Improved Algebraic MACs and Practical Keyed-Verification Anonymous Credentials

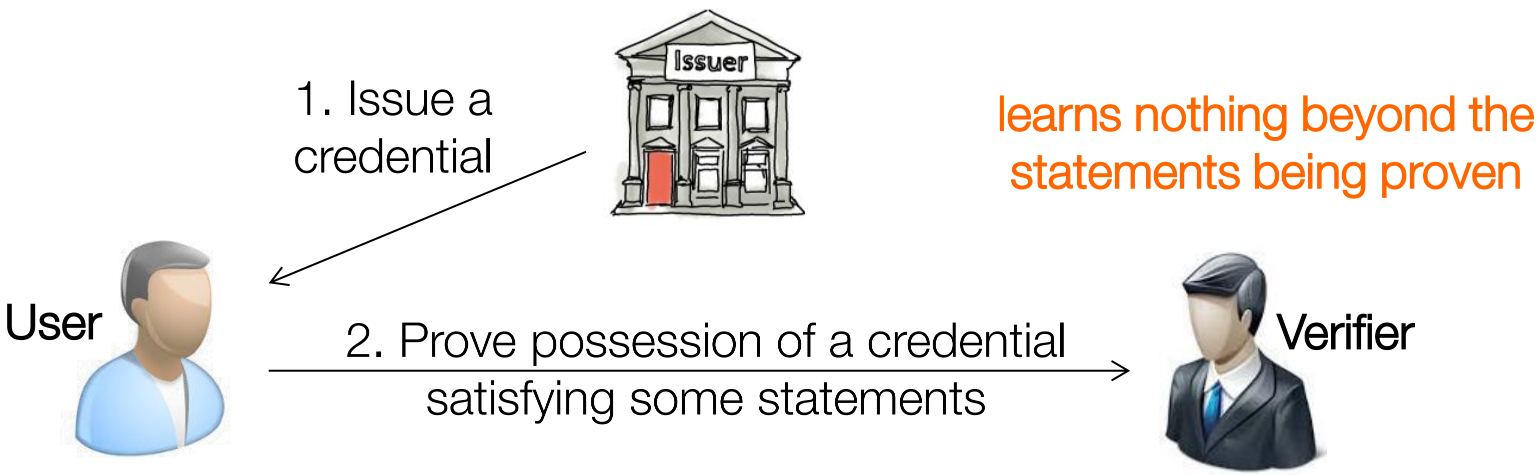
Amira Barki, Solenn Brunet, Nicolas Desmoulins and Jacques Traoré

August 11th, 2016
Selected Areas in Cryptography – SAC 2016

orange™

Anonymous Credentials Systems

- Introduced by Chaum [CommACM'85]



- Applications:
  (a service provider only needs to know that a user is legitimate)
  - e-cash systems
  - public transport
  - electronic toll

Previous Work

- No multi-show unlinkability:
  - Microsoft's U-Prove, not formally proven secure
  - Baldimsti and Lysyanskaya [CCS'13]
  - Fuchsbauer, Hanser and Slamanig [CRYPTO'15]
- Unsuitable for constrained devices (large RSA parameters):
  - IBM's Identity Mixer (Idemix)
- Require pairings on the prover's side:
  - Camenisch and Lysyanskaya [CRYPTO'04]
  - Akagi, Manabe and Okamoto [FC'08]
  - Camenisch, Dubovitskaya, Haralambiev and Kohlweiss [ASIACRYPT'15]

Keyed-Verification Anonymous Credentials (KVAC)
Chase, Meiklejohn and Zaverucha [CCS'14]

- Particular type of Anonymous Credentials: KVAC
  - the credentials issuer and the verifier share a set of secret keys
- Advantages:
  - more efficient: rely on symmetric key primitives (algebraic MACs)
  - possible switch between public key and keyed-verification anonymous credentials
- Shortcomings:
  - require as many secret keys as attributes
  - presentation proof linear in the number of group elements

Goals

- A secure anonymous credentials system that combines:

