Cryptanalysis of PASS II and MiniPass

Bok-Min Goi[1], Jintai Ding[2], and M.U. Siddiqi[1,⋆]

[1] Centre for Cryptography and Information Security (CCIS),
Faculty of Engineering, Multimedia University,
63100 Cyberjaya, Malaysia
bmgoi@mmu.edu.my

[2] Department of Mathematical Sciences,
University of Cincinnati, Cincinnati,
OH 45221-0025 USA
ding@math.uc.edu

Abstract. In ACISP '00, Wu *et al.* proposed attacks to break the Polynomial Authentication and Signature Scheme (PASS), in particular, they are able to generate valid authentication transcripts and digital signatures without knowing the private key and any previous transcripts/signatures. They showed that PASS can be broken with around $2^{38.3}$ trials. In this paper, we analyze the security of the improved versions of PASS; viz. PASS II and MiniPASS, and extend the Wu *et al.*'s attacks to PASS II and MiniPASS to break them. Furthermore, we discuss why and how these schemes are broken from the view point of the structure of cryptosystems and point out the fundamental weakness behind.

Keywords: Authentication scheme, digital signature scheme, cryptanalysis, NTRU, partial polynomial evaluation.

1 Introduction

For electronic communications and commerce in a common networked and open environment, security issues are always the main concern. In this context, public key authentication and digital signature schemes that provide authentication and non-repudiation services to communicating parties have been of steadily increasing interest in cryptographic studies. They are not only need to be secure, but also have to be fast and can be implemented on low power computing devices, i.e. low-cost smart cards and RFID devices. Since year 1996, researchers from NTRU Cryptosystem Inc. have proposed a group of fast public key cryptosystem based on the hard problems of partial evaluation of constrained polynomial over polynomial rings. These comprise of NTRU public key encryption algorithm [3], NTRUSign digital signature scheme [1], Polynomial Authentication and Signature Scheme, PASS [2], and its variant PASS II [5] and MiniPASS [4]. The hard problem underlying this group of cryptosystem can be related to short vectors in a lattice due to properties of short polynomials used in the system.

⋆ The first author acknowledges the Malaysia IRPA grant (04-99-01-00003-EAR).

S. Qing et al. (Eds.): ICICS 2005, LNCS 3783, pp. 159–168, 2005.
© Springer-Verlag Berlin Heidelberg 2005

In ACISP '00, Wu *et al.* presented two attacks on PASS [6]. In particular, they are able to generate valid authentication transcripts and digital signatures without knowing the private key or previous transcripts / signatures. Though their first attack can be easily prevented with some proper parameter settings, PASS can still be broken with around $2^{38.3}$ trials under their second attack.

In this paper, we further analyze the security of the improved versions of PASS; viz. PASS II and MiniPASS, and extend the Wu *et al.*'s attacks to PASS II and MiniPASS, and show how to break them efficiently as well. Furthermore, we discuss how and why these schemes are broken from the point view of the structure of cryptosystems and point out the fundamental weakness behind which allows these attacks, namely the participation of a verifier in the process setting up the challenge. Therefore, though we believe that the concept behind the construction of PASS, namely the hard problems of partial evaluation of constrained polynomials over polynomial rings, is still correct and sound, any new system based on the same idea as that of PASS needs to overcome such a fundamental weakness in order to work securely, in particular, in terms of resisting the type of attack like that of the Wu *et al.*'s attacks.

The paper is organized as following. In the next section, we will outline the authentications systems PASS, PASS II and MiniPASS [2,5,4]. We then briefly introduce the basic idea of the Wu *et al.*'s attacks [6] in Section 3. In Section 4, we will present the details of our attack to break the PASS II and MiniPASS, which is an extension of the idea of Wu *et al.*. Then, we elaborate the structure analysis of PASS cryptosystems in Section 5. Finally, we conclude in Section 6.

