Generalized Unsolicited Tests for Authentication Protocol Analysis *

Yongjian Li[1,2] † Jun Pang[3]

[1]Chinese Academy of Sciences, Institute of Software
Laboratory of Computer Science
[2]The State Key Laboratory of Information Security
P.O.Box 8718, Beijing, China
lyj238@ios.ac.cn

[3]University of Oldenburg
Department of Computer Science
Safety-critical Embedded Systems
26111 Oldenburg, Germany
jun.pang@informatik.uni-oldenburg.de

Abstract

Guttman and Thayer Fábrega introduced the notion of unsolicited authentication tests, and used it to prove the correctness of security protocols in which a key server authenticate its clients. As an example, they have applied unsolicited authentication tests to prove the authentication goals of the Otway-Rees protocol. However, unsolicited authentication tests seem not to be fully explored in that case study, and the proofs were complicated. In this paper, we revisit the unsolicited authentication tests, and show how to strengthen and apply them in more general cases. To justify our work, we also use this extension to prove all agents' authentication guarantee of the Otway-Rees protocol.

Keywords: *strand space, authentication, cryptographic protocols, the Otway-Rees protocol*

1 Introduction

A cryptographic protocol is a series of carefully message exchanging among two or more participants. These messages are often encrypted. Cryptographic protocols are designed to achieve specified goals like authentication and key distribution, even with the presence of a penetrator who can perform malicious actions. However, the design of these protocols is error-prone. Incorrectly designed protocols may become ideal entry points for various attacks. Therefore, we cannot only rely on informal ways of reasoning about their correctness. On the other hand, formal methods are mathematically based techniques for specifying and verifying systems and protocols. Their mathematical underpinning allows formal methods to analyze systems in a more precise and non-ambiguous fashion. This makes it possible to use formal description and verification techniques to obtain assurance that a protocol cannot be attacked by a penetrator.

Thayer Fábrega *et al.* developed the framework of strand spaces [7] for verifying security protocols. For a legitimate regular participant, a strand s represents a sequence of message that the participant would receive or send as part of a run as his/her role of the protocol. A typical message has the form of $\{\!|h|\!\}_K$ denoting the encryption of h using key K. An element of the set of messages is called a *term*. A term t' is a subterm of t is written as $t' \sqsubset t$. Usually, we call a strand element *node*. Nodes can be either positive, representting the transmission of a term, or negative, representing the reception of a term). For the penetrator, the strand represents atomic deductions. More complex deductions can be formed by connecting several penetrator strands. Hence, a strand space is simply a set of strands with a trace mapping. Two kinds of casual relation (arrow), $\rightarrow$ and $\Rightarrow$, are introduced to impose a graphic structure on the nodes of the space. The relation $\preceq$ is defined to be the reflexive and transitive closure these two arrows, modeling the casual order of the events in the protocol execution. The formal analysis based on strand spaces can be carried on the notion of bundles. A bundle is a casually well-founded set of nodes and the two arrows, which sufficiently formalizes a session of a protocol. In a bundle, it must be ensured that a node is included only if all nodes that proceed it are already included. For the strand corresponding to a principal in a given protocol run, we construct all possible bundles containing nodes of the strand. In fact, this set of bundles encodes all possible interactions of the environment with that principal in the

*The first author is supported by grants (No.60173020, 60421001) from National Natural Science Foundation of China.

†Corresponding author. Postal address: Chinese Academy of Sciences, Institute of Software, Laboratory of Computer Science, 4# South Fourth Street, Zhong Guan Cun, Beijing, China. Phone: +86-(10)62644486. Fax: +86-(10)62563894.

run. Normally, reasoning about the protocol takes place on this set of bundles.

To make strand space easy to apply, Guttman and Thayer Fábrega [1] introduced three kinds of authentication tests, namely *outgoing*, *incoming* and *unsolicited* tests, to prove authentication and secrecy properties for a wide range of security protocols. Among them, unsolicited authentication tests are mainly used to prove that a key server authenticates its clients. It was applied to prove a server's guarantee in the Otway-Rees protocol [4]. (The message exchanging process in the Otway-Rees protocol is presented in Figure 1.) But their proofs in [1] for an initiator's and a responder's authentication guarantee depend on the result of outgoing authentication tests and a side assumption requiring that no proper encrypted subterms are contained in the forwarding component H, which is corresponding to $\{\!|M, N_a, A, B|\!\}_{K_A}$ in the first and second messages of a responder (see Figure 1). However, this side assumption is not realistic since a responder cannot enforce such a constraint. In the intended case, H is a term encrypted by the initiator's long-term key, which is unintelligible to the responder. To remedy this deficiency, Guttman and Thayer Fábrega devoted one section (Section 5.1.3, [1]) in their paper to show that this constraint does not hide any attacks. In particular, if the penetrator can succeed without this restriction, then they can also succeed if this constrain is enforced. Their proof is rather complicated. First, they need to introduce another notion of *nearly equivalence* between a constrained Otway-Rees bundle and an unconstrained Otway-Rees bundle. Second, they need an intermediate result showing that a *nearly equivalent* constrained Otway-Rees bundle can be constructed from a unconstrained Otway-Rees bundle.

