MQDSS

NIST Postquantum Cryptography Project

Carl Miller

March 9, 2018

The Basics

- It’s a digital signature scheme.

- Security proof is based on the hardness of the ”MQ problem” (solving a random quadratic polynomial system). Claims to be the first such scheme. (?)

- Involves an identification protocol (i.e., a protocol that merely proves the identity of the sender) that is converted into a signature protocol.

The MQ Problem

The MQ Problem

- A different form of the problem is known to be NP-complete. (?)
- The authors imply that the best known classical algorithms for the problem are exponential. (They also measure the performance of Grover's algorithm.)

Starting Point:
The Sakumoto-Shirai-Hiwatari Protocol

The identification problem

Goal: Alice proves to Bob that she possesses the secret key, without revealing any information about the key.

The SSH 5-Pass Protocol

Alice generates random quadratic **F** and **v** := **F** (**s**).

s

F, v

The SSH 5-Pass Protocol

Alice generates random quadratic **F** and **v** := **F** (**s**).
Alice choose random $\mathbf{r_0}$ and sets $\mathbf{r_1} = \mathbf{s} - \mathbf{r_0}$.

s, $\mathbf{r_0}$, $\mathbf{r_1}$

F, **v**

The SSH 5-Pass Protocol

Alice generates random quadratic **F** and **v** := **F** (**s**).

Alice choose random $\mathbf{r_0}$ and sets $\mathbf{r_1} = \mathbf{s} - \mathbf{r_0}$.

Alice reveals some ``masked" information:

$$\alpha\, \mathbf{r_0} - \mathbf{t_0}\,, \quad \alpha\, \mathbf{F(r_0)} - \mathbf{e_0}\,, \quad \mathbf{r_1}$$

where $\mathbf{t_0}$, $\mathbf{e_0}$ are chosen by Alice and α is a scalar chosen by Bob.

s, $\mathbf{r_0}$, $\mathbf{r_1}$

F, v

The SSH 5-Pass Protocol

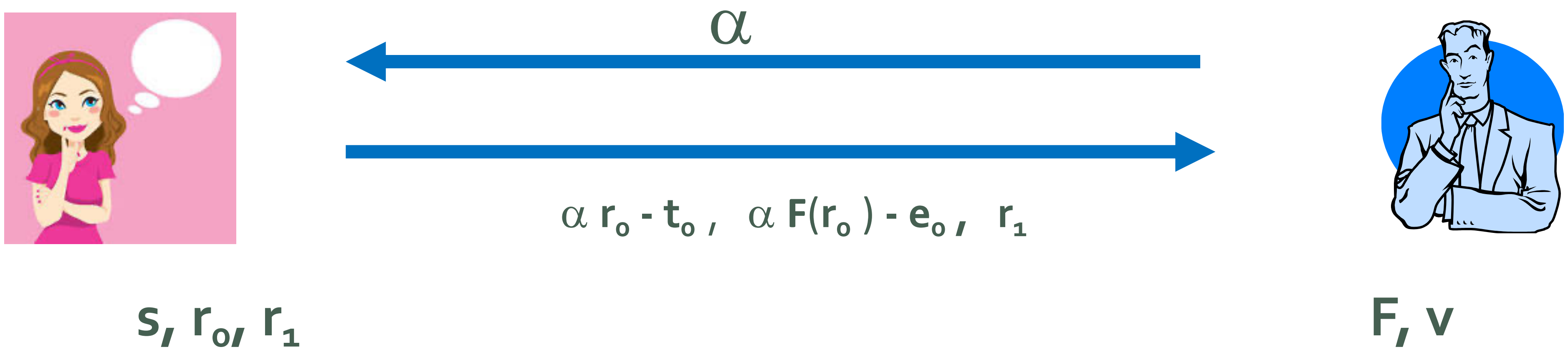
Alice generates random quadratic **F** and **v** := **F** (**s**).

Alice choose random $\mathbf{r_0}$ and sets $\mathbf{r_1} = \mathbf{s} - \mathbf{r_0}$.

Alice reveals some ``masked" information:

$$\alpha\, \mathbf{r_0} - \mathbf{t_0}\,, \quad \alpha\, \mathbf{F(r_0)} - \mathbf{e_0}\,, \quad \mathbf{r_1}$$

where $\mathbf{t_0}, \mathbf{e_0}$ are chosen by Alice and α is a scalar chosen by Bob.



