# MIT 6.875J/18.425J and Berkeley CS276
# Foundations of Cryptography (Fall 2020)

## Problem Set 3: Released September 29, Due October 20

The problem set is due on **Tuesday, October 20 at 10pm ET/7pm PT**. Please make sure to upload to the Gradescope course webpage by the deadline (all registered students should have access to this webpage on Gradescope). Be sure to mark on Gradescope where each problem's solution starts. Typed solutions using LaTeX are strongly encouraged (template provided on the course webpage).
Collaboration is permitted; however, you must write up your own solutions individually and acknowledge all of your collaborators.

**You only need to do 5/6 problems of your choice. If you do all 6, we will of course grade it and pick your 5 best.**

**Notation** Let $\mathcal{A}(x; r)$ denote the randomized algorithm $\mathcal{A}$ run on input $x$, using randomness $r$. We denote by $x||y$ the concatenation of the strings $x$ and $y$. We let $x \leftarrow \{0,1\}^n$ denote the process of sampling $x$ uniformly at random from $\{0,1\}^n$. We denote by $[n]$ the set of integers $\{1, \ldots, n\}$.

## Problem 1. RSA Security

Let $N = pq$ where $p, q$ are distinct large $\frac{n}{2}$-bit primes, and let $\phi(N) = (p-1)(q-1)$. Let $e$ be an integer with $1 < e < \phi(N)$ and $\mathsf{gcd}(e, \phi(N)) = 1$. Let $d = e^{-1} \pmod{\phi(N)}$.[1] The RSA permutation and the RSA inverse are defined for every $x \in \mathbb{Z}_N^*$ as:

$$\mathsf{RSA}_{N,e}(x) = x^e \pmod N, \qquad \mathsf{RSA}_{N,d}^{-1}(x) = x^d \pmod N.$$

Denote $pk = (N, e)$, $sk = (N, d)$ and $n = \lfloor \log_2(N) \rfloor$. Recall that the least significant bit ($\mathsf{lsb}$) is hardcore for the RSA function. That is, for every PPT algorithm $A$, $\Pr_{x \leftarrow \mathbb{Z}_N^*}[A(\mathsf{RSA}_{N,e}(x)) = \mathsf{lsb}(x)] = \frac{1}{2} + \mathsf{negl}(n)$.

**1.1** Show that the following encryption scheme is IND-secure:

$$\mathsf{Enc}(pk, m; r) = r^e \pmod N$$

where $m$ is a bit and $r \leftarrow \mathbb{Z}_N^*$ is uniformly random subject to the condition that $\mathsf{lsb}(r) = m$.

**1.2** Prove or refute that the following encryption scheme is IND-secure.

$$\mathsf{Enc}(pk, m; r) = r^e \pmod N$$

where $m$ has $n - \log(n)$ bits and $r \leftarrow \mathbb{Z}_N^*$ is uniformly random subject to the condition that $\mathsf{lsb}_{n-\log n}(r) = m$. (Here $\mathsf{lsb}_k(r)$ refers to the $k$ least significant bits of $r$.)

**1.3** Often in practice, people set the RSA exponent $e$ to be small to make the encryption algorithm faster. For example, they set $e = 3$ and $N$ is such that $\mathsf{gcd}(3, \phi(N)) = 1$. Assume $e = 3$ and prove or refute that the following encryption scheme is IND-secure.

$$\mathsf{Enc}(pk, m; r) = (m||r)^3 \pmod N$$

where $r \leftarrow \{0,1\}^\ell$, where $\ell < \frac{n}{3}$, and $m$ is a message of length $n - \ell$.

[1] One can compute $d$ using the Euclidian-gcd algorithm.

**1.4** Prove or refute that the following digital signature scheme is unforgeable:

$$\mathsf{Sign}(sk, m) = m^d \pmod N,$$

and $\mathsf{Verify}(pk, m, \sigma)$ returns 1 if $m = \sigma^e \pmod N$, otherwise returns 0.

## Problem 2. Public Key Encryption

Suppose that

- $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is an IND-secure public key encryption scheme for messages of arbitrary poly$(n)$ length;
- $\mathcal{F} = \left\{\{f_s : \{0,1\}^n \to \{0,1\}^\ell\}_s\right\}_n$ is a PRF family; and
- $G = \{G_n : \{0,1\}^n \to \{0,1\}^{2n}\}_n$ is a PRG.

Determine whether the following schemes are necessarily IND-secure public key encryption schemes (again, for arbitrary length messages). If so, specify the decryption algorithm and prove that the scheme is correct and secure. If not, construct (with proof) a counterexample.

