International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org, editorijaiem@gmail.com
Volume 2, Issue 7, July 2013 ISSN 2319 - 4847

SECURITY AND COLLABORATIVE ENFORCEMENT OF FIREWALL POLICIES IN VPNS

Dr. P Jayarekha[1], Sunil Kalaburgi[2], Dr. M Dakshayini[3]

[1]Associate Professor, Dept. of ISE, B.M.S College of Engineering, Bangalore, Karnataka, India,
[2]M.Tech Final year , Dept. of ISE, B.M.S College of Engineering, Bangalore, Karnataka, India,
[3]Professor, Dept of ISE, B.M.S College of Engineering, Bangalore, Karnataka, India,

ABSTRACT

A virtual private network (VPN) is a private network that uses public network to connect remote sites. It enables host computer to send and receive data across shared or public network as if they were integral part of private network with all functionality, security and management policies of private network. To access organization network's resource an encrypted VPN tunnel is formed between home network and foreign network. Although VPN technology is very useful, it brings down security threats on remote network because its firewall does not what traffic is flowing inside the VPN tunnel. To address this problem, VGuard was proposed, a frame work that allows a home network and a foreign network to collaboratively determine whether the request satisfies the policy without the home network knowing the request and the foreign network knowing the policy. In this paper we study a protocol called Xhash, which allows two parties, where each party has a key, to compare whether they have the same keys, without disclosing their keys to each other. VGuard frame work that uses Xhash as the basic building block. In order to make the existing approach better, this paper presents a DES-192 algorithm which uses 24 byte key input. This algorithm is used to increase the security among Policy Owner and Request Owner, which makes encryption algorithm more robust to attackers.

Keywords: Virtual Private Networks, Privacy, Network Security

1. INTRODUCTION

VPN is a globally used technology that can help to provide secure private network traffic over an unsecured network, such as the Internet. VPN helps to provide a secure mechanism for encrypting and encapsulating private network traffic and moving through an intermediate network. Date is encrypted for confidentiality and packets that might be intercepted on shared network or public network are unreadable without the correct encryption keys. Data is also encapsulated or wrapped with IP header containing routing information.

The two major concerns in supporting roaming users across administrative field are security and privacy. As we all know, VPN [4] is deployed in many organization to protect their users when they roam into foreign networks. When a roaming user establishes a VPN tunnel with his home network, he can access not only the private resources within the home network, but also redirect his Internet traffic through the VPN tunnel, which is typically encrypted to protect the privacy of user traffic. While roaming users enjoy the security protection offered by VPNs, little consideration has been given to the impact of such encrypted tunnels on foreign network. In particular, the foreign network's firewall cannot effectively regulate such tunnelled traffic, because it is unable to examine the encrypted connection properties, such as destination IP address and ports. As a result, certain connections that are normally prohibited by the foreign network, for either security or policy reasons, can now evade the firewall regulation. The existence of such unregulated tunnels not only weakens the security protection for roaming users, but more importantly leaves the foreign network widely open to various security threats from the public Internet. A typical VPN can deploy as shown in Figure 1.

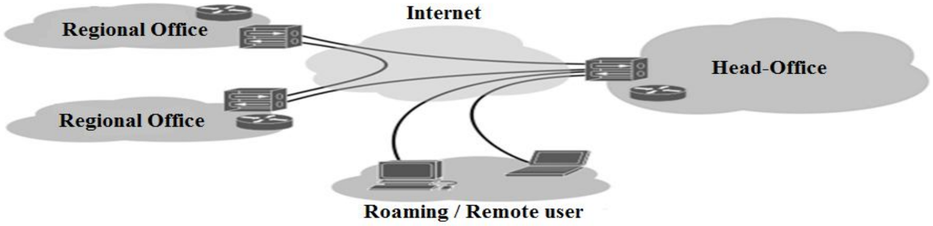


Figure 1: A Typical VPN Example

The rest of paper is organized as follows in section II we study detailed description of Xhash protocol and its disadvantages. In section III we study AODV protocol and its basic operations. In section IV we introduce our proposed algorithm. Section V presents the implementation details, section VI expected results. Finally, section VII presents conclusion and section VIII references.

