Introduction to Computer Security – G6077

Dr. Imran U Khan

Engineering and Informatics

Sussex University

Overview

- Introduction and checksum method
- Key characteristics
- General design of hashing algorithms
- Popular hashing algorithms
- Applications of hashing
- Salting hashing
- Collisions

Introduction and checksum method

- Cryptography is to provide privacy, prove identity and show integrity
- Secret key used to provide secrecy/privacy but we need a method to check integrity of a message
- Hashing used either to hide the original contents of a message or to check the integrity of data
- In the past, checksum method used to check integrity of data e.g.

Adding a value to list of number so that the total would be a multiple of 9

If 4, 5 and 13 were to be sent

4+5+13 = 22

What number shall we add to 22 to get a multiple of 9? Ans: 5 (22+5=27)

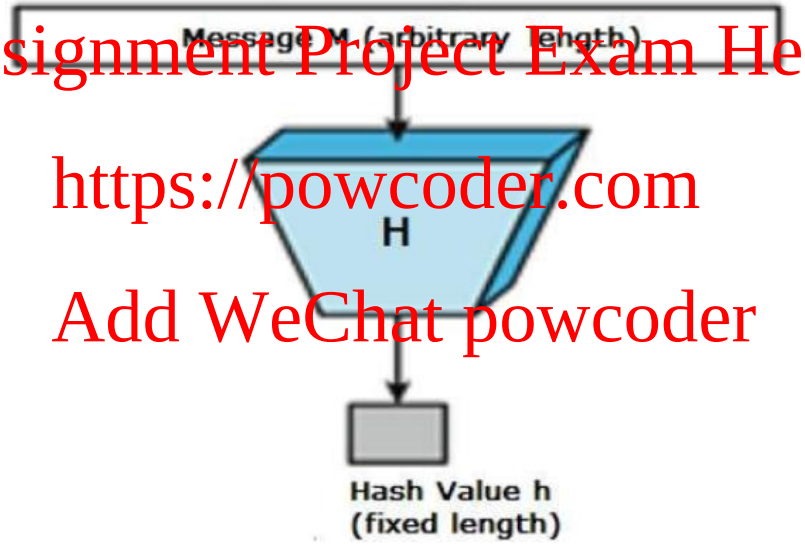
The following values/data will be rejected e.g.

4+5+13 with a checksum of 8

Will provide 30 (4+5+13+8(cs)) which is not a multiple of 9

- Hashing methods /functions: a mathematical function that converts a numerical input value into another compressed numerical value
- Input to the function varied in length but output is always of fixed length

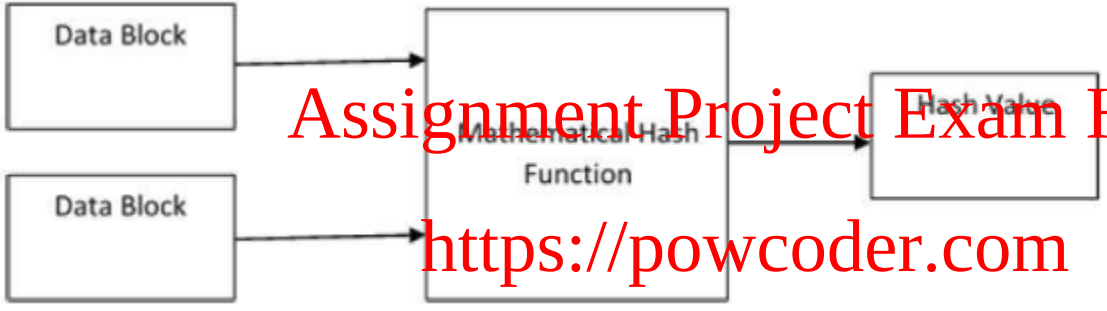
- Returned values are called hash values or message digest



