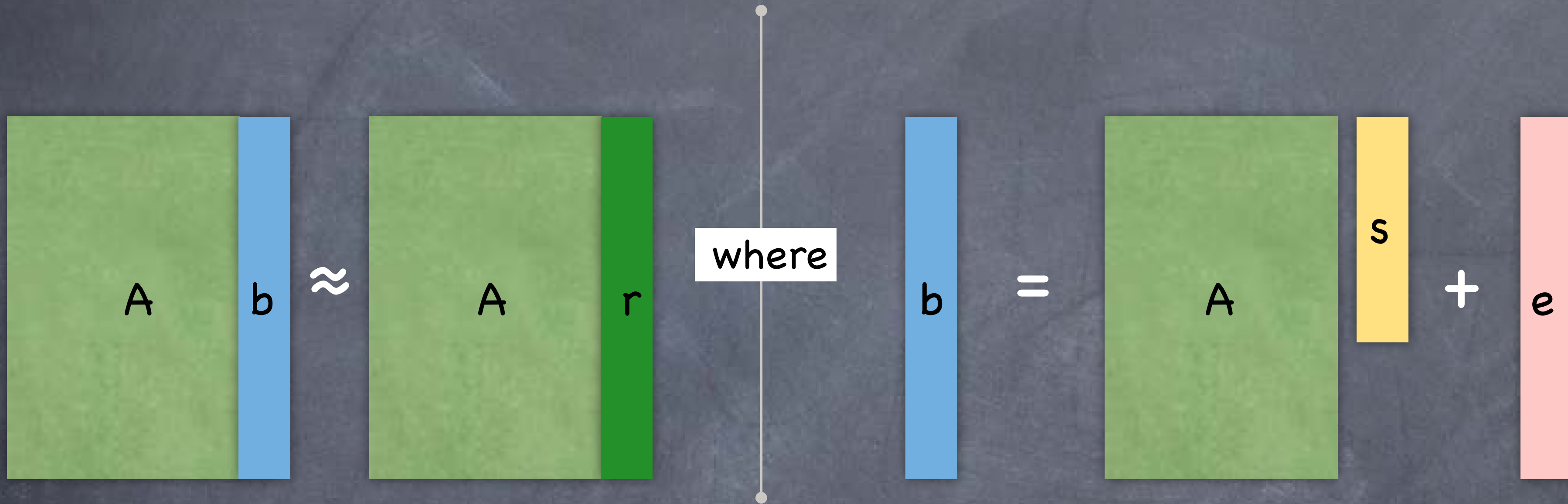
Lattice Cryptography: Towards Fully Homomorphic Encryption

