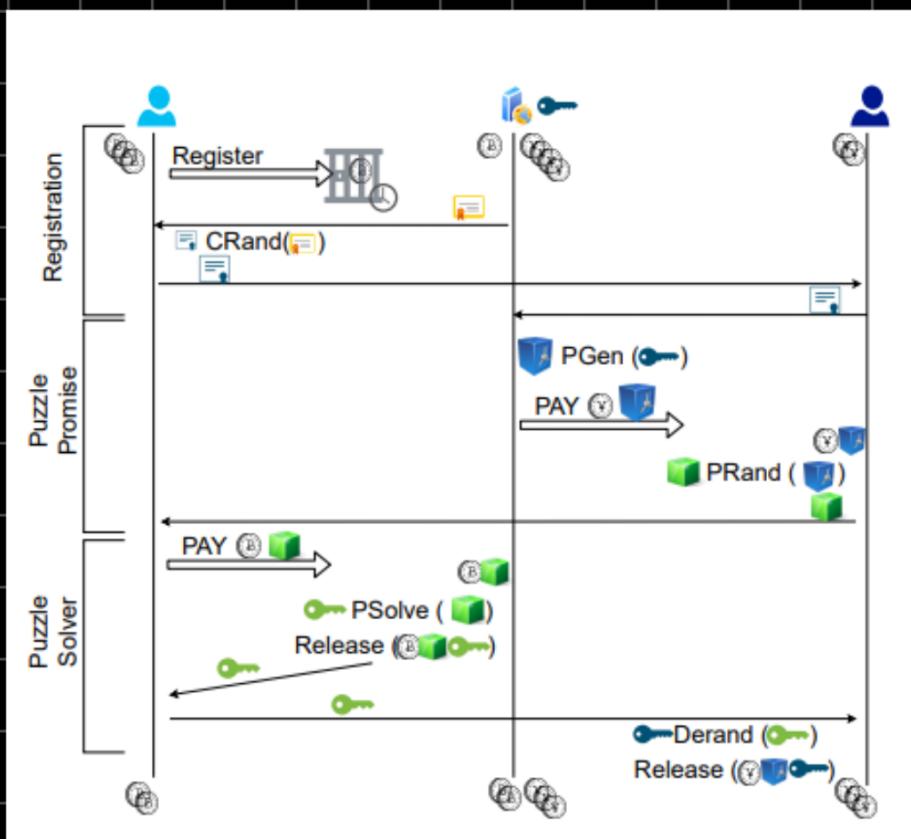


Core criptográfico do protocolo:

- Criptografia homomórfica
 - ↳ Esquema Castagnos - Laguillaumie (CL)
- Adaptor Signatures
 - ↳ Usando Schnorr / ECDSA
- Non-interactive Zero-Knowledge Proof (NIZK)
 - ↳ Para provar que α em A e Z^0 é o mesmo e não lixo matemático.
 - Blind randomizable Signature scheme.

Estrutura geral do protocolo:



O mesmo segredo $\alpha \in Z_g$ é usado no puzzle e na adaptor Sig.

- Registration: Garantir que P_s têm as moedas e a intenção de pagar P_f .
 - ↳ Evita ataques Dos por parte do P_f no tumbler (P_f). Briefing attacks.
 - ↳ Geração do segredo randomizado. P_s envia para P_f e P_f usa a credencial (NIZK) para pedir puzzle ao tumbler.
- Puzzle Promise: É gerado um puzzle e uma adaptar Signature de $P_f \rightarrow P_f$. (α)
 - ↳ P_f randomiza um puzzle com um fator secreto β e envia para o P_s por um canal privado. (α')
- Puzzle Solver:
 - P_s randomiza novamente o puzzle (ϵ sig), para manter irastreabilidade caso tumbler e receiver estejam em conluio. (α'')
 - ↳ P_s gera nova adaptar Sig e troca moedas ao P_f para revelar α'' .
 - ↳ Propriedade (L), P_f consegue achar α'' usando sua $privKey$ mesmo sem ter β e π .

↳ P_f destrava adaptor S_{if} usando α'' e pega moedas.

↳ P_s recebe α'' , remove fator β e envia para P_r .

↳ P_r recebe α' remove τ e tem α , então destrava a adaptador S_{if} e finalmente pega suas moedas.

Objetivos alcançados:

- Atomicidade: Ou todas as etapas ocorrem ou nenhuma ocorre.
- Irreversibilidade: Todas as etapas garantem que seja impossível "linkar" P_s com P_r .
 - ↳ Ao contrário da LN onde o hash transmitido nas HTLC (Hash Time-locked contracts) é o mesmo em todos os hops. Isso dá certa rastreabilidade mesmo com Onion Routing, especialmente em caminhos curtos ou se a mesma entidade controla mais de um nó no trajeto.

- Autenticidade: O PGM (tumbler) só consegue o pagamento se Ps (sender) se registraram com Pt, trouxeram moedas e receberam sua certificações. Isso é feito para evitar griefing attacks (DoS).
- Outros: Zero hops, liquidez alta e imediata, taxas menores, transações instantâneas (por conta de ser zero hops), simplicidade de uso, tudo isso mantendo irrefutabilidade e atomicidade, sem os trade-offs de PCMs.

