Anonymous Proxy Signatures

G. Fuchsbauer  D. Pointcheval
École normale supérieure
SCN ’08
09/11/2008

Application: GRID computing
User authenticates herself and starts process which needs to authenticate to resources / start subprocesses
⇒ Delegation and re-delegation of signing rights
No need to know that it was not the user herself to be authenticated

Our Results
- Algorithm specifications
- Security definitions
- Proof of concept: existence assuming trapdoor permutations

Anonymous Consecutive Delegation of Signing Rights

Delegation A delegator delegates his signing rights to a proxy signer (or delegatee) who can then sign on the delegator’s behalf

Consecutiveness A delegatee may re-delegate the received signing rights ⇒ intermediate delegators

Anonymity All intermediate delegators and the proxy signer remain anonymous

After verifying a proxy signature one knows that someone entitled signed but nothing more

Relation to Other Primitives

Anonymous proxy signatures are a generalization of

Proxy signatures (consecutive delegation) formalized by [BPW03]

Group signatures (anonymity) formalized by [BMW03, BSZ05]
  - dynamic (users can join after setup of group)
  - hierarchical (tree structure by consecutive delegations) [TW05]

and satisfy the respective security notions
Group Signatures

Group public key: \( pk \)

- **Issuer** (\( ik \))
- **Opener** (\( ok \))
- **Group members** (\( sk_i \))
- **Reg**
- **Open**
- **sign**
- **msg**
- **\( \sigma \)**

Security Definitions for (Dynamic) Group Signatures

- **Anonymity**: no one except the opener can tell who produced a signature
- **Traceability**: every valid signature can be traced to its signer by the opener
- **Non-Frameability**: no one can produce a signature that opens to a member who did not sign

Algorithms for (Dynamic) Group Signatures

- **Setup**: produces group public key, issuing key, opening key
- **Reg**: registers new members joining the group using the issuing key
- **Sig**: enables a group member to sign on behalf of the group
- **Ver**: checks validity of a group signature using the group public key
- **Open**: reveals the signer's identity using the opening key

Proxy Signatures

- **Delegator** (\( pk_D \))
- **Delegatee/Signer**
- **Verify** (\( pk_D, \text{msg}, \sigma \))
Proxy Signatures, Consecutive Delegations

Delegator (pk_D)

Delegator 2

Delegator 3

ANONYMOUS

Proxy Signer

Opener (ok_D)

σ

open

Example: Redelegation of Reduced Task Set

User x (pk_x, sk_x)

Del(sk_x, (2, 5, 7), pk_y) → warr

User y (pk_y, sk_y)

Del(sk_y, warr, (5, 7), pk_x) → warr'

User z (pk_y, sk_y)

PSig(sk_y, warr', 5, M) → σ

PVer(pk_y, 5, M, σ) → 1

Open(ok_x, 5, M, σ) → {pk_y, pk_z}

Tasks

Delegation by Certificate

Delegator signs a warrant containing the proxy’s public key pk_P
Proxy signs message with her own signing key
⇒ Verify signature on warrant (w.r.t. pk_D) and message (w.r.t. pk_P).

Delegation of Tasks

- possibility to delegate rights only for certain set of tasks
- re-delegate rights for restricted set of tasks

Delegation of TList, a set of natural numbers representing tasks

Algorithms of Anonymous Proxy Signature Scheme PS

Issuer (ik) → S_tup → pp, ik, ok
sk_x, [warr→x,] TList, pk_y → D_I → warr_[x→y]
ok_x, task, M, σ → Op n → a list of users or ⊥ (failure)

pk_y, task, M, σ → PVer → b ∈ {0, 1}

pk_x, task, M, σ → Op n → a list of users or ⊥ (failure)
Security for Anonymous Proxy Signatures

**Anonymity** intermediate delegators and proxy signer remain anonymous

**BUT:** the number of delegations may not remain hidden (if no restriction on number of delegations)

**Traceability** every valid signature can be traced to its intermediate delegators and proxy signer

**Non-Frameability** no one can produce a signature that, when opened, wrongfully reveals a delegator or signer

---

**Anonymity I**

Idea:
Adversary controls users and issuer
produces 2 warrants
one of them used to sign
Adversary must decide which one

