Combinatorial Digital Signature Scheme

*

1st Jean-Charles FAUGÈRE
INRIA and UPMC Uni Paris 6
Sorbonne Universities
Paris, France
jean-charles.faugere@inria.fr

2nd Eliane KOUSSA
Versailles Laboratory of Mathematics
UVSQ, University of Paris-Saclay
Versailles, France
eliane.koussa@uvsq.fr

3rd Gilles MACARIO-RAT
Orange Labs
Paris, France
gilles.macariorat@orange.com

4thJacques PATARIN
CNRS, Versailles Laboratory of Mathematics
UVSQ, University of Paris-Saclay
, Versailles, France
jpatarin@club-internet.fr

5th Ludovic PERRET
INRIA and UPMC Uni Paris 6
Sorbonne Universities
Paris, France
Ludovic.Perret@lip6.fr

***Abstract*—We present here a new signature scheme based on a combinatorial problem named the Permuted Kernel Problem (PKP) [Sha89]. PKP is an NP-complete [GJ79] algebraic problem that consists of simple mathematical operations and involves only basic linear algebra.**
To solve PKP is to find a particular kernel vector for a publicly known matrix. Through the complexity analysis of solving PKP, we found the opposite of what is presented in [JJ01]. Precisely, we noticed that the most efficient algorithm for solving PKP remains the one which was introduced by J. PATARIN and P. CHAUVAUD in [PC93]. Moreover, PKP has always had the reputation of being the best combinatorial algorithm known for authentication. It was used to build the first Identification Scheme (IDS) which has an efficient implementation on low-cost smart cards. Consequently, and via the traditional Fiat-Shamir (FS) paradigm, we derive the signature scheme PKP-DSS from a Zero-Knowledge Identification Scheme (ZK-IDS) based on PKP [Sha89].
Thus, PKP-DSS has a security that can be provably reduced, in the *(classical) random oracle model*, to essentially the hardness of random instances of PKP.
Also, we propose different sets of parameters according to several security levels. Each parameter set arises signatures of length comparable to the other signatures derived from Zero-Knowledge identification schemes. In particular, PKP-DSS-128 gives a signature size approximately about 18 KBytes for 128 bits of classical security, while the best known signature schemes built from a ZK-IDS (such as MQDSS [CHR+18], Picnic [CDG+17],...) give similar signatures ($\approx$ 16 KB for MQDSS, $\approx$ 33 KB for Picnic,...).
Since there are no known quantum attacks for solving PKP, we believe that the recommended sets of parameters provide a quantum secure scheme.

***Index Terms*—public-key cryptography, post-quantum cryptography, Fiat-Shamir, 5-pass identification scheme, Permuted Kernel Problem.**

I. INTRODUCTION

The construction of large quantum computers would break all public-key cryptographic schemes in use today based on the traditional number-theoretic problems. Despite the fact that it isn't clear when and even if enormous quantum computations would be feasible, it is important to anticipate a technological breakthrough and design new public key cryptosytems that are resistant to quantum attacks.

Due to the call for post-quantum standards of the NIST (https://www.nist.gov/), there has been renewed interest in the transformed Zero-Knowledge Identification Schemes into Digital Signatures Schemes (DSS) via the Fiat-Shamir paradigm [FS86]. This transformation method is important since it yields to efficient signature schemes in terms of minimal and sufficient security assumptions.

Particularly, we are interested in the post-quantum cryptographic schemes which belongs to the post-quantum branch whose security relies on the fact that there is no quantum algorithms known to solve NP-Complete problems [BBBV97]. Namely, the Permuted Kernel Problem: the problem of finding a permutation of a known vector such that the resulting vector is in the kernel of a given matrix.

Here, we study the application in cryptography of the PKP problem over a finite field. We are essentially concerned about this problem because it can be used to build a post-quantum signature scheme based on the hardness of solving random instances of PKP. It is an old-time combinatorial NP-Complete problem. It requires simple operations which involve basic linear algebra computations. For a little long time, no new attacks on PKP were reported which makes the construction of schemes based on hard instances of this problem more applicable.

II. MAIN RESULTS

The main contribution of this paper is to present a new post-quantum signature scheme.
In [JJ01], a new approach to attack PKP was introduced. The authors of this latter, assume that the technique described

in their article, is faster than any previously known method. But, after the complexity analysis of the PKP problem, it appears that we have different results: the improved algorithm presented in [PC93] form the best attack on PKP. Besides, we are particularly interested in the design of a signature scheme. Similarly to the approaches cited above, by applying the Fiat-Shamir transform, we study the design of post-quantum signature constructed from a 5-pass authentication scheme based on the PKP problem.

Our objective is to define the most optimal parameters for hard instances of this problem, with respect to the security levels identified by NIST [NIS].
The PKP-DSS scheme based on PKP compared well with the other similar (in terms of construction) schemes. We obtained the following results: comparable signature size for the same security levels. Then, this makes the signature scheme based on PKP a competitive cryptosystem.

