Exam May 5, 2020

solutions by Luigi Russo, 1699981

1 A PRG Candidate

Let $f : \{0,1\}^\lambda \to \{0,1\}^\lambda$ be a one-way permutation, and $G : \{0,1\}^\lambda \to \{0,1\}^{\lambda+l}$ be a pseudorandom generator with positive stretch (i.e., $l \geq 1$). Analyze the following derived construction of a pseudorandom generator $G'_f : \{0,1\}^\lambda \to \{0,1\}^{2\lambda+l}$, where $G'(s) := (f(s), G(s))$.

In case you think the derived construction is not secure, exhibit a concrete attack; otherwise, provide a proof of security.

SOLUTION

G' is not secure for any possible instantiation. Indeed, assume G be constructed from the OWP f, by using as hardcore predicate Goldreich-Levin l times.

Namely, let $G(s) = h(f(s))||h(f(f(s)))|| \dots ||h(f(\dots(f(s))))||f(\dots(f(s)))$, where h is the Goldreich-Levin hard-core predicate for f. In this case we can easily break the security of G', since we are given both $G(s)$ and $f(s)$.

2 PKE Combiners

Let $\Pi_1 = (\mathsf{KGen}_1, \mathsf{Enc}_1, \mathsf{Dec}_1)$ and $\Pi_2 = (\mathsf{KGen}_2, \mathsf{Enc}_2, \mathsf{Dec}_2)$ be two PKE schemes with the same message space $\mathcal{M} = \{0,1\}^n$. You know that at least one of the two PKE schemes is secure, but you don't know which one. Show how to combine Π_1 and Π_2 into a PKE scheme $\Pi = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$, with message space $\mathcal{M}$, such that Π satisfies IND-CPA security as long as at least one of Π and Π_2 satisfies IND-CPA security.

SOLUTION

Let $\Pi = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ be the following:

- KGen: it returns (pk, sk), with $\mathsf{pk} = (\mathsf{pk}_1, \mathsf{pk}_2)$, $sk = (\mathsf{sk}_1, \mathsf{sk}_2)$, where $(\mathsf{pk}_i, sk_i) \leftarrow_\$ \mathsf{KGen}_i$
- $\mathsf{Enc}(pk, m)$: returns (c_1, c_2), where $c_1 = \mathsf{Enc}_1(pk_1, (m \oplus r))$ and $c_2 = \mathsf{Enc}_2(pk_2, r)$ for $r \leftarrow_\$ \mathcal{M}$.
- $\mathsf{Dec}(sk, (c_1, c_2))$: it returns $\mathsf{Dec}_1(sk_1, c_1) \oplus \mathsf{Dec}_2(sk_2, c_2)$

Let assume Π_i is IND-CPA secure. If Π is not, then there exists a valid PPT adversary $\mathcal{A}$ that we can use as follows to build a PPT attacker $\mathcal{A}_i$ to Π_i:

- $\mathcal{A}_i$ generates $(\mathsf{pk}_{3-i}, \mathsf{sk}_{3-i}) \leftarrow_\$ \mathsf{KGen}_{3-i}(1^\lambda)$
- $\mathcal{A}_i$ receives (pk_i) from the challenger and forwards to $\mathcal{A}$ the new public key $\mathsf{pk} = (\mathsf{pk}_1, \mathsf{pk}_2)$

- $\mathcal{A}$ produces the challenge (m_0, m_1). $\mathcal{A}_i$ samples $r \leftarrow_\$ \mathcal{M}$ and forwards to the challenger the new pair $(m_0 \oplus r, m_1 \oplus r)$
- the challenger chooses at random a bit b and encrypts m_b, thus producing the ciphertext c_b^*
- $\mathcal{A}_i$ gives $\mathcal{A}$ the new ciphertext (c_1, c_2), where $c_i = c_b^*$ and $c_{3-i} \leftarrow_\$ \mathsf{Enc}_{3-i}(pk_{3-i}, r)$
- $\mathcal{A}$ returns a bit b', which is passed to the challenger

3 ID scheme based on RSA

Consider the following ID scheme $\Pi = (\mathsf{KGen}, \mathcal{P}, \mathcal{V})$.

- The key generation algorithm first computes parameters (N, e, d) as in the RSA cryptosystem. In particular, $N = p \cdot q$ for sufficiently large primes p, q, and $e \cdot d \equiv 1 \mod (p-1)(q-1)$ with e a prime number. Hence, it picks $x \leftarrow_\$ \mathbb{Z}_N^*$, computes $y = x^e \mod N$, and it returns $\mathsf{pk} = (N, e, y)$ and $\mathsf{sk} = x$.
- One execution of the ID scheme goes as follows:
  (1) The prover $\mathcal{P}$ picks $a \leftarrow_\$ \mathbb{Z}_N^*$, and sends $\alpha = a^e \mod N$ to $\mathcal{V}$;
  (2) The verifier $\mathcal{V}$ forwards a random challenge $\beta \leftarrow_\$ \mathbb{Z}_e$ to $\mathcal{P}$;
  (3) The prover $\mathcal{P}$ replies with $\gamma = x^\beta \cdot a \mod N$, where x is taken from the secret key and a is the same value sampled in the first round.
- The verifier $\mathcal{V}$ accepts a transcript $\tau = (\alpha, \beta, \gamma)$ if and only if

$$\gamma^e \cdot y^{-\beta} = \alpha \mod N.$$

Prove that Π is a canonical ID scheme satisfying completeness, special soundness and honest-verifier zero knowledge under the RSA assumption.

(**Hint:** To prove special soundness you can use the following fact: Given N, elements $u, v \in \mathbb{Z}_N^*$ and integers e, e' for which it holds that $gcd(|e|, |e'|) = 1$ and $u^e = v^{e'} \mod N$, an e-th root of v (modulo N) can be computed in polynomial time.)

SOLUTION

(completeness) $\forall x, \tau$ we have that $\gamma^e \cdot y^{-\beta} = (x^\beta \cdot a)^e \cdot y^{-\beta} = a^e = \alpha$

(special soundness) Assume there exists $\mathcal{A}$ able to produce two transcripts $\tau_1 = (\alpha, \beta_1, \gamma_1)$, $\tau_2 = (\alpha, \beta_2, \gamma_2)$. Then we have that $\gamma_1^e \cdot y^{-\beta_1} = \gamma_2^e \cdot y^{-\beta_2}$. We derive that $(\frac{\gamma_1}{\gamma_2})^e = y^{\beta_1 - \beta_2} \mod N$. We can compute[1] the e-th root of $y^{\beta_1 - \beta_2}$ that is equal to $x^{e \cdot (\beta_1 - \beta_2)} = x^{\beta_1 - \beta_2}$.

honest-verifier zero knowledge The simulator $\mathcal{S}(pk)$ produces the transcript $\tau = (\alpha, \beta, \gamma)$ in the following way:

- samples $\beta \leftarrow_\$ \mathbb{Z}_e$
- samples $\gamma \leftarrow_\$ \mathbb{Z}_N^*$
- computes $\alpha = \gamma^e \cdot y^{-\beta}$

[1] we use the hint