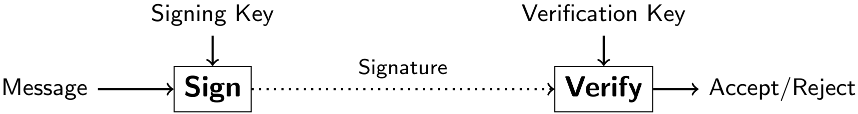
# Short Signatures with Short Public Keys From Homomorphic Trapdoor Functions

Jacob Alperin-Sheriff

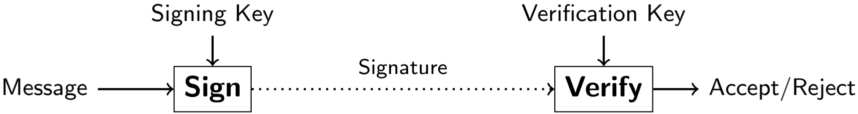
School of Computer Science
Georgia Tech

# Stateless Standard-Model Signature Schemes



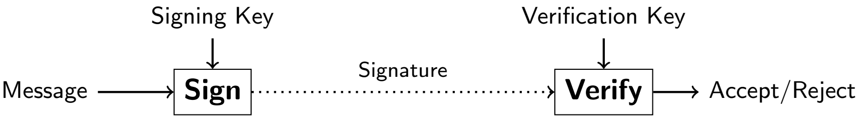

- Want short public key, secret key, signatures

# Stateless Standard-Model Signature Schemes



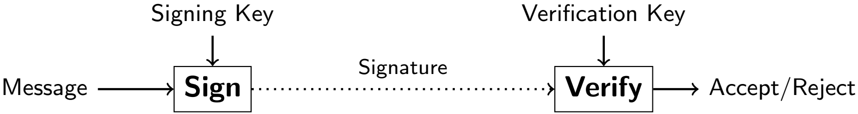

- Want short public key, secret key, signatures
- Under classical number-theoretic assumptions [Wat'09,HW'09]:

# Stateless Standard-Model Signature Schemes



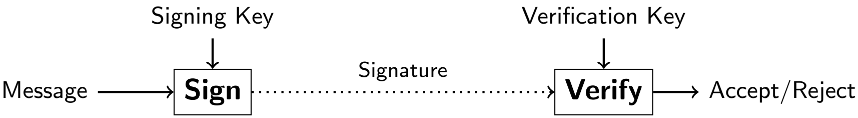

- Want short public key, secret key, signatures
- Under classical number-theoretic assumptions [Wat'09,HW'09]:
  - ⋆ Constant number of group elements in sk, sigs, vk
  - ⋆ Linear size in security parameter $\lambda$

# Stateless Standard-Model Signature Schemes



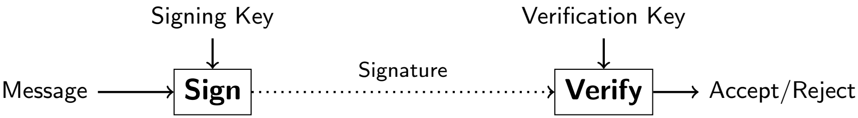

- Want short public key, secret key, signatures
- Under classical number-theoretic assumptions [Wat'09,HW'09]:
  - Constant number of group elements in sk, sigs, vk
  - Linear size in security parameter $\lambda$
- Under lattice-based assumptions (in ring setting)

# Stateless Standard-Model Signature Schemes



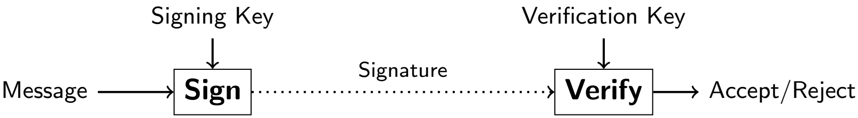

- Want short public key, secret key, signatures
- Under classical number-theoretic assumptions [Wat'09,HW'09]:
  - ⋆ Constant number of group elements in sk, sigs, vk
  - ⋆ Linear size in security parameter $\lambda$
- Under lattice-based assumptions (in ring setting)
  - ⋆ Trapdoors and pre-images already have quasilinear ($\tilde{O}(\lambda)$) size

# Stateless Standard-Model Signature Schemes




- Want short public key, secret key, signatures
- Under classical number-theoretic assumptions [Wat'09,HW'09]:
  - ⋆ Constant number of group elements in sk, sigs, vk
  - ⋆ Linear size in security parameter $\lambda$
- Under lattice-based assumptions (in ring setting)
  - ⋆ Trapdoors and pre-images already have quasilinear ($\tilde{O}(\lambda)$) size
  - ⋆ Best schemes require:
    - Logarithmic number of pre-images in signatures [BHJ+14]
    - Logarithmic number of trapdoors in the public key [DM14]

# Stateless Standard-Model Signature Schemes




- Want short public key, secret key, signatures
- Under classical number-theoretic assumptions [Wat'09,HW'09]:
  - ⋆ Constant number of group elements in sk, sigs, vk
  - ⋆ Linear size in security parameter $\lambda$
- Under lattice-based assumptions (in ring setting)
  - ⋆ Trapdoors and pre-images already have quasilinear ($\tilde{O}(\lambda)$) size
  - ⋆ Best schemes require:
    - Logarithmic number of pre-images in signatures [BHJ+14]
    - Logarithmic number of trapdoors in the public key [DM14]
  - ⋆ Can we do better?