The SSH 5-Pass Protocol

This information reveals nothing at all to Bob about s.

However – through the use of commitment functions – Bob can verify that Alice had to know a valid element of $\mathbf{F}^{-1}(\mathbf{v})$ to generate her part.

α ←

→ $\alpha\, \mathbf{r}_0 - \mathbf{t}_0\,,\ \ \alpha\, \mathbf{F}(\mathbf{r}_0) - \mathbf{e}_0\,,\ \ \mathbf{r}_1$

$\mathbf{s}, \mathbf{r}_0, \mathbf{r}_1$

$\mathbf{F}, \mathbf{v}$

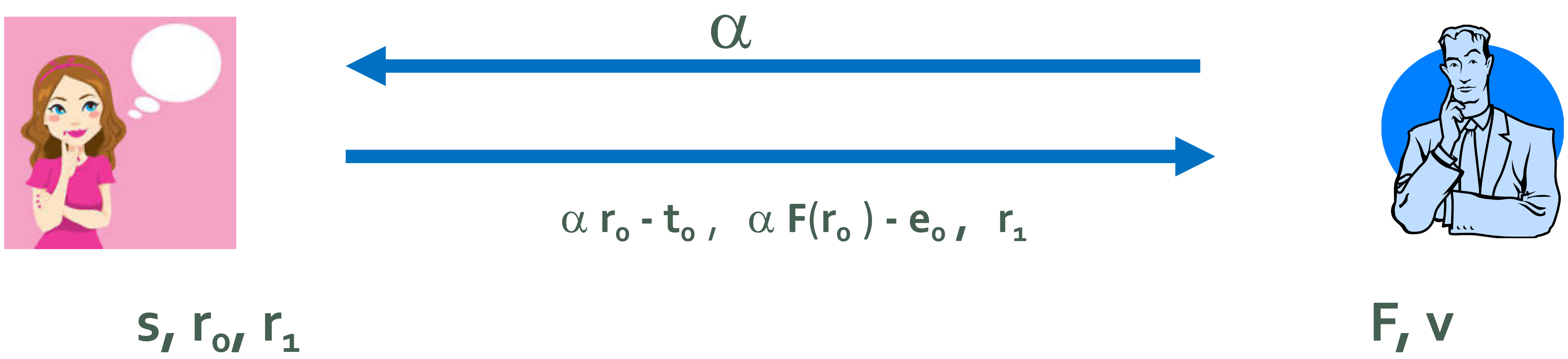
$\mathcal{P}(\mathsf{pk}, \mathsf{sk})$		$\mathcal{V}(\mathsf{pk})$
//setup		
$\mathbf{r}_0, \mathbf{t}_0 \leftarrow_R \mathbb{F}_q^n, \mathbf{e}_0 \leftarrow_R \mathbb{F}_q^m$		
$\mathbf{r}_1 \leftarrow \mathbf{s} - \mathbf{r}_0$		
//commit		
$c_0 \leftarrow Com(\mathbf{r}_0, \mathbf{t}_0, \mathbf{e}_0)$		
$c_1 \leftarrow Com(\mathbf{r}_1, \mathbf{G}(\mathbf{t}_0, \mathbf{r}_1) + \mathbf{e}_0)$	$\xrightarrow{\mathsf{com} = (c_0, c_1)}$	*//challenge 1* $\alpha \leftarrow_R \mathbb{F}_q$
//first response	$\xleftarrow{\mathsf{ch}_1 = \alpha}$	
$\mathbf{t}_1 \leftarrow \alpha\mathbf{r}_0 - \mathbf{t}_0$		
$\mathbf{e}_1 \leftarrow \alpha\mathbf{F}(\mathbf{r}_0) - \mathbf{e}_0$	$\xrightarrow{\mathsf{resp}_1 = (\mathbf{t}_1, \mathbf{e}_1)}$	*//challenge 2* $\mathsf{ch}_2 \leftarrow_R \{0, 1\}$
//second response	$\xleftarrow{\mathsf{ch}_2}$	
If $\mathsf{ch}_2 = 0,\ \mathsf{resp}_2 \leftarrow \mathbf{r}_0$		
Else $\mathsf{resp}_2 \leftarrow \mathbf{r}_1$	$\xrightarrow{\mathsf{resp}_2}$	*//verify*
		If $\mathsf{ch}_2 = 0,$ parse $\mathsf{resp}_2 = \mathbf{r}_0,$ check $c_0 \stackrel{?}{=} Com(\mathbf{r}_0, \alpha\mathbf{r}_0 - \mathbf{t}_1, \alpha\mathbf{F}(\mathbf{r}_0) - \mathbf{e}_1)$
		Else, parse $\mathsf{resp}_2 = \mathbf{r}_1,$ check $c_1 \stackrel{?}{=} Com(\mathbf{r}_1, \alpha(\mathbf{v} - \mathbf{F}(\mathbf{r}_1)) - \mathbf{G}(\mathbf{t}_1, \mathbf{r}_1) - \mathbf{e}_1)$

