Technische Universität München (I7)
Dr. M. Luttenberger / M. Schlund

Winter 2013/14

SOLUTION

Cryptography – Endterm

Exercise 1 **One Liners** **1.5P each = 12P**

For each of the following statements, state if it is true or false **and** give a *short* (one line) justification of your answer (e.g. sketch the argument or give a counter-example).

Example: "If the RSA problem is hard w.r.t. $\mathsf{Gen}\mathbb{P}^2$, then PRGs with variable strecth exist" is true because then the RSA problem yields a OWP family to which we can apply the Blum-Micali construction.

(a) If pseudorandom functions (PRF) exist, then CCA-secure ES exist.

(b) If pseudorandom functions (PRF) exist, then pseudorandom generators (PRG) exist.

(c) If DLP is hard w.r.t. $\mathsf{Gen}\mathbb{Z}^*_{\text{safe}}$, then ElGamal using $\mathsf{Gen}\mathbb{Z}^*_{\text{safe}}$ is CPA-secure.

(d) If the RSA problem is hard, then computing the Carmichael function $\lambda(N)$ for $N = pq$ (p, q prime) is hard.

(e) If the DLP is hard w.r.t. $\mathsf{Gen}\mathbb{QR}_{\text{safe}}$, then collision resistant hash functions exist.

(f) If $P \neq NP$, then computationally secret ES cannot exist.

(g) If F_k is a PRF, the cascading construction $F^*_k(x)$ together with the CBC-padding $F^*_k(\mathsf{pad}_{\text{CBC}}(m))$ yields a secure MAC.

(h) If F_k is a PRP and G is a PRG, then $\mathsf{Enc}_k(m) := F_k(m \oplus G(k+1))$ yields a CPA-secure ES.

Solution:

(a) True, one can combine F-rCTR (CPA-secure) and F-MAC via the Enc-then-Mac-construction to get a CCA-secure ES.

(b) True, $G(k) := F_k(\lfloor 1 \rceil)||F_k(\lfloor 2 \rceil)|| \cdots F_k(\lfloor s \rceil)$ is a PRG (of stretch $n \cdot s$).

(c) False, one can distinguish messages from $\mathbb{QR}_p$ and $\mathbb{Z}^*_p \setminus \mathbb{QR}_p$ via the Legendre symbol in polynomial time (and using ElGamal, the probability that a message from $\mathbb{QR}_p$ "stays within" $\mathbb{QR}_p$ is 3/4).

(d) True, otherwise we could compute the inverse $d = e^{-1} \bmod \lambda(N)$ via the extended euclidean algorithm.

(e) True, using the DLP-CCF and Merkle-Damgård-construction we can build a collision-resistant hash function.

(f) False, the OTP is perfectly secret and thus also computationally secret.

(g) True, the cascading construction $F^*_k(x)$ yields a secure MAC if the adversary can only pose prefix-free queries and the CBC-padding is prefix-free.

(h) False, the scheme is stateless and deterministic, hence cannot be CPA-secure.

Exercise 2 **2P+3P = 5P**

(a) Let $G : \{0,1\}^n \to \{0,1\}^{2n}$ be a PRG. Show that $G' : \{0,1\}^n \to \{0,1\}^{2n}$ with $G'(x) := \overline{G(x)}$ is also a PRG ($\overline{\cdot}$ is the bitwise negation).

(b) Prove or disprove: Let $G_1, G_2 : \{0,1\}^n \to \{0,1\}^{2n}$ be PRGs with $G_1 \neq G_2$. Then $G : \{0,1\}^n \to \{0,1\}^{4n}$ defined as $G(x) := G_1(x)||G_2(x)$ is a PRG.

Solution:

(a) Given any distinguisher $\mathcal{A}$ for G' we define a distinguisher $\mathcal{D}$ for G as follows:

- Input $y \in \{0,1\}^{2n}$
- $r \xleftarrow{r} \mathcal{A}(\overline{y})$
- output r

Then $\mathcal{P}[\mathsf{Win}^{\text{IndPRG}}_{n,G}(\mathcal{D})] = \mathcal{P}[\mathsf{Win}^{\text{IndPRG}}_{n,G'}(\mathcal{A})]$ which is negligible since G is a PRG. Hence, G' is a PRG.