Key characteristics of Hash functions

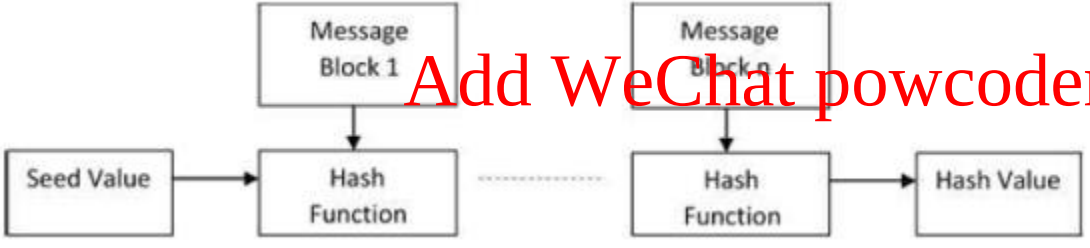
- Must be hard to retrieve the original text from the hash value
- Must be hard to find two different inputs of any length that result in the same hash. In other words, for a hash function h, it is hard to find any two different inputs x and y such that h(x) = h(y)

General design of hashing algorithms



- Function operates on two fixed-size blocks
- Block size varies depends on the algorithm, 128 bits to 512 bits

- Hashing algorithm involves rounds of above hash function like a block cipher
- Each round takes an input of fixed size of message block and output of the last round



Popular Hash functions

Message Digest (MD)

- MD2, MD4, MD5 and MD6
- 128-bit hash function
- MD5 most popular, widely used hash function
- In 2004 collisions were found in MD5
- Analytical attack reported
- MD5 is compromised and not recommended anymore

Secure Hash Function (SHA)

- Four SHA algorithms; SHA-0, SHA-1, SHA-2, and SHA-3
- SHA-0: 160-bit, released in 93 by NIST
- SHA-1: most popular and widely used, employed in protocols including SSL (Secure Socket Layer), in 2005 method was found for uncovering collisions for SHA-1
- SHA-2: SHA-224, SHA-256, SHA-384 and SHA-512, no successful attack on SHA-2 variants, weakness of SHA-2 is that it follows design principle of SHA-1

- SHA-3: 2012, Keccak algorithm, offers efficient performance and stronger resistance for attacks

RIPEMD

- RACE Integrity Primitives Evaluation Message Digest - known as European family of Hash functions
- RIPEMD-128, RIPEMD-160, 256 and 320
- Based on MD4 design principles, provide questionable security

Whirlpool

- 512-bit hash function
- Whirlpool-0 and Whirlpool-T
- Derived from AES

LM Hash

LM Hash. LM Hash is used in many version of Windows to store user passwords that are fewer than 15 characters long.

SHA-3

SHA-3. SHA-3 was known as Keccak and is a hash function designed by Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. MD5 and SHA-0 have been shown to be susceptible to attacks, along with theoretical attacks on SHA-1. NIST thus defined there was a need for a new hashing method which did not use the existing methods for hashing, and setup a competition for competing algorithms. In October 2012, Keccak won the NIST hash function competition, and is proposed as the SHA-3 standard.

Tiger

Bcrypt

Bcrypt. This creates a hash value which has salt.

RIPEMD

RIPEMD (RACE Integrity Primitives Evaluation Message Digest) and GOST. RIPEM160. RIPEMD is a 128-bit, 160-bit, 256-bit or 320-bit cryptographic hash function, and was created by Hans Dobbertin, Antoon Bosselaers and Bart Preneel. It is used on TrueCrypt, and is open source. The 160-bit version is seen as an alternative to SHA-1, and is part of ISO/IEC 10118

Tiger. Tiger is a 192-bit hash function, and was designed by Ross Anderson and Eli Biham in 1995. It is often used by clients within Gnutella file sharing networks, and does not suffer from known attacks on MD5 and SHA-0/SHA-1. Tiger2 is an addition, in which the message is padded with a byte of 0x80 (in a similar way to MD4, MD5 and SHA), whereas in Tiger it is 0x01. Otherwise the two methods are the same in their operation.

