Abdalla, Bellare, and Nevens' "Robust Encryption"

Andrew Herrmann

March 11, 2012

In our course, we have been introduced to CPA and CCA security in various senses (semantic, indistinguishability, etc.), and under various encryptions schemes. The authors Abdalla, Bellare, and Neven define a different sort of security in their paper "Robust Encryption."

Encryption robustness arises in the following setting: suppose I am a sender routinely communicating with a group of people, $id_1, \ldots, id_n$. Often I may want to anonymously communicate with a single individual of the group; that is, not only do I want my message to remain private, but I also desire the identity of the sender to remain anonymous. I am not allowed a private channel to communicate with this one member without revealing his identity.

One solution to this scenario is to assign a separate key ek_i (or, in the asymmetric setting, key pair (ek_i, dk_i)) to each member id_i in the group. However, this may lead to miscommunication: if I encrypt a message using a particular member id_i's key ek_i, all other members id_j will attempt to decrypt this message using their respective keys dk_j, which may easily result in miscommunication. Hence, we desire an encryption scheme that achieves the following:

- Privacy of messages (in the CPA or CCA setting)
- Anonymity of message recipient
- Prevention of miscommunication.

The authors solve the last two problems by requiring that for any key ek_i, encryption scheme $Enc_{ek_i}(\cdot)$ and decryption scheme $Dec_{dk_j}(\cdot)$, $Dec_{dk_j}(Enc_{ek_i}(M)) = M$ if $j = i$ and $\perp$ otherwise; that is, any cipher text C should only be a valid cipher text for a particular encryption key ek_i. They call this idea "robustness."

As always, we need a notion of adversarial advantage. Suppose we have a general encryption scheme GE equipped with a parameter generator PG, a key generator KG, and encryption and decryption oracles Enc and Dec. We also define $\mathcal{M}$ to be the message space, The authors define two notions of security in this setting:

$$\mathrm{Adv}^{WROB}_{GE}(A) = P((msk, par) \xleftarrow{\$}; id_0, id_1 \leftarrow A; (ek_0, dk_0), (ek_1, dk_1) \xleftarrow{\$} KG(par, id_b); m \leftarrow A : Dec_{dk_1}(Enc_{ek_0}(m)) \in \mathcal{M}).$$

$$\mathrm{Adv}^{SROB}_{GE}(A) = P((msk, par) \xleftarrow{\$}; id_0, id_1 \leftarrow A; (ek_0, dk_0), (ek_1, dk_1) \xleftarrow{\$} KG(par, id_b); C \leftarrow A : Dec_{dk_0}(C), Dec_{dk_1}(C) \in \mathcal{M}).$$

The first thing to notice here is that weak or strong robustness does not imply CPA or CCA security in itself: nowhere in the definition of adversarial advantage is it explicitly stated that a message remains private, or that no bits of the message is leaked. In fact, one may imagine a modification of a robust encryption scheme that appends the entire message to the ciphertext. The this ciphertext is still only valid under a single encryption key in the altered encryption scheme (since the original text was), but clearly we violate privacy in the worst way.

When I first read the requirement, adding robustness to an existing secure scheme seemed to be a gargantuan feat. Fortunately, the authors did not find this so, and have methods to add strong or weak robustness to a general encryption scheme (possibly public key) while maintaining privacy.

The first sense of robust encryption, weak "robustness," can be satisfied by adding carefully adding redundancy to the encryption oracle. By carefully here, we encrypt $m||r$, where r is a redundancy tag that depends on some random key K. To add weak robustness to any general encryption scheme GE, the authors recommend letting $r = K$. While I would have liked to see some other methods here (perhaps there are times when appending K to the message is unwise), the robustness

addition is a simple one. Further, if the original scheme is CCA or CPA secure, adding the redundancy tag will not yield additional advantage to the adversary. The authors omit this, but it is easy to see: if our original encryption of m was secure, then the encryption of $m||r$ will be as well.