III. The Permuted Kernel Problem

In order to introduce the signature scheme, we first present the PKP problem [Sha89]. We also present the best technique for solving it.

A. Introduction to PKP

PKP [Sha89], [GJ79] is the problem on which the security of PKP-DSS is based. PKP is a linear algebra problem which asks to find a kernel vector of given matrix under a vector-entries constraint. It's a generalization of the Partition problem [GJ79, pg.224]. More precisely, it is defined as follows:

Input. A finite field $\mathbb{F}_p$, a matrix $A \in \mathscr{M}_{m\times n}(\mathbb{F}_p)$ and a n-vector $V \in {\mathbb{F}_p}^n$.
Question. Find a permutation π over $(1,\ldots,n)$ *such that* $A \times V_\pi = 0$, *where* $V_\pi = (V_{\pi(j)}), j = 1,\ldots,n$.

A reduction of the 3-Partition problem proves PKP to be NP-Complete [GJ79] in the good reasoning (*i.e.*its hardness grows exponentially with p). A fundamental design assumption of PKP-DSS is that solving random instances of PKP are hard to solve in practice (Section IV). In fact, the solidity of PKP comes from, on the one hand, the big number of permutations, on the other hand, from the small number of possible permutations which may suit the kernel equations. More precisely, PKP is hard because it obligates the choice of a vector, with already fixed set of entries, from the kernel of the matrix A.
Note that, to reach higher security levels, it's more desirable that the n-vector V has distinct coordinates.

IV. Best known attacks

The implementation's efficiency of the first IDS, proposed by A. Shamir [Sha89], based on PKP problem has led to several solving tools. In fact, there are various attacks for PKP, which are all exponential.

In [Geo92], J. Georgiades presents symmetric polynomials equations which will be utilized by all the other attacks. The authors of [BCCG92] investigate also the security of PKP, where a time-memory trade-off was introduced. Moreover, J. Patarin and P. Chauvaud improve algorithms for the Permuted Kernel Problem [PC93]. Also, in [JJ01], a new time-memory trade-off was proposed. After all, it appears that the attack of Patarin-Chauvaud [PC93] is the most efficient one. The details of each attack and the numerical results are given in the main article.

We assume that the matrix $A \in \mathscr{M}_{m\times n}(\mathbb{F}_p)$ is of rank m, given in a systematic form:

$$A = (a_{ij})_{1\le i\le m, 1\le j\le n} \;=\; \left[A' | I\right],$$

where $A' = (a'_{ij})_{1\le i\le m, 1\le j\le n-m} \in \mathscr{M}_{m\times n-m}(\mathbb{F}_p)$ and I is the identity matrix of size m. By denoting $A_\pi = (a_{i\pi(j)})$, the effect of the permutation π over the columns of A, it's easy to see that $A_\pi V_\pi = AV$.

A. Brute-force search

First of all, let's consider the exhaustive search. This test consists of examining all the possible candidates (permutations of a set on n elements) for the solution in order to determine whether a candidate satisfies the problem's conditions. Despite the fact that this search technique is very general and naive, mainly in this case, where the search space is large, it is important to consider its complexity which is in $n!$.

B. A new Approach of A. Joux and E. Jaulmes

In [JJ01], A. Joux and E. Jaulmes introduce a new time-memory trade-off algorithm which is an application of the algorithm described in [JL01] to the Permuted Kernel Problem. In fact, the algorithm consists of two main steps: A-Phase and B-Phase. The authors of [JJ01] assume that the B-Phase controls the time complexity of this approach. Without going too far into the analysis of this technique, we found that the contrary is true. By considering a reasonable choices of parameters, it turns out that the time complexity of the algorithm is dominated by the A-Phase. This is one of the most interesting points of our article.

Thus, this attack is not the most efficient for solving PKP but, the attack of Patarin-Chauvaud [PC93]. This also shows that the PKP problem is more difficult to attack than we thought before our article, which has a good impact when it comes to the PKP-based signature scheme.

C. Improved algorithms for PKP

J. Patarin and P. Chauvaud combine in [PC93] the two ideas presented in the previous attacks (see [BCCG92], [Geo92]). The result was a reduction in the time required to attack PKP. They also present some new ideas in order to reduce this time the memory needed.
Thus, this leads to a new algorithm which is quicker and more efficient than the attacks cited above. The details and the numerical results are given in the main article [PC93]. Here is the Magma code that gives the time complexity of the last improved algorithm.

```
patarin6:=function(n,m,p)
cmin:=999999;
```

```
Lp:=Ceiling(Log(2,p));
nS :=Binomial(n,m);
NU:=Round((1-&*[ 1-p^(i-m) :
                 i in [0..m-1]])*nS);
PR:=[ Round( m*(nS-NU)*Binomial(m,i)
       *(1-1/p)^(m-i)/p^i)  : i in [0..m]];
a:=Maximum([ i+1 : i in [0..m] | PR[i+1] gt
                                    100 ]);
for k:=2 to m do;
r:=n-m-1+k;
for l:=1 to r do;
s:=r-l;
if s+a lt n-m then continue; end if;
pr:=Maximum(1,Factorial(n)/Factorial(n-l)
            /p^(k-1));
for t:=1 to s-1 do;
u:=s-t;
pr2:=Maximum(1,Factorial(n-u)/
                  Factorial(n-s)/p);
step1 := Factorial(n)/Factorial(n-l);
step20 := Factorial(n-u)/Factorial(n-s);
step2 := Binomial(n,u) * ( step20 +
             Factorial (u) * pr2 * pr);
ctime:=  Log(2,Maximum(step1, step2));
if ctime lt cmin then cmin:=ctime;
nbper:=Round(Log(2,Factorial(n)/p^m));
nbker:=Round(Log(2,Factorial(p)
            /Factorial(p-m)/p^m));
mem:=Round(Log(2,step1*l*Lp));
U:=ctime;
end if;
end for;
end for;
end for;
return U;
end function;
```

