New Constructions and Applications of Trapdoor DDH Groups

Yannick Seurin

ANSSI, France

March 1, PKC 2013

Introduction: CDH versus DDH

- group $\mathbb{G}$, element $G \in \mathbb{G}$ of large order
- CDH problem: given $X = G^x$ and $Y = G^y$, compute G^{xy}
- DDH problem: distinguish (G^x, G^y, G^{xy}) and (G^x, G^y, G^z)
- usual situations in cryptographic groups:
  1. CDH and DDH are both (presumably) hard
     $\rightarrow$ e.g. prime order subgroup of $\mathbb{Z}_p^*$
  2. CDH is (presumably) hard and DDH is universally easy
     $\rightarrow$ pairing groups

Introduction: CDH versus DDH

- group $\mathbb{G}$, element $G \in \mathbb{G}$ of large order
- CDH problem: given $X = G^x$ and $Y = G^y$, compute G^{xy}
- DDH problem: distinguish (G^x, G^y, G^{xy}) and (G^x, G^y, G^z)
- usual situations in cryptographic groups:
  1. CDH and DDH are both (presumably) hard
     $\rightarrow$ e.g. prime order subgroup of $\mathbb{Z}_p^*$
  2. CDH is (presumably) hard and DDH is universally easy
     $\rightarrow$ pairing groups

Introduction: CDH versus DDH

- group $\mathbb{G}$, element $G \in \mathbb{G}$ of large order
- CDH problem: given $X = G^x$ and $Y = G^y$, compute G^{xy}
- DDH problem: distinguish (G^x, G^y, G^{xy}) and (G^x, G^y, G^z)
- usual situations in cryptographic groups:
  1. CDH and DDH are both (presumably) hard
     $\rightarrow$ e.g. prime order subgroup of $\mathbb{Z}_p^*$
  2. CDH is (presumably) hard and DDH is universally easy
     $\rightarrow$ pairing groups

Introduction: trapdoor DDH groups

Trapdoor DDH groups (TDDH groups):

- lies somewhere between cases 1 and 2:
  $\rightarrow$ CDH is hard, while DDH is hard unless one has some trapdoor τ
- introduced by Dent and Galbraith [DG06]
- very few constructions (hidden pairing construction by [DG06])
- very few applications:
  - simple identification scheme [DG06]
  - statistically hiding sets [PX09]

Introduction: trapdoor DDH groups

Trapdoor DDH groups (TDDH groups):

- lies somewhere between cases 1 and 2:
  $\rightarrow$ CDH is hard, while DDH is hard unless one has some trapdoor τ
- introduced by Dent and Galbraith [DG06]
- very few constructions (hidden pairing construction by [DG06])
- very few applications:
  - simple identification scheme [DG06]
  - statistically hiding sets [PX09]

Introduction: trapdoor DDH groups

Trapdoor DDH groups (TDDH groups):

- lies somewhere between cases 1 and 2:
  $\rightarrow$ CDH is hard, while DDH is hard unless one has some trapdoor τ
- introduced by Dent and Galbraith [DG06]
- very few constructions (hidden pairing construction by [DG06])
- very few applications:
  - simple identification scheme [DG06]
  - statistically hiding sets [PX09]

Introduction: trapdoor DDH groups

Trapdoor DDH groups (TDDH groups):

- lies somewhere between cases 1 and 2:
  $\rightarrow$ CDH is hard, while DDH is hard unless one has some trapdoor τ
- introduced by Dent and Galbraith [DG06]
- very few constructions (hidden pairing construction by [DG06])
- very few applications:
  - simple identification scheme [DG06]
  - statistically hiding sets [PX09]

In this paper

Our contributions:

- we slightly refine the original definition of trapdoor DDH groups by [DG06]
- we introduce static trapdoor DDH groups
- we give new constructions of trapdoor DDH and static trapdoor DDH groups based on standard assumptions
- we show that (static) trapdoor DDH groups give very simple constructions of convertible undeniable signature schemes

In this paper

Our contributions:

- we slightly refine the original definition of trapdoor DDH groups by [DG06]
- we introduce static trapdoor DDH groups
- we give new constructions of trapdoor DDH and static trapdoor DDH groups based on standard assumptions
- we show that (static) trapdoor DDH groups give very simple constructions of convertible undeniable signature schemes

In this paper

Our contributions:

- we slightly refine the original definition of trapdoor DDH groups by [DG06]
- we introduce static trapdoor DDH groups
- we give new constructions of trapdoor DDH and static trapdoor DDH groups based on standard assumptions
- we show that (static) trapdoor DDH groups give very simple constructions of convertible undeniable signature schemes

In this paper

Our contributions:

- we slightly refine the original definition of trapdoor DDH groups by [DG06]
- we introduce static trapdoor DDH groups
- we give new constructions of trapdoor DDH and static trapdoor DDH groups based on standard assumptions
- we show that (static) trapdoor DDH groups give very simple constructions of convertible undeniable signature schemes

Outline

1. Definition of Trapdoor DDH Groups
2. New Constructions of TDDH and Static TDDH Groups
   - A TDDH group based on composite residuosity
   - A static TDDH group based on RSA
   - A static TDDH group based on factoring
3. Application to Convertible Undeniable Signatures

Outline

TDDH group: definition

Trapdoor DDH group

$(\mathbb{G}, G, \tau) \leftarrow \textsc{GpGen}(1^k)$ is a trapdoor DDH group if:

1. the DDH problem is hard for $(\mathbb{G}, G)$ without the trapdoor τ
2. the CDH problem is hard even with the trapdoor τ
3. there is a distinguishing algorithm $\textsc{Solve}(X, Y, Z, \tau)$ which:
   - always accepts when (X, Y, Z) is a DDH tuple (completeness)
   - accepts with negligible probability for any adversarially generated $Z \leftarrow \mathcal{A}(X, Y)$ (soundness)

When $\textsc{Solve}$ always rejects on input a non-DDH tuple (X, Y, Z), we say that the TDDH group has perfect soundness.

