Some Remarks on Andrew Secure RPC

Sirapat Boonkrong

King Mongkut's University of Technology North Bangkok
1518 Pibulsongkram Road
Bangsue, Bangkok
10800
Thailand
sirapatb@kmutnb.ac.th

Abstract: We review the Andrew secure RPC protocol and reveal some unsoundness of it. Some modifications are made to the protocol. The changes made include the encryption in the first message, the expansion of the second and third messages as well as the elimination of the fourth message. Our GNY analysis shows that even though changes have been made, the outcomes of the protocol do not change. That is, both client and server hold the same new secret key shared between themselves.

1 Introduction

Although more than twenty years old, the Andrew secure RPC [Sat89] is still widely used as an example in the literature. That is why we feel that there is the need to make it as secure and efficient as possible. Since the original protocol, several attempts [BM03, Low96] have been made in order to make the protocol more secure. Even that, we have discovered that the Andrew RPC still leaves rooms for improvements. We, therefore, make several modifications to the protocol. That is, we add encryption to the first message to prevent the known-plaintext attacks. Another nonce is added to the second message as a challenge for authentication purposes. We add an identity of the sender in the third message in order to prevent session-hijacking. Moreover, we agree with [Low96] that the fourth message really contains no information, hence no uses for security, so we eliminate the fourth message. The modified protocol was then proved for correctness using the logic of Gong, Needham and Yahalom, also known as the GNY logic [GNY90, MSnN94]. The analysis of the newly modified protocol shows that the outcomes do not change from the original, which means both client and server will end up having a new shared secret.

The rest of the paper is organised as follows. The notations of the GNY Logic [GNY90] and the background of the Andrew secure RPC, including the original protocol, the modification made in [BAN90] and the adapted Andrew RPC [Low96], are mentioned in Section 2. Section 3 presents some remarks on the Andrew secure RPC. The modified protocol as well as the GNY analysis will be in Section 4. Section 5 concludes the paper.

2 Background

This section contains a short description and notations of the GNY logic [GNY90, MSnN94] as well as the background knowledge on the Andrew secure RPC. The background on the Andrew RPC includes the description of the original protocol, the protocol after BAN analysis and the Adapted Andrew RPC protocol.

2.1 GNY Logic

The GNY logic is a formal tool that allows us to analyse cryptographic protocols, step by step according to the rules provided (they can be found in [GNY90]).

Here we list the notations of the GNY logic in the hope that the readers, who are unfamiliar with the logic will understand the protocol description as well as the proof of correctness better. The notations are extracted from [GNY90].

Let P and Q be principals. The followings are the basic notations used in the GNY protocol.

- $P \lhd X$: P *is told* formula X. P receives X, possibly after performing some computation such as decryption. That is, a formula being told can be the message itself, as well as any computable content of that message.
- $P \ni X$: P *possesses*, or is capable of possessing, formula X. At a particular stage of a run, this includes all the formulae that P has been told, all the formulae he started the session with, and all the ones he has generated in that run. In addition P possesses, or is capable of possessing, everything that is computable from the formulae he already possesses.
- $P \mid\sim X$: P *once conveyed* formula X. X can be a message itself or some content computable from such a message, i.e. a formula can be conveyed implicitly.
- $P \mid\equiv \sharp(X)$: P *believes*, or is entitled to believe, that formula X is *fresh*. That is, X has not been used for the same purpose at any time before the current run of the protocol.
- $P \mid\equiv \phi(X)$: P *believes*, or is entitled to believe, that formula X is *recognisable*. That is, P would recognise X if P has certain expectations about the contents of X before actually receiving X. P may recognise a particular value (e.g. his own identifier), a particular structure (e.g. the format of a timestamp), or a particular form of redundancy.
- $P \mid\equiv P \overset{S}{\leftrightarrow} Q$: P *believes*, or is entitled to believe, that S is a suitable *secret* for P and Q. S will never be discovered by any principal except P, Q. This notation is symmetrical: $Q \overset{S}{\leftrightarrow} P$ and $P \overset{S}{\leftrightarrow} Q$ can be used interchangeably.