# Our Results

- Constant number of trapdoors in public key, short signatures

# Our Results

- Constant number of trapdoors in public key, short signatures
- Starting Point: DM14 signature scheme

# Our Results

- Constant number of trapdoors in public key, short signatures
- Starting Point: DM14 signature scheme
  - DM14 used linear homomorphisms over trapdoor functions

# Our Results

- Constant number of trapdoors in public key, short signatures
- Starting Point: DM14 signature scheme
  - ⋆ DM14 used linear homomorphisms over trapdoor functions
  - ⋆ Our idea: Use full homomorphisms over trapdoor functions

# Our Results

- Constant number of trapdoors in public key, short signatures
- Starting Point: DM14 signature scheme
  - ⋆ DM14 used linear homomorphisms over trapdoor functions
  - ⋆ Our idea: Use full homomorphisms over trapdoor functions
- Comparison to previous work ($d = \omega(\log \log n)$)

| Scheme | pk<br>$R_q^{1\times k}$ mat. | sk<br>$R_q^{k\times k}$ mat. | Sig.<br>$R_q^k$ vec. | SIS param<br>$\beta$ |
|---|---|---|---|---|
| Boy10,MP12 | $n$ | $n$ | 1 | $\tilde{\Omega}(n^{5/2})$ |
| BHJ+14 | 1 | 1 | $d$ | $\tilde{\Omega}(n^{5/2})$ |
| DM14 | $d$ | 1 | 1 | $\tilde{\Omega}(n^{7/2})$ |
| This work | 1 | 1 | 1 | $\tilde{\Omega}(d^{2d} \cdot n^{11/2})$ |

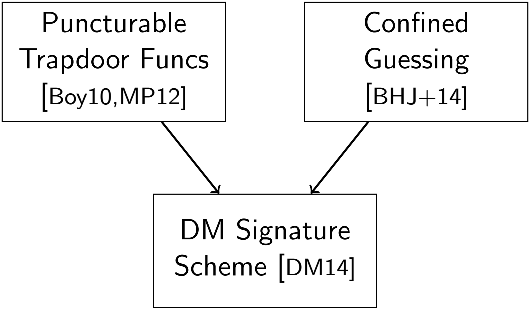
# Our Results

- Constant number of trapdoors in public key, short signatures
- Starting Point: DM14 signature scheme
  - DM14 used linear homomorphisms over trapdoor functions
  - Our idea: Use full homomorphisms over trapdoor functions
- Comparison to previous work ($d = \omega(\log\log n)$)

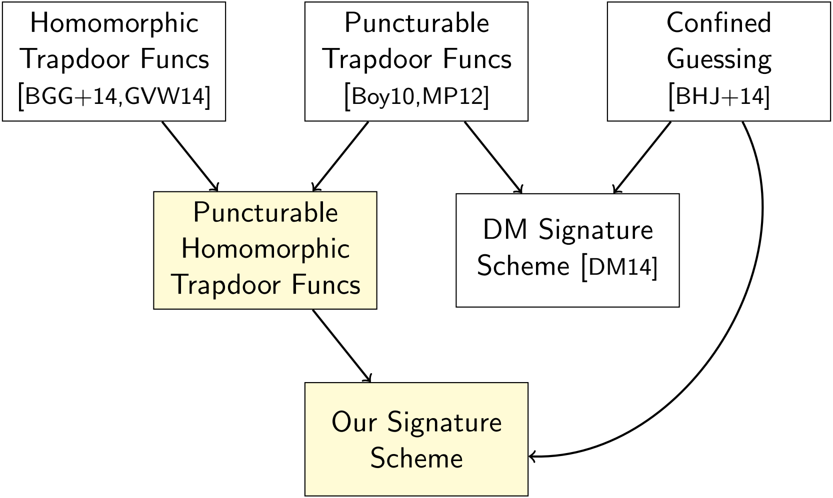
| Scheme | pk $R_q^{1\times k}$ mat. | sk $R_q^{k\times k}$ mat. | Sig. $R_q^k$ vec. | SIS param $\beta$ |
|---|---|---|---|---|
| Boy10,MP12 | $n$ | $n$ | 1 | $\tilde{\Omega}(n^{5/2})$ |
| BHJ+14 | 1 | 1 | $d$ | $\tilde{\Omega}(n^{5/2})$ |
| DM14 | $d$ | 1 | 1 | $\tilde{\Omega}(n^{7/2})$ |
| This work | 1 | 1 | 1 | $\tilde{\Omega}(d^{2d} \cdot n^{11/2})$ |

- SIS param can be (large) poly-sized if we set $d = O(\log n / \log\log n)$

# Construction Outline



# Construction Outline



# Puncturable Homomorphic Trapdoor Functions (PHTDF)

$$f_{pk,a,x}(u) \dashrightarrow v$$

- Trapdoor functions $a$ with associated (hidden) tag $t \in \mathcal{T}$.

