Techniques Criminals Use to Break Authentication and How to Defend Against These Attacks

Rochester Security Summit 2018

Agile Security – Adapting to Change

Rochester Riverside Convention Center
Rochester, NY
October 9 & 10, 2018

Danny Harris
Senior Security Consultant & Instructor
dharris@securityinnovation.com

SECURITY INNOVATION

Objectives

- To understand the common attacks against passwords and authentication services
- To become familiar with some password and authentication-related security patterns to better design and protect applications against criminal attackers
- To understand that “Defense in Depth” is necessary to protect systems from authentication attacks

Agenda

- Identification and Authentication
- Common Password Attacks
- Secure Authentication Practices

Identification

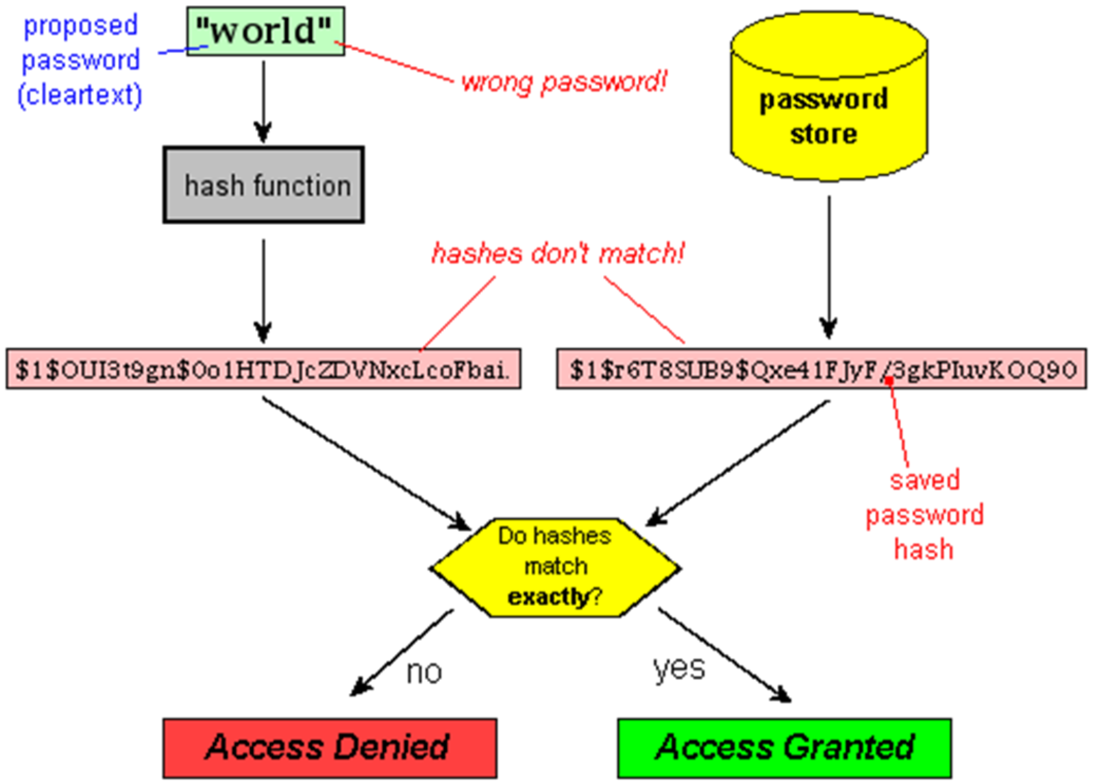
- **Identification** is the process of recognizing who the user/entity/system is
- The user/entity/system presents some kind of identifier to the system
  - For example, a UserID or name by which the user is known to the system
- Identifiers are used:
  - For billing and accounting
  - For authorization to access different components of the system
  - To detect misuse and abuse of system resources

Authentication

- **Authentication** is the process of proving to the system who you claim to be
  - The password “proves” who you are to the system
- Authentication is required when access to data and resources must be controlled and audited
- Authentication is done through a combination of one or more of the following:
  - Proof by knowledge – something you know (typically a password)
    - This will be the primary focus of this presentation
  - Proof by possession – something you have
  - Proof by property(ies) of the individual – something you are
  - Proof by location – where you are located

How Password Authentication Works

- Using passwords to gain access to computer systems requires two steps:
  - Determining who you are (identification)
  - Proving who you are (authentication)
- The system has a list of UserIDs and passwords in some form
- The system compares the user-entered UserID and password (or password hash) against what is already stored in the system
  - If a valid UserID and the corresponding password/hash match what the computer has stored, the user is given access
    - Otherwise, no access is granted



From: http://unixwiz.net/techtips/iguide-crypto-hashes.html

SECURITY INNOVATION

Agenda

- Identification and Authentication
- Common Password Attacks
- Secure Authentication Practices

Attacker Motives

- Credential Stuffing Attacks
  - Automated tools use databases of leaked/stolen credentials to try to login to various websites to fraudulently gain access to user accounts
  - Attackers will try those credentials on other sites because people tend to use the same credentials on more than one site

> “I use the same password for every site”
>
> https://www.washingtonpost.com/outlook/2018/09/04/im-teaching-email-security-democratic-campaigns-its-bad/

What Can An Attacker Do With Account Credentials?

- Criminals are motivated to steal authentication credentials to impersonate legitimate users or take over accounts in order to steal money, goods, or services, or cause other harm
  - Masquerade as the victim
  - Hijack/takeover accounts
  - Replay attacks
  - Extortion
- Cyber-predators acquire legitimate user credentials to exploit trusted network relationships in order to expand unauthorized access, maintain persistence, and exfiltrate data from targeted organizations under the guise of authorized activity

Sample Extortion Email with Password

Subject: (username + password)

It seems that, (**password**), is your password. You may not know me and you are probably wondering why you are getting this e mail, right?

actually, I setup a malware on the adult vids (porno) web-site and guess what, you visited this site to have fun (you know what I mean). While you were watching videos, your internet browser started out functioning as a RDP (Remote Desktop) having a keylogger which gave me accessibility to your screen and web cam. after that, my software program obtained all of your contacts from your Messenger, FB, as well as email.

What did I do?

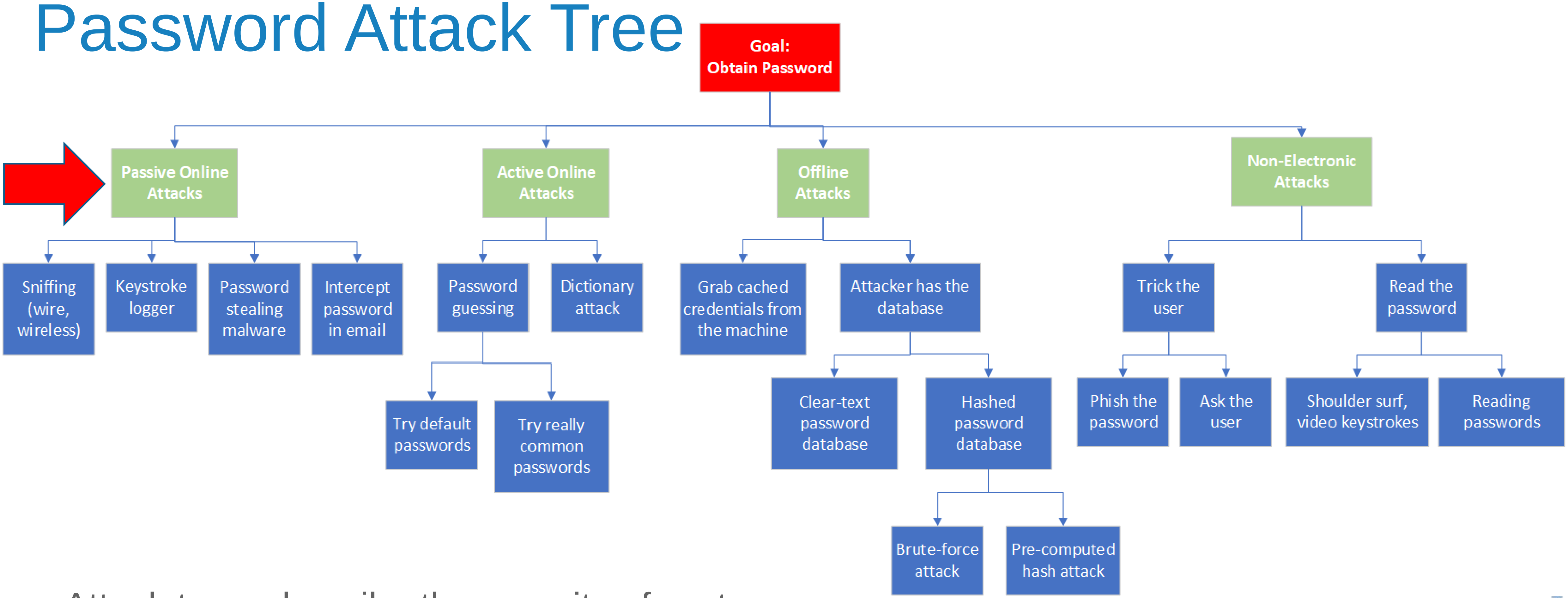
I created a double-screen video. 1st part shows the video you were watching (you've got a good taste haha . . .), and 2nd part shows the recording of your web cam.

exactly what should you do?