V. Identification scheme (IDS) based on PKP

In this section, we present the 5-pass Zero-Knowledge Identification Scheme (ZK-IDS) based on the computational hardness of PKP [Sha89], [LP11], noted here PKP-IDS.

We first quote and refer to some of the general definitions given in [CHR$^+$18] : Identification scheme, Completeness, Soundness (with soundness error), Honest-verifier zero-knowledge, and also in [HNO$^+$09], [Dam99] : statistically hiding commitment, computationally binding commitment. We then apply and adapt these definitions to the Identification scheme base on PKP and give and prove its own properties of performance and security. This approach will be more convenient for presenting the signature scheme in the next section.

A. Preliminaries

In what follows and as in [CHR$^+$18], we assume the existence of a non-interactive commitment scheme *Com* which verifies the two properties : statistically hiding and computationally binding (see [HNO$^+$09], [Dam99] for details). The commitments are computed using the function *Com*. Note that, it is possible to let *Com* be $\mathscr{H}$ a one way hash and collision intractable function, behaving like a random oracle.

B. PKP 5-pass IDS

In this section, we present (slightly modified version of) PKP-IDS. It can be described as three probabilistic polynomial time algorithms $IDS = (\text{KeyGen}, \mathscr{P}, \mathscr{V})$ for which we give below a literal description. The security parameter of the identification scheme is noted λ.

Generation of the public key and secret key in PKP-IDS. The users first agree on a prime number p, and a $m \times n$ matrix A with coefficients in $\mathbb{F}_p$. The public-key in PKP-IDS is given by an instance of PKP with a *preassigned* solution that will be the secret-key. Thus, each user picks a (right) kernel-vector $W \in$ Ker(A), then randomly generates a secret permutation of n elements $\mathsf{sk} = \pi$ and finishes by computing $V = W_{\pi^{-1}}$.

We summarize the public-key/secret-key generation in Algorithm 1. It takes the security parameter λ as input.

Algorithm 1 pk/sk generation in PKP-IDS

1: **procedure** PKP-IDS.KeyGen(1^λ)
2: pk.seed $\leftarrow$ Randomly sample λ bits
3: Randomly sample a matrix $A \in \mathscr{M}_{m\times n}(\mathbb{F}_p)$ using a pseudo-random generator with pk.seed
4: Randomly pick a n-vector $W \in Ker(A)$
5: sk.seed $\leftarrow$ Randomly sample λ bits
6: Generate a random permutation $\pi \in S_n$ using a pseudo-random generator with sk.seed
7: $\mathsf{sk} \leftarrow \pi$
8: Compute $V = W_{\pi^{-1}}$
9: $\mathsf{pk} \leftarrow (p,$ pk.seed$, V)$
10: **Return** $(\mathsf{pk}, \mathsf{sk})$
11: **end procedure**

One 5-pass round of identification : Prover $\mathscr{P}$ and Verifier $\mathscr{V}$. Prover and Verifier are interactive algorithms that realize the identification protocol in 5 passes. The 5 passes consist in one commitment and two responses transmitted from the prover to the verifier and two challenges transmitted from the verifier to the prover. Random choices of prover and verifier are made using the uniform distribution. The protocol of identification is summarized in Algorithm 2.

From Shamir in [Sha89] we have the following results.

Theorem 5.1: PKP-IDS is complete. PKP-IDS is statistically zero knowledge when the commitment scheme *Com* is computationally binding. PKP-IDS is sound with soundness error $\frac{p+1}{2p}$ when the commitment scheme *Com* is computationally binding.

Definition 5.2 (N rounds of PKP-IDS): Let PKP-IDS$=(\text{KeyGen}, \mathscr{P}, \mathscr{V})$ then $PKP-IDS^N = (\text{KeyGen}, \mathscr{P}^N, \mathscr{V}^N)$ is the parallel composition of N rounds of $PKP-IDS$.

