ELE2742 / ENG1467 - PUC-Rio, 2019.1

Lecture 3: RSA Signature & Rabin Signature

DATE

Lecturer: Yi-Fan Tseng *Scribe: Yi-Fan Tseng*

In this lecture, we will first introduce the definition of a signature scheme, then introduce the most-widely used signature scheme, i.e., RSA signature [2], and an variant, i.e., Rabin signature [1].

1 Definition

A signature scheme (usually) consists of the following three algorithms.

$\mathsf{KeyGen}(1^\lambda)$. Taking as input the security parameter, the algorithm outputs a pair of public/private key $(\mathsf{PK}, \mathsf{SK})$.

$\mathsf{Sign}(\mathsf{SK}, \mathsf{M})$. Taking as inputs the private key SK and a message $\boldsymbol{M}$, the algorithm outputs a signature σ on the message M.

$\mathsf{Verify}(\mathsf{PK}, \mathsf{M}, \sigma)$. Taking as inputs the public key PK, a message M, and a signature σ, the algorithm outputs 1 if σ is a valid signature on M, and outputs 0 otherwise.

Correctness. For all message M and $(\mathsf{PK}, \mathsf{SK}) \leftarrow \mathsf{KeyGen}(1^\lambda)$,

$$\mathsf{Verify}(\mathsf{PK}, \mathsf{M}, \mathsf{Sign}(\mathsf{SK}, \mathsf{M})) = 1.$$

Note that, though different signature scheme may supports different message space $\mathcal{M}$, we can always choose a proper cryptographic hash function $H : \{0,1\}^* \to \mathcal{M}$ to map any messages into a valid message. Besides, some signature schemes are insecure if the Sign algorithm does not sign on hash messages.

2 RSA Signature

RSA signature [2] is proposed by Rivest, Shamir, and Adleman in 1978, which may be the most widely used signature scheme. The construction of RSA signature is very similar to RSA encryption, due to the symmetry between the public key and private key. The details of the algorithms are shown as follows.

$\mathsf{KeyGen}(1^\lambda)$: Taking as input the security parameter 1^λ, the algorithm performs as follows.

1. Choose two large primes p, q.

2. Compute $N = p \cdot q$.

3. Choose e such that $\gcd(e, \phi(N)) = 1$.

4. Compute $d = e^{-1} \mod \phi(N)$.

5. Output $\mathsf{PK} = (N, e)$ as the public key, $\mathsf{SK} = d$ as the private key.

$\mathsf{Sign}(\mathsf{SK}, \mathsf{M})$: Taking as inputs the public key $\mathsf{SK} = d$ and a message $\mathsf{M} \in \mathbb{Z}_N^*$, the algorithm outputs the signature

$$\sigma = \mathsf{M}^d \mod N.$$

$\mathsf{Verify}(\mathsf{PK}, \mathsf{M}, \sigma)$: Taking as inputs the public key $\mathsf{PK} = (N, e)$, the message M, and the signature σ, the algorithm outputs 1 if

$$\mathsf{M} = (\sigma)^e \mod N,$$

and outputs 0 otherwise.

Correctness. Note that

$$ed = 1 \mod \phi(N),$$

and an element in $\mathbb{Z}_N^*$ has order $\phi(N)$. Thus we have that

$$(\sigma)^e \pmod N = \mathsf{M}^{ed} \pmod N = \mathsf{M}^{ed \pmod{\phi(N)}} \pmod N = \mathsf{M} \pmod N.$$

We then discuss on the assumption $\mathsf{M} \in \mathbb{Z}_N^*$. If M is not coprime to N, then we have $\gcd(\mathsf{M}, N) = p$ or $\gcd(\mathsf{M}, N) = q$. The probability

$$\mathbf{Pr}[\gcd(\mathsf{M}, N) \neq 1; \mathsf{M} \xleftarrow{\$} [0, N]] = \frac{p+q}{N} = \frac{p+q}{pq}.$$

If $|p| \approx |q|$ and $|N| = 1024$ bits, then the probability

$$\frac{p+q}{N} \approx \frac{2}{\sqrt{N}} \approx \frac{1}{2^{511}},$$

which can be viewed as a negligible term.

