recall: $B_i = AR_i + x_i^* G$
so $AR_i = B_i - x_i^* G$

$z_1^T = s^T A + e_1^T$

$z_1^T [R_1 | \cdots | R_\ell] = [s^T AR_1 + e_1^T R_1 \ | \cdots | \ s^T AR_\ell + e_1^T R_\ell]$

$= s^T [B_1 - x_1^* G \ | \cdots | \ B_\ell - x_\ell^* G] + e_1^T [R_1 | \cdots | R_\ell]$

$z' + \mu_0 \cdot \lfloor \frac{q}{2} \rceil = s^T p + e' + \mu_0 \cdot \lfloor \frac{q}{2} \rceil$

This is the distribution in Hyb_2.

Alternatively if z_1 and z_2 are uniform, then we have the distribution in Hyb_3.

Claim now follows by hybrid argument: Hyb_3 is independent of μ_0. Can apply same transitions in reverse to encrypt μ_1.

Key idea: Program x^* into the public key.
This yields a trapdoor for $[A \mid B_f]$ whenever $f(x^*) = 1$.
And ensures semantic security whenever $f(x^*) = 0$.

Predicate encryption: Want ciphertexts to additionally hide the attribute
- Weak attribute hiding: successful decryption also recovers attribute
- Strong attribute hiding: attribute remains hidden even if decryption succeeds
  ↳ implies functional encryption!

We will focus on the setting of weak attribute-hiding.

Key idea: Combine FHE with ABE. We will encrypt the attribute under ABE and homomorphically evaluate the predicate.
Challenge: How to decrypt the output of the predicate? We will use a "dual-use" technique where the underlying schemes share a common secret key.

First, we will generalize our homomorphic evaluation relations to support matrix-valued computations
- So far: for a function $f: \{0,1\}^\ell \to \{0,1\}$:

$$[B_1 | \cdots | B_\ell] \cdot H_f = B_f$$
$$[B_1 - x_1 G \mid \cdots \mid B_\ell - x_\ell \cdot G] \cdot H_{f,x} = B_f - f(x) \cdot G$$

- Suppose that $f: \{0,1\}^\ell \to \mathbb{Z}_q^{n \times m}$ is a matrix-valued function. Then, we will describe an analogous relation:

$$[B_1 | \cdots | B_\ell] \cdot H_f = B_f$$
$$[B_1 - x_1 G \mid \cdots \mid B_\ell - x_\ell \cdot G] \cdot H_{f,x} - f(x) \quad \text{where} \quad x = (x_1, \ldots, x_\ell)$$

- We take a bit by bit approach.

Let $f_{j,k}: \{0,1\}^\ell \to \{0,1\}$ be function that computes k^{th} bit of j^{th} entry of $f(x)$

Then, $[B_1 - x_1 G \mid \cdots \mid B_\ell - x_\ell G] \cdot H_{f_{j,k}, x} = B_{f_{j,k}} - [f(x)]_{j,k} \cdot G$ — input-dependent evaluation

↖ k^{th} bit of j^{th} element of $f(x)$

$$= [B_1 | \cdots | B_\ell] \cdot H_{f_{j,k}} - [f(x)]_{j,k} \cdot G$$

Let $E_j \in \mathbb{Z}_q^{n \times m}$ be the matrix that is 1 in position j (where j ranges over all $n \cdot m$ indices)

Then, we can write $f(x) = \sum_{j \in [n \cdot m]} \sum_{k \in [\log q]} [f(x)]_{j,k} \cdot 2^k E_j$

↖ bits of $f(x)$

Thus, we can write

$$\sum_{j,k} [B_1 - x_1 G \mid \cdots \mid B_\ell - x_\ell G] \cdot H_{f_{j,k},x} \cdot G^{-1}(2^k E_j) = \sum_{j,k} [B_1 \mid \cdots \mid B_\ell] \cdot H_{f_{j,k}} \cdot G^{-1}(2^k E_j) - \sum_{j,k} [f(x)]_{j,k} \cdot G \cdot G^{-1}(2^k E_j)$$

$$= [B_1 \mid \cdots \mid B_\ell] \cdot \sum_{j,k} H_{f_{j,k}} \cdot G^{-1}(2^k E_j) - f(x)$$

We thus define

$$H_{f,x} = \sum_{j,k} H_{f_{j,k,x}} \cdot G^{-1}(2^k E_j) \quad \text{and} \quad H_f = \sum_{j,k} H_{f_{j,k}} \cdot G^{-1}(2^k E_j)$$

Then, for a function $f : \{0,1\}^\ell \longrightarrow \mathbb{Z}_q^{n \times m}$, we have

$$[B_1 - x_1 G \mid \cdots \mid B_\ell - x_\ell G] \cdot H_{f,x} = [B_1 \mid \cdots \mid B_\ell] \cdot H_f - f(x)$$

Generalized matrix evaluation!

where $\|H_f\|, \|H_{f,x}\| \leq (n \log q)^{O(d)}$

Predicate encryption from LWE (combining ABE and FHE):

Setup$(1^\lambda, 1^\ell)$: $(A, td) \leftarrow \text{TrapGen}(n, q)$

Sample $B_1, \ldots, B_L \xleftarrow{R} \mathbb{Z}_q^{n \times m}$, $p \xleftarrow{R} \mathbb{Z}_q^n$ [$L = \text{poly}(\ell, n, \log q)$ — exact length determined by scheme] (length of attribute)

Output mpk $= (A, B_1, \ldots, B_\ell, p)$ and msk $= td$ (same as for ABE)

Encrypt(mpk, x, μ): Sample $s \xleftarrow{R} \mathbb{Z}_q^n$, $e \leftarrow \chi^m$

Compute the GSW ciphertext (GSW public key)

$$T_i = \begin{bmatrix} A \\ s^T A + e^T \end{bmatrix} R_i + x_i \cdot G$$

[GSW encryption of x_i]

where $R_i \leftarrow \{0,1\}^{(n+1)\log q \times (n+1)\log q}$.