Author: Prof Bill Buchanan

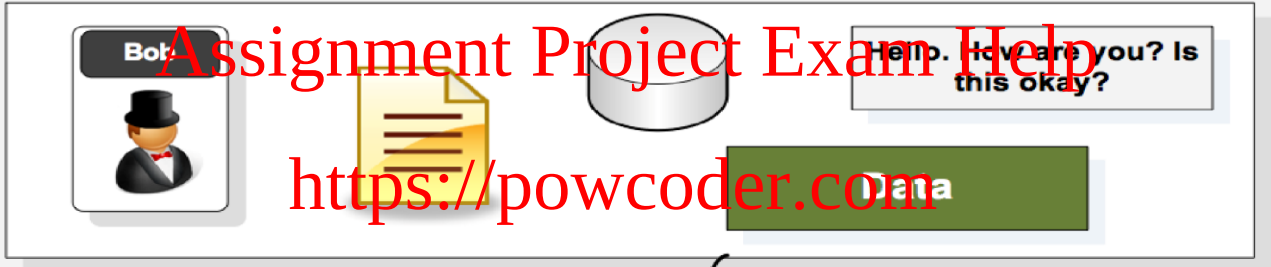
Murmur

While hashing methods such as MD5 and SHA-1 use cryto methods, the Murmur and FNV hashes uses a non-cryptographic hash function. The Murmur hash, designed by Austin Appleby, uses a non-cryptographic hash function. This can be used for general hash-based lookups. It has a good performance compared with other hashing methods, and generally provide a good balance between performance and CPU utilization. Also it performs well in terms of hash collisions.

FNV

FNV (Fowler–Noll–Vo) is a 64-bit non-cryptographic hash function developed by Glenn Fowler, Landon Curt Noll, and Phong Vo. There are two main versions, of which 1a is the most up-to-date version.

How do we get a finger-print for data?



With a fingerprint we can hopefully tell if Eve has modified any of the data

Solved by Prof Ron Rivest with the MD5 hash signature.

Author: Prof Bill Buchanan

	Base-64
hello	XUFAKrxLKna5cZ2REBfFkg
Hello	ixqZU8RhEpaoJ6v4xHgE1w
Hello. How are you?	CyszE5j+Zo7bCYZtTdsFiw
Napier	j4NXH5Mkrk4j13N1MFXHtg

	Hex
hello	5D41402ABC4B2A76B9719D911017C592
Hello	8B1A9953C4611296A827ABF8C47804D7
Hello. How are you?	CC708153987BF9AD833BEBF90239BF0F
Napier	8F83571F9324AE4E23D773753055C7B6

hello	qvTGHdzF6KLavt4PO0gs2a6pQ00=
Hello	9/+ei3uy4Jtwk1pdeF4MxdnQq/A=
Hello. How are you?	Pun2Am76bnjqE5TbTWtwsqbdFC8=
Napier	v4GxNaVod2b09GR2Tqw4yopOuro=

Base-64

hello	AAF4C61DDCC5E8A2DABEDE0F3B482CD9AEA9434D
Hello	F7FF9E8B7BB2E09B70935A5D785E0CC5D9D0ABF0
Hello. How are you?	3EE876026EFA6E18EA13995B4D6B70B2A6DD142F
Napier	BF81B135A5687766F4F464764EAC38CA8A4EBABA

Hex

Message Hash

Authentication

Security and mobility are two of the most important issues on the Internet, as they will allow users to secure their data transmissions, and also break their link with physical connections.

F94FBED3DAE05D223E6B963B9076C4EC

+U++09rgXSI+a5Y7kHbE7A==

Base-64

Security and mobility are two of the mast important issues on the Internet, as they will allow users to secure their data transmissions, and also break their link their physical connections.