In this paper, we generalize the notion of unsolicited authentication tests and give simpler proofs for authentication goals of the Otway-Rees protocol. The proofs still make use of results of unsolicited authentication tests in [1], but they differ from the proofs in [1] in several ways. We summarize our main contributions in this paper as follows:

- We use unsolicited authentication tests to prove regularity of nodes, namely that once $\{\!|h|\!\}_K$ occurs as a subterm of a node n in a bundle $\mathcal{B}$, and if the key K cannot be penetrated in this bundle, then there is a regular node m originating $\{\!|h|\!\}_K$. Moreover, we strengthen the result by additionally asserting that if m is the node originating $\{\!|h|\!\}_K$ as a subterm, and $\{\!|h|\!\}_K \not\sqsubset term(m')$ for any node $m' \preceq_{\mathcal{B}} m$. This strengthened property turns out to be very useful for security protocol analysis.

- Combining unicity property of a nonce in a nonce-based protocol with unsolicited authentication test, we review the Otway-Rees protocol and only use the results of unsolicited authentication tests to prove an initiator's and a responder's guarantees. In particular, we do not need the aforementioned side assumption and outgoing authentication tests in [1]. Our main result, which is different from the one of Guttman and Thayer Fábrega, lies in that we extend the result of a server's guarantees, in turn which can be used to prove the guarantees of an initiator and a responder.

- Furthermore, we formalize the theory of our unsolicited authentication tests and check the proofs in this paper using the theorem prover Isabelle/HOL [3].

Related work. There has been a large body of papers on applying formal methods to security protocol analysis. Among them, we find the work by Perrig and Song [8, 6] based on the strand space model closely related to ours. Athena [8, 6], which is an automatic tool for automatic security protocol generation, has incorporated the authentication tests in [1]. Athena can also check secrecy properties. Several efficient methods have been developed in order to increase the performance of Athena. Instead of supporting automatic generation of security protocols, we have an extension of the unsolicited authentication tests and formalize the theory in Isabelle/HOL [3], which can be an automatic framework for proving the correctness of authentication protocols.

Structure of the paper. In Section 2 we briefly review the basic concepts of the strand space model. We develop our notion of unsolicited authentication test in Section 3, and apply it to prove all agents' authentication guarantees of the Otway-Rees protocol in Section 4. Finally, we draw some concluding remarks in Section 5.

2 Preliminaries

In this section, we present some preliminaries of the strand space model. More detailed description can be found in [7].

2.1 Messages and actions

The set of messages is defined as the following BNF notation:

$$\begin{array}{rcl} h & ::= & \mathbf{name}(A) \mid \mathbf{nonce}(n) \\ & \mid & \mathbf{key}(K) \mid \{\!|h_1, h_2|\!\} \\ & \mid & \mathbf{enc}(h, K) \end{array}$$

where A is an element from a set of agents, n from a set of nonces, and K from a set of keys. We call a key symmetric if $K^{-1} = K$. Otherwise, K is a private key and K^{-1} is public. $\{\!|h_1, h_2|\!\}$ is called a composed message.

$\{\!|h_1, h_2|\!\} = \{\!|h_1', h_2'|\!\}$ if and only if $h_1 = h_1'$ and $h_2 = h_2'$. We abbreviate $\mathbf{enc}(h, K)$ as $\{\!|h|\!\}_K$, denoting the encryption of h using key K. In our formulation, we use function K_A to define a long-term key shared between an agent (or a client) A and a server, and this function K_A is injective for any A, i.e. if $K_A = K_{A'}$, then $A = A'$. An element of the set of messages is also called a *term*. Terms of the form $\mathbf{name}(A)$, $\mathbf{nonce}(n)$, or $\mathbf{key}(K)$ are said to be atomic. [1] The set of all messages is denoted by $\mathbf{Message}$. A message h is a text message if $h \neq \mathbf{key}(K)$ for any K. The set of all atomic text messages is denoted by T. We frequently need the subterm relation on messages. A term t' is a subterm of t is written as $t' \sqsubset t$.

Definition 1 *The subterm relation $\sqsubset$ is defined inductively as the smallest relation such that $t \sqsubset t$, $t \sqsubset \{\!|h|\!\}_K$ if $t \sqsubset h$, and $t \sqsubset \{\!|h_1, h_2|\!\}$ if $t \sqsubset h_1$ or $t \sqsubset h_2$.*

The transmission of a term t is denoted by $(+, t)$, and the reception of a term t is denoted by $(-, t)$. Both are the possible actions that participants and the penetrator can take. We represent the set of finite sequences of actions by $(\pm, \mathbf{Message})^*$.

2.2 Strands and strand spaces

A protocol defines the sequence of events (message transmission and reception) for each role of the participant. For a legitimate participant, a strand s represents a sequence of message that the participant would receive or send as part of a run as an instance of his role of the protocol.

A strand space Σ is a set of strands with a trace mapping $tr : \Sigma \to (\pm, \mathbf{Message})^*$. A strand element is called a node. (s, i) is the i-th node on strand s ($0 \leq i < length(s)$). We use $n \in s$ to denote that a node n belongs to the strand s. If $n = (s, i)$ and $tr(s)_i = (\sigma, t)$, then we define $term(n)$ and $sign(n)$ to be the term and sign of the node n respectively, namely $term(n) = t$, $sign(n) = \sigma$. We call a node positive if its term has sign $+$, and negative if its term has sign $-$. A strand is a protocol history from the receipt of a single peer of an agent in a protocol run, so we explicitly define an attribute function $attr : \Sigma \to A$ to indicate which agent's peer a strand is, namely, $attr(s) = a$ means that a is the agent who do actions of the strand s in the run.