2 An Overview of PASS and PASS II

2.1 Preliminary and Notations

The ring of truncated polynomials is defined as: $R = (Z/qZ)[X]/(X^N - 1)$, where q and N are co-relative prime integer. Note that all arithmetic operations are in R in this paper. The resultant coefficients are reduced modulo q and exponents are reduced modulo N. For the proposed schemes in [2, 5, 4], q and N were chosen be a prime number (i.e., 769) and a divisor of $(q-1)$ (i.e., 768), respectively. A element $g \in R$ is denoted as a polynomial with degree of $(N-1)$ and its coefficients $g_i \in Z/qZ$, for $i = 0, 1, ..., (N-1)$, as $g(X) = \sum_{i=0}^{N-1} g_i X^i = g_0 + g_1 X + g_2 X^2 + ... + g_{N-1} X^{N-1}$.

For each element $\alpha \in Z/qZ$, $g(\alpha)$ means that substituting the variable X in the polynomial g with the value α and the result is reduced modulo q. For multiplication of two polynomials in R, since the exponents of the product are reduced modulo N, thus it is a cyclic convolution multiplication. For example, given $f, g \in R$, the product $h = fg$ in R will be $h(X) = \sum_{i=0}^{N-1} f_i X^i \sum_{j=0}^{N-1} g_j X^j = \sum_{k=0}^{N-1} h_k X^k$, where $h_k = \sum_{i+j=k \bmod N} f_i g_j$. Note that if $\alpha^N = 1 \mod q$, then $h(\alpha) = f(\alpha) g(\alpha)$. Informally, a short polynomial g is a polynomial with small norm value $\|g\|_2 = \sqrt{\sum_{i=0}^{N-1} g_i^2}$. For example, it may contain many zero coefficients and few coefficients with small value (i.e., -1 or +1). A polynomial g in

Table 1. The notations

P	The prover
V	The verifier
c_o	The 80-bit challenge
M	The message to be signed
S	The set of t distinct non-zero elements of $\alpha \in Z/qZ$, where $\alpha^N = 1 \bmod q$ and $\alpha^{-1} \in S$. Note that $t \approx \frac{q}{2}$
f_1, f_2	The private key and its corresponding public key is $(f_1(\alpha), f_2(\alpha))$ for $\alpha \in S$. Note that only one private polynomial f for PASS II
g_1, g_2	The commitment polynomials with corresponding commitment values $(g_1(\alpha), g_2(\alpha))$ for $\alpha \in S$
d_i	The polynomial i which contains d_i coefficients equal to each of 1 and -1, and the rest equal to 0, for $i = f, g$ and c. For example, the suggested parameters for PASS are $d_f = d_g = 256, d_c = 1$
L_i	The public special subset of R, for $i = f, g, c$ and h. Note that h is computed based on the private key, commitment polynomials and the outputs of $Hash$. In more detail, L_f, L_g and L_c are those special subsets whose element polynomials contain d_f, d_g and d_c parameters, respectively; whereas, $L_h = h \in R : \|h\|_2 \leq \gamma q$
$Hash(\cdot)$	The special hash function which hashes c_o or M with the commitments to produce some polynomials in L_c

R is called *moderately* short if its norm is smaller than a constant times q, such that $\|g\|_2 \leq \gamma q$, where the value of γ is determined by the particular application via experiment.

For ease of explanation, we use the notations similar to those in [2, 6, 5], as shown in Table 1.

2.2 PASS

The prover P randomly chooses two polynomial (f_1, f_2) from L_f as his private key. Then he generates his public key as $(f_1(\alpha), f_2(\alpha))$ for $\alpha \in S$. The private key is well protected due to the fact that recovering a short polynomial in R by providing only a certain subset values of the polynomial is a hard problem. Furthermore, it is also difficult to find another two short polynomials (f_1', f_2'), which for $\alpha \in S$, must satisfy the conditions:

$$f_1'(\alpha) = f_1(\alpha) \quad \text{and} \quad f_2'(\alpha) = f_2(\alpha). \tag{1}$$

We refer readers to [2] for more security analysis of PASS. For simplicity, we depict the PASS authentication scheme and the PASS digital signature scheme in Fig. 1 and Fig. 2, respectively.

