More on the Security of Linear RFID Authentication Protocols

Matthias Krause and Dirk Stegemann

Theoretical Computer Science
University of Mannheim
Mannheim, Germany

Abstract. The limited computational resources available in RFID tags implied an intensive search for lightweight authentication protocols in the last years. The most promising suggestions were those of the HB-familiy (HB$^+$, HB$^\#$, TrustedHB, ...) initially introduced by Juels and Weis, which are provably secure (via reduction to the Learning Parity with Noise (LPN) problem) against passive and some kinds of active attacks. Their main drawbacks are large amounts of communicated bits and the fact that all known HB-type protocols have been proven to be insecure with respect to certain types of active attacks. As a possible alternative, authentication protocols based on choosing random elements from L secret linear n-dimensional subspaces of $\mathrm{GF}(2)^{n+k}$ (so called CKK-protocols) were introduced by Cichoń, Klonowski, and Kutyłowski. These protocols are special cases of (linear) (n, k, L)-protocols which we investigate in this paper. We present several active and passive attacks against (n, k, L)-protocols and propose $(n, k, L)^{++}$-protocols which we can prove to be secure against certain types of active attacks. We obtain some evidence that the security of (n, k, L)-protocols can be reduced to the hardness of the *learning unions of linear subspaces* (LULS) problem. We then present a learning algorithm for LULS based on solving overdefined systems of degree L in Ln variables. Under the hardness assumption that LULS-problems cannot be solved significantly faster, linear (n, k, L)-protocols (with properly chosen n, k, L) could be interesting for practical applications.

Keywords: Lightweight Cryptography, RFID Authentication, Algebraic Attacks, HB$^+$, CKK, CKK^2.

1 Introduction

In lightweight cryptography one tries to solve the problem of determining the minimal amount of computational resources which have to be invested for reaching certain security goals. This problem implies a lot of interesting and nontrivial theoretical questions. Since weak computational devices (e.g., mobile devices, RFIDs) are used in practice to a rapidly growing extent, results in lightweight cryptography are highly desired also from a practical point of view.

M.J. Jacobson Jr., V. Rijmen, and R. Safavi-Naini (Eds.): SAC 2009, LNCS 5867, pp. 182–196, 2009.

RFID (radio frequency identification) tags are small devices that are equipped with only little memory and computational power. Their main application is the identification of objects. In order to prevent cloning and tracing attacks and to preserve the tagged object's privacy, RFID tags should reveal their identities only to legitimate readers. Since most practically relevant RFID tags are too weak to execute standard authentication protocols, alternative measures are necessary. Besides technical approaches based on blocking or disturbing the communication, lightweight authentication protocols and corresponding security models are intensively discussed (see, e.g., [13,15]).

One of the most promising proposals was the HB^+ protocol due to Juels and Weis [14], which is provable secure (via reduction to the learning parity with noise (LPN) problem) with respect to passive and some kinds of active attacks. A severe drawback of the protocol is that presumably secure parameter combinations imply large amounts of transmitted data. Together with the small available bandwidth in RFID communication, this may add up to authentication times that are unacceptable for many applications. Another disadvantage is that HB^+ and all its variants suggested so far have been broken by man-in-the-middle (MITM) attacks. Particularly, the HB^+-protocol was broken by Gilbert, Robshaw and Sibert in [12], the $\mathsf{HB}^\#$-protocol introduced by Gilbert, Robshaw and Seurin in [11] was recently broken by Ouafi, Overbeck and Vaudenay in [16], and Trusted-HB introduced by Bringer and Chabanne in [3] was broken by Frumkin and Shamir in [10].

As a possible alternative to HB-type protocols, another class of lightweight authentication protocols (so called CKK-protocols) were introduced by Cichoń, Klonowski, and Kutyłowski [4]. These protocols can be generalized to linear (n, k, L)-protocols, in which the secret key (the identification information in the RFID tag) consists of the specification of L n-dimensional linear subspaces $V_1, \ldots, V_L$ of $\mathrm{GF}(2)^{n+k}$, while the identification is performed by collaboratively generating an element $v \in V_l$ for a random $l \in \{1, \ldots, L\}$. In [4], the CKK^2-protocol, a special linear $(n, k, 2)$-protocol, and the $\mathsf{CKK}^{\sigma,L}$-protocol, a special linear (n, k, L)-protocol, were suggested for practical application.

Compared with HB-type protocols, the advantages of (n, k, L)-protocols are that fewer bits have to be communicated, computational effort and memory requirements are lower on the prover's side (essentially, the prover has to generate random elements from L different n-dimensional subspaces of $\mathrm{GF}(2)^{n+k}$), and that (n, k, L)-protocols seem to be more resistant against active attacks. The drawback is that we can not yet prove the security of (n, k, L)-protocols by reduction to a well-established problem like the LPN-problem. However, in this paper we show that, similarly to HB-type protocols, the security of (n, k, L)-protocols can be related to the hardness of a certain learning problem, in this case the *Learning Unions of L linear subspaces* (LULS) problem.

Our strategy for designing a lightweight authentication protocol is the same as in the context of HB-type protocols and consists of the following two steps.

1. Define an appropriate lightweight symmetric encryption function $E : X \times K \longrightarrow Y$, the basis operation, which guarantees that that the basic

E-protocol is secure against a passive adversary. Hereby, X denotes an appropriate input-, K an appropriate key-, and Y an appropriate output space.
2. Define a protocol structure P over E such that the security of P with respect to active adversaries can be reduced to the security of the basic E-protocol against a passive adversary.

The basic E-protocol is defined as follows: Alice and Bob share a secret key $k \in K$. In one round, the verifier Alice sends *hello* to the prover Bob. Receiving *hello*, Bob chooses a random element $x \in X$, which is distributed according to a publicly known probability distribution $\Pr_B$, and sends $E_k(x)$ back to Alice. After a predefined number of rounds, Alice decides about accepting or rejecting by applying a verification operation to the messages sent by Bob. The definition of the verification operation depends on the definition of E. A passive adversary has only passive access to the insecure channel between Alice and Bob, i.e., she has to reach her goal on the basis of a set of observations $F_k(x_1), \ldots, F_k(x_m)$, where for all $i = 1, \ldots, m$, x_i is randomly and independently choosen according to $\Pr_B$.

