

Enhanced Privacy ID using Bilinear Pairing

Ernie Brickell and Jiangtao Li
Intel Corporation

NIST Identity Based Encryption Workshop, June 2008

1

Intel Corporation



Outline

- Motivation for Enhanced Privacy ID (EPID)
- Overview of EPID
- EPID from Bilinear Maps

2

Intel Corporation



Overview of EPID

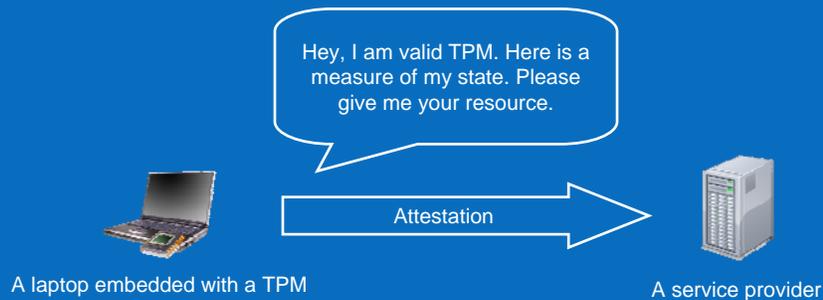
- EPID is a crypto protocol that provides proof of membership in a group with properties:
 - Anonymous
 - Unlinkable (optional)
 - Issuer does not keep a database of all members' private keys
 - Revocable if private key is revealed
 - Proof that private key not used in some specific previous transaction
 - Auditable revocation list
- EPID is a Direct Anonymous Attestation (DAA) scheme with enhanced revocation capabilities
 - DAA has been adopted in TCG Trusted Platform Module (TPM) Spec v1.2
- EPID is different from a group signatures scheme in that
 - Nobody cannot open a group signature and find out who signs it
 - Member's privacy is intact unless he has been revoked

3

Intel Corporation



Application of EPID: Anonymous Attestation



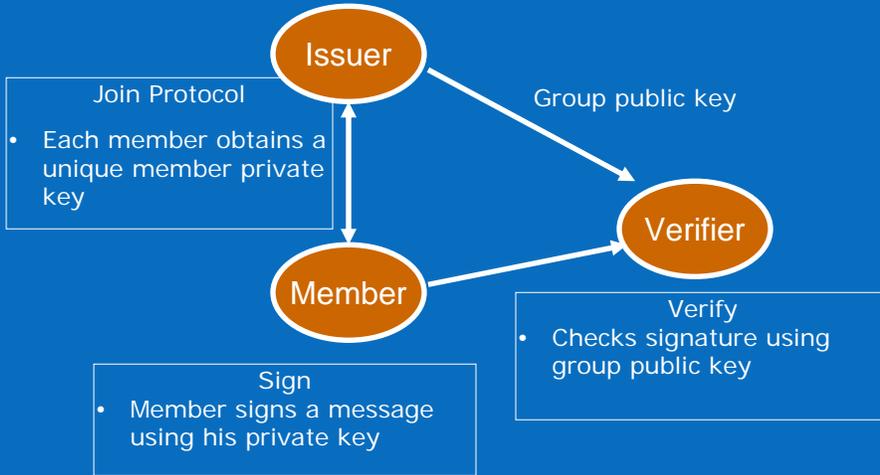
EPID can be used for authentication and attestation while preserving the privacy of the TPM

