_i, v_j)$ measures the probability that an external attacker attack the link (v_i, v_j). This parameter can be

very useful if we generalize our solution to heterogonous networks where the probability of wireless links is larger than the wired links. The other fields are dynamically updated according to the result of routing update validation.

TABLE II
NEIGHBOUR TABLE

Field	SIGNIFICATION
v_j	The neighbor identifier
$\alpha(v_i, v_j)$	Attack probability of the link
$auth(v_i, v_j)$	The number of authentic routing update received from the link (v_i, v_j)
$Nauth(v_i, v_j)$	The number of non authentic routing update received from the link (v_i, v_j)
$coh(v_i, v_j)$	The number of consistent routing update received from the neighbor v_j
$Ncoh(v_i, v_j)$	The number of consistent routing update received from the neighbor v_j

C. Security Indicator

The *Security Indicator* is used as a metric to measure the security level associated to a route with a neighbor node v_j as a next hop. For that, every S-DV node v_i uses its neighbour table to calculate this Security Indicator by the equation .2.

$$Sind(v_i, v_j) = \frac{Nauth(v_i, v_j)}{Nauth(v_i, v_j) + auth(v_i, v_j)} \times \frac{Ncoh(v_i, v_j)}{Ncoh(v_i, v_j) + coh(v_i, v_j)} \times \alpha(v_i, v_j) \quad (2)$$

The security Indicator of v_i's route that has v_j as a neighbour, is the multiplication of three quantities: the attack probability of a link (v_i, v_j), the frequency of non authentic messages, and the inconsistency route frequency received from this neighbor v_j. In our approach, in order to prefer one route to another, we need some mechanism of comparison between routes. This allows S-DV nodes to choose the better route in term of security. Generally, this route has low attacks probability, low frequency of non authentic routing updates and low inconsistence route frequency. We propose to prefer a choice of a route where the multiplication of these three parameters is minimal in a period of time.

For example, when v_i receives a route advertisement to the same destination from two different neighbours v_{j1} and v_{j2}, it chooses the v_{j1} as a next hop if: $Sind(v_i, v_{j1}) < Sind(v_i, v_{j2})$. Otherwise, it chose v_{j2} as a next hop.

Indeed, a secure route is the one which has not been subject to frequent internal and external attacks.

D. Treatments performed by S-DV nodes

When a node vi receives a routing update from its neighbor v_j, First, it checks the authenticity of the received message by the validation of the MAC of the message, according to the validation result, it updates the $Nauth(v_i, v_j)$ or $auth(v_i, v_j)$ variable.

If the received routing message is authentic, v_i checks if one of the advertised routes will be used to update its routing table. In our approach, an advertised route [*dest*, $D(v_j, dest)$, $pred(v_j, dest)$] from v_j is used to update v_i's routing table if one of the following conditions is verified.

1. v_i receives a route advertisement to a destination that doesn't exist in its routing table,
2. v_i receives a route advertisement with shorter distance than the current distance to a destination.
3. v_i receives a route advertisement with longer distance and better security Indicator than the current distance to a destination.

In this case, it performs additional validations such as: route self-consistency, Router/Prefix authentication and consistency check:

1) Self-consistency check

The fist treatment to perform is to check the self-consistency of the advertised route. Indeed, we check if the advertised route fields are not in contradiction without using DR mechanism.

v_i checks the self consistency of an advertised route [*dest*, $D(v_j, dest)$, $pred(v_j, dest)$] from v_j as follows:

1. if $D(v_j, dest) = 0$ then the predecessor of this route $pred(v_j, dest)$ must be null or vj. In this last case, v_j is an S-DV node.
2. if $1 \leq D(v_j, dest) \leq D_{max}$ then the predecessor of this route $pred(v_j, dest)$ must not be v_i since v_j should not advertise a route back to v_i from which it learns that route. Otherwise, the problem of counting to infinity occurs. Although RIP detects this problem and proposes split horizon with trigged update to solve it, a misbehaving node may not follow the rule and intentionally creates the problem.

If the advertised route is self-consistent, according to the distance $D(v_j, dest)$ of that route, v_j performs the following validations:

2) Router/Prefix authentication

If $D(v_j, dest)=0$, v_j advertises to v_i a route for itself or for a subnet directly attached to v_j. If the route is for v_j, the MAC of the routing message previously verified already provide data origin authentication [5]. If the route is for a subnet, the router prefix mapping (cf. A3) is used to check whether v_j is physically connected to that subnet.

