6.875J/18.425J Cryptography and Cryptanalysis	September 19, 2001
Lecture 4	
Instructor: Anna Lysyanskaya	*Scribed by: Emil Sit*

1 Recap

1.1 Notation

We will recall the notation by using an example.

Example: Let p be a boolean function. $(\{x_i \leftarrow A_i(y_i) \mid 1 \leq i \leq n\} \ : \ p(x_n))$ denotess the event that $p(x_n)$ is TRUE after the value x_n was obtained by running algorithms $A_1, \ldots, A_n$ on inputs $y_1, \ldots, y_n$.

The statement

$$\Pr[\{x_i \leftarrow A_i(y_i) \mid 1 \leq i \leq n\} \ : \ p(x_n)] = \alpha$$

denotes the probability that $p(x_n)$ is TRUE after the value x_n was obtained by running algorithms $A_1, \ldots, A_n$ on inputs $y_1, \ldots, y_n$, is α.

Note that $x_i \leftarrow A_i(y_i)$ is a value if A_i is deterministic; otherwise x is a probability distribution.

1.2 The Definition of Security for Binary Message Spaces

A cryptosystem consists of three algorithms, G, E, D. $(PK, SK) \leftarrow G(1^k)$, $c \leftarrow E(PK, m)$, $m = D(SK, c)$. Such a cryptosystem is considered secure if:

$$\begin{aligned}
&\forall PPTF\{Eve_k\}\, \exists f(k) \in \text{negl}\, \exists K \text{ such that for } k > K:\\
&\quad \Pr[\ (PK, SK) \leftarrow G(1^k);\\
&\qquad b \leftarrow \{0,1\};\\
&\qquad c \leftarrow E(PK, b);\\
&\qquad b' \leftarrow Eve(PK, c) \quad : \quad b = b'] \leq 1/2 + f(k)
\end{aligned}$$

Where negl denotes the set of negligible functions,[1] and $PPTF$ stands for "probabilistic polynomial time family" of Turing machines.

1.3 The Quadratic Residuosity Assumption

Roughly speaking, the Quadratic Residuousity Assumption says that it is hard to tell a quadratic residue modulo n from a quadratic non-residue with Jacobi symbol of one modulo n, where n is an RSA modulus. More formally, let $PRIMES_k$ be the set of prime numbers of bit length k. Let $J_1(n)$ denote the set of elements of $\mathbb{Z}_n^* \bmod n$ with Jacobi Symbol 1. Let $QC(a, n)$ denote the quadratic character of a modulo n. That is to say, $QC(a, n) =$ "square" if a is a square and $QC(a, n) =$ "non-square" otherwise. Then, the quadratic residuosity assumption is as follows:

[1] Recall that $f(k) \in \text{negl}$ if for all $c \in \mathbb{N}$, there exists a K such that for all $k \geq K$, $f(k) < 1/k^c$.

$$
\begin{array}{l}
\forall PPTF\{A_k\}\,\exists f(k) \in \text{negl} \\
\quad \Pr[\, p \leftarrow PRIMES_k; \\
\quad\quad q \leftarrow PRIMES_k; \\
\quad\quad a \leftarrow J_1(pq); \\
\quad\quad b' \leftarrow A_k(a,n) \quad : \quad b' = QC(a,n)\,] \leq 1/2 + f(k)
\end{array}
$$

Let $QR(n)$ denote the set of quadratic residues modulo n. Let $QNR(n)$ denote the set of quadratic non-residues modulo n with Jacobi symbol 1. Note that $a \leftarrow (QR(n) \cup QNR(n))$ is equivalent to $a \leftarrow J_1(n)$.

Recall (Fact 13 of the lecture notes for Lecture 3) that exactly half of the elements of $J_1(n)$ are squares, so in fact $|QR(n)| = |QNR(n)|$. Because of this, an equivalent way of writing the selection of a is: $(d \leftarrow \{0,1\}; a_0 \leftarrow QR(n); a_1 \leftarrow QNR(n); a \leftarrow a_d)$. That is to say, we first choose d, the quadratic character of a, and then choose a in accordance with d.

Moreover, let us abbreviate $(p \leftarrow PRIMES_k, q \leftarrow PRIMES_k, n \leftarrow pq)$ into $n \leftarrow RSAM_k$, i.e. $RSAM_k$ is the algorithm for selecting an RSA modulus with primes of length k.

