

COMP47860: Ethical Computer Hacking

Lecture 10: Password Cracking

Assoc. Prof. Mark Scanlon

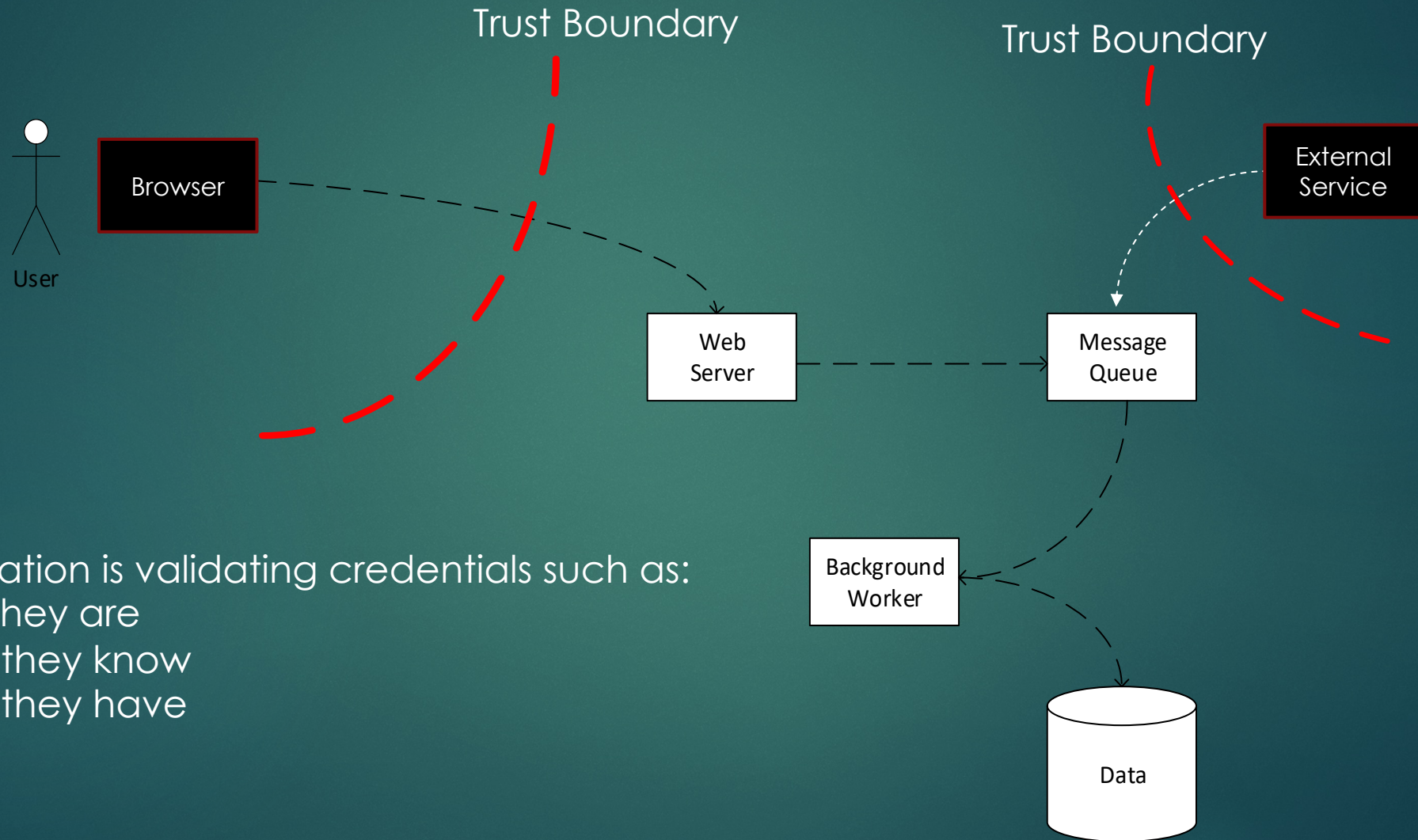
Agenda

- Authentication
- Authorisation
- Password History
- Password Usage
- Human Behaviour
- Password Cracking