  1. multi-show unlinkability
  2. suitability for resource constrained environments
  3. no pairing computations, or even computations in either $\mathbb{G}_2$ or $\mathbb{G}_T$, on the prover side
     - current SIM cards cannot handle them
  4. efficient presentation proof
     - complexity O(1) in the number of group elements

$\mathrm{MAC}_{\mathrm{BB}}$: A new algebraic MAC scheme
based on Boneh-Boyen's signature scheme

- Setup(1^k): Generate $pp = (\mathbb{G}, p, h, g_0, g_1, g)$ such that
  - $\mathbb{G}$ cylic group of prime order p, where DDH is hard
  - h, g_0, g_1, g four random generators of $\mathbb{G}$
- KeyGen(pp):
  - issuer's private key $y \in_R \mathbb{Z}_p$
  - optionally, the public key $\mathrm{Y} = \mathrm{g}_0^{\mathrm{y}}$
- MAC(m, y):
  - choose r, s at random
  - generate $\tau = (A, r, s)$ where $A = (g_1^m g^s h)^{\frac{1}{y+r}}$
- Verify$(m, (A, r, s), y)$: check if $(g_1^m g^s h)^{\frac{1}{y+r}} \stackrel{?}{=} A$

sUF-CMVA under the gap q-SDH assumption

$\mathrm{MAC}_{\mathrm{BB}}^{n}$: Extension to n messages

- Issuer and Verifier share a single secret key
- MAC$((m_1, \dots, m_n), y)$: $\tau = (A, r, s)$

  where r, s are chosen at random, $A = \left(g_1^{m_1} g_2^{m_2} \dots g_n^{m_n} g^s h\right)^{\frac{1}{y+r}}$
- Verify$(m, (A, r, s), y)$: $\left(g_1^{m_1} g_2^{m_2} \dots g_n^{m_n} g^s h\right)^{\frac{1}{y+r}} \stackrel{?}{=} A$

sUF-CMVA

+ MACs publicly verifiable:

Let $B = g_1^{m_1} g_2^{m_2} \dots g_n^{m_n} g^s h \cdot A^{-r} = A^y$.

No pairings: to convince a verifier, the issuer provides a ZKPK

$$\pi = \{\, \gamma : B = A^{\gamma} \wedge Y = g_0^{\gamma} \}$$

Security Requirements of a Keyed-Verification Anonymous Credentials (KVAC)

- Correctness

  → issued credentials are valid and their verification with respect to the associated attributes succeed

- Unforgeability

  → impossible to forge a valid proof of possession of a credential

- Anonymity

  → the presentation proof reveals nothing aside from the statement being proven

Our KVAC System: Set-Up and Key Generation

- **Set-Up**: Generate $pp = (\mathbb{G}, p, g_1, g_2, \dots, g_n, g, h, g_0, f)$ such that
  - $\mathbb{G}$ cyclic group of prime order p where DDH is hard
  - $(h, g, g_0, \{g_i\}_{i=1}^{n}, f)$ random generators of $\mathbb{G}$

  such that each g_i is associated with a specific type of attributes

- **Key Generation**:
  - Issuer: private key $y \in_R \mathbb{Z}_p$ associated to the public key $Y = g_0^y$
  - User: private key sk_u and public key pk_u (for authentication)

Blind Issuance of an Anonymous Credential

$\pi_1 = \{ \alpha_1, \ldots, \alpha_{n+1} : C_m = g_1^{\alpha_1} \ldots g_n^{\alpha_n} g^{\alpha_{n+1}} \}$

$m_1, \ldots, m_n$

sk_u

private key y

User

$C_m = g_1^{m_1} \ldots g_n^{m_n} g^s \;, \pi_1$

$(A, r, s') \;, \pi_2$

where $A = \left(C_m \cdot g^{s'} \cdot h\right)^{\frac{1}{y+r}}$

$\boldsymbol{\sigma = (A, r, s_u, \widetilde{C}_m)}$

where

$s_u = s + s'$

$\tilde{C}_m = C_m \, g^{s'} h$

$\pi_2 = \{ \gamma : Y = g_0^{\gamma} \wedge C_m \cdot g^{s'} h \cdot A^{-r} = A^{\gamma} \}$