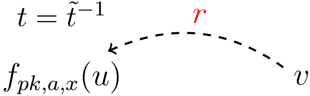
# Puncturable Homomorphic Trapdoor Functions (PHTDF)

$$t = \tilde{t}^{-1}$$

$$f_{pk,a,x}(u) \qquad v$$

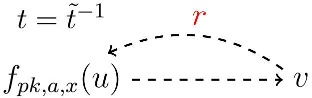
- Trapdoor functions $a$ with associated (hidden) tag $t \in \mathcal{T}$.
- Tag $t$ is invertible:

# Puncturable Homomorphic Trapdoor Functions (PHTDF)




- Trapdoor functions $a$ with associated (hidden) tag $t \in \mathcal{T}$.
- Tag $t$ is invertible: can invert $f$ with trapdoor $r$.

# Puncturable Homorphic Trapdoor Functions (PHTDF)




- Trapdoor functions $a$ with associated (hidden) tag $t \in \mathcal{T}$.
- Tag $t$ is invertible: can invert $f$ with trapdoor $r$.
- Distributional Equivalence:

$$(r, u \leftarrow \mathcal{U}, v \leftarrow f_{pk,a,x}(u)) \overset{s}{\approx} (r, u \leftarrow f^{-1}_{pk,a,x}(v), v \leftarrow \mathcal{V})$$

# Puncturable Homorphic Trapdoor Functions (PHTDF)

$t = 0$

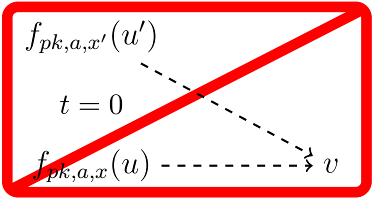
$f_{pk,a,x}(u) \dashrightarrow v$

- Trapdoor functions $a$ with associated (hidden) tag $t \in \mathcal{T}$.
- Tag $t$ is invertible: can invert $f$ with trapdoor $r$.
- Distributional Equivalence:

$$(r, u \leftarrow \mathcal{U}, v \leftarrow f_{pk,a,x}(u)) \overset{s}{\approx} (r, u \leftarrow f^{-1}_{pk,a,x}(v), v \leftarrow \mathcal{V})$$

- Tag $t = 0$: trapdoor function $a$ is "punctured:"

# Puncturable Homorphic Trapdoor Functions (PHTDF)

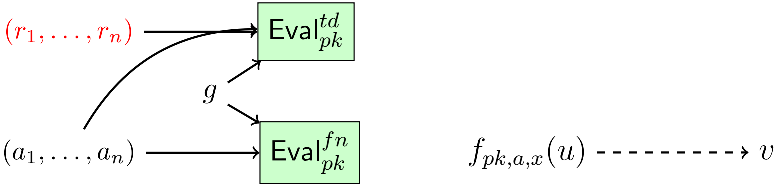



- Trapdoor functions $a$ with associated (hidden) tag $t \in \mathcal{T}$.
- Tag $t$ is invertible: can invert $f$ with trapdoor $r$.
- Distributional Equivalence:

$$(r, u \leftarrow \mathcal{U}, v \leftarrow f_{pk,a,x}(u)) \overset{s}{\approx} (r, u \leftarrow f^{-1}_{pk,a,x}(v), v \leftarrow \mathcal{V})$$

- Tag $t = 0$: trapdoor function $a$ is "punctured:"
  - ⋆ $f_{pk,a,\cdot}(\cdot)$ becomes collision resistant.

# Puncturable Homorphic Trapdoor Functions (PHTDF)

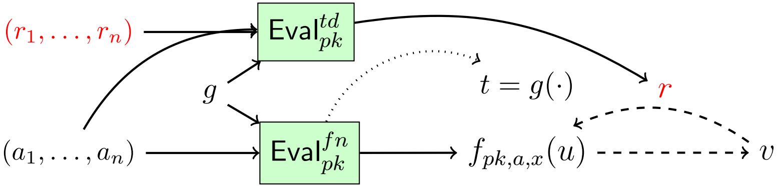



- Trapdoor functions $a$ with associated (hidden) tag $t \in \mathcal{T}$.
- Tag $t$ is invertible: can invert $f$ with trapdoor $r$.
- Distributional Equivalence:

$$(r, u \leftarrow \mathcal{U}, v \leftarrow f_{pk,a,x}(u)) \overset{s}{\approx} (r, u \leftarrow f^{-1}_{pk,a,x}(v), v \leftarrow \mathcal{V})$$

- Tag $t = 0$: trapdoor function $a$ is "punctured:"
  - $f_{pk,a,\cdot}(\cdot)$ becomes collision resistant.
- Homomorphic Properties
  - Can evaluate funcs $g$ over tags $t_i$ associated with $a_i$

# Puncturable Homorphic Trapdoor Functions (PHTDF)




- Trapdoor functions $a$ with associated (hidden) tag $t \in \mathcal{T}$.
- Tag $t$ is invertible: can invert $f$ with trapdoor $r$.
- Distributional Equivalence:

$$(r, u \leftarrow \mathcal{U}, v \leftarrow f_{pk,a,x}(u)) \overset{s}{\approx} (r, u \leftarrow f^{-1}_{pk,a,x}(v), v \leftarrow \mathcal{V})$$