2.3 PASS II and MiniPASS

In [5], Hoffstein and Silverman have proposed another improved version of PASS, called as PASS II. PASS II only involves single polynomial f in key pair creation

Prover, P		**Verifier,** V
P's private key: f_1, f_2		P's public key:
$g_1, g_2 \in_R L_g$		$(f_1(\alpha), f_2(\alpha))_{\alpha \in S}$
For $\alpha \in S$,		
compute the commitment:		
$(g_1(\alpha), g_2(\alpha))$		
	$\xrightarrow{(g_1(\alpha), g_2(\alpha))_{\alpha \in S}}$	
		Randomly select c_0
	$\xleftarrow{c_0}$	
$Hash(g_1(\alpha), g_2(\alpha), c_0)$		$Hash(g_1(\alpha), g_2(\alpha), c_0)$
$\rightarrow c_1, c_2, c_3, c_4 \in L_c$		$\rightarrow c_1, c_2, c_3, c_4 \in L_c$
Compute:		
$h = c_1 f_1 g_1 + c_2 f_1 g_2$		
$+c_3 f_2 g_1 + c_4 f_2 g_2$		
	$\xrightarrow{h}$	
		Verify:
		$h \in L_h$ and for $\alpha \in S$,
		$h(\alpha) = c_1(\alpha) f_1(\alpha) g_1(\alpha) + c_2(\alpha) f_1(\alpha) g_2(\alpha)$
		$+c_3(\alpha) f_2(\alpha) g_1(\alpha) + c_4(\alpha) f_2(\alpha) g_2(\alpha)$

Fig. 1. PASS authentication scheme

Prover, P		**Verifier,** V
P's private key: f_1, f_2		P's public key:
$g_1, g_2 \in_R L_g$		$(f_1(\alpha), f_2(\alpha))_{\alpha \in S}$
For $\alpha \in S$, compute		
$(g_1(\alpha), g_2(\alpha))$ and		
$Hash(g_1(\alpha), g_2(\alpha), M)$		
$\rightarrow c_1, c_2, c_3, c_4 \in L_c$		
Compute:		
$h = c_1 f_1 g_1 + c_2 f_1 g_2$		
$+c_3 f_2 g_1 + c_4 f_2 g_2$		
	$\xrightarrow{(g_1(\alpha), g_2(\alpha))_{\alpha \in S}, h, M}$	$Hash(g_1(\alpha), g_2(\alpha), M)$
		$\rightarrow c_1, c_2, c_3, c_4 \in L_c$
		Verify:
		$h \in L_h$ and for $\alpha \in S$,
		$h(\alpha) = c_1(\alpha) f_1(\alpha) g_1(\alpha) + c_2(\alpha) f_1(\alpha) g_2(\alpha)$
		$+c_3(\alpha) f_2(\alpha) g_1(\alpha) + c_4(\alpha) f_2(\alpha) g_2(\alpha)$

Fig. 2. PASS signature scheme

phase and only one set of values of $(g_1(\alpha))_{\alpha \in S}$ required for the verifier V. Therefore, it can further reduce the computational complexity and communication requirements. In CHES '00 [4], based on PASS II, the same authors presented a scheme which is called as MiniPASS and described how it can be implemented in highly constrained devices.

Prover, P		**Verifier,** V
P's private key: f		P's public key:
$g_1, g_2 \in_R L_g$		$(f(\alpha))_{\alpha \in S}$
For $\alpha \in S$, compute:		
$g_1(\alpha)$ and then,		
$Hash(g_1(\alpha), M)$		
$\rightarrow c_1, c_2 \in L_c$,		
where $c_1(\alpha) \neq 0$,		
for $2 \leq \alpha \leq q-2$ and $\alpha \notin S$.		
Compute:		
$h = (f + c_1 g_1 + c_2 g_2) g_2$		
	$\xrightarrow{(g_1(\alpha))_{\alpha \in S}, h, M}$	$Hash(g_1(\alpha), M)$
		$\rightarrow c_1, c_2 \in L_c$
		where $c_1(\alpha) \neq 0$
		for $2 \leq \alpha \leq q-2$ and $\alpha \notin S$
		Verify:
		$h \in L_h$ and for $\alpha \in S$,
		$(f(\alpha) + c_1(\alpha) g_1(\alpha))^2 + 4c_2(\alpha) h(\alpha)$
		quadratic residue mod q