TDDH group: definition

Trapdoor DDH group

$(\mathbb{G}, G, \tau) \leftarrow \textsc{GpGen}(1^k)$ is a trapdoor DDH group if:

1. the DDH problem is hard for $(\mathbb{G}, G)$ without the trapdoor τ
2. the CDH problem is hard even with the trapdoor τ
3. there is a distinguishing algorithm $\textsc{Solve}(X, Y, Z, \tau)$ which:
   - always accepts when (X, Y, Z) is a DDH tuple (completeness)
   - accepts with negligible probability for any adversarially generated $Z \leftarrow \mathcal{A}(X, Y)$ (soundness)

When $\textsc{Solve}$ always rejects on input a non-DDH tuple (X, Y, Z), we say that the TDDH group has perfect soundness.

TDDH group: definition

Trapdoor DDH group

$(\mathbb{G}, G, \tau) \leftarrow \textsc{GpGen}(1^k)$ is a trapdoor DDH group if:

1. the DDH problem is hard for $(\mathbb{G}, G)$ without the trapdoor τ
2. the CDH problem is hard even with the trapdoor τ
3. there is a distinguishing algorithm $\textsc{Solve}(X, Y, Z, \tau)$ which:
   - always accepts when (X, Y, Z) is a DDH tuple (completeness)
   - accepts with negligible probability for any adversarially generated $Z \leftarrow \mathcal{A}(X, Y)$ (soundness)

When $\textsc{Solve}$ always rejects on input a non-DDH tuple (X, Y, Z), we say that the TDDH group has perfect soundness.

TDDH group: definition

Trapdoor DDH group

$(\mathbb{G}, G, \tau) \leftarrow \textsc{GpGen}(1^k)$ is a trapdoor DDH group if:

1. the DDH problem is hard for $(\mathbb{G}, G)$ without the trapdoor τ
2. the CDH problem is hard even with the trapdoor τ
3. there is a distinguishing algorithm $\textsc{Solve}(X, Y, Z, \tau)$ which:
   - always accepts when (X, Y, Z) is a DDH tuple (completeness)
   - accepts with negligible probability for any adversarially generated $Z \leftarrow \mathcal{A}(X, Y)$ (soundness)

When $\textsc{Solve}$ always rejects on input a non-DDH tuple (X, Y, Z), we say that the TDDH group has perfect soundness.

TDDH group: definition

Trapdoor DDH group

$(\mathbb{G}, G, \tau) \leftarrow \textsc{GpGen}(1^k)$ is a trapdoor DDH group if:

1. the DDH problem is hard for $(\mathbb{G}, G)$ without the trapdoor τ
2. the CDH problem is hard even with the trapdoor τ
3. there is a distinguishing algorithm $\textsc{Solve}(X, Y, Z, \tau)$ which:
   - always accepts when (X, Y, Z) is a DDH tuple (completeness)
   - accepts with negligible probability for any adversarially generated $Z \leftarrow \mathcal{A}(X, Y)$ (soundness)

When $\textsc{Solve}$ always rejects on input a non-DDH tuple (X, Y, Z), we say that the TDDH group has perfect soundness.

Original proposals by Dent-Galbraith [DG06]

Dent and Galbraith originally proposed two TDDH group constructions:

1. disguised elliptic curve [Frey98]
   → broken by Morales [Mor08]

2. hidden pairing:
   - uses an elliptic curve E over the ring $\mathbb{Z}_N$, $N = p_1 p_2$
   - point $G \in E(\mathbb{Z}_N)$ of order $r_1 r_2$ where $r_1 | (p_1 + 1)$ and $r_2 | (p_2 + 1)$
   - the trapdoor is $\tau = (p_1, p_2, r_1, r_2)$
   - by the CRT, $(X, Y, Z) \in \langle G \rangle^3$ is a DDH tuple iff it reduces to a DDH tuple in $E(\mathbb{F}_{p_1})$ and $E(\mathbb{F}_{p_2})$
     → solve the DDH problem in $E(\mathbb{F}_{p_1})$ and $E(\mathbb{F}_{p_2})$ using a pairing
   - problem: no obvious way to hash into $\langle G \rangle$

Original proposals by Dent-Galbraith [DG06]

Dent and Galbraith originally proposed two TDDH group constructions:

1. disguised elliptic curve [Frey98]
   $\rightarrow$ broken by Morales [Mor08]
2. hidden pairing:
   - uses an elliptic curve E over the ring $\mathbb{Z}_N$, $N = p_1 p_2$
   - point $G \in E(\mathbb{Z}_N)$ of order $r_1 r_2$ where $r_1 | (p_1 + 1)$ and $r_2 | (p_2 + 1)$
   - the trapdoor is $\tau = (p_1, p_2, r_1, r_2)$
   - by the CRT, $(X, Y, Z) \in \langle G \rangle^3$ is a DDH tuple iff it reduces to a DDH tuple in $E(\mathbb{F}_{p_1})$ and $E(\mathbb{F}_{p_2})$
     $\rightarrow$ solve the DDH problem in $E(\mathbb{F}_{p_1})$ and $E(\mathbb{F}_{p_2})$ using a pairing
   - problem: no obvious way to hash into $\langle G \rangle$

Original proposals by Dent-Galbraith [DG06]

Dent and Galbraith originally proposed two TDDH group constructions:

1. disguised elliptic curve [Frey98]
   $\rightarrow$ broken by Morales [Mor08]

2. hidden pairing:
   - uses an elliptic curve E over the ring $\mathbb{Z}_N$, $N = p_1 p_2$
   - point $G \in E(\mathbb{Z}_N)$ of order $r_1 r_2$ where $r_1 | (p_1 + 1)$ and $r_2 | (p_2 + 1)$
   - the trapdoor is $\tau = (p_1, p_2, r_1, r_2)$
   - by the CRT, $(X, Y, Z) \in \langle G \rangle^3$ is a DDH tuple iff it reduces to a DDH tuple in $E(\mathbb{F}_{p_1})$ and $E(\mathbb{F}_{p_2})$
     $\rightarrow$ solve the DDH problem in $E(\mathbb{F}_{p_1})$ and $E(\mathbb{F}_{p_2})$ using a pairing
   - problem: no obvious way to hash into $\langle G \rangle$

Static TDDH groups

Static TDDH group = more restricted variant of TDDH group
→ the trapdoor τ_x is dedicated to some fixed element X

Static trapdoor DDH group

$(\mathbb{G}, G, \tau) \leftarrow \textsc{GpGen}(1^k)$ is a static TDDH group if there is a randomized algorithm $(X, \tau_x) \leftarrow \textsc{Sample}(\tau)$ taking the master trapdoor τ as input such that:

1. the DDH problem is hard for $(\mathbb{G}, G)$ without the trapdoor τ
2. the static CDH problem for (G, X) is hard even given τ_x
3. there is a distinguishing algorithm $\textsc{Solve}(X, Y, Z, \tau_x)$ which distinguishes DDH tuples from non-DDH tuples

Remark: in a static trapdoor DDH group, the Strong Diffie-Hellman problem (*i.e.* solving the CDH problem given a static DDH oracle) is hard

Outline

Outline

A TDDH group based on composite residuosity [BCP03]

- $N = pq$, with p, q safe primes
- $\mathbb{G} = \mathbb{QR}_{N^2}$ is the group of quadratic residues mod N^2
- G generator of $\mathbb{G}$

Partial discrete log (Paillier [Pai99])

Given the factorization of N, it is possible to compute efficiently the partial discrete log defined as:

$$\mathtt{PDlog}_G(X) := \mathtt{Dlog}_G(X) \bmod N \ .$$

A TDDH group based on composite residuosity [BCP03]

- $N = pq$, with p, q safe primes
- $\mathbb{G} = \mathbb{QR}_{N^2}$ is the group of quadratic residues mod N^2
- G generator of $\mathbb{G}$

Partial discrete log (Paillier [Pai99])

Given the factorization of N, it is possible to compute efficiently the partial discrete log defined as:

$$\mathtt{PDlog}_G(X) := \mathtt{Dlog}_G(X) \bmod N \ .$$

A TDDH group based on composite residuosity [BCP03]

GpGen(1^k):

- $N = pq$, with p, q safe primes
- $\mathbb{G} = \mathbb{QR}_{N^2}$ is the group of quadratic residues mod N^2
- G generator of $\mathbb{G}$
- trapdoor $\tau = (p, q)$

Solving the DDH problem in $(\mathbb{G}, G)$ using trapdoor $\tau = (p, q)$:

- input $(X, Y, Z) \in \mathbb{G}^3$
- compute $x' = \mathtt{PDlog}_G(X)$, $y' = \mathtt{PDlog}_G(Y)$, $z' = \mathtt{PDlog}_G(Z)$
- check whether $x'y' = z' \bmod N$

Described as a "DH gap group" by Bresson et al. [BCP08]

A TDDH group based on composite residuosity [BCP03]

GpGen(1^k):

- $N = pq$, with p, q safe primes
- $\mathbb{G} = \mathbb{QR}_{N^2}$ is the group of quadratic residues mod N^2
- G generator of $\mathbb{G}$
- trapdoor $\tau = (p, q)$

Solving the DDH problem in $(\mathbb{G}, G)$ using trapdoor $\tau = (p, q)$:

- input $(X, Y, Z) \in \mathbb{G}^3$
- compute $x' = \mathtt{PDlog}_G(X)$, $y' = \mathtt{PDlog}_G(Y)$, $z' = \mathtt{PDlog}_G(Z)$
- check whether $x'y' = z' \bmod N$

Described as a "DH gap group" by Bresson et al. [BCP08]

A TDDH group based on composite residuosity [BCP03]

$\textsc{GpGen}(1^k)$:

- $N = pq$, with p, q safe primes
- $\mathbb{G} = \mathbb{QR}_{N^2}$ is the group of quadratic residues mod N^2
- G generator of $\mathbb{G}$
- trapdoor $\tau = (p, q)$