Note that for HB-type protocols, $K = \mathrm{GF}(2)^n$, $X = Y = \mathrm{GF}(2)^n \times \mathrm{GF}(2)$, $y = \mathrm{GF}(2)$, and the basis operation is defined by

$$E((x, \nu), k) = (x, y) \ ,$$

where $y = x \cdot k \oplus \nu$. Bob chooses x with respect to the uniform distribution and sets the noise bit ν to one with probability $p < 0.5$. Alice accepts if the number of rounds in which $y_i = x_i \cdot k$ is satisfied exceeds a certain threshold.

Obviously, basic E-protocols are vulnerable to replay attacks. In both cases, HB-type- and linear protocols, the basic E-protocol is also vulnerable to active key recovery attacks (see [14] and the attack described in subsection 2.4, respectively). Consequently, solving challenge (2.) is an important task, which could not be done in a satisfactory way so far in the case of HB-type protocols.

Our results and the outline of this paper are as follows. In Subsect. 2.1 we define the basis operation of linear protocols and specify the adversary models. In Subsect. 2.2 we take a look at CKK-protocols [4], the first type of linear protocols occuring in the literature. We present a fast passive (polynomial time) attack against the CKK^2-protocol which allows to recover the secret key for the proposed parameters $(n, k) = (128, 30)$ in less than a second on a standard PC, while an earlier (exponential time) attack on CKK^2 published in [7] requires a couple of hours on comparable hardware.

In Subsect. 2.3 we describe special active key recovery attacks against linear protocols, so called equality attacks, and show that the basic linear (n, k, L)-protocol and the linear $(n, k, L)^+$-protocol (which is based on the same design principle as HB^+) are vulnerable to these attacks.

In Subsect. 2.4 we introduce $(n, k, L)^{++}$-protocols and prove their security against equality attacks.

In Subsect. 2.5 we list some generic attacks against linear protocols. Moreover, we introduce the *Learning Unions of L linear subspaces* (LULS) problem. The complexity of the LULS-problem characterizes the security of linear protocols with respect to passive adversaries. We give a generic exponential time algorithm to solve this problem, and we show that active adversaries that are able to efficiently solve the LULS-problem can break the $(n, k, L)^+$-protocol.

In Sect. 3 we present a nontrivial learning algorithm for the LULS-problem. We outline the algorithm in all details for the case $L = 2$ and describe how the ideas can be generalized to the case $L > 2$. The algorithm is based on generating and solving $\frac{k}{s}$ special overdefined systems of degree-L equations over $\mathrm{GF}(2^s)$ for appropriate $s \leq k$. Our hardness assumption is that the running time of this learning algorithm characterizes the complexity of the LULS-problem and the complexity of actively attacking $(n, k, L)^{++}$-protocols.

In Sect. 4 we discuss some aspects of the practical use of $(n, k, L)^{++}$-protocols. General (n, k, L)-protocols have a huge keylength of $L \cdot n \cdot n + k$. One idea could be to use $\mathsf{CKK}^{\sigma,L}$-protocols (see [4]), a special $(n, 1, L)$-protocol which is still unbroken. Other ideas for reducing the keylength in similar cases were discussed in the literature, e.g., using keys defined by Toeplitz matrices instead of random matrices [11], or defined by special Toeplitz matrices generated by Linear Feedback Shift Registers (LFSRs) [3]. The security analysis of the corresponding types of special (n, k, L)-protocols remains a matter of further research.

We have experimentally confirmed the correctness and efficiency of our attacks and algorithms with the computer algebra system Magma [2].

2 Linear (n, k, L)-Protocols

2.1 The Basis Operation and the Adversary Models

In a linear (n, k, L)-protocol, Alice (the verifier, e.g., an RFID reader) and Bob (the prover, e.g., an RFID tag) share a common secret information (the tag's ID) from a certain keyspace. As usual, we assume that the secret key is hardwired in the RFID tag, while Alice has legal access to a database containing Bob's secret information.

We define now the basis operation of linear (n, k, L)-protocols and denote for a positive integer N the set $\{1, \ldots, N\}$ by $[N]$.

The secret keys of the protocols consist of the specifications of L n-dimensional injective linear functions $F_1, \ldots, F_L : \mathrm{GF}(2)^n \longrightarrow \mathrm{GF}(2)^{n+k}$. The inputs are pairs (x, l), where $x \in \mathrm{GF}(2)^n$ and $l \in [L]$.

Let us denote by $V_1, \ldots, V_L$ the n-dimensional linear subspaces of $\mathrm{GF}(2)^{n+k}$ corresponding to the images of $F_1, \ldots, F_L$, respectively.

In the basic linear protocol, Alice accepts a message $w \in \mathrm{GF}(2)^{n+k}$ coming from Bob if $w \in V_l$ for some $l \in [L]$.

We analyze the security of (n, k, L)-protocols with respect to passive and active adversaries. A passive adversary is able to read the messages exchanged by Alice and Bob. His aim is (partial) key recovery, i.e., to try to compute nontrivial information about the secret key from a set of messages produced by the honest parties Alice and Bob.

An active adversary has the additional abilities

- To corrupt or to replace messages sent from Alice to Bob,
- To corrupt or to replace messages sent from Bob to Alice,
- To retrieve the information whether a (possibly corrupted) transcript has been accepted or rejected by Alice.

We assume that neither of the adversaries is able to read nor modify the keybits nor the inner state bits nor the private random bits of Alice or Bob.

2.2 The CKK-Protocols

The protocols CKK^1, CKK^2 and $\mathsf{CKK}^{\sigma,L}$ suggested by Cichoń, Klonowski and Kutyłowski in [4] are restricted types of (n,k,L)-protocols.

