Computer Science 418

Quadratic Residuosity, Goldwasser-Micali, IND-CCA2 Security

Mike Jacobson

Department of Computer Science
University of Calgary

Week 11

Outline

Quadratic Residuosity

Definition 1 (Quadratic residues and non-residues)

Let $m \in \mathbb{N}$ and $a \in \mathbb{Z}_m^*$. Then a is said to be a *quadratic residue* modulo m if there exists some x such that $x^2 \equiv a \pmod{m}$. a is a *quadratic non-residue* modulo m otherwise.

Notation:

- QR_m: set of quadratic residues modulo m.
- QN_m: set of quadratic non-residues modulo m.

Note 1

$\mathbb{Z}_m^* = QR_m \cup QN_m$.

Prime and Composite Moduli

Suppose $m = p$, a prime. Then $\mathbb{Z}_p^* = QR_p \cup QN_p$ and $|QR_p| = |QN_p| = (p-1)/2$.

Example 2

If $p = 7$ we have $1^2 \equiv 1 \pmod{7}$, $2^2 \equiv 4 \pmod{7}$, $3^2 \equiv 2 \pmod{7}$, $4^2 \equiv 2 \pmod{7}$, $5^2 \equiv 4 \pmod{7}$, and $6^2 \equiv 1 \pmod{7}$. Thus, $QR_7 = \{1, 2, 4\}$ and $QN_7 = \{3, 5, 6\}$.

Theorem 1

$a \in QR_n$ if and only of $a \in QR_p$ for all primes $p \mid n$.

Euler's Criterion

Recall Fermat's Theorem: $a^{p-1} \equiv 1 \pmod{p}$ for p prime and $a \in \mathbb{Z}_p^*$.

Taking square roots (assume p odd) yields $a^{(p-1)/2} \equiv \pm 1 \pmod{p}$.

Theorem 2 (Euler's Criterion)

$a \in QR_p$ if and only if $a^{\frac{p-1}{2}} \equiv 1 \pmod{p}$.

Then $a \in QN_p$ if and only if $a^{\frac{p-1}{2}} \equiv -1 \pmod{p}$.

The Legendre Symbol

Definition 3 (Legendre symbol)

Let p be an odd prime. The *Legendre symbol* $\left(\frac{a}{p}\right)$ is defined as:

$$\left(\frac{a}{p}\right) = \begin{cases} 0 & \text{if } p \mid a \\ 1 & \text{if } a \in QR_p \\ -1 & \text{if } a \in QN_p \end{cases}$$

We can compute Legendre symbols — and by Euler's criterion test whether or not $a \in QR_p$ — in polynomial time using binary exponentiation.

Revised Theorem

Example 4

$\left(\frac{2}{7}\right) = 1$ and $\left(\frac{3}{7}\right) = -1$.

Remark 2 (Reformulation of Theorem 1)

$a \in QR_n$ if and only if $\left(\frac{a}{p}\right) = 1$ for all primes $p \mid n$.

Note 3 (Euler's Criterion revisited)

$a^{\frac{p-1}{2}} \equiv \left(\frac{a}{p}\right) \pmod{p}$ for all $a \in \mathbb{Z}$.

The Jacobi Symbol

Definition 5 (Jacobi symbol)

Let $Q \in \mathbb{N}$ be odd with prime factorization $Q = \prod_{i=1}^{r} q_i^{e_i}$, and let $P \in \mathbb{Z}$.

The *Jacobi symbol* $\left(\frac{P}{Q}\right)$ is defined as

$$\left(\frac{P}{Q}\right) = \prod_{i=1}^{r} \left(\frac{P}{q_i}\right)^{e_i}$$

where $\left(\frac{P}{q_i}\right)$ is the Legendre symbol.

Note 4

If Q is prime, then the Jacobi symbol $\left(\frac{P}{Q}\right)$ and the Legendre symbol $\left(\frac{P}{Q}\right)$ are the same.

Note 5

Properties of the Jacobi Symbol

$$\left(\frac{P}{Q}\right) = \left(\frac{P \pmod{Q}}{Q}\right)$$
$$\left(\frac{P_1 P_2}{Q}\right) = \left(\frac{P_1}{Q}\right)\left(\frac{P_2}{Q}\right)$$
$$\left(\frac{P}{Q_1 Q_2}\right) = \left(\frac{P}{Q_1}\right)\left(\frac{P}{Q_2}\right)$$
$$\left(\frac{2}{Q}\right) = (-1)^{\frac{Q^2-1}{8}}$$
$$\left(\frac{P}{Q}\right) = \left(\frac{Q}{P}\right)^{\frac{P-1}{2}\frac{Q-1}{2}} \qquad \text{(quadratic reciprocity)}$$

Properties 1, 4, and 5 allow one to compute $\left(\frac{P}{Q}\right)$ in polynomial time *without* factoring Q.

