From: Ward Beullens <ward@beullens.com>
Sent: Wednesday, May 02, 2018 4:42 PM
To: pqc-comments
Cc: pqc-forum@list.nist.gov
Subject: OFFICIAL COMMENT: HiMQ-3

Dear all,

TL;DR: The security proof of HiMQ-3 (Theorem 4) is flawed.

The HiMQ-3 submission document claims that the HiMQ-3 signature scheme is EUF-CMA secure provided that it is hard to find a solution for a system of quadratic equations in the HiMQ-3 family. In other words, the claim is that if the scheme is UF-KOA secure (univeral forgery under key-only attack), then the scheme is also EUF-CMA secure.

The proof of this claim is to be found in [1] (Theorem 4.1), where the same claim is made for the ELSA signature scheme. The proof is very similar to the classic proof of [2] for the security of a hash-and-sign signature scheme based on a trapdoor permutation. However, the trapdoor function used by the HiMQ-3 scheme is not a permutation, and this causes the proof to fail.

The proof programs a random oracle by sampling random x, and returning P(x), where P is the public key. In the trapdoor permutation setting this is a valid approach, because there is no way to distinguish (x,P(x)) from (P^{-1}(y),y), for x and y uniformly distributed variables on the domain and codomain of P respectively. When P is no longer a permutation (as is the case for HiMQ-3 and ELSA) this might no longer be the case. (In fact, P^{-1}(y) is not even uniquely defined) This means that the adversary is no longer guaranteed to function correctly in the simulated environment and that the proof fails.

Kind regards,
Ward

[1] Shim, Kyung-Ah, Cheol-Min Park, and Namhun Koo. "An Existential Unforgeable Signature Scheme Based on Multivariate Quadratic Equations." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Cham, 2017.

[2] Bellare, Mihir, and Phillip Rogaway. "Random oracles are practical: A paradigm for designing efficient protocols." Proceedings of the 1st ACM conference on Computer and communications security. ACM, 1993.

From: Ryo Fujita <rfujita140411@gmail.com>
Sent: Wednesday, July 18, 2018 1:58 AM
To: pqc-forum
Cc: pqc-comments
Subject: Re: OFFICIAL COMMENT: HiMQ-3

EUF-CMA security on multivariate signature schemes was discussed in [3]. There, it is described how to modify the signature scheme to achieve EUF-CMA in the random oracle model. Likewise, it seems that HiMQ-3 may also achieve EUF-CMA.

Kind regards,
Ryo

[3] Sakumoto K., Shirai T., Hiwatari H. (2011) On Provable Security of UOV and HFE Signature Schemes against Chosen-Message
Attack. In: Yang BY. (eds) Post-Quantum Cryptography. PQCrypto 2011. Lecture Notes in Computer Science, vol 7071. Springer, Berlin, Heidelberg

2018年5月3日木曜日 5時42分28秒 UTC+9 Ward Beullens:

> Dear all,
>
> TL;DR: The security proof of HiMQ-3 (Theorem 4) is flawed.
>
> The HiMQ-3 submission document claims that the HiMQ-3 signature scheme is EUF-CMA secure provided that it is hard to find a solution for a system of quadratic equations in the HiMQ-3 family. In other words, the claim is that if the scheme is UF-KOA secure (univeral forgery under key-only attack), then the scheme is also EUF-CMA secure.
>
> The proof of this claim is to be found in [1] (Theorem 4.1), where the same claim is made for the ELSA signature scheme. The proof is very similar to the classic proof of [2] for the security of a hash-and-sign signature scheme based on a trapdoor permutation. However, the trapdoor function used by the HiMQ-3 scheme is not a permutation, and this causes the proof to fail.
>
> The proof programs a random oracle by sampling random x, and returning P(x), where P is the public key. In the trapdoor permutation setting this is a valid approach, because there is no way to distinguish (x,P(x)) from (P^{-1}(y),y), for x and y uniformly distributed variables on the domain and codomain of P respectively. When P is no longer a permutation (as is the case for HiMQ-3 and ELSA) this might no longer be the case. (In fact, P^{-1}(y) is not even uniquely defined) This means that the adversary is no longer guaranteed to function correctly in the simulated environment and that the proof fails.
>
> Kind regards,
> Ward
>
> [1] Shim, Kyung-Ah, Cheol-Min Park, and Namhun Koo. "An Existential Unforgeable Signature Scheme Based on Multivariate Quadratic Equations." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Cham, 2017.
>
> [2] Bellare, Mihir, and Phillip Rogaway. "Random oracles are practical: A paradigm for designing efficient protocols." Proceedings of the 1st ACM conference on Computer and communications security. ACM, 1993.