2.DESCRIPTION OF XHASH PROTOCOL

2.1 Xhash Protocol

In this module we study the simple and efficient Xhash protocol to achieve oblivious comparison [3]. The Xhash protocol works as follows first, Policy Owner sends N1⊕K1 to Request Owner then; Request Owner computes HMAC k (N1⊕K1⊕K2) and sends the results to Policy Owner. Second, Request Owner sends N2⊕K2 to Policy Owner. Third, Policy Owner computes HMAC k (N2⊕K2⊕K1) and compares it with HMAC k (N1⊕K1⊕K2), which was received from the request Owner. Finally, the condition N1=N2 holds if only if HMAC k (N2⊕K2⊕K1) = (N1⊕K1⊕K2).

The above function HMAC is a keyed –Hash Message Authentication Code, such as HMAC-MD5 or HMACSHA1, which satisfies the one-wayness property (i.e. given HMACk(x), it is impracticable to compute x and k) and the collision resistance property (i.e. it is computationally infeasible to find two distinct numbers x and y such that HMACk(x) = HMACk(y)). Note that the key shared between Policy Owner and Request Owner. Although hash collision for HMAC does exist in theory, the probability of collision is negligibly small in practice. Furthermore, by properly choosing shared key k, can safely assume that HMAC has no collision

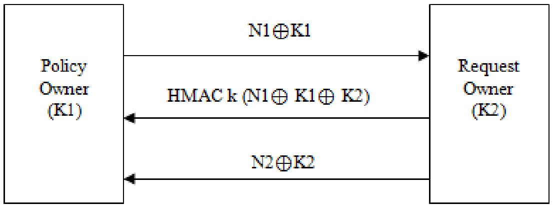


Figure 2: Xhash Protocol

2.1.1 Drawback of Xhash Protocol

Xhash uses HMAC−MD5 digital signature as authentication code and it has the following disadvantages:

- Expiry date: The certificate of digital signature comes with an expiry date and it is the duty of receiver to make sure that public key is valid.
- Need to buy certificates and verification software: Business using digital signature might have to spend more money.
- Compatibility issues: All available digital signatures are incompatible with each other

3.AODV ROUTING PROTOCOL AND ITS OPERATION

3.1 Aodv Protocol Overview

The AODV (Ad Hoc On Demand Distance Vector) routing protocol is a reactive routing protocol, hence routes are determined only when needed. Figure 3 shows how the messages are exchanged in the AODV protocol [11, 12].

Hello messages are used to detect and monitor links to neighbors. Each active node periodically broadcasts a Hello message to all its neighbors. A link break is detected, when a node fails to receive several Hello messages from its neighbors.

When a source wants to transmit the data to unknown destination, it broadcasts a Route Request (RREQ) fro that destination. A route to the source is created, when a RREQ is received at each intermediate node. If the receiving node has not received this RREQ before, is not the destination and does not have current route to the destination, it rebroadcasts the RREQ. If the receiving node is the destination or has a current route to the destination, it generates a Route Reply (RREP). The RREP is unicast in a hop by hop fashion to the source. Once the RREP propagates, each intermediate node creates a route to the destination. When the source receives the RREP, it records the route to the destination and can begin sending data. If multiple RREPs are received by source, the route with shortest hop count is chosen.