Lecture 20

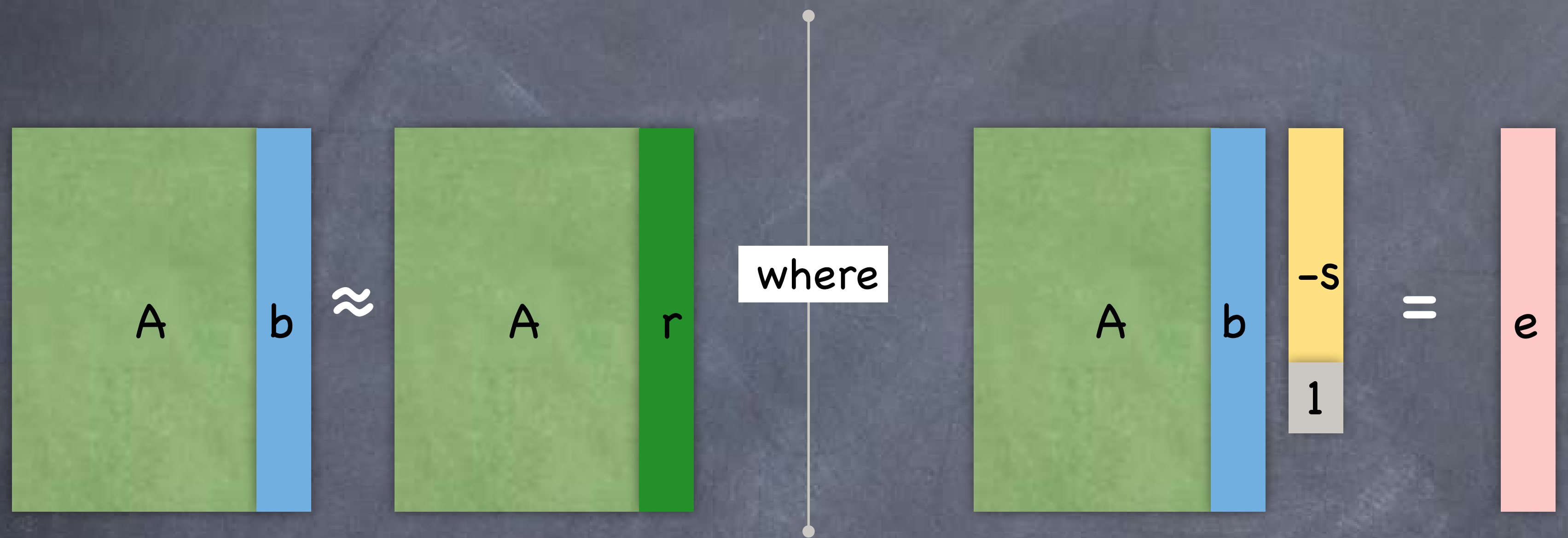
Recall

Learning With Errors



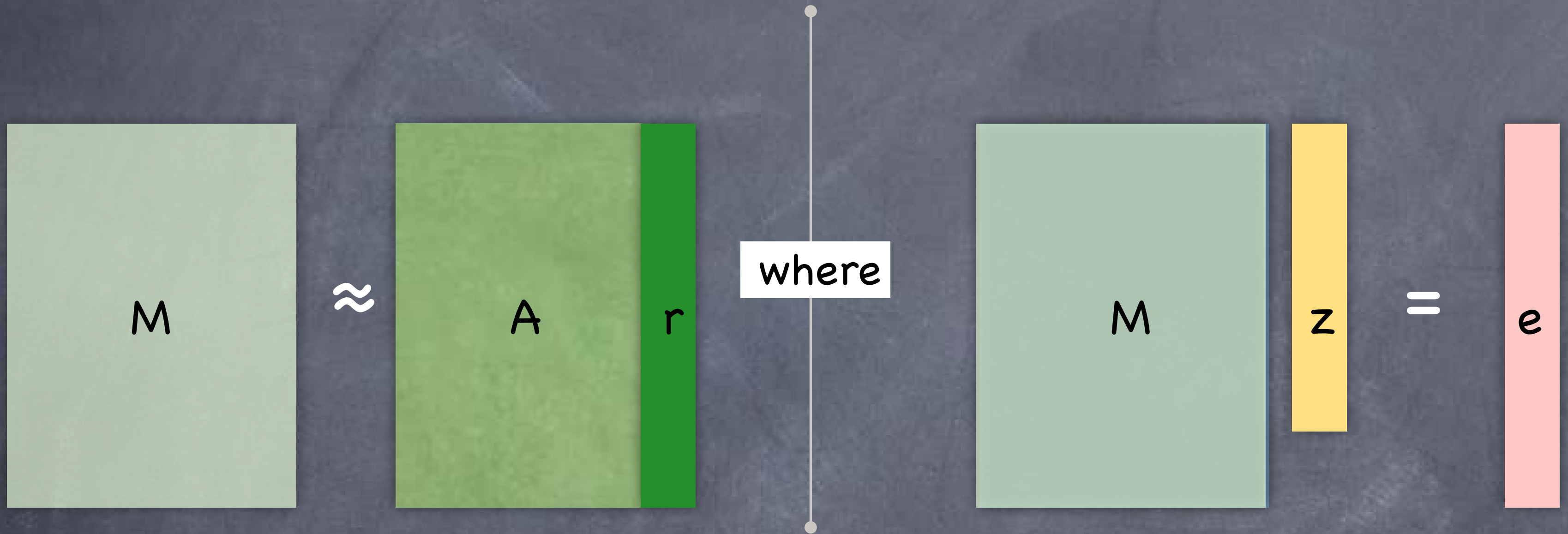
- LWE (decision version): $(A, A\underline{s}+\underline{e}) \approx (A, \underline{r})$, where A random matrix in $A \in \mathbb{Z}_q^{m \times n}$, $\underline{s}$ uniform, $\underline{e}$ has "small" entries from a Gaussian distribution, and $\underline{r}$ uniform.
- Average-case solution for LWE ⇒ Worst-case solution for GapSVP (for appropriate choice of parameters)