So let us rewrite the quadratic residuosity assumption as follows:

$$
\begin{array}{l}
\forall PPTF\{A_k\}\,\exists f(k) \in \text{negl} \\
\quad \Pr[\, n \leftarrow RSAM_k; \\
\quad\quad d \leftarrow \{0,1\}; \\
\quad\quad a_0 \leftarrow QR(n); \\
\quad\quad a_1 \leftarrow QNR(n); \\
\quad\quad d' \leftarrow A_k(a,n) \quad : \quad d' = QC(a,n)\,] \leq 1/2 + f(k)
\end{array}
$$

2 The Goldwasser-Micali Cryptosystem

The GM cryptosystem is as follows:

Key Generation G Alice chooses two random k-bit primes, p and q. Let $n = pq$. Alice also chooses a random $x \in QNR(n)$. I.e., x is a random non-square with Jacobi symbol 1. Alice's public key PK is the tuple (n, x), while her secret key SK is n's factorization (p, q).

Encryption E To encrypt a 0, Bob sends Alice a random square modulo n. To encrypt a 1, Bob sends Alice a random non-square with Jacobi symbol 1. More precisely, $c \leftarrow E(n, x, b)$ is computed as follows: $(r \leftarrow \mathbb{Z}_n^*; c \leftarrow r^2 x^b \bmod n)$.

Let's do this exercise: Why does this correspond to choosing a random square in the event that $b = 0$? Why does this correspond to choosing a random non-square with Jacobi symbol 1 in the event that $b = 1$?

Def: two distributions, X, Y on the domain D are identical if for all $d \in D$, $\Pr[d|X(D)] = \Pr[d|Y(D)]$.

To solve the exercise, use the number-theoretic fact that each $a \in QR(n)$ has exactly four square roots.

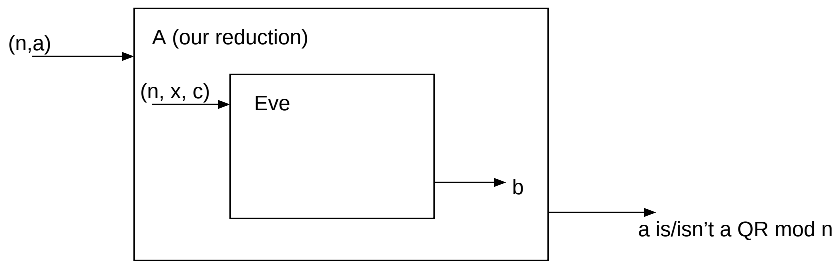
Let $a \in QR(n)$ be given. There are exactly $(p-1)(q-1)/4$ elements in $QR(n)$. Therefore, $\Pr[a|QR(n)] = 4/(p-1)(q-1)$. Now let us calculate the probability that a gets picked when we are encrypting a 0. a has four square roots. A given square root of a is picked with prob. $1/(p-1)(q-1)$. But there are four of them, so the prob that some square root is picked is $4/(p-1)(q-1)$. Therefore, the two distributions are identical.

Now what about encrypting a 1? We use a similar counting argument to show that the distributions are identical. Since $|QR(n)| = |QNR(n)|$, $\Pr[a|QNR(n)] = 4/(p-1)(q-1)$. Now we must find the probability that a is sent when encrypting a 1. Since x is fixed (as part of the public key), in order to receive a, we must choose r such that $r^2 = a/x$. What is the probability of this? Since there are four square roots of a/x, the probability is again $4/(p-1)(q-1)$. Again, the distributions are identical.

Decryption D To decrypt, Alice determines whether her received value c is a quadratic residue modulo n. More precisely, Alice tests whether $\left(\frac{c}{p}\right) = \left(\frac{c}{q}\right) = 1$. In Lecture 3, we saw an efficient algorithm for doing this.

3 Proof of Security for One-Bit Messages

We want to show that if there exists an Eve who can efficiently figure out the encrypted bit, then the quadratic residuosity assumption is false.[2]



So we need to construct an algorithm A that receives, as input, the values (n, a), where a has Jacobi symbol 1, and that uses Eve as a "black box." That is, it creates the inputs that Eve expects to receive, and interprets the output. In the event that Eve breaks the security of the encrypted bit, our algorithm A will break the QR assumption, i.e. it will, with probability non-negligibly better than 1/2, determine the quadratic character of a.[3]

[2] We could show instead that existence of Eve contradicts some other assumption. For example, a reduction that could solve circuit satisfiability, would do nicely. However, it is not known whether this is possible; actually showing a reduction from secure encryption to SAT is believed to be nearly as challenging as proving that $P \neq NP$. So instead, we focus on hardness assumptions that have nice properties.

[3] Our reduction must be very careful with the results returned by Eve. It may be that Eve operates in a way that is least favorable to us in any (and every situation). Thus, our reduction must be sure to be correct even in this case.

