Khoa Khoa học & Kỹ thuật máy tính
Trường ĐH Bách Khoa TP.HCM

Cryptography and Network Security
Lab 5
One-Way Hash Function and MAC

INTRODUCTION

The learning objective of this lab is for students to get familiar with one-way hash functions and Message Authentication Code (MAC). After finishing the lab, in addition to gaining a deeper undertanding of the concepts, students should be able to use tools and write programs to generate one-way hash value and MAC for a given message.

ENVIRONMENT

In this lab, we will use *openssl* commands and a *Hex editor* in **Ubuntu** environment.

1. Installing OpenSSL

You can install the *openssl* commands via *apt-get*

```
sudo apt-get update
sudo apt-get install openssl
```

2. Installing a Hex Editor

In this lab, we need to be able to view and modify files of binary format by using a hex editor called *GHex*. It allows the user to load data from any file, view and edit it in either hex or ascii.

```
sudo apt-get update
sudo apt-get install ghex
```

QUESTIONS AND TASKS

QUESTIONS

1. What types of attacks are addressed by message authentication?
2. What are some approaches to producing message authentication?
3. What is a message authentication code?
4. What is the difference between a message authentication code and a one-way hash function?
5. What changes in HMAC are required in order to replace one underlying hash function with another?

TASKS

Task 1. Generating Message Digest and MAC

In this task, we will play with various one-way hash algorithms. You can use the following openssl dgst command to generate the hash value for a file. To see the manuals, you can type man openssl and man dgst.

```
% openssl dgst dgsttype filename
```

Please replace the dgsttype with a specific one-way hash algorithm, such as -md5, -sha1, -sha256, etc.

In this task, you should try at least 3 different algorithms, and describe your observations. You can find the supported one-way hash algorithms by typing "man openssl".

Task 2. Keyed Hash and HMAC

In this task, we would like to generate a keyed hash (i.e. MAC) for a file. We can use the -hmac option (this option is currently undocumented, but it is supported by openssl).

The following example generates a keyed hash for a file using the HMAC-MD5 algorithm. The string following the -hmac option is the key.

```
% openssl dgst -md5 -hmac "abcdefg" filename
```

Please generate a keyed hash using HMAC-MD5, HMAC-SHA256, and HMAC-SHA1 for any file that you choose. Please try several keys with different length.

Do we have to use a key with a fixed size in HMAC? If so, what is the key size? If not, why?

Task 3. The Randomness of One-way Hash

To understand the properties of one-way hash functions, we would like to do the following exercise for MD5 and SHA256:

1. Create a text file of any length.
2. Generate the hash value H1 for this file using a specific hash algorithm.
3. Flip one bit of the input file. You can achieve this modification using Ghex.
4. Generate the hash value H2 for the modified file.
5. Please observe whether H1 and H2 are similar or not. Please describe your observations in the lab report.

You can write a short program to count how many bits are the same between H1 and H2.

You need to submit a detailed lab report to describe what you have done and what you have observed; you also need to provide explanation to the observations that are interesting or surprising. In your report, you need to answer all the questions listed in this lab.