Multiparty Generation of an RSA Modulus

Megan Chen Ran Cohen Jack Doerner
Yashvanth Kondi Eysa Lee Schuyler Rosefield
abhi shelat

Northeastern University
http://ia.cr/2020/370

In this work

Multiparty RSA modulus sampling
with malicious security
against a dishonest majority

In this work

Multiparty RSA modulus sampling with malicious security against a dishonest majority

- *First with n-parties from only OT and the Factoring Assumption*

In this work

Multiparty RSA modulus sampling with malicious security against a dishonest majority

- *First with n-parties from only OT and the Factoring Assumption*
- *First with cubic communication complexity in modulus length*

Why RSA Moduli?

Useful For

- Signatures and Encryption
  [RSA77], [Paillier-99]

Why RSA Moduli?

Useful For

- Signatures and Encryption
  [RSA77], [Paillier-99]
- Cryptographic Accumulators
  [Benaloh-deMare-93], [Camenisch-Lysyanskaya-02], [Li-Li-Xue-07], [Boneh-Bünz-Fisch-19]

Why RSA Moduli?

Useful For

- Signatures and Encryption
  [RSA77], [Paillier-99]
- Cryptographic Accumulators
  [Benaloh-deMare-93], [Camenisch-Lysyanskaya-02],
  [Li-Li-Xue-07], [Boneh-Bünz-Fisch-19]
- VDFs and Timelock Puzzles
  [Rivest-Shamir-Wagner-96], [Boneh-Bonneau-Bünz-Fisch-18],
  [Wesolowski-19], [Pietrzak-19], [Ephraim-Freitag-Komargodski-Pass-19]

Why RSA Moduli?

Useful For

- Signatures and Encryption
  [RSA77], [Paillier-99]
- Cryptographic Accumulators
  [Benaloh-deMare-93], [Camenisch-Lysyanskaya-02],
  [Li-Li-Xue-07], [Boneh-Bünz-Fisch-19]
- VDFs and Timelock Puzzles
  [Rivest-Shamir-Wagner-96], [Boneh-Bonneau-Bünz-Fisch-18],
  [Wesolowski-19], [Pietrzak-19], [Ephraim-Freitag-Komargodski-Pass-19]
- Efficient zk-SNARKs
  [Bünz-Fisch-Szepieniec-19], [Lai-Malavolta-19]

Why RSA Moduli?

Useful For

- Signatures and Encryption
  [RSA77], [Paillier-99]
- Cryptographic Accumulators
  [Benaloh-deMare-93], [Camenisch-Lysyanskaya-02],
  [Li-Li-Xue-07], [Boneh-Bünz-Fisch-19]
- VDFs and Timelock Puzzles
  [Rivest-Shamir-Wagner-96], [Boneh-Bonneau-Bünz-Fisch-18],
  [Wesolowski-19], [Pietrzak-19], [Ephraim-Freitag-Komargodski-Pass-19]
- Efficient zk-SNARKs
  [Bünz-Fisch-Szepieniec-19], [Lai-Malavolta-19]
- Many more...

Why RSA Moduli?

Useful For

- Signatures and Encryption
  [RSA77], [Paillier-99]
- Cryptographic Accumulators
  [Benaloh-deMare-93], [Camenisch-Lysyanskaya-02], [Li-Li-Xue-07], [Boneh-Bünz-Fisch-19]
- VDFs and Timelock Puzzles
  [Rivest-Shamir-Wagner-96], [Boneh-Bonneau-Bünz-Fisch-18], [Wesolowski-19], [Pietrzak-19], [Ephraim-Freitag-Komargodski-Pass-19]
- Efficient zk-SNARKs
  [Bünz-Fisch-Szepieniec-19], [Lai-Malavolta-19]
- Many more...

Local Biprime Sampling

1. Choose random $p \leftarrow \mathbb{Z}_{2^k}$
2. If p is not prime, repeat step 1
3. Sample q in the same way as p
4. Compute $N = p \cdot q$

Local Biprime Sampling

1. Choose random $p \leftarrow \mathbb{Z}_{2^k}$
2. If p is not prime, repeat step 1
3. Sample q in the same way as p
4. Compute $N = p \cdot q$

Miller-Rabin Test

$a^{p-1} \bmod p$

Local Biprime Sampling

1. Choose random $p \leftarrow \mathbb{Z}_{2^k}$
2. If p is not prime, repeat step 1
3. Sample q in the same way as p
4. Compute $N = p \cdot q$

Miller-Rabin Test

$a^{p-1} \bmod p$

MPC-unfriendly as p is secret

History of the Problem

1997

- [Boneh-Franklin-97/01]
- [Frankel-MacKenzie-Yung-98]
- [Poupard-Stern-98]
- [Gilboa-99]
- [Malkin-Wu-Boneh-99]
- [Algesheimer-Camenish-Shoup-02]
- [Damgård-Mikkelsen-10]
- [Gavin-12]
- [Hazay-Mikkelsen-Rabin-Toft-12]
- [Frederiksen-Lindell-Osheter-Pinkas-18]
- [Hazay-Mikkelsen-Rabin-Toft-Nicolosi-19]
- [This Work]
- [Chen-Hazay-Ishai-Kashnikov-Micciancio-Riviere-shelat-Muthu-Wang-20]