Well, in my opinion, $2900 is a fair price for our little secret.

https://www.bleepingcomputer.com/news/security/beware-of-extortion-scams-stating-they-have-video-of-you-on-adult-sites/

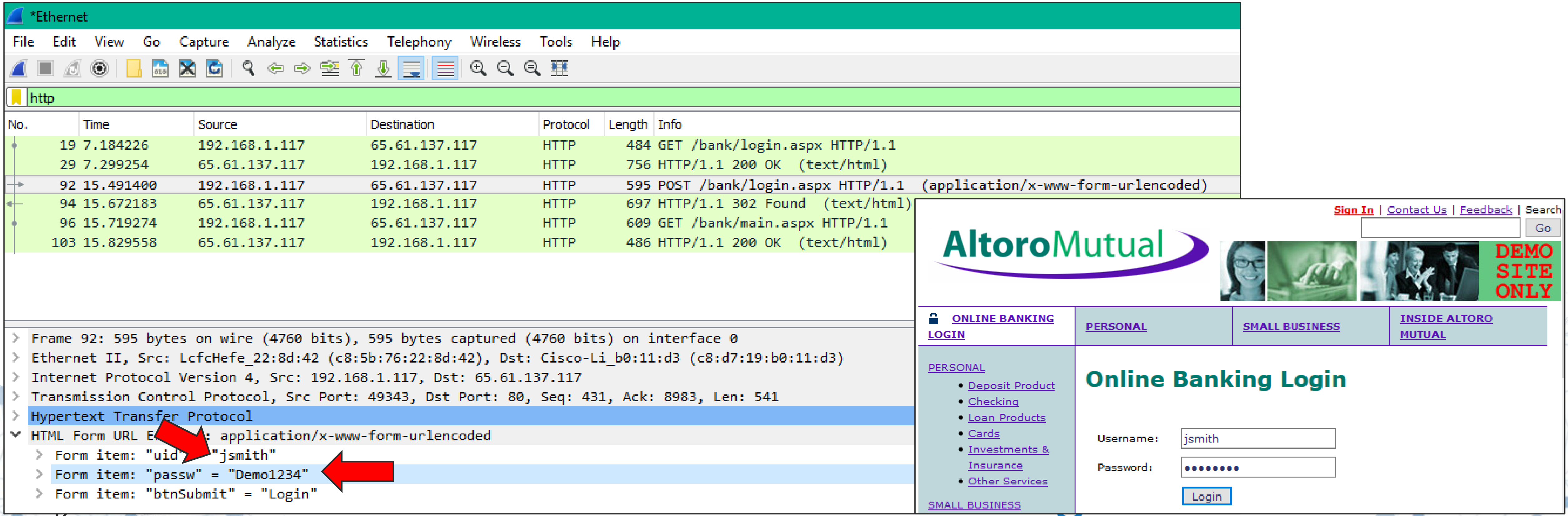
Password Attack Tree



- Attack trees describe the security of systems
  - Attacks are shown in a tree structure, with the goal as the root node and different ways of achieving that goal as leaf nodes
- We will discuss these common password attacks

Passive Online Attack: Sniffing

- Attacker uses a sniffer to capture and record network traffic
  - Physical access to the network is required if the credentials are pulled from the “wire”
  - No physical access is required to “sniff” credentials from a wireless network
- Specialized password sniffing tools ignore everything except for the authentication credentials – making sniffing credentials very easy on unencrypted networks



*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

http

```
No.   Time       Source          Destination     Protocol Length Info
   19 7.184226   192.168.1.117   65.61.137.117   HTTP      484 GET /bank/login.aspx HTTP/1.1
   29 7.299254   65.61.137.117   192.168.1.117   HTTP      756 HTTP/1.1 200 OK  (text/html)
   92 15.491400  192.168.1.117   65.61.137.117   HTTP      595 POST /bank/login.aspx HTTP/1.1  (application/x-www-form-urlencoded)
   94 15.672183  65.61.137.117   192.168.1.117   HTTP      697 HTTP/1.1 302 Found  (text/html)
   96 15.719274  192.168.1.117   65.61.137.117   HTTP      609 GET /bank/main.aspx HTTP/1.1
  103 15.829558  65.61.137.117   192.168.1.117   HTTP      486 HTTP/1.1 200 OK  (text/html)
```

```
> Frame 92: 595 bytes on wire (4760 bits), 595 bytes captured (4760 bits) on interface 0
> Ethernet II, Src: LcfcHefe_22:8d:42 (c8:5b:76:22:8d:42), Dst: Cisco-Li_b0:11:d3 (c8:d7:19:b0:11:d3)
> Internet Protocol Version 4, Src: 192.168.1.117, Dst: 65.61.137.117
> Transmission Control Protocol, Src Port: 49343, Dst Port: 80, Seq: 431, Ack: 8983, Len: 541
> Hypertext Transfer Protocol
v HTML Form URL E[illegible]: application/x-www-form-urlencoded
   > Form item: "uid" = "jsmith"
   > Form item: "passw" = "Demo1234"
   > Form item: "btnSubmit" = "Login"
```

Online Banking Login

Username: jsmith

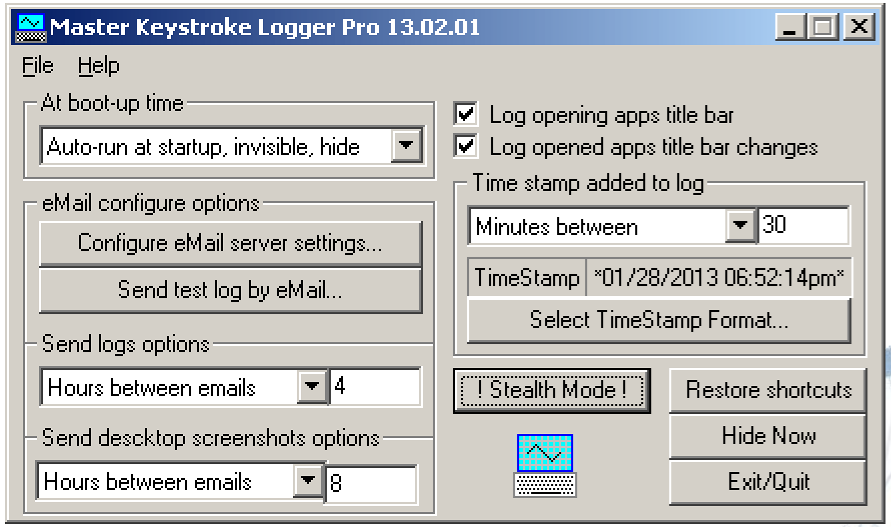
Password: ••••••••

Login

Passive Online Attack: Keystroke Loggers

- Attacker uses a keystroke logger to capture and record passwords
  - **Hardware keystroke loggers** require the attacker have physical access to the computer to install and retrieve the logger hardware
  - **Software keystroke loggers** can be installed remotely (drive-by download)
    - Captured keystrokes (including passwords) are sent to the attacker

http://www.keelog.com/



Master Keystroke Logger Pro 13.02.01

At boot-up time: Auto-run at startup, invisible, hide

☑ Log opening apps title bar

☑ Log opened apps title bar changes

Time stamp added to log: Minutes between 30

TimeStamp "01/28/2013 06:52:14pm"

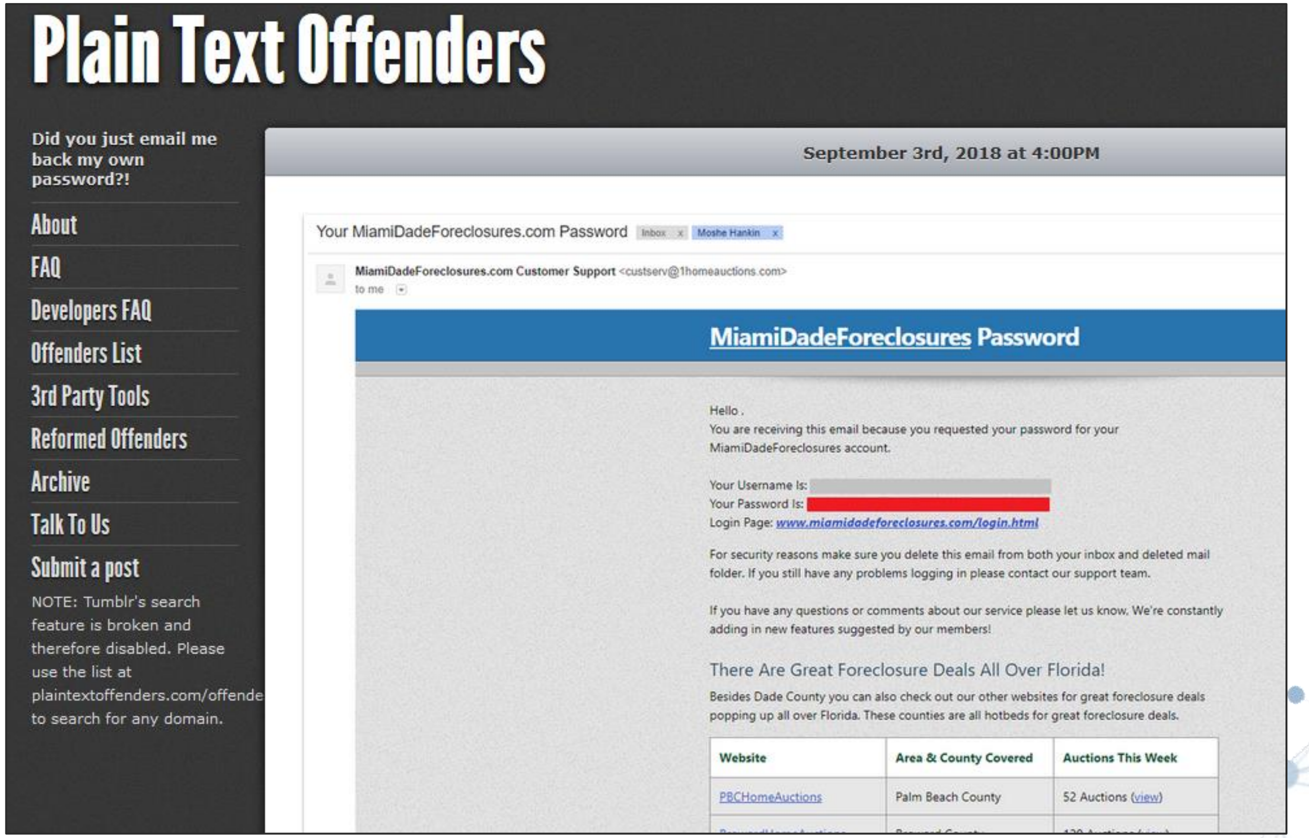
Send logs options: Hours between emails 4

Send descktop screenshots options: Hours between emails 8

File
Help
eMail configure options
Configure eMail server settings...
Send test log by eMail...
Select TimeStamp Format...
! Stealth Mode !
Restore shortcuts
Hide Now
Exit/Quit

Passive Online Attack: Intercept Passwords in Email

- Email is still very commonly used to send clear-text passwords
- Unencrypted mail is an extremely insecure channel
  - Mail often persists in storage
  - It may be accessible by system admins
  - It is readily forwarded and redistributed
- A site that emails a password means:
  - The password is stored as plaintext or is reversibly encrypted
    - The password is not being stored as a secure hash
  - The password is sent via an insecure channel



Plain Text Offenders

Did you just email me back my own password?!

- About
- FAQ
- Developers FAQ
- Offenders List
- 3rd Party Tools
- Reformed Offenders
- Archive
- Talk To Us
- Submit a post

NOTE: Tumblr's search feature is broken and therefore disabled. Please use the list at plaintextoffenders.com/offende to search for any domain.