Authentication

- ▶ Authentication enables organisations to secure their system by permitting only authenticated users or processes to access protected resources.
- ▶ Authentication is a process of determining:
 - ▶ Who someone is
 - ▶ What something isOR
 - ▶ Who or what they declare themselves to be

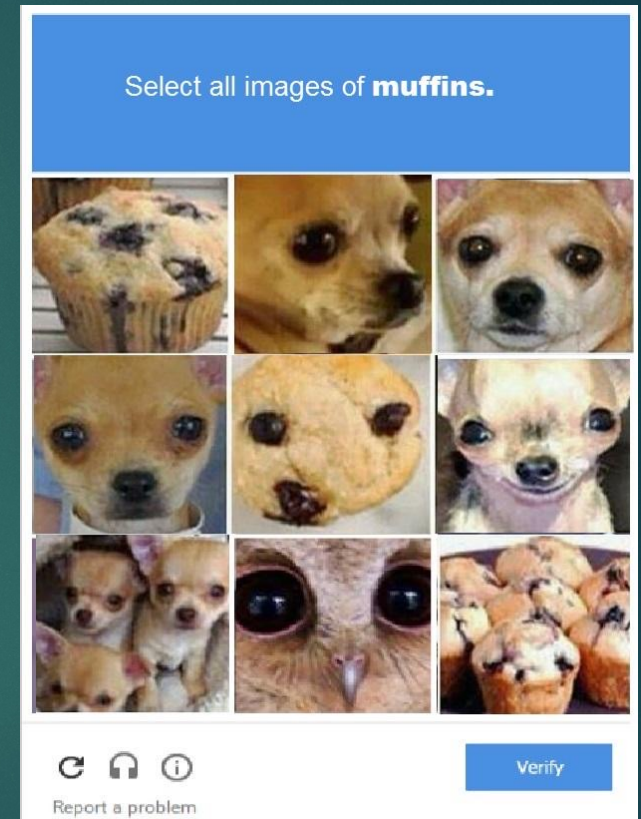
Authentication



- Authentication is validating credentials such as:
 - Who they are
 - What they know
 - What they have

Authentication

- ▶ How can you identify someone?
 - ▶ Something they know – password or pin
 - ▶ Something they have – mobile phone (one time password)
 - ▶ Something they are – Turing test

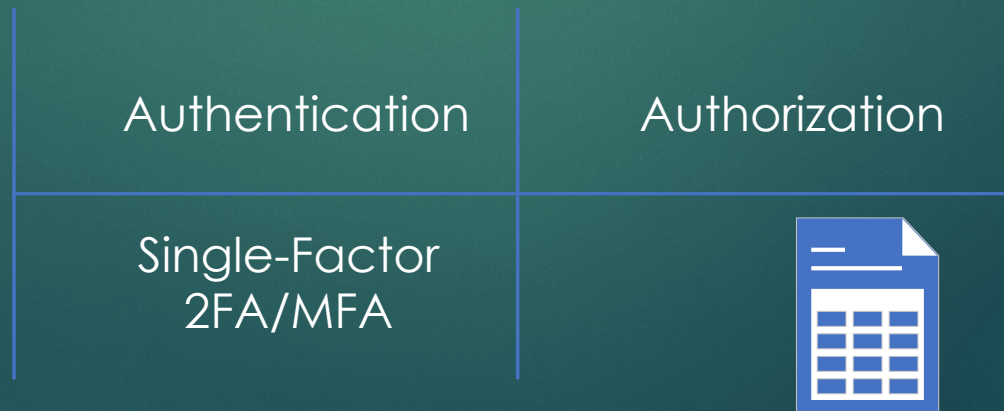


Authentication

- ▶ Single-Factor
 - ▶ Username/User ID and password to verify their identity.
- ▶ Two-Factor (2FA)
 - ▶ Using a combination of two factors
 - ▶ Something they know = PIN, password
 - ▶ Something they have = bank card, mobile phone (text)
- ▶ Multi-Factor (MFA)
 - ▶ Uses multiple factors that are independent of each other
 - ▶ Passwords
 - ▶ SMS – out of band SMS text message
 - ▶ Turing test