Learning With Errors



- LWE (decision version): $(A, A\underline{s}+\underline{e}) \approx (A, \underline{r})$, where A random matrix in $A \in \mathbb{Z}_q^{m \times n}$, $\underline{s}$ uniform, $\underline{e}$ has "small" entries from a Gaussian distribution, and $\underline{r}$ uniform.
- Average-case solution for LWE $\Rightarrow$ Worst-case solution for GapSVP (for appropriate choice of parameters)

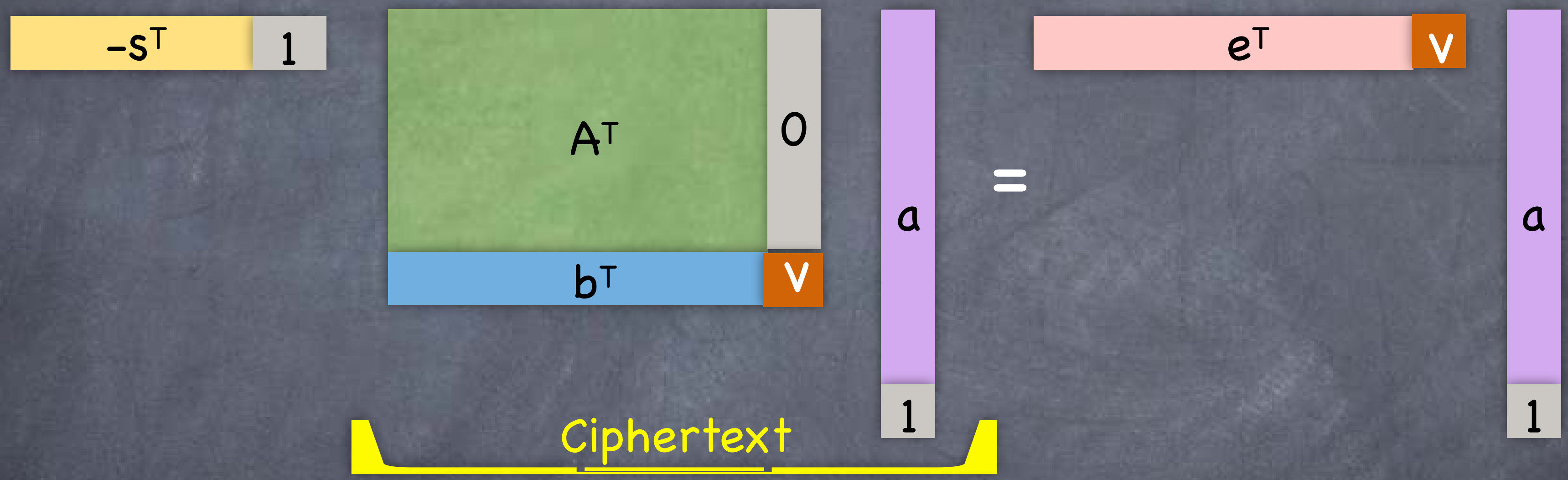
Learning With Errors



- i.e., a pseudorandom matrix $M \in \mathbb{Z}_q^{m \times n'}$ and non-zero $\underline{\mathbf{z}} \in \mathbb{Z}_q^{n'}$ s.t. entries of $M\underline{\mathbf{z}}$ are all small ($n'=n+1$)