September 3rd, 2018 at 4:00PM

Your MiamiDadeForeclosures.com Password

MiamiDadeForeclosures.com Customer Support <custserv@1homeauctions.com>
to me

MiamiDadeForeclosures Password

Hello .
You are receiving this email because you requested your password for your MiamiDadeForeclosures account.

Your Username Is: [redacted]
Your Password Is: [redacted]
Login Page: *www.miamidadeforeclosures.com/login.html*

For security reasons make sure you delete this email from both your inbox and deleted mail folder. If you still have any problems logging in please contact our support team.

If you have any questions or comments about our service please let us know. We're constantly adding in new features suggested by our members!

There Are Great Foreclosure Deals All Over Florida!

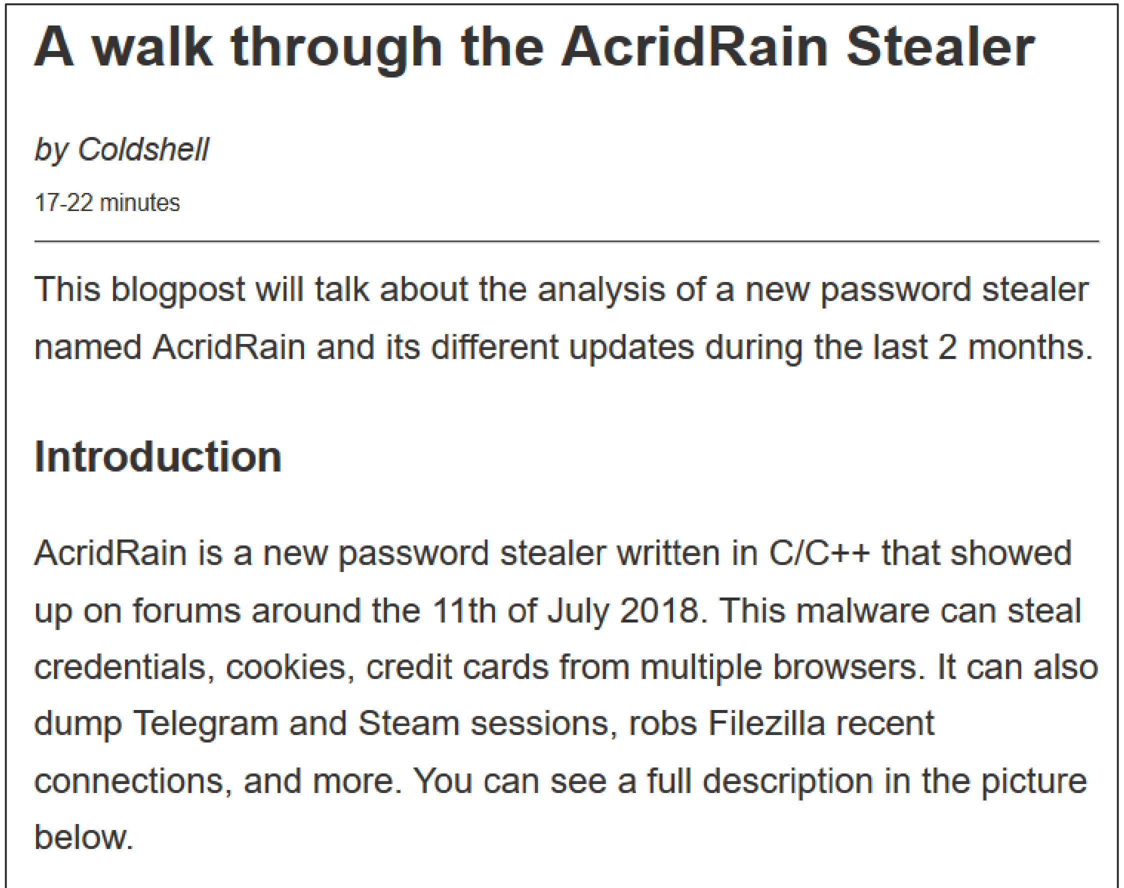
Besides Dade County you can also check out our other websites for great foreclosure deals popping up all over Florida. These counties are all hotbeds for great foreclosure deals.

Website	Area & County Covered	Auctions This Week
PBCHomeAuctions	Palm Beach County	52 Auctions (view)

http://plaintextoffenders.com/

Passive Online Attack: Password Stealer Malware

- Malware is used to steal passwords (and other data)
- Writing and selling password stealing software is a business for cyber criminals



A walk through the AcridRain Stealer

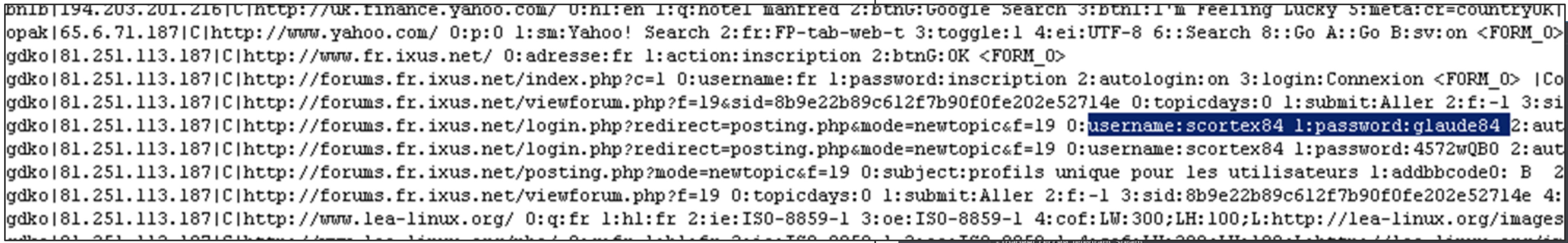
by Coldshell

17-22 minutes

This blogpost will talk about the analysis of a new password stealer named AcridRain and its different updates during the last 2 months.

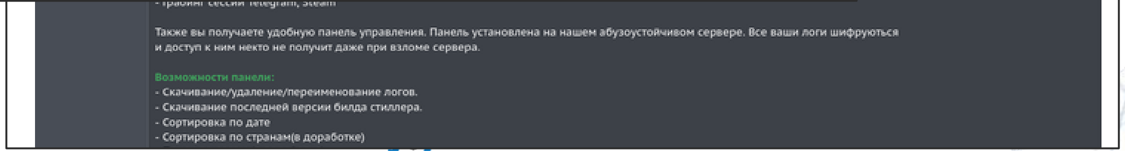
Introduction

AcridRain is a new password stealer written in C/C++ that showed up on forums around the 11th of July 2018. This malware can steal credentials, cookies, credit cards from multiple browsers. It can also dump Telegram and Steam sessions, robs Filezilla recent connections, and more. You can see a full description in the picture below.



```
bnlb|194.203.201.216|C|http://uk.finance.yahoo.com/ 0:hl:en 1:q:hotel manfred 2:btnG:Google Search 3:btnI:I'm Feeling Lucky 5:meta:cr=countryUK|
opak|65.6.71.187|C|http://www.yahoo.com/ 0:p:0 1:sm:Yahoo! Search 2:fr:FP-tab-web-t 3:toggle:1 4:ei:UTF-8 6::Search 8::Go A::Go B:sv:on <FORM_0>
gdko|81.251.113.187|C|http://www.fr.ixus.net/ 0:adresse:fr 1:action:inscription 2:btnG:OK <FORM_0>
gdko|81.251.113.187|C|http://forums.fr.ixus.net/index.php?c=1 0:username:fr 1:password:inscription 2:autologin:on 3:login:Connexion <FORM_0> |Co
gdko|81.251.113.187|C|http://forums.fr.ixus.net/viewforum.php?f=19&sid=8b9e22b89c612f7b90f0fe202e52714e 0:topicdays:0 1:submit:Aller 2:f:-1 3:si
gdko|81.251.113.187|C|http://forums.fr.ixus.net/login.php?redirect=posting.php&mode=newtopic&f=19 0:username:scortex84 1:password:glaude84 2:aut
gdko|81.251.113.187|C|http://forums.fr.ixus.net/login.php?redirect=posting.php&mode=newtopic&f=19 0:username:scortex84 1:password:4572wQB0 2:aut
gdko|81.251.113.187|C|http://forums.fr.ixus.net/posting.php?mode=newtopic&f=19 0:subject:profils unique pour les utilisateurs 1:addbbcode0: B  2
gdko|81.251.113.187|C|http://forums.fr.ixus.net/viewforum.php?f=19 0:topicdays:0 1:submit:Aller 2:f:-1 3:sid:8b9e22b89c612f7b90f0fe202e52714e 4:
gdko|81.251.113.187|C|http://www.lea-linux.org/ 0:q:fr 1:hl:fr 2:ie:ISO-8859-1 3:oe:ISO-8859-1 4:cof:LW:300;LH:100;L:http://lea-linux.org/images
```

Data from password stealer (2004)



- Грабинг сессии Telegram, Steam

Также вы получаете удобную панель управления. Панель установлена на нашем абузоустойчивом сервере. Все ваши логи шифруються и доступ к ним некто не получит даже при взломе сервера.