3) Consistency check

If $1 \leq D(v_j, dest) \leq D_{max}$, v_j advertises to v_i a reachable route to a destination *dest*. v_i will check the consistency of that route with $pred(v_j, dest)$ using DR mechanism described in section (IV.D) . We note that if there is no predecessor of that route, $pred(v_j, dest)$=null, v_i will accept that route without validation.

4) Infinite route

If $D(v_j, dest) \geq D_{max}$, v_j advertises to v_i a route to a destination *dest* which is infinite or unreachable from v_j. As a consequence v_i will drop this route and will not forward packet to *dest* through v_j. v_i will not validate an infinite route since it is very difficult to force a misbehaving node forward packets to a specific destination if it doesn't want to do so.

If all these validations succeed, v_i updates its routing table if the advertised route is more secure then the current route in use. Otherwise, this route will be rejected. In addition, v_i increments its $coh(v_i, v_j)$ or $Ncoh(v_i, v_j)$ according to the validation results.

VI. THREATS ANALYSES

A network node may have different malicious behaviours over routing information and DR messages. It may generate faulty information with respect to one or more fields that constitute the routing information. It may also block or modify received DR messages. We have already discussed the case of faulty prefix and distance announcements (cf. sections IV.C and IV.D).

In what follows, we discuss the case of faulty announcement of the predecessor, and the modification and suppression of DR messages. We demonstrate that our solution resists to this kind of active attacks. Furthermore, we verify that our cryptographic mechanisms used to guarantee data origin authentication for DR exchanges and route advertisements do not have security holes.

A. Predecessor fraud

A node v_j may advertise a wrong predecessor for a route. Two cases may happen:

1. A node v_j may advertise a non S-DV predecessor for a route. In this case, the first S-DV node that receives this advertisement can easily detected this fraud. Indeed, any S-DV node has a complete knowledge of all S-DV routers and shares a secret key with each of them (cf. assumption A2).
2. A node v_j may announce another predecessor instead of the valid one: for example, assume that v_j is 5 hops far from *dest*. If v_j learns that v_m is 1 hop far from *dest*, then it may pretend to be 2 hops far from *dest* and claims v_m as a predecessor of this route. According to our assumptions, this malicious advertisement can be detected by the first S-DV node which receives it. This detection relies on the information provided by the Distance Reply message sent by v_m. Indeed, this message contains two pieces of information: the distance between v_m and the destination *dest*, and the distance between v_m and v_j. If this latter distance is greater than 1, then v_i concludes that this advertisement is not coherent.

B. Modification or suppression of DR messages

The authentication mechanism based on shared keys to authenticate the origin of DR messages allows detecting DR messages' alteration. Indeed, Message Authentication Codes guarantee data integrity in addition to message origin authentication. The suppression of DR messages is a malicious behaviour that disturbs the route consistency check mechanism. As a consequence to this attack, a valid route might be rejected. We do not consider this attack as a major inconvenient, since the suppression of a route leads to the discovery of another one through the periodic routing information exchange. An adversary uses this kind of attacks when it refuses to forward traffic to a destination. In this case, it is difficult to force a node to forward traffic to destination if it refuses to cooperate.

C. Validation of the data origin authentication mechanism

We used AVISPA (Automated Validation of Internet Security Protocols and Applications) [29] to validate our data origin authentication scheme described in section IV.B. We specified our scheme using HLPSL (High Level Protocol Specification Language) [29] with data origin authentication as a security objective. In appendix A, we provide the whole HLPSL specification of our authentication scheme. AVISPA tool confirmed that the specified protocol is safe.

VII. OVERHEAD EVALUATION AND COMPARISON

In this section, we make a comparison between S-DV and S-RIP [28] with evaluating the number of generated messages in each approach. We consider the following parameters:

1. $\Phi_{SRIP}(dest)$ and $\Phi_{SDV}(dest)$: the average number of generated transmissions of messages by a node in order to check the route consistency to any destination *dest,* in each approach,
2. $\varphi_{SRIP}(dest)$ and $\varphi_{SDV}(dest)$: the number of affected nodes by the checking of a route consistency to any destination *dest,* in each approach,
3. $\Psi_{SRIP}(dest)$ and $\Psi_{SDV}(dest)$: the average number of generated transmissions of messages by all nodes for the checking of route consistency to any destination *dest* in each approach.