From: LOUISY Anne-Elise <anne-elise.louisy@thalesgroup.com>
Sent: Tuesday, August 07, 2018 10:38 AM
To: pqc-comments
Cc: pqc-forum@list.nist.gov
Subject: OFFICIAL COMMENT: HiMQ-3

Dear HiMQ-3 team,

There seem to be contradictions between the description of the third layer of the central map and the matrices presented in the analysis of known attacks (figure 2 of the supporting documentation).

In the description, it is written that the polynomials of the third layer of the central map are of the form:

f(x)= sum_i,j beta_i,j x_i x_j + theta(x) + theta'(x) + epsilon x_(o1+o2+k)

where the i,j in the sum are between v +1 and v1.

For the definition of theta and theta', it is written that the coefficients are such that symmetric matrix associated to the quadratic part of f has full rank, which implies that the quadratic part of f involves all n variables.

However, not all variables appear in f. For the k-th polynomial of the third layer, x_(v+1),…,x_(v1) appear in the sum, x_(v1+1),..,x_v2 appear in theta (assuming the modulo is o2 and that 1 is added to the result) and x_(v2+1),…, x_n appear in theta' (assuming again that 1 is added to the subscript). All the other variables, save for x_k that appears in theta and theta', are not in f.

Moreover, with the definition of the third layer given in the description of the central map, we get matrices with non-zero coefficients only in the square corresponding to the sum and on line k and column k resulting from theta and theta' (x_k appear in the products x_k x_i for several different i between v1+1 and n).

(the theoretical secret key size provided also suggests that they are more coefficients that the one given in the description),

Sincerely,

A-E. Louisy,

Student in cryptography at Versailles University

From: 심경아 <shimkah221@gmail.com>
Sent: Thursday, September 13, 2018 5:13 AM
To: pqc-comments
Subject: OFFICIAL COMMENT: HiMQ-3

Dear A-E. Louisy,

Thank you for your comments.

There is a typo. The current formulas

\Theta_i(x)=\sum_{j=1}^{v_1} \gamma_{i,j}x_i x_{v_1+(i+j-1) (mod o_3),

\Theta_i'(x)=\sum_{j=1}^{v_2} \gamma_{i,j}'x_i x_{v_2+(i+j-1) (mod o_3)

should be changed to

\Theta_i(x)=\sum_{j=1}^{v_1} \gamma_{i,j}x_j x_{v_1+(i+j-1) (mod o_2),

\Theta_i'(x)=\sum_{j=1}^{v_2} \gamma_{i,j}'x_j x_{v_2+(i+j-1) (mod o_3),

Note that $1 \le A (mod B) \le B$ for an integer A and a positive integer B,

in our definition.

Kind regards

Kyung-Ah Shim

****

Answer to Our Security Proof.

Due to the use of the multivariate quadratic map requiring additional random Vinegar variables, our trapdoor function is not permutation and the signature distribution is not uniformly distributed as presented in [1]. The authors [1] make the distribution of signatures uniform by using a random salt to the message being hashed and re-choosing a random salt instead of Vinegar variables.

We can use the same way to prove unforgeability of our scheme. For it, we need to propose a modified version: the modified signing algorithm is the same as the original one except that

-choose a random r \in {0, 1}^R, compute H(m, r)=h.

-If one of the linear systems has no signature then choose another random r’ and try again.

-Then the signature is (\tau, r).

In Verify algorithm, to verify a signature (\tau, r) on a message m, check whether the equation \cal P(\tau)=H(m, r) holds or not.

In the security proof, the H-query should be changed as:

For H-queries, the tuples in H-list are of the form (m_i, c_i, \tau_i, r_i, P(\tau_i)). When {\cal A} queries H at m_i \in \{0, 1\}^*,

i) If the query already appears on H-list in a tuple (m_i, c_i, \tau_i, r_i, P(\tau_i)) then {\cal} B returns H(m_i, r_i)=P(\tau_i).

ii) Otherwise, {\cal B} picks a random coin c_i \in {0,1} with Pr[c_i=0]=\frac{1}{q_S+1}.