- Tag $t = 0$: trapdoor function $a$ is "punctured:"
  - $f_{pk,a,\cdot}(\cdot)$ becomes collision resistant.
- Homomorphic Properties
  - Can evaluate funcs $g$ over tags $t_i$ associated with $a_i$
  - Yields new trapdoor function $a$ with tag $t$, trapdoor $r$

# Lattice-Based Construction of PHTDFs

$$f_{(\mathbf{A},\mathbf{B}),-\mathbf{A}\mathbf{R}+t\mathbf{G},\mathbf{x}}(\mathbf{u}) \dashrightarrow \mathbf{v} := [\mathbf{A} \mid -\mathbf{A}\mathbf{R} + t\mathbf{G}]\mathbf{u} + \mathbf{B}\mathbf{x}$$

- Construction itself is simple extension of MP12 trapdoors

# Lattice-Based Construction of PHTDFs

$$\mathbf{R}$$

$$f_{(\mathbf{A},\mathbf{B}),-\mathbf{A}\mathbf{R}+t\mathbf{G},\mathbf{x}}(\mathbf{u}) \dashrightarrow \mathbf{v} := [\mathbf{A} \mid -\mathbf{A}\mathbf{R} + t\mathbf{G}]\mathbf{u} + \mathbf{B}\mathbf{x}$$

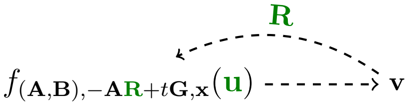
- Construction itself is simple extension of MP12 trapdoors
- Short $\mathbf{R}$ lets us sample short preimage $\mathbf{u}$ for a given $\mathbf{v} - \mathbf{B}\mathbf{x}$.

# Lattice-Based Construction of PHTDFs

$$f_{(\mathbf{A},\mathbf{B}),-\mathbf{A}\mathbf{R}+t\mathbf{G},\mathbf{x}}(\mathbf{u}) \dashrightarrow \mathbf{v} := [\mathbf{A} \mid -\mathbf{A}\mathbf{R} + t\mathbf{G}]\mathbf{u} + \mathbf{B}\mathbf{x}$$

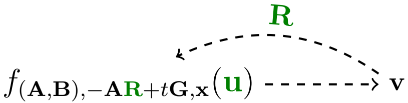
- Construction itself is simple extension of MP12 trapdoors
- Short $\mathbf{R}$ lets us sample short preimage $\mathbf{u}$ for a given $\mathbf{v} - \mathbf{B}\mathbf{x}$.
- Collision resistance when punctured follows from SIS.

# Lattice-Based Construction of PHTDFs




- Construction itself is simple extension of MP12 trapdoors
- Short $\mathbf{R}$ lets us sample short preimage $\mathbf{u}$ for a given $\mathbf{v} - \mathbf{B}\mathbf{x}$.
- Collision resistance when punctured follows from SIS.
- Tags may be arbitrary $n \times n$ matrices

# Lattice-Based Construction of PHTDFs




- Construction itself is simple extension of MP12 trapdoors
- Short $\mathbf{R}$ lets us sample short preimage $\mathbf{u}$ for a given $\mathbf{v} - \mathbf{B}\mathbf{x}$.
- Collision resistance when punctured follows from SIS.
- Tags may be arbitrary $n \times n$ matrices
  - For trapdoor multiplication, at least one must be scalar multiple of $\mathbf{I}$.

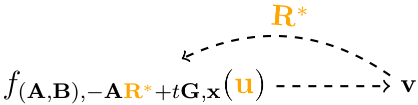
# Lattice-Based Construction of PHTDFs

$$\overset{\mathbf{R}^*}{\curvearrowleft}$$

$$f_{(\mathbf{A},\mathbf{B}),-\mathbf{A}\mathbf{R}^*+t\mathbf{G},\mathbf{x}}(\mathbf{u}) \dashrightarrow \mathbf{v}$$

- Construction itself is simple extension of MP12 trapdoors
- Short $\mathbf{R}$ lets us sample short preimage $\mathbf{u}$ for a given $\mathbf{v} - \mathbf{B}\mathbf{x}$.
- Collision resistance when punctured follows from SIS.
- Tags may be arbitrary $n \times n$ matrices
  - For trapdoor multiplication, at least one must be scalar multiple of $\mathbf{I}$.
- Trapdoor growth from homomorphic computations:
  - **Homom Addition:** Trapdoor grows additively.

# Lattice-Based Construction of PHTDFs




- Construction itself is simple extension of MP12 trapdoors
- Short $\mathbf{R}$ lets us sample short preimage $\mathbf{u}$ for a given $\mathbf{v} - \mathbf{B}\mathbf{x}$.
- Collision resistance when punctured follows from SIS.
- Tags may be arbitrary $n \times n$ matrices
  - For trapdoor multiplication, at least one must be scalar multiple of $\mathbf{I}$.
- Trapdoor growth from homomorphic computations:
  - **Homom Addition:** Trapdoor grows additively.
  - **Homom Multiplication:** Trapdoor grows asymmetrically in $\mathbf{R}_1, \mathbf{R}_2$

$$\mathbf{R}^* := \mathbf{R}_1 \cdot \mathrm{poly}(n) + t_1\mathbf{R}_2$$

# Lattice-Based Construction of PHTDFs

$$f_{(\mathbf{A},\mathbf{B}),-\mathbf{A}\mathbf{R}^*+t\mathbf{G},\mathbf{x}}(\mathbf{u}) \dashrightarrow \mathbf{v}$$

