# Authentication in Mobile Ad Hoc Networks

Sirapat Boonkrong
Department of Computer Science
University of Bath
cspsb@cs.bath.ac.uk

Russell Bradford
Department of Computer Science
University of Bath
rjb@cs.bath.ac.uk

**Abstract**

Our studies show that there are performance and security problems with the existing authentication and keying mechanisms which are currently employed by wireless ad hoc networks. We propose a new authentication protocol, which solves those problems using a combination of well known cryptographic tools in RSA and Diffie-Hellman. In addition to the actual authentication, a new pairwise session is generated as a result of this mechanism. We also point out that without any central authority, e.g., a central server (which is the nature of ad hoc networks), our authentication scheme can be carried out securely by any node at any time.

**Keywords**: MANET, Authentication

## 1 Introduction

Wireless ad hoc networks are gaining popularity in recent years due to their mobility, flexibility and ease of deployment. A mobile ad hoc wireless network is a network without any central authority, e.g., a central server. Without a central administration, network packets are forwarded from one machine to another by the nodes within the network. This means that each mobile station acts as both a machine and a router. Since nodes have to rely on each other to forward the network packets to destinations, several questions have to be asked. Which users/machines do you allow to join the network? How can you trust the routing node to send the packets to the destinations correctly? How do you know that a node will not send any forged packets? It can be seen that these questions are ones regarding trust. This is where authentication comes in.

Authentication is simply a process carried out by two parties in order to identify one another. Without authentication, an unauthorised node could easily "come in" and use the available resources within the network. The problem gets worse if the unauthorised node is a malicious user. Therefore, it is necessary to have a mechanism for preventing an "outsider" from being part of the network.

In the rest of the paper, we first briefly review existing authentication and keying mechanisms and their problems. Then, in Section 3, we describe our proposed authentication protocol. A brief proof of correctness is presented in Section 4. Section 5 discusses the protocol, followed by the conclusion in Section 6.

## 2 Related work

There are many authentication and keying protocols available nowadays, but they do not appear to be suited for ad hoc networks as we explain below.

ID-based cryptography [8, 4, 1, 9] has an advantage in that keys do not have to be distributed (therefore less traffic), but it is necessary to involve a third party node (which is not ideal for ad hoc networks) in order to establish a key so that network packets can be decrypted. This means that ad hoc networks would lose their flexibility and scalability. Threshold cryptography [2, 4, 12, 9] reduces the problem of a single point of failure, but the user has to contact a number of machines before being able to read a message. There is always a danger of not having enough reachable machines to establish a private key. If that happens, the user will have to move his machine in order to find more nodes, which can give him the rest of the secret shares. This implies that the user will be spending more time to obtain a secret key and wasting some necessary computing resources. Furthermore, having to contact other mobile nodes can add unnecessary traffic to the network.

A third protocol is cluster based authentication [11]. The cluster based architecture, as claimed by Venkatraman et al, is "most suitable for large networks with several nodes" [10], but basing the authentication on this architecture increases the security risks. The most obvious drawback of this protocol is its single point of failure. If the cluster head is compromised, then that particular cluster will no longer be able to function and communicate with other clusters.

There are also other protocols [7, 5], which are also not practical for use in mobile ad hoc networks. This is because both protocols require that the authenticating machines must be close to one another for the protocols to be secure.

## 3 Proposed authentication protocol

The problem we are facing here is that we would like to prevent unauthorised mobile machines from freely "joining" an existing private/local mobile ad hoc network. When designing the authentication protocol, we have to take the physical limitations of wireless ad hoc networks into consideration. They include bandwidth limitations, computational power limitations, memory limitations and battery life limitations. There are three main goals that need to be achieved with the authentication protocol. They are mutual authentication, pairwise session key establishment, and we also want it to be efficient, i.e., as fast as possible and as few messages as possible.

Mutual authentication is essential, because each node needs to know that the other party is who he says he is, and that the other party is authorised to be part of the existing ad hoc network. The establishment of a pairwise session key after the mutual authentication is necessary, because we will want to protect the privacy of the communication between the two participating parties. Finally, the protocol will need to be efficient due to the physical limitations to the mobile ad hoc networks described earlier.

Due to the space restriction, we will not be able to explain the analyses and proofs using the BCK formal model [6]. The proof of correctness using GNY logic [3] is presented very briefly in the next section. The proposed protocol is described below.

