University of Maryland
CMSC456 — Introduction to Cryptography
Professor Jonathan Katz

Lecture 32

1 Indistinguishable Public-Key Encryption

Last time, we gave a definition of security in the sense of indistinguishability for public-key encryption schemes. This definition is exactly analogous to the definition we gave in the case of private-key encryption. In the case of private-key encryption, indistinguishability was strictly stronger than security against ciphertext-only attacks. This is not the case for public-key encryption, as we show here. Specifically:

Theorem 1 *Let $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a public-key encryption scheme which is (t, ϵ)-secure against ciphertext-only attacks. Then $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ is $(t, \ell\epsilon)$-secure in the sense of indistinguishability (where the adversary is assumed to access the* LR *oracle ℓ times).*

Thus, as long as ϵ is small, and ℓ is within reason (of course, we always must have $\ell \leq t$), the scheme is secure in the sense of indistinguishability. Typical values might be $\epsilon = 2^{-80}$ and $\ell \leq 2^{18}$ or so (even if t is much higher), implying that $\ell\epsilon$ is still sufficiently small.

Proof We prove the theorem for the case $\ell = 2$, and leave the general case to the reader. Note that even the case $\ell = 2$ is already a vast improvement over the private-key case, where the one-time pad (for example) was $(t, 0)$- secure against ciphertext-only attacks, but not $(t, 1-\epsilon)$-secure (for any $\epsilon > 0$) in the sense of indistinguishability, even for $\ell = 2$.

Let A be an adversary attacking the encryption scheme in the sense of indistinguishability, and making two queries to the LR oracle. Let (m_1, m_1') and (m_2, m_2') denote the pairs of messages that A submits to the LR oracle (i.e., (m_1, m_1') are the messages submitted the first time and (m_2, m_2') are the messages submitted the second time). Then we are interested in bounding the following:

$$\begin{aligned}
&\left|2 \cdot \Pr[(PK, SK) \leftarrow \mathcal{K}; b \leftarrow \{0,1\} : A^{\mathsf{LR}_{b,PK}(\cdot,\cdot)}(PK) = b] - 1\right| \\
&= \left|\Pr[A^{\mathsf{LR}_{0,PK}(\cdot,\cdot)}(PK) = 0] - \Pr[A^{\mathsf{LR}_{1,PK}(\cdot,\cdot)}(PK) = 0]\right| \\
&= \left|\Pr[A(PK, \mathcal{E}_{PK}(m_1), \mathcal{E}_{PK}(m_2)) = 0] - \Pr[A(PK, \mathcal{E}_{PK}(m_1'), \mathcal{E}_{PK}(m_2')) = 0]\right|,
\end{aligned}$$

where we have been slightly informal (in particular, (PK, SK) are randomly generated in each experiment, and $\mathcal{E}_{PK}(m)$ refers to a random encryption of message m).

Before giving the details of the proof, we provide a high-level overview. Note that the final expression above is equal to:

$$\begin{aligned}
&\big|\Pr[A(PK, \mathcal{E}_{PK}(m_1), \mathcal{E}_{PK}(m_2)) = 0] - \Pr[A(PK, \mathcal{E}_{PK}(m_1'), \mathcal{E}_{PK}(m_2)) = 0] \\
&\quad + \Pr[A(PK, \mathcal{E}_{PK}(m_1'), \mathcal{E}_{PK}(m_2)) = 0] - \Pr[A(PK, \mathcal{E}_{PK}(m_1'), \mathcal{E}_{PK}(m_2')) = 0]\big| \quad (1)\\
\leq\ &\left|\Pr[A(PK, \mathcal{E}_{PK}(m_1), \mathcal{E}_{PK}(m_2)) = 0] - \Pr[A(PK, \mathcal{E}_{PK}(m_1'), \mathcal{E}_{PK}(m_2)) = 0]\right| \quad (2)\\
&\quad + \left|\Pr[A(PK, \mathcal{E}_{PK}(m_1'), \mathcal{E}_{PK}(m_2)) = 0] - \Pr[A(PK, \mathcal{E}_{PK}(m_1'), \mathcal{E}_{PK}(m_2')) = 0]\right|. \quad (3)
\end{aligned}$$

Using the fact that the encryption scheme is secure against ciphertext-only attacks, we will bound Expressions (2) and (3).

