h_2

Connor Taffe

March 2, 2016

1. Four digits with three possible keys: $b^l = 3^4$.

2. (a) A cryptographic hash offers the advantage that the key can never be reversed, but if someone has the account number they can easily create a magnetic strip that will work for that account.

   (b) Asymetric keys are better because even if the key is discovered, new cards cannot be created, as that private key will be safe.

   (c) Symmetric keys will prevent anyone from creating a card without the bank's key, but now the keys must be distributed to every place the card is used. If it is ever discovered, an attacker could create new cards, as well as steal the number from customers.

3. $1 + 2^1 + 2^2 = 7$

4. 200,000 words, the salt is just appended to each word before hashing. The point of a hash is to make a dictionary useless as it must be regenerated for each hash, a task that takes a long time. Unless you don't know the hash, then it is $200,000 * 2^{24}$

5. $500,000^{100}$, the size of the dictionary to the number of entries is the possible solution space for the 100 users' passwords

6. Yes, he is the user and the user has read/write permissions.

7. A salt is not meant to be secret, thusly the search spaces should be equivalent, that said the actual search space is smaller because the set of possible values are equal to the entropy of the username, which is significantly smaller than that of a long random value. So theoretically the search space is much smaller, and it could be possible to find a faster way to generate the resulting password hash or modify an existing dictioinary.

8. $$2^{20} = 1,048,576, 64^{10} = 281,474,976,710,656 \tag{1}$$

   No. Just a basic eight character password consisting of upper/lowercase letters, numbers and three symbols (base 64 encoding) garnered a 10^8 times larger space.

9. Interpret the following wher $n \in \mathbb{Z}$

$$20!(2^{20}) = 2,551,082,656,125,828,464,640,000 = n(10^{25}) \tag{2}$$

$$85^{16} = 8,953,136,790,196,197,357,146,289,012,736 = n(10^{31}) \tag{3}$$

It is much better, but still no match against a standard password with all the letters, numbers, and symbols at a length of 16. So yes, very much an improvement. For brute force a single guess is still has a 2^-20 chance of being correct, but one can't assume a previously guessed order will not be valid this time, so permutations come into play.