We begin the protocol by assuming that any node that would like to join the network must have the known shared secret. Since every node needs to have the shared secret, $K$, in order to be part of the local ad hoc net-

work, each of them will have to be authenticated before joining the network by proving that he does actually hold the shared secret.

Here we let the public key cryptosystem be secure against chosen ciphertext attacks, $E_K()$ be a pseudorandom function and secure against chosen plaintext attacks, and $MAC$ be a secure message integrity scheme.

**Step 0**: Initialisation A: $+K_a, -K_a$, B: $+K_b, -K_b$ and they both have a shared secret, $K$, where $+K_i$ is $i$'s public key and $-K_i$ is $i$'s private key.

**Step 1**: The initiator B chooses $seq \xleftarrow{R} \{0,1\}^n$ and $N_b \xleftarrow{R} \{0,1\}^n$. B computes $C = E_K(+K_b, N_b, seq)$ and $MAC(C)$, and sends $(Req, C, MAC(C))$ to A.

**Step 2**: Upon the receipt of $(Req, C, MAC(C))$, A checks the message integrity and computes $m = D_K(C)$. If both are successful then A accepts $m$, and he should be able to "make some sense" of the message. A increments $seq$ by 1. A chooses $R_A \xleftarrow{R} \{0,1\}^n$ and $N_a \xleftarrow{R} \{0,1\}^n$. A sends $E_{+K_b}(N_a, N_b, seq, R_A, +K_a)$ together with his signature $SIG_A(N_a, N_b, seq, R_A, +K_a, B)$ to B. A computes $C = E_K(R_A)$.

**Step 3**: Upon the receipt of $E_{+K_b}(N_a, N_b, seq, R_A, +K_a)$ and A's signature. B computes $m = D_{-K_b}(E_{+K_b}(N_a, N_b, seq, R_A, +K_a))$ and verifies the signature. If the signature is valid then B verifies the nonce $N_b$ and the sequence number. If they are what B expects then B knows that A also possesses the shared secret $K$. B increments $seq$ by 1. B computes $C' = E_K(R_A)$ and sends $E_{+K_a}(N_a, seq, C', \beta = g^y)$ together with his signature $SIG_B(N_a, seq, C', \beta = g^y, A)$ to A. B deletes $R_A$.

**Step 4**: Upon receipt of $E_{+K_a}(N_a, seq, C', \beta)$ and B's signature, A computes $m = D_{-K_a}(E_{+K_a}(N_a, seq, C', \beta))$ and verifies the signature. If the verification succeeds then A verifies the sequence number, the nonce $N_a$, and checks if $C = C'$. If successful then A increments $seq$ by 1 and sends to B $E_{+K_b}(N_b, seq, \alpha = g^x)$ together with his signature $SIG_A(N_a, seq, \alpha = g^x, \beta, B)$. A is now able to compute the pairwise session key $K_{ab} = \beta^x$. A erases $R_A$ and $x$.

**Step 5**: Upon receipt of $E_{+K_b}(N_b, seq, \alpha)$ and A's signature, B computes $m = D_{-K_b}(E_{+K_b}(N_b, seq, \alpha))$ and verifies the signature. If the signature is valid then B verifies the sequence number and nonce $N_b$. If successful then B computes the session key $K_{ab} = \alpha^y$. B erases $y$ and $R_A$.

# 4 GNY analysis

The proposed protocol is proved for correctness and security using the GNY cryptographic protocol. The GNY protocol is commonly used for the analysis of the security of cryptographic protocol. All of the symbols, notations and rules can be referenced in [3]. The four messages that are exchanged during the authentication process is written in the "language" of the GNY logic as follows:

1. B→A: A: $\lhd *Req, *\{*+K_b, *N_b, *seq\}_K, *MAC(*\{*+K_b, *N_b, *seq\}_K)$
2. A→B: B: $\lhd *\{*N_a, N_b, *R_A, *seq, *+K_a\}_{+K_b}, *\{*N_a, N_b, *R_A, *seq, *+K_a, B\}_{-K_a}$
3. B→A: A: $\lhd *\{N_a, *seq, *\{R_A\}_K, *K''\}_{+K_a}, *\{N_a, *seq, *\{R_A\}_K, *K'', A\}_{-K_b}$
   $\rightsquigarrow B \mid\equiv B \xleftrightarrow{K''} A$, where $K'' = g^y \bmod p$
4. A→B: B: $\lhd *\{N_b, *seq, *K'\}_{+K_b}, *\{N_b, *seq, *K', K'', B\}_{-K_a}$
   $\rightsquigarrow A \mid\equiv A \xleftrightarrow{K'} B$, where $K' = g^x \bmod p$