Fig. 3. PASS II signature scheme

Table 2. Comparison of PASS and PASS II signature scheme

Description	**Original PASS [2]**	**PASS II [5]**
Creating key pair	Two private polynomials f_1 and f_2 with the corresponding public values $(f_1(\alpha), f_2(\alpha))_{\alpha \in S}$.	Only one private polynomials f with the corresponding public values $(f(\alpha))_{\alpha \in S}$.
Generating commitment polynomials	Two commitment polynomials (g_1, g_2). Both $(g_1(\alpha), g_2(\alpha))$ have be sent to the verifier.	Two commitment polynomials (g_1, g_2). But only $g_1(\alpha)$ is required to be sent to the verifier.
Computing h	$h = c_1 f_1 g_1 + c_2 f_1 g_2 + c_3 f_2 g_1 + c_4 f_2 g_2$	$h = (f + c_1 g_1 + c_2 g_2) g_2$
Verification process	Check $h \in L_h$ and the values of $h(\alpha)$ for $\alpha \in S$	Check $h \in L_h$ and $(f(\alpha) + c_1(\alpha) g_1(\alpha))^2 + 4c_2(\alpha) h(\alpha) =$ quadratic residue mod q, for $\alpha \in S$.
Parameter settings	$d_f = d_g \approx \frac{q}{3}, d_c = 1, \gamma = 2.2$	$d_f \approx \frac{q}{3}, d_g \approx \frac{q}{6}, d_c = 2, \gamma = 1.8$.
Security level†	With $q = 769$, PASS is more secure than RSA 1024; With $q = 1153$, PASS is more secure than RSA 2048.	With $q = 769$, PASS II is more secure than RSA 512; With $q = 929$, PASS is more secure than RSA 1024.

† : as claimed by the original inventers in [2] and [5].

The PASS II signature scheme is described in Fig. 3. We omit the PASS II authentication scheme because it can be constructed in a similar way as PASS authentication scheme.

We conclude this section by providing the comparison between the original PASS and PASS II schemes in Table 2. (Note that this comparison could be applied directly to the authentication schemes as well.)

3 Two Attacks Due to Wu *et al.*

In [6], Wu *et al.* proposed two attacks on the PASS system. For ease of explanation and also lack of better names, we simply denote the first attack proposed in Section 3 of [6] as Attack 1 and the second attack proposed in Section 4 of [6] as Attack 2, respectively.

3.1 Wu *et al.*'s Attack 1

Wu *et al.* discovered that in order to break PASS which amounts to satisfying the expression:

$$\begin{aligned} h(\alpha) = c_1(\alpha)f_1(\alpha)g_1(\alpha) + c_2(\alpha)f_1(\alpha)g_2(\alpha) + \\ +c_3(\alpha)f_2(\alpha)g_1(\alpha) + c_4(\alpha)f_2(\alpha)g_2(\alpha) \end{aligned}$$

for $\alpha \in S$ in the verification process, is not required to find short polynomials (f_1', f_2') satisfying the condition in Eq. (1). However, they claimed that as long as $g_1(\alpha) = g_2(\alpha) = 0$ for certain $\alpha \in S$, then $f_1'(\alpha)$ and $f_2'(\alpha)$ can be of any values for other values of α. They proved that there are at most p non-zero elements in Z/Z_q satisfying $g(\alpha) \neq 0$. Note that p is a divisor of N and the coefficients of polynomial g from $\{-1, 0, +1\}$ are with period p. Based on this, Wu *et al.* came out with Attack 1 where an attacker can forge the authentication transcript/signature independent of the sizes of N and t. An attacker, with such small amount of computation, can fool B into thinking he is a legitimate counterpart A, even without A's private key and previous communicated transcripts.