2020

Recent Progress

	[HMRT12] [HMRTN19]	[FLOP18]
Security	Malicious	Malicious
Parties	n	2
Corruptions	$n-1$	1
Uses HE	Yes	No
Assumptions	DCR, DDH	OT
ZK over Crypto	Yes	Almost No
Extra Leakage	No	Yes
Covert abort(DoS)	No	Yes

Open Question

	[HMRT12] [HMRTN19]	[FLOP18]	Open
Security	Malicious	Malicious	Malicious
Parties	n	2	n
Corruptions	$n-1$	1	$n-1$
Uses HE	Yes	No	
Assumptions	DCR, DDH	OT	
ZK over Crypto	Yes	Almost No	
Extra Leakage	No	Yes	
Covert abort(DoS)	No	Yes	

Open Question

	[HMRT12] [HMRTN19]	[FLOP18]	Open
Security	Malicious	Malicious	Malicious
Parties	n	2	n
Corruptions	$n-1$	1	$n-1$
Uses HE	Yes	No	No
Assumptions	DCR, DDH	OT	OT
ZK over Crypto	Yes	Almost No	
Extra Leakage	No	Yes	
Covert abort(DoS)	No	Yes	

Open Question

	[HMRT12] [HMRTN19]	[FLOP18]	Open
Security	Malicious	Malicious	Malicious
Parties	n	2	n
Corruptions	$n-1$	1	$n-1$
Uses HE	Yes	No	No
Assumptions	DCR, DDH	OT	OT
ZK over Crypto	Yes	Almost No	No
Extra Leakage	No	Yes	
Covert abort(DoS)	No	Yes	

Open Question

	[HMRT12] [HMRTN19]	[FLOP18]	Open
Security	Malicious	Malicious	Malicious
Parties	n	2	n
Corruptions	$n-1$	1	$n-1$
Uses HE	Yes	No	No
Assumptions	DCR, DDH	OT	OT
ZK over Crypto	Yes	Almost No	No
Extra Leakage	No	Yes	No
Covert abort(DoS)	No	Yes	No

O(s) to fix generically

Our Contribution

	[HMRT12] [HMRTN19]	[FLOP18]	[This Work]
Security	Malicious	Malicious	Malicious
Parties	n	2	n
Corruptions	$n-1$	1	$n-1$
Uses HE	Yes	No	No
Assumptions	DCR, DDH	OT	OT
ZK over Crypto	Yes	Almost No	No
Extra Leakage	No	Yes	No
Covert abort(DoS)	No	Yes	No

O(s) to fix generically

Our Contribution

	[HMRT12] [HMRTN19]	[FLOP18]	[This Work]
Security	Malicious	Malicious	Malicious
Parties	n	2	n
Corruptions	$n-1$	1	$n-1$
Uses HE	Yes	No	No
Assumptions	DCR, DDH	OT	OT, Factoring
ZK over Crypto	Yes	Almost No	No
Extra Leakage	No	Yes	No
Covert abort(DoS)	No	Yes	No

O(s) to fix generically

Our Contribution

	[HMRT12] [HMRTN19]	[FLOP18]	[This Work]
Security	Malicious	Malicious	Malicious
Parties	n	2	n
Corruptions	$n-1$	1	$n-1$
Uses HE	Yes	No	No
Assumptions	DCR, DDH	OT	OT, Factoring
ZK over Crypto	Yes	Almost No	No
Extra Leakage	No	Yes	No
Covert abort(DoS)	No	Yes	No
Analysis	Monolithic	Monolithic	Modular

Our Contribution

	[HMRT12] [HMRTN19]	[FLOP18]	[This Work]
Security	Malicious	Malicious	Malicious
Parties	n	2	n
Corruptions	$n-1$	1	$n-1$
Uses HE	Yes	No	No
Assumptions	DCR, DDH	OT	OT, Factoring
ZK over Crypto	Yes	Almost No	No
Extra Leakage	No	Yes	No
Covert abort(DoS)	No	Yes	No
Analysis	Monolithic	Monolithic	Modular
Comm. Cost	?	$\tilde{O}(sk^4)$	$\tilde{O}(nsk^3)$

Protocol Overview

1. Sample p, q as integer shares

Protocol Overview

1. Sample p, q as integer shares
2. Compute $N = p \cdot q$

Protocol Overview

1. Sample p, q as integer shares
2. Compute $N = p \cdot q$
3. Biprimality test on N

Protocol Overview

1. Sample p, q as integer shares
2. Compute $N = p \cdot q$
3. Biprimality test on N

   [BF97]: Test biprimality of *N* instead of primality of *p*,*q* individually

Protocol Overview

1. Sample p, q as integer shares
2. Compute $N = p \cdot q$
3. Biprimality test on N
4. Retroactive Consistency check

Protocol Overview

1. Sample p, q as integer shares
2. Compute $N = p \cdot q$
3. Biprimality test on N
4. Retroactive Consistency check

Shared by most works

Protocol Overview

We leverage the **Chinese Remainder Theorem** for efficiency

- Sieving speed of OT-based trial division but without the leakage
- OT-based secure multiplication over small fields: cheaper than single equivalent large field mult
- Component-wise consistency check plugs covert DoS attack

See the longer talk for…

- More details on challenges and techniques to surmount them
- Component-wise cost analysis
- Jack’s pandemic hair

Thanks!

http://ia.cr/2020/370