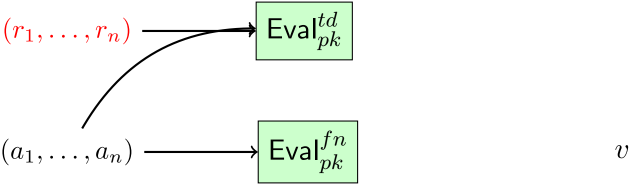
$$\mathbf{v} \xrightarrow{\mathbf{R}^*} \mathbf{u}$$

- Construction itself is simple extension of MP12 trapdoors
- Short $\mathbf{R}$ lets us sample short preimage $\mathbf{u}$ for a given $\mathbf{v} - \mathbf{B}\mathbf{x}$.
- Collision resistance when punctured follows from SIS.
- Tags may be arbitrary $n \times n$ matrices
  - For trapdoor multiplication, at least one must be scalar multiple of $\mathbf{I}$.
- Trapdoor growth from homomorphic computations:
  - **Homom Addition:** Trapdoor grows additively.
  - **Homom Multiplication:** Trapdoor grows asymmetrically in $\mathbf{R}_1, \mathbf{R}_2$

$$\mathbf{R}^* := \mathbf{R}_1 \cdot \mathrm{poly}(n) + t_1\mathbf{R}_2$$

- Larger Trapdoors $\rightarrow$ larger pre-images, larger SIS solutions.

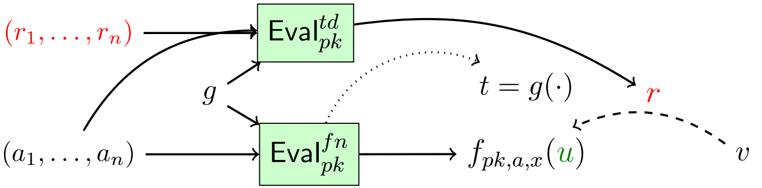
# Signatures from PHTDFs




## Signature Scheme

**Gen**$(1^\lambda)$: Choose $vk = (pk, a_1, \ldots, a_n, v)$, $sk = (r_1, \ldots, r_n)$

# Signatures from PHTDFs

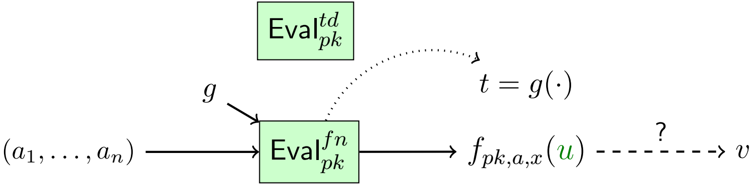



## Signature Scheme

**Gen**$(1^\lambda)$: Choose $vk = (pk, a_1, \ldots, a_n, v)$, $sk = (r_1, \ldots, r_n)$

**Sign**$(x)$: Sample $g \leftarrow \mathcal{G}$. Invert to valid $u$. Output $(u, g)$

# Signatures from PHTDFs



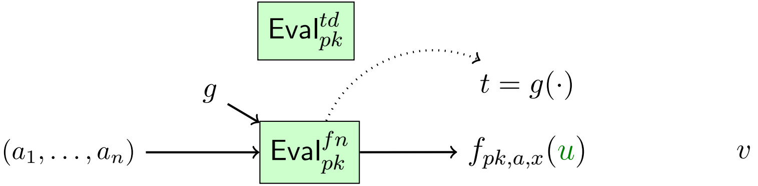

**Signature Scheme**

**Gen**$(1^\lambda)$: Choose $vk = (pk, a_1, \ldots, a_n, v)$, $sk = (r_1, \ldots, r_n)$

**Sign**$(x)$: Sample $g \leftarrow \mathcal{G}$. Invert to valid $u$. Output $(u, g)$

**Ver**$(x, (u, g))$: Verify that $u$ valid, $f_{pk,a,x}(u) = v$.

# Signatures from PHTDFs




## Signature Scheme

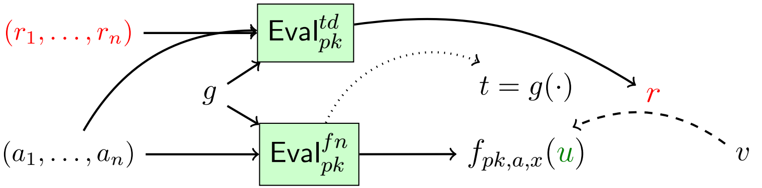
**Gen**$(1^\lambda)$: Choose $vk = (pk, a_1, \ldots, a_n, v)$, $sk = (r_1, \ldots, r_n)$

**Sign**$(x)$: Sample $g \leftarrow \mathcal{G}$. Invert to valid $u$. Output $(u, g)$

**Ver**$(x, (u, g))$: Verify that $u$ valid, $f_{pk,a,x}(u) = v$.

- Scheme security/correctness depend on properties of sampled $g \leftarrow \mathcal{G}$

