Group-Signature Schemes on Constrained Devices

Raphael Spreitzer and Jörn-Marc Schmidt

Institute for Applied Information Processing and Communications (IAIK)
Graz University of Technology
Inffeldgasse 16a, A-8010 Graz, Austria

raphael.spreitzer@iaik.tugraz.at

Group-Signature Schemes (GSS)

- Introduced by Chaum and van Heyst [CvH91]
- Members within a predefined group are able to sign messages on behalf of the group
- Verifier can only determine whether a signature stems from a specific group
- ... but verifier cannot determine the ID of the signer
- Participants
  - Signer
  - Verifier
  - Group manager (GM)

Motivation

- Why GSS on constrained devices?
- Scenarios
  - Prove the age of majority without revealing date of birth
  - Prove that you are in possession of a valid driving license
  - Anonymous entrance control
  - Travel anonymously within the EU?
- So where's the problem?
- GSS are based on a complex mathematical concept

Pairing-Based Cryptography (PBC)

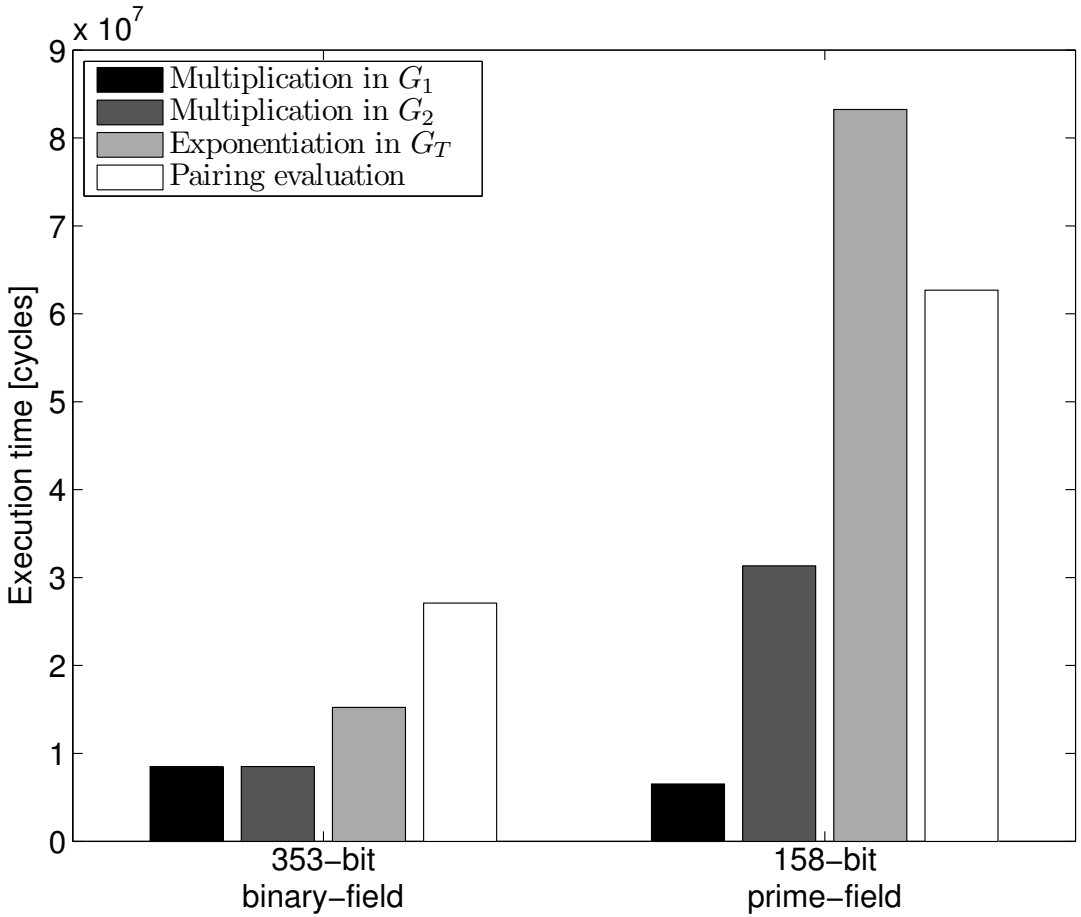
- $\mathbb{G}_1 = \langle g_1 \rangle$, $\mathbb{G}_2 = \langle g_2 \rangle$, and $\mathbb{G}_T$ are cyclic groups
- $\mathbb{G}_1$ points on $E(\mathbb{F}_q)$
- $\mathbb{G}_2$ points on $E(\mathbb{F}_{q^k})$
- $\mathbb{G}_T$ is a subgroup of $\mathbb{F}^*_{q^k}$
- Bilinear map: $e(u^a, v^b) = e(u, v)^{ab}$, $u \in \mathbb{G}_1$, $v \in \mathbb{G}_2$, and $a, b \in \mathbb{Z}^*_n$
- **Type 1:** $\mathbb{G}_1 = \mathbb{G}_2$
- **Type 3:** $\mathbb{G}_1 \neq \mathbb{G}_2$, no efficiently computable isomorphism
- PBC is a complex mathematical concept
- Implementations are available, *e.g.*, RELIC [AG]