**2.1** $\mathsf{Gen}'(1^n) = \mathsf{Gen}(1^n)$ and $\mathsf{Enc}'(pk, m; r) = \mathsf{Enc}(pk, m; G(r))$. We assume that $r$ has the appropriate length so that $\mathsf{Enc}'$ is well-defined.

**2.2** $\mathsf{Gen}'(1^n) = \mathsf{Gen}(1^n)$ and $\mathsf{Enc}'(pk, m; s) = \mathsf{Enc}(pk, m; f_s(m))$. We assume that $f_s(m) \in \{0,1\}^\ell$ has the appropriate length so that $\mathsf{Enc}'$ is well-defined.

**2.3** $\mathsf{Gen}'(1^n; \rho||s) = (pk||s, sk)$, where $\mathsf{Gen}(1^n; \rho) = (pk, sk)$, and $\mathsf{Enc}'(pk', m) = \mathsf{Enc}(pk, m; f_s(m))$, where $pk' = pk||s$. We assume that $f_s(m) \in \{0,1\}^\ell$ has the appropriate length so that $\mathsf{Enc}'$ is well-defined.

**2.4** $\mathsf{Gen}'(1^n; \rho||s) = (pk||s, sk||s)$, where $\mathsf{Gen}(1^n; \rho) = (pk, sk)$, and $\mathsf{Enc}'(pk', m) = \mathsf{Enc}(pk, f_s(m))$, where $pk' = pk||s$.

## Problem 3. Simultaneous Encryption and Authentication

Let $(\mathsf{Enc}, \mathsf{Dec})$ be a computationally indistinguishable symmetric encryption scheme and $\mathsf{MAC}$ an existentially unforgeable message authentication scheme under a chosen message attack: i.e., any adversary who can request MACs on messages $M_1, M_2, ...$ of his choice cannot generate a valid forgery on a new message $M'$.

Suppose Alice and Bob share two independent keys $K_1, K_2$ for privacy and authentication, respectively. They want to exchange messages $M$ in a private and authenticated way. Consider sending each of the following as a means to this end, and argue whether it is secure or not, where secure means both computationally indistinguishable and existentially unforgeable under CMA.

1. $M, \mathsf{MAC}_{K_2}(\mathsf{Enc}_{K_1}(M))$
2. $\mathsf{Enc}_{K_1}(M, \mathsf{MAC}_{K_2}(M))$
3. $\mathsf{Enc}_{K_1}(M), \mathsf{MAC}_{K_2}(M)$
4. $\mathsf{Enc}_{K_1}(M), \mathsf{MAC}_{K_2}(\mathsf{Enc}_{K_1}(M))$
5. $\mathsf{Enc}_{K_1}(M, A)$, where $A$ denotes the identity of Alice. Bob decrypts the ciphertext and checks that the second half of the plaintext is $A$.

## Problem 4. Circular Security

In this problem, we consider a security property (of encryption schemes) which is not explicitly guaranteed by IND-security; namely, security even when an adversary is given an encryption of the secret key $sk$. This kind of security is called *circular security.* One variant of circular security for encryption schemes is defined as follows.

**Definition 1.** *A secret key encryption scheme* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is said to be circular secure if for all ppt oracle algorithms* $\mathcal{A}^{\mathcal{O}(\cdot)}$*, we have that*

$$\mathop{\mathbf{E}}_{sk \leftarrow \mathsf{Gen}(1^n), r}\left[\mathcal{A}^{\mathrm{Left}_{sk}(\cdot)}(c^*)\right] = \mathop{\mathbf{E}}_{sk \leftarrow \mathsf{Gen}(1^n), r}\left[\mathcal{A}^{\mathrm{Right}_{sk}(\cdot)}(c^*)\right] + \mathrm{negl}(n)$$

*where* $c^* \leftarrow \mathsf{Enc}(sk, sk)$*,* $\mathrm{Left}_{sk}(\cdot)$ *is an oracle which on input* $(m_L, m_R)$ *outputs an encryption of* $m_L$*, and* $\mathrm{Right}_{sk}(\cdot)$ *is an oracle which on input* $(m_L, m_R)$ *outputs an encryption of* $m_R$*.*

**Definition 2.** *A public key encryption scheme* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is said to be circular secure if any ppt algorithm* $\mathcal{A}$ *wins the following game (interacting with a challenger) with probability at most* $\frac{1}{2} + \mathrm{negl}(n)$*:*