**Assumptions**

| | |
|---|---|
| A$\mid\equiv$A$\overset{K}{\longleftrightarrow}$B | B$\mid\equiv$B$\overset{K}{\longleftrightarrow}$A |
| A$\mid\equiv$A$\overset{K'}{\longleftrightarrow}$B | B$\mid\equiv$B$\overset{K''}{\longleftrightarrow}$A |
| A$\mid\equiv$B$\mid\Longrightarrow$B$\overset{K''}{\longleftrightarrow}$A | B$\mid\equiv$A$\mid\Longrightarrow$A$\overset{K'}{\longleftrightarrow}$B |
| A$\mid\equiv$B$\mid\Longrightarrow$B$\mid\equiv *$ | B$\mid\equiv$A$\mid\Longrightarrow$A$\mid\equiv *$ |
| A$\mid\equiv \sharp(N_a)$ | B$\mid\equiv \sharp(N_b)$ |
| A$\ni$K | B$\ni$K |
| A$\ni +K_a$ | B$\ni +K_b$ |
| A$\ni -K_a$ | B$\ni -K_b$ |
| A$\ni N_a$ | B$\ni N_b$ |
| A$\ni R_A$ | A$\mid\equiv \sharp(R_A)$ |

**Notations**:

- $+K_a$ - A's public key
- $+K_b$ - B's public key
- $-K_a$ - A's private key
- $-K_b$ - B's private key
- $K'$ - A's DH public component
- $K''$ - B's DH public component

**Analysis**

**Message 1**:
Applying T1, T3 and P1, we get $A \ni Req, +K_b, N_b, seq, MAC(\{+K_b, N_b, seq\}_K)$. Since $A \ni +K_b, N_b, seq$ and $A \ni K$, by P4, $A \ni H(\{+K_b, N_b, seq\}_K)$. Now A can check if $H(\{+K_b, N_b, seq\}_K)$ is equal to $MAC(\{+K_b, N_b, seq\}_K)$. If so, carry on.

**Message 2**:
Applying T1, T4, P1, F1, R1 and I4. The rule P1 shows that B now possesses all necessary components, namely random nonces, random challenge and A's RSA public key components. F1 and R1 confirm that the message is fresh and recognisable. The rule I4 gives us $B \mid\equiv A \mid\sim X$, which means that B believes that A sent the message.

**Message 3**:
$B \mid\equiv B \overset{K''}{\longleftrightarrow} A$ is valid because it is an initial assumption. We then apply T1, T4 and P1 to obtain $A \ni N_a, seq, \{R_A\}_K, K'', \{N_a, seq, \{R_A\}_K, K'', A\}_{-K_b}$ A can now verify $\{R_A\}_K$. We further apply the rules F1, R1, I4, J2 and J1 in order to obtain $A \mid\equiv B \overset{K''}{\longleftrightarrow} A$ (as well as making sure that the message is fresh and is really from B), which implies that $A \mid\equiv B \overset{K_{ab}}{\longleftrightarrow} A$.

**Message 4**:
$A \mid\equiv A \overset{K'}{\longleftrightarrow} B$ is valid because it is an initial assumption. We apply T1, T4, P1, F1, R1 and I4 to show that the message is fresh (i.e. not a replay) and to show B believes that A really sent the message together with his signature. The rules J2 and J1 are then used to get $B \mid\equiv A \overset{K'}{\longleftrightarrow} B$, which implies that $B \mid\equiv A \overset{K_{ab}}{\longleftrightarrow} B$.

Therefore, we get:

$A \mid\equiv B \overset{K''}{\longleftrightarrow} A$ and $B \mid\equiv A \overset{K'}{\longleftrightarrow} B$

$A \mid\equiv B \mid\equiv B \overset{K''}{\longleftrightarrow} A$ $B \mid\equiv A \mid\equiv A \overset{K'}{\longleftrightarrow} B$

which implies that

$A \mid\equiv B \overset{K_{ab}}{\longleftrightarrow} A$ and $B \mid\equiv A \overset{K_{ab}}{\longleftrightarrow} B$

$A \mid\equiv B \mid\equiv B \overset{K_{ab}}{\longleftrightarrow} A$ $B \mid\equiv A \mid\equiv A \overset{K_{ab}}{\longleftrightarrow} B$

where
$K_{ab} = (g^x)^y \bmod p = (g^y)^x \bmod p$.