Comparison of Group-Signature Schemes

- Investigated four schemes [BBS04, BS04, DP06, HLC$^+$11]
- Hide a user's certificate within a group signature - GM can decrypt the certificate
- Different ...
  - Mathematical assumptions
  - Types of pairings
  - Revocation mechanisms (in case of misbehavior)
    - Perform setup phase again
    - Private-key update
    - Verifier-local revocation (complicated opening mechanism)
  - Number of group operations
- BBS [BBS04], Type 1 pairings
- HLCCN [HLC$^+$11, Int13], Type 3 pairings
- Both types of pairings are implemented in RELIC

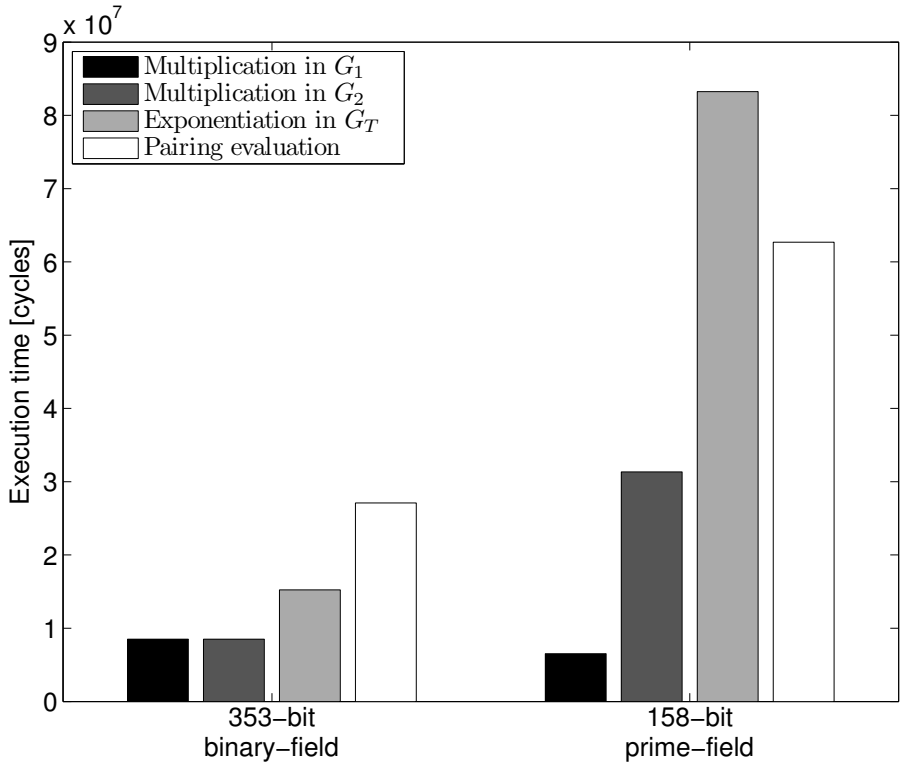
Implementation and Performance

- RELIC [AG]
  - η_T (eta-t) pairing over $E(\mathbb{F}_{2^{353}})$
  - optimal-ate pairing over 158-bit BN-curve $E(\mathbb{F}_p)$



High-Level Performance Optimization?

- Computation of $e(u, v)^a$, $u \in \mathbb{G}_1$, $v \in \mathbb{G}_2$, $a \in \mathbb{Z}$
  - E in $\mathbb{G}_1$, and evaluate pairing: $e(u^a, v)$
  - E in $\mathbb{G}_2$, and evaluate pairing: $e(u, v^a)$
  - E in $\mathbb{G}_T$: $e(u, v)^a$
- So, which one is the best?