However, this textbook version is not secure in the sense of unforgeability. Intuitively speaking, the unforgability states that, in a signature scheme, no one is able to generate a valid signature/message pair (σ, M) without the corresponding private key. Note that, any one is able to random value $\sigma \xleftarrow{\$} \mathbb{Z}_N$, and set $\mathsf{M} = \sigma^e \mod N$. Obviously, such (σ, M) is a valid signature/message pair. Though the message may be a seemly random string with overwhelming probability, in practice, the case may happen frequently. For example, in the concept of digital envelope, we encrypt a random short string as a symmetric key with the receiver's public key for the encryption of the later communications. To achieve the source verification, we will also send a signature on the short random key. A way to solve this problem is to apply a cryptgraphic hash function to the message first, then sign on the hash value. We show the modified version below.

$\mathsf{KeyGen}(1^\lambda)$: Taking as input the security parameter 1^λ, the algorithm performs as follows.

1. Choose two large primes p, q.
2. Compute $N = p \cdot q$.
3. Choose e such that $\gcd(e, \phi(N)) = 1$.
4. Compute $d = e^{-1} \mod \phi(N)$.
5. Choose a cryptographic hash function $H : \{0,1\}^* \to \mathbb{Z}_N$.
6. Output $\mathsf{PK} = (N, e, H)$ as the public key, $\mathsf{SK} = d$ as the private key.

$\mathsf{Sign}(\mathsf{SK}, \mathsf{M})$: Taking as inputs the public key $\mathsf{SK} = d$ and a message $\mathsf{M} \in \mathbb{Z}_N^*$, the algorithm outputs the signature

$$\sigma = H(\mathsf{M})^d \mod N.$$

$\mathsf{Verify}(\mathsf{PK}, \mathsf{M}, \sigma)$: Taking as inputs the public key $\mathsf{PK} = (N, e, H)$, the message M, and the signature σ, the algorithm outputs 1 if

$$H(\mathsf{M}) = (\sigma)^e \mod N,$$

and outputs 0 otherwise.

The correctness is easily verified. We then discuss the security against the aforementioned attack. If an attacker wants to proceed the attack, then it will first randomly choose $\sigma \xleftarrow{\$} \mathbb{Z}_N$. However, to find a message M such that $H(\mathsf{M}) = \sigma^e \pmod N$ is equivalent to violate the one-wayness of the hash function H.

2.1 Security Analysis

The security of RSA signature is believed to be based on the hardness of the factorization problem. It is easy to see that, having the factors p, q of N, an attacker can compute $\phi(N)$ and recover the private key d from e. One may argue that there might be some way to compute $\phi(N)$ without factorizing N. However, computing $phi(N)$ is equivalent to factorizing N. To see why, observe that $\phi(N) = (p-1)(q-1)$, we have the following equation system.

$$\begin{cases} N &= p \cdot q \\ \phi(N) &= (p-1)(q-1) = (N+1) - (p+q) \end{cases} \tag{1}$$

The solution to equation system 1 is the factors p and q.

3 Rabin Signature

Rabin signature [1] can be viewed as a variant of RSA signature. Intuitively, Rabin signature is the special case when we set $e = 2$ in RSA signature. However, if we set $e = 2$, then it is impossible to compute $d = e^{-1} \pmod{N}$. Therefore, to generate a signature, a signer computes the square root of the hash value of the message. The feature of Rabin signature is verifier-efficiency, only a modular multiplication is needed for the verification.

$\mathsf{KeyGen}(1^\lambda)$: Taking as input the security parameter 1^λ, the algorithm performs as follows.

1. Choose two large primes p, q.
2. Compute $N = p \cdot q$.
3. Choose a cryptographic hash function $H : \{0,1\}^* \to \mathbb{Z}_N$.
4. Output $\mathsf{PK} = (N, H)$ as the public key, $\mathsf{SK} = (p, q)$ as the private key.

$\mathsf{Sign}(\mathsf{SK}, \mathsf{M})$: Taking as inputs the public key $\mathsf{SK} = (p, q)$ and a message $\mathsf{M} \in \mathbb{Z}_N^*$, the algorithm outputs the signature

$$\sigma = H(\mathsf{M})^{\frac{1}{2}} \mod N.$$

It can be easily computed using the Chinese remainder theorem and the knowledge of p, q.

$\mathsf{Verify}(\mathsf{PK}, \mathsf{M}, \sigma)$: Taking as inputs the public key $\mathsf{PK} = (N, H)$, the message M, and the signature σ, the algorithm outputs 1 if

$$H(\mathsf{M}) = \sigma^2 \mod N,$$

and outputs 0 otherwise.

3.1 Security Analysis

The security of Rabin signature is based on the hardness of computing square roots without p, q, whose equivalence to the factorization problem has been shown in Chapter 2.

References

[1] M. O. Rabin. Digitalized signatures and public-key functions as intractable as factorization. Technical report, USA, 1979.

[2] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120126, Feb. 1978.