Observations on Attack 1. Here, we correct a typo in the expression in the proof of Theorem 1 [6], where a polynomial g with period p (obviously, p must be one of the factor of N) should be denoted as:

$$g(X) = (g_0 + g_1X + g_2X^2 + \cdots + g_{p-1}X^{p-1})(1 + X^p + X^{2p} + \cdots + X^{N-p}),$$

but not as:

$$g(X) = (g_0 + g_1X + g_2X^2 + \cdots + g_{p-1}X^{p-1})(1 + X^p + X^{2p} + \cdots + X^{N/p}).$$

We further remark that, in [6], in order to countermeasure Attack 1, Wu *et al.* suggested N to be chosen as a prime or with no small factor. However, PASS only works in the special case where $\alpha^N = 1 \mod q$ in ring R. This is to ensure homomorphism mapping. More precisely, N must be a divisor of $(q-1)$ and according to the fact of Fermat's Little Theorem [2], for all non-zero element, we can well define the homomorphic mapping from R to Z/qZ as: $g(X) \rightarrow g(\alpha^{(q-1)/N})$. Hence, N has to be determined carefully.

3.2 Wu *et al.*'s Attack 2

Wu *et al.* proposed Attack 2 by exploiting the fact that the space of L_c, $|L_c|$ is small (as $d_c = 1$). Attack 2 on the PASS signature scheme is described as:

1. Randomly select $r_1, r_2 \in R$ such that $r_1 c, r_2 c \in L_c$ for $c \in L_c$.
2. For $\alpha \in S$, compute the two sets (β_1, β_2) with no zero element, such that $\beta_1 = f_1(\alpha) + r_1(\alpha) f_2(\alpha)$ and $\beta_2 = f_1(\alpha) + r_2(\alpha) f_2(\alpha)$. Note that $\beta_i = \{\beta_{i1}, \beta_{i2}, \cdots, \beta_{it}\}$ where $i = \{1, 2\}$.
3. Compute two polynomial (f_1', f_2') such that $f_1'(\alpha) = \beta_1$ and $f_2'(\alpha) = \beta_2$.
4. Arbitrarily choose $h_1, h_2 \in R$ and compute the polynomials (g_1, g_2) which satisfy $h_1(\alpha) = f_1'(\alpha) g_1(\alpha)$ and $h_2(\alpha) = f_2'(\alpha) g_2(\alpha)$ for $\alpha \in S$. Note that (g_1, g_2) are not required to be short polynomial and fulfill d_g requirement. Wu *et al.* showed that these two polynomials can be computed easily in Theorem 3 of [6].
5. $Hash(g_1(\alpha), g_2(\alpha), M)$ and obtain $c_1, c_2, c_3, c_4 \in L_c$.
6. Check whether $c_3 = r_1 c_1$ and $c_4 = r_2 c_2$. If yes, set $h' = c_1 h_1 + c_2 h_2$ and obtain a valid signature of message M. Otherwise, repeat the attack by going back to Step 3.

Obviously, the obtained h' is equivalent to h. Attack 2 will succeed with probability $\frac{1}{|L_c|^2}$. More precisely, with the proposed parameters in [2]: $N = 768$ and $d_c = 1$, the PASS can only achieve the security level of 38.3 bits. Therefore, the PASS is not secure.

4 Our Attacks on PASS II and MiniPASS

In this section, we extend the Wu *et al.*'s attacks on PASS II and MiniPASS. We prove that they face the same problem as PASS. Namely, both with the proposed parameters cannot achieve 80-bit standard security requirement. Hence, PASS II and MiniPASS are insecure and not sound.

4.1 Extended Attack 1

Our first extended attack on PASS II works as follows:

1. Calculate the desired polynomials g_2 by setting appropriate period p.
2. Determine $S_1 \subset S$ such that for $\alpha \in S_1$, $g_2(\alpha) = 0$. Then, fix $S_2 = S - S_1$, such that for $\alpha \in S_2$, $g_2(\alpha) \neq 0$.
3. Compute the polynomial f', such that $f'(\alpha) = f(\alpha)$ for $\alpha \in S_2$. For $\alpha \in S_1$, due to Step 2, $h(\alpha)$ is always equal to zero regardless of the values of $f'(\alpha)$.
4. With the three short obtained polynomials (g_1, g_2, f'), where g_1 can be set arbitrarily, a valid forged authentication transcript satisfying $h \in L_h$ and for $\alpha \in S$, $(f(\alpha) + c_1(\alpha) g_1(\alpha))^2 + 4 c_2(\alpha) h(\alpha) =$ quadratic residue (mod q), can be produced.