The protocol $\mathsf{CKK}^{\sigma,L}$ is an (n,k,L)-protocol with the restriction $F_l(u) = \sigma^l(u||f(u))$ for all $l \in [L]$, where σ denotes a secret permutation $\sigma \in \mathcal{S}_{n+k}$ and f a secret linear function $f : \mathrm{GF}(2)^n \longrightarrow \mathrm{GF}(2)^k$. Hence, the secret keys have the form (f,σ).

The protocol CKK^2 is an $(n+k,k,2)$-protocol with the additional properties that $F_1(u,a) = (u,f(u),a)$ and $F_2(u,a) = (u,a,f(u))$ for all $u \in \mathrm{GF}(2)^n$ and $a \in \mathrm{GF}(2)^k$, where f denotes a secret linear function $f : \mathrm{GF}(2)^n \longrightarrow \mathrm{GF}(2)^k$.

CKK^2 and $\mathsf{CKK}^{\sigma,L}$ protocols were suggested for practical application in [4], with the parameters $n = 128$ and $k = 30$.

So far, the only nontrivial cryptanalytic result concerning linear (n,k,L)-protocols is due to Gołebięwski, Majcher and Zagórski [7]. They present an attack against the CKK^2-protocol, which cannot be applied to the general case. Its running time is proportional to $\sum_{s=0}^{k-1}\binom{n}{s}$, i.e., of order $n^{\Theta(k)}$.

We now describe a very fast attack against the CKK^2-protocol with parameters (n,k) whose running time is dominated by the effort required for inverting k $(n \times n)$-matrices.

Let $f : \mathrm{GF}(2)^n \longrightarrow \mathrm{GF}(2)^k$ denote the secret key, recall that

$$V_1 = \{(v, f(v), a),\ v \in \mathrm{GF}(2)^n, a \in \mathrm{GF}(2)^k\}\ ,$$
$$V_2 = \{(v, a, f(v)),\ v \in \mathrm{GF}(2)^n, a \in \mathrm{GF}(2)^k\}\ .$$

Let the functions $f^1, \ldots, f^k : \mathrm{GF}(2)^n \longrightarrow \mathrm{GF}(2)$ denote the component functions of the secret function f, i.e., $f(v) = (f^1(v), \ldots, f^k(v))$ for all $v \in \mathrm{GF}(2)^n$. The attack is based on the simple fact that if an observation (v,a,b) satisfies $a_r = b_r$ for some $r \in [k]$, which is true with probability 1/2, then we know that $f^r(v) = a_r = b_r$.

The attack works as follows.

```
1. Let $e_1, \ldots, e_n$ denote the standard basis of $\mathrm{GF}(2)^n$.
2. FOR $r \in [k]$
   2.1 Consider a set of messages produced by Bob and extract from it a set
       $O_r = ((v_{r,1}, a_{r,1}, b_{r,1}), \ldots, (v_{r,n}, a_{r,n}, b_{r,n}))$ such that $v_{r,1}, \ldots, v_{r,n}$ form a
       basis of $\mathrm{GF}(2)^n$ and $a_{r,i}(r) = b_{r,i}(r) = f^r(v_{r,i})$ for all $i \in [n]$.
   2.2 Derive $f^r(e_1), \ldots, f^r(e_n)$ from $O_r$.
```

Table 1. Performance of the passive attack on CKK^2

(n, k)	approx. number of observations	approx. attack time
$(128, 30)$	311	0.3 s
$(1024, 256)$	2197	179 s

The correctness of the attack follows straightforwardly from the definitions. The expected number of messages needed for constructing O_r can be estimated based on the following experiment.

```
1. Set $B := \emptyset$.
2. REPEAT
   2.1 Choose a random $v \in \mathrm{GF}(2)^n$ (w.r.t. the uniform distribution).
   2.2 $V := V \cup \{v\}$.
3. UNTIL $V$ is a generating system of $\mathrm{GF}(2)^n$.
```

Let $p(n)$ denote the probability that the experiment stops after n iterations (i.e., V is a basis of $\mathrm{GF}(2)^n$), and $E(n)$ denote the expected number of iterations of the experiment. It is known that $p(n) \approx 0.2887$ and $E(n) \approx n + 1.6067$ (see, e.g., [7]). Hence, an estimate for the expected number of messages for constructing O_r is $2 \cdot E(n) \approx 2n + 3.2134$. For the parameter choices proposed for practical applications, the attack is very efficient already on standard PC hardware (Magma V2.15-9 [2] on a 3.4 GHz Intel Pentium IV with 4 GB RAM), see Table 1.

2.3 Basic Protocol Types and Equality Attacks

In the basic linear protocol, Alice starts the communication by sending some signal triggering Bob to compute a proof w of his identity. In particular, Bob computes $w = F_l(u)$ for randomly (independently and uniformly) chosen $l \in [L]$ and $u \in \mathrm{GF}(2)^n$. Alice accepts a proof $\tilde{w}$ if there is an $l \in [L]$ such that $\tilde{w} \in V_l$ (see Fig. 1).

Obviously, this protocol is vulnerable to replay attacks, since an adversary can store a number of proofs and then impersonate Bob by presenting these proofs to Alice.