We construct an adversary A' mounting a ciphertext-only attack against the encryption scheme. Here, A' is given a ciphertext C which is either an encryption of m_1 or of m'_1:

$A'(PK, C)$
compute $C_2 \leftarrow \mathcal{E}_{PK}(m_2)$ (note that A' can do this since it knows PK)
run $A(PK, C, C_2)$
output whatever is output by A

By definition of A':

$$\begin{aligned}
&\left|\Pr[A'(PK, \mathcal{E}_{PK}(m_1)) = 0] - \Pr[A'(PK, \mathcal{E}_{PK}(m'_1)) = 0]\right| \\
&= \left|\Pr[A(PK, \mathcal{E}_{PK}(m_1), \mathcal{E}_{PK}(m_2)) = 0] - \Pr[A(PK, \mathcal{E}_{PK}(m'_1), \mathcal{E}_{PK}(m_2)) = 0]\right| \\
&\leq \epsilon,
\end{aligned}$$

where the final inequality holds since the encryption scheme is (t, ϵ)-secure against ciphertext-only attacks.

We now construct adversary A'', also mounting a ciphertext-only attack against the encryption scheme. Here, A'' is given a ciphertext C which is either an encryption of m_2 or m'_2:

$A''(PK, C)$
compute $C_1 \leftarrow \mathcal{E}_{PK}(m'_1)$ (again, A'' can do this since it knows PK)
run $A(PK, C_1, C)$
output whatever is output by A

By definition of A'':

$$\begin{aligned}
&\left|\Pr[A'(PK, \mathcal{E}_{PK}(m_2)) = 0] - \Pr[A'(PK, \mathcal{E}_{PK}(m'_2)) = 0]\right| \\
&= \left|\Pr[A(PK, \mathcal{E}_{PK}(m'_1), \mathcal{E}_{PK}(m_2)) = 0] - \Pr[A(PK, \mathcal{E}_{PK}(m'_1), \mathcal{E}_{PK}(m'_2)) = 0]\right| \\
&\leq \epsilon,
\end{aligned}$$

where, again, the final inequality holds since the encryption scheme is (t, ϵ)-secure against ciphertext-only attacks.

Thus, both Expressions (2) and (3) are bounded by ϵ, implying that Expression (1) is bounded by 2ϵ and proving the theorem. ■

An important corollary of this theorem is that once we have a secure public-key encryption scheme for messages of length ℓ, we may immediately use the scheme to encrypt arbitrary-length messages by breaking messages to be encrypted into a sequence of ℓ-bit blocks (padding if necessary) and encrypting each block separately (using fresh randomness each time). Note that this is "equivalent" to sequential encryptions of ℓ-bit messages, and is therefore secure by the above theorem.

We note the crucial difference between the private-key case and the public-key case. In the proof above, adversaries A' and A'' can generate (random) encryptions of m_2 and m'_1, respectively, *because they are explicitly given the public key PK*. The is *not* the case for private-key encryption, where the adversary does *not* get to learn the key and therefore cannot generate encryptions of other messages.

1.1 The Value of Theorem 1

Theorem 1 is very useful for proving the security of public-key encryption schemes. Out ultimate goal will always be to construct an indistinguishable encryption scheme. Yet in analyzing (and proving security of) such a scheme, we need only prove security against ciphertext-only attacks — a much simpler task. Once we have done so, however, we may immediately apply Theorem 1 to show that the scheme is in fact secure in the sense of indistinguishability. This makes the design of provably-secure schemes easier.

2 Hybrid Encryption

We have now seen two secure public-key encryption schemes. Let us look at the efficiency of each.

- The scheme based on quadratic residuosity was originally defined only for encryption of 1-bit messages. But it should be clear (since, by Theorem 1, the scheme is secure in the sense of indistinguishability and hence secure when multiple messages are encrypted) that ℓ-bit messages can be encrypted by simply concatenating (random) encryptions of each of the individual bits. Note that each encryption of a single bit results in a k-bit ciphertext (where k is the length of the modulus N), meaning that encrypting an ℓ-bit message results in a $k\ell$-bit ciphertext. In terms of computational efficiency, encryption of each bit requires 1–2 modular multiplications each taking time $O(k^2)$ (this can be improved, but it is not relevant here).

- The El Gamal encryption scheme had improved communication efficiency. Namely, encrypting a k-bit message resulted in a ciphertext of length $2k$, for an expansion factor of only 2. Computationally, however, the scheme is not much of an improvement over the previous scheme. In particular, encrypting a k-bit message requires two exponentiations each taking time $O(k^3)$. Thus, the amount of computation *per bit* is roughly the same as in the previous scheme. (Note: In fact, this comparison is slightly inaccurate, since different key sizes k might be used for the different schemes. However, the thrust of the argument is clear.)

In absolute terms, if we compare the efficiencies of public- and private-key encryption we see that private-key encryption (say, using a block cipher) is roughly 1000 times faster than public-key encryption. Again, this is only a rough estimate, as it depends on which public- and private-key schemes are being compared. Yet it is fair to say that private-key encryption is roughly 3 orders of magnitude faster than public-key encryption.

Clearly, then, we want to avoid using "public-key cryptography" to transmit very long messages. But how can we do so while retaining the benefits of public-key encryption? Next time, we discuss *hybrid encryption* which is a method for obtaining the best of both worlds.