Problem Set 1

Due: Monday, October 5, 2009, in class.

The relevant part of the course notes are the Introduction and the Block Ciphers chapters.

Collaboration is allowed on this problem set. See the course information sheet for collaboration rules.

Problem 1. [20 points] Discuss some privacy-related problem, created by modern technology, that concerns you. Explain why it concerns you, and what one might do about it. Then discuss some integrity (authenticity) related problem, created by modern technology, that concerns you. Explain why it concerns you, and what one might do about it.

Problem 2. [30 points] Let K be a 56-bit DES key, let L be a 64-bit string, and let M be a 64-bit plaintext. Let

$$\mathsf{DESY}(K \,\|\, L, M) \;=\; \mathsf{DES}(K, L \oplus M)$$
$$\mathsf{DESW}(K \,\|\, L, M) \;=\; L \oplus \mathsf{DES}(K, M)\ .$$

This defines block ciphers $\mathsf{DESY}, \mathsf{DESW}$: $\{0,1\}^{120} \times \{0,1\}^{64} \to \{0,1\}^{64}$.

Present the best possible key-recovery attacks that you can on these block ciphers. Your attacks should use very few input-output examples, not more than three. State the running time of your attacks.

Extra credit

The goal of a key-search attack (such as exhaustive key search) is to find the target key, but, as discussed in the notes and in class, such an attack might find a key that is consistent with the input-output examples but is not the target key. We glossed over this, saying it "usually" does not happen. This problem gives a sense of how cryptographers arrive at this type of conclusion and estimate what "usually" means.

We use what is called the *ideal cipher model.* Let $k, n \geq 1$ be integers. Let $K = 2^k$ and $N = 2^n$ and let $T_1, \ldots, T_K$ be some enumeration of the elements of $\{0,1\}^k$. We consider a thought experiment in which a block cipher is chosen at random. By this we mean that for each key T_i, we choose

Game EKS

procedure Initialize
$T^* \xleftarrow{\$} \{0,1\}^k$; $C^* \leftarrow E[T^*, M^*] \xleftarrow{\$} \{0,1\}^n$
Range$[T^*] \leftarrow \{C^*\}$
Return C^*

procedure E(T, M)
If not $E[T, M]$ then $E[T, M] \xleftarrow{\$} \{0,1\}^n \setminus \text{Range}[T]$
Range$[T] \leftarrow$ Range$[T] \cup \{E[T, M]\}$
Return $E[T, M]$

procedure Finalize(T)
Ret $(T = T^*)$

Figure 1: Game EKS for Problem 3.

$E(T_i, \cdot)$ as a random permutation on $\{0,1\}^n$. Fix a message $M^* \in \{0,1\}^n$ known to the adversary, who, given a ciphertext $C^* = E(T^*, M^*)$ for a random, unknown T^* attempts to find T^*. The adversary can access E (only) as an oracle.

We formalize this via the game EKS of Fig. 1. We will use games a lot so this is a good chance to start getting familiar with them. The game maintains a table E, representing the block cipher, and assumed to initially be $\perp$ (undefined) everywhere. It also associates to each key T a set Range$[T]$ that is initially empty. The game is executed with an adversary A. As this execution continues, the tables get populated, and the block cipher gets slowly defined. First, the **Initialize** procedure executes. It picks a random challenge key T^*, defines $E[T^*, M^*]$ to be a random n-bit string, and returns it to the adversary as the challenge ciphertext C^*. Now the adversary executes, and can make queries of the form T, M to procedure **E**. A query T, M creates the point $E[T, M]$. It is chosen at random, but, to ensure the permutation property of a block cipher, from the set

$$\{0,1\}^n \setminus \text{Range}[T] = \{0,1\}^n \setminus \{\, E(T, M') \,:\, E(T, M') \neq \perp \,\} \,.$$

The test "If not $E[T, M]$" returns true iff $E[T, M]$ is undefined, meaning equal to $\perp$ rather than an n-bit string. The set Range$[T]$ contains all points $E[T, M]$ that are currently defined. When the adversary is done, it outputs its guess T for the value of T^*. This becomes input to the **Finalize** procedure that returns true if $T = T^*$ and false otherwise. The output of **Finalize** is called the output of the game or execution, and we let $\Pr[\text{EKS}^A]$ denote the probability that this output is true. The probability is over the random choices in the game, as well as those of the adversary, if any.

Here we are considering a very simple form of key search where there is only one input-output example.

Now, using this model, we can try to calculate the probability that an attack returns the target key, as opposed to some non-target key consistent with the input-output examples.

Problem 3. [60 points] Let $k, n \geq 1$ be integers. Let $K = 2^k$ and $N = 2^n$. Fix $M^* \in \{0,1\}^n$ and

let $T_1, \ldots, T_K$ be some enumeration of the elements of $\{0,1\}^k$. Consider the following adversary for game EKS:

adversary $A(C^*)$
For $i = 1, \ldots, K$ do
  If $\mathbf{E}(T_i, M^*) = C^*$ then $G \leftarrow T_i$; return G

This adversary calls the $\mathbf{E}$ oracle up to K times as shown. Let $\mathbf{Adv}^{\text{eks}}(K,N) = \Pr[\text{EKS}^A]$. This is the probability that the key G output by A in its execution with EKS equals the target key T^* chosen by **Initialize**.

1. **[35 points]** Prove that

$$\mathbf{Adv}^{\text{eks}}(K,N) \;=\; \frac{N}{K} \cdot \left[1 - \left(1 - \frac{1}{N}\right)^K\right] . \tag{1}$$

2. **[10 points]** It is difficult to get a quantitative feel from Equation (1). We will now lower bound it via a simpler expression. To do so we first recall an inequality. Namely let x be a real number in the range $0 \le x \le 1$. Let m, l be integers such that $0 \le l \le m$ and l *is even*. Then

$$(1-x)^m \;\le\; \sum_{i=0}^{l} \binom{m}{i} (-x)^i . \tag{2}$$

Use this and the result of **1.** above to show that

$$\mathbf{Adv}^{\text{eks}}(K,N) \;\ge\; 1 - \frac{K-1}{2N} . \tag{3}$$

3. **[5 points]** Let k, n be (respectively) the key-length and block-length parameters of DES. Use the result of **2.** to numerically estimate $\mathbf{Adv}^{\text{eks}}(K,N)$ in this case. Do the same when k, n are the parameters of AES.

4. **[10 points]** What do these results tell us about the success probability of an exhaustive key-search attack on DES? What about on AES? Is DES an ideal cipher? Is AES an ideal cipher? Discuss.