Implementation of Schemes

- BBS
  - Use cached pairings
- HLCCN

$$R_2 = e(D_2, h_1)^{r_x} e(w, h_\theta)^{-r_\alpha} e(w, h_1)^{-r_\gamma} e(g_2, h_1)^{r_y}$$

$$R_2 = e(D_2^{r_x} w^{-r_\gamma} g_2^{r_y}, h_1) e(w^{-r_\alpha}, h_\theta)$$

Consequence?

$$R_2 = \boxed{e(D_2, h_1)}^{r_x} \boxed{e(w, h_\theta)}^{-r_\alpha} \boxed{e(w, h_1)}^{-r_\gamma} \boxed{e(g_2, h_1)}^{r_y}$$

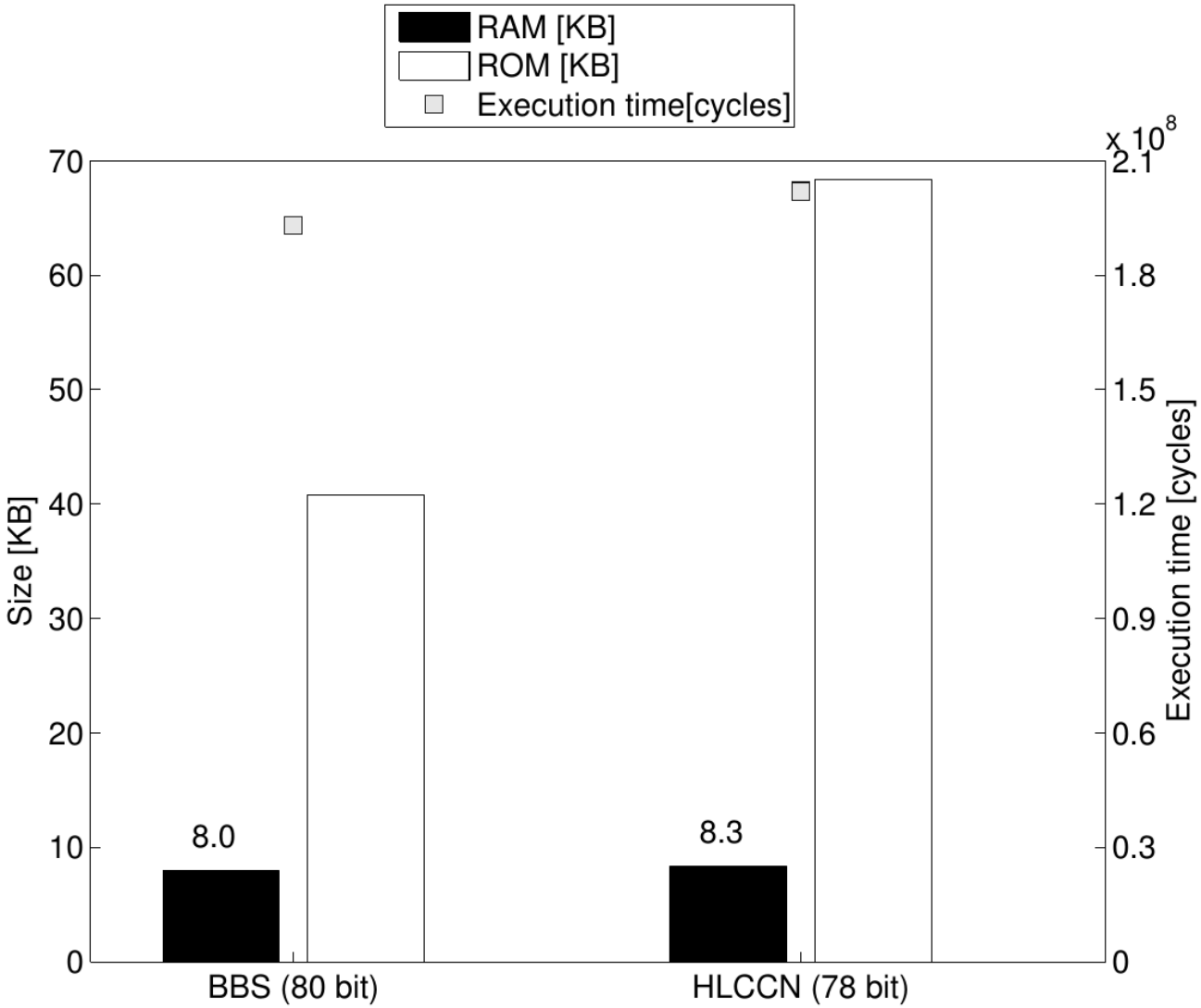
4 E in $\mathbb{G}_T$	$4 \times 83.2 \cdot 10^6$
Σ	$332.8 \cdot 10^6$

