Braden Scribner - 4/4/2024

Hashing Overview

****

What is a hash?

Firstly, let's take a look at what a password hash even is.
A Password Hash is the result returned after running a Hashing Algorithm over a plaintext password. These garbly hashes are the things actually stored and checked against every time you log into a service.
Hashing is used because a system should never store its passwords in plain, readable text. If they did, anybody who could see where passwords are stored, could read anybody's password.
For the same reason, **hashing algorithms only work one-way**; you can't just run it back against the hash to get the original string.

Examples of Hashing Algorithms

Below are some common hashing algorithms:

- MD5
- MD4
- SHA-512
- SHA-256
- bcrypt
- yescrypt

Braden Scribner - 4/4/2024

Examples of Hashed Words

In the table below, you can see the plaintext string, followed by its corresponding MD5 hash.

john	527bd5b5d689e2c32ae974c6229ff785
John	61409aa1fd47d4a5332de23cbf59a36f
hamburger	140d3ea2b0c7a720b8fcc236deedd04f
supercalifragilisticexpialidocious	08206e04e240edb96b7b6066ee1087af
1234	81dc9bdb52d04dc20036dbd8313ed055

Notice how, despite the difference in length between many of the hashes, the output remains the same length? – just trust me, the spacing is a bit weird 😄 – and how 'john' and 'John' gave completely separate hashes? That's because hash algorithms strive for strict uniqueness between hashes. If the hash for 'john' was super similar to the hash for 'John', somebody attempting to crack that hash would be able to narrow down your password a lot faster.

How do Hash Cracking Programs work?

So we understand how a hashing algorithm works, but now how do we 'crack' it? Like I said before, you can't just run the hashing algorithm back against it, so what do we do?
Imagine you have been given a hash, and told that the corresponding password was written on last week's grocery list. How would you find out which password it is? Simple. You would just take every word from your grocery list, and run the algorithm on it! Then, you could easily match the given hash to the calculated one, and find your mystery word.

Programs like JtR work just like that. You specify the algorithm, the hash(es) to crack, and the password attack mode, and then it runs hundreds of thousands of words through that algorithm, until it finds your match. Read more about attack modes below.

What types of Password Attacks are there?

In programs like JtR, there are often 3 cracking modes: Incremental(Brute-force), Wordlist, and Hybrid. Read each section below to learn about each.

Braden Scribner - 4/4/2024

Incremental Mode

Cracking a password in incremental mode is the least efficient, ‘dumbest’ way to crack a password. It simply tries every combination of characters, forever, until it ***eventually*** finds the right one. Incremental attacks are usually unsuccessful, because of the time and computing power they take to properly run their course. For this reason, Rules are often supplied. These constrain the system to using only certain symbols, or keyspaces, using certain patterns, lengths, etc. Despite using Rules, it is still much more efficient to use other information you have, with a Wordlist or Hybrid attack.

Wordlist Mode

A wordlist attack, or dictionary attack, is exactly like that grocery list example from earlier. A list is supplied (usually containing leaked or common passwords), and the algorithm is applied to each entry in the list, stopping and returning the match.
The most commonly used password list for cybersecurity activities is the RockYou password list. The RockYou list contains **over 10 million passwords**, and was compiled after the 2009 breach that exposed 32 million accounts information.

Wordlists dont always contain prefabricated passwords, though. Sometimes, like many of the ones on the SecLists collection, they are just the jumping-off point for a Hybrid/Mask attack.

Hybrid Mode

A hybrid attack, or mask attack as it's called in john, is a – you guessed it – hybrid of a wordlist attack and bruteforce attack. Usually, this just means altering or “fuzzing” entries from a list. If you know that the password contains somebody’s first name, last initial, and then a 2 digit identifier, (jamesx00), you could use a wordlist of common first names, and then ‘fuzz’ with a single letter and then 2 numbers. While this seems like it would take a long time, you have to remember that these tools **run upwards of a thousand attempts every minute**.

A little deceptively, a mask attack in john does not *require* a list; if you know the pattern of a password, like if it was just the first initial instead of the whole name, you could instead just run a mask attack of “?l?l?d?d” (two letters and two numbers).

****

Braden Scribner - 4/4/2024

Wrap-Up

Now that you understand the basics of password cracking, you can move on to the activities. These activities and lessons would be great to use in any Cryptography, Social Engineering, or other related unit in a Cyber class.

Credit:

Hey! This document was written by me, Braden Scribner. I really enjoy Cybersecurity and doing these sorts of activities, so I hope you enjoyed my attempt at explaining it!