4

Intel Corporation



Basic EPID Scheme



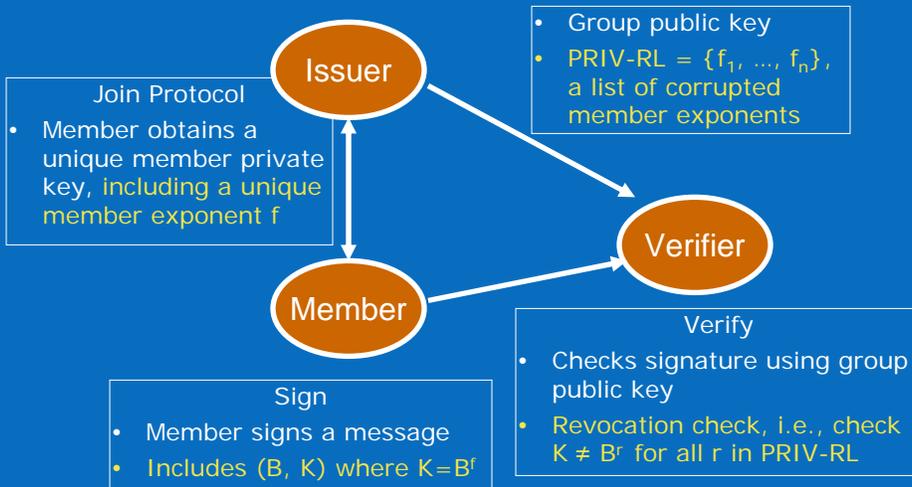
Let us temporarily put aside the revocation issue

5

Intel Corporation



Private Key Based Revocation in EPID



B is called Base, K is called Pseudonym

6

Intel Corporation



Random Base or Name Base

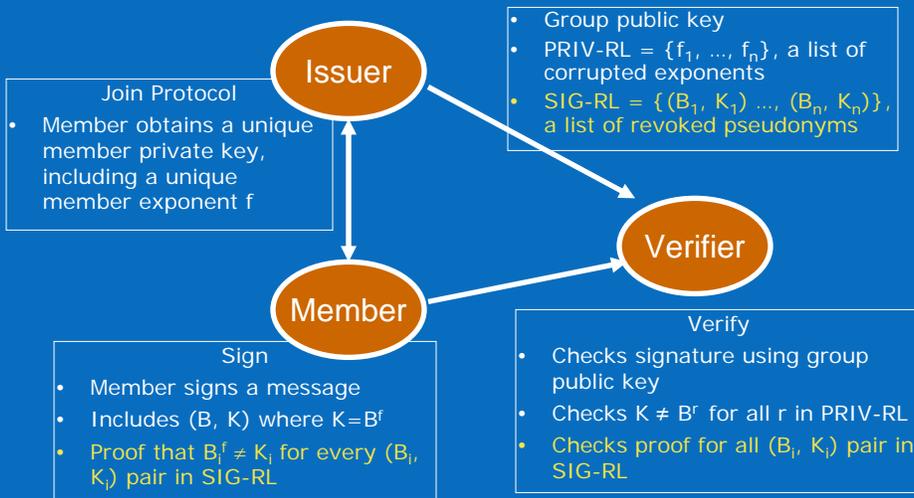
- In **random base option**, B is chosen randomly each time by the member
 - Given (B_1, K_1) and (B_2, K_2) from two signatures, where $K_1 = B_1^{f_1}$ and $K_2 = B_2^{f_2}$, if B_1 and B_2 are chosen randomly, the verifier cannot tell whether $f_1 = f_2$ under the DDH assumption
 - EPID signatures are unlinkable in random base option
- In **name base option**, B derives from the verifier's basename
 - E.g., $B = \text{Hash}(\text{verifier's basename})$
 - K becomes a pseudonym for the member w.r.t. a verifier
 - EPID signatures are no longer unlinkable to a verifier
 - We sometimes use this option to prevent abuse of privacy

7

Intel Corporation



Signature Based Revocation in EPID



$$PK\{ (f) : K = B^f \text{ and } K_i \neq B_i^{f_i} \}$$

8

Intel Corporation



Efficiency of Revocation Methods

- Private-key based revocation
 - The member does not need to do anything besides computing (B, K)
 - The verifier needs to compute B^f (1 EXP) for each f in PRIV-RL
 - For name base option, the verifier can pre-compute all B^f
- Signature based revocation
 - We could use Camenisch-Shoup non-equality proof
 - For each item in SIG-RL, the member needs to perform ~ 3 EXP
 - For each item in SIG-RL, the verifier needs to perform ~ 2 EXP
 - The member can pre-compute non-revoked proofs without knowledge of message to be signed
- We expect the revocation lists to be small
 - We only need to revoke if (hardware) attacks happen
 - E.g., change ownership of a TPM will not result in a revocation – it is still a valid TPM