Two kinds of casual relation (arrow), $\to$ and $\Rightarrow$, are introduced to impose a graphic structure on the nodes of Σ. The relation $n \Rightarrow n'$ holds between nodes n and n' if $n = (s, i)$ and $n' = (s, i + 1)$. This relation corresponds to the casual ordering of actions on the same strand. On the other hand, the relation $n \to n'$ holds for nodes n and n' if $term(n) = term(n')$, $sign(n) = +$ and $sign(n') = -$ for some term t. This represents that n sends a message t and n' receives the message. The relation $\preceq$ is defined to be the reflexive and transitive closure of $\to$ and $\Rightarrow$, modeling the casual order of the events in the protocol execution. We say that a term t *originates* at a node n if and only if n is positive and $t \sqsubset term(n)$; t uniquely originates from node n if and only if t originates on a unique node n. Nonces and other freshly generated terms such as session keys are usually uniquely originated. We say that t uniquely originates if and only if there exists a node n such that t uniquely originates from node n.

[1] We often write A, n, and K instead of $\mathbf{name}(A)$, $\mathbf{nonce}(n)$, and $\mathbf{key}(K)$.

2.3 Penetrator strands

The symbol $\mathbb{P}$ is defined to denote the set of all the penetrators, and if an agent is not in $\mathbb{P}$, then it is regular. There is a set of keys that are known initially to all the penetrators, denoted as $\mathbf{K}_{\mathcal{P}}$. $\mathbf{K}_{\mathcal{P}}$ usually contains all the public keys, all the private keys of all the penetrators, and all the symmetric keys initially shared between all the penetrators and principals playing by the protocol rules. It can also contain some keys to model known-key attacks. A penetrator can intercept messages, generate messages that are computable from its initial knowledge and the messages it intercepts. These actions are modelled by a set of penetrator strands, and they represent atomic deductions. More complex deduction actions can be formed by connecting several penetrator strands. In our theory, we assume that penetrators share their initial knowledge and can cooperate each other by composing their strands.

Definition 2 *A penetrator' trace relative to $\mathbf{K}_{\mathcal{P}}$ is one of the following:*

- M_t *(text message):* $[(+, t)]$, *where* $t \in T$.
- K_K *(key):* $[(+, K)]$, *where* $K \in \mathbf{K}_{\mathcal{P}}$.
- $C_{g,h}$ *(concatenation):* $[(-, g), (-, h), (+, \{\!|g, h|\!\})]$.
- $S_{g,h}$ *(separation):* $[(-, \{\!|g, h|\!\}), (+, g), (+, h)]$.
- $E_{h,K}$ *(encryption):* $[(-, K), (-, h), (+, \{\!|h|\!\}_K)]$.
- $D_{h,K}$ *(decryption):* $[(-, K^{-1}), (-, \{\!|h|\!\}_K), (+, h)]$.

In our theory, if a strand s belongs to a penetrator, namely, $attr(s) \in \mathbb{P}$, then s must be a penetrator strand. A node is called *regular* if it is not in the penetrator strands.

2.4 Bundles

The formal analysis based on strand spaces is carried on the notion of bundles, which represents the protocol execution under some configuration. A bundle is a casually well-founded set of nodes and the two types of arrows $\to$ and $\Rightarrow$,

which sufficiently formalizes a session of a protocol. In a bundle, it must be ensured that a node is included only if all nodes that proceed it are already included. Suppose $\mathcal{B}$ is a bundle, we use $n \in \mathcal{B}$ if n is a node in $\mathcal{B}$, and use $\preceq_{\mathcal{B}}$ to denote the transitive closure of the relation $\rightarrow$ and $\Rightarrow$ in $\mathcal{B}$. $\mathcal{B}$ has the following properties:

- $\mathcal{B}$ is a finite graph;
- If the sign of a node n is $-$, and $n \in \mathcal{B}$, then there is a unique positive node n' such that $n' \rightarrow n$ and $n' \in \mathcal{B}$;
- If $n' \Rightarrow n$ and $n \in \mathcal{B}$, then $n' \in \mathcal{B}$ and $n' \Rightarrow n \in \mathcal{B}$.
- $\mathcal{B}$ is acyclic.

Lemma 1 (Bundle well foundedness, [7]) *Let $\mathcal{B}$ be a bundle. Then $\preceq_{\mathcal{B}}$ is a partial order. Every non-empty subset of the nodes in $\mathcal{B}$ has $\preceq_{\mathcal{B}}$ minimal members.*

We have used the theorem prover Isabelle/HOL to prove that a bundle $\mathcal{B}$ is up-wards closed under $\preceq_{\mathcal{B}}$, those standard properties, and Lemma 1 [9].

3 Unsolicited Authentication Tests Revisited

Unsolicited authentication tests are frequently used to prove that a server authenticate its clients. In this section, we develop our notion of unsolicited authentication tests. In order to explain why our notion is different from its original form, we need to present how Guttman and Javier Thâyer defined their notion in (Section 4.2.3, [1]). They first define *unsolicited tests*, i.e. a negative node n is an unsolicited test for $\{\!|h|\!\}_K$, if $\{\!|h|\!\}_K$ is a *test component* for any atomic text a in n, and K cannot be penetrated in the strand space. Then, they claim that an unsolicited test for $\{\!|h|\!\}_K$ in a bundle $\mathcal{B}$ can guarantee the existence of a positive regular node of which $\{\!|h|\!\}_K$ is a component. We simplify this definition of unsolicited tests by the following two aspects:

1. we consider a node n is an unsolicited test for $\{\!|h|\!\}_K$ in a bundle $\mathcal{B}$;
2. we only require that $\{\!|h|\!\}_K$ is a subterm of the term of n, and K is regular in the bundle $\mathcal{B}$ (instead of a strand space).