Fig. 3.1: The SSH 5-pass IDS by Sakumoto, Shirai, and Hiwatari [41]

The SSH 5-Pass Protocol

This is proved secure if the MQ problem is hard and if the commitment functions are secure. (?)

(Note: At best, the protocol is only sound with probability close to ½. So, it needs to be repeated to work.)



The Main Protocol

Toolbox

MQDSS makes use of:

- Hash functions.

  01011...10 → **01011...10**

  Variable length → Fixed length

- Pseudorandom number generators

  01011...10 → **01011......**

  seed → Unbounded length

- Extendable output functions

- Commitment functions

All are derived from SHA-3.

The Fiat-Shamir Transform

The FT transform converts an **identification protocol** into a **digital signature scheme.**

Suppose given an identification scheme.
Suppose that Alice wishes to sign a message, **M**.

Secret key

Public key

The Fiat-Shamir Transform

Alice runs the identification protocol with herself in Bob’s place.

Left-Alice generates all her private randomness from the secret key.
Right-Alice generates all her private randomness from the public key **and** the message **M**.

Secret key

Public key

The Fiat-Shamir Transform

Alice records a transcript of the protocol and sends it to Bob.
Bob checks that it is valid using the public key.

Transcript

M (message)

The Fiat-Shamir Transform

If the identification protocol satisfied certain security assumptions, then the derived signature scheme is EUF-CMA. (?)

Theorem 5.2 (EU-CMA security of $q2$-signature schemes [16]). *Let $k \in \mathbb{N}$, $\mathsf{IDS}(1^k)$ a q2-IDS that has a key relation R, is KOW secure, is honest-verifier zero-knowledge, and has a q2-extractor $\mathcal{E}$. Then $q2\text{-}\mathsf{Dss}(1^k)$, the q2-signature scheme derived applying Construction 5.1 is existentially unforgeable under adaptive chosen message attacks.*

The MQDSS Protocol is a Fiat-Shamir transformation of several copies of the SSH 5-Pass Protocol.

The MQDSS Signature Scheme

$\mathsf{Sign}(\mathsf{sk}, M)$

$S_{\mathbf{F}}, S_{\mathbf{s}}, S_{\mathbf{rte}} \leftarrow \mathrm{PRG}_{\mathsf{sk}}(\mathsf{sk})$

$\mathbf{F} \leftarrow \mathrm{XOF}_{\mathbf{F}}(S_{\mathbf{F}})$

$\mathbf{s} \leftarrow \mathrm{PRG}_{\mathbf{s}}(S_{\mathbf{s}})$

$\mathsf{pk} := (S_{\mathbf{F}}, \mathbf{F}(\mathbf{s}))$

$R \leftarrow \mathcal{H}(\mathsf{sk}||M)$

$D \leftarrow \mathcal{H}(\mathsf{pk}||R||M)$

$\mathbf{r}_0^{(1)}, \ldots, \mathbf{r}_0^{(r)}, \mathbf{t}_0^{(1)}, \ldots, \mathbf{t}_0^{(r)}, \mathbf{e}_0^{(1)}, \ldots, \mathbf{e}_0^{(r)} \leftarrow \mathrm{PRG}_{\mathbf{rte}}(S_{\mathbf{rte}}, D)$

For $j \in \{1, \ldots, r\}$ **do**

$\quad \mathbf{r}_1^{(j)} \leftarrow \mathbf{s} - \mathbf{r}_0^{(j)}$

$\quad c_0^{(j)} \leftarrow Com_0(\mathbf{r}_0^{(j)}, \mathbf{t}_0^{(j)}, \mathbf{e}_0^{(j)})$

$\quad c_1^{(j)} \leftarrow Com_1(\mathbf{r}_1^{(j)}, \mathbf{G}(\mathbf{t}_0^{(j)}, \mathbf{r}_1^{(j)}) + \mathbf{e}_0^{(j)})$