Note that in S-RIP, as shown in figure (Fig. 3), a node v_0 checks the route consistency by asking recursively all the intermediate nodes which have forwarded the route advertisement. S-RIP proposes also a new mechanism based on the reputation of nodes to determine the number of nodes which must be asked, to ensure that the route advertisement is correct. This mechanism reduces the number of required nodes to be asked but it makes non deterministic detection of malicious (erroneous) routing updates. In what follows, we consider the deterministic case, i.e., the case where the checking of route consistency requires the asking of all intermediate nodes which have forwarded the route advertisement. Thus, if the average length of a route is £+1, the consistency checking of this route requires the sending of 2*£ messages of type (request/reply). The first message of request will traverse 2 hops to reach the first *next hop* of this route. The last message of request will traverse the £+1 hops to reach the last *next hop* of this route. A reply message will traverse the reverse path of the corresponding request message. Then, we obtain:

$\Phi_{SRIP}(dest)$ =2*[2+3+ ... +(£ +1)]=£*(£ +3) transmissions

In our approach, as shown in figure (Fig. 4) an S-DV node v_0 checks the consistency of a route by asking the predecessor of this route. Then, if we suppose that the average distance between two successive S-DV nodes of a route is Ƙ hops, then

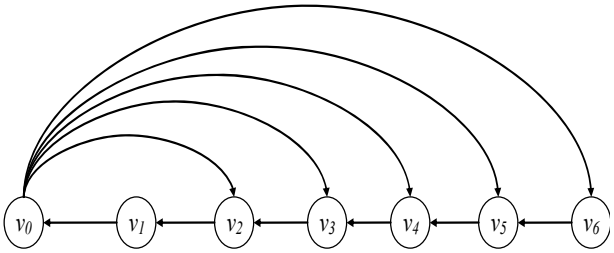

Fig. 3 Number of messages generated by the consistency check mechanism of one route in S-RIP

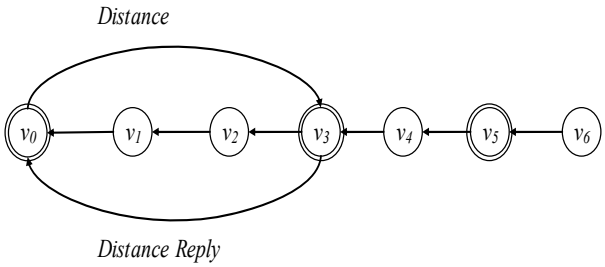

Fig. 4. Number of messages generated by the consistency check mechanism of one route in S-DV

the checking of consistency of this route requires the sending of 2 messages of type (request/reply). The first message of request will traverse 𝕂 hops to reach the predecessor of this route, and the reply message will traverse the reverse path of the corresponding request message. Thus, we obtain $\Phi_{SDV}(dest)$ =2* 𝕂 transmissions. For simplicity reasons in order to make a comparison between SRIP and S-DV, we consider that 𝕂=£. This implies that $\Phi_{SDV}(dest)$=2*£ . Note that £ is generally bigger than 𝕂.

In a routing protocol, a route advertisement propagates from node to node until it reaches all the nodes in the network. In S-RIP all the nodes are affected by the checking of route consistency, where in our approach only the S-DV nodes are affected by the checking of route consistency. Thus, in a network containing S S-DV nodes among n nodes, we have $\varphi_{SRIP}(dest)= n\text{-}1$ and $\varphi_{SDV}(dest)=S\text{-}1$, because the first node in the traversed path by this route doesn't check the consistency of this route. There is neither *next hop* nor predecessor for this route.

From $\Phi(dest)$ and $\varphi(dest)$ we deduce $\Psi(dest)$ which is equal to the multiplication of these two parameters. This last parameter measures the induced *overhead* by each approach for the checking of route consistency. This happens especially when the network nodes or S-DV nodes set the routes in their routing tables for the first time.

