言語学の暗号適用とその応用：HBプロトコルの改良

アンドレアス・グートマン†　　松浦 幹太‡

†カールスルーエ工科大学
andreas.gutmann@student.kit.edu

‡〒153-8505 東京都目黒区駒場4-6-1 東京大学生産技術研究所 第3部 松浦研究室
kanta@iis.u-tokyo.ac.jp

あらまし　1991年に松本と今井によって提案された人間識別プロトコル (Human Identification Protocol, HIP) は [1]、20年経った現在においても挑戦的な課題である。HIPの最終的な目標は信頼されたハードウェアやソフトウェアを用いずに証明者がその人間かどうかを識別することであり、HopperとBlumによって2001年にLPN仮定に基づく最初のHIP(HBプロトコル)が提案された [2]。本研究ではHBプロトコルの改良を試みる。具体的にはまず、より強力な攻撃モデルの導入及び安全性モデルの拡張を行う。その後、その安全性モデルを満たす改良方式を示す。本研究の核となるアイディアは、人間の持つ識別能力と言語学的な知識の利用である。

The use of linguistics in cryptography and its application to improve the HB protocol

Andreas Gutmann†　　Kanta Matsuura‡

†Karlsruhe Institute of Technology
Kaiserstraße 12, 76131 Karlsruhe, Germany
andreas.gutmann@student.kit.edu

‡Matsuura Laboratory
Department of Informatics and Electronics / Institute of Industrial Science, The University of Tokyo
4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, Japan
kanta@iis.u-tokyo.ac.jp

Abstract More than 20 years after the introduction by Matsumoto and Imai [1], human identification protocols (HIP) are still a challenging task for the cryptographic community. One much-noticed HIP was designed by Hopper and Blum (HB) in 2001 [2].

In this paper, we provide a novel improvement to the HB protocol. As our main result, we first extend the HIP security model by assuming an attacker who predicts random decisions made by the human. Then we suggest an improvement–based on human cognitive abilities–to the HB protocol and prove it secure under the LPN assumption in this new threat model.

1 Introduction

Today's most common method to identify a user on a computational device (e.g. PC) is the classic password method. The user knows a secret and the computational device can verify this secret to be correct. But passwords are not the only way to authenticate a user. While every approach shares one similarity, namely the use of something secret, the different methods of authentication can nevertheless be divided into three areas, while some are in between: What I know (e.g. PIN), what I have (e.g. token) and what I am (e.g. biometrics).

The problem of human identification arises when we consider a situation in which an unaided human wants to give a proof of his identity to a computational device, but the channel of communication is insecure. Even more complicated, the human wants to reuse his secret to identify himself to the same device multiple times. For any secure human identification protocol (HIP), an attacker shall not be able to identify himself as the targeted human even after observing a certain amount of successful authentications done by that human.

An example is the case of credit card skimming. The attacker could attach a device over the card slot of an ATM to read the magnetic strip of an unaware human's credit card while a camera records the PIN. Obviously, the method of a password (e.g. the PIN) in combination with a token (e.g. the credit card) fails in this scenario. Also biometrics does not guarantee security, because the devices for biometric measurement can be tampered the same way.

This threat model clearly defines a real-world scenario and any protocol not secure in this model will be prone to real-world attacks. Further research on HIP has the potential of solving many problems in today's user authentication schemes and hence can significantly improve the state-of-the-art system security.

Related works The research on HIP was pioneered by Matsumoto and Imai [1]. Hopper and Blum [2] presented the HB protocol based on the problem of learning parity with noise (see [3], for example). Further improvements of the HB protocol (e.g. the HB+ protocol [4]) concentrated–unlike our improvement–on the use within smart cards.

Problems from the research area of artificial intelligence (AI) have been suggested by [5]. The catchphrase is CAPTCHA (completely automated public Turing test to tell computers and humans apart) and the idea is to apply functions that are computable by humans, but infeasible for computational devices. Our approach, in contrast, is based on linguistics instead of AI[1], does not require to be infeasible for computational devices and is thus not a CAPTCHA.

