Secure AODV Route Updates Over Android

Engineering

KEYWORDS : AES, Diffie Hellman, AODV, Android.

Srisusindhran K	Department of Computer Science and Engineering, SRM University, Chennai.
Ashiq Irphan K	Department of Computer Science and Engineering, SRM University, Chennai.

ABSTRACT

Two important security related issues in an ad-hoc network are, Key Management (KM), how security keys are exchanged and Secure Routing (SR), selecting legitimate nodes for routing. To achieve security for AODV route updates a predefined key is to be exchanged between legitimate nodes using Diffie Hellman key exchange mechanism. The exchanged public key is to be fed to a 256-bit AES encryption process to encrypt all the route request RREQ control messages. The primary goal of this work is to ensure that the route selection and route updates are secured. Since the network considered in this project work is android based mobile node which involves different processing capability it is clear that the latency or processing efficiency might not be constant at all times. However, the project work is aimed at increasing the security of route selection in an AODV enabled ad hoc network.

INTRODUCTION

We consider multiple node scenarios with android based mobile node. Wireless adapter used in this network is of higher importance, for this purpose IEEE 802.11 adapter with ad-hoc support is been used. The communication between the nodes in ad-hoc network is based on AODV protocol. Communication between nodes is carried out without the use of access point.

An ad-hoc network is a systematic way of communicating between nodes in wireless environment. This allows peer-to-peer communication. An ad-hoc network does not depend on pre-installed infrastructure like router, access point, etc[15]. Network connectivity between nodes is carried out dynamically and each node takes part in routing by forwarding data to other nodes.

Related Work

Ad-hoc networks use routing policy in order to adapt routing of data according to the current environment. Among these, routing protocols, the two main types are proactive and reactive [3].

Proactive Protocols

These protocols will keep on updating the information about the link with other nodes, each node has its own routing table[11]. The routing table is updated constantly in the network, even though there is no traffic, this adds processing overhead to all the nodes in the network. The following are a few examples of proactive ad hoc routing protocols.

- OLSR (Optimize Link State Routing Protocol);
- DSDV (Highly Dynamic Destination-Sequenced Distance Vector routing Protocol).

Reactive Protocols

These protocols perform updating on routing table`s information on-demand basis. Finding a route between the nodes operation is performed only when source node requests for a path. This avoids the details of message until the real communication between the nodes takes place. Hence, in this way, the control traffic results to be null if there are opened data session. However, there is no overhead when source finds the destination node route successfully. The cost of this overhead can be considered also as latency time for route discovering. In spite of that, this family of protocols grants a better power consumption for those battery constraint devices [12]. The following are a few examples of reactive ad hoc routing protocols

- AODV (Ad hoc On Demand Distance Vector routing protocol);
- DSR (Dynamic Source Routing).

Ad Hoc On-Demand Distance Vector Protocol

The AODV is classified as a dynamic reactive routing protocol. In a reactive routing protocol, route will be established based on demand basis. The process of discovering routing path from source to destination node. AODV route discovery uses two control messages namely Route Request (RREQ) and Route Reply (RREP). Both control messages carry a field called destination sequence number and it is incremented to determine freshness of a particular route. Unlike most other protocols that are used in an Ad Hoc environment AODV is not a modification of routing protocols used in infrastructure based network, it has been specifically designed for the use in ad hoc environment. AODV achieves its efficiency from minimizing the route acquisition latency and the use of minimal control overhead [11].

In this routing protocol, a node does not have any information about other nodes until a communication is needed. By broadcasting HELLO packets in a regular interval, local connectivity information is maintained between known neighbors.

Ad Hoc on Android

Android devices are usually small handheld devices with limited power source, but generally includes fully functional Wi-Fi adapter, as per IEEE 802.11 standards[4]. In order to make a handheld android device be able to participate in a ad hoc network, two main factors are to be considered, they are, the Wi-Fi adapter has to support ad hoc mode of operation and the linux based android kernel must be able to accept user defined kernel modules, in order to generate and transmit customized AODV control messages [5] [9].

Cryptographic Algorithms

Symmetric-key algorithms are a class of algorithms for cryptography that use the same cryptographic keys for both encryption of plaintext and decryption of cipher text. The keys may be identical or there may be a simple transformation to go between the two keys. The keys, in practice, represent a shared secret between two or more parties that can be used to maintain a private information link. This requirement that both parties have access to the secret key is one of the main drawbacks of symmetric key encryption, in comparison to asymmetric key encryption. This is also known as private key encryption.