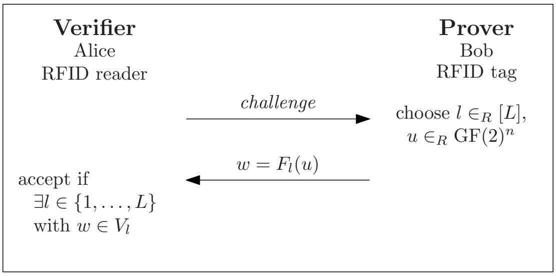

Fig. 1. Basic Communication Mode

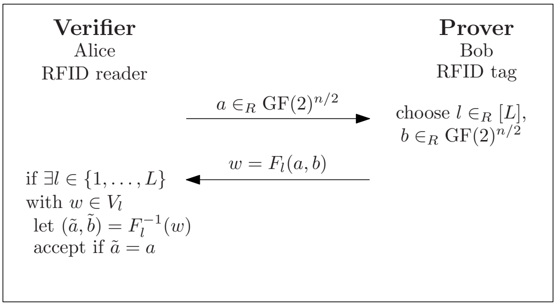
Verifier
Alice
RFID reader
Prover
Bob
RFID tag
$a \in_R \mathrm{GF}(2)^{n/2}$
choose $l \in_R [L]$,
$b \in_R \mathrm{GF}(2)^{n/2}$
$w = F_l(a, b)$
if $\exists l \in \{1, \ldots, L\}$
with $w \in V_l$
let $(\tilde{a}, \tilde{b}) = F_l^{-1}(w)$
accept if $\tilde{a} = a$

Fig. 2. $(n, k, L)^+$ Communication Mode

Moreover, an active adversary can successfully recover the key as follows.

1. Collect a set of messages $O = \{v^1, \ldots, v^s\}$ sent by Bob. The parameter s should be chosen in such a way that O contains a basis for V_l for all $l \in [L]$ with high probability (This can be achieved for $s \in \Theta(L \cdot E(n)) = \Theta(Ln)$, see Sect. 2.2.)
2. Construct an $s \times s$-matrix M over $\{0, 1\}$, where $M_{i,j} = 1$ iff Alice accepts $v^i \oplus v^j$.

Note that if v^i and v^j belong to the same subspace V_l, the probability for $M_{i,j} = 1$ is one. If $\{v^i, v^j\} \not\subseteq V_l$ for all $l \in [L]$ then the probability that $M_{i,j} = 1$ equals the probability that $v_i \oplus v_j \in \bigcup_{l=1}^{L} V_l$, which is at most $(L - 2)2^{-k}$. Hence, it is possible to efficiently compute specifications of $V_1, \ldots, V_L$ and to impersonate Bob by replying with $w \in V_l$ for arbitrary $l \in [L]$.

To prevent this kind of attack we consider the following distributed communication mode, which, analogously to the HB$^+$-protocols, defines $(n, k, L)^+$-protocols. Alice starts by sending a random $a \in \mathrm{GF}(2)^{n/2}$ to Bob. Bob chooses random values $b \in \mathrm{GF}(2)^{n/2}$ and $l \in [L]$ and sends $w = F_l(a, b)$ to Alice. Alice accepts $w \in \mathrm{GF}(2)^{n+k}$ if there is some $l \in [L]$ with $w \in V_l$ and the prefix of length $n/2$ of $F_l^{-1}(w)$ equals a (see Fig. 2).

However, also $(n, k, L)^+$-protocols can be broken by an MITM attack:

```
1. Fix $a_1 \neq \mathbf{0}$ in $\mathrm{GF}(2)^{n/2}$.
2. Send $a_1$ to Bob and receive $w_1 \in V_l$ for some $l \in [L]$.
3. FOR $r = 2, \ldots, s$
   3.1 REPEAT
      3.1.1 Catch $a$ from Alice.
      3.1.2 Send $a' := a \oplus a_1$ to Bob and receive $w'$.
      UNTIL Alice accepts $w' \oplus w_1$ (which happens with probability at least $1/L$).
   3.2 Define $a_r := a'$ and $w_r := w'$.
```

The parameter s is chosen such that $\{w_1, \ldots, w_s\}$ contains a basis of V_l with high probability (see Sect. 2.2). This procedure will be repeated until specifications of $V_1, \ldots, V_L$ have been computed.

In the next subsection we propose linear $(n, k, L)^{++}$-protocols, a slightly modified version of $(n, k, L)^{+}$-protocols, and show that they are secure against a certain type of MITM-attack.

2.4 Linear $(n, k, L)^{++}$-Protocols and Provable Security against MITM-Attacks

The parameters n, k, L as well as V_l, F_l for $l \in [L]$ are defined as above. Let $n = 2N$. The $(n, k, L)^{++}$-protocol works similarly to the $(n, k, L)^{+}$-protocol, but uses an additional publicly known invertible function $f : \mathrm{GF}(2)^n \longrightarrow \mathrm{GF}(2)^n$, which we call connection function (see Fig. 3).

1. Alice chooses a random $a \in \mathrm{GF}(2)^N$, $a \neq \mathbf{0}$, moves to the inner state a and sends a to Bob.
2. Bob chooses random values $b \in \mathrm{GF}(2)^N$ and $l \in [L]$ and sends $w = F_l(f(a, b))$ back to Alice.
3. Alice accepts a message $w \in \mathrm{GF}(2)^n$ in inner state a if
   - $w \neq \mathbf{0}$, and
   - $\exists l \in [L]$ such that $w \in V_l$, and
   - $f^{-1}(F_l^{-1}(w))$ has the form (a, b) for some $b \in \mathrm{GF}(2)^N$.

Note that choosing f to be the identity yields the $(n, k, L)^{+}$-protocol.

We construct now a connection function f which yields provable security of $(n, k, L)^{++}$-protocols with respect to a certain type of MITM-attack which we call (x, y)-equality attack.

The aim of an (x, y)-equality attacker Eve is to generate two messages $w \neq w' \in \mathrm{GF}(2)^{n+k}$ and to efficiently test by MITM-access to the protocol if w and $w \oplus w'$ belong to the same linear subspace V_l for some $l \in [L]$. As described above, such an attack can be used to efficiently compute specifications of the subspaces $V_1, \ldots, V_L$.

Eve works in three phases:

1. Send a message $y \in \mathrm{GF}(2)^N$ to Bob and receive $w' = F_l(f(y, b'))$.
2. Observe a challenge $a \in \mathrm{GF}(2)^N$ sent by Alice.