Credential Presentation

$$\pi_3 = \{\alpha, \beta, \gamma, \delta_1, \dots, \delta_{n+1}, \gamma, \theta : E \cdot h^{-1} = g_1^{\delta_1} \dots g_n^{\delta_n} g^{\delta_{n+1}} B_0^{\lambda} f^{\beta} \wedge E = C^{\alpha} f^{\beta} \wedge \; C = E^{\theta} f^{\gamma}\}$$

$m_1, \dots, m_n$

sk_u

User

$B_0 = A^l, C = \tilde{C}_m^l B_0^{-r}$

$E = C^{\frac{1}{l}} f^t$, π_3

private key y

Verifier

Anonymous Credential

$\boldsymbol{\sigma = (A, r, s_u, \tilde{C}_m)}$

No pairings

Verification

1. Compute $C' = B_0^y$
2. Check if $C' = C$
3. Check π_3

From Keyed-Verification Anonymous Credential to Public Key Anonymous Credential

- Set-up: $pp = (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, p, g_1, \ldots g_n, g, h, g_0, f, \tilde{g}_0, e)$ such that
  - $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ cyclic groups of prime order p where DDH is hard
  - $(h, g, g_0, \{g_i\}_{i=1}^{n}, f)$ random generators of $\mathbb{G}_1$
  - $\tilde{g}_0$ random generator of $\mathbb{G}_2$
- Key Generation: additional issuer's public key $W = \tilde{g}_0^{y}$
- Blind Issuance: does not change (no pairings)
- Credential Presentation:

private key y unknown to the verifier $\Longrightarrow$ pairings (only on the verifier side)

$\rightarrow$ the verifier checks if $e(C, \tilde{g}_0) = e(B_0, W)$

Efficiency Comparison: Presentation Proof Computational Cost

Scheme	Credential size	Number of exponentiations
$U\text{–}Prove$	$1024s$	$2c$ 2-exp, 1 $(n$-r+1)-exp
$Idemix$	5369	1 1-exp(2048), c 2-exp(256,2046), c 2-exp(592,2385), 1 $(n$-r+2)-exp(456,3060,592,…,592)
$\mathrm{MAC_{GGM}}$	512	3 1-exp, 2 $(n$-$r)$ 2-exp, 1 $(n$-r+1)-exp
$\mathrm{MAC_{DDH}}$	1024	6 1-exp, 2 $(n$-$r+1)$ 2-exp, 2 $(n$-r+1)-exp
$\mathbf{MAC_{BB}^{n}}$	1024	1 1-exp, 4 2-exp, 1 $(n-r+3)$-exp

Credential size and Computational cost ($\boldsymbol{n}$ attributes with $\boldsymbol{c}$ unrevealed)

→ Competitive with U-Prove (no multi-show unlinkability)
More efficient when $c > 4$

Implementation Benchmarks: Credential Presentation

- User's side: NFC enabled SIM card* + NFC-equipped smartphone

All computations are performed by the SIM card

Off-line part (card) Battery-On : (1352-1392) 1378 ms			
Total On-line part			
Battery-On		Battery-Off	
y known	y unknown	y known	y unknown
(84-100) 88 ms	(86-103) 93 ms	(126-137) 128 ms	(128-141) 133 ms

Timings results ((min-max) average) for 3 attributes with 1 unrevealed

*Global platform 2.2 compliant

Conclusion

- Algebraic MAC scheme based on Boneh Boyen's signature scheme
  - a single secret key
  - proven strong UF-CMVA
- An efficient keyed-verification anonymous credentials system
  - provides required security properties (unforgeability and anonymity)
  - presentation proof of complexity O(1) in the number of group elements
  - \+ easily turned into an efficient public key anonymous credential system
- Efficient and suitable for SIM cards

  88 ms for a credential with 3 attributes, one of which is undisclosed

Thank you for your attention

Questions ?