- $P \lhd *X$: P is told a formula which he did not convey previously in the current run. That is, X can be regarded as a *not-originated-here* formula.
- Let C be a statement. $P \mid\equiv C$: P believes, or P would be entitled to believe, that statement C holds.

2.2 Andrew Secure RPC

The Andrew secure RPC was introduced in [Sat89]. It allows two parties, A and B (usually a client and a server), who already share a key K_{ab}, to agree upon a new key K'_{ab}. The protocol also performs an authentication handshake. There are four messages in the protocol exchange. The first three, A and B perform a handshake using a shared secret K_{ab}. In the final message, B sends a new key K'_{ab} to A. The protocol can be summarised as follows. Note that nonce N_a is chosen by A and nonces N_b, N'_b are chosen by B.

$$\begin{array}{ll} \text{Message 1.} & A \rightarrow B : \{N_a\}_{K_{ab}} \\ \text{Message 2.} & B \rightarrow A : \{N_a + 1, N_b\}_{K_{ab}} \\ \text{Message 3.} & A \rightarrow B : \{N_b + 1\}_{K_{ab}} \\ \text{Message 4.} & B \rightarrow A : \{K'_{ab}, N'_b\}_{K_{ab}} \end{array}$$

Unfortunately, Burrows *et al.* [BAN90] have pointed out that there is a problem with the freshness of the new key K'_{ab}. That is, there is nothing that can guarantee that K'_{ab} is fresh. Another problem has been mentioned by Clark and Jacob [CJ95] that an intruder could record the second message and substitute it in place of the fourth. The result is that A would accept $N_a + 1$ as a new key. However, for this attack to be successful, it depends on the property of the nonce N_a, i.e., whether or not the nonce is predictable. Due to the problems stated, Burrows *et al.* revised the protocol.

2.3 Andrew Secure RPC after BAN

Burrows *et al.* carried out an analysis on Andrew secure RPC using their logic of authentication or BAN [BAN90]. The result of the analysis shows that the original Andrew secure RPC could suffer from a replay attack, as mentioned in the previous section. Therefore, the original protocol was revised and the resultant protocol is as follows.

$$\begin{array}{ll} \text{Message 1.} & A \rightarrow B : A, N_a \\ \text{Message 2.} & B \rightarrow A : \{N_a, K'_{ab}\}_{K_{ab}} \\ \text{Message 3.} & A \rightarrow B : \{N_a\}_{K'_{ab}} \\ \text{Message 4.} & B \rightarrow A : N'_b \end{array}$$

Lowe [Low96] exposed the weakness of this revised protocol by introducing an attack on it. Lowe's attack shows that an intruder could engage in two protocol runs in parallel.

In run number one, A tries to contact B but an intruder I intercepts the message, and masquerades as B. In run number two, the intruder initiates the session with A while impersonating B. The description of the attack can be seen in [Low96]. Bird *et al.* have also presented the similar attack on the protocol [BGH$^+$91, BBG$^+$93]. As a result, Lowe fixed the problem to make it less vulnerable to this kind of attack.

2.4 Adapted Andrew RPC

Lowe [Low96] addressed the problem, stated in the previous section, by changing message 2 to include an encrypted copy of the sender's identity. This can prevent the attack in that an intruder will not be able to replay the message anymore. The Adapted Andrew RPC is described as follows. Note that message 2 now carries the identity of B.

$$\begin{array}{ll} \text{Message 1.} & A \rightarrow B : A, N_a \\ \text{Message 2.} & B \rightarrow A : \{N_a, K'_{ab}, B\}_{K_{ab}} \\ \text{Message 3.} & A \rightarrow B : \{N_a\}_{K'_{ab}} \\ \text{Message 4.} & B \rightarrow A : N'_b \end{array}$$

Even though problems with the Andrew secure RPC have been found and addressed, we believe that there are still things that need to be mentioned. They include possibilities of an attack as well as the efficiency of the protocol.

3 Remarks on Andrew Secure RPC

As mentioned earlier, since Andrew secure RPC still appears a lot in literature, we believe that if possible, we should make an attempt to make it as secure and efficient as possible. This section presents some remarks that we have on the Andrew secure RPC.