Возможности панели:
- Скачивание/удаление/переименование логов.
- Скачивание последней версии билда стиллера.
- Сортировка по дате
- Сортировка по странам(в доработке)

https://thisissecurity.stormshield.com/2018/08/28/acridrain-stealer/

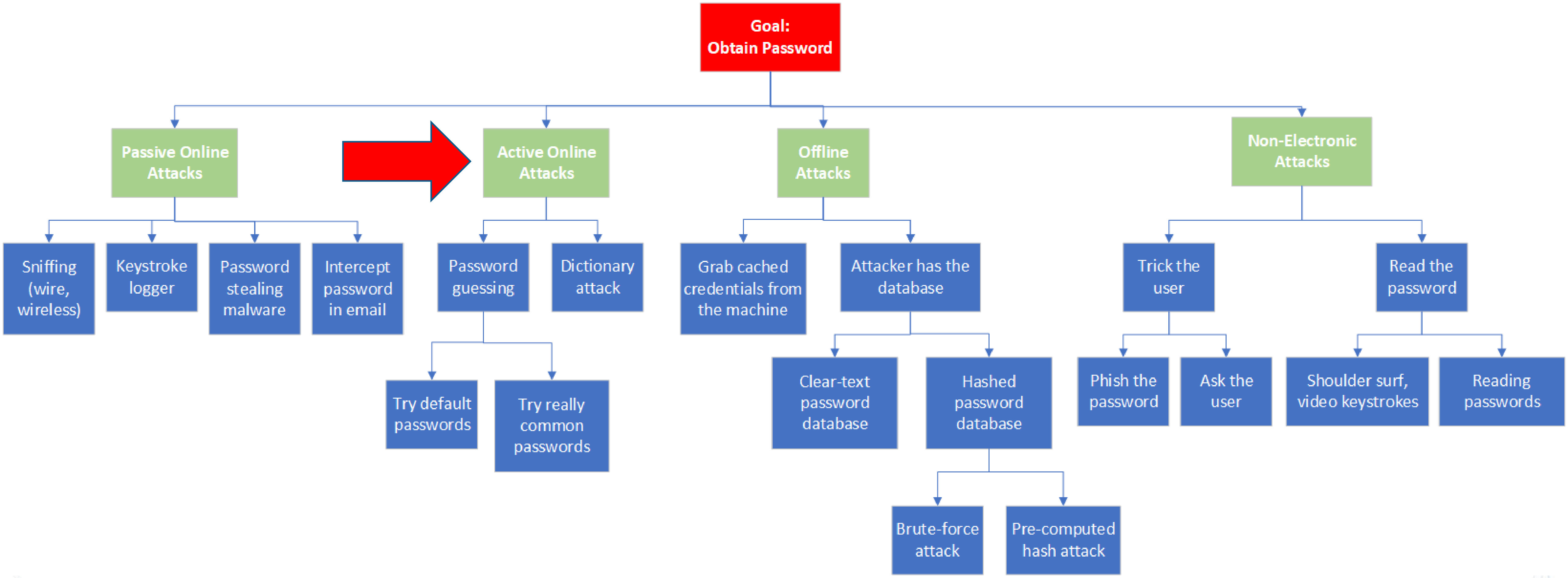
Passive Online Attacks
Attack Success Factors

- Unencrypted network communications enables sniffing attacks
- Visiting a malicious website or clicking an untrusted link can result in a keystroke logger or other malware automatically installing on the victim’s machine
- An attacker with physical access to a machine can install a hardware keystroke logger
- Sending unencrypted email with passwords is just so easy

Passive Online Attacks Countermeasures

- Encrypt all network communications to prevent sniffing attacks
- Be exceedingly careful what software is installed and running on your system
  - Use current anti-malware software
  - Be very concerned about “drive-by-downloads” installing keystroke loggers
- Restrict physical access to systems
- Don’t send unencrypted email with passwords

Password Attack Tree



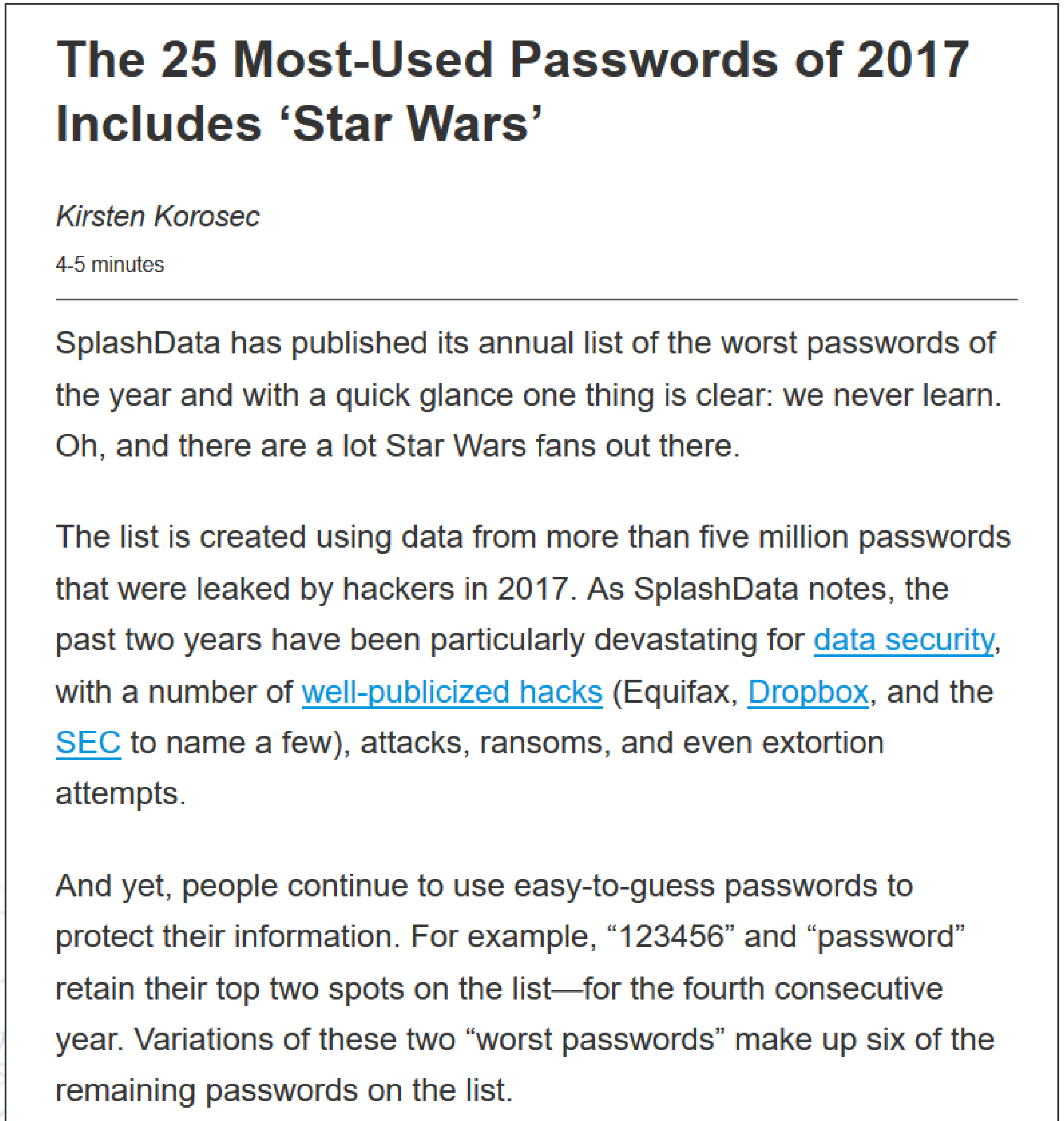
Active Online Attacks: Password Guessing

- Guess various passwords

```
FOR a given account_name
    TRY logging in with a **guessed** password
        IF success
        THEN impersonate the victim
        ELSE try again until success, too tired, or the account locks
```

- Many systems come with standard, default passwords pre-installed for convenience that may not be changed before moving a system to production
  - Examples of default passwords: http://www.defaultpassword.com/
- Attackers will often try very common passwords
- If done at the keyboard, it is very time consuming
  - Automated tools can speed things up dramatically

Common/Easily Guessed Passwords



The 25 Most-Used Passwords of 2017 Includes ‘Star Wars’

Kirsten Korosec

4-5 minutes

SplashData has published its annual list of the worst passwords of the year and with a quick glance one thing is clear: we never learn. Oh, and there are a lot Star Wars fans out there.

The list is created using data from more than five million passwords that were leaked by hackers in 2017. As SplashData notes, the past two years have been particularly devastating for data security, with a number of well-publicized hacks (Equifax, Dropbox, and the SEC to name a few), attacks, ransoms, and even extortion attempts.

And yet, people continue to use easy-to-guess passwords to protect their information. For example, “123456” and “password” retain their top two spots on the list—for the fourth consecutive year. Variations of these two “worst passwords” make up six of the remaining passwords on the list.

1. **123456** (Unchanged)
2. **Password** (Unchanged)
3. **12345678** (Up 1)
4. **qwerty** (Up 2)
5. **12345** (Down 2)
6. **123456789** (New)
7. **letmein** (New)
8. **1234567** (Unchanged)
9. **football** (Down 4)
10. **iloveyou** (New)
11. **admin** (Up 4)
12. **welcome** (Unchanged)
13. **monkey** (New)
14. **login** (Down 3)
15. **abc123** (Down 1)
16. **starwars** (New)
17. **123123** (New)
18. **dragon** (Up 1)
19. **passw0rd** (Down 1)
20. **master** (Up 1)
21. **hello** (New)
22. **freedom** (New)
23. **whatever** (New)
24. **qazwsx** (New)
25. **trustno1** (New)

http://fortune.com/2017/12/19/the-25-most-used-hackable-passwords-2017-star-wars-freedom/

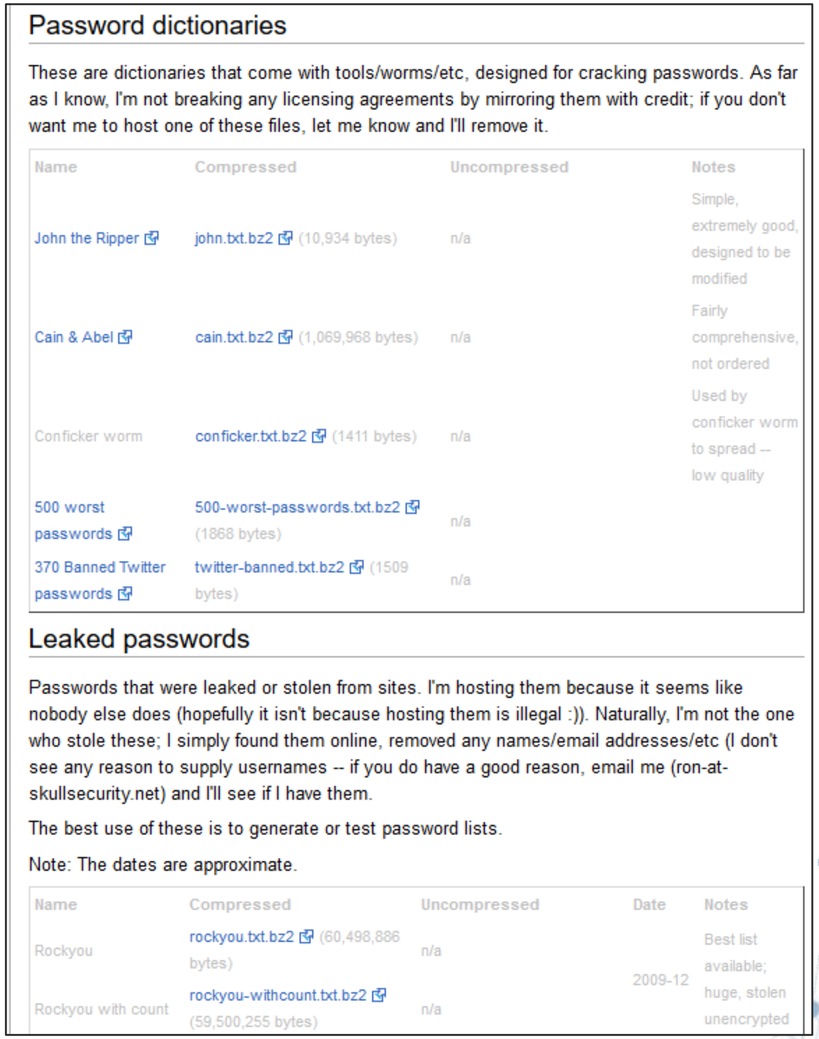
SECURITY INNOVATION

Active Online Attacks:
Dictionary Attack with Clear Text Passwords

- The attacker successively tries all the words in an exhaustive list called a **dictionary**
  - Hacker wordlists cover many topics in many languages
  - To speed up the attack, they try the most likely possibilities
- The attacker keeps trying dictionary words as passwords until:
  - The password works
  - The account gets locked
  - Attacker is tired or too much time has been expended

```
FOR a given account_name
    TRY logging in with a dictionary word
        IF success
        THEN impersonate the victim
        ELSE try again until success, too tired, or the account locks
```

- Tools can send authentication requests to the application very quickly



Password dictionaries

These are dictionaries that come with tools/worms/etc, designed for cracking passwords. As far as I know, I'm not breaking any licensing agreements by mirroring them with credit; if you don't want me to host one of these files, let me know and I'll remove it.