Solving the DDH problem in $(\mathbb{G}, G)$ using trapdoor $\tau = (p, q)$:

- input $(X, Y, Z) \in \mathbb{G}^3$
- compute $x' = \mathtt{PDlog}_G(X)$, $y' = \mathtt{PDlog}_G(Y)$, $z' = \mathtt{PDlog}_G(Z)$
- check whether $x'y' = z' \bmod N$

Described as a "DH gap group" by Bresson et al. [BCP08]

A TDDH group based on composite residuosity [BCP03]

- The soundness property relies on the following problem:

Partial CDH problem

Given N and G generator of $\mathbb{G} = \mathbb{QR}_{N^2}$, and $X, Y \leftarrow_\$ \mathbb{G}$, output Z such that $\mathtt{PDlog}_G(Z) = \mathtt{PDlog}_G(X) \times \mathtt{PDlog}_G(Y) \bmod N$.

- Issue: this TDDH group does not have perfect soundness
  The $\textsc{Solve}$ algorithm accepts even for a non-DDH tuple (X, Y, Z) such that $\mathtt{PDlog}_G(Z) = \mathtt{PDlog}_G(X) \times \mathtt{PDlog}_G(Y) \bmod N$.
- Given a DDH tuple (X, Y, Z), anyone can compute $Z' = ZU^N$, and (X, Y, Z') is a non-DDH tuple which fools the $\textsc{Solve}$ algorithm
  $\rightarrow$ problem for some applications (esp. undeniable signatures)

A TDDH group based on composite residuosity [BCP03]

- The soundness property relies on the following problem:

Partial CDH problem

Given N and G generator of $\mathbb{G} = \mathbb{QR}_{N^2}$, and $X, Y \leftarrow_{\$} \mathbb{G}$, output Z such that $\texttt{PDlog}_G(Z) = \texttt{PDlog}_G(X) \times \texttt{PDlog}_G(Y) \bmod N$.

- Issue: this TDDH group does not have perfect soundness
  The $\textsc{Solve}$ algorithm accepts even for a non-DDH tuple (X, Y, Z) such that $\texttt{PDlog}_G(Z) = \texttt{PDlog}_G(X) \times \texttt{PDlog}_G(Y) \bmod N$.
- Given a DDH tuple (X, Y, Z), anyone can compute $Z' = ZU^N$, and (X, Y, Z') is a non-DDH tuple which fools the $\textsc{Solve}$ algorithm
  $\rightarrow$ problem for some applications (esp. undeniable signatures)

A TDDH group based on composite residuosity [BCP03]

- The soundness property relies on the following problem:

Partial CDH problem

Given N and G generator of $\mathbb{G} = \mathbb{QR}_{N^2}$, and $X, Y \leftarrow_\$ \mathbb{G}$, output Z such that $\mathtt{PDlog}_G(Z) = \mathtt{PDlog}_G(X) \times \mathtt{PDlog}_G(Y) \bmod N$.

- Issue: this TDDH group does not have perfect soundness
  The $\textsc{Solve}$ algorithm accepts even for a non-DDH tuple (X, Y, Z) such that $\mathtt{PDlog}_G(Z) = \mathtt{PDlog}_G(X) \times \mathtt{PDlog}_G(Y) \bmod N$.
- Given a DDH tuple (X, Y, Z), anyone can compute $Z' = ZU^N$, and (X, Y, Z') is a non-DDH tuple which fools the $\textsc{Solve}$ algorithm
  $\rightarrow$ problem for some applications (esp. undeniable signatures)

Outline

A static TDDH group based on RSA

- $\textsc{GpGen}(1^k)$:
  - $N = pq$, with p, q safe primes
  - $\mathbb{G} = \mathbb{J}_N$ is the subgroup of $\mathbb{Z}_N^*$ of elements with Jacobi symbol 1
  - G generator of $\mathbb{G}$
  - master trapdoor $\tau = (p, q)$
- sampling a group element and the corresponding trapdoor:
  - draw $x \leftarrow_\$ \{1, \ldots, |\mathbb{J}_N|\}$, let $X = G^x$
  - the trapdoor is $\tau_x = 1/x \bmod \mathrm{ord}(\mathbb{J}_N)$
- solving the DDH problem for $(X, Y, Z) \in \mathbb{G}^3$:
  $\rightarrow$ check whether $Z^{\tau_x} = Y$ (satisfied iff $Z = Y^x$)
- Theorem: Under the DDH assumption and the RSA assumption, this is a static TDDH group with perfect soundness
- NB: implies that Strong DH is hard in $\mathbb{J}_N$ under the RSA assumption

A static TDDH group based on RSA

- $\textsc{GpGen}(1^k)$:
  - $N = pq$, with p, q safe primes
  - $\mathbb{G} = \mathbb{J}_N$ is the subgroup of $\mathbb{Z}_N^*$ of elements with Jacobi symbol 1
  - G generator of $\mathbb{G}$
  - master trapdoor $\tau = (p, q)$
- sampling a group element and the corresponding trapdoor:
  - draw $x \leftarrow_{\$} \{1, \ldots, |\mathbb{J}_N|\}$, let $X = G^x$
  - the trapdoor is $\tau_x = 1/x \bmod \mathrm{ord}(\mathbb{J}_N)$
- solving the DDH problem for $(X, Y, Z) \in \mathbb{G}^3$:
  $\rightarrow$ check whether $Z^{\tau_x} = Y$ (satisfied iff $Z = Y^x$)
- Theorem: Under the DDH assumption and the RSA assumption, this is a static TDDH group with perfect soundness
- NB: implies that Strong DH is hard in $\mathbb{J}_N$ under the RSA assumption

