Massachusetts Institute of Technology
6.857: Network and Computer Security
Professor Ronald L. Rivest

Handout 2
February 9, 2015
Due: February 23, 2015

Problem Set 1

This problem set is due on *Monday, February 23* at **11:59 PM**. Please note our late submission penalty policy in the course information handout. Please submit your problem set, in PDF format, on Stellar. *Each problem should be in a separate PDF.* Have **one and only one group member** submit the finished problem writeups. Please title each PDF with the Kerberos of your group members as well as the problem set number and problem number (i.e. *kerberos1_kerberos2_kerberos3_pset1_problem1.pdf*).

You are to work on this problem set with your assigned group of three or four people. Please see the course website for a listing of groups for this problem set. If you have not been assigned a group, please email `6.857-tas@mit.edu`. Be sure that all group members can explain the solutions. See Handout 1 (*Course Information*) for our policy on collaboration.

Homework must be submitted electronically! Each problem answer must appear on a separate page. Mark the top of each page with your group member names, the course number (6.857), the problem set number and question, and the date. We have provided templates for LaTeX and Microsoft Word on the course website (see the *Resources* page).

Grading: All problems are worth 10 points.

With the authors' permission, we may distribute our favorite solution to each problem as the "official" solution—this is your chance to become famous! If you do not wish for your homework to be used as an official solution, or if you wish that it only be used anonymously, please note this in your profile on your homework submission.

Our department is collecting statistics on how much time students are spending on psets, etc. For each problem, please give your estimate of the number of person-hours your team spent on that problem.

Problem 1-1. Security Policy for edX

The massive open online courses (MOOCs) have emerged as a popular mode of learning. An example of an online learning platform which offers MOOCs is edX, which is governed by MIT and Harvard.

Write a security policy for an online learning platform like edX. Make sure to include all the relevant roles, functions, and policies.

If you can't find relevant material on edX as currently implemented, invent new material as appropriate. Try to be as complete as you can, but emphasize the edX-specific aspects.

In particular, what are the different roles that edX have, and what should each principal be allowed to do?

What security goals are the most relevant for edX? For example, when a user should be eligible to receive a verified certificate of achievement?

This problem is a bit open-ended, but should give you excellent practice in writing a security policy. You can find sample solutions from similar questions in previous years on the course website.

Problem 1-2. Double-pad

It is well known that re-using a "one-time pad" can be insecure. This problem explores this issue, with some variations.

In this problem all characters are represented as 8-bit bytes with the usual US-ASCII encoding (e.g."A" is encoded as 0x41). The bitwise exclusive-or of two bytes x and y is denoted $x \oplus y$.

Let $M = (m_1, m_2, \ldots, m_n)$ be a message, consisting of a sequence of n message bytes, to be encrypted. Let $P = (p_1, p_2, \ldots, p_n)$ denote a pad, consisting of a coresponding sequence of (randomly chosen) "pad bytes" (key bytes).

In the usual one-time pad, the sequence $C = (c_1, c_2, \ldots, c_n)$ of ciphertext bytes is obtained by xor-ing each message byte with the corresponding pad byte:

$$c_i = m_i \oplus p_i, \text{ for } i = 1 \ldots n \ .$$

When we talk about more than one message, we will denote the messages as $M_1, M_2, \ldots, M_k$ and the bytes of message M_j as m_{ji}, namely $M_j = (m_{j1}, \ldots, m_{jn})$; we'll also use similar notation for the corresponding ciphertexts.

(a) Here are two 11-character English words encrypted with a "one-time pad". Decide whether they were encrypted with the same pad or with different pads. If they are different pads, then explain why they cannot be the same pad. If they are the same pad, then decrypt the ciphertexts.

```
d2 6b a5 0d 27 6a 34 2d 8e 53 0e
de 6e a7 00 30 74 34 2b 8e 51 11
```

(b) Ben Bitdiddle decided to fix this problem by using a "double-pad".

In his scheme, he generates a list of pads and for each message, picks two distinct pads to use from the list of pads.

$$c_i = m_i \oplus p_i \oplus p_{i'}$$

Ben believes that his scheme is secure because the adversary cannot xor any two messages together to cancel the pad.

You are given 4 of Ben's ciphertexts and told that Ben's list of pads only contained three pads. (These ciphertexts are also in a file in the Handouts section of the class website.) Find two of the ciphertexts you can decrypt and submit the ciphertexts, their decryptions, and relevant pad information along with a careful explanation of how you found them, and any code you used to help find the messages. The most important part is the explanation.

```
56 b0 d6 51 d3 7b f8 9a 52 1e 9b 19 e4 56 01 f4 65 a8 8a b3 9f 5e 2f 24 ac 09 cf 9b 0f
67 8d 16 a8 bd 22 db 25 c3 29 62 32 8e fd 7f 0f b5 6f 6b d8 ca 2e 2d 4a c9 b2 10 fb 48
38 17 b4 21 71 a5 88 2b df 02 88 02 31 32 9a 3d 4c 3a 57 70 4b fb 72 29 68 db 1a 0a 42
ce 81 de 71 de 39
```

```
02 ff 3a dd 8e ec b2 b3 ee 70 b9 aa b4 8e 5e 2f 1a 9e b3 ff 68 60 c5 ee cd ff b2 f2 30
11 5b ca 43 5c 1b f2 ab a8 bc bf 62 be df 50 b6 5f a7 ac 38 99 05 6c 4c 2d 26 c1 f4 60
d1 90 1f ff 46 4a 1f 76 d3 21 73 ce a8 49 26 8c c8 ce 20 3c 80 2e 77 f7 cb 5a e9 80 66
4b 61 d7 a9 17 e4
```

```
1d 6f 80 e5 36 e7 22 58 92 08 43 91 7f 8a 3a ac 1c 0a 4a 6e 9e 47 c0 e5 5a ae 09 19 5c
53 9b a1 95 d0 16 4c d2 0e a3 f8 15 43 56 49 c1 d2 bf a7 98 63 17 38 79 81 aa 96 74 4b
d0 ea c5 a5 16 d3 ee 3e 60 5f 8f f1 ec 18 c4 94 ee 90 1b 6a b4 ba 7d fb db ba 80 e7 59
bd 8f 23 ff f4 ba
```

```
56 b7 d7 18 dc 7b ab 80 5a 1c 84 1e ed 11 40 a4 7a a6 8b b7 93 5d 39 28 f8 22 c1 82 42
71 8a 1e a1 ef 6d 97 08 c2 3a 62 02 93 b4 6c 0d f4 2d 70 ce 8b 31 65 4a 9f ba 0d f1 1c
75 16 ae 34 76 ab 8e 72 93 13 de 17 31 24 d8 35 48 2e 5b 39 4c b5 75 35 6c 95 19 09 49
82 c5 de 71 de 39
```

Problem 1-3. Subverting Cryptography

Watch last month's talk with Bruce Schneier and Edward Snowden:

`https://www.youtube.com/watch?v=7Ui3tLbzIgQ`

Snowden mentions that attacks against encryption usually do not involve breaking the cryptography (math) involved. Suppose you are the NSA. Describe techniques for subverting encryption without breaking the algorithms directly. Consider software, hardware, social, political, and economic issues. List at least 8 techniques. For each technique, include one sentence describing it, and one sentence proposing a defense. These resources can give you some ideas:

`http://www.theguardian.com/us-news/the-nsa-files`

`http://slashdot.org/stories/security`

Try to come up with some novel techniques if you can.