# Signatures from PHTDFs




**Signature Scheme**

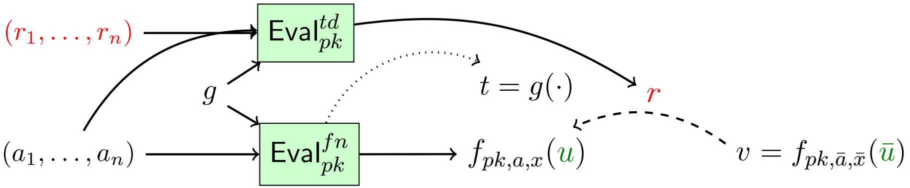
**Gen**$(1^\lambda)$: Choose $vk = (pk, a_1, \ldots, a_n, v)$, $sk = (r_1, \ldots, r_n)$

**Sign**$(x)$: Sample $g \leftarrow \mathcal{G}$. Invert to valid $u$. Output $(u, g)$

**Ver**$(x, (u, g))$: Verify that $u$ valid, $f_{pk,a,x}(u) = v$.

- Scheme security/correctness depend on properties of sampled $g \leftarrow \mathcal{G}$
- Actual Scheme: $t$ must always be invertible.

# Signatures from PHTDFs




**Signature Scheme**

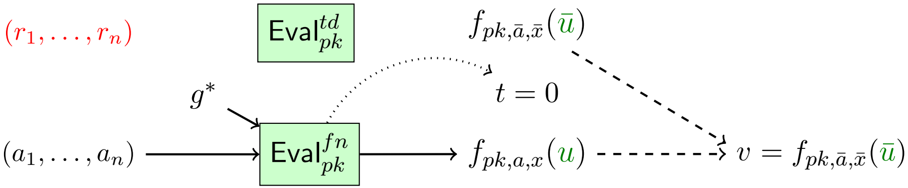
**Gen**$(1^\lambda)$: Choose $vk = (pk, a_1, \ldots, a_n, v)$, $sk = (r_1, \ldots, r_n)$

**Sign**$(x)$: Sample $g \leftarrow \mathcal{G}$. Invert to valid $u$. Output $(u, g)$

**Ver**$(x, (u, g))$: Verify that $u$ valid, $f_{pk,a,x}(u) = v$.

- Scheme security/correctness depend on properties of sampled $g \leftarrow \mathcal{G}$
- Actual Scheme: $t$ must always be invertible.
- Security reduction against $\mathcal{A}$ (with non-negligble probability):
    1. $t = g(\cdot)$ must be invertible for all but one of queries made by $\mathcal{A}$.

# Signatures from PHTDFs




## Signature Scheme

**Gen**$(1^\lambda)$: Choose $vk = (pk, a_1, \dots, a_n, v)$, $sk = (r_1, \dots, r_n)$

**Sign**$(x)$: Sample $g \leftarrow \mathcal{G}$. Invert to valid $u$. Output $(u, g)$

**Ver**$(x, (u, g))$: Verify that $u$ valid, $f_{pk,a,x}(u) = v$.

- Scheme security/correctness depend on properties of sampled $g \leftarrow \mathcal{G}$
- Actual Scheme: $t$ must always be invertible.
- Security reduction against $\mathcal{A}$ (with non-negligble probability):
  1. $t = g(\cdot)$ must be invertible for all but one of queries made by $\mathcal{A}$.
  2. $\mathcal{A}$ chooses $g^*$ for forgery such that $t = g^*(\cdot) = 0$

# Tags and $g$

- Each $g \in \mathcal{G}$ is uniquely specified by a tag $t^{(g)} \in \{0,1\}^n$

# Tags and $g$

$$t^{(g)} = \underbrace{\underbrace{\underbrace{\underbrace{0}_{t_1^{(g)}}\;1}_{t_2^{(g)}}\;0\;1}_{t_3^{(g)}}\;1\;0\;0\;1}_{t_4^{(g)}}\;\cdots$$

- Each $g \in \mathcal{G}$ is uniquely specified by a tag $t^{(g)} \in \{0,1\}^n$
- Tags decompose into prefixes $t_i^{(g)}$ of length $c_i = 2^i$

# Tags and $g$

$$t^{(g)} = \underbrace{0\ 1\ 0\ 1}_{t_3^{(g)}}\ 1\ 0\ 0\ 1\ \cdots \qquad \longrightarrow \qquad x + x^3$$

- Each $g \in \mathcal{G}$ is uniquely specified by a tag $t^{(g)} \in \{0,1\}^n$
- Tags decompose into prefixes $t_i^{(g)}$ of length $c_i = 2^i$
  - Prefixes embed into GF$(2^n)$ (higher coefficients set to 0)

# Tags and $g$

$$t^{(g)} = \underbrace{0\ 1\ 0\ 1}_{t_3^{(g)}}\ 1\ 0\ 0\ 1\ \cdots \qquad x + x^3$$

- Each $g \in \mathcal{G}$ is uniquely specified by a tag $t^{(g)} \in \{0,1\}^n$
- Tags decompose into prefixes $t_i^{(g)}$ of length $c_i = 2^i$
  - ⋆ Prefixes embed into GF$(2^n)$ (higher coefficients set to 0)
  - ⋆ Key Point: Embedding of $t^{(g)} - t^{(g')}$ is invertible for $g \neq g'$.