A static TDDH group based on RSA

- $\text{GpGen}(1^k)$:
  - $N = pq$, with p, q safe primes
  - $\mathbb{G} = \mathbb{J}_N$ is the subgroup of $\mathbb{Z}_N^*$ of elements with Jacobi symbol 1
  - G generator of $\mathbb{G}$
  - master trapdoor $\tau = (p, q)$
- sampling a group element and the corresponding trapdoor:
  - draw $x \leftarrow_{\$} \{1, \ldots, |\mathbb{J}_N|\}$, let $X = G^x$
  - the trapdoor is $\tau_x = 1/x \bmod \mathrm{ord}(\mathbb{J}_N)$
- solving the DDH problem for $(X, Y, Z) \in \mathbb{G}^3$:
  $\rightarrow$ check whether $Z^{\tau_x} = Y$ (satisfied iff $Z = Y^x$)
- Theorem: Under the DDH assumption and the RSA assumption, this is a static TDDH group with perfect soundness
- NB: implies that Strong DH is hard in $\mathbb{J}_N$ under the RSA assumption

A static TDDH group based on RSA

- $\textsc{GpGen}(1^k)$:
  - $N = pq$, with p, q safe primes
  - $\mathbb{G} = \mathbb{J}_N$ is the subgroup of $\mathbb{Z}_N^*$ of elements with Jacobi symbol 1
  - G generator of $\mathbb{G}$
  - master trapdoor $\tau = (p, q)$
- sampling a group element and the corresponding trapdoor:
  - draw $x \leftarrow_\$ \{1, \ldots, |\mathbb{J}_N|\}$, let $X = G^x$
  - the trapdoor is $\tau_x = 1/x \bmod \mathrm{ord}(\mathbb{J}_N)$
- solving the DDH problem for $(X, Y, Z) \in \mathbb{G}^3$:
  $\rightarrow$ check whether $Z^{\tau_x} = Y$ (satisfied iff $Z = Y^x$)
- Theorem: Under the DDH assumption and the RSA assumption, this is a static TDDH group with perfect soundness
- NB: implies that Strong DH is hard in $\mathbb{J}_N$ under the RSA assumption

A static TDDH group based on RSA

- $\textsc{GpGen}(1^k)$:
  - $N = pq$, with p, q safe primes
  - $\mathbb{G} = \mathbb{J}_N$ is the subgroup of $\mathbb{Z}_N^*$ of elements with Jacobi symbol 1
  - G generator of $\mathbb{G}$
  - master trapdoor $\tau = (p, q)$
- sampling a group element and the corresponding trapdoor:
  - draw $x \leftarrow_{\$} \{1, \ldots, |\mathbb{J}_N|\}$, let $X = G^x$
  - the trapdoor is $\tau_x = 1/x \bmod \mathrm{ord}(\mathbb{J}_N)$
- solving the DDH problem for $(X, Y, Z) \in \mathbb{G}^3$:
  $\rightarrow$ check whether $Z^{\tau_x} = Y$ (satisfied iff $Z = Y^x$)
- Theorem: Under the DDH assumption and the RSA assumption, this is a static TDDH group with perfect soundness
- NB: implies that Strong DH is hard in $\mathbb{J}_N$ under the RSA assumption

Outline

A static TDDH group based on factoring

- $\textsc{GpGen}(1^k)$:
  - $N = pq$, with p, q safe primes
  - $\mathbb{G} = \mathbb{J}_N^+ = \mathbb{J}_N \cap [1, (N-1)/2]$, group operation: $a * b := |a \cdot b \bmod N|$
    $\mathbb{J}_N^+ \simeq \mathbb{J}_N / \{+1, -1\}$ (group of *signed quadratic residues* [HK09])
  - generator G of $\mathbb{G}$
  - master trapdoor $\tau = (p, q)$
- sampling a group element and the corresponding trapdoor:
  - draw $x \leftarrow_{\$} \{1, \ldots, |\mathbb{J}_N^+|\}$, let $X = G^x$
  - the trapdoor is $\tau_x = 2x \pm m$ with $m = \mathrm{ord}(\mathbb{J}_N^+)$
- solving the DDH problem for $(X, Y, Z) \in \mathbb{G}^3$:
  $\rightarrow$ check whether $Z^2 = Y^{\tau_x}$ (satisfied iff $Z = Y^x$)
- Theorem: Under the DDH assumption and the factoring assumption, this group is a static TDDH group with perfect soundness
- NB: implies that Strong DH is hard for $\mathbb{J}_N^+$ under the factoring assumption [HK09]