8A8BDC3FF80A01917D0432800201CFBF

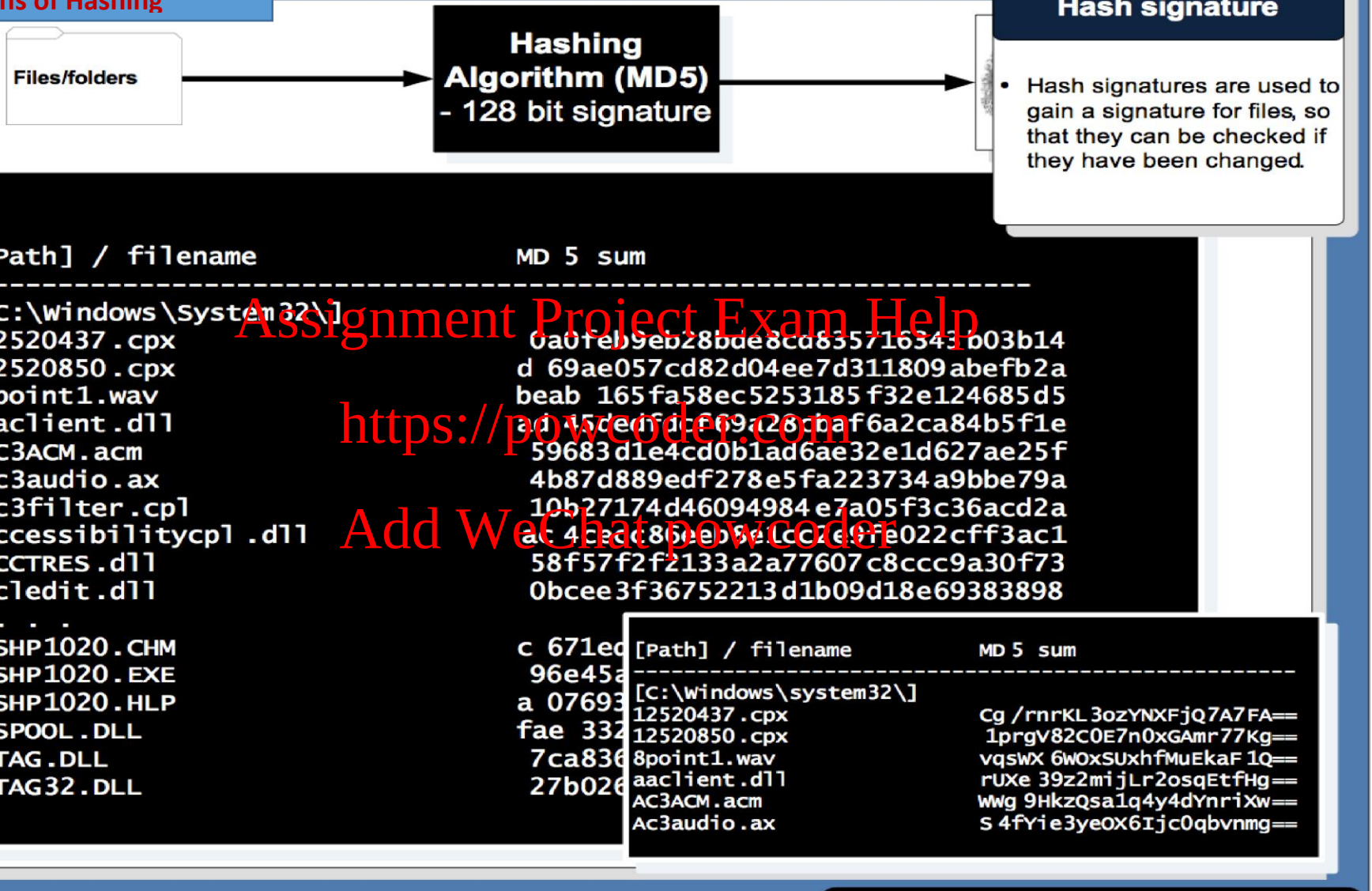
iovcP/gKAZF9BDKAAgHPvw==

Hex

Message Hash

Authentication

Applications of Hashing



MD5 hash algorithm

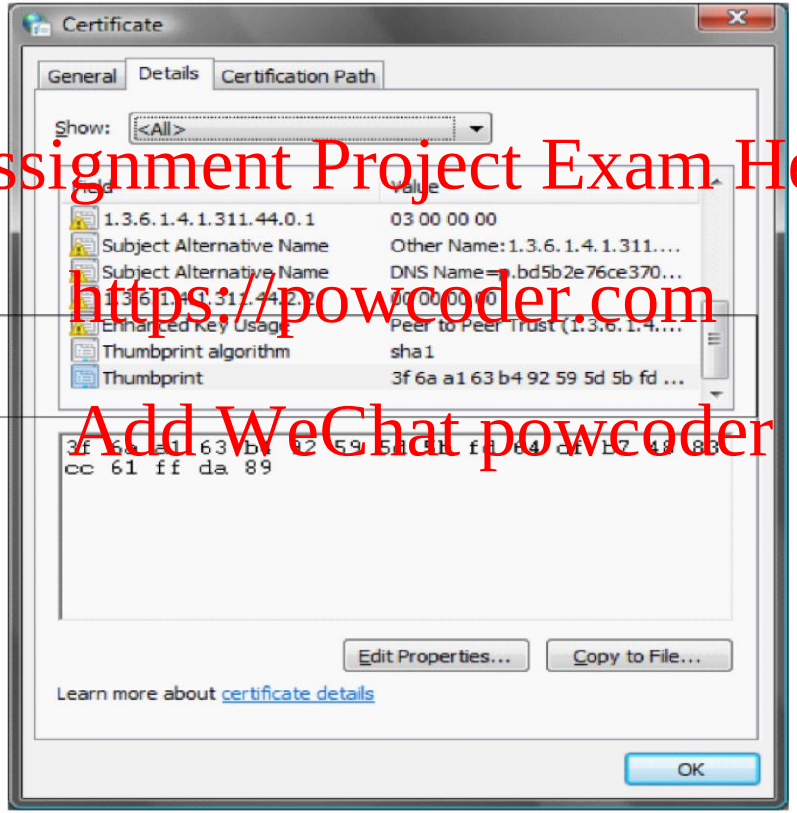
Files/folders

Hashing Algorithm (MD5) - 128 bit signature

Hash signature

- Hash signatures are used to identify that a file/certificate has not been changed.

Bob



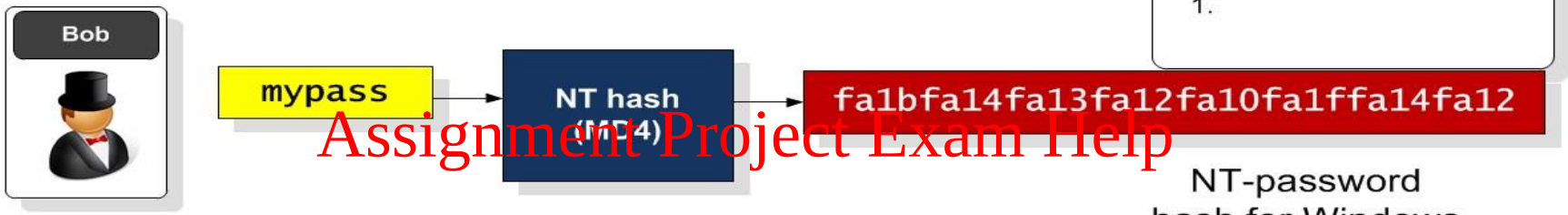
The digital certificate has an SHA-1 hash thumbprint (3f6a...89) which will be checked, and if the thumbprint is different, the certificate will be invalid.

Author: Prof Bill Buchanan

One-way hash

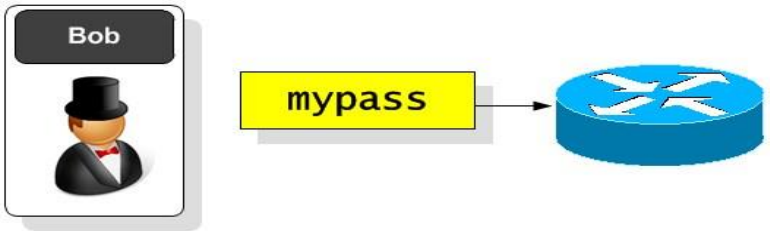
- Hashes are used for digital fingerprints (see the next unit) and for secure password storage.
- Typical methods are NT hash, MD4, MD5, and SHA-1.

Windows login/ authentication



NT-password hash for Windows NT, XP and Vista

Cisco password storage (MD5)




```
Router#config t
(config)# enable secret test

Current configuration : 542 bytes
!
version 12.1
no service single-slot-reload-enable
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router
!
enable secret 5 $1$/Nwk$knsEQYxZVenGjWOGj/TGk0
```

MD5 encoded password

One-way hash

- Hashing suffers from dictionary attacks, thus it is important that any passwords are not standard words, such as to change **password** for **pA55wOrd**.