3.1 Attacks

After having studied the Adapted Andrew RPC, the latest variation of the Andrew secure RPC, we reckon there are a couple of vulnerabilities to the protocol. The first is the *known-plaintext attack*. The second is *session hi-jacking*. We discuss each of them in turn.

3.1.1 Known-Plaintext Attack

By definition, a known-plaintext attack occurs when a cryptanalyst or an attacker has access to the plaintext and the ciphertext of one or more pieces of data, and is at liberty to make use of them to reveal secret information, such as the encryption key. Let us take a look at the first and third messages of the Adapted Andrew RPC.

$$\text{Message 1.} \quad A \rightarrow B : A, N_a$$
$$\text{Message 3.} \quad A \rightarrow B : \{N_a\}_{K'_{ab}}$$

It can easily be observed that in message 1, the nonce N_a is sent in clear. That means, an attacker could eavesdrop and record the nonce. A little later, message 3 is sent. This time the content of the message is the nonce N_a encrypted with the new key K'_{ab}. Again, the same attacker could eavesdrop the conversation and record the encrypted copy of the nonce N_a that he has recorded earlier. Now, the attacker holds the plaintext, N_a, and the ciphertext, $\{N_a\}_{K'_{ab}}$. By having the plaintext and ciphertext pair, the attacker could initiate a *known-plaintext attack*. We understand that having one pair of plaintext and ciphertext may not be enough to successfully attack the protocol this way, but we do think that it is worth pointing out this weakness.

3.1.2 Session-Hijacking

Session-hijacking occurs when an attacker takes over a conversation between two parties. Here, we explain that message 3 of the Adapted Andrew RPC could lead to "session hijacking". We put the words in quote, because we do not think that the attacker could steal the session per se. What he could do is as follows.

By looking at message 3 of the Adapted Andrew RPC,

$$\text{Message 3.} \quad A \rightarrow B : \{N_a\}_{K'_{ab}}$$

we see that it is sent from A to B in order to confirm that A has correctly received the new key K'_{ab}. Without his identity as part of the message, A sends *only* the nonce N_a encrypted with the new key K'_{ab}. The implication of this is that an attacker could intercept the message and forward it to B. B would think that this message comes from the attacker, not A. We acknowledge that this vulnerability on its own does not reveal any secret, but B could then send subsequent messages to the attacker instead of A.

3.2 Excessive Message

In the previous section, a couple vulnerabilities in the Adapted Andrew RPC are introduced. Here, we look at the efficiency of the protocol. By efficiency, we mean the number of messages used to complete the protocol.

Having studied the Adapted Andrew RPC, we agree with [Low96] that message 4 of the protocol does not contain any information. We would like to emphasise this claim here that Message 4 : $B \rightarrow A : N'_b$ is *not* necessary for the main purpose of the protocol. That is, no security information is transferred from A to B. We, therefore, claim that message 4 can be eliminated from the procedure. The next section will show that even if this message is removed, the procedure can still accomplish the same thing as before. That is, both A and B hold the new shared key K'_{ab}.

In this section, we have mentioned the two vulnerabilities that could potentially lead to an attack on the Adapted Andrew RPC. Next section, we make an attempt to modify the Adapted Andrew RPC in order to address the weaknesses. We then give the analysis of the protocol to show that after the changes both parties, A and B, still hold the same secret key K'_{ab}.

4 Modified Andrew Secure RPC

First, the two vulnerabilities stated in the previous section will be addressed. The modified protocol will then be proved for correctness using the logic of Gong, Needham and Yahalom [GNY90, MSnN94].

4.1 The Protocol

In order to address the potential known-plaintext attack, we recommend that the first message should be encrypted using the already known shared key K_{ab}. By encrypting the first message, we get rid of the known-plaintext attack in that an attacker cannot have any plaintext and ciphertext pair anymore. For the second weakness, session-hijacking, we suggest that the identity of the sender should be a part of the message. By adding the identity, the attacker could still intercept and forward the message. However, the recipient would know who created that message, hence subsequent messages would then be sent to the legitimate party. Moreover, the fourth message of the Adapted Andrew RPC is removed from the procedure to increase the efficiency. Last, but not least, we think that the sender of the second message should add a new nonce to the message. This new nonce would act as a *fresh* challenge for the response in message 3. The resultant protocol is as follows.

$$\begin{array}{ll} \text{Message 1.} & A \rightarrow B : \{A, N_a\}_{K_{ab}} \\ \text{Message 2.} & B \rightarrow A : \{N_a, N_b, K'_{ab}, B\}_{K_{ab}} \\ \text{Message 3.} & A \rightarrow B : \{A, N_b\}_{K'_{ab}} \end{array}$$

4.2 Protocol Analysis

The section presents the analysis of the modified protocol. The GNY logic is used for the analysis. Therefore, all the postulates can be seen in [GNY90, MSnN94].