A static TDDH group based on factoring

- $\textsc{GpGen}(1^k)$:
  - $N = pq$, with p, q safe primes
  - $\mathbb{G} = \mathbb{J}_N^+ = \mathbb{J}_N \cap [1, (N-1)/2]$, group operation: $a * b := |a \cdot b \bmod N|$
    $\mathbb{J}_N^+ \simeq \mathbb{J}_N / \{+1, -1\}$ (group of *signed quadratic residues* [HK09])
  - generator G of $\mathbb{G}$
  - master trapdoor $\tau = (p, q)$
- sampling a group element and the corresponding trapdoor:
  - draw $x \leftarrow_\$ \{1, \ldots, |\mathbb{J}_N^+|\}$, let $X = G^x$
  - the trapdoor is $\tau_x = 2x \pm m$ with $m = \mathrm{ord}(\mathbb{J}_N^+)$
- solving the DDH problem for $(X, Y, Z) \in \mathbb{G}^3$:
  $\rightarrow$ check whether $Z^2 = Y^{\tau_x}$ (satisfied iff $Z = Y^x$)
- Theorem: Under the DDH assumption and the factoring assumption, this group is a static TDDH group with perfect soundness
- NB: implies that Strong DH is hard for $\mathbb{J}_N^+$ under the factoring assumption [HK09]

A static TDDH group based on factoring

- $\textsc{GpGen}(1^k)$:
  - $N = pq$, with p, q safe primes
  - $\mathbb{G} = \mathbb{J}_N^+ = \mathbb{J}_N \cap [1, (N-1)/2]$, group operation: $a * b := |a \cdot b \bmod N|$
    $\mathbb{J}_N^+ \simeq \mathbb{J}_N / \{+1, -1\}$ (group of *signed quadratic residues* [HK09])
  - generator G of $\mathbb{G}$
  - master trapdoor $\tau = (p, q)$
- sampling a group element and the corresponding trapdoor:
  - draw $x \leftarrow_{\$} \{1, \ldots, |\mathbb{J}_N^+|\}$, let $X = G^x$
  - the trapdoor is $\tau_x = 2x \pm m$ with $m = \mathrm{ord}(\mathbb{J}_N^+)$
- solving the DDH problem for $(X, Y, Z) \in \mathbb{G}^3$:
  $\rightarrow$ check whether $Z^2 = Y^{\tau_x}$ (satisfied iff $Z = Y^x$)
- Theorem: Under the DDH assumption and the factoring assumption, this group is a static TDDH group with perfect soundness
- NB: implies that Strong DH is hard for $\mathbb{J}_N^+$ under the factoring assumption [HK09]

A static TDDH group based on factoring

- $\textsc{GpGen}(1^k)$:
  - $N = pq$, with p, q safe primes
  - $\mathbb{G} = \mathbb{J}_N^+ = \mathbb{J}_N \cap [1, (N-1)/2]$, group operation: $a * b := |a \cdot b \bmod N|$
    $\mathbb{J}_N^+ \simeq \mathbb{J}_N/\{+1, -1\}$ (group of *signed quadratic residues* [HK09])
  - generator G of $\mathbb{G}$
  - master trapdoor $\tau = (p, q)$
- sampling a group element and the corresponding trapdoor:
  - draw $x \leftarrow_\$ \{1, \ldots, |\mathbb{J}_N^+|\}$, let $X = G^x$
  - the trapdoor is $\tau_x = 2x \pm m$ with $m = \mathrm{ord}(\mathbb{J}_N^+)$
- solving the DDH problem for $(X, Y, Z) \in \mathbb{G}^3$:
  $\rightarrow$ check whether $Z^2 = Y^{\tau_x}$ (satisfied iff $Z = Y^x$)
- Theorem: Under the DDH assumption and the factoring assumption, this group is a static TDDH group with perfect soundness
- NB: implies that Strong DH is hard for $\mathbb{J}_N^+$ under the factoring assumption [HK09]

A static TDDH group based on factoring

- $\textsc{GpGen}(1^k)$:
  - $N = pq$, with p, q safe primes
  - $\mathbb{G} = \mathbb{J}_N^+ = \mathbb{J}_N \cap [1, (N-1)/2]$, group operation: $a * b := |a \cdot b \bmod N|$
    $\mathbb{J}_N^+ \simeq \mathbb{J}_N / \{+1, -1\}$ (group of *signed quadratic residues* [HK09])
  - generator G of $\mathbb{G}$
  - master trapdoor $\tau = (p, q)$
- sampling a group element and the corresponding trapdoor:
  - draw $x \leftarrow_\$ \{1, \ldots, |\mathbb{J}_N^+|\}$, let $X = G^x$
  - the trapdoor is $\tau_x = 2x \pm m$ with $m = \mathrm{ord}(\mathbb{J}_N^+)$
- solving the DDH problem for $(X, Y, Z) \in \mathbb{G}^3$:
  $\rightarrow$ check whether $Z^2 = Y^{\tau_x}$ (satisfied iff $Z = Y^x$)
- Theorem: Under the DDH assumption and the factoring assumption, this group is a static TDDH group with perfect soundness
- NB: implies that Strong DH is hard for $\mathbb{J}_N^+$ under the factoring assumption [HK09]

Hashing into groups

For both previous cases, it is possible to securely hash into the underlying group $\mathbb{G}$.