The Step 3 above, in determining the polynomial f' such that $f'(\alpha) = f(\alpha)$ for $\alpha \in S_2$, is the most costly one. On average, the computation complexity is about $q^{|S_2|}$, where $|S_2|$ is the space of S_2 . However, due to the careful selection of g_2 (e.g., with the period $p = 6$), $|S_2|$ is quite small (e.g., $|S_2| \approx \frac{p}{2}$), thus computing f' turn to be an easy task. In particular, for $|S_2| = 3$, even with brute-force to randomly search the coefficients of f', we only require around $q^{|S_2|} = 769^3 = 2^{28.8}$ trials to forge a valid authentication transcripts/signature in PASS II with the proposed parameters, $q = 769$ in [5]. Therefore, PASS II can only provide 28.8-bit security level under this attack. Note that PASS II is much more vulnerable under our extended attack as compared to Attack 1 on PASS, because only one private polynomial f' needs to be determined.

4.2 Extended Attack 2

In [5], Hoffstein and Silverman have increased the space for L_c by setting $d_c = 2$. (In the original PASS, $d_c = 1$.) This is mainly due to only two challenge polynomials (c_1, c_2) are involved in this improved scheme, but not to counter against Attack 2. They claimed that with the proposed parameters, $N = 768$, then $|L_c| = 2^{36}$. Hence, the space of challenge should be the space of pairs (c_1, c_2) of elements of $|L_c|^2$ and equal to 2^{72}. However, in this subsection, we show that it is not true.

In the PASS II scheme, the authentication transcript/signature is expressed as:

$$\begin{aligned} h &= (f + c_1 g_1 + c_2 g_2) g_2 \\ &= f g_2 + (c_1 g_1 + c_2 g_2) g_2 \\ &= f g_2 + c_1 (g_1 + r g_2) g_2 \\ &= h_1 + c_1 h_2 \end{aligned}$$

where $c_2 = rc_1$, $h_1 = fg_2$ and $h_2 = (g_1 + rg_2)g_2$.

Our second extended attack on the PASS II signature scheme works as follows:

1. Select $r \in_R R$ such that $rc \in L_c$ for $c \in L_c$ (for simplicity, set $r = 1$).
2. Arbitrarily choose $h_2, g_2 \in_R R$. Then, for $\alpha \in S$, compute the polynomial g_1 satisfying $h_2(\alpha) = (g_1(\alpha) + r(\alpha)g_2(\alpha))g_2(\alpha)$. Note that g_1 is not required to be short and fulfill d_g requirement .
3. $Hash(g_1(\alpha), M)$ and obtain $c_1, c_2 \in L_c$.
4. Check whether $c_2 = rc_1$. If yes, compute the polynomials h_1, which is not required to be short, satisfying $h_1(\alpha) = f(\alpha)g_2(\alpha)$ and set $h = h_1 + c_1 h_2$, then generate a valid signature of message M which is $(g_1(\alpha), h)_{\alpha \in S}$. Otherwise, repeat the attack by going back to Step 2.

Surprisingly, we found out that our proposed extended attack on PASS II works even more efficient than the original Attack 2 on PASS, in terms of number of steps and computational complexity. In particular, there is no need to compute the two sets (β_1, β_2) and the two polynomial (f_1', f_2') as in the case of PASS, but only compute one polynomial g_1 that satisfies $h_2(\alpha) = (g_1(\alpha) + r(\alpha)g_2(\alpha))g_2(\alpha)$. Furthermore, Attack 2 has to be improved in order to work on PASS for the case where the two sets (β_1, β_2) contain zero element. This will affect the success rate of this attack.

However, in our attack, the attacker just need to check whether $c_2 = rc_1$ for every single trial. The success rate will be $\frac{1}{|L_c|}$, but not $\frac{1}{|L_c|^2}$ as claimed in [5]. More precisely, with the suggested parameters in [5], viz. $N = 768$ and $d_c = 2$, the space of L_c, $|L_c| = \frac{N!}{(N-2d_c)!(d_c!)^2} = 2^{36.3}$. Therefore, PASS II can only offer the security level of 36-bit, and is insecure and not sound.

SUMMARY: Even with proper parameter settings – on choosing good N – the PASS and PASS II schemes are still vulnerable under Attack 2. This is mainly due to the *decomposition property* of h.