Key sizes. The secret key is the permutation π obtained using a pseudo random generator that takes as input a seed of

Algorithm 2 One round of the 5-pass identification scheme

procedures $\mathscr{P}(\mathsf{sk}), \mathscr{V}(\mathsf{pk})$
  //Prover setup
  $\mathscr{P}$ sets $R \leftarrow$ Random vector in ${\mathbb{F}_p}^n$
  $\mathscr{P}$ sets $\sigma.\mathsf{seed} \leftarrow$ Random seed of λ bits
  $\mathscr{P}$ sets $\sigma \leftarrow$ Random permutation in S_n using a pseudo-random generator with $\sigma.\mathsf{seed}$
  //Commitment step by the Prover
  $\mathscr{P}$ sets $\mathsf{C}_0 \leftarrow Com\big(\sigma, AR\big)$
  $\mathscr{P}$ sets $\mathsf{C}_1 \leftarrow Com\big(\pi\sigma, R_\sigma\big)$
  $\mathscr{P}$ sends $(\mathsf{C}_0, \mathsf{C}_1)$ to $\mathscr{V}$
  //First challenge by the verifier
  $\mathscr{V}$ sets $\mathsf{Ch}_0 \leftarrow$ c random in $\mathbb{F}_p$
  $\mathscr{V}$ sends Ch_0 to $\mathscr{P}$
  $\mathscr{P}$ sets $Z \leftarrow R_\sigma + cV_{\pi\sigma}$ and sends Z to $\mathscr{V}$
  $\mathscr{V}$ sets $\mathsf{Ch}_1 \leftarrow$ b random bit
  $\mathscr{V}$ sends Ch_1 to $\mathscr{P}$
  if $\mathsf{Ch}_1 = 0$ **then**
    $\mathscr{P}$ reveals $\sigma.\mathsf{seed}$ to $\mathscr{V}$
    $\mathscr{V}$ accepts if $Com\big(\sigma, A_\sigma Z\big) = \mathsf{C}_0$
  else
    $\mathscr{P}$ reveals $\pi\sigma$ to $\mathscr{V}$
    $\mathscr{V}$ accepts if $Com\big(\pi\sigma, Z - cV_{\pi\sigma}\big) = \mathsf{C}_1$
  end if
end procedure

λ bits. The size of the public vector V is $n\log_2(p)$ bits. The bit size of the public key (p, A, V) is:

$$\log_2(p) + \lambda + n\log_2(p) \ bits.$$

Performance of the scheme. We can now provide the communication complexity of the IDS, where its fraud's probability is $\frac{p+1}{2p}$. Consider that the commitment function *Com* used in the protocol, returns values of 2λ bits. The transfer of the n-vector $Z \in {\mathbb{F}_p}^n$ requires $n\log_2 p$ Thus, the fourth passes demand $4\lambda + (n+1)\log_2 p + 1$ bits.

Note also that, compared to the original scheme of Shamir in [Sha89], we have reduced the complexity in communication by revealing only the seed used to generate the random elements. More precisely, instead of revealing the random permutation σ, the prover $\mathscr{P}$ only sends its seed sigma.seed.

So, the last pass needs, according to Ch_1, λ bits to reveal the permutation σ if $\mathsf{Ch}_1 = 0$; and $\log_2(n!)$ bits to reveal the permutation $\pi\sigma$, if $\mathsf{Ch}_1 = 1$.
In total, the weighted average bit complexity of the scheme repeated N rounds is given by:

$$\Big(4\lambda + (n+1)\log_2 p + 1 + \frac{1}{2}(\lambda + \log_2(n!))\Big) \times N.$$

VI. Digital signature scheme (DSS) based on PKP

We present here the main contribution of this work which is to construct a DSS *i.e.* a digital signature scheme, based on the PKP problem, from the IDS defined in Section V-B. This construction uses the well-known Fiat Shamir transformation [FS86].

So next, we introduce the basic definitions needed. Then, similarly to the MQ-based signatures and Picnic, we define our scheme, and we finish with a comparison with other cryptosystems.

A. Introduction

The classical method of Fiat-Shamir (FS) transforms an interactive proof of knowledge (identification scheme) into a non interactive one (signature scheme). This work is a direct application of this method to get PKP-DSS from PKP-IDS. **Fiat-Shamir transform for PKP-IDS.** We recall that PKP-IDS the previously defined identification scheme achieves soundness with soundness error $\kappa = \frac{1+p}{2p}$. We select N the number of parallel rounds of PKP-IDS such that κ^N is negligible in λ. We select two cryptographic hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{F}_p^N$ and $H_2 : \{0,1\}^* \rightarrow \{0,1\}^N$. By applying Construction 4.7 in [CHR+18], we get PKP-DSS= (KeyGen, Sign, Verify). See Algorithms 3 and 4.

A valid signature of a message m by PKP-DSS is then a tuple $(m, \sigma_0, \sigma_1, \sigma_2)$, where $\sigma_0, \sigma_1, \sigma_2$ hold the (vector of parallel) commitments and responses of the non interactive prover. The implicit values $h_1 = H_1(m, \sigma_0)$ and $h2 = H_2(m, \sigma_0, h_1, \sigma_1)$ represent the (vector of parallel) challenges of the non interactive verifier.