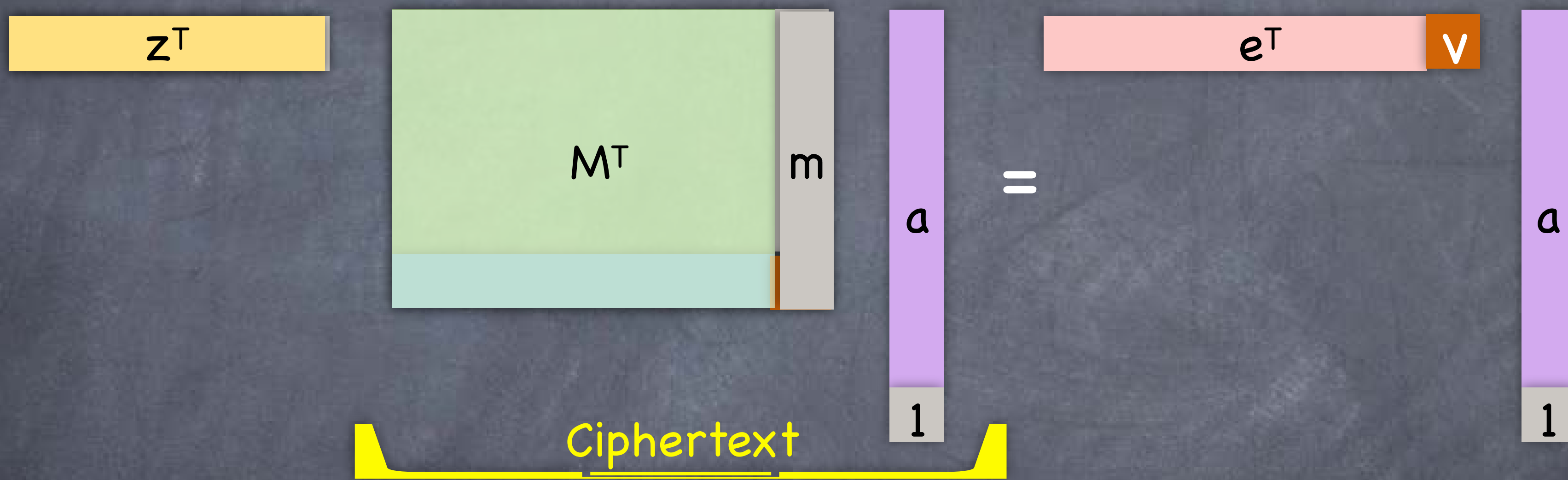
Recall

PKE from LWE



Recall

PKE from LWE



- Ciphertext = $M^T\mathbf{a} + \mathbf{m}$ where $\mathbf{m}$ encodes the message and $\mathbf{a} \in \{0,1\}^m$
- Decryptng: From $\mathbf{z}^T(M^T\mathbf{a} + \mathbf{m}) = \mathbf{e}^T\mathbf{a} + \mathbf{z}^T\mathbf{m}$ where $\mathbf{e}^T\mathbf{a}$ is small. To allow decoding from this for, say $\mu \in \{0,1\}$, let $\mathbf{z}^T\mathbf{m} = v \approx \mu(q/2)$.
- CPA security: $M^T\mathbf{a}$ is pseudorandom
  - Claim: If $M \in \mathbb{Z}_q^{m\times n'}$ is truly random, $\mathbf{a} \in \{0,1\}^m \setminus \{0^m\}$, $m >> n' \log q$, then $M^T\mathbf{a}$ is very close to being uniform

Randomness Extraction

- Entries in $\underline{\mathbf{a}}$ are not uniformly random over $\mathbb{Z}_q^m$, but concentrated on a small subset $\{0,1\}^m$. We need $M^T\underline{\mathbf{a}}$ to be uniform over $\mathbb{Z}_q^{n'}$
- Follows from two more generally useful facts:
  - $H_M(\underline{\mathbf{a}}) = M^T\underline{\mathbf{a}}$ is a 2-Universal Hash Function (for non-zero $\underline{\mathbf{a}}$)
  - If H is a 2-UHF, then it is a good **randomness extractor**
    - If $m \gg n' \log q$, the entropy of $\underline{\mathbf{a}}$ (m bits) is significantly more than that of a uniform vector in $\mathbb{Z}_q^{n'}$ and a good randomness extractor will produce an almost uniform output

