The University of Queensland
School of Information Technology and Electrical Engineering

Semester 2, 2017

COMS3000/7003 – Tutorial 2, Answers

Q1) Explain the difference between identification and authentication.

Identification is saying who you are. Authentication is providing evidence for that claim.

Q2) What are the key properties of a cryptographic one-way hash function h()?

- Take arbitrary length input and produce fixed-length output.
- h(x) is easy to compute

- One-way property (also called 'pre-image resistance')
  - Impossible to 'invert', i.e. for a given y=h(x), it is computationally infeasible to find x

- Collision resistance
  - Weak collision resistance (also called "2nd pre-image resistance")
    - For **a given x1** and y=h(x1), it is impossible to find another input x2 so that h(x1) = h(x2)
    - This is a difficult attack, so is easier to resist.

  - Strong collision resistance
    - It is impossible to find **any two** inputs x1 and x2 so that h(x1) = h(x2)
    - (It is much easier for an attacker to find a collision if he/she can choose *both* values x1 and x2 compared to having no choice in one and having to find x2 for a given x1. Therefore, resistance to this easier attack is called 'strong collision resistance'.)

Q3) Consider the following function h():
The function accepts messages of any length. h() returns a bit-string of 32 0s if the input has an even number of characters and returns a bit string of 32 1s if the input has an odd number of characters.

a) Evaluate this function against the key properties of cryptographic one-way hash functions provided in the Lecture.

h() takes arbitrary length input and produces a fixed-length output.
h() is easy to compute.

h() is one-way, according to our definition. It is impossible to determine the input from the output, since the output has only two possible values and there are an infinite number of possible inputs.

h() is neither weak nor strong collision resistant.

- By randomly choosing an input, we have a 50% chance of getting either of the two possible outputs. It is therefore easy to find an input that has a specified output (either 111…1 or 000…0). It is also easy to find two inputs with the same output.

b) What's the probability of a collision in h() for two randomly chosen inputs x1 and x2?

If we consider the length of the input to be random, then the output is random with a 50% probability for each of the two outcomes. The chance of a collision is the same as getting twice 'heads' or twice 'tails' when tossing two coins, which is 50%. P(collision) = 0.5

c) What's the probability of a collision between two randomly chosen inputs x1 and x2 for an ideal ("random oracle model") cryptographic one-way hash function with a 32 bit output (n=32) ?

$P(\text{collision}) = 0.5^n = 0.5^{32} = 2.33 * 10^{-10}$

For the following questions, you can use one of the many online Cryptographic Hash Calculators, such as: http://onlinemd5.com/

Q4) What is the (4-bit) hash of "1234567890"? (The md5 or SHA-1 output is 128-bit or 160-bit in hexadecimal representation, so to get the 4-bit hash, simply take the first 4 bits) What happens to the hash value if one bit of this input is changed, so the string becomes "1234567880"? Try some other values.
Does this hash function fit the "elf coin tossing" or "random oracle" model discussed in the lecture?

md5("1234567890") = e (e807f1fcf82d132f9bb018ca6738a19f)
md5("1234567880") = 0 (04aa751d28bce6efa9e4cc25e69b4f90)
sha1("1234567890") = 0 (01b307acba4f54f55aafc33bb06bbbf6ca803e9a)
sha1("1234567880") = 9 (9d13dad3c8296570a21658a7803c4c02484f1d2e)

A small change in the input results in a totally different ("random looking") output. From what we can observe, md5 and SHA fit the random oracle model.

Q5) (One-way property.) Try to find a string that has a hash value of 7 (0111). Propose a systematic search pattern. How many attempts does it take?

I tried 0, 1, 2, 3, … with the (4-bit) md5 hash.

md5("0") = c	md5("1") = c	md5("2") = c
md5("3") = e	md5("4") = a	md5("5") = e
md5("6") = 1	md5("7") = 8	md5("8") = c
md5("9") = 4	md5("10") = d	md5("11") = 6
md5("12") = c	md5("13") = c	md5("14") = a
md5("15") = 9	md5("16") = c	md5("17") = 7

It took 18 tries.
(Using SHA-1 with the same search pattern takes only 4 tries.)
Using the random oracle model, we expect to find the string after 16 tries on average. All outputs are equally likely and have a probability of 1/16. So on average, we need 16 tries.

Q6) ("Weak collision resistance".) Try to find a string that has a hash value that is the same as the hash of "My student number is xxxxxxxx." (no 's' in your student number). Propose a systematic search pattern. How many attempts did this take?

I calculated the hash md5("My student number is 31849753")=d. Then I used the same sequence of attempts as in the previous question. It would have taken 11 attempts (0-10).

Q7) ("Strong Collision Resistance") Find any two strings (containing your student number) that hash to the same value. Propose a systematic search pattern. How many attempts does this take?

I calculated the hash md5("My student number is 31849753")=d. Then I changed the first character "M" using the sequence above without changing the student number. I got the first collision on the 4th(0,1,2,3) try:
1: 2y student number is 31849753 = 5 and 3y student number is 31849753 = 5; then
2: 4y student number is 31849753 = 0 and 9y student number is 31849753 = 0; then
3: 7y student number is 31849753 = b and dy student number is 31849753 = b; then
4: 0y student number is 31849753 = 2 and ey student number is 31849753 = 2; then
5: cy student number is 31849753 = 6 and gy student number is 31849753 = 6; then
6: 0y student number is 31849753 = 2 and hy student number is 31849753 = 2; then
7: ey student number is 31849753 = 2 and hy student number is 31849753 = 2; then
8: 2y student number is 31849753 = 5 and iy student number is 31849753 = 5; then
9: 3y student number is 31849753 = 5 and iy student number is 31849753 = 5; etc.
After just 27 substitutions (just digits 0-9 and a-q lowercase letters), without even getting to uppercase or symbols, I had produced collisions with ALL previous 26 hash results. (Some of the 16 possible outputs had collided multiple times.)