We claim that the existence of this newly defined unsolicited test for $\{\!|h|\!\}_K$ in a bundle $\mathcal{B}$ can guarantee the existence of a (positive) regular node m, which originates $\{\!|h|\!\}_K$ as a subterm.

First we need to define a key is regular in a bundle.

Definition 3 *A key K is* regular *in a bundle $\mathcal{B}$ if and only if the following condition holds: for any node n in $\mathcal{B}$, if $term(n) = K$, then n must be regular.*

Note that we are mainly interested in the fact that K cannot be penetrated in a bundle that we are considering. This is rather different from the notions of *penetrable keys* or *safe keys* in [1], where Guttman and Thayer Fábrega considered whether a key can potentially be penetrated in a strand space. In most cases, we only consider security properties for a protocol in a given bundle, so it is natural for us to just consider whether a key can potentially be penetrated in this bundle.

In our formulation, unsolicited authentication test is a kind of regularity about an encrypted term $\{\!|h|\!\}_K$, where K is a long-term regular key (e.g. a shared key between a regular agent and the server in a symmetric setting). Once $\{\!|h|\!\}_K$ occurs as a subterm of a node n in a bundle $\mathcal{B}$, it can be ensured that there is a positive regular node m originating $\{\!|h|\!\}_K$ as a subterm, i.e. m has $\{\!|h|\!\}_K$ as a subterm, and it also holds that $\{\!|h|\!\}_K \not\sqsubset term(m')$ for any node $m' \preceq_{\mathcal{B}} m$. Intuitively, the reason why m must be regular lies in that k cannot be penetrated in $\mathcal{B}$. So the penetrator cannot create $\{\!|h|\!\}_K$ by encrypting h with K.

Definition 4 (Unsolicited test) *Given a bundle $\mathcal{B}$. A node n in $\mathcal{B}$ is an* unsolicited test *for $\{\!|h|\!\}_K$ if $\{\!|h|\!\}_K \sqsubset term(n)$ and K is regular in $\mathcal{B}$.*

Lemma 2 (Unsolicited authentication test) *Given a bundle $\mathcal{B}$. Let n be an unsolicited test for $\{\!|h|\!\}_K$. Then there exists a positive regular node m in $\mathcal{B}$ such that $\{\!|h|\!\}_K \sqsubset term(m)$ and $\{\!|h|\!\}_K \not\sqsubset term(m')$ for any node m' such that $m' \preceq_{\mathcal{B}} m$.*

Proof. Let

$$P =_{df} \{x \mid x \in \mathcal{B} \wedge \{\!|h|\!\}_K \sqsubset term(x)\}.$$

Obviously, $n \in P$. By the well-foundedness of a bundle, i.e. there exists a node m such that m is minimal in P, which means $\{\!|h|\!\}_K \sqsubset term(m)$, $m \in \mathcal{B}$, and for all $m' \in \mathcal{B}$, if $m' \preceq_{\mathcal{B}} m$ then $m' \notin P$ and $\{\!|h|\!\}_K \not\sqsubset term(m')$.

First, we prove that the sign of m is positive. If $sign(m) = -$, then by upward-closed property of a bundle there must be another node m'' in $\mathcal{B}$ such that $sign(m'') = +$ and $m'' \rightarrow m'$. This contradicts with the minimality of m.

Second, we prove that m is regular by deriving contradictions if m is in a penetrator strand. Here we only analyze the cases when m is in either $C_{g,g'}$ (concatenation strand) or $E_{g,K}$ (encryption strand). Other cases are either straightforward or can be analyzed in a similar way.

- CASE 1: m is in $i \in C_{g,g'}$.

  By the form of the strand $C_{g,g'}$ and the fact that m is a positive node, we have $m = (i, 2)$, $term(m') = \{\!|g, g'|\!\}$, $term\ (i, 0) = g$, and $term\ (i, 1) = g'$ for

some g, g'. By the upwards-closed property of a bundle, we have that nodes $(i,0)$ and $(i,1)$ must be in $\mathcal{B}$. By $\{\!|h|\!\}_K \sqsubset \{\!|g,g'|\!\}$, we have either $\{\!|h|\!\}_K \sqsubset g$ or $\{\!|h|\!\}_K \sqsubset g'$. So either node $(i,0) \in P$, or node $(i,1) \in P$. Both contradict with the minimality of m.

- CASE 2: m is in $i \in E_{g,K'}$.

  By the form of the strand $E_{g,K'}$ and the fact that m is a positive node, we have $m = (i,2)$, $term(m) = \{\!|g|\!\}_{K'}$, $term(i,0) = K'$, and $term(i,1) = g$ for some g, K'. So $\{\!|h|\!\}_K \sqsubset \{\!|g|\!\}_{K'}$. Hence, it is straightforward that either (1) $\{\!|h|\!\}_K \sqsubset g$ or (2) $h = g$ and $K = K'$. For (1), we have $\{\!|h|\!\}_K \sqsubset term(i,1)$. It is easy to derive a contradiction by the same argument as in CASE 1. For (2), by the assumption that K must be regular in $\mathcal{B}$, $term(i,0)$ must be regular, and this contradicts with the fact that i is a penetrator strand.

■

The proof totally depends on the well-founded induction principle on bundles, and we have formalized the proof of this lemma in Isabelle/HOL [3] in our inductive strand space model [2], and the proof scripts can be obtained at [9]. Although the proof is not difficult, we find that this extension of unsolicited authentication test can be applied to more general cases. The evidence is our new proofs for authentication goals for the Otway-Rees protocol in next section.