$\quad \mathsf{com}^{(j)} := (c_0^{(j)}, c_1^{(j)})$

$\sigma_0 \leftarrow \mathcal{H}(\mathsf{com}^{(1)}||\mathsf{com}^{(2)}||\ldots||\mathsf{com}^{(r)})$

$\mathsf{ch}_1 \leftarrow H_1(D, \sigma_0)$

Parse ch_1 as $\mathsf{ch}_1 = (\alpha^{(1)}, \alpha^{(2)}, \ldots, \alpha^{(r)}), \alpha^{(j)} \in \mathbb{F}_q$

Generate private "randomness" from a secret key.

The MQDSS Signature Scheme

$\mathsf{Sign}(\mathsf{sk}, M)$

$S_{\mathbf{F}}, S_{\mathbf{s}}, S_{\mathbf{rte}} \leftarrow \mathrm{PRG}_{\mathsf{sk}}(\mathsf{sk})$
$\mathbf{F} \leftarrow \mathrm{XOF}_{\mathbf{F}}(S_{\mathbf{F}})$
$\mathbf{s} \leftarrow \mathrm{PRG}_{\mathbf{s}}(S_{\mathbf{s}})$
$\mathsf{pk} := (S_{\mathbf{F}}, \mathbf{F}(\mathbf{s}))$
$R \leftarrow \mathcal{H}(\mathsf{sk}||M)$
$D \leftarrow \mathcal{H}(\mathsf{pk}||R||M)$
$\mathbf{r}_0^{(1)}, \ldots, \mathbf{r}_0^{(r)}, \mathbf{t}_0^{(1)}, \ldots, \mathbf{t}_0^{(r)}, \mathbf{e}_0^{(1)}, \ldots, \mathbf{e}_0^{(r)} \leftarrow \mathrm{PRG}_{\mathbf{rte}}(S_{\mathbf{rte}}, D)$
For $j \in \{1, \ldots, r\}$ **do**
  $\mathbf{r}_1^{(j)} \leftarrow \mathbf{s} - \mathbf{r}_0^{(j)}$
  $c_0^{(j)} \leftarrow Com_0(\mathbf{r}_0^{(j)}, \mathbf{t}_0^{(j)}, \mathbf{e}_0^{(j)})$
  $c_1^{(j)} \leftarrow Com_1(\mathbf{r}_1^{(j)}, \mathbf{G}(\mathbf{t}_0^{(j)}, \mathbf{r}_1^{(j)}) + \mathbf{e}_0^{(j)})$
  $\mathsf{com}^{(j)} := (c_0^{(j)}, c_1^{(j)})$
$\sigma_0 \leftarrow \mathcal{H}(\mathsf{com}^{(1)}||\mathsf{com}^{(2)}||\ldots||\mathsf{com}^{(r)})$
$\mathsf{ch}_1 \leftarrow H_1(D, \sigma_0)$
Parse ch_1 as $\mathsf{ch}_1 = (\alpha^{(1)}, \alpha^{(2)}, \ldots, \alpha^{(r)}), \alpha^{(j)} \in \mathbb{F}_q$

Pick quadratic function **F** and random vector **s.**

The MQDSS Signature Scheme

$\mathsf{Sign}(\mathsf{sk}, M)$

$S_{\mathbf{F}}, S_{\mathbf{s}}, S_{\mathbf{rte}} \leftarrow \mathrm{PRG}_{\mathsf{sk}}(\mathsf{sk})$

$\mathbf{F} \leftarrow \mathrm{XOF}_{\mathbf{F}}(S_{\mathbf{F}})$

$\mathbf{s} \leftarrow \mathrm{PRG}_{\mathbf{s}}(S_{\mathbf{s}})$

$\mathsf{pk} := (S_{\mathbf{F}}, \mathbf{F}(\mathbf{s}))$

$R \leftarrow \mathcal{H}(\mathsf{sk}||M)$

$D \leftarrow \mathcal{H}(\mathsf{pk}||R||M)$

$\mathbf{r}_0^{(1)}, \ldots, \mathbf{r}_0^{(r)}, \mathbf{t}_0^{(1)}, \ldots, \mathbf{t}_0^{(r)}, \mathbf{e}_0^{(1)}, \ldots, \mathbf{e}_0^{(r)} \leftarrow \mathrm{PRG}_{\mathbf{rte}}(S_{\mathbf{rte}}, D)$