(b) The statement is false: Take $G_1, \overline{G_1}$ for some PRG G_1. By the result of the previous exercise, $\overline{G_1}$ is a PRG. However, the following distinguisher $\mathcal{D}$ will distinguish $G(x) = G_1(x)||\overline{G_1}(x)$ from a truly random string:

- Input $y = y_1||y_2 \in \{0,1\}^{2n}$
- If $y_2 = \overline{y_1}$ then output 1, else output 0.

Its success probability is

Case $b = 0$: $\mathcal{P}_{b=0}[\mathsf{Win}^{\text{IndPRG}}_{n,G}(\mathcal{D})] = 1 - 2^{-n}$

Case $b = 1$: $\mathcal{P}_{b=1}[\mathsf{Win}^{\text{IndPRG}}_{n,G}(\mathcal{D})] = 1$

So altogether $\mathcal{P}[\mathsf{Win}^{\text{IndPRG}}_{n,G}(\mathcal{D})] = 1 - 2^{-(n+1)}$ which is not negligible. Hence, G is not a PRG.

Exercise 3 — 3P

Consider the following message authentication code built from a PRF F of key and block length n:

- Gen: On input 1^n, output $k \xleftarrow{u} \{0,1\}^n$.
- Mac: Let $\mathcal{M}_n := \{m \in \{0,1\}^* \mid n \mid |m| \wedge |m| < 2^n\}$.

  On input $k \in \{0,1\}^n$ and $m \in \mathcal{M}_n$, partition m into subsequent n-bit blocks $m = m^{(1)}|| \ldots ||m^{(l)}$.

  Then output $t := F_k(m^{(1)} \oplus \lfloor 1 \rceil) \oplus F_k(m^{(2)} \oplus \lfloor 2 \rceil) \oplus \cdots \oplus F_k(m^{(l)} \oplus \lfloor l \rceil)$ for some encoding $\lfloor : \rceil \{0, 1, \ldots, 2^n - 1\} \to \{0,1\}^n$.
- Vrf: On input $k \in \{0,1\}^n$, $m \in \mathcal{M}_n$, and $t \in \{0,1\}^n$, output 1 if $\mathsf{Mac}_k(m) = t$, otherwise ouput 0.

Show that this MAC is not secure ("existentially unforgeable under an adaptive chosen-message attack"). E.g. show how to forge a valid tag for the message $0^n||0^n$.

Solution: A simple solution with one query: Query $0^n||0^n|||\lfloor 3 \rceil|||\lfloor 4 \rceil$, get tag t. Then t is valid for $0^n||0^n$.

A different solution with three queries:

- Query 0^n, get $t_0 = F_k(\lfloor 1 \rceil)$.
- Query 1^n, get $t_1 = F_k(\overline{\lfloor 1 \rceil})$.
- Query $1^n||0^n$, get $t_2 = F_k(\overline{\lfloor 1 \rceil}) \oplus F_k(\lfloor 2 \rceil)$.

Then $t = t_0 \oplus t_1 \oplus t_2 = F_k(\lfloor 1 \rceil) \oplus F_k(\lfloor 2 \rceil)$ is valid for $0^n||0^n$.

Here is a much simpler solution:

It is trivial to forge a valid MAC for the message $m = \lfloor 1 \rceil|||\lfloor 2 \rceil$ as $\mathsf{Mac}_k(m) = F_k(\lfloor 1 \rceil \oplus \lfloor 1 \rceil) \oplus F_k(\lfloor 2 \rceil \oplus \lfloor 2 \rceil) = 0^n$. So, we do not even need to query the oracle once.

Exercise 4 — 2P+2P+2P+1P = 7P

Consider the multiplicative group $\mathbb{Z}_p^*$ modulo the prime $p = 53$.

(a) Is 2 a generator of $\mathbb{Z}_p^*$?

(b) Compute the probability that $x \xleftarrow{u} \mathbb{Z}_p^*$ is a generator of $\mathbb{Z}_p^*$.

(c) Let $f_k \colon \mathbb{Z}_p^* \to \mathbb{Z}_p^* \colon x \mapsto (x^k \bmod p)$.

For which $e \in \mathbb{Z}$ exists a $d \in \mathbb{Z}$ such that for all $x \in \mathbb{Z}_p^*$ we have $f_d(f_e(x)) = x$?