Name	Compressed	Uncompressed	Notes
John the Ripper	john.txt.bz2 (10,934 bytes)	n/a	Simple, extremely good, designed to be modified
Cain & Abel	cain.txt.bz2 (1,069,968 bytes)	n/a	Fairly comprehensive, not ordered
Conficker worm	conficker.txt.bz2 (1411 bytes)	n/a	Used by conficker worm to spread -- low quality
500 worst passwords	500-worst-passwords.txt.bz2 (1868 bytes)	n/a	
370 Banned Twitter passwords	twitter-banned.txt.bz2 (1509 bytes)	n/a	

Leaked passwords

Passwords that were leaked or stolen from sites. I'm hosting them because it seems like nobody else does (hopefully it isn't because hosting them is illegal :)). Naturally, I'm not the one who stole these; I simply found them online, removed any names/email addresses/etc (I don't see any reason to supply usernames -- if you do have a good reason, email me (ron-at-skullsecurity.net) and I'll see if I have them.

The best use of these is to generate or test password lists.

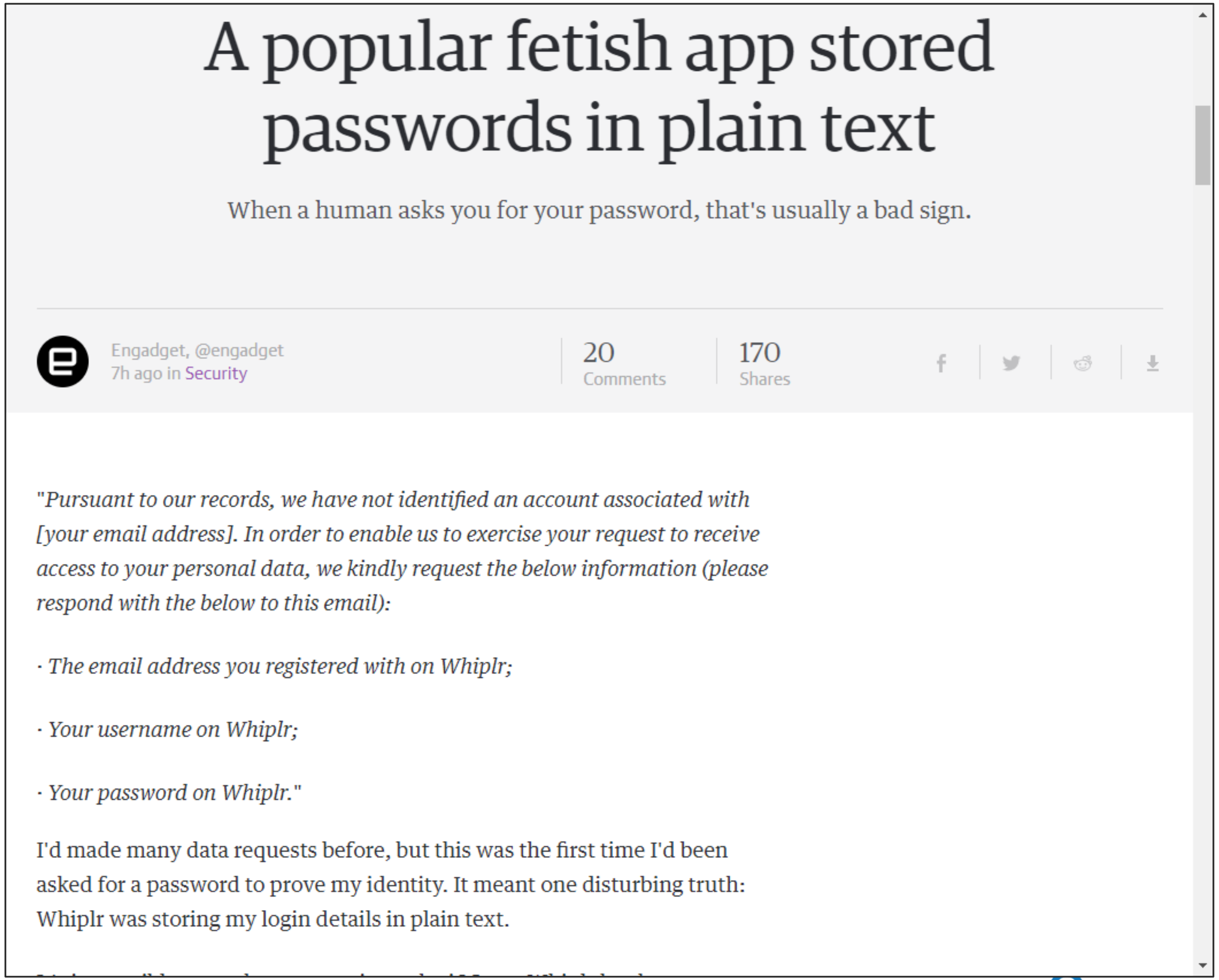
Note: The dates are approximate.

Name	Compressed	Uncompressed	Date	Notes
Rockyou	rockyou.txt.bz2 (60,498,886 bytes)	n/a	2009-12	Best list available; huge, stolen unencrypted
Rockyou with count	rockyou-withcount.txt.bz2 (59,500,255 bytes)	n/a		

https://wiki.skullsecurity.org/Passwords

Active Online Attacks
Plain Text Passwords

From 07 Sept 2018



A popular fetish app stored passwords in plain text

When a human asks you for your password, that's usually a bad sign.

Engadget, @engadget
7h ago in Security

20 Comments | 170 Shares

"Pursuant to our records, we have not identified an account associated with [your email address]. In order to enable us to exercise your request to receive access to your personal data, we kindly request the below information (please respond with the below to this email):

· The email address you registered with on Whiplr;

· Your username on Whiplr;

· Your password on Whiplr."

I'd made many data requests before, but this was the first time I'd been asked for a password to prove my identity. It meant one disturbing truth: Whiplr was storing my login details in plain text.

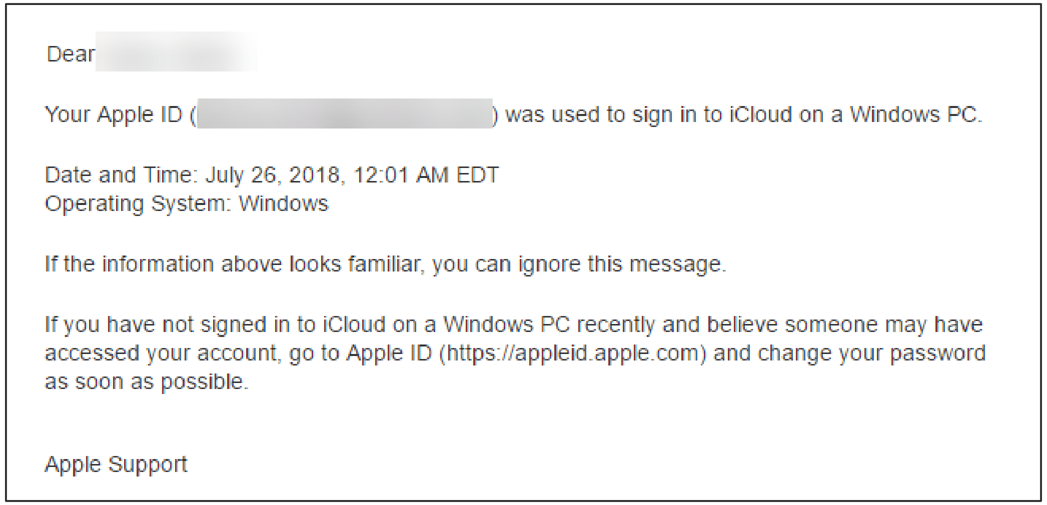
https://www.engadget.com/2018/09/07/whiplr-plain-text-passwords/

Active Online Attacks
Attack Success Factors

- People pick common, easily guessed passwords, such as dictionary words and tend to use the same password at different sites
- Default passwords don’t get changed prior to bringing a system into production
- Systems often do not have or do not enforce strong password policies
  - They allow weak and/or short passwords
- Systems may not implement failed login detection to lockout an account or throttling to slow the login process after a few consecutive failed logins
- Poor physical security or systems that are too easily accessible (anyone could walk up and use them) could allow guessing attacks

Active Online Attacks: General Countermeasures

- Monitor systems for authentication failures
- Immediately notify SysAdmin and other people who need to know of the attack
  - Immediately notify the account user of an attack by sending email to the address on file
- Log the important information and keep it for a sufficiently long period (6-12 months) to assist in forensics and incident response



Dear [redacted]

Your Apple ID ([redacted]) was used to sign in to iCloud on a Windows PC.

Date and Time: July 26, 2018, 12:01 AM EDT
Operating System: Windows

If the information above looks familiar, you can ignore this message.

If you have not signed in to iCloud on a Windows PC recently and believe someone may have accessed your account, go to Apple ID (https://appleid.apple.com) and change your password as soon as possible.