Windows login/ authentication

mypass → NT hash (MD4) → fa1bfa14fa13fa12fa10fa1ffa14fa12

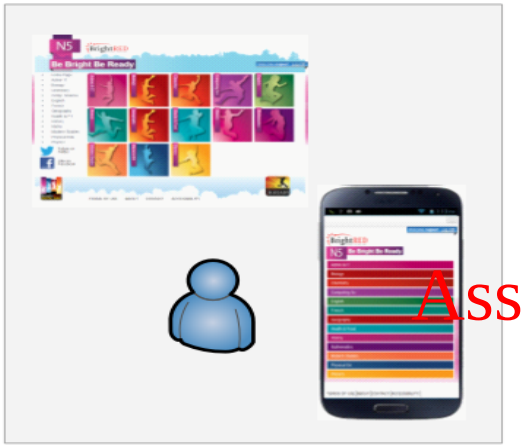
NT-password hash for Windows NT, XP and Vista

Hashing suffers from **dictionary attacks** where the signatures of well know words are stored in a table, and the intruders does a lookup on this.

mypast	effahd13fa12fa10fgffa1ffa14fa144
mypass	fa1bfa14fa13fa12fa10fa1ffa14fa12
mypose	ff12189043210954defff0123444512d

test1	aabbfce023215546dfeddd0101001cd

Risk 4: One Password Fits All



Adobe

150 million accounts compromised

#	Count	Ciphertext	Plaintext
1.	1911938	EQ7fIpT7i/Q=	123456
2.	446162	j9p+HwtWWT86aMjgZFLzYg==	123456789
3.	345834	L8qbAD3jl3jioxG6CatHBw==	password
4.	211659	BB4e6X+b2xLioxG6CatHBw==	adobe123
5.	201580	jPp+HwtWWT/ioxG6CatHBw==	12345678
6.	130838	5DjyXzQd2ws=	qwerty
7.	124253	dQi0asWPYvQ=	1234567
8.	113884	7LqYzKVeq8I=	111111
9.	83411	PMDTbP0LZxu03SwrFUvYGA==	photoshop
10.	82694	e6MPXQ5G6a8=	123123

TJ-maxx Marshalls

47 million accounts

LinkedIn

6.5 million accounts (June 2013)

Windows Live

YouTube

One account hack … leads to others

1 million accounts – in plain text. 77 million compromised

Dropbox compromised 2013

citigroup

200,000 client accounts

Adding salt

- Salt increases the range of the possible signatures

mypass → NT hash (MD4) → fa1bfa14fa13fa12fa10fa1ffa14fa12

NT-password hash for Windows NT, XP and Vista

Salt increase the range of the signatures

mypast

effahd13fa12fa10fgffa1ffa14fa144

caaahdd3fa12ccfae342345500011aff

Ddde43296945031040 3010d0ae000100

password

1fred$bATAk8UUH/IDAp9sd6IUv/

1

fred

bATAk8UUH/IDAp9sd6IUv/

password

bATAk8UUH/IDAp9sd6IUv/

fred

Author: Prof Bill Buchanan

A major factor with hash signatures is:

- **Collision.** This is where another match is found, no matter the similarity of the original message. This can be defined as a **Collision attack**.
- **Similar context**. This is where part of the message has some significance to the original, and generates the same hash signature. The can be defined as a Pre-image attack.
- **Full context**. This is where an alternative message is created with the same hash signature, and has a direct relation to the original message. This is an extension to a Pre-image attack.

In 2006 it was shown that MD5 can produce collision within less than a minute.

Note, in 2006, for SHA-1 the best time has been 18 hours

```
d131dd02c5e6eec4693d9a0698aff95c
2fcab58712467eab4004583eb8fb7f89
55ad340609f4b30283e4888325 71415a
085125e8f7cdc99fd91dbdf280373c5b
d8823e3156348f5bae6dacd436c919c6
dd53e2b487da03fd02396306d248cda0
e99f33420f577ee8ce54b67080a80d1e
c69821bcb6a8839396f9652b6ff72a70
```

```
d131dd02c5e6eec4693d9a0698aff95c
2fcab50712467eab4004583eb8fb7f89
55ad340609f4b30283e4888325f1415a
085125e8f7cdc99fd91dbd7280373c5b
d8823e3156348f5bae6dacd436c919c6
dd53e23487da03fd02396306d248cda0
e99f33420f577ee8ce54b6708280d1e
c69821bcb6a8839396f965ab6ff72a70
```

The MD5 signature gives the same result

79054025255FB1A26E4BC422AEF54EB4