We get the similar result as Th. 5.1 in [CHR+18].

Theorem 6.1: PKP-DSS is *Existential-Unforgeable under Chosen Adaptive Message Attacks* (EU-CMA) in the random oracle model, if

- the search version of the Permuted Kernel problem is intractable,
- the hash functions are modeled as random oracles,
- the commitment functions are computationally binding, computationally hiding, and the probability that their output takes a given value is negligible in the security parameter,
- the pseudo-random generators are modeled as random oracle, and
- the pseudo-random generators have outputs computationally indistinguishable from random.

The proof is exactly the same as in [CHR+18].

B. Quantum analysis of PKP

Since we are comparing PKP-DSS to other post-quantum schemes, it is important to define the security of our scheme against quantum attacks. This can be done by investigating quantum algorithms for solving PKP. However, till now there are no quantum versions of the known attacks on PKP cited above IV. Also, there is no gain of considering Grover's algorithm because the best attack for solving PKP is much more efficient than the exhaustive search (by a quadratic factor).

Moreover, the post-quantum security of the Fiat-Shamir transform has been studied in [Unr15], [Unr17]. The author of [Unr15] explains that the classic Fiat-Shamir transform might not be secure against quantum computers. Thus, a new technique with the extra property of "extractability" was

Algorithm 3 Signing process in PKP-DSS

procedure PKP-DSS.SIGN(m, sk)
  $\mathsf{R} \leftarrow \mathscr{H}_0(\mathsf{sk} \,||\, m)$, R is a message-dependent random value
  $\mathsf{D} \leftarrow \mathscr{H}_0(\mathsf{pk} \,||\, \mathsf{R} \,||\, m)$, D is the randomized message digest
  $\mathsf{R}^{(1)}, \ldots, \mathsf{R}^{(\mathsf{N})} \leftarrow \mathsf{RG}_0(\mathsf{R.seed},\ \mathsf{D})$
  $\sigma^{(1)}, \ldots, \sigma^{(\mathsf{N})} \leftarrow \mathsf{RG}_1(\sigma.\mathsf{seed},\ \mathsf{D})$
  for j *from* 1 *to* N **do**
    $\mathsf{C}_0^{(j)} = Com(\sigma^{(j)}, AR^{(j)})$,
    $\mathsf{C}_1^{(j)} = Com(\pi\sigma^{(j)}, R^{(j)}_{\sigma^{(j)}})$.
    $\mathsf{COM^{(j)}} := (\mathsf{C}_0^{(j)},\ \mathsf{C}_1^{(j)})$
  end for
  $\mathscr{S}_0 \leftarrow \mathscr{H}_0(\mathsf{COM}^{(1)}|| \ \ldots\ ||\mathsf{COM}^{(N)})$.
  $\mathsf{Ch}_0 \leftarrow \mathscr{H}_1(\mathsf{D},\ \mathscr{S}_0)$
  Parse Ch_0 *as* $\mathsf{Ch}_0 := (c^{(1)}, \ldots, c^{(N)})$, $c^{(j)} \in \mathbb{F}_p$
  for j *from* 1 *to* N **do**
    $Z^{(j)} \leftarrow R^{(j)}_{\sigma^{(j)}} + c^{(j)} V_{\pi\sigma^{(j)}}$,
    $\mathsf{resp}_0^{(j)} := Z^{(j)}$.
  end for
  $\mathscr{S}_1 \leftarrow (\mathsf{resp}_0^{(1)}|| \ \ldots\ ||\mathsf{resp}_0^{(N)}) = (Z^{(1)}|| \ \ldots\ ||, Z^{(N)})$.
  $\mathsf{Ch}_1 \leftarrow \mathscr{H}_2(\mathsf{D},\ \mathscr{S}_0,\ \mathsf{Ch}_0,\ \mathscr{S}_1)$
  Parse Ch_1 *as* $\mathsf{Ch}_1 := (b^{(1)}, \ldots, b^{(N)})$, $b^{(j)} \in \{0,1\}$
  for j *in* $(1 \ldots N)$ **do**
    if $b^{(j)} = 0$ **then**
      $\mathsf{resp}_1^{(j)} \leftarrow \sigma^{(j)}$.
    else
      $\mathsf{resp}_1^{(j)} \leftarrow \pi\sigma^{(j)}$.
    end if
  end for
  $\mathscr{S}_2 \leftarrow (\mathsf{resp}_1^{(1)}|| \ \ldots\ ||\mathsf{resp}_1^{(N)}||\mathsf{C}^{(1)}_{1-b^{(1)}}|| \ \ldots\ ||\mathsf{C}^{(N)}_{1-b^{(N)}})$.
Return $(\mathsf{R},\ \mathscr{S}_0,\ \mathscr{S}_1,\ \mathscr{S}_2)$.
end procedure

