Exercise 1, Password cracking and the CIA triad

T-110.4200/6 course staff

deadline Friday 2.10.2009 23:55

Introduction

This is the first exercise paper for the dual courses T-110.4200 and T-110.4206. Even though most of the exercises will be done on the course Moodle webpage at `http://hiljainen.cs.hut.fi/moodle/`, we will still publish an exercise paper for each round at Noppa for the sake of clarity, even if the paper would just instruct you to go to some other webpage.

1 Crack your own password

1.1 Background

Systems don't usually store passwords in plaintext because of obvious confidentiality issues. What they do instead, is that they run the password through a *hash function* and store the result. When a user gives his or her password, the hash of the given password is computed and compared to the stored hash.

The hash function transforms the password $f(x) \rightarrow y$. The most basic idea of the hash function is that it is one-way, i.e. that it is impossible to generate another function so that $g(y) \rightarrow x$. A very good hash function is also an injection so that no two inputs generate the same output.

In the general case, if you know the hash of a password you are none the wiser about the actual password because of the one-way property of hash functions. However, if you happen to know some features of the password, you may be able to limit the potential *password space* so that we can compute the hashes for each possible password in a sensible time. In general, the password space is n^k where n is the number of letters that can appear in the password and k is the length of the password.

1.2 The actual assignment

On the course web page there is a list of usernames and password hashes. Your username is your student number with an X added to the beginning.

The password space of the passwords generated for you is limited. Each password starts with your student number followed by dash and 3 random lower-case ASCII alphabet characters. So if your student number is 12345X your password would be 12345X-xxx where xxx is the 3 random characters. The hash function we used was *md5*.

Write a program or a script to crack your password. The example solution is written in *bash* but you can use e.g. perl or python if you know them.

Once you know your password, use it to log in to Moodle and *change your password!*

If you crack your password on IT service or Niksula machines, remember to use nice(1) to make sure that you won't accidentally use up all the resources on some machine. nice(1) is a utility for lowering the priority of a process.

1.3 Hints

- some machines have a command line program or alias called *md5*, others have *openssl* which can also be used to compute hashes
- for instance the spices at Niksula have md5 (sokeri, suola, pippuri, cayenne.niksula.hut.fi)
- you are not asked for a general solution but only a solution for passwords of length 3. Perhaps using 3 nested for loops would do the trick.
- If you are wondering why we dare to publish all the crackable hashes, see `http://www.finlex.fi/en/laki/kaannokset/1889/en18890039.pdf` or `http://www.finlex.fi/fi/laki/ajantasa/1889/18890039001` Chapter 38, section 8.

2 CIA

If you missed the lectures, review the Wikipedia article on CIA at `http://en.wikipedia.org/wiki/CIA_triad`. Please keep in mind that Wikipedia is not a perfectly reliable source. However, at least the definitions of the key concepts seemed to be factually correct when the course assistant wrote this assignment.

After you feel that you understand the definitions of *confidentiality*, *integrity* and *availability*, answer the questions in Moodle. Nothing fancy is

required and just understanding the definitions should be quite sufficient to pass the exercises. You need to retake this quiz until you get all the answers right.