U.S. DEPARTMENT OF ENERGY'S
CYBERFORCE® PROGRAM

CyberForce® 101

# Hashing

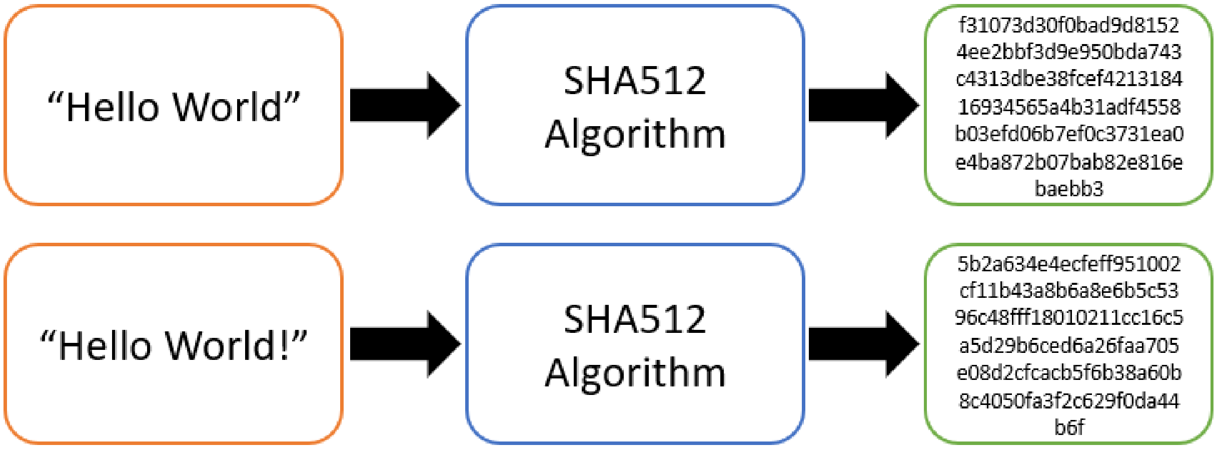
October 2023

cyberforcecompetition@anl.gov

# Hashing 101

## Hashing

Hashing is the process of putting messages or data (plaintext) through an algorithm that makes it unrecognizable. A hashing algorithm must be irreversible. This means there is no way to reverse the hashed product (ciphertext) to retrieve the original message. The hash should also contain both numbers and letters to help meet the irreversible requirement. This means that hashing is not susceptible to most attacks. The only known attack that works is a brute force attack where an attacker systematically tries different passwords until they find the correct one.



"Hello World"
SHA512 Algorithm
f31073d30f0bad9d8152 4ee2bbf3d9e950bda743 c4313dbe38fcef4213184 16934565a4b31adf4558 b03efd06b7ef0c3731ea0 e4ba872b07bab82e816e baebb3
"Hello World!"
SHA512 Algorithm
5b2a634e4ecfeff951002 cf11b43a8b6a8e6b5c53 96c48fff18010211cc16c5 a5d29b6ced6a26faa705 e08d2cfcacb5f6b38a60b 8c4050fa3f2c629f0da44 b6f

## Hash vs Encryption

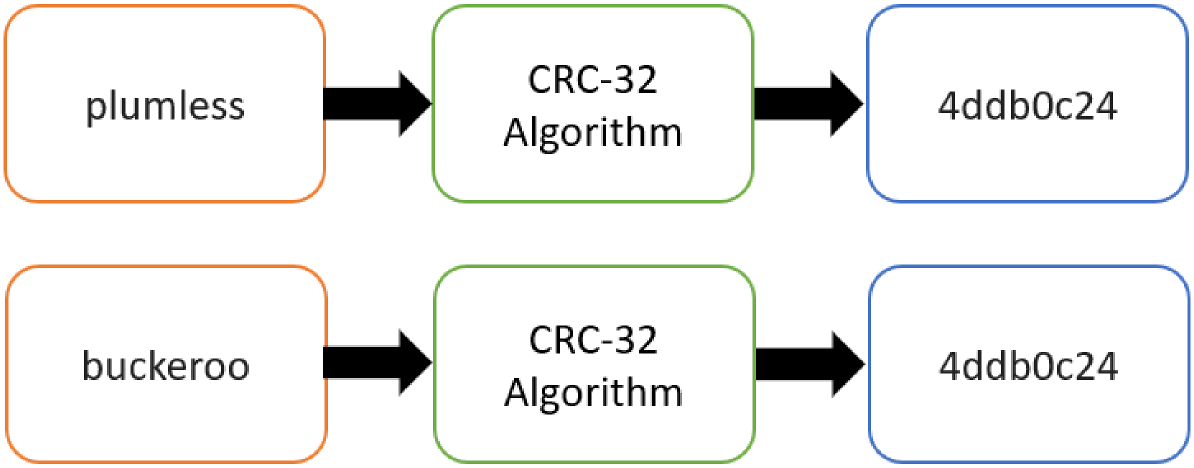
Hashing is different from encrypting because a hash is irreversible. Encryption must be able to be reversed (go from plaintext to ciphertext and back to plaintext), so the decryption part is just as important as the encryption.

## Salting

Salting is the process of adding characters to a plaintext message before running it through a hashing algorithm. This makes the data more secure. For example, if two people have the same passwords and they use the same hashing algorithm, then they will have the same hash. By adding a different salt to each person's password, an attacker cannot tell if the two passwords are the same because the added salt will completely change the hash.

## Collisions

A collision happens when two or more plaintext messages are put through the same algorithm and have the same hash. This is very rare and when it happens the algorithm is no longer considered secure because it allows attackers to trick the system into believing they have the correct password. The algorithm is usually faded out of use once a collision has been discovered.



## Rainbow Tables

Rainbow tables are a large collection of password hashes. They find the hashes of everything that is placed in the table. It stores the password and the hash. Password cracking software then uses rainbow tables to compare the system's hashed password to the hashes in the table. This process is losing popularity because more systems are implementing precautions such as salts to defend against these attacks.

## Uses

Hashes have many uses. For example, most systems use hashes to store passwords in a secure way. When a user enters a password, the system hashes it and compares it to the stored hash of the correct password. This is preferable to storing the passwords as plaintext, because if an attacker gets into the system, it prevents them from easily collecting all the system's password.

Hashes are also used in integrity checks. If someone wants to share a document with another person/group, they can send/post the file's hash with the original file. That way the recipient can hash the file (using the same algorithm) and if the hashes match, they know the file has not been corrupted.

## Sources

- Hashing, Hashing Algorithms, and Collisions - Cryptography - Practical TLS
- https://emn178.github.io/online-tools/sha256.html
- Hash Collision Probabilities (preshing.com)
- https://www.techtarget.com/searchsecurity/definition/salt
- https://www.comparitech.com/privacy-security-tools/password-strength-test/#password-test-tool