We have shown in this section that according to the GNY analysis, our protocol is correct and secure. It also results in a new pairwise session key, which is one of the requirements stated in the previous section.

# 5 Discussion

The first thing we would like to point out is that there is no trusted third party, e.g., a RADIUS server, involved during the actual authentication process.

This means that any mobile node within the network can carry out the authentication at any time. Secondly, by not having any node solely responsible for the authentication, we have eliminated the problem of a single point of failure. That is, even if a mobile node within the network is compro-

mised, the authentication can still be carried out by other nodes.

Moreover, only four packets are required to achieve the mutual authentication and pairwise key establishment. Note that the mutual authentication is achieved by having the two participating parties prove that they both hold the shared secret key. It should also be noted that if a node is not a valid member, no packets sent by him will be accepted, except for the request-to-join packet.

We have also run a simulation, which demonstrates that the protocol does work as expected, i.e. it carries out the mutual authentication and both participating parties end up with the same pairwise session key.

Finally, our protocol has been proved to be secure and correct using the BCK formal model and the GNY protocol.

# 6 Conclusion

We have proposed a new authentication protocol, which we consider as the first line of defence against any attacks on mobile ad hoc networks. Our protocol addresses the problems of other existing protocols by using a combination of some well known cryptographic tools. The proposed scheme has applied two well-known protocols, the RSA cryptosystem and the Diffie-Hellman key agreement method.

The resultant protocol achieves mutual authentication and pairwise session key. It is also efficient in that it only requires four messages to complete the protocol. The BCK analysis (which is not presented here) shows that the protocol is secure in the unauthenticated-links model. The GNY proof of correctness that we have shown briefly demonstrates further that our protocol is secure.

# References


[1] Dan Boneh and Matthew Franklin. Identity-based encryption from the Weil Pairing, 2001. This is the full version of an earlier published in Crypto ’2001.

[2] Peter S. Gemmell. An introduction to threshold cryptography. *CryptoBytes*, 2(3):7–12, Winter 1997.

[3] Li Gong, Roger Needham, and Raphael Yahalom. Reasoning about belief in cryptographic protocols. In *Proceedings of the IEEE Symposium on Research in Security and Privacy*, pages 234–248, Oakland, CA, May 1990. IEEE Computer Society, Technical Committee on Security and Privacy, IEEE Computer Society Press.

[4] Aram Khalili, Jonathan Katz, and William A. Arbaugh. Toward secure key distribution in truly ad-hoc networks. In *Proceedings of the IEEE Workshop on Security and Assurance in Ad hoc Networks, in conjunction with the 2003 International Symposium on Applications and the Internet*, Orlando, FL, January 2003.

[5] Tim Kindberg and Kan Zhang. Secure spontaneous interactions in ubiquitous computing, February 2004. Seminar given at the University of Bristol.

[6] Ran Canetti Mihir Bellare and Hugo Krawczyk. A modular approach to the design and analysis of authentication and key exchange protocols. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 419–428, Dallas, Texas, USA, 1998.

[7] A O Salako. Authentication in ad hac networking. In *Proceedings of*

*London Communications Symposium 2002*, 2002.

[8] Adi Shamir. Identity-based cryptosystems and signature schemes. In G. R. Blakley and David Chaum, editors, *Advances in Cryptology: Proceedings of CRYPTO 84*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer-Verlag, 1985, 19–22 August 1984.

[9] Tyron Stading. Secure communication in a distributed system using identity based encryption. In *Proceedings of Cluster Computing and the Grid*, pages 414–420, 2003.

[10] Lakshmi Venkatraman and Dharma P. Agrawal. A novel authentication scheme for ad hoc networks. In *WCNC2000:IEEE Wireless Communications and Networking Conference 2000*, September 2000.

[11] William A. Wulf, Alec Yasinsac, Katie S. Oliver, and Ramesh Peri. A technique for remote authentication. In *Proceedings of the Internet Society Symposium on Network and Distributed System Security*, pages 159–164, 1994.

[12] Lidong Zhou and Zygmunt J. Haas. Securing ad hoc networks. *IEEE Network*, 13(6):24–30, 1999.