-If c_i=1 then {\cal B} chooses a random \tau_i \in F_q^n and r_i \in {0, 1}^R, adds a tuple (m_i, c_i, \tau_i, r_i, P(\tau_i)) to H-list and returns H(m_i, r_i)=P(\tau_i).

- If c_i=0 then {\cal B} adds (m_i, c_i, r^*, *, \eta) to H-list from the instance and returns H(m_i, r^*)=\eta, where \eta is the given MQ-instance.

For Sign Queries. When {\cal A} makes a Sign-query on m_i, \cal B finds the corresponding tuple (m_i, c_i, \tau_i, r_i, P(\tau_i)) from H-list.

-If c_i=1 then \cal B responds with (\tau_i, r_i).

-If c_i=0 then \cal B reports failure and terminates.

Then the distribution of the outputs H(m_i, r_i) of our random oracle is identical to the distribution of \cal P(\tau), \tau \in_R F_q^n, since \tau is uniformly distributed over F_q^n and it is a valid signature satisfying \cal P(\tau)=H(m, r).

The rest of the proof is the same as that in [2].

[1] Koichi Sakumoto, Taizo Shirai, and Harunaga Hiwatari, On Provable Security of UOV and HFE Signature Schemes against Chosen-Message Attack, PQCrypto 2011, LNCS 7071, pp. 68–82, 2011.

[2] Kyung-Ah Shim, Cheol-Min Park, Namhun Koo: An Existential Unforgeable Signature Scheme Based on Multivariate Quadratic Equations. ASIACRYPT (1) 2017: pp. 37-64, 2017.

From: Ward Beullens <Ward@beullens.com>
Sent: Wednesday, October 24, 2018 9:45 AM
To: pqc-comments; pqc-forum@list.nist.gov
Subject: OFFICIAL COMMENT: HIMQ-3 :Key Recovery Attack

Dear all,

I have found a more efficient variant of the Key Recovery attack on HiMQ-3 that was described in the submission (Theorem 1). The main observation is that there are more equivalent keys than those described in Lemma 2. Concretely, the o_1 x o_2 and o_2 x o_1 part of the matrix \Sigma need not be zero. This allows for a better 'Good Key', and results in a more efficient Key Recovery Attack.

The most expensive step in the attack is solving a system of n − 1 bi-homogeneous equations and m quadratic equations in n variables. An upper bound to the complexity of solving this system is obtained by treating the equations as semi-regular.

This gives an estimated complexity of 2^124.8 field operations (F_256) for the HiMQ-3(256,31,15,15,14) parameter set, and 2^109.9 field operations for the HiMQ-3F(256,24,11,17,15) parameter set. So these parameters do not seem to reach Security Level 1.

Because of the bi-homogeneous structure of n-1 of the equations the actual complexity will be a bit lower (e.g. see p.11 of [1])

I communicated with the designers and they told me they had independently found a similar attack with the same complexity, and that they will be posting a message to the forum soon.

All the best,
Ward Beullens

[1] https://eprint.iacr.org/2012/223.pdf

From: 심경아 <shimkah221@gmail.com>
Sent: Thursday, October 25, 2018 5:09 AM
To: pqc-comments
Cc: pqc-forum@list.nist.gov
Subject: OFFICIAL COMMENT: HiMQ-3
Attachments: KRA-HiMQ-3.pdf

Dear all,

We are HiMQ-3 team. There is a missing point in our security analysis of HiMQ-3 against key recovery attacks (KRAs) using equivalent keys and good keys. We found KRAs using simpler equivalent keys and their corresponding good keys than ones in the submitted version. The result is that the complexity of the KRAs using good keys on HiMQ-3 is determined by solving n-1 bihomogeneous equations and m quadratic equations with n (not n+min(o_1, o_2)) variables.

We describe analysis of HiMQ-3 against KRAs using new equivalent keys and good keys in the attached file. According to this result, our submitted parameters are not sufficient. So, we select slightly modified parameters for given security level.

Best regards,

-HiMQ-3 Team

1 Key Recovery Attacks using Equivalent Keys on HiMQ-3

To find simpler equivalent keys, we consider the generalized version of our central map, $\overline{\mathcal{F}}^{(k)}$ for $1 \leq k \leq m$, given in Fig. 3.

