
Protocol $\text{ROTe}_{\xi,\ell}(x) \rightarrow (m_0, m_1, m_x)$

Maliciously secure ROT extension protocol from SoftSpokenOT [Roy22] for a batch with ξ messages of $\ell \times \kappa$ bits each, setting $\eta = \xi \times \ell \times \kappa$ and $\mu = \eta / \sigma$. It uses a pseudo-random generator $\text{PRG}: \mathbb{Z}_2^\kappa \mapsto \mathbb{Z}_2^{\eta'}$ for $\eta' = \eta + \sigma$ (e.g., $\text{TmHash}_{\eta'}$), a transcript T (e.g., Scheme ??), a base OT (BBOT) and a hash $\mathsf{H}: \mathbb{Z}_2^\kappa \mapsto \mathbb{Z}_2^\kappa$

Players: sender \mathcal{S} , and receiver \mathcal{R}

Inputs: $\mathcal{R}: x \in \mathbb{Z}_2^\xi$, the choice bits

Outputs: $\mathcal{S}: m_0, m_1 \in \mathbb{Z}_2^\eta$, pairs of random messages

$\mathcal{R}: m_x \in \mathbb{Z}_2^\eta$, chosen messages such that

$$m_{x(i,j)} = m_{0(i,j)}x(i) \oplus m_{1(i,j)}(1 - x(i)) \quad \forall i \in [\xi] \quad \forall j \in [\ell]$$

$\mathcal{S} \& \mathcal{R}.\text{Setup}() \dashrightarrow \mathcal{S}: (k_0, k_1); \mathcal{R}: k_b$

- 1: \mathcal{S} samples $b \xleftarrow{\$} \mathbb{Z}_2^\kappa$ as the base OT choice bits
- 2: \mathcal{R} runs $\text{BBOT}_\kappa()$ as the base OT sender, obtaining $k_0, k_1 \in \mathbb{Z}_2^{\kappa \times \kappa}$
- 3: \mathcal{S} runs $\text{BBOT}_\kappa(b)$ as the base OT receiver, receiving $k_b \in \mathbb{Z}_2^{\kappa \times \kappa}$

$\mathcal{R}.\text{Round1}(x \in \mathbb{Z}_2^\xi) \dashrightarrow u, \dot{x}, \dot{t}, m_x$

- 1: Set $x_{\text{rep}} \leftarrow \{\{x_{(i)}, x_{(i)}, \dots, x_{(i)}\}_{i \in [\xi]}\}$ by repeating ℓ times x
 - 2: Sample $x_\sigma \xleftarrow{\$} \mathbb{Z}_2^\sigma$ and concatenate $x' \leftarrow x_{\text{rep}} \parallel x_\sigma$
 - 3: Extend $(t_0, t_1) \leftarrow \{\text{PRG}(k_{0(i)}), \text{PRG}(k_{1(i)})\}_{i \in [\kappa]}$ with $t_0, t_1 \in \mathbb{Z}_2^{\kappa \times \eta'}$
 - 4: $u \leftarrow \{\{t_{0(i,j)} \oplus t_{1(i,j)} \oplus x'_{(j)}\}_{j \in [\eta']}\}_{i \in [\kappa]}$ with $u \in \mathbb{Z}_2^{\kappa \times \eta'}$
 - 5: Run $\mathsf{T}.\text{Append}(u)$ and $\chi \leftarrow \mathsf{T}.\text{Extract}(\eta)$ to get the challenge $\chi \in \mathbb{Z}_2^{\mu \times \sigma}$
 - 6: Compute the challenge response (\dot{x}, \dot{t}) as:
 - 6.a: $\dot{x} \leftarrow \{x_{\sigma(k)} \oplus \bigoplus_{m=1}^\mu \chi(m,k) \cdot x_{\text{rep}(\sigma m + k)}\}_{k \in [\sigma]}$ with $\dot{x} \in \mathbb{Z}_2^\sigma$
 - 6.b: $\dot{t} \leftarrow \{\{t_{0(i,\eta+k)} \oplus \bigoplus_{m=1}^\mu \chi(m,k) \cdot t_{0(i,\sigma m + k)}\}_{k \in [\sigma]}\}_{i \in [\kappa]}$ with $\dot{t} \in \mathbb{Z}_2^{\kappa \times \sigma}$
 - 7: Transpose $t'_0 \leftarrow \{\{t_{0(i,j)}\}_{i \in [\kappa]}\}_{j \in [\eta']}$ with $t'_0 \in \mathbb{Z}_2^{\eta' \times \kappa}$
 - 8: Send $(u, \dot{x}, \dot{t}) \rightarrow \mathcal{S}$
 - 9: $m_x \leftarrow \{\{\mathsf{H}(j \parallel t'_{0(j\ell+l)})\}_{l \in [\ell]}\}_{j \in [\eta]}$
- return m_x**

$\mathcal{S}.\text{Round2}(u, \dot{x}, \dot{t}) \dashrightarrow (m_0, m_1)$

- 1: Extend $t_b \leftarrow \{\text{PRG}(k_{b(i)})\}_{i \in [\kappa]}$ with $t_b \in \mathbb{Z}_2^{\kappa \times \eta'}$
 - 2: $q \leftarrow \{\{b_{(i)} \cdot u_{(i,j)} \oplus t_{b(i,j)}\}_{j \in [\eta']}\}_{i \in [\kappa]}$ with $q \in \mathbb{Z}_2^{\kappa \times \eta'}$
 - 3: Run $\mathsf{T}.\text{Append}(u)$ and $\chi \leftarrow \mathsf{T}.\text{Extract}(\eta)$ to get the challenge $\chi \in \mathbb{Z}_2^{\mu \times \sigma}$
 - 4: Verify the challenge response:
 - 4.a: $\dot{q} \leftarrow \{\{q_{(i,\eta+k)} \oplus \bigoplus_{m=1}^\mu \chi(m,k) \cdot q_{(i,\sigma m + k)}\}_{k \in [\sigma]}\}_{i \in [\kappa]}$ with $\dot{q} \in \mathbb{Z}_2^{\kappa \times \sigma}$
 - 4.b: Check $q_{(i,k)} \stackrel{?}{=} \dot{q}_{(i,k)} \quad \forall i \in [\kappa] \quad \forall k \in [\eta']$; otherwise **ABORT**
 - 5: Transpose $q' \leftarrow \{\{q_{(i,j)}\}_{i \in [\kappa]}\}_{j \in [\eta']}$ with $q' \in \mathbb{Z}_2^{\eta' \times \kappa}$
 - 6: $(m_0, m_1) \leftarrow \{\{\mathsf{H}(j \parallel q'_{(j)}), \mathsf{H}(j \parallel (q_{(j\ell+l)} \oplus b_{(j\ell+l)}))\}_{l \in [\ell]}\}_{j \in [\xi]}$
- return (m_0, m_1)**
-

References

- [Roy22] Lawrence Roy. Softspokenot: Quieter ot extension from small-field silent vole in the minicrypt model. In *Advances in Cryptology–CRYPTO 2022: 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15–18, 2022, Proceedings, Part I*, pages 657–687. Springer, 2022.