4 Example: The Otway-Rees Protocol

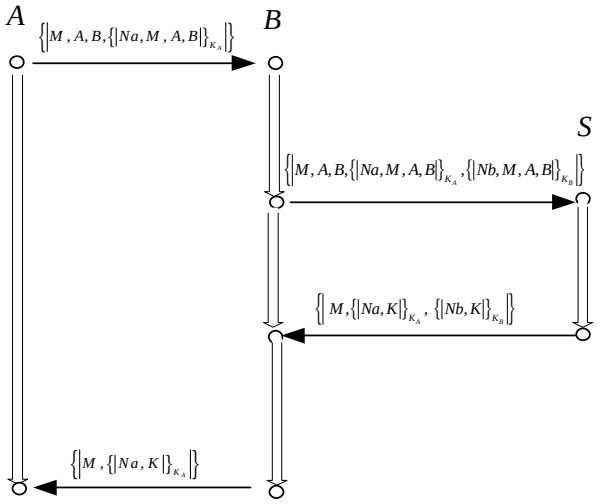

Figure 1. Message exchanging in the Otway-Rees protocol

The Otway-Rees protocol [4] (see Figure 1) uses a long-term symmetric keys shared with the server and its clients to distribute a new session key for a conversation between two clients. For our convenience, we will use $Init[A,B,N_a,M,K]$ to denote the set of all initiator strands of the Otway-Rees protocol with initiator A, responder B, nonce N_a, round number M, and session key K. Similarly we define the set of all responder strands as $Resp[A,B,N_b,M,K,H,H']$ and the set of all server strands as $Serv[A,B,N_a,N_b,M,K]$. In the following discussion, we will also use $s \in Init[A,B,N_a,M,*]$ to denote $\exists K.s \in Init[A,B,N_a,M,K]$, the set of all initiator strands with A, B, N_a, M, and any value of K. we can define the same notations for the other two kinds of stands, respectively. We will also abbreviate a form like $Init[A,B,*,*,*]$ to $Init[A,B,**]$. The regular strands are defined as follows:

- For $s \in Init[A,B,N_a,M,K]$, its trace is of the form:
$$\left[\begin{array}{l}(+,\{\!|M,A,B,\{\!|N,M,A,B|\!\}_{K_A}|\!\}),\\(-,\{\!|M,\{\!|N_a,K|\!\}_{K_A}|\!\})\end{array}\right]$$
- For $r \in Resp[A,B,N_b,M,K,H,H']$, its trace is of the form:
$$\left[\begin{array}{l}(-,\{\!|M,A,B,H),\\(+,\{\!|M,A,B,H,\{\!|N_b,M,A,B|\!\}_{K_B}|\!\}),\\(-,\{\!|M,H',\{\!|N_b,K|\!\}_{K_B}|\!\}),\\(+,\{\!|M,H'|\!\})\end{array}\right]$$
- For $s \in Server[A,B,N_a,N_b,M,K]$, its trace is of the form:
$$\left[\begin{array}{l}(-,\{\!|M,A,B,\{\!|N_a,M,A,B|\!\}_{K_A},\\\qquad\{\!|N_b,M,A,B|\!\}_{K_B}|\!\}),\\(+,\{\!|M,\{\!|N_a,K|\!\}_{K_A},\{\!|N_b,K|\!\}_{K_B}|\!\})\end{array}\right]$$

In our proofs we will implicitly use three axioms on the Otway-Rees protocol: The first specifies that a regular strand can only be an initiator or a responder or a server strand in one Otway-Rees protocol strand space; the second specifies that if an agent is not a penetrator then his shared key is not in the initial knowledge of the penetrators; the third specifies that the server in the Otway-Rees protocol distributes a new session key which are not agents' long term shared key with the server.

Axiom 1 *A regular strand can only be an initiator or a responder or a server strand in an Otway-Rees protocol strand space.*

Axiom 2 *If $A \notin \mathbb{P}$, then $K_A \notin \mathbf{K}_{\mathcal{P}}$.*

Axiom 3 *For any server strand s such that $s \in Server[A,B,N_a,N_b,M,K]$. Then $K \neq K_C$ for any regular agent C.*

In the following discussion, we assume $\mathcal{B}$ as a bundle of the Otway-Rees strand space. In order to prove the main results of authentication guarantees, we need some auxiliary results first.

For any node $n \in \mathcal{B}$, $term(n)$ cannot be a long-term symmetric key of a regular agent, because no regular key are sent as a part of a message in the Otway-Rees protocol.

Lemma 3 *Let $n \in \mathcal{B}$, if $A \notin \mathbb{P}$, then $K_A \not\sqsubset term(n)$.*

Proof. Assuem $A \notin \mathbb{P}$. Let

$$P =_{df} \{x \mid x \in \mathcal{B} \wedge K_A \sqsubset term(x)\}$$

We show that P is empty by contradiction. If there is a node $n \in P$, then by the well-foundedness of a bundle, there exists a node m such that m is minimal in P. Namely, $m \in \mathcal{B}$, $K_A \sqsubset term(m)$, and for all $m' \in \mathcal{B}$, if $m' \preceq_{\mathcal{B}} m$ then $m' \notin P$ and $K_A \not\sqsubset term(m')$.

We prove that the sign of m is positive. If $sign(m) = -$, then by upward-closed property of a bundle there must be another node m'' in the bundle $\mathcal{B}$ such that $sign(m'') = +$ and $m'' \rightarrow m$. This contradicts with the minimality of m. Then m is either in a regular strand or in a penetrator strand.