Advanced Encryption Standard

The Advanced Encryption Standard (AES) is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology [2].

The key size used for an AES cipher specifies the number of repetitions of transformation rounds that convert the input, called the plaintext, into the final output, called the cipher text. The number of cycles of repetition is as follows:

- 10 cycles of repetition for 128-bit keys.
- 12 cycles of repetition for 192-bit keys.
- 14 cycles of repetition for 256-bit keys.

Diffie Hellman Key Exchange Mechanism

Diffie–Hellman key exchange is a specific method of exchanging cryptographic keys. It is one of the earliest practical examples of key exchange implemented within the field of cryptography.

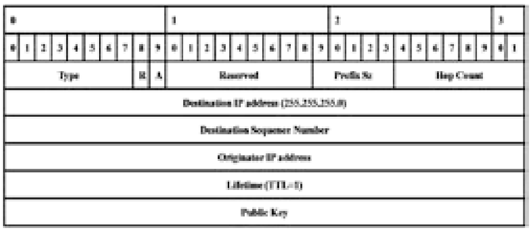
Diffie and Hellman first brought out the concept of public key and digital signature called key agreement protocol [6]. Firstly, the system announce two public integer, g and p, amount them p is a prime number and g is the element of prime number "p". When both sides have the secret communication (for convenience, called Node A and Node B), then process the following steps:

- Node A generates a random private value $X_{a,}$ (Best the same bit with p), then emerge $y_a = g^{X\_a}$ modpgety_a (1)
- Node B generates a random private value $X_{b,}$ (Best the same bit with p), then y_b= g^(X_b) modpgety_b (2)
- Node A sends $Y_a$to Node B, Node B also sends Ybto Node A(Node A does not know $X_{b,\,,}$ Node B does not know $X_{a,}$ either)
- Node A calculates K_a= ⟦y_b⟧^(X_a) modp (3)
- Node B calculates K_b= ⟦y_a⟧^(X_b) modp (4)
- Then Node A and Node B both have secret key K and supposed to be SecretKey=g^(x_a x_b) modp (5)

Because equation 5 is discrete logarithm, so for Y, = gx" mod p, already know Y, p and g requesting number for X, would be very difficult. In other word, when middleperson attacker knows Y, p and g, then it would be difficult to infer the number of X. In addition, system request p and g own the special restriction as well: p should be one big prime number and at least 512 bits. For the security need, p may set to be 1024 or higher bits, furthermore, e of this prime number L should be a prime number or prime factor number [8].

Encryption in AODV

In this system, an environment is created by connecting devices like Android (Mobile device) in an ad hoc network. Then AODV routing protocol operation is performed. The modified Hello Packet Message consists of 32 bit key value in addition. RREP message format are encrypted and decrypted using AES 256 bits.



<table>
<tr><td colspan="10">0</td><td colspan="10">1</td><td colspan="10">2</td><td colspan="2">3</td></tr>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>0</td><td>1</td></tr>
<tr><td colspan="8">Type</td><td>R</td><td>A</td><td colspan="9">Reserved</td><td colspan="5">Prefix Sz</td><td colspan="8">Hop Count</td></tr>
<tr><td colspan="32">Destination IP address (255.255.255.0)</td></tr>
<tr><td colspan="32">Destination Sequence Number</td></tr>
<tr><td colspan="32">Originator IP address</td></tr>
<tr><td colspan="32">Lifetime (TTL=1)</td></tr>
<tr><td colspan="32">Public Key</td></tr>
</table>

Figure 1 Modified Hello Packet

Nodes learn of their neighbors in one of two ways. Whenever a node receives a broadcast from a neighbor, it updates its local connectivity information to ensure that it includes this neighbor, it broadcasts to its neighbors a hello message (a special unsolicited RREP), containing its identity and sequence number. This hello message is prevented from being rebroadcast outside the neighborhood of the node by setting up the time to live (TTL) value to 1.

Control Message Exchange

A node keeps track of neighbor node using HELLO message that each node broadcast at set of time interval. Diffie Hellman processing is done over the uniquely shared common prime number and base value, to generate symmetric key. This symmetric key or public key(whereas here and) is shared among neighbor through the hello message. The symmetric key shard is stored in the routing table along with the private key value. The private key value is stored due to it random generation of value [14]. After exchanging the symmetric key the second phase of the Diffie Hellman is continued to obtain secret key. For encryption and decryption of AES, the obtained secret key act as a key value.