For $j \in \{1, \ldots, r\}$ **do**

$\quad \mathbf{r}_1^{(j)} \leftarrow \mathbf{s} - \mathbf{r}_0^{(j)}$

$\quad c_0^{(j)} \leftarrow Com_0(\mathbf{r}_0^{(j)}, \mathbf{t}_0^{(j)}, \mathbf{e}_0^{(j)})$

$\quad c_1^{(j)} \leftarrow Com_1(\mathbf{r}_1^{(j)}, \mathbf{G}(\mathbf{t}_0^{(j)}, \mathbf{r}_1^{(j)}) + \mathbf{e}_0^{(j)})$

$\quad \mathsf{com}^{(j)} := (c_0^{(j)}, c_1^{(j)})$

$\sigma_0 \leftarrow \mathcal{H}(\mathsf{com}^{(1)}||\mathsf{com}^{(2)}||\ldots||\mathsf{com}^{(r)})$

$\mathsf{ch}_1 \leftarrow H_1(D, \sigma_0)$

Parse ch_1 as $\mathsf{ch}_1 = (\alpha^{(1)}, \alpha^{(2)}, \ldots, \alpha^{(r)}), \alpha^{(j)} \in \mathbb{F}_q$

Split **s** randomly into a sum of two vectors (in several ways).

The MQDSS Signature Scheme

For $j \in \{1, \ldots, r\}$ **do**

$$\mathbf{t}_1^{(j)} \leftarrow \alpha^{(j)}\mathbf{r}_0^{(j)} - \mathbf{t}_0^{(j)}, \quad \mathbf{e}_1^{(j)} \leftarrow \alpha^{(j)}\mathbf{F}(\mathbf{r}_0^{(j)}) - \mathbf{e}_0^{(j)}$$

$$\mathsf{resp}_1^{(j)} := (\mathbf{t}_1^{(j)}, \mathbf{e}_1^{(j)})$$

$$\sigma_1 \leftarrow (\mathsf{resp}_1^{(1)} || \mathsf{resp}_1^{(2)} || \ldots || \mathsf{resp}_1^{(r)})$$

$$\mathsf{ch}_2 \leftarrow H_2(D, \sigma_0, \mathsf{ch}_1, \sigma_1)$$

Parse ch_2 as $\mathsf{ch}_2 = (b^{(1)}, b^{(2)}, \ldots, b^{(r)}), b^{(j)} \in \{0, 1\}$

For $j \in \{1, \ldots, r\}$ **do**

$$\mathsf{resp}_2^{(j)} \leftarrow \mathbf{r}_{b^{(j)}}^{(j)}$$

$$\sigma_2 \leftarrow (\mathsf{resp}_2^{(1)} || \mathsf{resp}_2^{(2)} || \ldots || \mathsf{resp}_2^{(r)} || c_{1-b^{(1)}}^{(1)} || c_{1-b^{(2)}}^{(2)} || \ldots || c_{1-b^{(r)}}^{(r)})$$

Return $\sigma = (R, \sigma_0, \sigma_1, \sigma_2)$

Fig. 7.2: MQDSS-q-n signature generation

Simulate 5-Pass SSH Protocol

The MQDSS Signature Scheme

For $j \in \{1, \ldots, r\}$ **do**

$\mathbf{t}_1^{(j)} \leftarrow \alpha^{(j)} \mathbf{r}_0^{(j)} - \mathbf{t}_0^{(j)}, \quad \mathbf{e}_1^{(j)} \leftarrow \alpha^{(j)} \mathbf{F}(\mathbf{r}_0^{(j)}) - \mathbf{e}_0^{(j)}$

$\mathsf{resp}_1^{(j)} := (\mathbf{t}_1^{(j)}, \mathbf{e}_1^{(j)})$

$\sigma_1 \leftarrow (\mathsf{resp}_1^{(1)} || \mathsf{resp}_1^{(2)} || \ldots || \mathsf{resp}_1^{(r)})$

$\mathsf{ch}_2 \leftarrow H_2(D, \sigma_0, \mathsf{ch}_1, \sigma_1)$

Parse ch_2 as $\mathsf{ch}_2 = (b^{(1)}, b^{(2)}, \ldots, b^{(r)}), b^{(j)} \in \{0, 1\}$