5 Structure Analysis

One of the original intention of our work is to find ways to improve the PASS cryptosystem and try to make it work, because we think that the basic concept behind the construction of the PASS, which is based on the hard problems of partial evaluation of constrained polynomials over polynomial rings, is still sound. To do so, we start from further understanding why and how the PASS systems are broken from the viewpoint of the structure of cryptosystems.

The key point of the Attack 2 is that it successfully transforms the difficult problem of finding the private keys (f_1, f_2) given $f_1(\alpha)$ and$f_2(\alpha)$ for $\alpha \in S$ to another easier task – finding (g_1, g_2) – which does not anymore have to be short polynomials (binary or trinary polynomials with many zero coefficients) and fulfill certain requirements on the degree. For example, as stated in [4], by using Discrete Fourier Transform (DFT) method, every coefficients of a polynomial G can be determined providing all the values of G via the well-known formula of DFT. In this case, the Attack 2 shows the system's security is not really based on the hard problems of partial evaluation of constrained polynomials over polynomial rings, but rather something else. If one looks deeper, one should realize that the story does not just stop here. The real reason behind is that g_1 and g_2 are actually provided by the prover, which therefore allows the attacker to choose the g_1 and g_2. Namely in a PASS authentication system, **a prover actually participates in the process in setting up the challenge**. This is very much like leaking partially the secret automatically to the attacker. Such a weakness is the real cause why Attack 2 worked from the structure point of view. Therefore, this, we believe, is a fundamental structure flaw from the point view of designing a secure cryptosystem, because it gives an attacker automatically an edge in attacking the system from the very beginning. This teaches us a good lesson in designing authentication schemes, namely one should not allow a prover to participate in setting up the challenge.

6 Concluding Remarks

In this paper, we have comprehensively analyzed the security of PASS II and MiniPASS. We have proposed two extended attacks due to Wu *et al.* [6] and showed how they work on PASS II and MiniPASS in detail. We have further pointed out the main reason for PASS and its variant are broken is due to some flaws in the structure design of the cryptosystems. However, the concept behind the construction of the PASS, based on the hard problems of partial evaluation of constrained polynomials over polynomial rings, is correct and secure.

At the very beginning, we had some suggestions to improve the system. For example: (1) to ensure that the verifier always knows the prover's commitments prior to receiving any new signature, so that the attacker is unable to control the polynomials (g_1, g_2) and perform Attack 2; (2) to modify the expression of h so that it won't face the decomposition problems. However, we realize these suggestions do not fundamentally eliminate the flaw of the system. As of now, we have not been able to come up with any good solution, which, we believe, is a very interesting and challenging question.

Acknowledgement

We would like to thank anonymous referees for their constructive and detailed comments that have greatly improved this paper.

References

1. J. Hoffstein, N. Graham, J. Pipher, J. Silverman and W. Whyte. NTRUSign: Digital Signatures Using the NTRU Lattice. In *Proceeding of CT-RSA '03*, LNCS, vol. 2612, Springer-Verlag, pp.122-140, 2003.
2. J. Hoffstein, D. Lieman, J. Silverman. Polynomial Rings and Efficient Public Key Authentication. In *Proceeding of CrypTEC '99*, City University of Hong Kong Press, pp. 7-19, 1999.
3. J. Hoffstein, J. Pipher and J. Silverman. NTRU: A Ring-Based Public Key Cryptosystem. In *Proceeding of ANTS III*, LNCS, vol. 1423, Springer-Verlag, pp. 267-288, 1998.
4. J. Hoffstein and J. Silverman. MiniPASS: Authentication and Digital Signatures in a Constrained Environment. In *Proceeding of CHES '00*, LNCS, vol. 1965, Springer-Verlag, pp. 328-339, 2000.
5. J. Hoffstein and J. Silverman. Polynomial Rings and Efficient Public Key Authentication II. Available at www.ntru.com.
6. Hongjun Wu, Feng Bao, Dingfeng Ye, Robert Deng. Cryptanalysis of Polynomial Authentication and Signature Scheme. In *Proceeding of ACISP '00*, LNCS, vol. 1841. Springer-Verlag, pp. 278-288, 2000.