Lemma 2. For the generalized central map given in Fig. 3, we can find equivalent keys S' and T' of the form given in Fig. 4 with high probability, where gray parts denote arbitrary entries and white parts denote zero entries and there are ones at the diagonal.

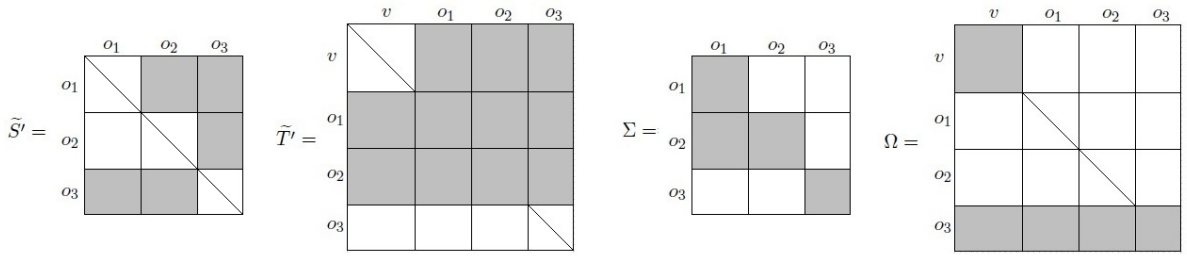

Fig. 4. Equivalent Key of HiMQ-3.

Proof. As in [48], we can find Σ and Ω given in Fig. 4 with high probability. Then there exist equivalent keys (S', T') of the form given in Fig. 4. □

To recover the equivalent key above, we need to solve a system of

$$\frac{m(o_1+o_2)(n+v+1)+o_3^2(o_3+1)}{2} - o_1o_2(v+1) - o_1^2 - o_2^2$$

cubic equations with $n(n-o_3)+m^2-(v^2+o_1^2+o_2^2+o_3^2)-o_1o_2$ variables.

2 Key Recovery Attacks using Good Keys on HiMQ-3

To further decrease this complexity, we use the notion of good keys which is a generalization of equivalent keys.

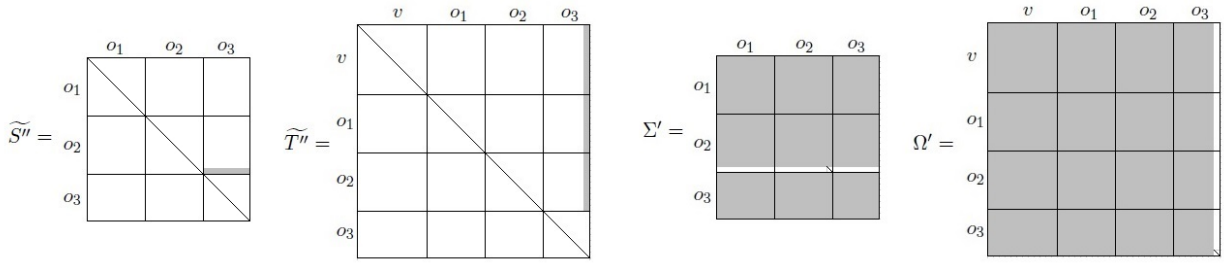

Fig. 5. Good Keys of HiMQ-3.

Lemma 3. Let S' and T' be equivalent keys for HiMQ-3 of the form given in Fig. 4. Then there are good keys S'' and T'' of the form given in Fig. 5. Only the last column of $\widetilde{T''}$ contains

arbitrary values in the first $v + o_1 + o_2$ rows, which are equal to the corresponding values in $\widetilde{T'}$. Respectively, only o_3 values of the $(o_1 + o_2)$-th row of $\widetilde{S''}$ contain arbitrary values, which are equal to the corresponding values in $\widetilde{S'}$.

Proof. We can find Σ' and Ω' given in Fig. 5 with high probability. Then there exist good keys (S'', T'') of the form given in Fig. 5. □

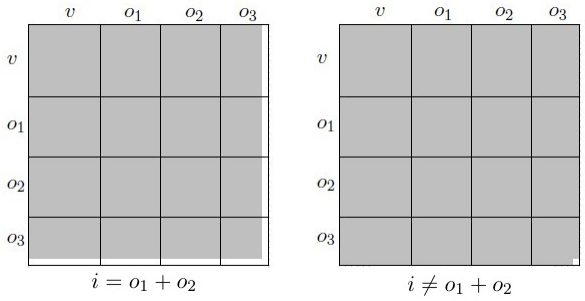

Figure 1: Quadratic Parts of $\mathcal{F}''^{(i)}$.

