DHBW
Duale Hochschule
Baden-Württemberg
CAS
Center for Advanced Studies

IT-Security Cryptography and Secure Communications

Excercise: **Hash Functions**
Lecturer: *Prof. Dr. Michael Eichberg*
Version: 2023-10-12

1. Why is a simple hash function that computes a 256bit hash value by performing an XOR over all blocks of a message generally inapproriate?
2. Compute the MD5 and SHA256 hash of a file that just contains "0x00" values. What happens if the file is longer?

   (Hint: use `dd` to read zeros from `/dev/zero` and `md5(sum)` and `sha(256)sum` tools on your computer. On Windows use the Linux subsystem, a virtual machine, ...)
3. Why are second-preimage resistance and collision resistance not relevant if the hash algorithm is used for password hashing?
4. Where is the problem when you apply a simple XOR to the 64bit blocks of a message and then encrypt the entire message using DES with CBC mode. (Hence, our hash is also encrypted!)
5. Determine the passwords hashed with plain MD5:

   1. `81dc9bdb52d04dc20036dbd8313ed055`

      Hint: the password is short and just consits of digits!
   2. `7c6a180b36896a0a8c02787eeafb0e4c`

      Hint: the password consists of letters and digits. However, it is a password used very frequently.

   You can use Hashcat (https://hashcat.net/hashcat/) or write a bash script or develop a small solution in the programming language of your choice.
6. Why is it important that hashing of passwords is deliberately inefficient while other cryptographic hash functions strive for efficiency? In both cases, we want to thwart an attack!