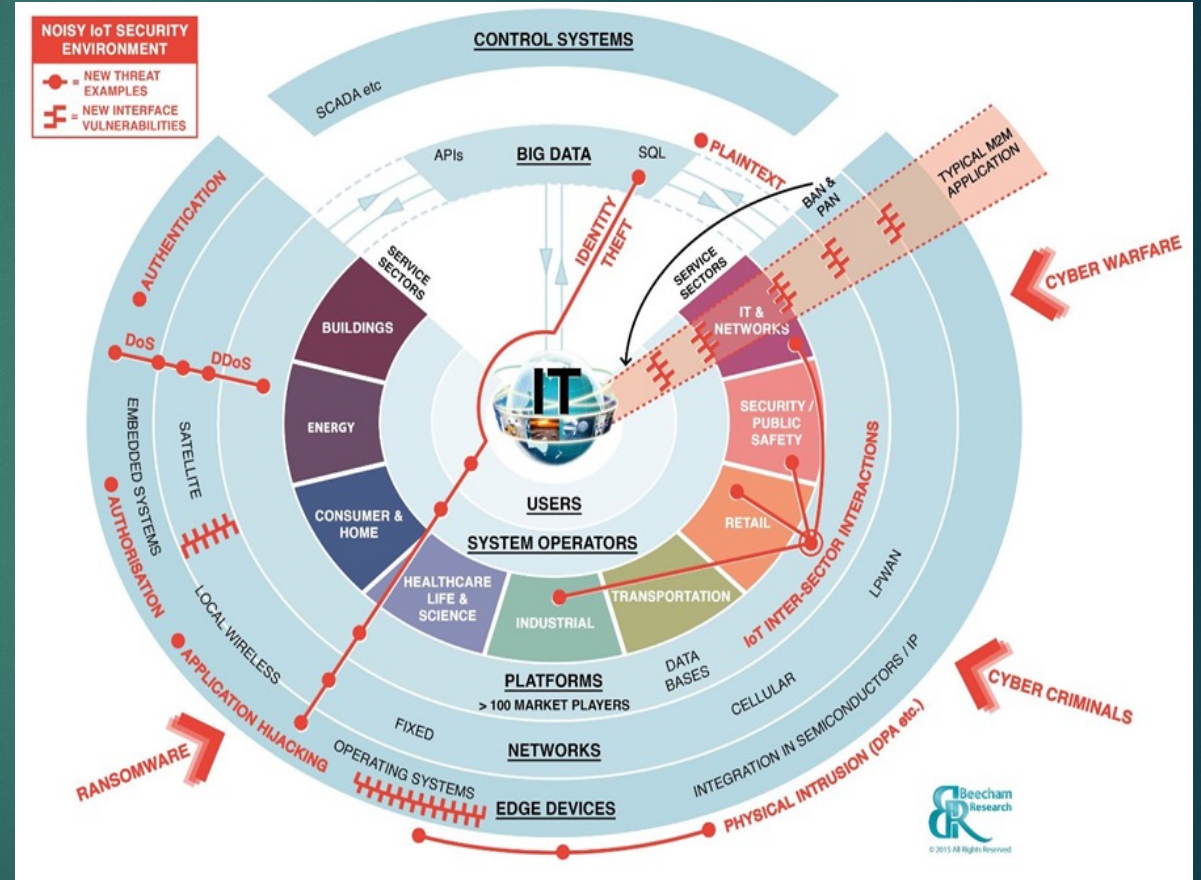
Authorisation

- ▶ Authorisation is the process of giving someone/something permission access a resource.
 - ▶ It occurs after identity has been validated
 - ▶ Verifies the rights to access resources, determined by business logic
 - ▶ Access Control List (ACL):
 - ▶ A table that tells the Operating System (OS) which access rights each user has to a particular system object.



