An Attack on Some Signature Schemes Constructed From 5-Pass Identification Schemes

Daniel Kales Greg Zaverucha

CANS 2020, December 14-16, 2020

Outline

- Background
- Attacking MQDSS
- A General Attack Framework
- Investigating other Schemes

Background

Canonical 5-pass Identification Schemes

Prover(sk)		**Verifier**(pk)
$\mathsf{com} \leftarrow \mathcal{P}_0(sk)$		
	$\xrightarrow{\mathsf{com}}$	
	$\xleftarrow{\mathsf{ch}_1}$	$\mathsf{ch}_1 \xleftarrow{\$} \mathsf{ChS}_1(1^k)$
$\mathsf{resp}_1 \leftarrow \mathcal{P}_1(sk, \mathsf{com}, \mathsf{ch}_1)$		
	$\xrightarrow{\mathsf{resp}_1}$	
	$\xleftarrow{\mathsf{ch}_2}$	$\mathsf{ch}_2 \xleftarrow{\$} \mathsf{ChS}_2(1^k)$
$\mathsf{resp}_2 \leftarrow \mathcal{P}_2(sk, \mathsf{com}, \mathsf{ch}_1, \mathsf{resp}_1, \mathsf{ch}_2)$		
	$\xrightarrow{\mathsf{resp}_2}$	
		$b \leftarrow \mathcal{V}(pk, \mathsf{com}, \mathsf{ch}_1, \mathsf{resp}_1, \mathsf{ch}_2, \mathsf{resp}_2)$

Easily generalized to (2n+1)-pass Identification Schemes.

Fiat-Shamir Transformation

Prover(sk)		**Verifier**(pk)
$\mathsf{com} \leftarrow \mathcal{P}_0(sk)$		
$\mathsf{ch}_1 \leftarrow \mathcal{H}(\mathsf{com})$		
$\mathsf{resp}_1 \leftarrow \mathcal{P}_1(sk, \mathsf{com}, \mathsf{ch}_1)$		
$\mathsf{ch}_2 \leftarrow \mathcal{H}(\mathsf{com}, \mathsf{ch}_1, \mathsf{resp}_1)$		
$\mathsf{resp}_2 \leftarrow \mathcal{P}_2(sk, \mathsf{com}, \mathsf{ch}_1, \mathsf{resp}_1, \mathsf{ch}_2)$		
	$\xrightarrow{\mathsf{com}, \mathsf{resp}_1, \mathsf{resp}_2}$	
		$\mathsf{ch}_1 \leftarrow \mathcal{H}(\mathsf{com})$
		$\mathsf{ch}_2 \leftarrow \mathcal{H}(\mathsf{com}, \mathsf{ch}_1, \mathsf{resp}_1)$
		$b \leftarrow \mathcal{V}(pk, \mathsf{com}, \mathsf{ch}_1, \mathsf{resp}_1, \mathsf{ch}_2, \mathsf{resp}_2)$

- Decades old standard technique to make public-coin protocols non-interactive.
- Also include message in $\mathcal{H}$ to get signature

Parallel Repetition

Soundness error of one invocation of the protcol might be too large!

- Many existing protocols have a soundness error of $\epsilon \approx \frac{1}{2}$
- We can boost the soundness by executing many repetitions in parallel

com	com	com	com	com	com	com	com
ch_1	ch_1	ch_1	ch_1	ch_1	ch_1	ch_1	ch_1
$resp_1$	$resp_1$	$resp_1$	$resp_1$	$resp_1$	$resp_1$	$resp_1$	$resp_1$
ch_2	ch_2	ch_2	ch_2	ch_2	ch_2	ch_2	ch_2
$resp_2$	$resp_2$	$resp_2$	$resp_2$	$resp_2$	$resp_2$	$resp_2$	$resp_2$

The ϵ^r-Heurisitic

How many parallel repetitions do we need? Usually, r is chosen so that

$$\epsilon^r < 2^{-\kappa},$$

where ϵ is the soundness error of a single repetition, κ the security parameter.

- Secure for many interactive protocols
- Things get more complicated when we combine
  - 5 rounds instead of 3 rounds,
  - Parallel repetition and
  - Fiat-Shamir transformation.

Attacking MQDSS

MQDSS

MQDSS is a post-quantum signature scheme by Chen et al.

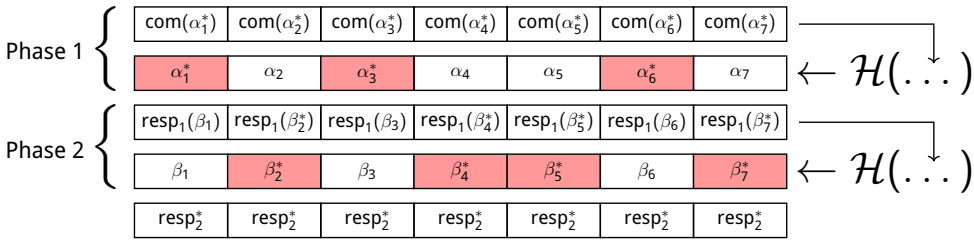
- Second-round candidate in NIST post-quantum standardization project
- 5-pass identification scheme of (Sakumoto et al., CRYPTO'11)
  - Security based on hardness of $\mathcal{MQ}$ problem
  - One-way-ness of Multivariate Quadratic maps
- Chen et al. give a proof of security of their 5-round Fiat-Shamir transformation
  - Proof non-tight due to use of forking lemma
  - # of parallel repetitions chosen with ϵ^r heuristic

Attack on MQDSS

Observation: Guessing only one of the two challenges allows an attacker to cheat. Even better, guessing wrong in the first challenge can be corrected by later guessing second challenge.