Restrictions:
- $U_1$ must be registered with the opener
- both warrants correctly formed
- both delegation chains of same length

---

**Anonymity II**

$\text{Exp}^{\text{anon}}_{\text{PS}, A}(\lambda)$

$pp, ik, ock \leftarrow \text{Setup}(\lambda)$

$pk, M, \text{task}$

$(sk_0, \text{warr}_0)$

$(sk_1, \text{warr}_1)$

$\sigma \leftarrow \text{PSig}(sk_b, \text{warr}_b, \text{task}, M)$

$\sigma' \leftarrow \text{PSig}(sk_{b'}, \text{warr}_{b'}, \text{task}, M)$

Adversary

Oracle:
- USndToO
- ISndToO
- OK
- Op

$b, b' \leftarrow \{0, 1\}$

$\Pr[\text{Exp}^{\text{anon}}_{\text{PS}, A}(\lambda) = 1] - \frac{1}{2} = \text{negl}(\lambda)$

---

**Anonymity III**

The experiment $\text{Exp}^{\text{anon}}_{\text{PS}, A}(\lambda)$ returns 1 if $b = b'$
no queries $\text{OK}(pk)$ and $\text{Open}(pk, \text{task}, M, \sigma)$ made

Definition
A proxy signature scheme is anonymous if for all p.p.t. adversaries $A$

$\Pr[\text{Exp}^{\text{anon}}_{\text{PS}, A}(\lambda) = 1] - \frac{1}{2} = \text{negl}(\lambda)$
Traceability I

Idea:
Adversary can corrupt users and opener (which follows the protocol) gets SndToI and SndToO oracles for Reg that return a transcript between them and opening key must produce signature that is valid but not openable

Definition
A proxy signature scheme is **traceable** if for all p.p.t. adversaries $A$

$$\Pr[\text{Exp}_{\text{trace}}^{\text{PSA}}(\lambda) = 1] = \text{negl}(\lambda)$$

Non-Frameability I

$$\text{Exp}^{\text{n-frame}}_{\text{PSA}}(\lambda)$$

- $pp, ik, ock \leftarrow S \text{setup}(\lambda)$
- $(pp, ik, ock, ok)$
  - $\text{Adversary}$
  - $\text{Oracles:}$
    - SndToU
    - OSndToU
    - SK
    - Del
    - PSig

The experiment keeps a list $HU$ of honest users with whom $A$ interacts via SndToU and OSndToU. $SK$ returns the secret key of a user after deleting her from $HU$.

Traceability II

$$\text{Exp}^{\text{trace}}_{\text{PSA}}(\lambda)$$

- $pp, ik, ock \leftarrow S \text{setup}(\lambda)$
- $(pp, pk_1, task, M, \sigma)$
- $\text{Adversary}$
- $\text{Oracles:}$
  - SndToI
  - SndToO

$r \text{ turn } 1 \text{ if}$
- $\text{PV}(pk, task, M, \sigma) = 1$ and
- $\text{Op}(\text{OK}(pk), task, M, \sigma) = \bot$
- $\text{else return } 0$

The experiment returns 1 if $\sigma$ is valid and its opening reveals either a delegation by an honest user which was not queried via Del or an honest proxy signer who was not queried via PSig

Non-Frameability II

Definition
A proxy signature scheme is **non-frameable** if for all p.p.t. adversaries $A$

$$\Pr[\text{Exp}_{\text{n-frame}}^{\text{PSA}}(\lambda) = 1] = \text{negl}(\lambda)$$
Generic Construction

- Digital signatures (EUF-CMA)
- Public-key encryption (IND-CCA)
- NIZK (simulation sound)

(follow from trapdoor permutations)

Conclusion

- Defined new primitive encompassing group and proxy signatures (satisfies rigorous security notions of both)
- Non-frameable dynamic hierarchical group signatures

Open Problem

- Efficient implementation