Password Evolution

- ▶ Password mechanism used to share a mainframe in 1961
 - ▶ Used to restrict access to specific files and allotted time slots.
 - ▶ Primarily used by University professors and researchers
 - ▶ In a time before mainstream Hacking.
- ▶ Passwords have gone beyond the original researchers:
 - ▶ Used by a wide range of individuals and systems to:
 - ▶ e-commerce (banking, shopping etc.)
 - ▶ Company information and systems
 - ▶ Critical infrastructure systems, Nuclear power station etc.
 - ▶ Emails accounts (AOL, Hotmail, Gmail etc.)
 - ▶ University - student information.



Breaches

- ▶ But how safe are passwords?
 - ▶ Recent breaches include:
 - ▶ BlueKai (2020, potentially 2b recorded left unprotected)
 - ▶ Capital One (2019, 100m details stolen)
 - ▶ British Airways (2019, 380K details stolen)
 - ▶ Marriot Hotels (2018, 500m details stolen)
- ▶ Cost – runs into Billions
 - ▶ Fix the breach and cover account monitoring for the victims
 - ▶ Actual losses credit (hackers monetising the crime)

select a category below to filter

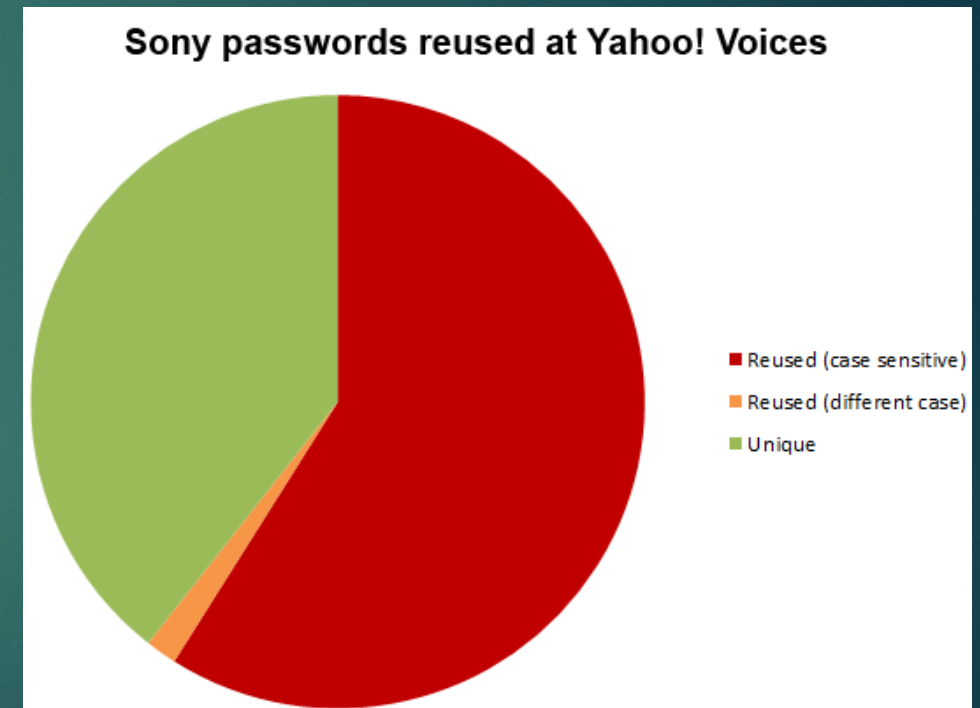
source: 20+ data breaches // data

Password Reuse

► Stats

- 1% of passwords contain non-alphanumeric character
- 4% contain two character types
- 93% are 6 to 10 characters long

- A year after the Sony breach¹:
 - “59% of people were still using the exact same password on Yahoo! Voices.”
 - A further 2% of passwords only differed by case.