Algorithm 4 Verification process in PKP-DSS

procedure PKP-DSS.VERIFY$(m, \mathsf{pk}, \mathscr{S} = (\mathsf{R}, \mathscr{S}_0, \mathscr{S}_1, \mathscr{S}_2))$
  $\mathsf{D} \leftarrow \mathscr{H}_0(\mathsf{pk} \,||\, \mathsf{R} \,||\, m)$, D is the randomized message digest
  $\mathsf{Ch}_0 \leftarrow \mathscr{H}_1(\ (\mathsf{D},\ \mathscr{S}_0)\ )$
  Parse Ch_0 *as* $\mathsf{Ch}_0 := (c^{(1)}, \ldots, c^{(N)})$, $c^{(j)} \in \mathbb{F}_p$
  $\mathsf{Ch}_1 \leftarrow \mathscr{H}_2(\mathsf{D},\ \mathscr{S}_0,\ \mathsf{Ch}_0,\ \mathscr{S}_1)$
  Parse Ch_1 *as* $\mathsf{Ch}_1 := (b^{(1)}, \ldots, b^{(N)})$, $b^{(j)} \in \{0,1\}$
  Parse $\mathscr{S}_1$ *as* $\mathscr{S}_1 := (\mathsf{resp}_0^{(1)}|| \ \ldots\ ||\mathsf{resp}_0^{(N)})$
  Parse $\mathscr{S}_2$ *as* $\mathscr{S}_2 := (\mathsf{resp}_1^{(1)}|| \ \ldots\ ||\mathsf{resp}_1^{(N)}||\mathsf{C}^{(1)}_{1-b^{(1)}}|| \ \ldots\ ||\mathsf{C}^{(N)}_{1-b^{(N)}})$.
  for j *in* $(1 \ldots N)$ **do**
    $Z^{(j)} := \mathsf{resp}_0^{(j)}$,
    if $b^{(j)} = 0$ **then**
      $\sigma^{(j)} := \mathsf{resp}_1^{(j)}$,
      $\mathsf{C}_0^{(j)} := Com(\sigma^{(j)}, A_{\sigma^{(j)}} Z^{(j)})$
    else
      $\pi\sigma^{(j)=\mathsf{resp}_1^{(j)}}$
      $\mathsf{C}_1^{(j)} = Com(\pi\sigma^{(j)}, Z^{(j)} - c^{(j)} V_{\pi\sigma^{(j)}})$
    end if
    $\mathsf{COM^{(j)}} := (\mathsf{C}_0^{(j)},\ \mathsf{C}_1^{(j)})$
  end for
  $\mathscr{S}'_0 \leftarrow \mathscr{H}_0(\mathsf{COM}^{(1)}|| \ \ldots\ ||\mathsf{COM}^{(N)})$.
  return $\mathscr{S}'_0 = \mathscr{S}_0$.
end procedure

developed to obtain a quantum-secure transform.
In [Unr17], D. UNRUH justifies that the FIAT-SHAMIR transform is secure under certain conditions. Currently, it seems impractical to apply the Unruh transform since the obtained scheme is costly in terms of signature's size. Therefore, we can keep the initial FIAT-SHAMIR as long as there is neither perfect proof nor quantum attack.

C. Performance of the scheme

Our main goal is to find the best parameters which can ensure the minimal size of a signature. We show, in the next sections, that the PKP-based signature scheme provides a signature's size less than the other signature schemes, precisely MQDSS [CHR+18] and Picnic [CDG+17].

Signature size: We said that our signing scheme is constructed from the iterations of the IDS (given in 2). Now, to have the total cost, it is important to define the number of rounds N needed to achieve **EU-CMA** for λ bits of security. By considering the scheme where the fraud's probability is $P_f = \frac{p+1}{2p}$. We require that

$$P_f^N \leq 2^{-\lambda},$$

as an attacker could perform a preimage search to control the challenges. Hence, we get that $N \geq \lambda / \log_2(\frac{p+1}{2p})$.

We begin to present how to compute the complexity in bits. Recall that the signature is composed of R the message-dependent random value, $\mathscr{S}_0$, $\mathscr{S}_1$ and $\mathscr{S}_2$, where $\mathscr{S}_0$ is the hashed value of the commitments of all rounds, $\mathscr{S}_1$ is formed by the first responses, and $\mathscr{S}_2$ is the concatenation of the some commitments and the second responses to the challenges.
For $\mathscr{S}_0$ which is a hashed value, it costs 2λ bits. $\mathscr{S}_1$ depends on the size of Z, so it is in $N \times n \log_2 p$. For $\mathscr{S}_2$, we present next each case:

- **b=0:** The signer reveals one seed sigma.seed (similarly to 2) as a response. It costs the seed size which is presented by λ bits. In addition to the size of the commitment C_1, we have in average:
$$A = \frac{1}{2}\big(Size(\mathsf{C}_1) + Size(\mathsf{resp}_1)\big) = \frac{3}{2}\lambda.$$
- **b=1:** The signer reveals the permutation $\pi\sigma^{(j)}$ as a response resp_1 to the challenge $b^{(j)}$. By adding also the commitment C_0 of size 2λ bits, we have in total:
$$B = \frac{1}{2}\big(2\lambda + \log_2(n!)\big).$$