Universal Hashing

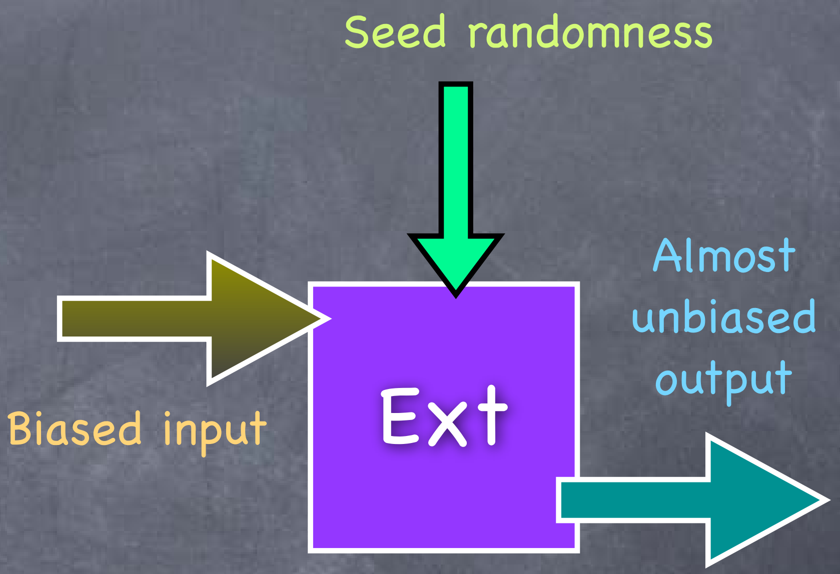
- Combinatorial HF: A→(x,y); h←$\mathcal{H}$. h(x)=h(y) w.n.p
- Even better: 2-Universal Hash Functions
  - "Uniform" and "Pairwise-independent"
  - $\forall x,z\ \Pr_{h\leftarrow\mathcal{H}}[\ h(x)=z\ ] = 1/|Z|$ (where $h:X\rightarrow Z$)
  - $\forall x\neq y,w,z\ \Pr_{h\leftarrow\mathcal{H}}[\ h(x)=w,\ h(y)=z\ ] = 1/|Z|^2$
    - $\Rightarrow\ \forall x\neq y\ \Pr_{h\leftarrow\mathcal{H}}[\ h(x)=h(y)\ ] = 1/|Z|$

x	$h_1(x)$	$h_2(x)$	$h_3(x)$	$h_4(x)$
0	0	0	1	1
1	0	1	0	1
2	1	0	0	1

Negligible collision-probability if super-polynomial-sized range

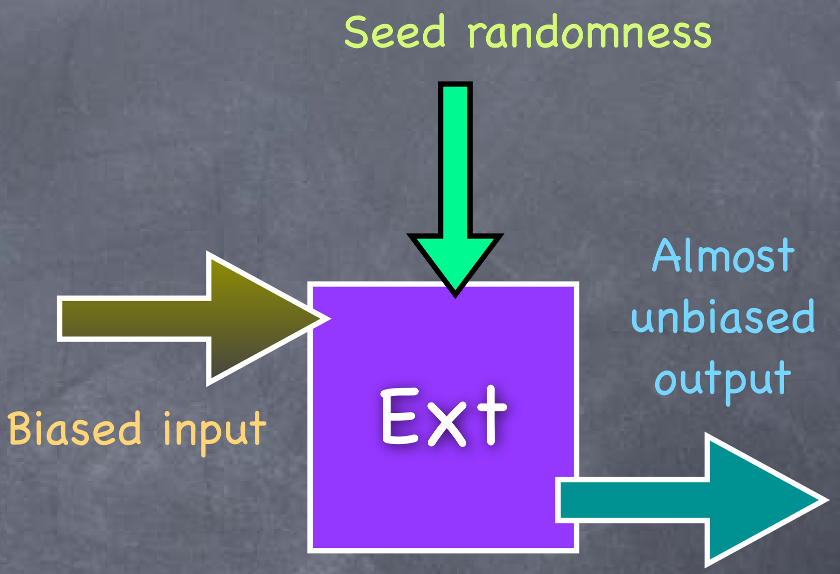
- e.g. $h_{a,b}(x) = ax+b$ (in a finite field, X=Z)
  - $\Pr_{a,b}[\ ax+b = z\ ] = \Pr_{a,b}[\ b = z-ax\ ] = 1/|Z|$
  - $\Pr_{a,b}[\ ax+b = w,\ ay+b = z] =$? Exactly one (a,b) satisfying the two equations (for $x\neq y$)
    - $\Pr_{a,b}[\ ax+b = w,\ ay+b = z] = 1/|Z|^2$
- Exercise: M$\underline{x}$ (M random matrix) is a 2-UHF for non-zero boolean $\underline{x}$