An example of a wrong approach: the reduction receives (n, a) and wants to know whether a is a quadratic residue. Creates $PK = (n, u)$, where u is just some random thing. Then it gives $(a, (n, u))$ to Eve. If Eve says "0" we say "square," otherwise we say "non-square." The problem is like this: Suppose Eve is sneaky

Assume, for contradiction, that there exists an adversary Eve such that there exists an inverse-polynomial $p(k)$ such that for infinitely many k,

$$\begin{aligned}
&\Pr[\,(PK,SK) \leftarrow G(1^k);\\
&\quad b \leftarrow \{0,1\};\\
&\quad c \leftarrow E(PK,b);\\
&\quad b' \leftarrow Eve(PK,c) : \quad b' = b] \geq 1/2 + p(k)
\end{aligned}$$

Consider the following algorithm: on input (n,a), set $PK = (n,a)$. Then, choose b,r at random and let $c = r^2a^b \bmod n$. Obtain $b' = \text{Eve}(PK,c)$. If $b' = b$, output "Non-square" (or 1) Otherwise, output "Square" (or 0).

Claim 1: if a is a non-square, then the probability that $b' = b$ in this experiment is the same as the probability that Eve guesses the encrypted bit correctly. More precisely,

$$\begin{aligned}
&\Pr[\,n \leftarrow RSAM_k;\\
&\quad a \leftarrow QR(n);\\
&\quad b \leftarrow \{0,1\};\\
&\quad r \leftarrow \mathbb{Z}_n^*;\\
&\quad c \leftarrow r^2a^b;\\
&\quad b' \leftarrow Eve((n,a),c) : \quad b' = b] \geq 1/2 + p(k)
\end{aligned}$$

Proof: The claim follows because in this case we are copying the GM cryptosystem set-up exactly. The first two steps comprise G, the next step is the selection of the message, and the next two steps are the encryption E. This is exactly what we have assumed that Eve can do. □

Claim 2: if a is a square, then the probability that $b' = b$ is exactly $1/2$. More precisely:

$$\begin{aligned}
&\forall PPTF\{Eve_k\}\\
&\quad \Pr[\,n \leftarrow RSAM_k;\\
&\qquad a \leftarrow QR(n);\\
&\qquad b \leftarrow \{0,1\};\\
&\qquad r \leftarrow \mathbb{Z}_n^*;\\
&\qquad c \leftarrow r^2a^b;\\
&\qquad b' \leftarrow Eve_k((n,a),c) : \quad b' = b] = 1/2
\end{aligned}$$

Proof (intuition): Suppose a is a quadratic residue. Then, no matter what b is, c is distributed in the same way, i.e. c is always a random quadratic residue. Thus b is distributed independently on (n,a,c). Therefore, we can view Eve's output b' as fixed before b is even decided. Therefore, the probability that b will come out equal to b' is exactly $1/2$.

Proof (formal): Suppose a is a quadratic residue and n as given. Then, $E((n,a),0)$ and $E((n,a),1)$ have distributions identical to that of a randomly selected quadratic residue. Consider a quadratic residue r. Observe first that $|QR| = \varphi(n)/4 = (p-1)(q-1)/4$. Thus,

and she can really tell QRs from QNRs. So here's what she does: if (n,u) is a valid PK (i.e., u is a QNR), then she gives the right answer. Otherwise, she sneakily does this: if $\left(\frac{u}{n}\right) = -1$ she always outputs 0. Otherwise, if u is a square, she gives the *wrong* answer. Then our reduction can only work if it can pick u correctly! But u was picked at random, so with prob $1/2$ it is good, and we are right, and with prob $1/2$ it is bad and we are wrong, so we are really not doing any better than $1/2$.

So we have to outsmart sneaky Eve so she has no clue.

for a quadratic residue chosen uniformly at random from the space of quadratic residues modulo n, probability of selecting r is $|QR|^{-1}$, or $4/\varphi(n)$.

Now, let us consider $\rho_1 = \Pr[r' \leftarrow E((n,a),1) : r' = r]$. This means we take a random x^2 and transform it by multiplication by a. Since a is a quadratic residue (by assumption), the product is also a quadratic residue. We show that this is a bijective map.

one-to-one We need a^{-1} to exist; but since this is a subset of $\mathbb{Z}_n^*$, it does.

onto We must show that given y^2, there is x^2 such that $y^2 = ax^2$. Let $\alpha^2 = a$. Then $(y/\alpha)^2 = x^2$.

Thus, ρ_1 is the same as the probability of selecting any quadratic residue, or $4/\varphi(n)$.

It is clear that
$rho_0 = Pr[r' \leftarrow E((n,a),0) : r' = r]$ is also $4/\varphi(n)$ since encrypting a zero involves exactly selecting a random number and squaring it.

