COSC130 Fundamentals of Cybersecurity and Privacy

Tutorial Week 6 Solutions

1. For each of the following ways to store passwords, comment on how secure they are and how efficiently then can be performed.
    a. Store passwords in plaintext
    b. Encrypt all passwords using the same key, and store the cyphertexts; store the encryption key in a separate database requiring higher privileges to access
    c. Encrypt each password using a unique key; for each password store the cyphertext, and the unique key used for that password
    d. Hash all the passwords, and store hash codes
    e. Append the same salt to each password, hash it and store the hash codes; store the salt in a different, more secure database
    f. Append a unique salt to each password and hash it; for each password, store the obtained hash, and the unique salt used
    g. For each password create a salt that is the same as the first two characters in the password; append salts to passwords and hash them; for each password, store the obtained hash
    h. Append a unique salt to each password and hash it; for each password, store the obtained hash, and a ciphertext of the unique salt used; all salts and encrypted with the same key, which is stored in a different secure database

Solution:

a. Store passwords in plaintext

Username	**Password**
John	letmein
Mary	letmein

Not secure at all; an attacker who manages to obtain the file with passwords has immediate access to all the passwords and usually usernames.

Very efficient for server.

b. Encrypt all passwords using the same key, and store the cyphertexts; store the encryption key in a separate database requiring higher privileges to access

Username	**Password (encrypted)**
John	jsdchy
Mary	jsdchy

An attacker who steals the file can see which passwords are the same.

Symmetric encryption is fast, so this way is efficient for the server.

c. Encrypt each password using a unique key; for each password store the cyphertext, and the unique key used for that password

Username	**Password (ciphertext)**	**Key**
John	mngmht	727639
Mary	Usds7c	Uihh78

An attacker can use the keys to decrypt passwords so in terms of security this is almost no better than storing passwords in plaintext.

It is efficient for the server.

d. Hash all the passwords, and store hash codes

Username	**Password (hashed)**
John	jsdchy
Mary	jsdchy

An attacker can see which passwords are the same.

It is a bit less efficient than encryption for the server because hashing is designed to be slow.

e. Append the same salt to each password, hash it and store the hash codes; store the salt in a different, more secure database

Username	**Password (the same salt appended, then hashed)**
John	Yd67en
Mary	Yd67en

An attacker can see which passwords are the same.

It is a bit less efficient than encryption for the server because hashing is designed to be slow.

f. Append a unique salt to each password and hash it; for each password, store the obtained hash, and the unique salt used

Username	**Password (unique salt appended, then hashed)**	**Salt**
John	Iweu79	Nzh689
Mary	8s9iw7	Nz6zy7

It is pretty secure, the attacker can't recognise the same passwords; they can still create a rainbow table for each password but that is a lot of work.

It is a bit less efficient than encryption for the server because hashing is designed to be slow.

g. For each password create a salt that is the same as the first two characters in the password; append salts to passwords and hash them; for each password, store the obtained hash

Username	**Password (corresponding salt appended, then hashed)**	**Salt**
John	Ncha6&	le
Mary	Ncha6&	le

An attacker can see which passwords are the same, plus they have the first two characters of each password.

It is a bit less efficient than encryption for the server because hashing is designed to be slow.

h. Append a unique salt to each password and hash it; for each password, store the obtained hash, and a ciphertext of the unique salt used; all salts and encrypted with the same key, which is stored in a different secure database

Username	Password (unique salt appended, then hashed)	Salt (ciphertext)
John	Nhj^78	H%6Ti8
Mary	GH^yn9	BnT56*

More secure than f), as the attacker cannot even compute the rainbow table for each password, as the salt is encrypted (unless they break the encryption first or obtain the key)

It is a bit less efficient than encryption for the server because hashing is designed to be slow. Also needs to do additional encryption.

2. Rank the following passwords by their strength; justify your ranking.
    a. Bnjd^57j&*
    b. apple1
    c. Dfgnua
    d. 000000
    e. G56&hk
    f. Apples

Solution:

1) 000000 – trivial password

2) apple1 – uses lowercase letters and digits; the last character is a digit – a bad idea as many people would use digits at the end

3) Apples – uses 52 characters in total

4) Dfgnua uses only letters but they don't form a word – looks like a random combination of letters; the first letter is capitalised – a bad idea as it is easy to guess

5) G56&hk – looks random, except that the capital letter is first, uses everything (letters in both cases, digits, and special characters)

6) Bnjd^57j&* - same as the previous one but longer

To illustrate the importance of the length of the password for the brute force attack, suppose we only use 10 different characters.

How many passwords are there that are one character long? 10

2 characters long? 10 * 10 = 100

3? 10*10*10=1000

3. Whose responsibility is to protect against cybercrime? Select the best answer.
    a. Federal police
    b. Local police
    c. Cybersecurity professionals
    d. Everybody
    e. Government

4. Which one of the following can be used as an attack on passwords?
    a. Looking over someone shoulder while they are withdrawing money from an ATM
    b. Dictionary attack
    c. Brute Force attack
    d. All of the above
    e. None of the above

5. Research password managers available in 2024. Identify features that are commonly supported. Create a table to compare at least 5 password managers in respect to price and identified features.