- CASE 1: m is in a regular strand.

  Then by Axiom 1, there are three cases. Here we only analyze the case when m is in a server strand $s \in Server[A, B, N_a, N_b, M, K]$. The other two cases are either straightforward or can be analyzed in a similar way. By inspection on the trace form of a server strand, we have $m' = (s, 1)$, $K_A \sqsubset term(s, 1)$, and $term(s, 1) = \{\!|M, \{\!|N_a, K|\!\}_{K_A}, \{\!|N_b, K|\!\}_{K_B}|\!\}$. Therefore, $K_A = K$. This contradicts with Axiom 3.

- CASE 2: m is in a penetrator strand p.

  Here we only analyze the cases when p is either K_K (key strand) or $C_{g,h}$ (concatenation). Other cases are either straightforward or can be analyzed in a similar way.

  - p is K_K. We have $m = (p, 0)$ and $K_A \sqsubset K$. Then $K_A = K \in \mathbf{K}_{\mathcal{P}}$. This contradicts with Axiom 2.
  - p is $C_{g,h}$. We have $m' = (p, 2)$ and $K_A \sqsubset \{\!|g, h|\!\}$. By the definition of $\sqsubset$, we have $K_A \sqsubset g$, or $K_A \sqsubset h$. If $K_A \sqsubset g$, then $K_A \sqsubset term(p, 0)$. This contradicts with the minimality of m. The case when $K_A \sqsubset h$ can be analyzed similarly.

■

Following this lemma, it is easy to prove that a long-term symmetric key of a regular agent cannot be penetrated in the bundle.

Lemma 4 *If $A \notin \mathbb{P}$, then K_A is regular in $\mathcal{B}$.*

As in [1], we assume a nonce originates uniquely in some strand space. If N_a originates uniquely, and $i \in Init[A, B, N_a, M, K]$, then the nonce can uniquely identify this strand i, which means if another initiator strand i' satisfies $i' \in Init[A', B', N_a, M', K']$, then $i = i'$. This is captured by the following lemma.

Lemma 5 *If some nonce N_a originates uniquely, $i \in Init[A, B, N_a, M, K]$ and $i' \in Init[A', B', N_a, M', K']$, then $i = i'$, i.e. $A = A'$, $B = B'$, $M = M'$, and $K = K'$.*

Now we come to prove the server's authentication guarantees. Our main technique is Lemma 2 in Section 3. The main differences between our proofs and the original proof of Guttman and Thayer Fábrega lie in the guarantees of the existence of a server $s \in Serv[A, B, N_a, N_b, M, *]$. Guttman and Thayer Fábrega only analyzed the case when $A \neq B$, while we consider more general cases without the restriction $A \neq B$. We show that if there is either a server $s \in Serv[A, B, N_a, N_b, M, *]$ in $\mathcal{B}$ and A is regular, then there is a regular initiator $i \in Init[A, B, N_a, M, *]$, or a regular responder $r \in Resp[A, B, N_a, M, **]$ with $A = B$.

Lemma 6 (Server's gunrantee 1) *Suppose $A \notin \mathbb{P}$, $s \in Serv[A, B, N_a, N_b, M, *]$, and $(s, 0) \in \mathcal{B}$. Then there exists either $i \in Init[A, B, N_a, M, *]$ and $(i, 0) \in \mathcal{B}$; or $r \in Resp[A, B, N_a, M, **]$ with $A = B$, and $(r, 1) \in \mathcal{B}$.*

Proof. Suppose we have $n = (s, 0)$, and term of $(s, 0)$ is $\{\!|M, A, B, \{\!|N_a, M, A, B|\!\}_{K_A}, \{\!|N_b, M, A, B|\!\}_{K_B}|\!\}$. By Lemma 4 and the facts that $A \notin \mathbb{P}$, K_A is regular. So n is an unsolicited test for $\{\!|N_a, M, A, B|\!\}_{K_A}$. Therefore, by Lemma 2 there is a positive regular node m such that $\{\!|N_a, M, A, B|\!\}_{K_A} \sqsubset term(m)$, and $\{\!|N_a, M, A, B|\!\}_{K_A} \not\sqsubset term(m')$ for all m' such that $m' \preceq_{\mathcal{B}} m$.

By the trace form of regular strands, we have either (1) m is in an initiator strand $i \in Init[A', B', N'_a, M', K']$ for some A', B', N'_a, M', K', or (2) m is in a responder strand $r \in Resp[A', B', N'_b, M', K', H, H']$ for some $A', B', N'_b, M', K', H, H'$.

If (1) holds, then by inspection on the trace form of an initiator strand, $m = (i, 0)$, and $\{\!|N'_a, M', A', B'|\!\}_{K_{A'}} = \{\!|N_a, M, A, B|\!\}_{K_A}$, then $N'_a = N_a$, $M' = M$, $A' = A$, $B' = B$.

If (2) holds, then by inspection on the trace form of a responder strand, either $m = (r, 1)$ or $m = (r, 3)$. $m = (r, 3)$ is not possible. Otherwise, $\{\!|N_a, M, A, B|\!\}_{K_A} \sqsubset H'$. However, H' also occurs in $(r, 2)$. We have $m = (r, 1)$, then either (i) $\{\!|N_a, M, A, B|\!\}_{K_A} \sqsubset H$ (ii) or $\{\!|N'_b, M', A', B'|\!\}_{K_{B'}} = \{\!|N_a, M, A, B|\!\}_{K_A}$. (i) is not possible, since H also occurs in $(r, 0)$. So (ii) must hold, then we have $N'_b = N_a$, $M' = M$, $A' = A$, $B' = B$,

$K_{B'} = K_A$. By the injectivity of $K_{B'}$ and K_A, we have $B' = A$. Then $A = B$. ■

Note that if we strengthen the assumptions of Lemma 6 with $A \neq B$, then the second case of the conclusion of Lemma 6 can be excluded.