Secure Routing

The route request process is similar to that of the existing system, as usual the node broadcasts a RREQ message to its neighbors with the same control information throughout the network. Once the RREQ is received by the destination node or an intermediate node which has a route to the destination, that specific node generates a RREP, encrypted using AES, based on the secret key already generated by the second phase of Diffie Hellman[13].

RREP is unicasted, each intermediate node has to perform AES decryption using the secret key value generated by Diffie Hellman that is by making use of symmetric key and private key from the routing table. If the neighbor has a valid key, it will be able to read the RREP message and forward to designation address. For which any node is free to respond by sending a RREP message, but the choice of accepting the route is with the original requestor. On receipt of a RREP packet a node checks if it has a route entry in its routing table, if not a route is added, else the sequence numbers in RREP and the route entry are compare. If both the sequence number are the same the RREP is discarded, if the sequence number in the RREP message is greater than the one in the routing table entry, the next hop is checked, if the next hop node is same in both the RREP message and the routing table entry, the sequence number in the routing table entry is updated.

Simulation

The simulation of this work is more concern about two control messages, RREQ and RREP, the goal is to demonstrate the trustworthiness and security addition to the conventional AODV routing mechanism, this additions to the control messages prevent attacks such as Black Hole attack [7] and Replay Attack [1], which involves involvement of illegitimate node, which could possibly not have the Diffie Hellman credentials, thus preventing its participation in route discovery process [10].

Conclusion

The proposed system provides a novel key management and secure routing scheme without interdependency to each other, Diffie Hellman supports the key management through the broadcasted Hello Packets and secure routing mechanism is provide by AES for control packet RREP of AODV.

REFERENCE

[1] Abd Jalil, K., Ahmad, Z., Manan, J.A., "Securing routing table update in AODV routing protocol," Open Systems (ICOS), 2011 IEEE Conference on , vol., no., pp.116,121, 25-28 Sept. 2011. | [2] Advanced Encryption System Algorithm http://en.wikipedia.org/wiki/Advanced_Encryption_Standard, accessed on 25.10.2013. | [3] Charles E. Perkins, "Ad Hoc Networking", Chapter No: 6, pp 173-220, Addison-Wesley Professional, 1st Edition, 2001. | [4] Corriero, N.; Mottola, A.; Zhupa, E., "How to Work with Android within a (FB-)Aodv Network," P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2011 IEEE International Conference on , vol., no., pp.37,42, 26-28 Oct. 2011 | [5] D. Bovet and M. Cesati, "Understanding the Linux kernel".O'Reilly. | [6] Diffie Hellman Key Algorithm http://www.en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange, accessed on 17.10.2013. | [7] Dokurer, Semih."Simulation of Black hole attack in wireless Ad-hoc networks". Master's thesis, AtılımUniversity, September 2006. | [8] Fengqing Tian, Haili Xue, Xue Haiyang, "A Secure 17 Key Encryption from Computational Linear Diffe-Hellman Problem," Computational Intelligence and Security (CIS), 2012 Eighth International Conference on , vol., no., pp.464,468, 17-18 Nov. 2012. | [9] L. Torvalds. Linux kernel homepage. www.kernel.org, accessed on 25.9.2013.. | [10] P. Ning and K. Sun. "How to misuse AODV: A case study of insider attacks against mobile ad hoc routing protocols". In IEEE Systems, Man and Cybernetics Society Information Assurance Workshop, pages 60–67, June 2003. | [11] Perkins, C.E., Royer, E.M., "Ad-hoc on-demand distance vector routing," Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA '99. Second IEEE Workshop on , vol., no., pp.90,100, 25-26 Feb 1999. | [12] Shih-Lin Wu, Yu chee Tseng. "Wireless Ad-hoc Networking". Auerbach Publications, 2007. | [13] Shushan Zhao, Robert Kent, Akshai Aggarwal, "A key management and secure routing integrated framework for Mobile Ad-hoc Networks", Ad Hoc Networks, Elsevier Volume 11, Issue 3, May 2013, Pages 1046-1061,ISSN1570-8705. | [14] Yi-Fung Huang, Kun-Li Wen, Ruei-Hau Hsu, Chu-Hsing Lin, "Infinite generating keys based on publish system," Systems, Man and Cybernetics, 2003. IEEE International Conference on , vol.4, no., pp.3238,3243 vol.4, 5-8 Oct. 2003. | [15] Zygmunt J. Haas, Jing Deng, Ben Liang, Panagiotis Papadimitratos, S. Sajama, "Wireless Ad-hoc Networks" 2002.