By Remark 2, we can have $\left(\frac{a}{n}\right) = 1$, but $a \notin QR_n$.

Pseudosquares

Definition 6 (Pseudosquare)

Let $n = pq$ with distinct primes p, q. A *pseudosquare* $\pmod{n}$ is an integer $a \in \mathbb{Z}$ with $\left(\frac{a}{p}\right)\left(\frac{a}{q}\right) = 1$.

Note that $\left(\frac{a}{n}\right) = 1$ makes a "look like" a square $\pmod{n}$, but $a \notin QR_n$.

The Quadratic Residuosity Problem (QRP)

Definition 7 (Quadratic Residuosity Problem (QRP))

Given an odd composite integer n and any a with $\left(\frac{a}{n}\right) = 1$, determine whether $a \in QR_n$.

Note 6

By Theorem 1 or Remark 2, the IFP is at least as hard has the QRP. Equivalence is believed, but unproved.

Eg. Leakage in RSA

Another weakness of RSA is *leakage* of information: $C \equiv M^e \pmod{n}$ implies

$$\left(\frac{C}{n}\right) = \left(\frac{M}{n}\right)^e = \left(\frac{M}{n}\right) ,$$

since e is odd.

Thus, one bit of information about the message is leaked (namely the value of the Jacobi symbol $\left(\frac{M}{n}\right)$.

- Thus, basic RSA is *not* sematically secure.
- This would not happen if the ciphertext in RSA were randomized.

Eg. ElGamal is not Semantically Secure

An attacker can chose $M_1 \in QR_p$ and $M_2 \in QN_p$ and distinguish between their encryptions in polynomial time.

- uses properties of quadratic residues and the Legendre symbol

Solution: replace g by $h \equiv g^2 \pmod{p}$ everywhere

- every quantity occurring in ElGamal is a quadratic residue modulo p.
- can prove that this variation of ElGamal *is* semantically secure.

The Goldwasser-Micali PKC

Achieves semantic security assuming the intractability of the QRP.

- Private key: $\{p, q\}$ where p and q are distinct large primes.
- Public key: $\{n, y\}$ where $n = pq$ and y is a pseudo-square modulo n.

Note 7

How to find y:

- Generate random integers $y \in \mathbb{Z}_n^*$ until a pseudosquare is found.
- Since there are four combinations $(\pm 1, \pm 1)$ for $\left(\left(\frac{y}{p}\right), \left(\frac{y}{q}\right)\right)$, one in four choices of y yields $(-1, -1)$.
- Hence, we expect to find a pseudosquare $\pmod{n}$ after four trials at a value of y.

Encryption

To encrypt a message M intended for a user with the above public/private key pair, proceed as follows:

1. Represent M as a bit-string $(m_1 m_2 \ldots m_t)$ $(m_i = 0, 1)$.
2. For $i = 1, \ldots, t$:
   1. Select random $r_i \in \mathbb{Z}_n^*$.
   2. Put $c_i \equiv y^{m_i} r_i^2 \pmod{n}$ with $0 < c_i < n$ (so $c_i \equiv r_i^2 \pmod{n}$ if $m_i = 0$ and $c_i \equiv y r_i^2 \pmod{n}$ if $m_i = 1$).
3. Send $C = (c_1 c_2 \ldots c_t)$.

Decryption

To decrypt, the recipient proceeds as follows:

1. for $i = 1, \ldots, t$:
   1. Compute the Legendre symbol $e_i = \left(\frac{c_i}{p}\right)$.
   2. $m_i = (1 - e_i)/2$ (so $m_i - 0$ if $e_i = 1$ and $m_i = 1$ if $e_i = -1$).
2. $M = (m_1 m_2 \ldots m_t)$.

Correctness of Decryption

Proof that decryption is correct.

Let $i \in \{1, \ldots, t\}$. Note that $\left(\frac{r_i^2}{p}\right) = \left(\frac{r_i}{p}\right)^2 = (\pm 1)^2 = 1$. Thus,

$$e_i = \left(\frac{c_i}{p}\right) = \left(\frac{y^{m_1} r_i^2}{p}\right) = \left(\frac{y^{m_1}}{p}\right)\left(\frac{r_i^2}{p}\right) = \left(\frac{y}{p}\right)^{m_i} = (-1)^{m_i}$$

Thus, if $e_i = 1$ then $m_i = 0$ and if $e_i = -1$ then $m_i = 1$. □

Polynomial Security of Goldwasser-Micali

Proof sketch of polynomial security.