$$R_2 = e(D_2^{r_x} w^{-r_\gamma} g_2^{r_y}, h_1) e(w^{-r_\alpha}, h_\theta)$$

4 M in $\mathbb{G}_1$	$4 \times 6.5 \cdot 10^6$
2 pairings	$2 \times 62.7 \cdot 10^6$
Σ	$151.4 \cdot 10^6$

x2

Overall Performance



Conclusion

- Type 1 pairings are considered insecure [GGMZ13, Jou13, Sma]
- Type 3 pairings seem to be the desirable choice
- Top-down approach instead of bottom-up approach
- Cached pairings vs. evaluation of pairings
  - Speedup of factor of 2
- 6 seconds on a 32 MHz microcontroller
- Future work
  - Instruction-set extensions
  - Secure delegation

Group-Signature Schemes on Constrained Devices

Raphael Spreitzer and Jörn-Marc Schmidt

Institute for Applied Information Processing and Communications (IAIK)
Graz University of Technology
Inffeldgasse 16a, A-8010 Graz, Austria

raphael.spreitzer@iaik.tugraz.at

Bibliography I

[AG] D. F. Aranha and C. P. L. Gouvêa.
RELIC is an Efficient LIbrary for Cryptography.
http://code.google.com/p/relic-toolkit/.

[BBS04] Dan Boneh, Xavier Boyen, and Hovav Shacham.
Short Group Signatures.
In Matt Franklin, editor, *Advances in Cryptology - CRYPTO 2004*, volume 3152 of *LNCS*, pages 41–55. Springer Berlin Heidelberg, 2004.

[BS04] Dan Boneh and Hovav Shacham.
Group Signatures with Verifier-Local Revocation.
In *Proceedings of the 11th ACM conference on Computer and communications security*, CCS '04, pages 168–177, New York, NY, USA, 2004. ACM.

[CvH91] David Chaum and Eugène van Heyst.
Group Signatures.
In DonaldW. Davies, editor, *Advances in Cryptology - EUROCRYPT '91*, volume 547 of *LNCS*, pages 257–265. Springer Berlin Heidelberg, 1991.

[DP06] Cécile Delerablée and David Pointcheval.
Dynamic Fully Anonymous Short Group Signatures.
In PhongQ. Nguyen, editor, *VIETCRYPT*, volume 4341 of *LNCS*, pages 193–210, 2006.

[GGMZ13] Faruk Göloğlu, Robert Granger, Gary McGuire, and Jens Zumbrägel.
On the Function Field Sieve and the Impact of Higher Splitting Probabilities: Application to Discrete Logarithms in $\mathbb{F}_{2^{1971}}$ and $\mathbb{F}_{2^{3164}}$.
Cryptology ePrint Archive, Report 2013/074, 2013.
http://eprint.iacr.org/.

[HLC+11] Jung Yeon Hwang, Sokjoon Lee, Byung-Ho Chung, Hyun Sook Cho, and DaeHun Nyang.
Short Group Signatures with Controllable Linkability.
In *Proceedings of the 2011 Workshop on Lightweight Security & Privacy: Devices, Protocols, and Applications*, LIGHTSEC '11, pages 44–52, Washington, DC, USA, 2011. IEEE Computer Society.

Bibliography II

[Int13] Internationl Organization for Standardization (ISO).
ISO/IEC 20008-2: Information technology - Security techniques - Anonymous digital signatures - Part 2: Mechanisms using a group public key, November 2013.

[Jou13] Antoine Joux.
A new index calculus algorithm with complexity $L(1/4 + o(1))$ in very small characteristic.
Cryptology ePrint Archive, Report 2013/095, 2013.
http://eprint.iacr.org/.

[Sma] Niegel Smart.
Discrete Logarithms.
http://bristolcrypto.blogspot.co.uk/2013/02/discrete-logarithms.html.