We have thus the following signature size:

$$\overbrace{2\lambda}^{size\ of\ \mathsf{R}} + \overbrace{2\lambda}^{size\ of\ \mathscr{S}_0} + \underbrace{N\big(n\log_2(p)+A+B\big)}_{size\ of\ \mathscr{S}_1\ and\ \mathscr{S}_2}.$$

How parameters affect performance As we said previously, the DSS is mainly affected by the following set of parameters: (p,n,m). We now explicitly detail the choice of parameters. Recall that firstly the IDS [Sha89] was designed to suit small devices. Thus, A. SHAMIR proposed $p=256$. Nowadays, with the $64-$bit computer architecture, the computations modulo a prime number of 32 or 64 bits are feasible. Thus, we consider that p is of $8,\ 16,\ 32,$ or 64 bits.

A solution of a random instance of PKP is to find a kernel n-vector (V_π) with distinct coordinates in $\mathbb{F}_p$. Hence, the probability to find such vector shouldn't be too small. Also in [Sha89], A. SHAMIR estimated n to be between 32 and 64. Later on, several attacks [BCCG92], [PC93] shows that the choice $n=32$ is not recommended for strong security requirements. So, to find an n-vector with no double in $\mathbb{F}_p$, and by considering the Birthday Paradox, we keep the choice of n around 64, in addition to $n\approx\mathscr{O}\big(\sqrt{p}\big)$.

On the other hand, the probability of an arbitrary vector to be in the kernel of the matrix $A\in\mathscr{M}_{m\times n}$ whose rank is equal to m, is p^{-m}. Moreover, if the n-vector V has no double, the cardinal of its orbit under the possible permutations π is $n!$. Thus, in order to get one solution, we have the following constraint: $n!\approx p^m$.

Hence, following these criteria, we have in total:

$$p\approx\mathscr{O}\big(n^2\big),$$
$$n!\approx n^n\approx p^m.$$

This leads to take $m\approx n\log(n)/log(p)\approx n/2$.

How to choose the security parameter λ. Recall that, the security parameter λ controls the number of iterations $N=\lambda/\log_2(\frac{p+1}{2p})$ performed to achieve a security level needed. It also defines the output of the hash and commitments functions which is in 2λ , in addition to the seeds length.

In general, the hash and commitment functions require collision resistance, preimage resistance, and/or second preimage resistance. Thus, in this article, to reach for example a security of 128 bits, we initiate λ to be exactly of 128 bits. As well for the others security levels (192 and 256).

However, as shown in [GS94], it is always possible to reduce this choice of 256-bit hash values while keeping a security level of 128 bits. Yet, to compare PKP-DSS to the other schemes (as MQDSS) we keep this doubling. Note that, the optimization of [GS94] can be applied to PKP-DSS as well to the other schemes (MQDSS, Picnic,...).

In the following table we present several parameters sets for different levels of security. We define these parameters by considering the formulas given in Section VI-C and the criteria defined above. Furthermore, our parameters raise a secure scheme against all the attacks described in Section IV, mainly, against the most efficient attack: the algorithm of PATARIN-CHAUVAUD [PC93].

Parameters Set	λ	p	n	m	Iterations number N	Best classical attack
PKP-DSS-128	128	977	61	28	129	2^{130} $op.$
PKP-DSS-192	192	1409	87	42	193	2^{198} $op.$
PKP-DSS-256	256	1889	111	55	257	2^{262} $op.$

TABLE I
PKP-DSS PARAMETERS SETS

Next, we compare PKP-DSS to MQDSS [CHR+18] and Picnic [CDG+17]. We consider the public/secret $(\mathsf{pk}/\mathsf{sk})$ keys size and the signature size, for different security levels.

Security level	Parameters Sets	Secret key size (Bytes)	Public key size (Bytes)	Signature size (KBytes)
128	PKP-DSS-128	16	93	16.83
	MQDSS-31-48	16	46	16.15
	Picnic-L1-FS	16	32	33.2
192	PKP-DSS-192	24	139.1	38.02
	MQDSS-31-64	24	64	33.23
	Picnic-L3-FS	24	48	74.9
256	PKP-DSS-256	32	184.4	67.49
	MQDSS-31-88	32	87	60.28
	Picnic-L5-FS	32	64	129.7

TABLE II
COMPARISON OF DIFFERENT SCHEMES

One can conclude that the IDS based on PKP constitutes one of the most efficient schemes.

VII. CONCLUSION

In this article, we discussed some of the most well-known technique for solving PKP. Particularly, we drew attention to the fact that, the Approach of PATARIN-CHAUVAUD [PC93] is the most efficient attack on PKP.