Apple Support

Example notification e-mail

Active Online Attacks: Dictionary Attack Countermeasures – 1

- Do not permit weak passwords to be used
  - Use a tool to check for weak passwords
    - Open Password Filter for ActiveDirectory
    - Pwned Passwords: List of 320 million passwords from different data breaches
- NIST Special Publication 800-63B (*Digital Identity Guidelines*):

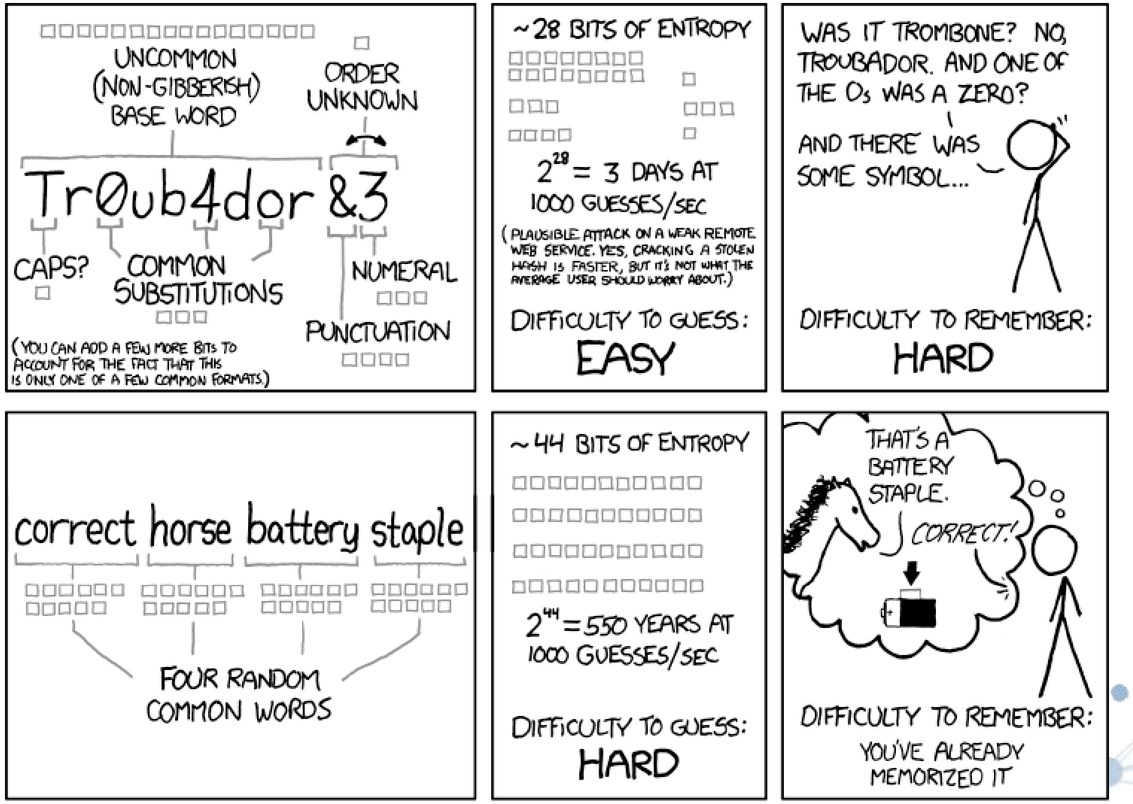
  “When processing requests to establish and change memorized secrets, verifiers SHALL compare the prospective secrets against a list that contains values known to be commonly-used, expected, or compromised. For example, the list MAY include, but is not limited to:

    - Passwords obtained from previous breach corpuses.
    - Dictionary words.
    - Repetitive or sequential characters (e.g. ‘aaaaaa’, ‘1234abcd’).
    - Context-specific words, such as the name of the service, the username, and derivatives thereof.

  If the chosen secret is found in the list, the CSP [credential service provider] or verifier SHALL advise the subscriber that they need to select a different secret, SHALL provide the reason for rejection, and SHALL require the subscriber to choose a different value.”

Active Online Attacks: Dictionary Attack Countermeasures – 2

- Require stronger (longer) passwords
  - Generate a long, random string of characters for a password using a password manager
    - No need to remember it since it is stored in a password manager
  - Use passphrases, sentences
    - [Diceware]
  - Requirements to use special characters do increase the cryptographic entropy, but make remembering passwords much harder
  - Many systems support very long (>128 characters password)
    - Use passwords at least 20 characters long
- Change default passwords on newly fielded and on upgraded systems


UNCOMMON (NON-GIBBERISH) BASE WORD
ORDER UNKNOWN
Tr0ub4dor &3
CAPS?
COMMON SUBSTITUTIONS
NUMERAL
PUNCTUATION
(YOU CAN ADD A FEW MORE BITS TO ACCOUNT FOR THE FACT THAT THIS IS ONLY ONE OF A FEW COMMON FORMATS.)
~28 BITS OF ENTROPY
2^28 = 3 DAYS AT 1000 GUESSES/SEC
(PLAUSIBLE ATTACK ON A WEAK REMOTE WEB SERVICE. YES, CRACKING A STOLEN HASH IS FASTER, BUT IT'S NOT WHAT THE AVERAGE USER SHOULD WORRY ABOUT.)
DIFFICULTY TO GUESS: EASY
WAS IT TROMBONE? NO, TROUBADOR. AND ONE OF THE Os WAS A ZERO?
AND THERE WAS SOME SYMBOL...
DIFFICULTY TO REMEMBER: HARD
correct horse battery staple
FOUR RANDOM COMMON WORDS
~44 BITS OF ENTROPY
2^44 = 550 YEARS AT 1000 GUESSES/SEC
DIFFICULTY TO GUESS: HARD
THAT'S A BATTERY STAPLE.
CORRECT!
DIFFICULTY TO REMEMBER: YOU'VE ALREADY MEMORIZED IT

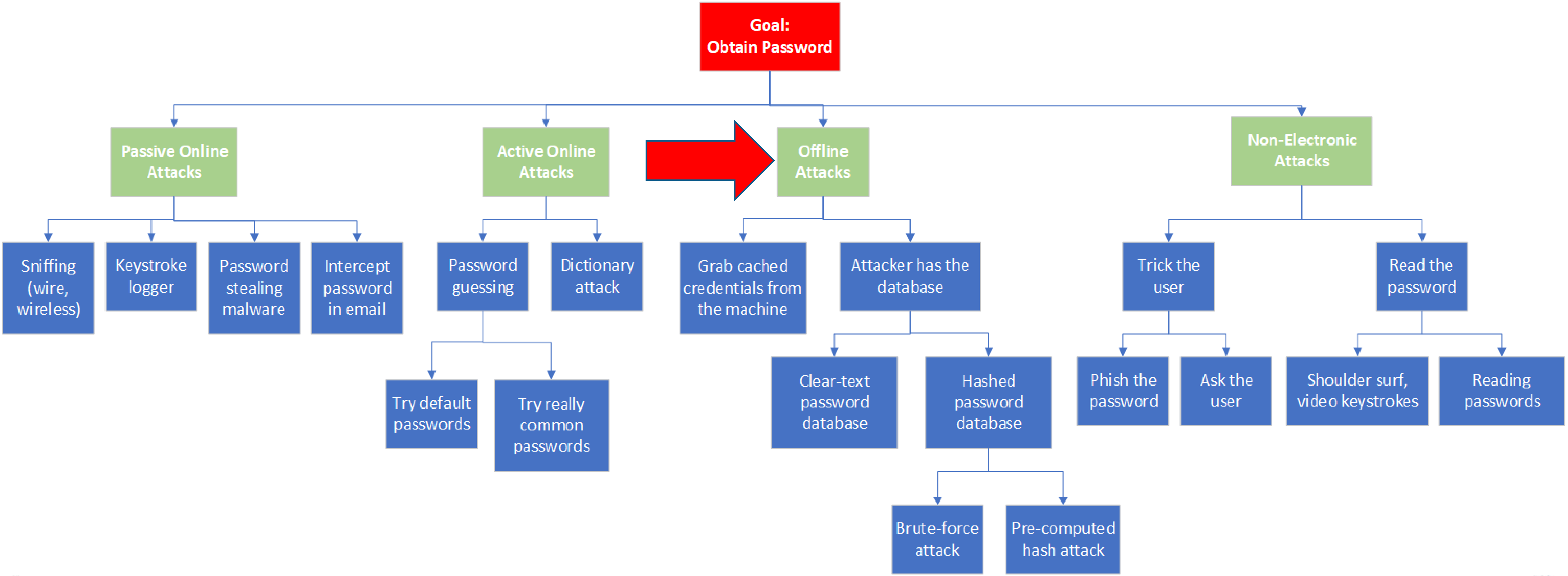
THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

https://xkcd.com/936/

Active Online Attacks: Dictionary Attack Countermeasures – 3

- Use anti-automation tools to slow an attacker
- Slowing an attacker down can make it too expensive (time-wise) to continue attacking
  - Make the login process take slightly longer
    - Not terribly annoying to the user, but can really hinder or stop an automated attack
      - CAPTCHA (prove you are a human)
      - Security questions (prove you know details about the claimed identity)
      - Take increasingly more time (+5, 10, 30 or more seconds) with **each successive** failed login
- Lock an account after a small number of consecutive failed login attempts
  - The account can be locked for some **specified amount** of time period, or
  - The account can be **permanently** locked until it gets reset by an administrator
- Destroy data on the device after a certain number of consecutive login failures have been exceeded (used in mobile devices)