1. *The Challenger samples* $(pk, sk) \leftarrow \mathsf{Gen}(1^n)$ *and sends* $(pk, c^*)$ *to* $\mathcal{A}$*, where* $c^* \leftarrow \mathsf{Enc}(pk, sk)$ *is a ciphertext corresponding to the secret key.*
2. $\mathcal{A}$ *sends two messages* $(m_0, m_1)$ *to the Challenger.*
3. *The Challenger selects* $b^* \xleftarrow{\$} \{0,1\}$ *and sends* $c_{b^*} \leftarrow \mathsf{Enc}(pk, m_{b^*})$ *to* $\mathcal{A}$*.*
4. $\mathcal{A}$ *outputs a bit* $b$*. We say that* $\mathcal{A}$ *wins if* $b^* = b$*.*

**4.1 Circular Secure Public Key Encryption** Is every IND-secure public key encryption scheme also circular secure? Either prove that this is the case, or construct a counterexample (i.e. an IND-secure PKE scheme which is not circular secure).

**4.2 Secret-Key Regev* Encryption is Circular Secure** In this part, you will show that a variant of the LWE-based secret key encryption we saw in class *does* satisfy circular security (under an LWE-like assumption). In particular, we consider a variant of the LWE problem where the secret $s$ is a uniformly random *binary string.*

**Definition 3** (LWE* Assumption)**.** *The LWE* assumption with error distribution* $\chi$ *states that the following two distributions are computationally indistinguishable:*

$$\left\{\mathbf{s} \leftarrow \{0,1\}^n, \mathbf{A} \leftarrow \mathbb{Z}_q^{n\times m}, \mathbf{e} \leftarrow \chi^m : (\mathbf{A}, \mathbf{s}^\top \mathbf{A} + \mathbf{e}^\top)\right\} \approx_c \left\{\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times m}, \mathbf{b} \leftarrow \mathbb{Z}_q^m : (\mathbf{A}, \mathbf{b})\right\}.$$

Under the LWE* Assumption (with $m$ any polynomial in $n$, and with error distribution $\chi$ bounded by some $B \ll q$ with overwhelming probability), prove that the $n$-bit encryption scheme defined by

$$\mathsf{Enc}(\mathbf{s}, \mathbf{m} \in \{0,1\}^n, \mathbf{R} \leftarrow \mathbb{Z}_q^{n\times n}, \mathbf{e} \leftarrow \chi^n) = \mathbf{R} || \left(\mathbf{s}^\top \mathbf{R} + \mathbf{e}^\top + \left\lfloor \frac{q}{2} \right\rfloor \mathbf{m}^\top\right)$$

is a circular secure secret key encryption scheme.

**Hint:** Show that it is possible to generate an encryption of $s$ given only an encryption of 0.

## Problem 5. Proof of Solvency in Bitcoin

Solvency is the ability of an organization to pay its commitments to individuals (or other organizations). Organizations holding the Bitcoin have to prove their solvency at various times. In this problem, consider that organization `BankOfBitcoin` has to prove that the total of its Bitcoin assets is equal or higher than the sum of the Bitcoin amounts it owes its customers. Answer the following questions based on the (simplified) Bitcoin protocol taught in class. State any assumptions you are making.

**5.1** BankOfBitcoin tells an auditor that BankOfBitcoin has a total of $T$ Bitcoins. How can the auditor verify that BankOfBitcoin indeed owns $T$? Prove why BankOfBitcoin cannot cheat here.

**5.2** Consider that all customers with assets in BankOfBitcoin are willing to collaborate with the auditor to check BankOfBitcoin's solvency, but some other customers can be malicious and claim they also have assets in BankOfBitcoin when they don't. Describe a protocol by which BankOfBitcoin can prove to an auditor that it is solvent. The protocol should include protection against a malicious customer who claims BankOfBitcoin owns them some amount of money when it does not. Describe the protocol step by step and prove why BankOfBitcoin cannot cheat in this protocol.

**5.3** Assume the customers know $T$ from a trusted entity. Design a protocol where there is no auditor and the customers are checking BankOfBitcoin in a distributed way with no communication with each other. In the protocol each customer who have assets in BankOfBitcoin will perform a verification, and the verification time for each customer should be sublinear in the number of customers. Prove that BankOfBitcoin cannot cheat in this protocol.

## Problem 6. Encryption implies OWFs

In this problem you will show that one-way functions are necessary for encryption. You may assume that any probabilistic polynomial time algorithm in the encryption scheme uses exactly $n$ random bits, where $n$ is the security parameter.

**6.1** Show that public-key encryption implies the existence of one-way functions.
**Hint:** Think about the key generation algorithm

**6.2** Show that secret-key encryption implies the existence of one-way functions.
**Hint:** Think about the encryption algorithm. Also make sure that your argument does not construct a one-way function from a one-time pad.