Our contribution Our first contribution is the extension of the HIP threat model in [1] by assuming a strong attacker who predicts random decisions made by the human.

We then introduce a novel method of using linguistics in cryptography. Our second contribution is a function–based on linguistics–with the property, that humans can not only easily compute it, but also don't need to learn how to evaluate it.

Finally we use this new function to suggest an improvement to the HB protocol by "derandomizing" the computations done by the human user. We proof it to be secure in the new threat model by giving a reduction to the original HB protocol.

Outline In section 2 we provide notations, definitions and assumptions–including our extended threat model and the novel function based on linguistics. In section 3 we first define our improvement to the HB protocol and then proof it to be secure in the new threat model. In section 4 we conclude our work.

2 Preliminaries

We first introduce some general notations. Then we give a definition of the term human

[1] One could argument that language understanding is also an AI research area.

identification protocol. This is followed by the mathematical assumptions and security definitions used in this paper. Finally we state the linguistic definitions and assumptions, including our novel function.

2.1 Notations

Throughout this paper, we denote the prover by $\mathcal{H}$. Sometimes we will call the prover $\mathcal{H}$ a human user, a human or a user. Likewise we denote the verifier by $\mathcal{C}$ and sometimes call it a computer. The adversary will be denoted by $\mathcal{A}$, sometimes called an attacker, and will always be considered to be a probabilistic polynomial time Turing machine. In accordance with [1, 2] we call the attack of the adversary an impersonification attack.

For the sake of readability, we omit to explicitly state for every algorithm the given calculation time depending on the security parameter.

A protocol is a system of rules on how public and probabilistic interactive Turing machines (ITMs) interact with each other. For two public and probabilistic ITMs, denoted as $(\mathcal{H}, \mathcal{C})$, the result of the interaction, with inputs x and y respectively, is denoted by $\langle\mathcal{H}(\mathrm{x}), \mathcal{C}(\mathrm{y})\rangle$. The transcript of this interaction is denoted by $\mathrm{Trans}(\mathcal{H}(\mathrm{x}), \mathcal{C}(\mathrm{y}))$.

Any vector x is a column vector unless otherwise specified. |x| denotes the length of the vector x. For any vectors x, y with $|\mathrm{x}| = |\mathrm{y}|$, $\mathrm{x} \cdot \mathrm{y}^T$ is the dot product of the vector x and the transpose of the vector y. If it is obvious, we omit the superscript T.

A function $\mu(\cdot)$: $\mathbb{N} \to \mathbb{R}$ is negligible, if for every positive polynomial $\mathrm{p}(\cdot)$ there exists an integer $\mathrm{n}_0 \in \mathbb{N}$ such that for all $\mathrm{n} > \mathrm{n}_0$ $|\mu(n)| < {}^1\!/_{p(n)}$.

Throughout this paper, we use the terms identification and authentication interchangeable.

2.2 Mathematical assumptions

Here we give a definition of the well known and NP-hard LPN problem.

Conjecture 1. *Learning Parity in the Presence of Noise (LPN) (search version).* Let A be a random m × k binary matrix, let x be a random k-bit vector, let $\eta \in \,]0, {}^1\!/_2[$ be a constant noise parameter, and let ν be a random m-bit vector such that $|\nu| \leq \eta\mathrm{m}$.

The challenge in the search version of LPN is, given A, η, and $\mathrm{r} = (\mathrm{A} \cdot \mathrm{x}) \oplus \nu$, to find a k-bit vector x′ such that $|(\mathrm{A} \cdot \mathrm{x}'\ ) \oplus \mathrm{r}| \leq \eta\mathrm{m}$.

2.3 Security definitions

First we define a human identification protocol as the interaction between two ITMs, with one ITM as the proofer and the other as the verifier, under the condition that the computations of the proofer are done in the mind of a human. Then we give a detailed explanation of our extension to the threat model from [1]. This extension is our first contribution.