Password Attack Tree



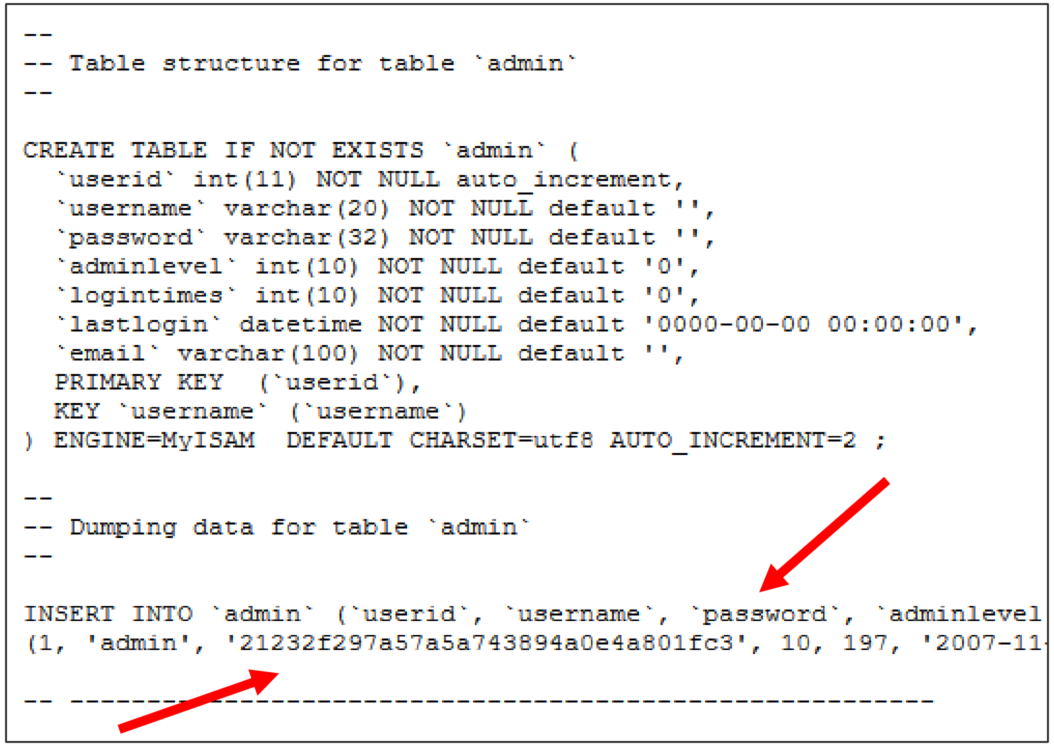
SECURITY INNOVATION

Offline Attacks: Grab Cached Credentials from the Machine

- If the attacker has physical access to the machine, there are a variety of places where cached credentials may be found
  - Web browsers can store login credentials for various websites
  - The operating system may store cached credentials when the domain controller is unavailable
    - There are tools (e.g., mimikatz) that can retrieve credentials using very clever in-memory techniques
  - Applications of various types may have credentials cached (“remember the password”)

Offline Attacks:
Acquiring the Password Database

- In offline attacks, the attacker has a copy of the account database or has access to the credentials database
  - Cleartext database
  - Encrypted database
  - Database with password hashes
- There are numerous ways to acquire a password database
  - Unprotected backup or debug copies are sometimes left on-line
  - SQL injection attacks can be used dump the database
  - Misconfigured database or database servers
    - Often found by doing simple Google or Shodan searches
      - For example: filetype:sql +dump
  - Stolen databases are sold or shared with other criminals
- The attacker can conduct the attack at his leisure and can apply significant resources toward the goal of cracking the password database



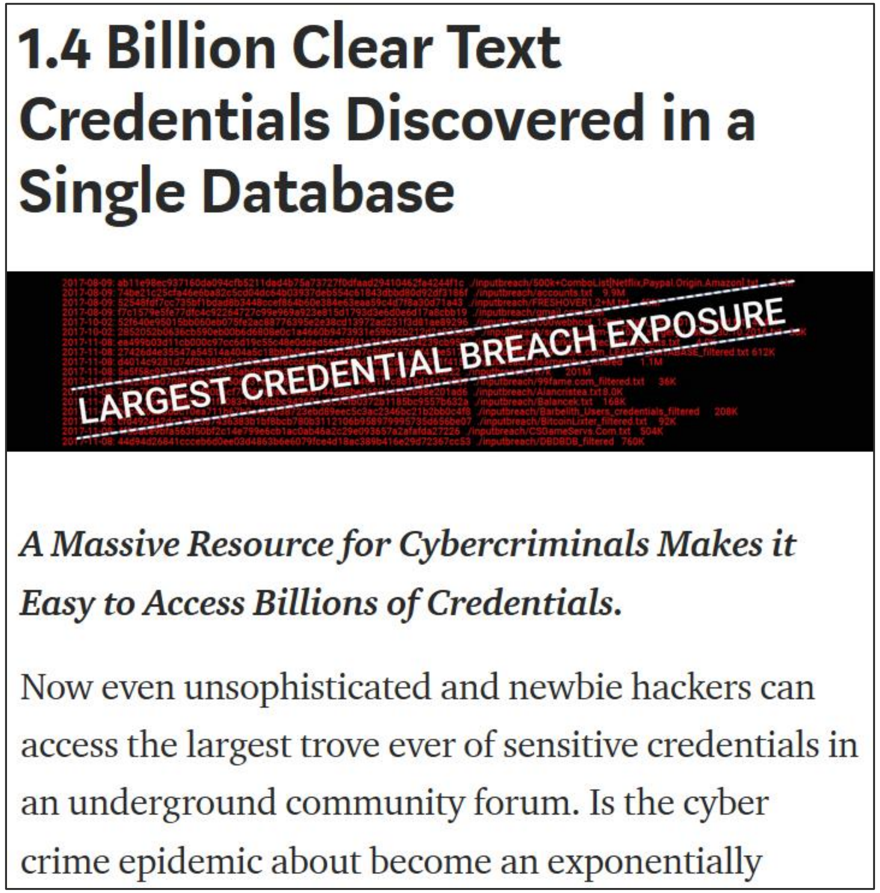
```
--
-- Table structure for table `admin`
--

CREATE TABLE IF NOT EXISTS `admin` (
  `userid` int(11) NOT NULL auto_increment,
  `username` varchar(20) NOT NULL default '',
  `password` varchar(32) NOT NULL default '',
  `adminlevel` int(10) NOT NULL default '0',
  `logintimes` int(10) NOT NULL default '0',
  `lastlogin` datetime NOT NULL default '0000-00-00 00:00:00',
  `email` varchar(100) NOT NULL default '',
  PRIMARY KEY  (`userid`),
  KEY `username` (`username`)
) ENGINE=MyISAM  DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;

--
-- Dumping data for table `admin`
--

INSERT INTO `admin` (`userid`, `username`, `password`, `adminlevel
(1, 'admin', '21232f297a57a5a743894a0e4a801fc3', 10, 197, '2007-11-

-- ---------------------------------------------------------
```

Example database found on-line using a Google search

SECURITY INNOVATION

Offline Attacks:
Attacker has a Clear-Text Password Database

- Even with a database containing encrypted passwords, an attacker may be able to have the system return the entire clear-text data using a SQL injection attack
    - The system does the decryption
- Once an attacker has the **clear-text** password database, the attacker has access to all of the accounts!



1.4 Billion Clear Text Credentials Discovered in a Single Database

A Massive Resource for Cybercriminals Makes it Easy to Access Billions of Credentials.

Now even unsophisticated and newbie hackers can access the largest trove ever of sensitive credentials in an underground community forum. Is the cyber crime epidemic about become an exponentially

https://medium.com/4iqdelvedeep/1-4-billion-clear-text-credentials-discovered-in-a-single-database-3131d0a1ae14

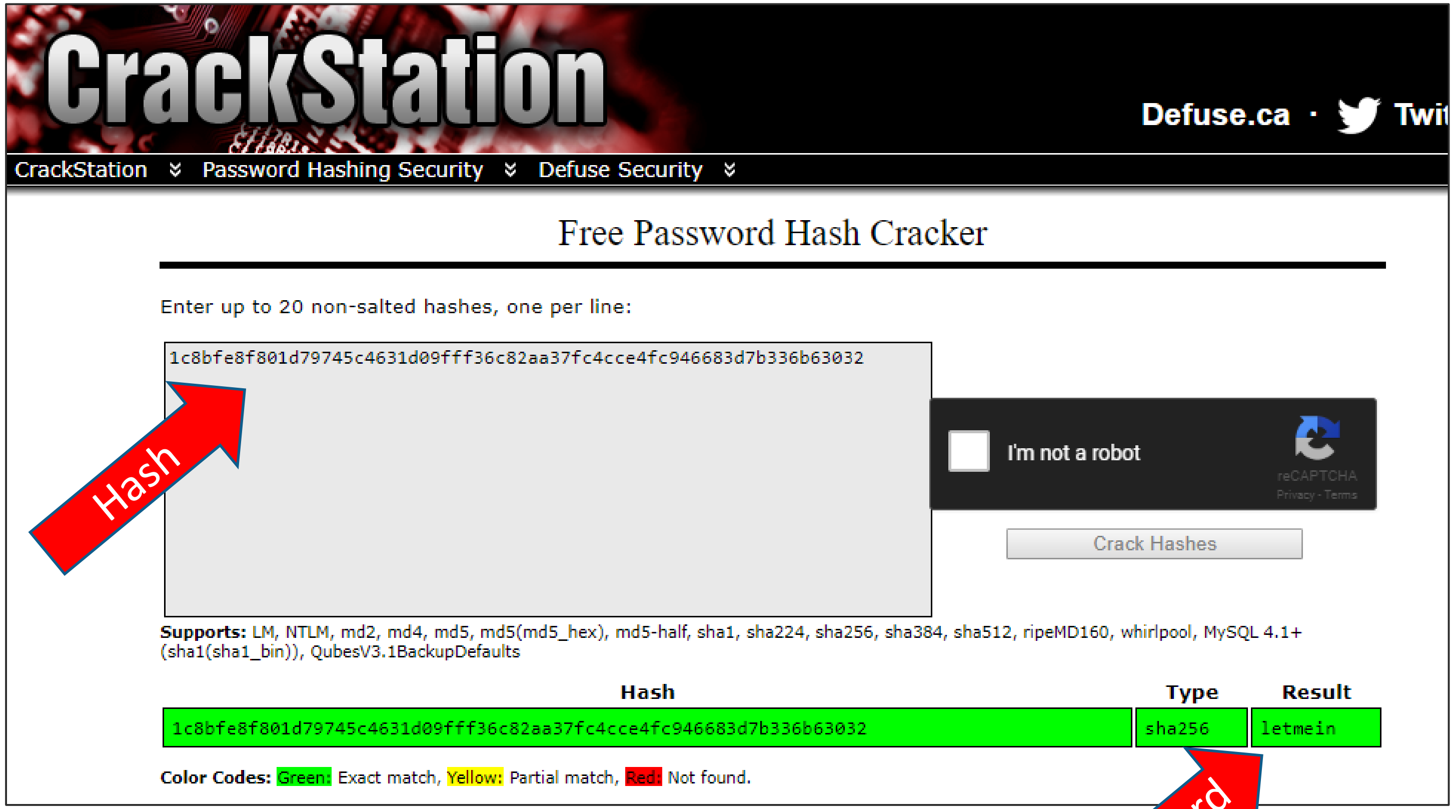
Offline Attacks:
Dictionary Attack with Hashed Passwords

- Instead of storing clear text passwords in a database, the developer stores the HASH(password)
    - Developers often use **general purpose cryptographic hash algorithms** like MD5 or SHA-1
- If the attacker can steal the database of the password hashes, then he can deduce what the passwords are!

```
Pre-compute HASH of each word in the dictionary
For each hash in the stolen PASS_DB
    IF hash from PASS_DB == pre-computed hash of dictionary word
    THEN we know what the password is
    ELSE try the next dictionary word from the stolen PASS_DB
```