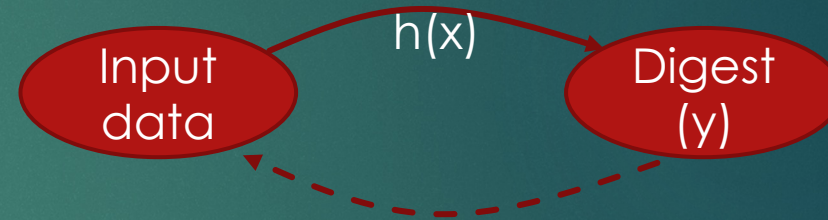
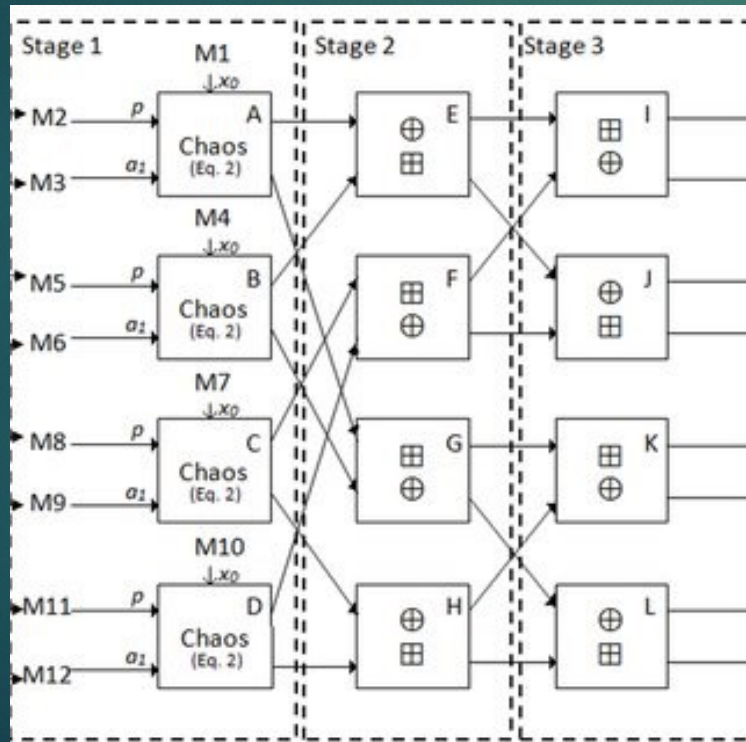
[1] <https://www.troyhunt.com/what-do-sony-and-yahoo-have-in-common/>

Password Storage: Cryptographic Hash Functions

- ▶ Hash property
 - ▶ What does hashing give us?
 - ▶ Hides the password
 - ▶ Hash functions are normally public knowledge
 - ▶ What if a hacker could determine the original password (message) from the hash (digest)?
- ▶ Properties:
 - ▶ Deterministic
 - ▶ Collision Resistance (CR)
 - ▶ Target Collision Resistance (TCR), weak collision resistance
 - ▶ Looks random
 - ▶ Non-Malleability
 - ▶ Public
 - ▶ Onewayness

Password Storage: Cryptographic Hash Functions

- Infeasible to reverse engineer x
 - Algorithm is complex and difficult to reverse



Set A (Input Data)	Set B (Hash)
Philip	da39a3ee5e6b4b0d3255bfef95601890afd80709
Philip	9ed7bcc82bc8cc78aa550908f37a532bf79112e5
Philip1	80647a60762f0671c400073af3ec8704888ca515
Philip3	4e1456cb612fa7eebaac4714668aa7209de183da

Password Salting

- ▶ Without salt, attackers can pre-compute hashes of all dictionary words once for all password entries
 - ▶ Same hash function on all UNIX machines
 - ▶ Identical passwords hash to identical values; one table of hash values can be used for all password files
- ▶ With salt, attacker must compute hashes of all dictionary words once for each password entry
 - ▶ With 12-bit random salt, same password can hash to 2^{12} different hash values
 - ▶ Attacker must try all dictionary words for each salt value in the password file