TABLE III
COMPARISON BETWEEN S-RIP AND S-DV

Parameter	S-RIP	S-DV	Contribution
Φ(dest)	£*(£+3)	2*£	£*(£+1)
$\varphi(dest)$	n-1	S-1	n-S
$\Psi(dest)$	£*(£+3) * (n-1)	2*£*(S-1)	£*(£+3)*(n-1)-2*£*(S-1)

Table III summarizes the measured parameters for each approach, and show the important contribution of our approach compared to S-RIP:

From Table 3, we deduce that with S-DV we save:

1. £*(£+1) transmissions of messages induced by a node for the checking of route consistency,
2. n-S nodes' affectation by the checking of route consistency, then less memory resources and CPU time used by the whole network nodes.
3. £*(£+3)*(n-1)-2*£*(S-1) message transmissions induced by the whole network nodes for the checking of a single route.

We have seen through the comparison between S-DV and S-RIP the very important contribution introduced by our approach. Besides, we note that our approach is very adapted for large scale networks, because its contribution is in direct relation with £ the average length of a route. In fact, with equitable deployment of S-DV nodes in the network: more the number of S-DV nodes S increases, more the average distance 𝕂 between two successive S-DV nodes in a route decreases. Consequently, the guaranteed security level increases in our approach although the very reduced overall *overhead*.

VIII. CONCLUSION

In this paper, we presented a new approach to secure distance vector routing protocols. We have shown the efficiency of our approach to detect all types of malicious routing updates. The proposed DR mechanism to control the route consistency is executed only by few trusted routers which are the S-DV routers. This reduces the overhead and increases the scalability of our protocol. Unlike existing approaches in literature, our approach doesn't give an absolute priority to the path with shortest distance but to the more secure path. This is guaranteed through a metric which measures the frequency of malicious routing updates, received from each neighbouring node.

In our future works, we aim to expand our approach in the case of DSDV [19] and AODV [18] routing protocols used in ad hoc networks, where the trustiness is a very serious issue in the absence third trust parties.

REFERENCES

[1] F. Baker, R. Atkinson, "RIP-II MD5 Authentication". RFC 2082, January, 1997.
[2] R. E. Bellman, "Dynamic Programming", Princeton University Press, Princeton, New Jersey, 1957.
[3] S. M. Bellovin, "Security Problems in the TCP/IP Protocol Suite". ACM Computer Review, 19(2), pp. 32-48, April, 2001.
[4] A. Barbir, S. Murphy, Y. Yang, "Generic Threats to Routing Protocols". Internet Draft <draft-ietf-rpsec-routing-threats-07.txt>, October, 2004.
[5] Y. Challal, H. Bettahar, A. Bouabdallah, "A Taxonomy of Multicast Data Origin authentication: Issues and Solutions", IEEE