Randomness Extractor



- Input has high "min-entropy"
  - i.e., probability of any particular input string is very low
- Seed uniform and independent of input
- Output vector is shorter than the input
- Ext(inp,seed)) ≈ Uniform
  - Statistical closeness
- A **strong extractor**: (seed, Ext(inp,seed)) ≈ (seed,Uniform)
  - i.e., for any input distribution, most choices of seed yield a good deterministic extractor

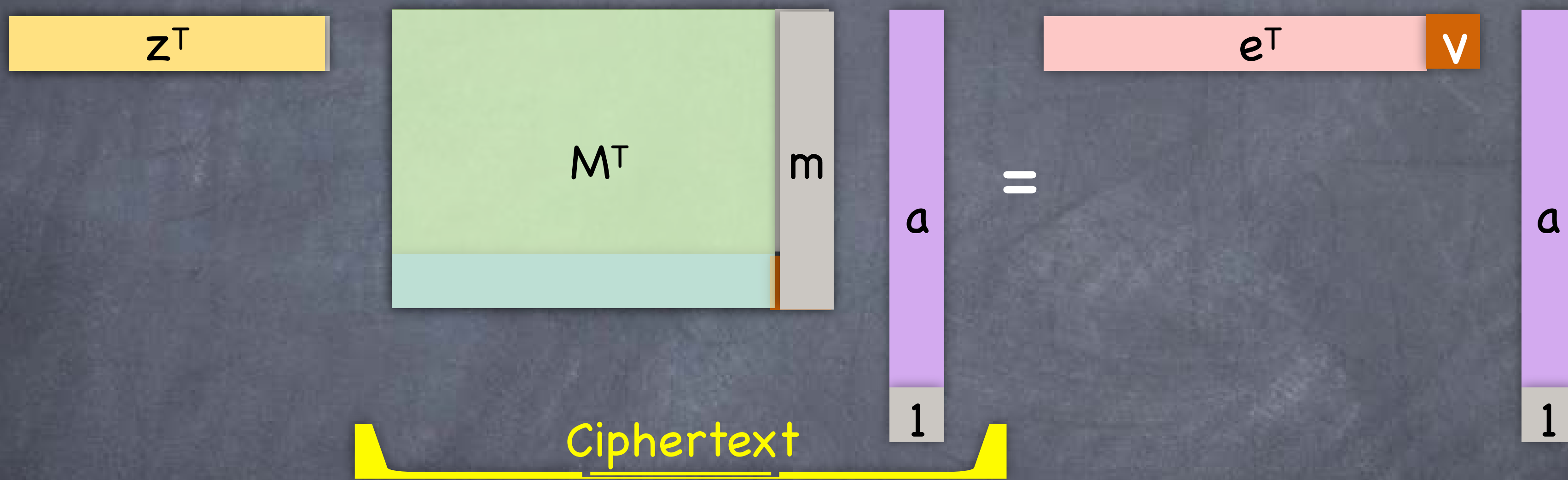
Randomness Extractor



- Leftover Hash Lemma:
  - Any 2-UHF is a strong extractor that can extract almost all of the min-entropy in the input
- A very useful result
  - We need only a special case here:
    - Only for a particular 2-UHF ($H_M(\underline{\mathbf{x}}) = M\underline{\mathbf{x}}$)
    - Only for a particular input distribution ($\underline{\mathbf{x}}$ uniform over $\{0,1\}^m$)