Estruturação/Overview das Primitivas:

Propriedades de criptografia homomórfica: Puzzle

$$c = \text{Enc}(\alpha) \quad \forall \alpha \in \mathbb{Z}_q \pmod{q}$$

$$c^\beta = \text{Enc}(\alpha \cdot \beta)$$

$$c' = \text{Enc}(\alpha') \rightarrow \text{Randomização 1}$$

$$c'^\tau = \text{Enc}(\alpha' \cdot \tau)$$

$$c'' = \text{Enc}(\alpha'') \rightarrow \text{Randomização 2}$$

Usando propriedade de CL:

$\text{Dec}(\psi'') = \alpha'' \Rightarrow$ Usando chave privada

$$\alpha' = \alpha'' \cdot \gamma^{-1}$$

$$\alpha = \alpha' \cdot \beta^{-1}$$

Adaptor Signatures:

$\hat{\sigma} = \sigma - \alpha$; onde $\hat{\sigma}$ é uma assinatura inválida
pois faltou o segredo α , solução do
puzzle.

É gerado junto com o puzzle:

$A = g^\alpha$; onde α é ~impossível de encontrar
por conta do log discreto, e g é a forma
quadrática geradora (de ordem q).

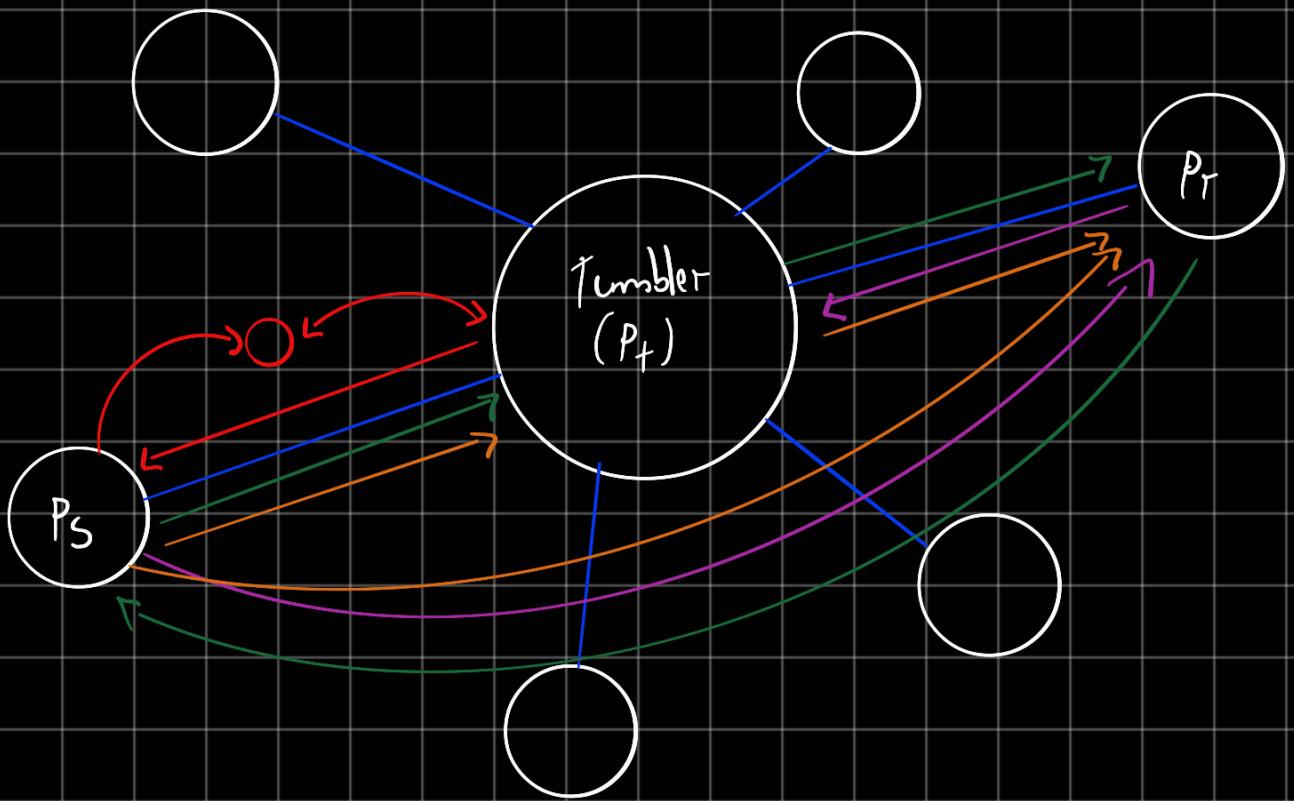
Logo, um puzzle é composto por:

$$\mathcal{Z} = (A, c)$$

$\sim // \sim$

Apêndice:

Arquitetura dos PCKs e A2L é a seguinte:



- Camais
- Fase de registro e credencial
- Envio de credencial $Ps \rightarrow Pr$ e aprovação de Pt
- Gerado do puzzle + Sigs e seu fluxo de vida.
- Desenvolvimento de adaptor Sigs e fluxo dos segredos d.