Obfuscation

$P \xrightarrow{Obf} \tilde{P}$

P: program, circuit, TM

$\tilde{P}$: program

Correctness: $\forall P : \Pr\left[\overbrace{\forall x : P(x) = \tilde{P}(x)}^{P \equiv \tilde{P}}\right] = 1 - \mathrm{negl}(\lambda)$

$\tilde{P} \leftarrow Obf(1^\lambda, P)$

Security:* Anything one can learn from $\tilde{P}$ one can also learn from black-box access to P.

Goals:

- software protection
- software patching
- symmetric-key enc $\Rightarrow$ public key enc.
- FHE ....
- RO = Obfuscate a PRF

✗ Ideal Olf: $\exists$ PPT Sim s.t. $\forall P$

$$\tilde{P} \leftarrow Olf(1^\lambda, P) \approx Sim^P(1^\lambda)$$

$$P_\alpha(x) = \begin{cases} 1 & \text{if } x=\alpha \\ 0 & \text{else} \end{cases}$$

✗ VBB Olf: $\forall$ PPT A, $\exists$ PPT Sim $\forall P$

$$\left| Pr[A(Olf(1^\lambda, P))=1] - Pr[Sim^P(1^\lambda, 1^{|P|})=1] \right| \leq negl(\lambda)$$

- Impossible for TMs.

"self eating programs"

$$P_{\alpha,\beta,\gamma}(x) := \begin{cases} \beta & \text{if } x=\alpha \\ \gamma & \text{if } x(\alpha)=\beta \\ \perp & \text{else} \end{cases}$$

If α, β, γ random $\leftarrow \{0,1\}^\lambda$

1. Black box access $P_{\alpha,\beta,\gamma}$ always output $\perp$.

2. Given $\tilde{P} \equiv P_{\alpha,\beta,\gamma}$ then can learn γ.

$\tilde{P}(\tilde{P})$

- Impossible for circuits

  Assume FHE exists (can be constructed from VBB)

$$C_{\alpha,\beta,\gamma,sk,r}(x) = \begin{cases} ct = Enc_{sk}(\alpha; r) & \text{if } x = 0^n \\ \beta & \text{if } x = \alpha \\ \gamma & \text{if } Dec_{sk}(x) = \beta \\ \perp & \text{else} \end{cases}$$

1. Given oracle access $C_{\dots}$ can't learn γ.

2. Given any $\tilde{C} \equiv C_{\dots}$ can recover γ: $Eval(\tilde{C}, \tilde{C}(0)) = Enc_{sk}(\beta)$

$$\gamma \leq \tilde{C}\left(Enc_{sk}(P)\right)$$

Obf: $\forall$ PPT A, ~~$\exists$ PPT~~ Sim

$\forall$ P

$$\left| Pr\left[A(Obf(1^\lambda, P)) = 1\right] - Pr\left[Sim^{P}(1^\lambda, 1^{|P|}) = 1\right]\right| \leq negl(\lambda)$$

$\Leftrightarrow$

- indistinguishability obfuscation (circuits)

$\forall \quad C \equiv C', \quad |C| = |C'|$

$$Obf(1^\lambda, C) \approx Obf(1^\lambda, C')$$

Candidate constructions of IO.

[Garg et al. 2013], ...

Is IO useful? Yes!!!!

IO = "Best possible obfuscation"

Assume Obf' "good obfuscation scheme"

then if Obf is an IO scheme ...

$$Obf(pad(C)) \approx Obf(\underline{Obf'(C)})$$

same size

PKE from IO + PRG : $\{0,1\}^\lambda \to \{0,1\}^{2\lambda}$

$KeyGen(1^\lambda) \to (sk, pk)$

$sk \leftarrow \{0,1\}^\lambda$

$pk = PRG(sk)$

$Enc_{pk}(m)$: $Obf(1^\lambda, C_{pk,m})$

$C_{pk,m}(x)$: {

if $PRG(x) = pk$

then output m

else $\perp$

}

$Dec_{sk}(c)$: $C(sk)$

Security :

$(pk, \; Enc_{pk}(m) = \tilde{C}_{pk,m}) : pk = PRG(sk)$

$\tilde{C} = Obf(1^\lambda, C_{pk,m})$

$\approx$ PRG

$(pk', \; \tilde{C}_{pk',m}) : pk' \leftarrow \{0,1\}^{2\lambda}$

$\approx$ IO

$(pk', \; \tilde{C}_\perp = Obf(1^\lambda, C_\perp))$

Puncturable PRFs

$F(K, x)$

$K\{x\} \leftarrow Puncture(K, x)$

correct: $F(k, x') = F(k\{x\}, x')$

$\forall\ x' \neq x$

security: $(k\{x\}, F(k, x))$

$\approx$

$(k\{x\}, \$)$

Construction from OWFs

(GGM construction)

NIKE: $[crs \leftarrow CRSGen(1^\lambda)]$

$pk \leftarrow KeyGen(sk)$

$S \leftarrow SharedKey(pk_1, \ldots, pk_\ell, sk_i)$

correctness: $\forall i,j$ $SharedKey(pk_i, sk_i) = SharedKey(pk_j, sk_j)$

Security : $(pk_1, \ldots, pk_\ell, S) \approx (pk_1, \ldots, pk_\ell, \$)$

Construction from iO, puncturable PRF

$pk = PRG(sk)$

$crs = \tilde{C} \leftarrow iO(r, C_k)$

$C_k(\overline{pk}, sk_i)$:

if $\exists\, i$ s.t. $PRG(sk_i) = pk_i$

output $F(K, \overline{pk})$

else $\perp$

$\text{Sim}_{\text{real}}(\tilde{C}, \overline{pk}, sk_i) = \tilde{C}(\overline{pk}, sk_i)$

$(crs, \overline{pk}, S) \approx (crs, \overline{pk}, \$)$

① $pk_i \leftarrow \{0,1\}^{2\lambda}$ // PRG

② $\tilde{C}$ use $K\{\overline{pk}\}$ // iO

(3) change $S = F(K, \overline{PW})$ to $\$$ // punctured PRF