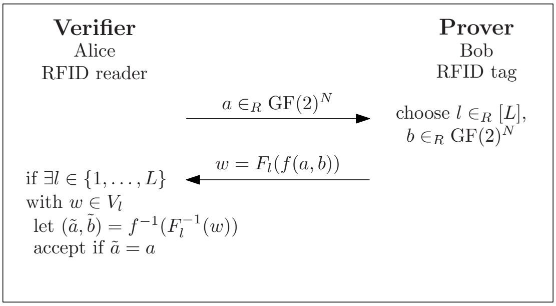
Verifier
Alice
RFID reader
Prover
Bob
RFID tag
$a \in_R \mathrm{GF}(2)^N$
choose $l \in_R [L]$,
$b \in_R \mathrm{GF}(2)^N$
$w = F_l(f(a, b))$
if $\exists l \in \{1, \ldots, L\}$
with $w \in V_l$
let $(\tilde{a}, \tilde{b}) = f^{-1}(F_l^{-1}(w))$
accept if $\tilde{a} = a$

Fig. 3. $(n, k, L)^{++}$ Communication Mode

3. Compute a value $x = x(y, w', a) \in \mathrm{GF}(2)^N$, send it to Bob, receive the message $w = F_r(f(x, b))$ and send $w \oplus w'$ to Alice.

The success probability of the attack is given by the probability that Alice accepts $w \oplus w'$ given that $l = r$.

Note that if f is GF(2)-linear (as in the $(n, k, L)^+$-protocol), then setting $x = a \oplus y$ yields an attack with success probability one.

We define now a connection function which yields provable security against (x, y)-equality attacks. In the following we identify $\{0,1\}^N$ with the finite field $K = \mathbf{F}_{2^N}$ and denote by $+, \cdot$ the addition and multiplication in K. Let the function value $f(a, b)$ for all $a, b \in K$ be defined by

$$f(a, b) = (ab, ab^3) \ .$$

Thus, Alice accepts a message w with $F_l^{-1}(w) = (u, v) \in K^2$ in inner state $a \in K^*$ if $(a^{-1}u)^3 = a^{-1}v$, which is equivalent to $u^3 = a^2 v$.

Theorem 1. *The success probability of an (x, y)-equality attacker against the $(n, k, L)^{++}$-protocol is at most $\frac{3}{2^N-1}$.*

Proof. For given $y, a \in K^*$, Eve has to choose an element $x \in K^*$ such that $w + w' = (u, v) \in K \times K$ will be accepted by Alice in inner state a, where $w = F_l(x, b)$ and $w' = F_l(y, b')$ for some $l \in [L]$, and $b, b' \in K^*$. Note that Eve has no information about b, b', and that $u = xb + yb'$ and $v = xb^3 + yb'^3$.

Consequently, Eve's choice for the value x has to satisfy

$$(xb + yb')^3 = a^2(xb^3 + yb'^3) \ .$$

This is equivalent to

$$(x + yc)^3 = a^2(x + yc^3) \ ,$$

where $c = b'(b^{-1})$, which is equivalent to $P(x, c) = 0$, where the polynomial $P(x, d)$ is for all $d \in K^*$ defined as

$$P(x, d) = x^3 + (yd)x^2 + (y^2d^2 + a^2)x + d^3(y^3 + y^2a^2) \ .$$

Note that there are $|K^*| = 2^N - 1$ different polynomials of type $P(x, d)$ with respect to the variable x (Look at the coefficient yd of x^2).

For all $x \in K^*$ let $P(x) = \{d, P(x, d) = 0\}$. Note that $P(x, d)$ is a polynomial of degree 3 also in the unknown d. This implies that for all $x \in K^*$ it holds $|P(x)| \leq 3$.

Eve has to choose an x that satisfies $c \in P(x)$. Since she does not have any information about c, her success probability is bounded by $\frac{3}{2^N-1}$. □

2.5 Security of (n, k, L)-Protocols and the LULS-Problem

There are several exhaustive search strategies for computing specifications of the secret subspaces $V_1, \ldots, V_L$, see, e.g., the search-for-a-basis heuristic described in Appendix A. The parameters (n, k) should be chosen in such a way that these attacks become infeasible. Moreover, k should be large enough such that the probability p of a random $v \in \mathrm{GF}(2)^{n+k}$ belonging to $\bigcup_{l=1}^{L} V_l$ is negligibly small. Note that $p < L2^{-k}$.

The subspaces $V_1, \ldots, V_L$ should have the property $V_i \oplus V_j = \mathrm{GF}(2)^{n+k}$ for all $i \neq j \in [L]$, otherwise the effective keylength would be reduced. This implies $n \geq k$.

The Learning Unions of L Linear Subspaces (LULS) Problem refers to the following communication game between a learner and an oracle. The oracle holds the specifications of L n-dimensionial linear subspaces $V_1, \ldots, V_L$ of $\mathrm{GF}(2)^{n+k}$. The learner can send requests *hello* to the oracle. If the oracle receives *hello*, it chooses randomly and uniformly an $l \in [L]$ and $v \in V_l$ and sends the (positive) example v to the learner. The aim of the learner is to compute specifications of $V_1, \ldots, V_L$ from a sufficiently large set $v^1, \ldots, v^s$ of examples produced by the oracle. Note that this corresponds to a passive key recovery attack against (n, k, L)-type protocols. As described above, a possible strategy is the search-for-a-basis heuristic, which we outline in Appendix A together with implied suggestions on how to choose n and k.

An active adversary who is able to solve the LULS-problem efficiently can break the $(n, k, L)^+$-protocol. In particular, knowing specifications of the secret subspaces $V_1, \ldots, V_L$, he can generate specifications of the subspaces $V_l(a)$ (i.e., the image of $F_l(a, \cdot)$), for arbitrary $a \in \mathrm{GF}(2)^{n/2}$ and $l \in [L]$ by repeatedly sending a to Bob. Then the adversary uses $N = n/2$ subspaces $V_l(a_i), \ldots, V_l(a_N)$ for $\{a_1, \ldots, a_N\}$ linearly independent to forge a response for a challenge $a = \sum_{i=1}^{N} \alpha_i a_i$ by computing

$$
\begin{aligned}
w &= \sum_{i=1}^{N} \alpha_i v_i \text{ with } v_i \in_R V_l(a_i) \\
&= \sum_{i=1}^{N} \alpha_i F_l(a_i, b_i) \\
&= F_l(a, b') \text{ with } b = \sum_{i=1}^{N} b_i \ .
\end{aligned}
$$

In the case of the $(n, k, L)^{++}$-protocol, the adversary cannot just return a random $w \in V_l(a)$, but has to make sure that the first half of $f^{-1}(F_l^{-1}(w))$ corresponds to a. How such a w can be found efficiently (possibly based on the specifications of the subspaces $V_l(a)$) is a matter of further research.