# Tags and $g$

$a_0$

$\vdots$

$\hat{t}_{i^*}$

- Each $g \in \mathcal{G}$ is uniquely specified by a tag $t^{(g)} \in \{0,1\}^n$
- Tags decompose into prefixes $t_i^{(g)}$ of length $c_i = 2^i$
  - Prefixes embed into GF$(2^n)$ (higher coefficients set to 0)
  - Key Point: Embedding of $t^{(g)} - t^{(g')}$ is invertible for $g \neq g'$.
- Trapdoor $a_0$ in public key has random tag prefix of length $c_{i^*}$.

# Tags and $g$

$$a_0 \\ \vdots \\ \hat{t}_{i^*}$$

- Each $g \in \mathcal{G}$ is uniquely specified by a tag $t^{(g)} \in \{0,1\}^n$
- Tags decompose into prefixes $t_i^{(g)}$ of length $c_i = 2^i$
  - Prefixes embed into GF$(2^n)$ (higher coefficients set to 0)
  - Key Point: Embedding of $t^{(g)} - t^{(g')}$ is invertible for $g \neq g'$.
- Trapdoor $a_0$ in public key has random tag prefix of length $c_{i^*}$.
- Choice of $i^*$ via confined guessing [BHJ+14,DM14]

# Tags and $g$

$$a_0 \\ \vdots \\ \hat{t}_{i^*}$$

- Each $g \in \mathcal{G}$ is uniquely specified by a tag $t^{(g)} \in \{0,1\}^n$
- Tags decompose into prefixes $t_i^{(g)}$ of length $c_i = 2^i$
  - Prefixes embed into $\mathsf{GF}(2^n)$ (higher coefficients set to 0)
  - Key Point: Embedding of $t^{(g)} - t^{(g')}$ is invertible for $g \neq g'$.
- Trapdoor $a_0$ in public key has random tag prefix of length $c_{i^*}$.
- Choice of $i^*$ via confined guessing [BHJ+14,DM14]
  - $\mathcal{A}$ makes $Q$ queries, succeeds with probability $\epsilon$
  - Choose smallest $i^*$ such that $2Q^2/\epsilon \leq 2^{c_{i^*}}$

# Tags and $g$

$$a_0 \quad \vdots \quad \hat{t}_{i^*}$$

- Each $g \in \mathcal{G}$ is uniquely specified by a tag $t^{(g)} \in \{0,1\}^n$
- Tags decompose into prefixes $t_i^{(g)}$ of length $c_i = 2^i$
  - ⋆ Prefixes embed into GF$(2^n)$ (higher coefficients set to 0)
  - ⋆ Key Point: Embedding of $t^{(g)} - t^{(g')}$ is invertible for $g \neq g'$.
- Trapdoor $a_0$ in public key has random tag prefix of length $c_{i^*}$.
- Choice of $i^*$ via confined guessing [BHJ+14,DM14]
  - ⋆ $\mathcal{A}$ makes $Q$ queries, succeeds with probability $\epsilon$
  - ⋆ Choose smallest $i^*$ such that $2Q^2/\epsilon \leq 2^{c_{i^*}}$
  - ⋆ $Q$ random tags will have distinct length $c_i^*$ prefixes with prob $\epsilon/2$

# Tags and $g$

$$a_0$$
$$\vdots$$
$$\hat{t}_{i^*}$$

- Each $g \in \mathcal{G}$ is uniquely specified by a tag $t^{(g)} \in \{0,1\}^n$
- Tags decompose into prefixes $t_i^{(g)}$ of length $c_i = 2^i$
  - ⋆ Prefixes embed into GF$(2^n)$ (higher coefficients set to 0)
  - ⋆ Key Point: Embedding of $t^{(g)} - t^{(g')}$ is invertible for $g \neq g'$.
- Trapdoor $a_0$ in public key has random tag prefix of length $c_{i^*}$.
- Choice of $i^*$ via confined guessing [BHJ+14,DM14]
  - ⋆ $\mathcal{A}$ makes $Q$ queries, succeeds with probability $\epsilon$
  - ⋆ Choose smallest $i^*$ such that $2Q^2/\epsilon \leq 2^{c_{i^*}}$
  - ⋆ $Q$ random tags will have distinct length $c_i^*$ prefixes with prob $\epsilon/2$
- $i^*$ must be kept secret from $\mathcal{A}$

# Computing $g$ Homomorphically

$$a_0 \\ \vdots \\ \hat{t}_{i^*}$$

- Evaluation of $g$: Subtract $i^*$th prefix of $t^{(g)}$ from $\hat{t}_{i^*}$.