Since r_i is selected at random:

- r_i^2 is a random quadratic residue modulo n
- thus, yr_i^2 is a random pseudosquare modulo n.

The cryptanalyst only sees a sequence of r_i^2 or yr_i^2 (quadratic residues and pseudosquares), and as the QRP is hard, he cannot distinguish one from the other. □

Major disadvantage:

- huge message expansion, by a factor of $\log_t(n)$
- A t-bit message yields a ciphertext of length $\approx t \log_2(n)$.

Active Attacks

Semantic and polynomial security provide a good notion of security against passive attacks. However, many (deterministic and randomized) PKCs are not secure against active attacks (CCA's).

Take the example of RSA. Note that RSA is *multiplicative*:

$$(M_1 M_2)^e \equiv M_1^e M_2^e \equiv C_1 C_2 \pmod{n}$$

i.e., a factorization of the plaintext implies one of the corresponding ciphertext. This property can be exploited in two attacks.

Multiplicative CCA on RSA

An attacker wishing the decryption of some RSA ciphertext C proceeds as follows:

1. Generates a random $X \in \mathbb{Z}_n^*$ with $X^e \not\equiv 1 \pmod{n}$.
2. Computes $C' \equiv CX^e \pmod{n}$ (this is the chosen ciphertext; note that $C' \neq C$).
3. Obtains the corresponding plaintext

$$M' \equiv C'^d \equiv C^d (X^e)^d \equiv MX \pmod{n}$$

4. Computes $M \equiv MX^{-1} \pmod{n}$.

Meet-in-the-Middle Attack on RSA

If $M \approx 2^l$ for some l, then with non-negligible probability, M is composite and satisfies $M = M_1 M_2$ with $M_1, M_2 \approx 2^{l/2}$.

- The probability that a number of 40 − 64 bits factors into equal-size factors is between 18 and 50 percent (see Table 1 of "Why textbook El Gamal and RSA encryption are insecure (extended abstract)" by Boneh, Joux, and Nguyen, in ASIACRYPT 2000)).

The adversary builds a list $\{1^e, 2^e \pmod{n}, \ldots, (2^{l/2})^e \pmod{n}\}$ and their inverses $\pmod{n}$.

- He then searches for a match $Ci^{-e} \pmod{n}$ in the list (i^{-e} is the modular inverse of i^e).
- If $Ci^{-e} \equiv j^e \pmod{n}$ for some j, then $M \equiv ij \pmod{n}$.

Requires $2 \cdot 2^{l/2}$ modular exponentiations (rest is negligible).

Example Application of Meet-in-the-Middle

Hybrid encryption: consider the case where 1024-bit RSA modulus is used to encrypt a 56-bit DES key.

- The list takes $2^{28} \cdot 1024 = 2^{38}$ bits of storage (about 32 GB)
- Requires 2^{29} modular exponentiations.
- This is easily done on a good PC.

Protecting against the Multiplicative Property

The multiplicative property of RSA can be obscured by randomizing the plaintext input in a fixed way, thus overcoming these problems.

Can defeat CCA by rejecting decryptions of "invalid" messages.

One example is RSA-OAEP (discussed below):

- RSA plus optimal asymmetric encryption padding
- plaintext is padded with 0's and transformed to a statistically random bit string via a reversible, randomized, unkeyed transformation.

IND-CCA2 Security

To address active attacks (CCA's), we need even stronger security notions than semantic security

Definition 8 (IND-CCA2 security)

A PKC is IND-CCA2 secure if it satisfies *indistinguishability under adaptive chosen ciphertext attacks*; in other words, no adversary can in expected polynomial time select two plaintext messages M_1 and M_2 and then correctly distinguish between encryptions of M_1 and M_2 with probability significantly greater than $1/2$, even when adaptive chosen ciphertext attacks are permitted.

IND-CCA2 Security, cont.

IND-CCA2 has the same definition as as polynomial security except that active attacks (in particular adaptive CCA's) are permitted.

- It is the active attack equivalent of semantic security.

Other security levels:

- IND-CCA1 — indistinguishability under (non-adaptive) chosen ciphertext attacks
- IND-CPA — indistinguishability under chosen plaintext attacks (same as polynomial security)

Note that IND-CCA2 $\Longrightarrow$ IND-CCA1 $\Longrightarrow$ IND-CPA.

Non-malleability

Definition 9 (Non-malleability)

A PKC is *non-malleable* if given a ciphertext C corresponding to some message M, it is computationally infeasible to generate a different ciphertext C' whose decryption M' is related to M in some known manner, i.e., $M' = f(M)$ for some arbitrary but known function f.