Given $H : \{0,1\}^* \rightarrow \mathbb{Z}_N$, let a be an integer with $\left(\frac{a}{N}\right) = -1$

- for $\mathbb{G} = \mathbb{J}_N$, define

$$H'(x) = \begin{cases} H(x) & \text{if } \left(\frac{H(x)}{N}\right) = 1 \\ a \cdot H(x) \bmod N & \text{if } \left(\frac{H(x)}{N}\right) = -1 \end{cases}$$

- for $\mathbb{G} = \mathbb{J}_N^+$, define

$$H'(x) = \begin{cases} |H(x)| & \text{if } \left(\frac{H(x)}{N}\right) = 1 \\ |a \cdot H(x) \bmod N| & \text{if } \left(\frac{H(x)}{N}\right) = -1 \end{cases}$$

Outline

Definition of a CUS scheme

Undeniable signature = signature that cannot be verified without the cooperation of the signer

Convertible Undeniable Signature Scheme:

- $\texttt{KeyGen}(1^k)$: outputs a public/secret key pair $(\texttt{pk}, \texttt{sk})$ for the signer.
- $\texttt{USign}(\texttt{pk}, \texttt{sk}, m)$: outputs an undeniable signature σ for message m.
- $\Pi_{\text{con}} = (\mathcal{P}_{\text{con}}, \mathcal{V}_{\text{con}})$: confirmation protocol for a valid signature σ
- $\Pi_{\text{dis}} = (\mathcal{P}_{\text{dis}}, \mathcal{V}_{\text{dis}})$: disavowal protocol for an invalid signature σ'
- $\texttt{UConvert}(\texttt{pk}, \texttt{sk})$: outputs a universal receipt ρ_u enabling to universally verify signatures created under $(\texttt{pk}, \texttt{sk})$.
- $\texttt{UVer}(\texttt{pk}, \rho_u, m, \sigma)$: signature verification algorithm using the universal receipt ρ_u

The Chaum-van Antwerpen scheme [CvA89]

Parameters:

- a group $\mathbb{G}$ and a gen. G such that the DDH problem is hard
- a hash function $H : \{0,1\}^* \rightarrow \mathbb{G}$

CvA undeniable signature scheme

- Key generation: $\mathtt{sk} := x \leftarrow_\$ \{1, \ldots, |\langle G \rangle|\}$, $\mathtt{pk} := X := G^x$
- Signing a message m: compute $M = H(m) \in \mathbb{G}$, and $S = M^x$
- Confirming a sig. S for m: prove that $(X, H(M), S)$ is a DDH tuple
  $\rightarrow$ Chaum-Pedersen proof of equality of DL [CP92]
- Denying a sig. S' for m: prove that $(X, H(M), S')$ is a non-DDH tuple
  $\rightarrow$ Camenish-Shoup proof of inequality of DL [CS03]

Note: using a pairing group where DDH is easy yields the Boneh-Lynn-Shacham signature scheme [BLS04]

The CvA scheme with a static TDDH group

Using the CvA scheme with a (static) TDDH group gives new properties.

$\rightarrow$ New $\texttt{KeyGen}$: $(\mathbb{G}, G, \tau) \leftarrow \textsc{GpGen}(1^k)$, $(X, \tau_X) \leftarrow \textsc{Sample}(\tau)$

- signer public key: $\mathrm{pk} = X = G^x$
- signer secret key: $\mathrm{sk} = (x, \tau_X)$, where τ_X is the trapdoor for solving the static DDH problem for X

The signer now can use the trapdoor τ_X as follows:

- delegated verification: disclose the trapdoor τ_X to the delegated verifier DV
  $\rightarrow$ DV can confirm/disavow signatures using witness τ_X
- universal convertibility: simply make the trapdoor τ_X public
  $\rightarrow$ anyone can verify signatures S using $\textsc{Solve}(X, H(m), S, \tau_X)$

Caveat: requires perfect soundness

The CvA scheme with a static TDDH group

Using the CvA scheme with a (static) TDDH group gives new properties.

$\rightarrow$ New KeyGen: $(\mathbb{G}, G, \tau) \leftarrow \textsc{GpGen}(1^k)$, $(X, \tau_X) \leftarrow \textsc{Sample}(\tau)$

- signer public key: $\mathtt{pk} = X = G^x$
- signer secret key: $\mathtt{sk} = (x, \tau_X)$, where τ_X is the trapdoor for solving the static DDH problem for X

The signer now can use the trapdoor τ_X as follows:

- delegated verification: disclose the trapdoor τ_X to the delegated verifier DV
  $\rightarrow$ DV can confirm/disavow signatures using witness τ_X
- universal convertibility: simply make the trapdoor τ_X public
  $\rightarrow$ anyone can verify signatures S using $\textsc{Solve}(X, H(m), S, \tau_X)$

Caveat: requires perfect soundness

The CvA scheme with a static TDDH group

Using the CvA scheme with a (static) TDDH group gives new properties.

$\rightarrow$ New KeyGen: $(\mathbb{G}, G, \tau) \leftarrow \textsc{GpGen}(1^k)$, $(X, \tau_x) \leftarrow \textsc{Sample}(\tau)$

- signer public key: $\mathtt{pk} = X = G^x$
- signer secret key: $\mathtt{sk} = (x, \tau_x)$, where τ_x is the trapdoor for solving the static DDH problem for X

The signer now can use the trapdoor τ_x as follows:

- delegated verification: disclose the trapdoor τ_x to the delegated verifier DV
  $\rightarrow$ DV can confirm/disavow signatures using witness τ_x
- universal convertibility: simply make the trapdoor τ_x public
  $\rightarrow$ anyone can verify signatures S using $\textsc{Solve}(X, H(m), S, \tau_x)$

Caveat: requires perfect soundness

The CvA scheme with a static TDDH group

Using the CvA scheme with a (static) TDDH group gives new properties.