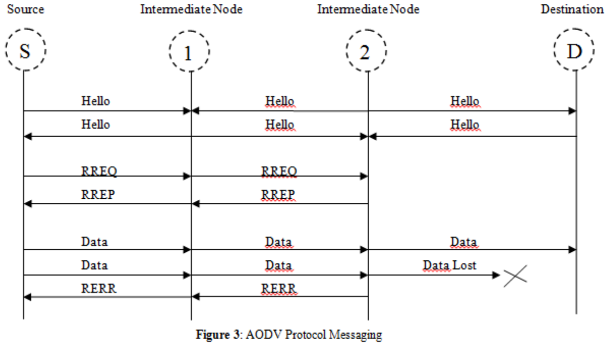


Figure 3: AODV Protocol Messaging

4.PROPOSED ALGORITHM

4.1 Des-192 Encryption Algorithms

This module deals with an enhanced approach of the proposed sytem that is DES-192 (Data Encryption Standard) encryption algorithm. The goal is provide security among Policy Owner and Request Owner. This algorithm is composed of following steps. The 28 round Feistel network, which constitutes the cryptographic core of DES, splites the 192-bit data blocks into two 96-bit words and they are denoted as LBlock and RBlock (L0 and R0). In each round, the second word Ri is fed to function f and the result is added to the first word Li. Then both words are swapped and algorithm proceeds to the next iteration. The function f of DES-192 algorithn is key dependent and consists of four phases.

4.1.1Expansion (E): The 96-bit input word is expaneded to 204-bits by duplicating and reordering half of bits[5].

4.1.2 Key Mixing: The expaned word is XORed with round key constructed by selecting 144-bits from the 168-bit secret key, a differenet selection is used in each round.

4.1.3 Substitution: The 144-bit is split into twentyfour 6-bit words which are substituted in twentyfour parallel 6*12-bit S-boxes. All twentyfour S-oxes are different but havethe same special structure.

4.1.4 Permutation (P): The resulting 96 bits are reordered according to a fixed permutation before being sent to the output.

Algorithm for DES-192 as follows:

```
Pseudo Code: DES−192
INPUT: plaintext p1 . . . p192; 192-bit key K=k1 . . . k192
(includes 24 parity bits)
OUTPUT: 192-bit cipher text block C=c1 . . . c192
1. (Key schedule) Compute sixteen 204-bit round keys Ki, from K.
2. (L0, R0) ← IP (p1, p2 . . . p192) (Use IP Table to permute bits; split the result into left and right 96-bit halves L0=p170, p146 . . . p24, R0=p169 p145 . . . p23)
3. (28 rounds) for i from 1 to 28, compute Li and Ri as follows:
    3.1 Li=Ri-1
    3.2 Ri = Li-1 XOR f (R i-1, Ki)
    Where f (Ri-1, Ki) = P(S (E (Ri-1) XOR Ki)), computed as follows:
    (a) Expand Ri-1 = r1r2 . . . r192 from 192 to 204 bits,
    T ←E (Ri-1)
    (b) T '←T XOR Ki. Represent T ' as twenty four 6-bit character strings: T '= (B1 . . . B24)
    (c)T "← (S1 (B1), S2 (B2). . . . S24 (B24)). Here Si (Bi) maps to the 12-bit entry in row r and column c of Si
    (d)T"'←P (T") (Use P per table to permute the 96 bits of T"=t1t2 . . . t96, yielding t48t21 . . . t75.)
4. b1, b2 . . . b192←(R48, L48). (Exchange final blocks L48, R48)
5. C ←  IP-1 (b1b2 . . . b192)
```

Algorithm 1: DES−192

5.IMPLEMENTATION DETAIL

The proposed routing algorithm has been simulated in NS 2.34. These have been made assuming a network having dimensions 300 x 300 meters. The number of nodes is assumed to be 50. The nodes are generated and placed randomly. The Figure 4 shows node deployment using NS2 network animator.

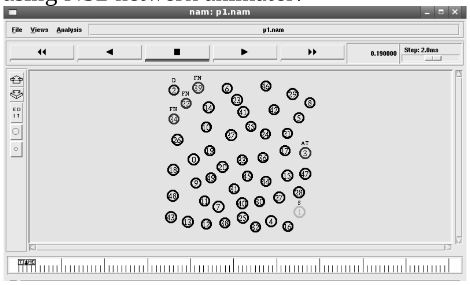


Figure 4: shows the deployment of mobile nodes, All nodes are deployed randomly in the 300 x 300 area

Table 1: Node Configuration

Sl. No.	Parameter	Value
1	Channel type	Wireless channel
2	Antenna type	Omni antenna
3	MAC type	MAC 802_11
4	Queue type	Droptail / Priqueue
5	Propagation type	Two ray ground
6	MAC trace	ON
7	Router trace	ON
8	Traffic agent	UDP