Finally, we get the central map $\mathcal{F}''$ in Fig. 6 after applying the transformations Σ' and Ω'. Thus, we obtain the following Theorem. It gives the same result as Rainbow.

Theorem 1. The complexity of the key recovery attack using good keys on HiMQ-3 is determined by solving $n - 1$ bihomogeneous equations and m quadratic equations with n variables.

After obtaining one column of T' and one row of S', all the other parts of T' and S' are revealed by linear equations as in [48]. Consequently, we recover the equivalent keys T' and S'.

HiMQ-3	# of Equations	# of Variables	d_{reg}	Complexity
KRA	144,654(Cubic)	8,586	555	2^{6060}
KRA with Equi. Keys	72,675(Cubic)	5,175	376	2^{3989}
KRA with Good Keys	125(Quad.)	81	16	2^{132}

Table. 3 Lower-bound on the Complexity of the KRAs using Equivalent Keys and Good Keys for HiMQ-3($\mathbb{F}_{2^8}, 36, 15, 15, 15$)

Table 3 shows improvements of lower bounds ($\alpha = 2$) on the complexities of solving the resulting systems by HF5 achieved by the KRAs using equivalent keys and good keys for HiMQ($\mathbb{F}_{2^8}, 36, 15, 15, 15$). In general, only the number of variables is reduced, as we find simpler equivalent keys maintaining the number of equations. However, the number of equations in our KRAs with equivalent keys is also changed, as we use the equivalent keys for the generalized central map given in Fig. 3.

3 Selection of Parameters

According to our security analysis of HiMQ-3, we summarize its complexities of HiMQ-3 against all the known attacks.

- Direct attacks: Complexity of HiMQ-3 against the direct attacks is estimated as

$$C_{Direct}(q, m, n) = C_{MQ}(q, m, n),$$

where $C_{MQ}(q, m, n)$ denotes complexity of solving a semi-regular system of m equations in n variables defined over $\mathbb{F}_q$ by using HF5 algorithm.

- KRAs: Complexity of HiMQ-3 against the KRAs using good keys is

$$C_{KRAg}(q,\ m,\ n) = C_{MQ}(q,\ m + n - 1,\ n).$$

- MinRank Attacks: Complexity of HiMQ-3 against the MinRank attacks is

$$C_{MR}(q, v, o_1, m) = o_1 \cdot g^{v-o_1+3}.$$

- HighRank Attacks: Complexity of HiMQ-3 against the HighRank attacks is

$$C_{HR}(q, o_3, n) = q^{o_3} \cdot \frac{n^3}{6}.$$

- Kipnis-Shamir Attacks: Complexity of HiMQ-3 against the Kipnis-Shamir Attacks is

$$C_{KS}(q, v, o_1, o_2, o_3) = q^{v+o_1+o_2-o_3}.$$

The complexities of HiMQ-3F are the same as those of HiMQ-3. Finally, we select secure parameters of HiMQ-3 at 128, 192 and 256-bit security level and summarize complexities of our selected parameter against the known attacks in Table 5. For computing of complexities against direct attacks and KRAs using good keys, we use HF5 algorithm with $\alpha = 2$.

λ	$(\mathbb{F}_q, v, o_1, o_2, o_3)$	Direct	KRA	Kipnis-Shamir	MinRank	HighRank
128	HiMQ-3($\mathbb{F}_{2^8}, 36, 15, 15, 15$)	2^{135}	2^{132}	2^{408}	2^{195}	2^{136}
	HiMQ-3F($\mathbb{F}_{2^8}, 36, 13, 17, 15$)	2^{135}	2^{132}	2^{408}	2^{211}	2^{136}
192	HiMQ-3($\mathbb{F}_{2^8}, 56, 25, 25, 25$)	2^{213}	2^{195}	2^{648}	2^{276}	2^{218}
256	HiMQ-3($\mathbb{F}_{2^8}, 84, 33, 33, 32$)	2^{273}	2^{263}	2^{936}	2^{437}	2^{275}

Table 5. Complexities of HiMQ-3 and HiMQ-3F against All Known Attacs