Recall

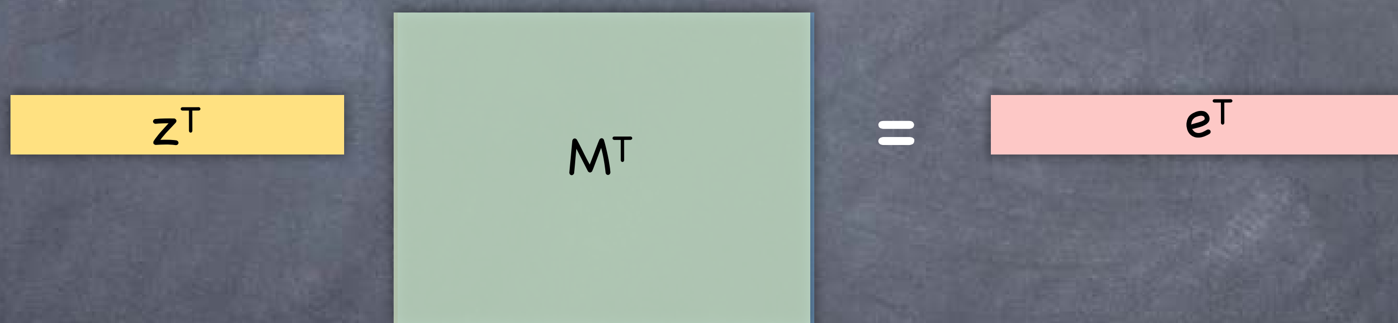
PKE from LWE



- Ciphertext = $M^T\underline{\mathbf{a}} + \underline{\mathbf{m}}$ where $\underline{\mathbf{m}}$ encodes the message and $\mathbf{a} \in \{0,1\}^m$
- Decryptng: From $\underline{\mathbf{z}}^T(M^T\underline{\mathbf{a}} + \underline{\mathbf{m}}) = \underline{\mathbf{e}}^T\underline{\mathbf{a}} + \underline{\mathbf{z}}^T\underline{\mathbf{m}}$ where $\underline{\mathbf{e}}^T\underline{\mathbf{a}}$ is small. To allow decoding from this for, say $\mu \in \{0,1\}$, let $\underline{\mathbf{z}}^T\underline{\mathbf{m}} = v \approx \mu(q/2)$.
- CPA security: $M^T\underline{\mathbf{a}}$ is pseudorandom
  - Claim: If $M \in \mathbb{Z}_q^{m \times n'}$ is <u>truly random</u>, $\mathbf{a} \in \{0,1\}^m \setminus \{0^m\}$, $m >> n' \log q$, then $M^T\underline{\mathbf{a}}$ is very close to being <u>uniform</u>

Gentry-Sahai-Waters

- Want to allow homomorphic operations on the ciphertext
- Idea: Ciphertext is a matrix masked by a pseudorandom matrix that can be "annihilated" with secret key. Addition and multiplication of messages given by addition and multiplication of ciphertexts.
- Recall from LWE: $M \in \mathbb{Z}_q^{m\times n}$ and $\underline{\mathbf{z}} \in \mathbb{Z}_q^{n}$ s.t. $\underline{\mathbf{z}}^T M^T$ has small entries



- First attempt: Public-Key = M, Secret-key = $\underline{\mathbf{z}}$
    - $\mathrm{Enc}(\mu) = M^T R + \mu I$ where $\mu \in \{0,1\}$, $R \leftarrow \{0,1\}^{m\times n}$, and $I_{n\times n}$ identity
    - Security: LWE (and LHL) $\Rightarrow$ $M^T R$ is pseudorandom
    - $\mathrm{Dec}_z(C)$: $z^T C = e^T R + \mu z^T$ has "error" $\underline{\delta}^T = e^T R$. Can recover μ since error has small entries (w.h.p.)