Lemma 7 (Server's gunrantee to an initiator) *Suppose $A \notin \mathbb{P}$, $A \neq B$, $s \in Serv[A, B, N_a, N_b, M, *]$, and $(s, 0) \in \mathcal{B}$. Then there exists $i \in Init[A, B, N_a, M, *]$ and $(i, 0) \in \mathcal{B}$.*

Similar to Lemma 6, we can also prove a server's guarantee using the unsolicited test $\{\!|N_b, M, A, B|\!\}_{K_B}$. By the assumption that a server $s \in Serv[A, B, N_a, N_b, M, *]$ exists in a bundle $\mathcal{B}$ and the fact that K_B is regular, there is a regular responder $r \in Resp[A, B, N_b, M, **]$, or a regular initiator $i \in Init[A, B, N_b, M, *]$ with $A = B$.

Lemma 8 (Server's gunrantee 2) *Suppose $B \notin \mathbb{P}$, $s \in Serv[A, B, N_a, N_b, M, *]$, and $(s, 0) \in \mathcal{B}$. Then there exists either $r \in Resp[A, B, N_b, M, **]$ and $(r, 1) \in \mathcal{B}$; or $i \in Init[A, B, N_b, M, *]$ with $A = B$, and $(i, 0) \in \mathcal{B}$.*

If we require $A \neq B$, we can also exclude the second part of the conclusion in Lemma 8.

Lemma 9 (Server's gunrantee to a responder) *Suppose $B \notin \mathbb{P}$, $A \neq B$, and $s \in Serv[A, B, N_a, N_b, M, *]$, and $(s, 0) \in \mathcal{B}$. Then there exists $r \in Resp[A, B, N_b, M, **]$, and $(r, 1) \in \mathcal{B}$.*

In order to prove the authentication guarantee of an initiator $i \in Init[A, B, N_a, M, K]$ with $A \neq B$, we can use $\{\!|N_a, K|\!\}_{K_A}$ as an unsolicited test to prove the existence of a server $s \in Serv[A', B', N'_a, *, M', K']$. Then with the above results of the guarantee of s, and the unique-origination of N_a, we can ensure that $N'_a = N_a$, $K' = K$, $A' = A$, $M' = M$, and $B' = B$.

Lemma 10 (Initiator's guarantee) *Suppose $A \notin \mathbb{P}$, $A \neq B$, $i \in Init[A, B, N_a, M, K]$, $(i, 1) \in \mathcal{B}$, and N_a originates uniquely. Then there exists $s \in Serv[A, B, N_a, *, M, K]$, and $(s, 1) \in \mathcal{B}$.*

Proof. Suppose $n = (i, 1)$, term of $(i, 1)$ is $\{\!|M, \{\!|N_a, K|\!\}_{K_A}|\!\}$. $A \notin \mathbb{P}$, by Lemma 4 K_A is regular. Hence, $\{\!|N_a, K|\!\}_{K_A}$ is an unsolicited test. By Lemma 2, there is a positive regular node m such that $\{\!|N_a, K|\!\}_{K_A} \sqsubset term(m)$, and $\{\!|N_a, K|\!\}_{K_A} \not\sqsubset term(m')$ for all m' such that $m' \preceq_{\mathcal{B}} m$.

By the trace form of regular strands, m cannot be in an initiator's strand because no positive node has a subterm of the form $\{\!|N_a, K|\!\}_{K_A}$ in an initiator strand. If m is in a responder's strand, since a subterm of the form $\{\!|N_a, K|\!\}_{K_A}$ can only occur in the second or the forth nodes, we have $\{\!|N_a, K|\!\}_{K_A} \sqsubset H$ or $\{\!|N_a, K|\!\}_{K_A} \sqsubset H'$. However, neither H nor H' occurs as new in the strand. (H appears as a subterm of node $(r, 0)$, and H' appears as a subterm of node $(r, 2)$). So m can only be in a server strand $Serv[A', B', N'_a, N'_b, M', K']$ for some $A', B', N'_a, N'_b, M', K'$. By inspection on the trace form of a server strand, m can only be the second node in this strand, so either (1) $\{\!|N_a, K|\!\}_{K_A} \sqsubset \{\!|N'_a, K'|\!\}_{K'_A}$ or (2) $\{\!|N_a, K|\!\}_{K_A} \sqsubset \{\!|N'_b, K'|\!\}_{K'_B}$.

If (1) holds, then $N'_a = N_a$, $K' = K$, $A' = A$. We have $s \in Serv[A, B', N_a, N'_b, M', K]$. By Lemma 6, there exists either an initiator strand $i' \in Init[A, B', N_a, M', K]$ and $(i', 0) \in \mathcal{B}$, or $r \in Resp[A, B', N_a, M', **]$ with $A = B'$, and $(r, 1) \in \mathcal{B}$. We first prove the second case cannot hold. Suppose that there exists $r \in Resp[A, B', N_a, M', **]$, then by the trace forms of a responder strand and an initiator strand, both $(i, 0)$ and $(r, 2)$ will be nodes originating N_a, and this leads to a contradiction. So it can only be the case when there exists an initiator strand $i' \in Init[A, B', N_a, M', K]$. Then by the facts $i \in Init[A, B, N_a, M, K]$ and $i' \in Init[A, B', N_a, M', K]$, and by Lemma 5, we have $B' = B$, $M' = M$. Hence, $s \in Serv[A, B, N_a, *, M, K]$ and $(s, 1) \in \mathcal{B}$.