9

Intel Corporation



Privacy and Revocation Properties of Schemes

	PKI	DAA with Random Base	DAA with Name Base	EPID
Unique Public Key	Yes	No	No	No
Unique Private Key	Yes	Yes	Yes	Yes
Anonymous	No	Yes	Yes	Yes
Unlinkable	No	Yes	No	Yes
Check for revealed private key	Yes	Yes	Yes	Yes
Revoke the signer of a signature	Yes	No	Yes	Yes

10

Intel Corporation



EPID Scheme from Strong RSA Assumption

- Protocol builds on top on
 - Camenisch and Lysyanskaya's signature scheme
 - Brickell, Camenisch, Chen's DAA scheme
- Properties of this EPID Protocol
 - Using 2048-bit RSA modulus
 - Size of a member private key = 670 bytes
 - Size of a EPID signature ~ 2800 bytes
- Security based on
 - Strong RSA Assumption
 - Decisional Diffie-Hellman Assumption

11

Intel Corporation



EPID Scheme from Bilinear Pairing

- Protocol builds on top on
 - Boneh, Boyen, Shacham's group signature scheme
 - Boneh and Shacham's group signature scheme
- Properties of this EPID Protocol
 - Using 256-bit elliptic curves
 - Size of a member private key = 96 bytes
 - Size of a EPID signature = 512 bytes
- Security based on
 - Strong Diffie-Hellman Assumption on Bilinear Groups
 - Decision Linear Assumption
 - Decisional Diffie-Hellman Assumption

12

Intel Corporation



EPID Scheme from Bilinear Maps in Details

- Issuer setup
 - Chooses a bilinear group pair G_1 and G_2 of prime order p with generators g_1 and g_2 , respectively
 - Let $e: G_1 \times G_2 \rightarrow G_T$ be a computable bilinear map function
 - Chooses a group G_3 of prime order p with generator g_3
 - Chooses a random $\gamma \in \mathbb{Z}_p$, and computes $w = g_2^\gamma$
 - The group public key is $(p, G_1, G_2, G_3, G_T, w)$
 - The issuer's private key is γ
- Join
 - The issuer chooses a random $f \in \mathbb{Z}_p$
 - The issuer computes $A = g_1^{1/(\gamma+f)}$
 - The (A, f) pair is the member's private key
 - Observe that $e(A, w \cdot g_2^f) = e(A, g_2)^{\gamma+f} = e(g_1, g_2)$

13

Intel Corporation



EPID Scheme from Bilinear Maps in Details (cont.)

- Sign
 - If random base option, the member chooses B from G_3 randomly
 - If name base option, the member derives B from the verifier's basename
 - The member computes $K = B^f$
 - The member computes $PK\{ (A, f) : e(A, w \cdot g_2^f) = e(g_1, g_2) \text{ and } K = B^f \}$
 - The member computes $PK\{ (f) : K = B^f \text{ and } K_i \neq B_i^f \}$ for each (B_i, K_i) pair in SIG-RL
- Verify
 - If random base option, verifies that B is an element in G_3
 - If name base option, derives B from the verifier's basename
 - Verifies that K is an element in G_3
 - Verifies $PK\{ (A, f) : e(A, w \cdot g_2^f) = e(g_1, g_2) \text{ and } K = B^f \}$
 - Verifies that $K \neq B_i^f$ for each f_i in PRIV-RL
 - Verifies $PK\{ (f) : K = B^f \text{ and } K_i \neq B_i^f \}$ for each (B_i, K_i) pair in SIG-RL

14

Intel Corporation



Summary

- For any transaction in which identity is not explicitly required for the transaction, then EPID can be used to provide same level of security and with privacy
- Example: EPID can be used instead of PKI for any use of PKI in which verifier needs to know only “is this request from an authorized party”