Let $t_1, \ldots, t_L$ be the binary representation of $T = [T_1 \mid \cdots \mid T_\ell]$ (encryption of $x = (x_1, \ldots, x_\ell)$)

We now encode the bits of $t_1, \ldots, t_L$:

$c_0^T \leftarrow s^T A + e_0^T$ $\qquad e_0 \leftarrow \chi^m$

$c_j^T \leftarrow s^T [B_j - t_j \cdot \bar{G}] + e_j^T$ $\qquad e_j \leftarrow \chi^m$

gadget matrix G without the last row $[\bar{G} \in \mathbb{Z}_q^{n \times (n+1)\log q}]$:

$$\bar{G} = \left[\begin{array}{ccc|c} 1 \;\, 2 \cdots 2^{\lceil \log q \rceil - 1} & & & \\ & \ddots & & 0^{n \times \log q} \\ & & 1 \;\, 2 \cdots 2^{\lceil \log q \rceil - 1} & \end{array}\right]$$

Compute $c' \leftarrow s^T p + \mu \cdot \lfloor \frac{q}{2} \rceil + e'$ $\qquad e' \leftarrow \chi$

Output the ciphertext $ct = (T, c_0, c_1, \ldots, c_L, c')$

KeyGen(msk $\overset{=\text{td}}{}$, f): Let $T = [T_1 | \cdots | T_L]$ be an encryption of $x = (x_1, \dots, x_\ell)$.

Let $T_f := \text{FHE.Eval}(f, T)$

and let $\overline{T}_f$ be T_f excluding the last row. ← [T_f is a GSW encryption of $f(x)$.] Recall GSW decryption: $-s^T \overline{T}_i + \underline{t}_i = e^T R_i + x_i [-s^T \overline{G} | g] \approx x_i \cdot [-s^T | 1] \cdot G$

$\underline{t}_f$ be the last row of T_f:

$$T_f = \begin{bmatrix} \overline{T}_f \\ \hdashline \underline{t}_f \end{bmatrix}$$

Let $\hat{f}$ be the circuit that maps $T \mapsto \overline{T}_f$ — $\hat{f}$ homomorphically evaluates f on T and outputs all but the last row of T_f (this is a matrix-valued function)

Let $B_1, \dots, B_L, \hat{f} \longmapsto B_{\hat{f}}$

and use td to sample short z_f such that

$$[A | B_{\hat{f}}] \cdot z_f = p$$

Output the secret key $sk_f = z_f$

Decrypt((sk_f, f), ct): Homomorphically evaluate $\hat{f}$ on the encoding $[c_1 | \cdots | c_L]$

$$c_{\hat{f}} \leftarrow [c_1 | \cdots | c_L] \cdot H_{\hat{f}, T}$$

Homomorphically compute $T_f \leftarrow \text{FHE.Eval}(f, T)$ and let $\underline{t}_f$ be the last row of T_f

Compute $c' - [c_0 | c_{\hat{f}} + \underline{t}_f] \cdot z_f$ and round the result

Correctness: By construction, $c_{\hat{f}} = [c_1 | \cdots | c_L] \cdot H_{\hat{f},T} = s^T [B_1 - t_1 \overline{G} | \cdots | B_L - t_L \overline{G}] H_{\hat{f},T} + [e_1^T | \cdots | e_L^T] H_{\hat{f},T}$

$$\approx s^T (B_{\hat{f}} - \overline{T}_f)$$

$$\begin{aligned} c_{\hat{f}} + \underline{t}_f &\approx s^T B_{\hat{f}} - s^T \overline{T}_f + \underline{t}_f \\ &= s^T B_{\hat{f}} + [-\bar{s}^T | 1] T_f \\ &\approx s^T B_{\hat{f}} + f(x) \cdot [-s | 1] G \end{aligned}$$

since T_f is a GSW ciphertext encrypting $f(x)$ under the same secret key s

When $f(x) = 0$, then $c_{\hat{f}} + \underline{t}_f \approx s^T B_{\hat{f}}$ and

$$[c_0 | c_{\hat{f}} + \underline{t}_f] z_f \approx s^T [A | B_{\hat{f}}] z_f = p$$

Then $c' - [c_0 | c_{\hat{f}} + \underline{t}_f] z_f \approx \mu \cdot \lfloor \frac{q}{2} \rceil$ and decryption succeeds.

Key idea: Using ABE evaluation (for matrix-valued relations), we can compute

$$s^T [B_1 - x_1 G | \cdots | B_\ell - x_\ell G] H_{f,x} = s^T (B_f - f(x) \cdot G)$$

Evaluating this requires knowledge of x (to construct $H_{f,x}$).

To hide the attribute x, we encrypt x and homomorphically evaluate the FHE evaluation function

$$\text{Enc}(x) \longmapsto \text{Enc}(f(x))$$

Now, if s is also the GSW secret key, then $s^T f(x)$ "effectively" implements GSW decryption

] "dual use" = same s used for GSW and ABE

Security. Follows by a similar argument as in ABE security (embed encryption of x^* into public parameters)