Definition 1. *Human identification protocol.* Let $(\mathcal{H}, \mathcal{C})$ be a pair of public and probabilistic ITMs. Their interaction is an identification protocol if for any shared input x, $\Pr[\langle\mathcal{H}(\mathrm{x}), \mathcal{C}(\mathrm{x})\rangle = \mathrm{accept}] \geq \Delta$ and for each pair of inputs $\mathrm{x} \neq \mathrm{y}$ $\Pr[\langle\mathcal{H}(\mathrm{x}), \mathcal{C}(\mathrm{y})\rangle = \mathrm{accept}] \leq 1 - \Delta$, where $0.5 < \Delta \leq 1$.

It is a human identification protocol, if the computations of the ITM $\mathcal{H}$ are done in the mind of a human.

Extended threat model. Our threat model is an extension of the threat model of Matsumoto and Imai. In this model the unaided human $\mathcal{H}$ wants be authenticated by the computer $\mathcal{C}$. Therefore, $\mathcal{H}$ and $\mathcal{C}$ might exchange messages until $\mathcal{C}$ decides to either accept or reject $\mathcal{H}$. The messages exchanged between $\mathcal{H}$ and $\mathcal{C}$ are relayed by a terminal.

The passive adversary $\mathcal{A}$ has the ability to witness the human $\mathcal{H}$, his input to the terminal, the computations done inside the terminal and the messages delivered between the terminal and $\mathcal{C}$.

In our extension of the threat model, we allow the adversary to predict the random coins of $\mathcal{H}$. We believe that this stronger adversary gives a better representation of the real world. The reason for this assumption is that we think humans are no good random generators and, in addition, we believe that an attacker can make conclusions by observing the human.

As a real-world example, recall the previous mentioned credit card skimming attack. Here an attacker might derive information based on the camera observation of the human $\mathcal{H}$ or his response time. This possibility is not covered by the previous threat model.

Another reason, why we think that our extended threat model is important, is a recent result on machine learning [6]. It was shown, that there exists an algorithm such that, given any instance of the LPN problem with structured noise, the secret vector can be found in polynomial time. Structured noise means that there won't be an arbitrary burst of noise. In other words, it is guaranteed that at most p out of m bits are noisy.

Definition 2. *Passive adversary [2].* An identification protocol $(\mathcal{H}, \mathcal{C})$ is (p, k)-secure against passive adversaries if for all computationally bounded adversaries $\mathcal{A}$,

$$\Pr\big[\mathcal{A}\big(\text{Trans}^k(\mathcal{H}(\text{x}), \mathcal{C}(\text{x})), \mathcal{C}(\text{x})\big) = \text{accept}\big] \leq \text{p} \ ,$$

where $\text{Trans}^k(\mathcal{H}(\text{x}),\ \mathcal{C}(\text{x}))$ is a random variable sampled from k independent transcripts $\text{Trans}(\mathcal{H}(\text{x}), \mathcal{C}(\text{x}))$.

2.4 Linguistics

We start with an explanation of the term semantics. This is followed by the definitions of a semantic relation, a semantic network and our novel function about related words. We conclude this section by giving a more formal explanation of our assumption on the human-computability of that function.

Semantics. Semantics is a sub discipline of linguistics–the scientific study of language. It focuses on the intuition of native speakers about the meaning of words and expressions. It is–among other things–concerned with the meaning of words, their relations and how they combine to form sentence meanings.

Definition 3. *Semantic relation.* The term semantic relationship refers to relations between different words and their various meanings. A semantic relation is a relation that maps two or more concepts to the truth set.

Examples of semantic relations are synonymy (A denotes the same as B), hyponymy (A is more specific than B) and meronymy (A is part of B).

Definition 4. *Semantic network.* Let $\mathcal{S} := \{\mathcal{S}_k\}_{k \in \mathbb{N}}$ be the space drawn from a dictionary such that $\mathcal{S}_k \subseteq \{0,1\}^k$. Then a semantic network $\mathcal{N}(\mathcal{S})$ is a weighted partially directed graph. It represents (a subset of) semantic relations as edges–weighted between zero and one by the strength of the semantic relation–between the elements of $\mathcal{S}$ (the vertices).