In Sect. 3 we present and discuss an algebraic learning algorithm for LULS.

3 On Solving the LULS-Problem

3.1 A Learning Algorithm for the LULS-Problem

Recall that the LULS-problem with parameters n, k, L consists in computing specifications of L secret n-dimensional linear subspaces of $\mathrm{GF}(2)^{n+k}$ from positive examples v produced by an oracle which chooses randomly and uniformly $l \in [L]$ and $v \in V_l$. In this paper we treat the case $L = 2$ and consider the special case that $V_l = \{(v, f(v)), v \in \mathrm{GF}(2)^n\}$, $l \in \{1, 2\}$ for secret linear functions $f_1, f_2 : \mathrm{GF}(2)^n \longrightarrow \mathrm{GF}(2)^k$. Our algorithm computes for all $i \in [k]$ specifications of the i-th component functions $f_1^i, f_2^i : \mathrm{GF}(2)^n \longrightarrow \mathrm{GF}(2)$ separately, i.e., it suffices to consider the case $k = 1$. The learning algorithm is based on the following reasoning.

1. Take a set $O = \{(v^1, w_1), \ldots, (v^n, w_n)\} \subseteq \mathrm{GF}(2)^{n+1}$ of examples such that $B = \{v^1, \ldots, v^n\}$ forms a basis of $\mathrm{GF}(2)^n$. For all $i \in [n]$ let x_i and y_i denote the variables corresponding to $f_1(v^i)$ and $f_2(v^i)$, respectively.
2. For $b \in \{0, 1\}$ let $I_b = \{i \in [n], w_i = b\}$.
3. For all $i \in [n]$ let $t_i = x_i \oplus y_i$, and for all $i < j \in [n]$ let $t_{i,j} = x_i y_j \oplus x_j y_i$.
4. Observe that for all $i \in [n]$ the equality $(w_i \oplus x_i)(w_i \oplus y_i) = 0$ holds. This implies
$$x_i y_i = 0 \text{ if } i \in I_0 \text{ and } x_i y_i = 1 \oplus t_i \text{ if } i \in I_1 \ . \tag{1}$$
5. Observe that for each example $(v, w) \in \mathrm{GF}(2)^{n+1}$, $v \notin B$, the following holds: If $v = \bigoplus_{i \in I} v_i$, (i.e., $I \subseteq [n]$ defines the unique representation of v w.r.t. B), then
$$\left(w \oplus \bigoplus_{i \in I} x_i\right)\left(w \oplus \bigoplus_{i \in I} y_i\right) = 0 \ . \tag{2}$$
Observe that relation (2) can be rewritten as a relation $T_B(I, w)$ in the variables t_i and $t_{i,j}$ in the following way. If $w = 0$ then relation (2) is equivalent to $\bigoplus_{i \in I} x_i y_i \oplus \bigoplus_{i < j \in I} t_{i,j} = 0$. Together with relation (1) this implies $\bigoplus_{i \in I_1 \cap I}(t_i \oplus 1) \oplus \bigoplus_{i < j \in I} t_{i,j} = 0$ for $w = 0$. Consequently, for $w = 0$ we define $T_B(I, w)$ as
$$\bigoplus_{i \in I \cap I_1} t_i \oplus \bigoplus_{i < j \in I} t_{i,j} = \begin{cases} 0 & \text{if } |I \cap I_1| \text{ is even} \\ 1 & \text{if } |I \cap I_1| \text{ is odd} \end{cases} .$$
If $w = 1$ then relation (2) is equivalent to $1 \oplus \bigoplus_{i \in I} t_i \oplus \bigoplus_{i \in I \cap I_1}(t_i \oplus 1) \oplus \bigoplus_{i < j \in I} t_{i,j} = 0$. Hence, for $w = 1$ we define $T_B(I, w)$ as
$$\bigoplus_{i \in I \cap I_0} t_i \oplus \bigoplus_{i < j \in I} t_{i,j} = \begin{cases} 0 & \text{if } |I \cap I_1| \text{ is odd} \\ 1 & \text{if } |I \cap I_1| \text{ is even} \end{cases} .$$

Note that a relation similar to relation (2) was also exhibited in [1] for designing an algebraic attack against so-called F_f-protocols.

The learning algorithm now proceeds as follows.

1. Let initially the system LES of linear equations in the $\frac{1}{2}(n^2 + n)$ variables t_i $(i \in [n])$ and $t_{i,j}$ $(i < j \in [n])$ be empty.
2. **REPEAT**
   - 2.1 Choose an observation (v, w), $v \notin B \cup \{\mathbf{0}\}$, and compute the unique subset $I \subseteq [n]$ with $v = \bigoplus_{i \in I} v^i$.

```
    2.2 Enlarge the system $LES$ by the linear equation $T_B(I, w)$.
 3. UNTIL the system $LES$ has $\frac{1}{2}(n^2 + n)$ linearly independent equations.
 4. Compute by Gaussian elimination the unique solution $\theta$ of the system $LES$.
 5. Compute from $\theta$ the unique correct assignments to $x_i$, $y_i$ for all $i \in [n]$.
```

The correct assignments to the x_i and y_i variables (step 5 of the algorithm) can be computed from $\theta = (\theta_i)_{i \in [n]} \, (\theta_{i,j})_{i<j \in [n]}$ as follows.