Moreover, our main motivation is the construction of a post-quantum secure cryptosystem. In [Sha89], a Zero-knowledge identification scheme (ZK-IDS) was introduced. A well-known method, namely FIAT-SHAMIR technique [FS86], is used to turn an IDS into a digital signature scheme (DSS).

The authors of [CHR+18], presents a DSS, named MQDSS. It was built from an IDS based on the MQ problem (Multivariate quadratic equations solving problem). Thus, they give several sets of parameters which provide post-quantum security.

As well, Picnic [CDG+17] is designed to be secure against classical and quantum attacks. It was also constructed from a Zero-knowledge identification scheme to match different security levels.

Hence, similarly to the technique used to build these schemes, we have constructed a DSS based on the PKP problem. We utilized the ZK-authentication scheme presented in [Sha89] to deduce the signature scheme. In order to compare this latter to the other schemes, we have tested the most known techniques to solve PKP.

We finally conclude several sets of parameters given in VI-C which provides 128, 192 **and** 256 **bits of classical security. Mainly, we conclude that the DSS based on PKP gives signatures with a size comparable to the ones in MQDSS and smaller than the ones given by Picnic. Consequently, this is what makes from this PKP-DSS a competitive scheme to the other related cryptosystems.**

ACKNOWLEDGMENT

My sincere gratitude and appreciation go to my supervisos, Prof. Jacques Patarin and R.D Jean-Charles Faugère for their valuable comments and suggestions. Our continuous discussion, their immense knowledge and thought provoking questions helped me to get results of better quality. Also, i would especially like to show my deepest thanks for A.P Ludovic Perret for sharing his enormous experiences and illuminating views on all the issues related to my Phd. His professional and aspiring guidance pushed me to widen my work from different perspectives.

I take this opportunity to express thanks, to Mr. Gilles Macario-Rat for also helping me to conduct this PhD research.

I am also grateful to the University of Pierre and Marie Curie (UPMC), and in particular, to the *LIP6* laboratory and the *PolSys* team for providing me with all the essential facilities being required for my work.

REFERENCES

[BBBV97] Charles H Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM journal on Computing*, 26(5):1510–1523, 1997.

[BCCG92] Thierry Baritaud, Mireille Campana, Pascal Chauvaud, and Henri Gilbert. On the security of the permuted kernel identification scheme. In *Annual International Cryptology Conference*, pages 305–311. Springer, 1992.

[CDG+17] Melissa Chase, David Derler, Steven Goldfeder, Claudio Orlandi, Sebastian Ramacher, Christian Rechberger, Daniel Slamanig, and Greg Zaverucha. Post-quantum zero-knowledge and signatures from symmetric-key primitives. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1825–1842. ACM, 2017.

[CHR+18] Ming-Shing Chen, Andreas Hülsing, Joost Rijneveld, Simona Samardjiska, and Peter Schwabe. Mqdss specifications, 2018.

[Dam99] Ivan Damgård. *Commitment Schemes and Zero-Knowledge Protocols*, pages 63–86. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999.

[FS86] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *Conference on the Theory and Application of Cryptographic Techniques*, pages 186–194. Springer, 1986.

[Geo92] Jean Georgiades. Some remarks on the security of the identification scheme based on permuted kernels. *Journal of Cryptology*, 5(2):133–137, 1992.

[GJ79] Michael R Garey and David S Johnson. Computers and intractability: a guide to np-completeness, 1979.

[GS94] Marc Girault and Jacques Stern. On the length of cryptographic hash-values used in identification schemes. In *Annual International Cryptology Conference*, pages 202–215. Springer, 1994.

[HNO+09] Iftach Haitner, Minh-Huyen Nguyen, Shien Jin Ong, Omer Reingold, and Salil Vadhan. Statistically hiding commitments and statistical zero-knowledge arguments from any one-way function. *SIAM Journal on Computing*, 39(3):1153–1218, 2009.

[JJ01] Éliane Jaulmes and Antoine Joux. Cryptanalysis of pkp: a new approach. In *International Workshop on Public Key Cryptography*, pages 165–172. Springer, 2001.

[JL01] Antoine Joux and Reynald Lercier. "chinese & match", an alternative to atkin's "match and sort" method used in the sea algorithm. *Mathematics of computation*, 70(234):827–836, 2001.

[LP11] Rodolphe Lampe and Jacques Patarin. Analysis of some natural variants of the pkp algorithm. *IACR Cryptology ePrint Archive*, 2011:686, 2011.

[NIS] Security strength categories. https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf.

[PC93] Jaques Patarin and Pascal Chauvaud. Improved algorithms for the permuted kernel problem. In *Annual International Cryptology Conference*, pages 391–402. Springer, 1993.

[Sha89] Adi Shamir. An efficient identification scheme based on permuted kernels. In *Conference on the Theory and Application of Cryptology*, pages 606–609. Springer, 1989.

[Unr15] Dominique Unruh. Non-interactive zero-knowledge proofs in the quantum random oracle model. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 755–784. Springer, 2015.

[Unr17] Dominique Unruh. Post-quantum security of fiat-shamir. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 65–95. Springer, 2017.