Kieren Ng

2/7/18

Homework 3

2.9

Digital signature- a signature from the sender that encrypts documents

Encryption and Decryption- Sender encrypts the message with the public key of the recipient

Exchange of keys- Exchange a session key between two sides

2.10

The secret key is the key in a symmetric encryption, and the private key is used in public key encryption, and is the pair to the public key.

2.5a. Yes, Bob would detect this because the auth would not match.

2.5b. Bob will not detect this, because the data isn’t changed, so the auth would match.

2.5c. Yes in both cases, he can check the auth(x)s sent from Oscar to see if it matches his, and he can also check Alice’s public key to authenticate x.

2.5d. Alice is the only one who can generate auth(x) since it’s generated with a private key, so she can clear the message.

2.6

It’s not true because there is a limit on the outputs, 2^n I believe. If the message length is arbitrary like it says in the question it could be super large, so there would be some overlap and the function would not be one to one.

3.1

Four factors- what you have, what you are, what you know, and what you do

3.1a. This unsuitable because it is too short, it’s only five characters long. It’s good that it uses both letters and numbers though.

3.1b. This password isn’t a dictionary word, and is pretty hard to guess, so it’s suitable.

3.1c. It’s a name, possibly the user or someone related to the user, so it could be easy to guess- so not suitable

3.1d. This is just the name of a place, could be vulnerable- so not suitable probably

3.1e. This is just another name of a person, who’s famous too- so it’s not suitable

3.1f. This uses two common words, but separated by a number, I don’t think it is as vulnerable to dictionary attacks because there’s both, and theres numbers too, so suitable.

3.1g. This is just numbers, probably insecure because it may be guessable because it’s a sequence.

3.1h. This seems to be just a random string of letters, so probably not guessable by a person or not vulnerable to a dictionary attack, so its suitable.

3.3a. 26 possibilities for 4 slots each, so its 26\*26\*26\*26 = 26^4 = 456978 seconds /2 = 228488 seconds = 63.5 hours

3.3b. 26/2 = 13 \* 4 slots = 52 seconds

3.4a. Each position in length k can be any r value, so r^k. To pick correctly on the first try is just that, 1/r^k.

3.4b.

3.4c. The target elements have length p instead of length k, so its r^p, the probability is 1/r^p.

3.6

95characters^10slots = 5.98\*10^19 passwords

At 6.4 million per second, test will take 296653 years.