- Attacker: Use non-interactive nature of the protocol!
  - Try new com until ch_1 guessed correctly in N_1 repetitions.
  - Save this state
  - Try new resp_1 values until ch_2 guessed correctly in the remaining $r - N_1$ repetitions
  - Use correct guesses of ch_1 or ch_2 to cheat
    - One can compare this step to a HVZK-Simulator

Attack on MQDSS



- Repeat Phase 1 until we have N_1 correct guesses
- Then repeat Phase 2 until the remaining repetitions are guessed correctly
- Main cost of attack: Calling $\mathcal{H}$ (SHAKE256)

Choosing Attack Parameters

Important question: How many repetitions to attack for the first challenge?

- Depends on the relative size of challenge spaces!
- Both phases should be of similar cost for best attack tradeoff
- In MQDSS:
  - $|\mathrm{ChSp}_1| = 31$, $|\mathrm{ChSp}_2| = 2$
  - e.g., for 40 repetitions, attacking 12 in the first phase and 28 in the second phase is best
  - exact formula in the paper

Practical Verification of the Attack

Verification by attacking repetition-reduced variants of MQDSS:

- Implementation using MQDSS reference code
- Toy-Instance: **40** parallel repetitions (down from 135 for L1)
  - Should provide **38** bits of security (by ϵ^r-heuristic)
  - Theoretical attack estimate: 2^{29} calls to $\mathcal{H}$
  - Practical results: average of $2^{27.95}$ SHAKE256 calls
- Implementation publicly available:
  `https://github.com/dkales/mqdss-forgery`

Impact on MQDSS Instances

Applying the attack to full MQDSS v2 instances:

- In general: $\approx$ 40% more repetitions are needed to meet security target
- Below: attack complexity in red, new round numbers in bold

Parameter set	κ	$m = n$	q	r	N_1	$\#\mathcal{H}$	r_{new}
MQDSS-L1	128	48	31	135	41	2^{95}	**184**
MQDSS-L3	192	64	31	202	61	2^{141}	**277**
MQDSS-L5*	256	88	31	268	82	2^{180}	**370**

Results in a significant increase in signature size

A General Attack Framework

Genralization of Attack

What class of 5-round identification schemes can we attack?

- We defined a property called piecewise simulatability
- Property of the simulator of the HVZK IDS
  - Can output the transcript in two parts
  - While allowing one of the two challenges to be chosen by the environment
  - Simulator can still choose other challenge freely, and therefore produce a simulated transcript

Early Abort

Some 5-pass IDS have an interesting property we named early abort

- This captures the ability of the verifier to detect a wrong response to the first challenge
  - Detect as soon as it is received
  - Detect only after receiving the second challenge
- Another way to look at this property:
  - Can the 5-pass protocol be split into two interleaved 3-pass protocols?

Generic Attack Costs

We give formulas for the cost of the attack for schemes with and without early abort.

- Size of the two challenge spaces only factor in attack cost
- Attack optimal if both phases have similar cost

We show that ID schemes with early abort require fewer parallel repetitions since the attack cost is higher

- Attacker cannot correct wrong guesses for first challenge in second phase

Investigating other Schemes

A Five Round Picnic

Picnic2 is another PQ signature scheme based on non-interactive ZK, built using only symmetric primitives and the MPC-in-the-Head proof paradigm.

- The initial protocol is 5-round:
  - First challenge checks the pre-processing phase of the MPC
  - Second challenge checks the online phase of all but one MPC party
- In Picnic2, this is collapsed to 3 rounds:
  - Overhead: The prover must perform the online phase for all possible pre-processing phases.

Can we avoid the overhead with the 5-round variant?

A Five Round Picnic

Instance	# offline phases sign (verify)	# online phases sign (verify)	max. signature size [KiB]
`Picnic2-L1-FS`	343 (316)	343 (27)	13.47
`Picnic2-5-L1-FS`	2752 (2709)	43 (43)	16.46

- 5-round Picnic2: early abort + equal challenge spaces
- Best-case for attacker $\rightarrow$ double rounds vs. ϵ^r heuristic
  - Worse runtime and signature size when compared to collapsed 3-round variant

Other Schemes: PKP-DSS & Legroast

We also investigated other published PQ signature schemes

- PKP-DSS
  - Based on Permuted-Kernel-Problem, 5-rounds, ϵ^r heuristic
  - Parameters updated after announcement of the MQDSS attack on the NIST PQC mailing list.
- Legroast
  - Based on Legendre PRF
  - 7-rounds, attack not directly applicable

Conclusion

- Attack on signatures from 5-round ID schemes
  - Conceptually simple
  - Attacks parameter choices based on the ϵ^r-heuristic
- Breaks proposed parameter sets of MQDSS 2.0
  - Fix: MQDSS 2.1 adds $\approx 40\%$ more repetitions
  - MQDSS has not advanced to Round 3 of NIST PQC
- Analysis of other schemes

Future Work

- (Non-)Tightness of Proofs
  - Forking lemma introduces inherent loss
  - Alternative proof techniques?
- Collapsing 5-round protocols to 3 rounds
  - Picnic2 collapses to 3-rounds by committing to all possible first-round challenges
  - Can we do better?