The purpose of definition 4 is to give a visualisation that might help the reader to understand the following definition.

Definition 5. *Related words.* $\mathcal{R}_{g,\delta}(\xi,\, s)$ is a binary relation with g $\in \mathbb{N}$, $\delta \in (0,\ 1]$, s $\in \mathcal{S}$ and ξ is a formula with literals from $\mathcal{S}$ and operators from the set of logical operators $\{\wedge, \vee, \neg\}$. Then ξ defines a subset of the semantic network $\mathcal{N}(\mathcal{S})$.

We further define $\mathcal{R}_{g,\delta}(\xi,\, s) = 1$ if and only if there exists a path between the subset ξ and term s in $\mathcal{N}(\mathcal{S})$, whereby the length of the path is at most g and the weight of each edge is greater than some threshold δ.

Definition 5 states our second contribution. While we defined it as a relation, it is obvious that this relation is well-behaved and thus a function. To be more precisely, it is a family of functions, because for different indexes g and δ it results in a different function. In section 3 we will show how this family of functions can be of use in cryptography.

Conjecture 2. *Human-computability of* $\mathcal{R}_{g,\delta}(\xi, s)$. We assume that there exist g, δ, $\mathcal{S}$ such that for every s $\in$ $\mathcal{S}$ and ξ as defined above most humans can easily compute $\mathcal{R}_{g,\delta}(\xi, s)$ in their mind.

This assumption is based on the common believe in semantics, that native speakers have a good intuition about the meaning of and relation between words and expressions.

3 New results

We begin this section by restating the HB protocol from [2] and recapitulating its security properties. Then we give a brief analysation of the HB protocol in the extended threat model. Finally we describe our improvement to the HB protocol and conclude this section with the corresponding security proof.

3.1 Original HB protocol

Informally speaking, the computer $\mathcal{C}$ generates the matrix $A^{m\times k}$ and sends this matrix to the human user. The human $\mathcal{H}$ computes r = (A $\cdot$ x) with some errors and replies with this vector to $\mathcal{C}$.

If sufficient many bits in the response are correct, $\mathcal{C}$ has evidence that $\mathcal{H}$ is the claimed user and will–possibly after some repetitions to strengthen the evidence–accept $\mathcal{H}$.

Shared secrets: $\mathcal{H}$ and $\mathcal{C}$ share a secret vector x $\in \{0, 1\}^k$.
Public parameters: $\eta \in (0, 1/2)$

(C1) $\mathcal{C}$ sets i := 0
(C2) $\mathcal{C}$ selects m random challenges $c_1,\ldots,c_m$ from $\{0, 1\}^n$ and sends them to $\mathcal{H}$
(H1) For every challenge c_j: With probability $1 - \eta$, $\mathcal{H}$ responds with $r_j := c_j \cdot x$, otherwise $\mathcal{H}$ calculates his response as $r_j := 1 - c_j \cdot x$
(C3) For every challenge c_j: If $r_j = c_j \cdot x$, then $\mathcal{C}$ increments i
(C4) If i $\geq (1 - \eta) \cdot$ m, $\mathcal{C}$ accepts $\mathcal{H}$

3.1.1 Security of the HB protocol

Next we compare the security of the HB protocol in both threat models.

Security in the original threat model We will cite two theorems from [2]. The first theorem states that if $\mathcal{H}$ responses with random answers, he will be accepted with only negligible probability. The second theorem states that the security of this protocol is based on the LPN assumption. For proofs of these theorems, we refer to section 4 in [2].

Theorem 1. [2] If $\mathcal{H}$ guesses random responses r, $\mathcal{C}$ will accept $\mathcal{H}$ with probability at most

$$\left(\frac{1}{2}\right)^m \sum_{i=(1-\eta)m}^{m} \binom{m}{i} \leq e^{-c_o m},$$

where $c_0 \geq \frac{2}{3}$ is a constant depending only on η.