# Computing $g$ Homomorphically

$a_0$

$\vdots$

$\hat{t}_{i^*}$

- Evaluation of $g$: Subtract $i^*$th prefix of $t^{(g)}$ from $\hat{t}_{i^*}$.
- Selection of prefix of $t^{(g)}$ needs to be done homomorphically

# Computing $g$ Homomorphically

| | | | | | | |
|---|---|---|---|---|---|---|
| $a_0$ | $a_1$ | $a_2$ | $\cdots$ | $a_{i^*}$ | $\cdots$ | $a_d$ |
| $\hat{t}_{i^*}$ | $0$ | $0$ | | $1$ | | $0$ |

- Evaluation of $g$: Subtract $i^*$th prefix of $t^{(g)}$ from $\hat{t}_{i^*}$.
- Selection of prefix of $t^{(g)}$ needs to be done homomorphically
- Old way: [DM14] Length $d$ indicator vector for $i^*$ in pk

# Computing $g$ Homomorphically

$a_0$ $\quad$ $b$

$\vdots$ $\quad$ $\vdots$

$\hat{t}_{i^*}$ $\quad$ $i^*$

- Evaluation of $g$: Subtract $i^*$th prefix of $t^{(g)}$ from $\hat{t}_{i^*}$.
- Selection of prefix of $t^{(g)}$ needs to be done homomorphically
- Old way: [DM14] Length $d$ indicator vector for $i^*$ in pk
- Our way: Store $i^*$ as tag in pk

# Computing $g$ Homomorphically

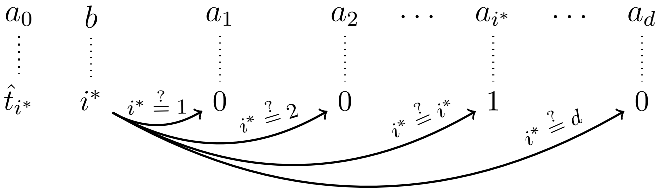
$a_0$ $\quad$ $b$

$\vdots$ $\quad$ $\vdots$

$\hat{t}_{i^*}$ $\quad$ $i^*$

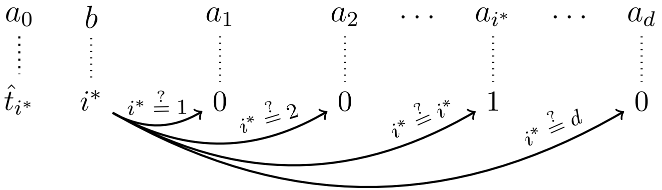
- Evaluation of $g$: Subtract $i^*$th prefix of $t^{(g)}$ from $\hat{t}_{i^*}$.
- Selection of prefix of $t^{(g)}$ needs to be done homomorphically
- Old way: [DM14] Length $d$ indicator vector for $i^*$ in pk
- Our way: Store $i^*$ as tag in pk
  - Degree $d-1$ interpolation polynomial computes $i \stackrel{?}{=} i^*$

# Computing $g$ Homomorphically



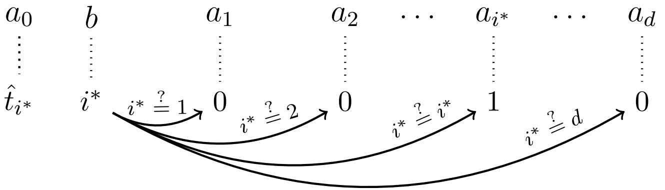

- Evaluation of $g$: Subtract $i^*$th prefix of $t^{(g)}$ from $\hat{t}_{i^*}$.
- Selection of prefix of $t^{(g)}$ needs to be done homomorphically
- Old way: [DM14] Length $d$ indicator vector for $i^*$ in pk
- Our way: Store $i^*$ as tag in pk
  - Degree $d-1$ interpolation polynomial computes $i \stackrel{?}{=} i^*$
  - Can compute vector homomorphically from $b$ alone.

# Computing $g$ Homomorphically




- Evaluation of $g$: Subtract $i^*$th prefix of $t^{(g)}$ from $\hat{t}_{i^*}$.
- Selection of prefix of $t^{(g)}$ needs to be done homomorphically
- Old way: [DM14] Length $d$ indicator vector for $i^*$ in pk
- Our way: Store $i^*$ as tag in pk
  - Degree $d-1$ interpolation polynomial computes $i \stackrel{?}{=} i^*$
  - Can compute vector homomorphically from $b$ alone.
- Asymmetric trapdoor growth in multiplication is key

# Computing $g$ Homomorphically




- Evaluation of $g$: Subtract $i^*$th prefix of $t^{(g)}$ from $\hat{t}_{i^*}$.
- Selection of prefix of $t^{(g)}$ needs to be done homomorphically
- Old way: [DM14] Length $d$ indicator vector for $i^*$ in pk
- Our way: Store $i^*$ as tag in pk
  - Degree $d-1$ interpolation polynomial computes $i \stackrel{?}{=} i^*$
  - Can compute vector homomorphically from $b$ alone.
- Asymmetric trapdoor growth in multiplication is key
  - Yields $d^d \operatorname{poly}(n)$ growth instead of $d^d n^{\log d}$ growth

# Concluding Thoughts

- Open problems:

# Concluding Thoughts

- Open problems:
  - Short signatures with short public keys with less trapdoor growth?

# Concluding Thoughts

- Open problems:
  - Short signatures with short public keys with less trapdoor growth?
  - Can techniques be extended to achieve fully secure IBE?

# Concluding Thoughts

- Open problems:
  - ⋆ Short signatures with short public keys with less trapdoor growth?
  - ⋆ Can techniques be extended to achieve fully secure IBE?

Hiring? Talk to me!