For $b = 0, 1$ let K_b denote the set $K_b = \{i \in [n], \theta_i = b\}$. We know that for all $i \in K_0$ it holds that $x_i = y_i = w_i$, and for all $i \in K_1$ it holds that $y_i = x_i \oplus 1$. This implies that for all $i < j$ in K_1, $\theta_{i,j}$ satisfies

$$\theta_{i,j} = x_i(x_j \oplus 1) \oplus x_j(x_i \oplus 1) = x_i \oplus x_j \ .$$

This yields a system LES^* of $1/2|K_1|(|K_1| - 1)$ linear equations in the variables x_i, $i \in K_1$, of rank $|K_1| - 1$. As it does not matter which of the two secret linear subspaces we denote by V_1 and which by V_2, we have the freedom to set $x_k = 0$ for some fixed $k \in K_1$. The system LES^* together with $x_k = 0$ yields a system of full rank and allows to compute the correct assigment to the x_i-variables by Gaussian elimination.

3.2 Analysis and Experimental Results

The background for the fact that the repeat cycle of the algorithm is left after a finite number of rounds is that the following $(2^n - (n+1)) \times (n(n+1)/2)$-matrix $M(n)$ over GF(2) has full row rank (which is not hard to show). The row indices of $M(n)$ are all subsets $I \subseteq [n]$ with $|I| \geq 2$, the column indices are $[n] \cup \{(i,j), 1 \leq i < j \leq n\}$. We have $M(n)_{I,i} = 1$ iff $i \in I$ and $M(n)_{I,(i,j)} = 1$ iff $\{i,j\} \subseteq [n]$.

We do not give here a theoretical analysis of the expected number of rounds of the repeat cycle. Our experiments show that the algorithm needs only slightly more than $\frac{1}{2}(n^2 + n) + n$ observations to compute the secret functions f_1 and f_2. Particularly for $n = 128$, we need approx. 8390 examples and 4 minutes on a 3.4 GHz Intel Pentium IV with 4 GB RAM and Magma V2.15-9 [2].

How severe is the restriction that the secret subspaces have the special form $V = \{(v, f(v)), v \in \mathrm{GF}(2)^n\}$ for some surjective linear mapping $f : \mathrm{GF}(2)^n \longrightarrow \mathrm{GF}(2)^k$? Let us consider the general case $V = \{A \circ v, v \in \mathrm{GF}(2)^n\}$ for an $(n+k) \times n$ matrix A. V can be written in the special form iff the first n rows of A are linearly independent. For randomly chosen A this is true with probability $p(n) \approx 0.2887$ (see Sect. 2.2).

We have seen that we could solve the LULS-problem with parameters $(n, k, 2)$ by solving k LULS-problems with parameters $(n, 1, 2)$.

For the special LULS-problem with parameters $(n, 1, L)$, $L > 2$, we can define a similar system LES consisting of degree-L equations in the variables x_i^l, $i \in [n]$, $l \in [L]$, induced as above by equations of the form

$$\left(w \oplus \bigoplus_{i \in I} x_i^1 \right) \ldots \left(w \oplus \bigoplus_{i \in I} x_i^L \right) = 0 \ . \tag{3}$$

The problem is that for $L > 2$ the equations have several symmetries such that the system can not be solved uniquely. The way out is to

- Choose an appropriate parameter $s < k$ which divides k, let $k = s \cdot p$,
- Write vectors $w \in \mathrm{GF}(2)^k$ as vectors $w \in \mathrm{GF}(2^s)^p$, and
- Solve the corresponding p LULS-problem with parameters $(n, 1, L)$ over $\mathrm{GF}(2^s)$.

How to find the best choices of s is a matter of further theoretical and experimental research.

We are convinced that there is no faster way to solve an (n, k, L)-LULS-problem other than solving a system of degree-L equations in Ln variables (if n, k, L are appropriately chosen). Such a system is defined over at least $\Phi(n, L) = \binom{n}{L} + 2\sum_{k=1}^{L-1}\binom{n}{k}$ different monomials, i.e., solving it by linearization means to solve a system of linear equations of size $\Phi(n, L)$. This will cost $\mathcal{O}(\Phi(n, L)^3)$ operations, which can be considered infeasible already for $(n, L) \in \{(128, 5), (256, 4)\}$, since $\Phi(128, 5) \approx 2^{28}$ and $\Phi(256, 4) \approx 2^{27}$.

4 Summary

We have seen that the secret key of CKK^2-protocols can be computed very quickly from a sufficiently large set of messages sent by Bob. This kind of protocol should not be used in practice.

The parameters of $(n, k, L)^{++}$-protocols have to be chosen in such a way that solving the LULS problem with parameters $(\frac{n}{2}, k, L)$ is infeasible. We recommend to use $n = 256$, $k = 64$ and $L = 5$.

Another interesting question is to search for simpler nonlinear connection functions f, for which a security proof can be found. In our proposal, for computing $f(a, b)$ Bob has to perform three multiplications in the finite field of order $2^{n/2}$.

It is another interesting open question whether the very symmetrically structured systems of degree-L equations arising in our LULS-algorithm in Sect. 3 can be more efficiently solved by more advanced techniques like the F4- or F5-algorithm or cube attacks [8,9,5,6]. If one could generate convincing evidence that such algorithms cannot beat our linearization attack, then $(n, k, L)^{++}$-protocols with the above parameters could be seriously considered for practical use.

A problem of (n, k, L)-protocols is the large key length in the case that random mappings $F_1, \ldots, F_L$ are used. It is an important task to look for secure and efficient ways to generate pseudorandom keys. In this context, the (still unbroken) $\mathsf{CKK}^{\sigma,L}$-protocols could become interesting. However, we conjecture that $\mathsf{CKK}^{\sigma,L}$-protocols can be efficently broken.

Interesting suggestions for keylength reductions have been made in [11] and [3]. Adapting these ideas to (n, k, L)-protocols would mean

- To consider special forms of secret subspaces $V_l = \{(A_l \circ v), v \in \mathrm{GF}(2)^n\}$, where A_l denotes a secret $(n + k) \times n$ Toeplitz matrix [11], and

- To define the Toeplitz matrix A_l to be generated by a secret Linear Feedback Shift Register [3].