Since these three are identically distributed then, we can rewrite the Claim 2 as:

$$
\begin{array}{l}
\forall PPTF\{Eve_k\} \\
\quad \Pr[\ n \leftarrow RSAM_k; \\
\qquad a \leftarrow QR(n); \\
\qquad c \leftarrow QR(n); \\
\qquad b' \leftarrow Eve_k((n,a),c); \\
\qquad b \leftarrow \{0,1\} : \quad b' = b] = 1/2
\end{array}
$$

That is, since encryption produces essentially a random quadratic residue (if a is a QR), Eve's output is essentially fixed: it does not matter what b is chosen. Since we are randomly selecting the plaintext, Eve's output will match ours half the time. $\square$

To complete the proof of the reduction, note that:

$$
\begin{aligned}
\Pr[A \text{ is correct}] &= \Pr[A \text{ is correct } | a \in QR(n)] \Pr[a \in QR(n)] + \\
&\quad \Pr[A \text{ is correct } | a \in QNR(n)] \Pr[a \in QNR(n)] \\
&= \frac{1}{2} \cdot \frac{1}{2} + \left(\frac{1}{2} + p(k)\right) \frac{1}{2} \\
&= \frac{1}{2} + \frac{p(k)}{2}
\end{aligned}
$$

This is a basic fact from probability where we condition on all the possible cases for a. And so if Eve's advantage over $1/2$, $p(k)$, is non-negligible, then our algorithm A will also have a non-negligible advantage $p(k)/2$.

More formally, the probability that A is correct can be written:

$$
\begin{array}{l}
\Pr[\ n \leftarrow RSAM_k; \\
\quad d \leftarrow \{0,1\}; \\
\quad a_0 \leftarrow QR(n); a_1 \leftarrow QNR(n); a \leftarrow a_d; \\
\quad d' \leftarrow A_k(a,n) : \quad d' = d]
\end{array}
$$

Conditioning on whether a is a QR, this can be rewritten as:

$$
\begin{aligned}
& \Pr[\, n \leftarrow RSAM_k; a \leftarrow QR(n); d' \leftarrow A_k(a,n) \quad : d' = 0] \times \Pr[d \leftarrow \{0,1\} : d = 0] \\
+ \; & \Pr[\, n \leftarrow RSAM_k; a \leftarrow QNR(n); d' \leftarrow A_k(a,n) : d' = 1] \times \Pr[d \leftarrow \{0,1\} : d = 1]
\end{aligned}
$$

But, since we know that A_k really makes use of Eve_k, this is exactly equal to:

$$
\begin{aligned}
& \Pr[\, n \leftarrow RSAM_k; a \leftarrow QR(n); b \leftarrow \{0,1\}; \\
& \quad r \leftarrow \mathbb{Z}_n^*; c \leftarrow r^2 a^b; b' = Eve_k((n,a),c) \quad : b' = b] \times \Pr[d \leftarrow \{0,1\} : d = 0] \\
+ \; & \Pr[\, n \leftarrow RSAM_k; a \leftarrow QNR(n); b \leftarrow \{0,1\}; \\
& \quad r \leftarrow \mathbb{Z}_n^*; c \leftarrow r^2 a^b; b' = Eve_k((n,a),c) \quad : b' = b] \times \Pr[d \leftarrow \{0,1\} : d = 1]
\end{aligned}
$$

Fortunately, the claims above allow us to substitute in values. Claim 2 allows us to substitute exactly $1/2 \times 1/2$ for the first addend and Claim 1 says that the second addend is greater than $(1/2 + p(k)) \times 1/2$. If we let ρ be the above probability, then:

$$
\begin{aligned}
\rho & \geq \left(\frac{1}{2}\right)\frac{1}{2} + \left(\frac{1}{2} + p(k)\right)\frac{1}{2} \\
& \geq \frac{1}{2} + \frac{p(k)}{2}
\end{aligned}
$$

3.1 A Special Case

Suppose $p \equiv 3 \bmod 4$ and $q \equiv 3 \bmod 4$. That is $p = 4m_1 + 3$ and $q = 4m_2 + 3$. Call QRA' the QRA limited to this case. Then:

$$
\begin{aligned}
\left(\frac{-1}{n}\right) & = \left(\frac{-1}{p}\right)\left(\frac{-1}{q}\right) \\
& = (-1)^{\frac{p-1}{2}}(-1)^{\frac{q-1}{2}} \\
& = (-1)^{2m_1+1}(-1)^{2m_2+1} \\
& = (-1)(-1) \\
& = 1
\end{aligned}
$$

Thus, $-1 \in QNR(n)$; that is, a quadratic non-residue with Jacobi symbol 1. Consider a revised G and E in the GM-cryptosystem as follows:

Revised G Let PK consist just of n.

Revised E If $b = 0$, output r^2, if $b = 1$, output $-r^2$.

As an exercise, verify that in this case the following simpler reduction works: on input (n, x), just run Eve on (n, x) and output what Eve outputs.