Non-malleability provides data integrity with public-key encryption *without* any source identification. We have

- NM-CPA $\Longrightarrow$ IND-CPA
- NM-CCA1 $\Longrightarrow$ IND-CCA1
- NM-CCA2 $\Longleftrightarrow$ IND-CCA2

It is known that IND-CPA $\not\Longrightarrow$ NM-CPA and IND-CCA1 $\not\Longrightarrow$ NM-CCA1.

Plaintext Awareness

Definition 10 (Plaintext awareness)

A PKC is *plaintext-aware* if it is computationally infeasible for an adversary to produce a "valid" ciphertext (having prescribed redundancy) without knowledge of the corresponding plaintext.

A plaintext-aware PKC resists adaptive attacks because any adaptive modification of a target ciphertext will with high probability not be "valid."

Plaintext awareness $\Longrightarrow$ Non-malleability.

Optimal Asymmetric Encryption Padding (OAEP)

Optimal Asymmetric Encryption Padding (OAEP):

- Bellare and Rogaway, Eurocrypt 1994
- An invertible transformation from a PKC plaintext space to the domain of a one-way trapdoor function.

OAEP augments PKCs to provide the above security properties by adding redundancy and transforming the plaintext before encryption. It works with most PKCs.

RSA-OAEP

Standardized in RSA's PKCS#1, IEEE P1363, e-commerce protocol SET (Secure Electronic Transaction)

Parameters

- n — length of plaintext messages to encrypt (in bits)
- (N, e) — Alice's RSA public key (N has $k = n + k_0 + k_1$ bits, where 2^{-k_0} and 2^{-k_1} must be sufficiently small). For example, if $k = 3072$, can take $k_0 = k_1 = 128$ and $n = 2816$.
- d — Alice's RSA private key
- $G : \{0,1\}^{k_0} \mapsto \{0,1\}^{k-k_0}$ (random function)
- $H : \{0,1\}^{k-k_0} \mapsto \{0,1\}^{k_0}$ (random function)

Encryption

Encryption (message M):

$$C \equiv \Big((M\|0^{k_1} \oplus G(r)) \parallel (r \oplus H(M\|0^{k_1} \oplus G(r))) \Big)^e \pmod N .$$

1. Generate a random k_0-bit number r.
2. Compute $s = (M\|0^{k_1}) \oplus G(r)$ (append k_1 0 bits to M for data integrity checking and XOR with $G(r)$). Note: s has $n + k_1 = k - k_0$ bits.
3. Compute $t = r \oplus H(s)$. Note: t has k_0 bits (same as N), but could be a bit bigger than N. If $(s\|t) \geq N$, go to 1 (make sure concatenation of s and t as an integer is less than the RSA modulus).
4. RSA-encrypt $(s\|t)$, i.e., compute $C \equiv (s\|t)^e \pmod N$.

Decryption

Decryption (ciphertext C):

1. Compute $(s\|t) \equiv C^d \pmod N$.
2. Compute $u = t \oplus H(s)$ (k_0 bit) and $v = s \oplus G(u)$ ($k - k_0$ bits).
3. Output M if $v = (M\|0^{k_1})$ (*i.e.* the decrypted message has the required redundancy), otherwise reject as invalid.

Security of RSA-OAEP

Can be proven to be plaintext-aware assuming that the RSA problem (computing eth roots modulo n) is hard:

- Defeats CCAs because only messages with the prescribed redundancy (0^{k_1} appended) are accepted. Probability of a random ciphertext decrypting to an acceptable value is 2^{-k_1}.
- Plaintext is also randomized — prevents small message space attacks (2^{k_0} possible encryptions of each message).

Random Oracle Model

RSA-OAEP's proof of security relies on the assumption that the functions G and H are random, i.e., mathematical functions mapping every possible query to a random response from its output domain.

Such functions are referred to as *random oracles*, and security proofs relying on this type of assumption are said to use the *random oracle model* (ROM).

In practice, G and H are realized with a hash function like SHA-1.

- In this case, the encryption scheme cannot be proven to be plaintext-aware.
- Nevertheless provides much greater security assurances than standard RSA

IND-CCA2 Security without Random Oracles

A variation of El Gamal due to Cramer and Shoup (CRYPTO 1998) is IND-CCA2 secure under the assumption that the decision Diffie-Hellman problem (given $g, g^a, g^b, g^c \in G$, does $g^c = g^{ab}$) is hard.

- The proof does *not* use the ROM.
- A recent result (Dent, EUROCRYPT 2006) shows that it is also plaintext aware, again without assuming random oracles.

Further Reading

Koblitz and Menezes, "Another look at provable security" (I and II), see links on "external links" page.

- discusses some issues with these types of security results, especially their relevance for practical cryptography.