First, the modified protocol is idealised into the logic of GNY as follows.

$$\begin{array}{ll} \text{Message 1.} & B \lhd *\{*A, *N_a\}_{K_{ab}} \\ \text{Message 2.} & A \lhd *\{N_a, *N_b, *K'_{ab}, *B\}_{K_{ab}} \rightsquigarrow B \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B \\ \text{Message 3.} & B \lhd *\{*A, N_b\}_{K'_{ab}} \end{array}$$

The followings are assumptions of the Andrew secure RPC made in [BAN90]. Note that we do *not* add any new assumptions to this modified protocol.

$$
\begin{array}{ll}
A \mid\equiv A \overset{K_{ab}}{\longleftrightarrow} B & B \mid\equiv A \overset{K_{ab}}{\longleftrightarrow} B \\
A \mid\equiv B \mid\Longrightarrow A \overset{K'_{ab}}{\longleftrightarrow} B & B \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B \\
A \mid\equiv \sharp(N_a) & B \mid\equiv \sharp(N_b)
\end{array}
$$

We now carry out the GNY analysis on the protocol.

Message 1: Applying the postulates T1 and T3, we obtain $B \lhd A, N_a$. That is, B has received or has been told A and N_a. Then the postulate P1 is applied, and we obtain $B \ni A, N_a$. That is, B now possesses A's identity and nonce N_a.

Message 2: First, we note that the extension to the message, $B \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B$, is valid because it is evident from the initial assumption.

Applying the postulates T1, T3 and P1, we obtain $A \lhd N_a, N_b, K'_{ab}, B$. That is, A now possesses the nonces N_a and N_b, the new key K'_{ab} and B's identity.

Applying F1, we obtain $A \mid\equiv \sharp(N_a, N_b, K'_{ab}, B)$. That is, A believes that the message is fresh, i.e., not a replay.

Applying R1, we obtain $A \mid\equiv \phi(N_a, N_b, K'_{ab}, B)$. That is, A believes that the contents of the message is recognisable.

Applying I1, we obtain $A \mid\equiv B \mid\sim (N_a, N_b, K'_{ab}, B)$, $A \mid\equiv B \mid\sim \{N_a, N_b, K'_{ab}, B\}_{K_{ab}}$, $A \mid\equiv B \ni K_{ab}$. That is, A believes that the message is originated from B and A believes that B possesses the key K_{ab}.

Applying I6, we obtain $A \mid\equiv B \ni N_a, N_b, K'_{ab}, B$. That is, A believes that B possesses the nonces N_a and N_b, his own identity B, and the new key K'_{ab}.

Applying J2, we obtain $A \mid\equiv B \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B$. That is, A believes that B believes that K'_{ab} is a good key for A and B.

Applying J1, we obtain $A \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B$. That is, A believes that K'_{ab} is a good key for A and B.

Therefore, at the end of the second message, A possesses the new shared key K'_{ab} and A also believes that K'_{ab} is a good key shared between A and B. Furthermore, A recognises his own nonce N_a, which means that B has decrypted the first message correctly. That, in turn, means that B possesses the same key K_{ab}, hence A has authenticated B.

Message 3: Applying the postulates T1, T3 and P1, we obtain $B \ni A, N_b$. That is, B possesses A's identity and nonce N_b.

Applying F1, we obtain $B \mid\equiv \sharp\{A, N_b\}_{K'_{ab}}$. That is, B believes that the message is fresh, i.e., not a replay.

Applying R2, we obtain $B \mid\equiv \phi\{A, N_b\}_{K'_{ab}}$. That is, B believes that the contents of the message is recognisable.