Checking the feasibility and security of these constructions should be a matter of further research.

Acknowledgement

We are very thankful to Mirek Kutyłowski for the introduction to the topic, and to Matthias Hamann, Stefan Lucks, Willi Meier, Frederik Armknecht, and Erik Zenner for helpful discussions.

References

1. Blass, E.-O., Kurmus, A., Molva, R., Noubir, G., Shikfa, A.: The F_f-family of protocols for RFID-privacy and authentication, `http://eprint.iacr.org/2008/476`
2. Bosma, W., Cannon, J., Playoust, C.: The magma algebra system. i. the user language. J. Symbolic Comput. 24, 235–265 (1997)
3. Bringer, J., Chabanne, H.: Trusted-HB: A low cost version of HB$^+$ secure against a man-in-the-middle attack. IEEE Trans. Inform. Theor. 54, 4339–4342 (2008)
4. Cichoń, J., Klonowski, M., Kutyłowski, M.: Privacy protection for RFID with hidden subset identifiers. In: Indulska, J., Patterson, D.J., Rodden, T., Ott, M. (eds.) PERVASIVE 2008. LNCS, vol. 5013, pp. 298–314. Springer, Heidelberg (2008)
5. Dinur, I., Shamir, A.: Cube attacks on tweakable black box polynomials. Cryptology ePrint Archive, Report 2008/385 (2008), `http://eprint.iacr.org`
6. Dinur, I., Shamir, A.: Cube attacks on tweakable black box polynomials. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 278–299. Springer, Heidelberg (2009)
7. Gołębiewski, Z., Majcher, K., Zagórski, F.: Attacks on CKK family of RFID authentication protocols. In: Coudert, D., Simplot-Ryl, D., Stojmenovic, I. (eds.) ADHOC-NOW 2008. LNCS, vol. 5198, pp. 241–250. Springer, Heidelberg (2008)
8. Faugère, J.-C.: A new efficient algorithm for computing Gröbner bases (F4). J. Pure Appl. Algebra 139, 61–68 (1999)
9. Faugère, J.-C.: A new efficient algorithm for computing Gröbner basis without reduction to zero (F5). In: Mora, T. (ed.) ISSAC 2002, pp. 75–83. ACM Press, New York (2002)
10. Frumkin, D., Shamir, A.: Untrusted-HB: Security vulnerabilities of Trusted-HB. Cryptology ePrint Archive, Report 2009/044 (2009), `http://eprint.iacr.org`
11. Gilbert, H., Robshaw, M.J.B., Seurin, Y.: HB$^\#$: Increasing the security and efficiency of HB$^+$. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 361–378. Springer, Heidelberg (2008)
12. Gilbert, H., Robshaw, M.J.B., Sibert, H.: Active attack against HB$^+$: A provable secure lightweight authentication protocol. Electronic Letters 41, 1169–1170 (2005)
13. Juels, A.: RFID privacy: A technical primer for the non-technical reader. In: Strandburg, K., Raicu, D.S. (eds.) Privacy and Technologies of Identity: A Cross-Disciplinary Conversation. Springer, Heidelberg (2005)
14. Juels, A., Weis, S.A.: Authenticating pervasive devices with human protocols. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 293–308. Springer, Heidelberg (2005)

15. Langheinrich, M.: A survey of RFID privacy approaches. J. Personal and Ubiquitous Comp. 13, 413–421 (2009)
16. Ouafi, K., Overbeck, R., Vaudenay, S.: On the security of HB$^\#$ against a man-in-the-middle attack. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 108–124. Springer, Heidelberg (2008)

A The Search-for-a-Basis Heuristic

The search-for-a-basis heuristic tries to construct a set Q of examples which form a basis of V_l for some $l \in L$. For all linearly independent sets Q of n examples let $p(Q)$ denote the probability that an example coming from the oracle belongs to the linear span $< Q >$ of Q. It is quite obvious that $p(Q)$ is maximal if Q is a basis of V_l for some $l \in L$. If $p(Q)$ is not too small, we can compute an approximation $\tilde{p}(Q)$ of $p(q)$ by testing for $w \in< Q >$ for a sufficiently large number of examples w.

For $v \in Q$ and $w \notin Q$ we denote by $Q(v, w)$ the set obtained by replacing v by w in Q.

The idea of the heuristic is to start with an arbitrary linear independent set Q of n examples and to try to improve this set by finding $v \in Q$ and $w \notin Q$ such that $\tilde{p}(Q) < \tilde{p}(Q(v, w))$. Iterating this at most n times yields a basis for V_l for some $l \in [L]$.

This kind of heuristic is infeasible if the following condition is fulfilled. For a random linear independent set Q of n examples the probability $p(Q)$ is negligibly small with probability $1 - \epsilon$, ϵ negligibly small. The parameters n, k should be chosen in such a way that this condition is guaranteed.

We estimate the probability $p(Q)$ for the case $L = 2$. For a linear independent set Q of n examples let $Q = Q_1 \cup Q_2$, where $Q_1 \subseteq V_1$ and $Q_2 \subseteq V_2 \setminus V_1$. W.l.o.g. let $|Q_1| = n/2 + s$ and $|Q_2| = n/2 - s$. The event $w \in< Q >$ happens iff $w \in V_1 \cap < Q_1 >$ or $w \in V_2$ and $w \in V_2 \cap < Q_1 >$, i.e.,

$$p(Q) \leq \frac{1}{2}\left(2^{s-n/2} + 2^-k\right) \quad .$$

(Note that $\dim(V_1 \cap V_2) = n - k$ for random n-dimensional subspaces V_1, V_2). If n, k are chosen in such a way that 2^{-k}, $2^{-n/4}$ and the probability that $|v| \notin [n/4, 3n/4]$ are negligibly small, then the above condition is fulfilled (note that the expected value of s is $2^{-k}n/2$).