For $j \in \{1, \ldots, r\}$ **do**

$\mathsf{resp}_2^{(j)} \leftarrow \mathbf{r}_{b^{(j)}}^{(j)}$

$\sigma_2 \leftarrow (\mathsf{resp}_2^{(1)} || \mathsf{resp}_2^{(2)} || \ldots || \mathsf{resp}_2^{(r)} || c_{1-b^{(1)}}^{(1)} || c_{1-b^{(2)}}^{(2)} || \ldots || c_{1-b^{(r)}}^{(r)})$

Return $\sigma = (R, \sigma_0, \sigma_1, \sigma_2)$

Fig. 7.2: MQDSS-q-n signature generation

Send transcript

The MQDSS Signature Scheme

Theorem: If the various SHA-3 derived functions are secure, and if the MQ problem is hard, then MQDSS is EUF-CMA secure in the random oracle model.

Theorem 10.1. MQDSS *is EU-CMA-secure in the random oracle model, if the following conditions are satisfied:*

- *the search version of the $\mathcal{MQ}$ problem is intractable in the average case,*
- *the hash functions $\mathcal{H}$, H_1, and H_2 are modeled as random oracles,*
- *the commitment functions Com_0 and Com_1 are computationally binding, computationally hiding, and have $\mathcal{O}(k)$ bits of output entropy,*
- *the function XOF_F is modeled as random oracle and*
- *the pseudorandom generators PRG_{sk}, PRG_{s} and $PRG_{\mathbf{rte}}$ have outputs computationally indistinguishable from random for any polynomial time adversary.*

Performance Claims

k = secret key size
q = finite field size
r = # of copies of SSH

Security category	k	q	n	r	Public key size (bytes)	Secret key size (bytes)	Signature size (bytes)
1-2	256	4	88	378	54	32	37108
1-2	256	16	56	281	60	32	32660
1-2	256	32	48	268	62	32	32760
1-2	256	64	40	262	62	32	32028
3-4	384	4	128	567	80	48	81744
3-4	384	16	72	421	84	48	65772
3-4	384	32	64	402	88	48	67632
3-4	384	64	64	393	102	48	82626
5-6	512	4	160	756	104	64	139232
5-6	512	16	96	562	112	64	117024
5-6	512	31	88	537	119	64	123101
5-6	512	32	88	536	119	64	122872
5-6	512	64	88	524	130	64	137416

Performance Claims

Security category	q	n	Best classical attack		Best quantum attack		
			algorithm	Field op.	algorithm	Gates	Depth
1-2	4	88	Crossbread	2^{152}	Crossbread	2^{93}	2^{83}
1-2	16	56	Crossbread	2^{163}	Crossbread	2^{98}	2^{89}
1-2	32	48	HybridF5	2^{159}	Crossbread	2^{96}	2^{88}
1-2	64	40	HybridF5	2^{143}	Crossbread	2^{89}	2^{81}
3-4	4	128	Crossbread	2^{226}	Crossbread	2^{129}	2^{119}
3-4	16	72	HybridF5	2^{210}	Crossbread	2^{123}	2^{113}
3-4	32	64	HybridF5	2^{205}	Crossbread	2^{125}	2^{115}
3-4	64	64	HybridF5	2^{217}	Crossbread	2^{136}	2^{127}
5-6	4	160	Crossbread	2^{287}	Crossbread	2^{158}	2^{147}
5-6	16	96	HybridF5	2^{273}	Crossbread	2^{162}	2^{152}
5-6	31	88	HybridF5	2^{273}	Crossbread	2^{179}	2^{168}
5-6	32	88	HybridF5	2^{274}	Crossbread	2^{174}	2^{164}
5-6	64	88	HybridF5	2^{291}	Crossbread	2^{203}	2^{192}

Table 8.4: Best classical and quantum attacks against the additional parameter sets

Performance Claims

We compiled the code using GCC version `6.3.0-18`, with the compiler optimization flag `-O3`. The median resulting cycle counts are listed in the table below.

	keygen	signing	verification
MQDSS-31-48	1 206 730	52 466 398	38 686 506
MQDSS-31-64	2 806 750	169 298 364	123 239 874

Advantages and Limitations

+ A security proof based on a simple problem.

"the first multivariate signature scheme that is provably secure … We believe MQDSS … [is] a step towards regaining confidence in MQ cryptography."

+ Small keys.

- Large signatures.

- EUF-CMA proof is in ROM (random oracle model) rather than QROM (quantum random oracle model).

MQDSS

NIST Postquantum Cryptography Project

Carl Miller

March 9, 2018