(d) Why is RSA not hard w.r.t. $\mathsf{Gen}\mathbb{Z}^*_{\text{prime}}$?

Solution:

(a) Since $|\mathbb{Z}_p^*| = 52 = 2^2 \cdot 13$ we use the generator test

- $2^4 = 16 \neq 1 \bmod 53$
- $2^{26} = \left(2^6\right)^4 \cdot 2^2 = 11^4 \cdot 4 = 13 \cdot 4 = -1 \neq 1 \bmod 53$

Hence 2 is a generator of $\mathbb{Z}_p^*$.

(b) $\mathbb{Z}_p^*$ has $\varphi(\varphi(p))$ many generators, therefor $\mathcal{P}[\langle x \rangle = \mathbb{Z}_p^*] = \frac{\varphi(\varphi(p))}{\varphi(p)} = \frac{24}{52} = \frac{6}{13}$.

(c) f_e is invertible for all e with $\gcd(e, \lambda(p)) = 1$, i.e. for all $e \in \mathbb{Z}$ such that $2 \nmid e$ and $13 \nmid e$.

Exercise 5 — 1P+2P+3P = 6P

Consider the multiplicative group $\mathbb{Z}_n^*$ modulo $n = 13 \cdot 17 = 221$.

(a) Compute the order of $\mathbb{Z}_n^*$.

(b) Compute the exponent of $\mathbb{Z}_n^*$.

(c) Use the CRT to compute a generator of a largest cyclic subgroup of Z_n^*.

Solution:

(a) $|\mathbb{Z}_n^*| = \varphi(n) = 12 \cdot 16 = 192$

(b) $\lambda_{\mathbb{Z}_n^*} = \lambda(n) = \operatorname{lcm}(12, 16) = 48$

(c) $\mathbb{Z}_{221}^* \simeq \mathbb{Z}_{13}^* \times \mathbb{Z}_{17}^*$. We start by finding generators of the two smaller groups.

$\langle 2 \rangle = \mathbb{Z}_{13}^*$ since $2^4 \equiv_{13} 3 \neq 1$ and $2^6 \equiv_{13} -1$. and $\langle 3 \rangle = \mathbb{Z}_{17}^*$ since $3^8 \equiv_{17} 16 \neq 1$.

Hence $(2, 3)$ is a generator of a subgroup of $\mathbb{Z}_{13}^* \times \mathbb{Z}_{17}^*$ of order $\operatorname{lcm}(12, 16) = 48$ (the largest cyclic subgroup).

Since $1 = 4 \cdot 13 - 3 \cdot 17$ (by the extended euclidean algorithm), we get the corresponding generator in $\mathbb{Z}_{221}^*$ via $h^{-1}(2, 3) = 4 \cdot 13 \cdot 3 - 2 \cdot 3 \cdot 17 = 54 \in \mathbb{Z}_{221}^*$.

Exercise 6 — 1P+2P+2P+2P = 7P

Given a pseudorandom permutation (PRP) F with key-length n and block-length $2n$, consider the fixed-length ES $\mathcal{E}$ (partially) defined by

- Gen: On input 1^n return $k \xleftarrow{u} \{0,1\}^n$.
- Enc: On input $k, m \in \{0,1\}^n$, choose $\rho \xleftarrow{u} \{0,1\}^n$ and return $\mathsf{Enc}_k(m) := F_k(\rho || m)$.

(a) Complete the definition of $\mathcal{E}$ by defining Dec.

(b) Above ES $\mathcal{E}$, given a key of length n, can only encrypt messages of length n.

Under the assumption that $\mathcal{E}$ is CPA-secure, describe how to build from above ES $\mathcal{E}$ an ES $\mathcal{E}'$ which (1) can handle messages of arbitrary length (this rules out some padding schemes!), and (2) is also CPA-secure.

(c) Assume further that $\mathcal{E}$ is even CCA-secure. Is then $\mathcal{E}'$ (your answer to (b)) also CCA-secure? Prove your answer!

(d) Show that $\mathcal{E}$ is CPA-secure if F is a PRP. To this end, analyze the success probability of the following PPT-distinguisher $\mathcal{D}$ for F where $\mathcal{A}$ is any PPT-CPA-attack on $\mathcal{E}$.