Theorem 2. [2] If LPN is hard, then [... the HB protocol is] secure against a passive adversary.

Security in the extended threat model The assumed hardness of the LPN problem is based on the distribution of the noise. In the

HB protocol, this distribution is based on the humans random decisions.

But in the extended threat model, the adversary can predict the humans random decisions. Therefore the distribution and frequency of the noise are known to the attacker. It follows that the LPN instance degenerates and becomes solvable by the adversary within linear time by Gaussian elimination.

Even by assuming a slightly weaker attacker, that can only partially predict the humans decisions, it was shown in [6] that the attacker might be able to find the secret vector in polynomial time by exploiting some known structure of the noise.

3.2 Improved HB protocol

The goal of our proposed improvement is to completely relieve the human from the task of making random decisions while keeping a good distribution of his noisy answers. We achieve this by using our previously defined function about related words to communicate a secret bit between the computer and the human.

The following is the improved HB protocol:
Shared secrets: $\mathcal{H}$ and $\mathcal{C}$ share a secret vector x $\in \{0, 1\}^k$ and a secret $\xi \subset \mathcal{N}(\mathcal{S})$.
Public parameters: g $\in \mathbb{N}$, $\delta \in (0, 1]$, $\mathcal{N}(\mathcal{S})$

(C1) $\mathcal{C}$ selects m random words s$_1$,...,s_m from the dictionary $\mathcal{S}$

(C2) $\mathcal{C}$ selects m random challenges c$_1$,...,c$_m$ from $\{0, 1\}^n$ and sends s$_1$,...,s$_m$ and c$_1$,...,c$_m$ to $\mathcal{H}$

(H1) For every challenge-word pair (c$_j$,s$_j$): If $\mathcal{R}_{g,\delta}(\xi, s_j) = 0$, $\mathcal{H}$ responds with r$_j$:= c$_j$ · x, otherwise $\mathcal{H}$ responds with r$_j$:= 1 − c$_j$ · x

(C3) For all challenge-word-response triples (c$_1$,s$_1$,r$_1$),...,(c$_m$,s$_m$,,r$_m$): If $\mathcal{R}_{g,\delta}(\xi, s_j)$ = 0, $\mathcal{C}$ checks if r$_j$:= c$_j$ · x, else $\mathcal{C}$ checks if r$_j$:= 1 − c$_j$ · x. If any r$_j$ is wrong, $\mathcal{C}$ rejects.

3.2.1 Security proof

Our security proof is organized as a sequence of games. Game number one will be the original HB protocol. The second game serves as a bridging step to allow us a clearer indistinguishability-based transition to game 3. This transition covers the major changes in the protocol and allows the adversary to gain a negligible advantage. The final game is another bridging step and results in the improved HB protocol.

Theorem 3 is the main theorem of this section and will be proven at the end.

Theorem 3. If the HB protocol is secure under the LPN assumption in the threat model defined by [1], then the extended HB protocol is secure under the LPN assumption in the extended threat model.

Game 1

Game 1 is the original HB protocol as stated in section 3.1.

Game 2

This is the same as game 1, except that the computer performs the following additional steps: First the computer draws a number of random words from the dictionary equal to the number of challenges it generates. Then the computer sends those words to the human.

To be more precisely, we add the steps (C1.1) and (C2.1), while adjusting step (C2) as follows:

(C1.1) $\mathcal{C}$ selects m random words s$_1$,...,s_m from the dictionary $\mathcal{S}$

(C2) $\mathcal{C}$ selects m random challenges c$_1$,...,c$_m$ from $\{0, 1\}^n$

(C2.1) $\mathcal{C}$ sends s$_1$,...,s$_m$ and c$_1$,...,c$_m$ to $\mathcal{H}$

Lemma 1. For any passive adversary $\mathcal{A}$, the advantage for an impersonification attack in game 2 is equal to the advantage for an impersonification attack in game 1.