Applying I1, we obtain $B \mid\equiv A \mid\sim (A, N_b)$, $B \mid\equiv A \mid\sim \{A, N_b\}_{K'_{ab}}$, $B \mid\equiv A \ni K'_{ab}$.

That is, B believes that the message is originated from A and B believes that A now possesses the new key K'_{ab}.

Here, the third message alone shows that B recognises his own nonce N_b, which means that A has decrypted the second message correctly. That, in turn, means that A possesses the key K_{ab}, hence B has authenticated A. Furthermore, B now believes that A also holds the new secret ey K'_{ab}, which is the same as the one B is holding.

On the whole, at the end of the protocol run, we obtain:

$$A \mid\equiv A \overset{K'_{ab}}{\longleftrightarrow} B \quad \text{and} \quad B \mid\equiv A \ni K'_{ab}.$$

This means that both A and B are now holding the new secret key K'_{ab}. A and B both believe that the new key K'_{ab} is a good key for subsequent communications. Moreover, A and B know that the other party possesses K'_{ab} as well.

5 Conclusions

We have presented an overview of the three variations of the Andrew secure RPC. They include original Andrew secure RPC, the Andrew secure RPC after BAN analysis and the Adapted Andrew RPC. We have also shown that weaknesses have been discovered and exploited in the original Andrew secure RPC and the Andrew secure RPC after BAN.

In this paper, a couple of vulnerabilities have been found in the adapted Andrew RPC. Those vulnerabilities could potentially lead to a known-plaintext attack as well as a session-hijack. The problem of known-plaintext has been addressed by encrypting the first message. The problem of session hijacking in the third message of the Adapted Andrew RPC has been fixed by adding the identity of the sender as part of the message. Furthermore, we have recommended that the sender of the second message should add a newly generated nonce to the message in order to make the authentication challenge fresh. In addition to those weaknesses, the efficiency of the protocol has been considered. It has been mentioned in this paper that the fourth message of the Andrew secure RPC has no use in security at all. We have, therefore, suggested that it should be removed from the protocol.

Having designed a protocol to address all the vulnerabilities mentioned in this paper, a GNY analysis on the resultant protocol has been carried out. It has been pointed out that despite the modifications made, the outcomes of the protocol have not been altered. That is, the protocol achieves mutual authentication. Both parties involved in the protocol run end up holding the same new secret key. Finally, they both believe that the new key is good for encrypting and decrypting subsequent messages, and they both believe that the other party possesses the new secret key as well.

References

[BAN90] Michael Burrows, Mart Abadi und Roger Needham. A logic of authentication. *ACM Transactions on Computer Systems*, 8:18–36, 1990.

[BBG+93] R. Bird, R. Bird, I. Gopal, I. Gopal, A. Herzberg, A. Herzberg, P. Janson, P. Janson, S. Kutten, S. Kutten, R. Molva, R. Molva, M. Yung und M. Yung. Systematic Design of a Family of Attack-Resistant Authentication Protocols, 1993.

[BGH+91] Ray Bird, Inder Gopal, Amir Herzberg, Phil Janson, Shay Kutten, Refik Molva und Moti Yung. Systematic Design of Two-Party Authentication Protocols, 1991.

[BM03] Colin Boyd und Anish Mathuria. *Protocols for Authentication and Key Establishment*. Springer, Berlin; London, 2003.

[CJ95] John Clark und Jeremy Jacob. On the Security of Recent Protocols. *Information Processing Letters*, 56:151–155, 1995.

[GNY90] Li Gong, Roger Needham und Raphael Yahalom. Reasoning about Belief in Cryptographic Protocols. In *Proceedings 1990 IEEE Symposium on Research in Security and Privacy*, Seiten 234–248. IEEE Computer Society Press, 1990.

[Low96] Gavin Lowe. Some New Attacks upon Security Protocols. Seiten 162–169. Society Press, 1996.

[MSnN94] Anish Mathuria, Reihaneh Safavi-naini und Peter Nickolas. Some Remarks on the Logic of Gong, Needham and Yahalom, 1994.

[Sat89] M. Satyanarayanan. Integrating Security in a Large Distributed System. *ACM Transactions on Computer Systems*, 7:247–280, 1989.