A more curious point in the paper is that while weak robustness can be added with no loss of CPA/CCA security, strong robustness cannot. In fact, the authors call upon the notions of hiding security, as well as the addition of weak-robustness to the initial scheme GE to achieve strong-robustness. The scheme they deliver then has CPA/CCA security lower than the original general encryption. They give the adversary at least three times the advantage given by the original scheme; whether this is consequential is left unexamined, though I assume it is not. Their addition of strong-robustness also first requires the addition of weak-robustness; though left unexplored, I assume there is no substantial increase in runtime that arises from the alteration of the encryption scheme.

The authors give several applications of robustness, and in particular, they modify the Cramer-Shoup PKE scheme to accommodate strong-robustness in a more efficient manner than outlined by their general alteration. They also show other schemes are or are not SROB or WROB secure. It's curious to see that many known encryption schemes do satisfy this notion of security, since the idea is first formalized in this paper. It should, however, be reassuring: no "accidental" decryptions take place under existing encryption schemes.

I'm curious on the following generalization of the problem the authors open with: suppose I wanted to send a message to some *subset* S of the userbase N without unmasking the identity of any member of S, even to other members of the subset. The authors don't address this problem, and so I will do so here.

I desire the following from our encryption scheme GE:

- GE maintains message privacy
- for any subset S of the userbase N, the scheme GE may encipher a message m with some key ek_S such that $Dec_{dk_j}(Enc_{ek_S}(m)) = m$ if $j \in S$ and $\perp$ otherwise.

A notion of adversary advantage is given by the following:

$$\begin{aligned}\mathrm{Adv}^{wsub}_{GE}(A) = P((pars, msk) \xleftarrow{\$} PG; S = (id_{k_1}, \ldots, id_{k_j}), id_0 \leftarrow A;\\ (ek_S, dk_S) \xleftarrow{\$} KG(pars, id_{k_1}, \ldots, id_{k_j}); (ek_0, dk_0) \xleftarrow{\$} KG(pars, id_0); m \leftarrow A : Dec_{dk_0}(Enc_{ek_S}(m)) \in \mathcal{M}).\\ \mathrm{Adv}^{ssub}_{GE}(A) = P((pars, msk) \xleftarrow{\$} PG; S = (id_{k_1}, \ldots, id_{k_j}), id_0 \leftarrow A;\\ (ek_S, dk_S) \xleftarrow{\$} KG(pars, id_{k_1}, \ldots, id_{k_j}); (ek_0, dk_0) \xleftarrow{\$} KG(pars, id_0); C \leftarrow A : Dec_{dk_0}(C), Dec_{dk_S}(C) \in \mathcal{M}).\end{aligned}$$

It is assumed here that the adversary is unable to allow $id_0 \in S$. The idea expressed in the former case is that given any subset of the adversaries choosing, and any "outsider" of that subset, the adversary should be unable to develop a message that is encryped using the key associated to the subset S that produces a valid message under decryption by the outsider. In the latter case, the adversary should be unable to produce a cipher text that decrypts to a valid message for a subset of our group N and an outsider of that subset.

Note that if an encryption scheme is weak or strong subset secure, it is naturally weakly or strongly robust. I don't offer any encryption scheme capable of delivering this security. A primary difficulty in ensuring such a scheme would be ensuring that for every subset S, every member of that subset would be allowed to decrypt messages sent under that subset's collective key. Further, since there are $2^{|}N| - 1$ subsets of N, this may not be a practical request: it could be the keyspace required will be far largely than our encryption scheme can efficiently handle to achieve subset security under any substantially sized N.

It also seems that this situation may be irrelevant—the sender might merely encrypt and send the same message $|S|$ times, each time using a key specific to a particular member of S. The runtime of this simple solution would never be larger than $|N|$ times the runtime of GE, and so may be acceptable in practice.