Proof. The additional steps in game 2 are not related to the security properties of game 1.

Thus the attacker does not gain any advantage. □

Game 3

In Game 3 we do the following changes to game 2: The ITMs $\mathcal{H}$ and $\mathcal{C}$ share an additional secret $\xi \subset \mathcal{N}(\mathcal{S})$. The parameters g, δ and $\mathcal{N}(\mathcal{S})$ are the new public parameters. η is substituted for the probability that for a random s $\in \mathcal{S}$, the relation $\mathcal{R}_{g,\delta}(\xi,\ s)$ evaluates to 1 and the computations of $\mathcal{H}$ take this substitution into account.

Summarized, this means the following parts of the protocol are changed or added:

Shared secrets: $\mathcal{H}$ and $\mathcal{C}$ share a secret vector x $\in \{0,\ 1\}^k$ and a secret $\xi \subset \mathcal{N}(\mathcal{S})$.

Public parameters: $g \in \mathbb{N}$, $\delta \in (0,\ 1]$, $\mathcal{N}(\mathcal{S})$

(C1.2) $\mathcal{C}$ sets η to be the probability, that for s $\xleftarrow{r} \mathcal{S}$ the relation $\mathcal{R}_{g,\delta}(\xi,\ s)$ evaluates to 1.

(H1) For every challenge-word pair (c_j,s_j): If $\mathcal{R}_{g,\delta}(\xi,\ s_j) = 0$, $\mathcal{H}$ responds with $\mathrm{r}_j := \mathrm{c}_j \cdot \mathrm{x}$, otherwise $\mathcal{H}$ responds with $\mathrm{r}_j := 1 - \mathrm{c}_j \cdot \mathrm{x}$

Remark 1. We will ignore the different public parameters in the security proof by assuming dummy (fake) public parameters.

Conjecture 3. We assume, that the size of the dictionary $\mathcal{S}$ and the secret ξ are such, that in step (C1.2) the value of η is set to be strictly between 0 and $^1/_2$.

This is a very likely assumption if $\mathcal{S}$ is a dictionary of a natural language and ξ is restricted to be easily rememberable by most humans.

Then in step (H1) of game 3, the probability $\Pr\left[\mathcal{R}_{g,\delta}(\xi,\ s_j) = 0\right]$ is $1 - \eta$ and thus equal to the probability distribution in step (H1) of game 1 and game 2.

Lemma 2. For the advantage of the passive adversary $\mathcal{A}$ in performing an impersonification attack, the following equation holds:

$$Adv^{imp}_{game\,3,\mathcal{A}} \leq Adv^{imp}_{game\,2,\mathcal{A}} + negligible\ .$$

Proof. Let $\mathcal{D}$ be a PPT algorithm, distinguishing between game 2 and game 3. The only noticeable difference between game 2 and game 3 could be in the human's answers. To notice this difference, $\mathcal{D}$ needs to know the secret ξ.

To compute ξ in any given instance of this protocol is equal to solving the corresponding LPN problem. Therefore, the distinguisher needs to guess ξ.

Because $\mathcal{S}$ is assumed to be the dictionary of a natural language, the distinguishers advantage in guessing ξ can be considered negligible. It follows, that the advantage of $\mathcal{D}$ in distinguishing game 2 from game 3 is negligible.

Then the advantage of a passive attacker $\mathcal{A}$ in game 3 is:

$$Adv^{imp}_{game\,3,\mathcal{A}} \leq Adv^{imp}_{game\,2,\mathcal{A}} + negligible\ .$$

□

Game 4

In game 4 we change the calculations done to determine if $\mathcal{H}$ is accepted by $\mathcal{C}$ or not. The steps (C1), (C1.2), (C3) and (C4) in game 3 are substituted by the new (C3) in game 4:

(C3) For all challenge-word-response triples (c_1,s_1,r_1),...,(c_m,s_m,,r_m): If $\mathcal{R}_{g,\delta}(\xi,\ s_j) = 0$, $\mathcal{C}$ checks if $\mathrm{r}_j := \mathrm{c}_j \cdot \mathrm{x}$, else $\mathcal{C}$ checks if $\mathrm{r}_j := 1 - \mathrm{c}_j \cdot \mathrm{x}$. If any r_j is wrong, $\mathcal{C}$ rejects.