Definition of $\mathcal{D}^{\mathcal{O}}(1^n)$:

- Let $\mathsf{Enc}^{\mathrm{sim}}$ by the following function:

  On input $m \in \{0,1\}^n$ choose $\rho \xleftarrow{u} \{0,1\}^n$, then output $\mathsf{Enc}^{\mathrm{sim}}(m) := \mathcal{O}(\rho || m)$.
- $m_0, m_1 \xleftarrow{r} \mathcal{A}(1^n)^{\mathsf{Enc}^{\mathrm{sim}}}$.
- Choose $b \xleftarrow{u} \{0,1\}$.
- $c \xleftarrow{r} \mathsf{Enc}^{\mathrm{sim}}(m_b)$.
- $r \xleftarrow{r} \mathcal{A}^{\mathsf{Enc}^{\mathrm{sim}}}(c)$.
- If $r = b$ output 1 ("$\mathcal{O}$ contains F"); else output 0 ("$\mathcal{O}$ contains RPO").

Remarks: Recall $\mathcal{D}$ has access to an oracle $\mathcal{O}$ where $\mathcal{O}$ is either $\mathcal{O}_0$ ("perfect world") or $\mathcal{O}_1$ ("real world"): $\mathcal{O}_0$ is a random permutation oracle (RPO), i.e. on creation it chooses uniformly at random permutation from the set of all permutations of $\{0,1\}^{2n}$ which it uses to answer all queries; $\mathcal{O}_1$ chooses $k \xleftarrow{u} \{0,1\}^n$ on creation and answers all queries using F_k.

Solution:

(a) $\mathsf{Dec}_k(c)$: Compute $F_k^{-1}(c) =: y_1 y_2 \dots y_{2n}$. Then output $y_{n+1} y_{n+2} \dots y_{2n}$.

(b) Let $\mathsf{pad}\colon \{0,1\}^* \to \{0,1\}^*$ be defined as follows: On input m, let $\mathsf{pad}(m)$ be the shortest prefix of $m10^*$ which (1) is a suffix of $m1$ and (2) whose length is a multiple of n.

Gen': as Gen.

Enc': given m, k, compute $\mathsf{pad}(m)$ and split this into subsequent blocks consisting of n bits each: $\mathsf{pad}(m) = m^{(1)}||m^{(2)}||\dots||m^{(l)}$; then output $\mathsf{Enc}'_k(m) := \mathsf{Enc}_k(m^{(1)})||\mathsf{Enc}_k(m^{(2)})||\dots||\mathsf{Enc}_k(m^{(l)})$.

Dec': given c, k, split c into subsequent blocks consisting of $2n$bits each $c^{(1)}||c^{(2)}||\dots||c^{(l)}$; then compute $m' := \mathsf{Dec}_k(c^{(1)})||\dots||\mathsf{D}$ Finally, remove from m' all trailing 0s up to and including the first 1 from the right to obtain the plaintext.

This ES is then still CPA-secure (if F is a (strong) PRP) but not CCA-secure (if F is a strong PRP) as we can e.g. simply remove $2n$-bit blocks from the ciphertext.

(c) (This is much more detailed than what was expected in the exam.)

We have to show the advantage $\varepsilon_{\mathcal{A}}$ of $\mathcal{A}$ in the CPA-game v.s. $\mathcal{E}$ is negligible w.r.t. the key length n.

Recall that

$$\Pr[\mathcal{A} \text{ wins the CPA-game v.s. } \mathcal{E}] = \frac{1}{2} + \varepsilon_{\mathcal{A}} \geq \frac{1}{2}.$$

(We always can assume that $\mathcal{A}$ wins with at least probability 1/2.)

To this end, we need to relate $\varepsilon_{\mathcal{A}}$ to the advantage $\varepsilon_{\mathcal{D}}$ of $\mathcal{D}$ in the PRP-game v.s. F

$$\Pr[\mathcal{D} \text{ wins the PRP-game v.s. } F] = \frac{1}{2} + \varepsilon_{\mathcal{D}} \geq \frac{1}{2}.$$

As F is assumed to be a PRP, we know that $\varepsilon_{\mathcal{D}}$ is negligible.