If (2) holds, then $N'_b = N_a$, $K' = K$, $B' = A$. We have $s \in Serv[A', A, N'_a, N_a, M', K]$. By Lemma 8, there exists either a responder strand $r \in Resp[A', A, N_a, M', K, **]$ and $(r, 1) \in \mathcal{B}$, or $i' \in Init[A', A, N_a, M', K]$ with $A' = A$, and $(i', 0) \in \mathcal{B}$. If the first case holds, then by the definition of a responder's trace and an initiator's trace, both $(i, 0)$ and $(r, 1)$ can be the node originating N_a. This leads to a contradiction. If the second case holds, then by the facts $i \in Init[A, B, N_a, M, K]$ and $i' \in Init[A, A, N_a, M', K]$, then by Lemma 5, we have $B = A$. This contradicts with the assumption $A \neq B$. ■

Similarly, we can prove a responder's authentication guarantee.

Lemma 11 (Responder's guarantee) *Suppose $B \notin \mathbb{P}$, $A \neq B$, $r \in Resp[A, B, N_b, M, K, **]$, $(r, 2) \in \mathcal{B}$, and N_b originates uniquely. Then there exists $s \in Serv[A, B, *, N_b, M, K]$, and $(s, 1) \in \mathcal{B}$.*

To sump up, we mainly use unsolicited tests and the unicity property of nonces to derive the above proofs of authentication guarantees. Here, we emphasize that we strengthen Lemma 2 by asserting the existence of a regular node m which originates $\{\!|h|\!\}_K$. So for any n such that $n \preceq_{\mathcal{B}} m$, $\{\!|h|\!\}_K$ is not a subterm of m. We frequently use this in the above proofs to ensure that a node can only be in an intended regular node. For example, we use this result to prove that the node which originates $\{\!|N_a, M, A, B|\!\}_{K_A}$ can only be the second node if it is in a responder strand (Lemma 6). We have checked the above proofs in Isabelle/HOL, the proof scripts can be obtained at [9]. Be-

sides, Lemmas 7, 9, 10, and 11 prove that Otway-Rees protocol actually achieves the authentication goals when we require that an initiator A and a responder B cannot be the same agent in one session. We observe that the protocol does not establish that the same key is delivered to both A and B, only that if either A or B reaches the end of its strand, then the other has submitted the expected matching original request $\{\!|N_b, M, A, B|\!\}_{K_B}$ or $\{\!|N_a, M, A, B|\!\}_{K_A}$. These are security properties as explored in [7, 1].

5 Concluding Remarks

In this paper, we have developed an extension of unsolicited authentication tests [1]. With our experience, our formulation of unsolicited authentication tests can be applied more generally than their original form in [1], if unsolicited tests are combined with unicity property of a nonce-based protocol. Especially, our formulation is useful in proving regularity for an encrypted term $\{\!|h|\!\}_K$, where K is a long-term regular key. In more details, if $\{\!|h|\!\}_K$ occurs as a subterm of a node, then it can be ensured that a regular node m which originates $\{\!|h|\!\}_K$ as a subterm must exist. In order to demonstrate their feasibility, we have used our results to give new proofs for the authentication goals of the Otway-Rees protocol. Compared with the proofs in [1], we did not use any side assumptions and the proofs are much simpler. We have also applied our extension to prove the authentication guarantee of a responder in a variant of Woo-Lam protocol in [9]. As future work, we would like to apply our results to more complicated protocols.

References

[1] J. D. Guttman and F. J. Thayer Fábrega. Authentication tests and the structure of bundles. *Theoretical Computer Science*, 283(2): 333-380, 2001.

[2] Y. Li. The inductive approach to strand space. In *Proceedings of 25th Conference on Formal Techniques for Networked and Distributed Systems*, LNCS 3731, pp. 547-552. Springer-Verlag, 2005.

[3] T. Nipkow, L. C. Paulson, and M. Wenzel. *Isabelle/HOL: A proof assistant for higher-order logic*. LNCS 2283. Springer-Verlag, 2002.

[4] D. Otway and O. Rees. Efficient and timely mutual authentication. *Operating Systems Reviews*, 27(2):10–14, 1987.

[5] L. C. Paulson. The inductive approach to verifying cryptographic protocols. *Journal of Computer Security*, 6:85–128, 1998.

[6] A. Perrig and D. X. Song. Looking for diamonds in the desert: Extending automatic protocol generation to three-party authentication and key agreement protocols. In *Proceedings of 13th IEEE Computer Security Foundations Workshop*, pp. 64-76. IEEE Computer Society Press, 2000.

[7] F. J. Thayer Fábrega, J. C. Herzog, and J. D. Guttman. Strand spaces: Proving security protocols correct. *Journal of Computer Security*, 7(2/3): 191-230, 1999.

[8] D. X. Song. Athena: A new efficient automated checker for security protocol analysis. In *Proceedings of 12th IEEE Computer Security Foundations Workshop*, pp. 192-202. IEEE Computer Society Press, 1999.

[9] Strand Space and Security Protocols. `http://lcs.ios.ac.cn/~lyj238/strand.html`.