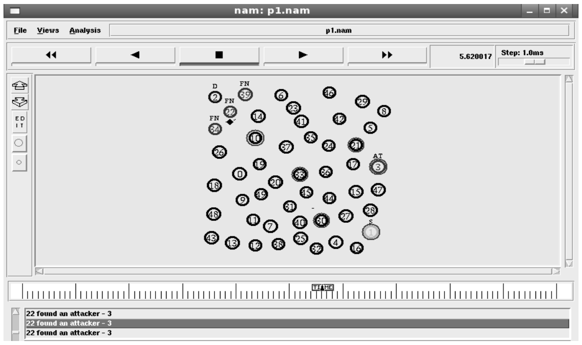
Simulation Environment: The source node S and the destination node D are not moved and are set on the field of 300×300m. The node S transmits data packets to the node D during 60 seconds. Other nodes are set to the random position at first and move with the random way point movement model. It is the movement model which moves at a certain fixed speed below maximum speed from a certain position to a certain destination one and stops during a pause time after arriving at the destination one and starts moving again after the pause time.

The buffer size of each node is 64 packets and each node drops buffered packets after 30 seconds. The battery of each node is consumed at the time of sending and receiving packets and at the time of idle state and it is impossible to communicate when the battery is empty. In this simulation, an RREP-ACK and a Gratuitous-RREP are not used. And for each parameter of AODV, the default value of ns-2 is used. Table 1 shows the simulation parameters used in this implementation.

With the above simulation environment, it is simulated by changing three parameters of the number of nodes, the source and maximum speed of the nodes. When it changes a certain parameter, other two parameters are fixed as the number of nodes is 50, as number of source is 1 and maximum speed of the node is 20 m/s.

Table 2: Simulation Parameter

Serial No.	Parameters	Value
1	Simulator	2.34
2	Number Of Nodes	50
3	Simulation Time	600 Sec
4	Area	300*300 M^2
5	Max Speed	20 M/S
6	Traffic Source	CBR
7	Pause Time	0 ,20,30,40,100
8	Packet Size	512
9	Packet Rate	4 Packets/S
10	Delay	10 Ms
11	Mobility Model	Random Way Point
12	Bandwidth	10 Mbps



22 found an attacker - 3
22 found an attacker - 3
22 found an attacker - 3

Figure 5: shows that packets are being dropped at node 22

In this process attacker which is assumed to be at node 3 named as AT starts sending his packets during 180 seconds. The attacker keeps sending his packets using the same path as authenticated user used to send his packets to destination. Thus the attacker pretended to be an authenticated user and starts sending his own packets to destination. Here node 22 is assumed to be firewall and it is used to keep track of incoming packets. Firewall founds that packets from node 3 are unauthenticated and drops it immediately. Thus, unauthenticated packets are being dropped at node 22. As shown in Figure 5 the packets are being dropped at node 22.

6. RESULTS

In this paper, we evaluate the throughput(i.e, packet delivery ratio) and bandwidth between node S and node D. We assumed node S as Policy Owner and node D as Request Owner in our simulation results. Results shows that the proposed algorithm increases security among Policy Owner and Request owner. Figure 6 shows total packet delivery ratio between node S and node D.

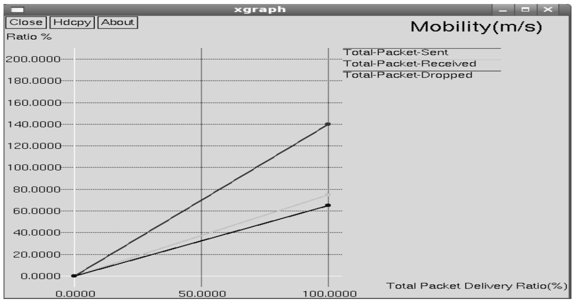


Figure 6: Total Packet Delivery Ratio between node S and node D

The graph shows Mobility of packets between node S and node D. The horizontal axis presents the total packet delivery ratio and the vertical axis shows the total number packets. The graph indicates that the total number of packets received is higher than the total number of packets dropped.

Figure 7 shows estimated bandwidth graph.