Gentry-Sahai-Waters

- First attempt:
  - $Enc(\mu) = M^T R + \mu I$
  - $Dec_z(C)$: $z^T C = e^T R + \mu z^T$ has error $\underline{\delta}^T = e^T R$
  - $C_1 + C_2 = M^T(R_1 + R_2) + (\mu_1 + \mu_2) I$ has error $\underline{\delta}^T = \underline{\delta}_1^T + \underline{\delta}_2^T$
    - Error adds up with each operation
    - OK if there is an a priori bound on the <u>depth</u> of computation: Levelled Homomorphic Encryption
  - $C_1 \times C_2$: Error = ?
    - $z^T C_1 C_2 = (\underline{\delta}_1^T + \mu_1 z^T) C_2 = \underline{\delta}_1^T C_2 + \mu_1(\underline{\delta}_2^T + \mu_2 z^T)$
    - Error = $\underline{\delta}_1^T C_2 + \mu_1 \underline{\delta}_2^T$
    - Problem: Entries in $\underline{\delta}_1^T C_2$ may not be small, as entries in C_2 are not small! (Since $\mu_1 \in \{0,1\}$, $\mu_1 \underline{\delta}_2^T$ does have small entries)

Gentry-Sahai-Waters

- Problem: Entries in $\underline{\delta}_1^T C_2$ may not be small
- Solution Idea: Represent ciphertext as bits!
  - But homomorphic operations will be affected
  - Observation: Reconstructing a number from bits is a linear operation
  - If $\alpha \in \mathbb{Z}_q^m$ has bit-representation $B(\alpha) \in \{0,1\}^{km}$ $(k=O(\log q))$, then $G\, B(\alpha) = \alpha$, where $G \in \mathbb{Z}_q^{m \times km}$ (all operations in $\mathbb{Z}_q$)
    - B can be applied to matrices also as $B : \mathbb{Z}_q^{m \times n} \to \mathbb{Z}_q^{km \times n}$ and we have $G\, B(\alpha) = \alpha$

Gentry-Sahai-Waters

The Actual Scheme

- Supports messages $\mu \in \{0,1\}$ and NAND operations up to an a priori bounded depth of NANDs
- Public key $M \in \mathbb{Z}_q^{m\times n}$ and private key $\underline{\mathbf{z}}$ s.t. $\underline{\mathbf{z}}^T M$ has small entries
- **Enc(μ)** = $M^T R + \mu G$ where $R \leftarrow \{0,1\}^{m\times km}$ (and $G \in \mathbb{Z}_q^{n\times km}$ the matrix to reverse bit-decomposition)
- **Dec$_z$(C)** : $\underline{\mathbf{z}}^T C = \underline{\delta}^T + \mu \underline{\mathbf{z}}^T G$ where $\underline{\delta}^T = e^T R$

Decrypting G yields 1

- **NAND(C_1,C_2)** : $G - C_1 \cdot B(C_2)$
  - $\underline{\mathbf{z}}^T C_1 \cdot B(C_2) = \underline{\mathbf{z}}^T C_1 \cdot B(C_2) = (\underline{\delta}_1^T + \mu_1 \underline{\mathbf{z}}^T G)\, B(C_2)$

    $= \underline{\delta}_1^T B(C_2) + \mu_1 \underline{\mathbf{z}}^T C_2 = \underline{\delta}^T + \mu_1\mu_2 \underline{\mathbf{z}}^T G$

    where $\underline{\delta}^T = \underline{\delta}_1^T B(C_2) + \mu_1 \underline{\delta}_2^T$ has small entries

Only "left depth" counts, since $\underline{\delta} \leq k \cdot m \cdot \underline{\delta}_1 + \underline{\delta}_2$

  - In general, error gets multiplied by km. Allows depth $\approx \log_{km} q$