Human Behaviour

- ▶ Humans:
 - ▶ Use the same password for multiple accounts
 - ▶ Use simple easy to remember passwords
 - ▶ Fail to update the passwords
- ▶ Password resets:
 - ▶ Leak information on social media
 - ▶ Link accounts with each other
 - ▶ Poor security questions

Password Strength

► Password strength checker¹

Password:
Six-character minimum with no spaces
[Learn how to create a strong, memorable password.](#)

Password strength: Weak

Password:
Six-character minimum with no spaces
[Learn how to create a strong, memorable password.](#)

Password strength: Medium

Password:
Six-character minimum with no spaces
[Learn how to create a strong, memorable password.](#)

Password strength: Strong

Weak password pattern	Is it memorable?	Time to crack
A common word (example: december)	Yes	18 milliseconds
An easily-typed spatial word (example: qwerty)	Yes	10 milliseconds
The family dog (example: rex)	Yes	27 milliseconds
An important number, such as DOB, Wedding (example: 03261981)	Memorable to the user	2.213 seconds
A word with trivial letter→number substitutions (example: pa55w0rd)	Sort of memorable, but you may forget which letters are substituted for numbers.	639 milliseconds

[1] <http://designinginterfaces.com/patterns/password-strength-meter/>

Password Attacks

- ▶ Dictionary attack
 - ▶ List for words and word-pattern: Example: password, pas55w0rd etc.
- ▶ Brute force attack
 - ▶ Try everything, Fuzz the pattern
- ▶ Reverse Lookup tables attack
- ▶ Social Engineering/Phishing
- ▶ Malware (key loggers, memory scrappers)
- ▶ Offline cracking
- ▶ Spidering

Password Cracking: Reverse Lookup Tables

HashKiller

Hash Cracker ▾

List Manager ▾

Tools ▾

Downloads ▾

Hashcat GUI

Discord

Forums

What is HashKiller?

HashKiller's purpose is to serve as a meeting place for computer hobbyists, security researchers and penetration testers. It serves as a central location to promote greater security on the internet by demonstrating the weakness of using weak hash based storage / authentication.
HashKiller.co.uk is a hash lookup service. This allows you to input a hash and search for its corresponding plaintext ("found") in our database of already-cracked hashes.
In other words, we are not cracking your hash in realtime - we're just caching the hard work of many cracking enthusiasts over the years.

Need a hash cracking?

Crack Some Hashes

Note that we do **not** use terms like "decrypted", "dehashed", or "reversed" - hashes can only be looked up quickly *after they've been cracked the hard way.*