Communications Surveys and Tutorials, Volume 6 number 3: pp. 34-57, October, 2004.
[6] A. Chakrabarti, G. Manimaran, "Internet Infrastructure Security: A Taxonomy," IEEE Network, vol.16, no.6, pp. 13-21, November/December, 2002.
[7] A. Chakrabarti, G. Manimaran, "An Efficient Algorithm for Malicious Update Detection Recovery in Distance Vector Protocols". in Proc. IEEE ICC, pp. 1952-1956, Anchorage, Alaska, May 2003.
[8] J. J. Garcia-Luna-Aceves, S. Murphy, "A Loop-Free Algorithm Based on Predecessor Information". In Proc. of IEEE INFOCOM'1995, April, 1995.
[9] C. Hedric, "Routing Information Protocol". RFC 1058, June, 1988.
[10] A. Heffernan, "Protection of BGP Sessions via the TCP MD5 Signature Option". RFC 2385, August, 1998.
[11] R. Housley, T. Polk, W. Ford, D. Solo, "Internet X.509 public key infrastructure certificate and certificate revocation list CRL profile". RFC 3280, April, 2002.
[12] Y. C. Hu, A. Perrig, D. B. Johnson Nita-Rotaru, "Efficient Security Mechanisms for Routing Protocols". In Proc. Of NDSS, San Diego, USA, February, 2003.
[13] S. Kent, C. Lynn, K. Seo, "Secure Border Gateway Protocol (S-BGP)". IEEE Journal on Selected Areas in Communications, vol. 18, no. 4, pp. 582-592, April, 2000.
[14] G. Malkin, "RIP version 2". RFC 2453, November, 1998.
[15] S. Murphy, M. Badger, B. Wellington, "OSPF with Digitial Signatures". RFC 2154, Jun, 1997.
[16] J. Moy , "OSPF version 2". RFC2328, September, 1998.
[17] V. Mittal, G. Vigna, "Sensor-Based Intrusion Detection for Intra-Domain Distance-Vector Routing". In Proc. of 9th ACM Conf. On Computer and Communication Security 02, pp. 127-137, November, 2002.
[18] C. Perkins, E. Belding-Royer, S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing". RFC 3561, July, 2003.
[19] C. E. Perkins, P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers". In Proc. of the conference on Communications, Architectures, Protocols, and Applications, pp. 234-344, August, 1994.
[20] R. Perlman, "Network Layer Protocols with Byzantine Robustness". Ph.D thesis. Department of Electrical Engineering and Computer Science, MIT, August, 1988.
[21] P. Papadimitratos, Z. J. Haas, "Securing the Internet Routing Infrastructure". IEEE Communication Magazine, pp. 60-68, October, 2002.
[22] D. Pei, D. Massey, L. Zhang, "Detection of invalid Announcements in RIP Protocol". IEEE Globecom, December, 2003.
[23] D. Pei, D. Massey, L. Zhang, "A Framework for Resilient Internet Routing Protocols", IEEE Network Special Issue on Protection, Restoration, and Disaster Recovery, April, 2004.
[24] R. Rivest ," The MD5 Message-Digest Algorithm", RFC 1321, April, 2002.
[25] Y. Rekhter , T. Li, " A Border Gateway Protocol 4 (BGP-4)", RFC 1771, March, 1995.
[26] B. R. Smith, S. Murthy, J. J. Garcia-Luna-Aceves, "Securing Distance-Vector Routing Protocols", In Proc. of Network and Distributed Systems Security, San Diego, USA, February, 1997.
[27] R. White, "Deployment Considerations for Secure Origin BGP (soBGP)". Internet Draft <draft-white-sobgp-bgp-deployment-01.txt>, Jun, 2003.
[28] T. Wan, E. Kranakis, P. C. Van Oorschot, " S-RIP: A Secure Distance Vector Routing Protocol". In Proc. of Applied Cryptography and Network Security, Yellow Mountain, China, Jun, 2004.
[29] Automated Validation of Internet Security Protocols and Applications "AVISPA project", http://avispa-project.org/.

IX. APPENDIX A

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Alice Bob notation :
% A->B : Msg.MAC(Kab,Msg.Sn)
% Where
% Msg : is the exchanged message.
% Kab : is a preshared secret
% Sn : is a sequence number
% MAC : is a Message Authentication Code
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

role route_adv ( A,B : agent,
                 K : symmetric_key,
                 MD5 : hash,
                 Snd, Rcv : channel(dy))
played_by A def=
 local  State: nat,
        Msg   : text, %it can be anything
         Sn : text %sequence number

 init  State := 0

 transition

   1. State = 0
      /\ Rcv(start)
      =|>
      State' := 1
      /\ Msg' := new()
      /\ Sn' := new()
      /\ Snd(Msg'.MD5(Msg'.Sn'.K))
      /\ witness(A,B,sn,Sn')

end role


role route_disc (A, B: agent,
                 K : symmetric_key,
                 MD5 : hash,
                 Rec : channel(dy))
played_by B def=
  local  State     : nat,
         Msg : text, %it can be anything
         Sn : text %sequence number

  init  State := 0

  transition

   1. State = 0
      /\ Rec(Msg'.MD5(Msg'.Sn'.K))
      =|>
      State' := 1
      /\ request(B,A,sn,Sn')
end role


role session (A, B : agent,
              K : symmetric_key,
              MD5: hash) def=

  local SA, RA, RB: channel (dy)

  const sn : protocol_id

  composition

           route_adv(A,B,K,MD5,SA,RA)
        /\ route_disc(A,B,K,MD5,RB)
end role


role environment() def=

  const a, b     : agent,
        kab, kai, kbi : symmetric_key,
         md5 : hash

  intruder_knowledge={a,b,kai,kbi, md5}

  composition
        session(a,b,kab,md5)
     /\ session(a,i,kai,md5)
     /\ session(i,b,kbi,md5)
end role

goal
   authentication_on sn
end goal
environment()
```