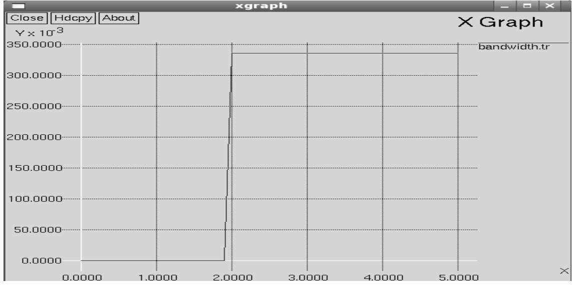


Figure 7: Estimated Bandwidth

The graph shows Bandwidth. The horizontal axis presents the time and the vertical axis shows bandwidth in the multiple of 10^3. The graph indicates that the uniform bandwidth throughout transmission. Here the transmission start at time 2.0 and ends at 5.0. The graph shows Bandwidth of 3.4 Mbps between time interval 2.0 and 5.0, but generally there is no fluctuation in bandwidth.

7.CONCLUSION

We implemented proposed algorithm using NS 2.34 network simulator. We carried out our experiments on a Fedora/Linux operating syetm with 4GB memory and intel core i3 processor. Our results shows that proposed algorithm provides security among Policy Owner and Request Owner which is robust to attackers.

In this paper, we present a DES-192 algorithm, which is an enhancement to VGuard technique. VGuard is more secure of two main reasons. First, VGuard converts existing firewall policies of an ordered list of overlapping rules to an equivalent non-ordered set of non-overlapping rules. Second, VGuard makes sure rule decisions, which helps to prevent Policy Owner from knowing the decision for the given packet.

Our proposed algorithm has following advantages:

1. The VGuard framewwork along with DES-192 algorithm helps to preserve the privacy of communication in VPN.

2. The proposed algorithm not only focus on privacy preservation but also on robustness to attackers.

Future work intends to improve the security among the third party and the Policy Owner. For that time based firewall policies is used to find the appropriate time to transfer the packets between third party and Policy Owner.

REFERENCES

[1.] National Bureau of Standards – Data Encryption Standard, Fips Publication 46,1977

[2.] M. Bawa, R. Bayardo, and R. Agrawal, “Privacy-preserving indexing of documents on the network,” in Proc. VLDB, 2003.

[3.] Alex X. Liu and Fei Chen "Privacy Preserving Collaborative Enforcement of Firewall Policies in Virtual Private Networks" In Proc. IEEE Int. Conf on Network Protocols (ICNP), 2011

[4.] Cheng, J., Yang, H., Starsky Wong, H.Y. and Lu, S. “Design and Implementation of Cross- Domain Cooperative Firewall,” Proc.IEEE Int‟l Conf. Network Protocols (ICNP), 2007.

[5.] W. Stallings, Cryptography and Network Security: Principles and Practices, 5th ed., Prentice Hall, 1999.

[6.] A.X.Liu and M.G. Gouda, "Diverse Firewall Design" proc.intl conf.Dependable systems and networks,pp.595-604,June 2004.

[7.] Fei Chen and Alex X. Liu. SafeQ: Secure and fficient query processing in sensor networks. InProc. IEEE Int. Conf on Computer Communications (INFOCOM), 2010.

[8.] Pankaj Gupta and Nick McKeown. Algorithms for packet classification. IEEE Network, 15(2):24–32, 2001.

[9.] Lam, Ho-Yu, Donghan (Jarod) Wang, Jonathan,H. Chao Department of Electrical and Computer Engineering Polytechnic Institute of New York University, Brooklyn, NY ”A Traffic-aware Top-N Firewall Ruleset Approximation”IEEE ,2011.

[10.]O.P. Verma, Ritu Agarwal, Dhiraj Dafouti,Shobha Tyagi — Performance Analysis Of Data Encryption Algorithms — , 2011

[11.]C. E. Perkins, E. M. Belding-Royer, and S. Das. Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 3561, July 2003.

[12.]C. E. Perkins and E. M. Royer. The Ad hoc On-Demand Distance Vector Protocol. In C. E. Perkins, editor, Ad hoc Networking, pages 173.219. Addison-Wesley, 2000.