For readability, the step (C1.1) is renamed to (C1) and the steps (C2) and (C2.1) are combined to the new step (C2).

Step (C3) in game 4 takes advantage of the fact, that the human behaves deterministically. While in the original HB protocol the computer has to check the distribution of all answers, now $\mathcal{C}$ can calculate for every answer separately if it should be $\mathrm{c}_j \cdot \mathrm{x}$ or $1 - \mathrm{c}_j \cdot \mathrm{x}$.

Lemma 3. In game 4, the advantage of the passive adversary $\mathcal{A}$ in performing an impersonification attack is:

$$Adv^{imp}_{game\,4,\mathcal{A}} = Adv^{imp}_{game\,3,\mathcal{A}}\ .$$

Proof. Sine the changes done in game 4 neither affect the correctness of the protocol nor any exchanged messages between $\mathcal{C}$ and $\mathcal{H}$, the advantage of adversary $\mathcal{A}$ does not change. □

Proof of theorem 3. The correctness of theorem 3 directly follows from the lemmata 1, 2 and 3. □

Informally speaking, the adversary doesn't gain a significant advantage compared to the original HB protocol, because the noise has a similar distribution in both protocols and the additionally send words $\mathrm{s}_j \in \mathcal{S}$ don't yield to any advantage for the adversary.

4 Conclusion

In this paper we have extended the threat model for human identification protocols by assuming a stronger attacker who can predict random decisions made by the human. We have stated why this model is reasonable in the real-world setting. The extension becomes particularly important considering recent advances on machine learning algorithms [6].

We have introduced a novel function based on human linguistic abilities. This function about the relation of words in a natural language has the benefit, that the evaluation of it is a natural task for most humans and doesn't have to be learned.

In the context of the extended threat model, the HB protocol has been improved by applying the function about related words. We have given a proof for the security of the resulting protocol in the new threat model.

4.1 Open questions

Our function about the relation of words is based on assumptions. It remains an open question to find more evidence that this function fulfils the believed properties.

The function, as defined and used in this paper, leads to the exchange of one bit of information. It is an open question if it is possible to increase the amount of information exchanged without noticeably reducing the usability of the function.

Another questions is whether our improved HB protocol leads to either innovative attacks, or gives the possibility of better security guarantees.

参考文献

[1] Matsumoto, T., & Imai, H. (1991). Human identification through insecure channel. In *Advances in Cryptology–EUROCRYPT 1991* (pp. 409-421). Springer Berlin Heidelberg.

[2] Hopper, N. J., & Blum, M. (2001). Secure human identification protocols. In *Advances in Cryptology–ASIACRYPT 2001* (pp. 52-66). Springer Berlin Heidelberg.

[3] Blum, A., Kalai, A., & Wasserman, H. (2003). Noise-tolerant learning, the parity problem, and the statistical query model. In *Journal of the ACM (JACM), 50(4)* (pp. 506-519).

[4] Juels, A., & Weis, S. A. (2005). Authenticating pervasive devices with human protocols. In *Advances in Cryptology–CRYPTO 2005* (pp. 293-308). Springer Berlin Heidelberg.

[5] Von Ahn, L., Blum, M., Hopper, N. J., & Langford, J. (2003). CAPTCHA: Using hard AI problems for security. In *Advances in Cryptology–EUROCRYPT 2003* (pp. 294-311). Springer Berlin Heidelberg.

[6] Arora, S., & Ge, R. (2011). New algorithms for learning in presence of errors. In *Automata, Languages and Programming* (pp. 403-415). Springer Berlin Heidelberg.