As always, we make a case distinction w.r.t. the oracle $\mathcal{D}$ is interacting with:

$$\Pr[\mathcal{D} \text{ wins the PRP-game v.s. } F] = \frac{1}{2}\Pr\left[\mathcal{D}^{\mathcal{O}_0}(1^n) = 0\right] + \frac{1}{2}\Pr\left[\mathcal{D}^{\mathcal{O}_1}(1^n) = 1\right]$$

If $\mathcal{O} = \mathcal{O}_1$, then $\mathsf{Enc}^{\mathrm{sim}}$ becomes the encryption method of $\mathcal{E}$ instantiated on some truely random key, that is, $\mathcal{D}$ simply plays the CPA-game (acting as Alice&Bob) vs. $\mathcal{A}$ using the ES $\mathcal{E}$, and by definition of $\mathcal{D}$, we have

$$\Pr\left[\mathcal{D}^{\mathcal{O}_1}(1^n) = 1\right] = \Pr[\mathcal{A} \text{ wins the CPA-game v.s. } \mathcal{E}] = \frac{1}{2} + \varepsilon_{\mathcal{A}}$$

If $\mathcal{O} = \mathcal{O}_0$, then $\mathsf{Enc}^{\mathrm{sim}}$ uses the RPO for computing the encryptions: Here the intuition should be that $\mathcal{D}$ can only win with probability (roughly) 1/2: as long as we never query the RPO for the same input twice, it will choose its answers independently of the actual input uniformly at random from all the remaining possible outputs. Assume for now that we never query the RPO for the same input twice. Then in any computation of $\mathcal{A}$, when we come to the step $c \xleftarrow{r} \mathsf{Enc}^{\mathrm{sim}}(m_b)$ the value of c is chosen independently of the value of b, i.e. the same computation of $\mathcal{A}$ will take place with the same probability for both $b = 0$ and $b = 1$. As $b \xleftarrow{u} \{0,1\}$, the probability that $r = b$ is thus inded exactly 1/2 under the assumption that the RPO is only queried for distinct values.

Note that we certainly won't query the RPO for the same input twice, if we do not choose the same ρ twice, i.e. as long as we do not get a collision in the ρs. Hence, let C be the event that we choose the same ρ twice. Just as in the case of rCTR, $\Pr[C]$ is negligible. Precisely, we have

$$\Pr[C] \leq \frac{q(n)^2}{2^n}$$

where $q(n)$ is the number of encryptions done over the course of the computation of $\mathcal{D}$, so $q(n)$ is a polynomial. Then:

$$\Pr\left[\mathcal{D}^{\mathcal{O}_1}(1^n) = 1 \mid \overline{C}\right] = 1/2$$

In total:

$$1/2 + \varepsilon_{\mathcal{D}} = \Pr[\mathcal{D} \text{ wins the PRP-game v.s. } F] = \frac{1}{2}(1/2 + \varepsilon_{\mathcal{A}}) + \frac{1}{2}(\Pr\left[\mathcal{D}^{\mathcal{O}_1}(1^n) = 1 \mid C\right]\Pr[C] + 1/2(1 - \Pr[C]))$$

This leads to:

$$0 \leq \varepsilon_{\mathcal{A}} = 2\varepsilon_{\mathcal{D}} + (1 - \Pr\left[\mathcal{D}^{\mathcal{O}_1}(1^n) = 1 \mid C\right])\Pr[C] \leq 2\varepsilon_{\mathcal{D}} + \Pr[C]$$

As F is a PRP, $\varepsilon_{\mathcal{D}}$ has to be negligible; further $\Pr[C]$ is negligible. So $\varepsilon_{\mathcal{A}}$ is negligible, too.

Abbreviations

- RO = random oracle
- RPO = random permutation oracle
- OWF = one-way function (family/collection)
- OWP = one-way permutation (family/collection)
- TDP = trapdoor one-way permutation
- PRG = pseudorandom generator
- PRF = pseudorandom function
- PRP = pseudorandom permutation
- TBC = tweakable block cipher
- ES = (PPT) private-key encryption scheme
- PKES = (PPT) public-key encryption scheme
- UOWHF = universal one-way hash function (family/collection)
- CRHF = collision resistant hash function (family/collection)
- MAC = (PPT) message authentication code
- DSS = (PPT) digital signature scheme
- DLP = discrete logarithm problem
- CDH = computational Diffie-Hellman problem
- DDH = decisional Diffie-Hellman problem
- $\oplus$ = bitwise XOR
- OFB = output feedback
- CBC = cipher block chaining