Last 50 successful hash cracks / founds

#	Hash Type	Hash / Salt	Password	Cracked By	Date
1	SHA1	4097a6f4b6e1ed76b845adec3fe5a9ae4622c5a9	dandym123	blandyuk	15-Aug-2019 13:52:53
2	SHA1	718e7140aee18c330c5d176eb0239e398ae120fd	thiagow40	gearjunkie	15-Aug-2019 13:52:52
3	SHA1	5ff18ddde7532a718f0170cc683acd6630734fc8	clau0800	blandyuk	15-Aug-2019 13:52:51
4	SHA1	bbf5c8eaa2bf0fcacd30a392ff4154c3d50162bb	pit32216814	blandyuk	15-Aug-2019 13:52:50
5	SHA1	89846e225e2443cc9dd4a2bfaaaa902409298942	83658367	blandyuk	15-Aug-2019 13:52:49
6	SHA1	6ef7007fb736f6e651112f1c07fdb54a5d15ea2	gregory157946821365	gearjunkie	15-Aug-2019 13:52:48
7	MD5	c63271d6b2f678cb09e84c092971077b	kozchulebg	vetronexe	15-Aug-2019 13:52:48
8	SHA1	2af134e1da00d35b914ba3c56fd513145fe9c476	220215du	gearjunkie	15-Aug-2019 13:52:47
9	SHA1	c65ee92e4edac04fbec3db1037c31c69c906bb96	berlinberlin123	gearjunkie	15-Aug-2019 13:52:46
10	SHA1	2af134e1da00d35b914ba3c56fd513145fe9c476	220215du	gearjunkie	15-Aug-2019 13:52:45
11	SHA1	55896d28cb472e6f6b1ee8cf7eb466928527ed1a	11121314mae	blandyuk	15-Aug-2019 13:52:44
12	SHA1	fde8d8009eb9209f0db54807c876751924fa13d2	ekinho3		15-Aug-2019 13:52:43
13	SHA1	55896d28cb472e6f6b1ee8cf7eb466928527ed1a	11121314mae	blandyuk	15-Aug-2019 13:52:41
14	SHA1	18ef4866c50f105b7ab0a24dd8edf3cc693c0824	pedroolavo1	gearjunkie	15-Aug-2019 13:52:40
15	MySQL4.1/MySQL5	d696d2ea474c98f6ade698f07cf9df18e0c987a4	karakara23	cvsi	15-Aug-2019 13:52:39
16	SHA1	e1bdfa8db292acc85552625b7bfc00315c1cf6f4	42754275	blandyuk	15-Aug-2019 13:52:39
17	SHA1	44e362957b565d8992246c390c10195df099e2ec	testoland	blandyuk	15-Aug-2019 13:52:38
18	SHA1	086fc153ac2a532a8246f6dbfac8a7781fc59556	gwn8cdty	blandyuk	15-Aug-2019 13:52:37
19	SHA1	e9883145dce8b41d02bcd49c39f520b89c8acaae	102769		15-Aug-2019 13:52:36

Password Cracking: Reverse Lookup Tables

[HashKiller](#)[Hash Cracker ▾](#)[List Manager ▾](#)[Tools ▾](#)[Downloads ▾](#)[Hashcat GUI](#)[Discord](#)[Forums](#)

Please list your hashes below ...

Please input the hash hashes that you would like to look up. NOTE that the space character is replaced with **[space]**.

Your Hashes:

```
7c91ec228f38e3cdd81f782765bb6b124850bc7f
```

Crack my Hashes

Upload button disabled? We use Google reCAPTCHA v3.

Cracker Results:

```
7c91ec228f38e3cdd81f782765bb6b124850bc7f SHA1 myPa55word
```

HashKiller.co.uk is a hash lookup service. This allows you to input an hash and search for its corresponding plaintext ("found") in our database of all cracked hashes.

It's like having your own massive password-cracking cluster - but with immediate results!

We have been building our hash database since August 2007.

Note that we do **not** use terms like "decrypted", "dehashed", or "reversed" as hashes can only be looked up quickly *after they've been cracked the hard way*.

In other words, we are not cracking your hash in realtime - we're just caching the hard work of many cracking enthusiasts over the years.

Output Formats :

- Found : `$hash[:$salt] $type $pass`
- Not Found : `$hash[:$salt] [No Match]`
- Invalid : `$hash [Invalid]`

Password Cracking Tools

- ▶ A password cracking program can be used to detect weak passwords amongst the system so they can be changed
- ▶ Popular programs for password cracking:
 - LC4
 - Sam Inside
 - Crack
 - John the Ripper (JTR)
- ▶ John the Ripper is a fast password cracker, currently available for many flavors of Unix, macOS, and Windows.



Password Best Practices

- ▶ Make sure your passwords are long and strong
- ▶ Don't reuse passwords
- ▶ Enable multi-factor authentication when it's an option
- ▶ Consider using a password manager
- ▶ Change passwords quickly if there is a breach

Summary

- ▶ Authentication
- ▶ Authorisation
- ▶ Password History
- ▶ Password Usage
- ▶ Human Behaviour
- ▶ Password Cracking