→ New KeyGen: $(\mathbb{G}, G, \tau) \leftarrow \textsc{GpGen}(1^k)$, $(X, \tau_x) \leftarrow \textsc{Sample}(\tau)$

- signer public key: $\texttt{pk} = X = G^x$
- signer secret key: $\texttt{sk} = (x, \tau_x)$, where τ_x is the trapdoor for solving the static DDH problem for X

The signer now can use the trapdoor τ_x as follows:

- delegated verification: disclose the trapdoor τ_x to the delegated verifier DV
  → DV can confirm/disavow signatures using witness τ_x
- universal convertibility: simply make the trapdoor τ_x public
  → anyone can verify signatures S using $\textsc{Solve}(X, H(m), S, \tau_x)$

Caveat: requires perfect soundness

The CvA scheme with a static TDDH group

New KeyGen:

- signer public key $\mathtt{pk} = X = G^x$
- signer secret key $\mathtt{sk} = (x, \tau_x)$, where τ_x is the trapdoor for solving the static DDH problem for X

Security properties:

- unforgeability under CMA attacks:
  $\rightarrow$ relies on hardness of the CDH problem (even given τ_x)
- invisibility under CMA attacks (impossibility to distinguish a valid signature from an random one):
  $\rightarrow$ relies on hardness of the DDH problem (without τ_x)

Instantiations

The Chaum-van Antwerpen scheme can be instantiated with the two proposed static TDDH groups:

- RSA-based static TDDH group $\mathbb{J}_N$:
  $\rightarrow$ scheme similar to the one by Gennaro, Rabin, and Krawczyk [GRK00]
- factoring-based static TDDH group $\mathbb{J}_N^+$:
  $\rightarrow$ scheme similar to the one by Galbraith and Mao [GM03]

Key generation must be done with care. One needs to certify that:

- $\mathbb{Z}_N^*$ has no small order subgroup
- G is a generator of the specified group

$\rightarrow$ demand that the signer proves in ZK that N is a product of safe primes

Instantiations

The Chaum-van Antwerpen scheme can be instantiated with the two proposed static TDDH groups:

- RSA-based static TDDH group $\mathbb{J}_N$:
  $\rightarrow$ scheme similar to the one by Gennaro, Rabin, and Krawczyk [GRK00]
- factoring-based static TDDH group $\mathbb{J}_N^+$:
  $\rightarrow$ scheme similar to the one by Galbraith and Mao [GM03]

Key generation must be done with care. One needs to certify that:

- $\mathbb{Z}_N^*$ has no small order subgroup
- G is a generator of the specified group

$\rightarrow$ demand that the signer proves in ZK that N is a product of safe primes

Instantiations

The Chaum-van Antwerpen scheme can be instantiated with the two proposed static TDDH groups:

- RSA-based static TDDH group $\mathbb{J}_N$:
  $\rightarrow$ scheme similar to the one by Gennaro, Rabin, and Krawczyk [GRK00]
- factoring-based static TDDH group $\mathbb{J}_N^+$:
  $\rightarrow$ scheme similar to the one by Galbraith and Mao [GM03]

Key generation must be done with care. One needs to certify that:

- $\mathbb{Z}_N^*$ has no small order subgroup
- G is a generator of the specified group

$\rightarrow$ demand that the signer proves in ZK that N is a product of safe primes

Instantiations

The Chaum-van Antwerpen scheme can be instantiated with the two proposed static TDDH groups:

- RSA-based static TDDH group $\mathbb{J}_N$:
  $\rightarrow$ scheme similar to the one by Gennaro, Rabin, and Krawczyk [GRK00]
- factoring-based static TDDH group $\mathbb{J}_N^+$:
  $\rightarrow$ scheme similar to the one by Galbraith and Mao [GM03]

Key generation must be done with care. One needs to certify that:

- $\mathbb{Z}_N^*$ has no small order subgroup
- G is a generator of the specified group

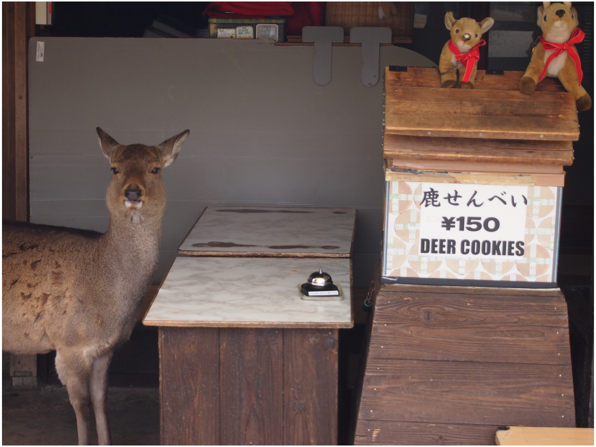
$\rightarrow$ demand that the signer proves in ZK that N is a product of safe primes

Conclusion

Open problems:

- build a TDDH group with perfect soundness and a way to securely hash into it
- build a TDDH group with prime order
- other applications of TDDH groups?
  $\rightarrow$ suggested by a PKC reviewer:
  generic construction of extractable hash proof system [Wee10]
  $\Rightarrow$ CCA-secure KEM

Thanks for your attention! Comments or questions?



Damn! Where's my wallet?