- Attackers will also compute common password permutations
    - password1, drowssap, passwordpassword, etc.
- Automation enables these attacks to be done very quickly

Offline Attacks:
Dictionary Attack with Hashed Passwords



Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

```
1c8bfe8f801d79745c4631d09fff36c82aa37fc4cce4fc946683d7b336b63032
```

□ I'm not a robot

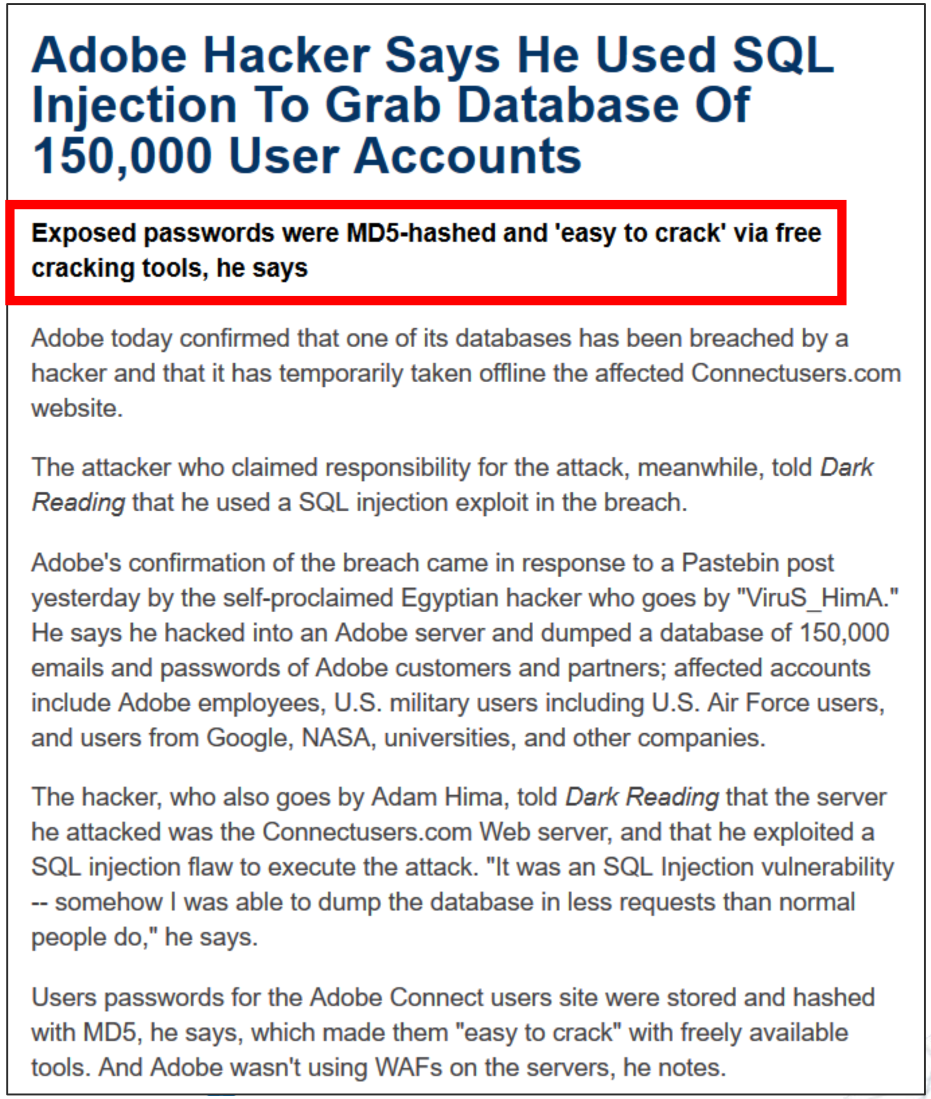
Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
1c8bfe8f801d79745c4631d09fff36c82aa37fc4cce4fc946683d7b336b63032	sha256	letmein

Color Codes: 🟩 Green: Exact match, 🟨 Yellow: Partial match, 🟥 Red: Not found.

https://crackstation.net/



Adobe Hacker Says He Used SQL Injection To Grab Database Of 150,000 User Accounts

Exposed passwords were MD5-hashed and 'easy to crack' via free cracking tools, he says

Adobe today confirmed that one of its databases has been breached by a hacker and that it has temporarily taken offline the affected Connectusers.com website.

The attacker who claimed responsibility for the attack, meanwhile, told *Dark Reading* that he used a SQL injection exploit in the breach.

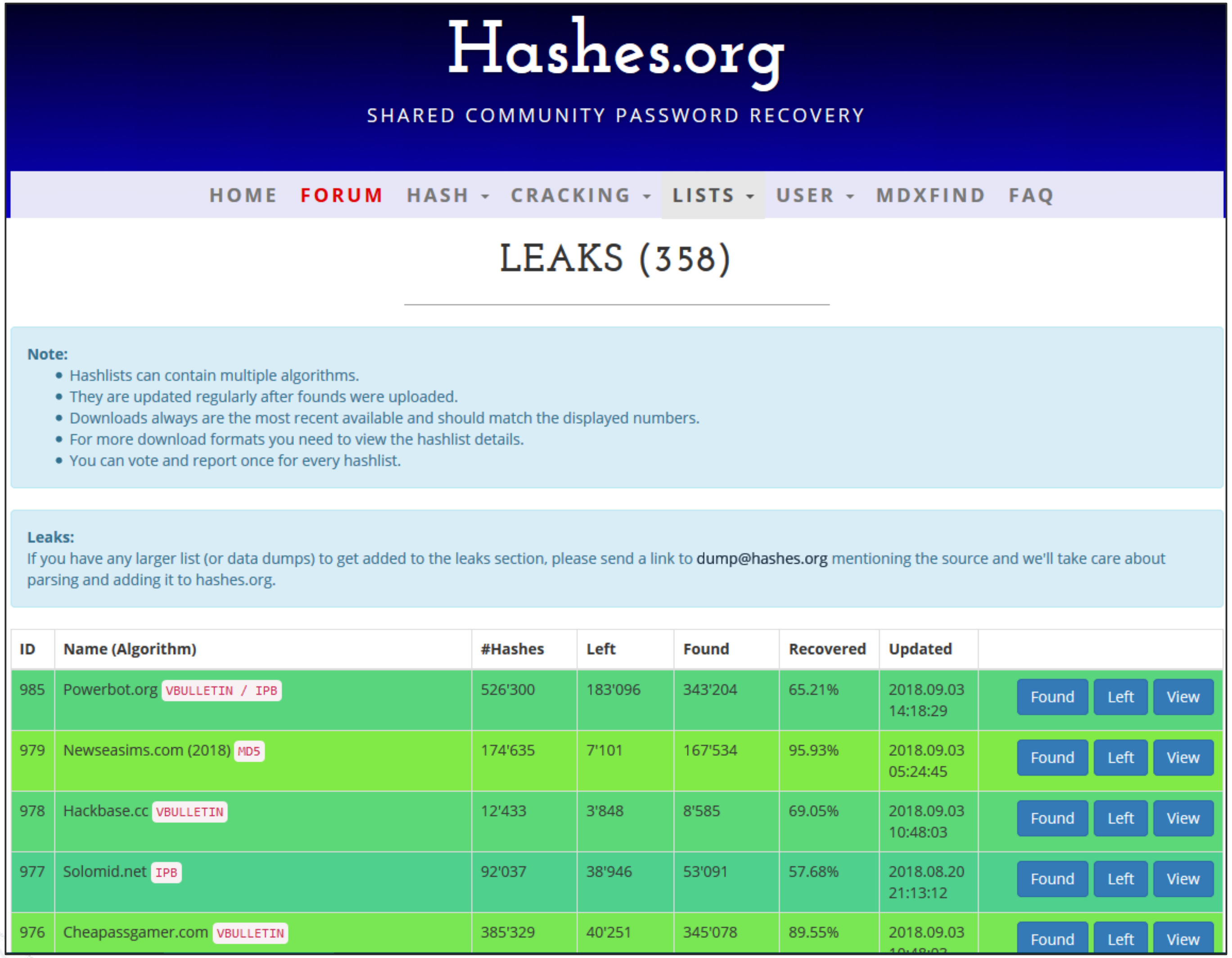
Adobe's confirmation of the breach came in response to a Pastebin post yesterday by the self-proclaimed Egyptian hacker who goes by "ViruS_HimA." He says he hacked into an Adobe server and dumped a database of 150,000 emails and passwords of Adobe customers and partners; affected accounts include Adobe employees, U.S. military users including U.S. Air Force users, and users from Google, NASA, universities, and other companies.

The hacker, who also goes by Adam Hima, told *Dark Reading* that the server he attacked was the Connectusers.com Web server, and that he exploited a SQL injection flaw to execute the attack. "It was an SQL Injection vulnerability -- somehow I was able to dump the database in less requests than normal people do," he says.

Users passwords for the Adobe Connect users site were stored and hashed with MD5, he says, which made them "easy to crack" with freely available tools. And Adobe wasn't using WAFs on the servers, he notes.

https://www.darkreading.com/attacks-breaches/adobe-hacker-says-he-used-sql-injection-to-grab-database-of-150000-user-accounts/d/d-id/1138677

Offline Attacks:
Dictionary Attack with Hashed Passwords



LEAKS (358)

Note:

- Hashlists can contain multiple algorithms.
- They are updated regularly after founds were uploaded.
- Downloads always are the most recent available and should match the displayed numbers.
- For more download formats you need to view the hashlist details.
- You can vote and report once for every hashlist.

Leaks:
If you have any larger list (or data dumps) to get added to the leaks section, please send a link to dump@hashes.org mentioning the source and we'll take care about parsing and adding it to hashes.org.

ID	Name (Algorithm)	#Hashes	Left	Found	Recovered	Updated	
985	Powerbot.org VBULLETIN / IPB	526'300	183'096	343'204	65.21%	2018.09.03 14:18:29	Found Left View
979	Newseasims.com (2018) MD5	174'635	7'101	167'534	95.93%	2018.09.03 05:24:45	Found Left View
978	Hackbase.cc VBULLETIN	12'433	3'848	8'585	69.05%	2018.09.03 10:48:03	Found Left View
977	Solomid.net IPB	92'037	38'946	53'091	57.68%	2018.08.20 21:13:12	Found Left View
976	Cheapassgamer.com VBULLETIN	385'329	40'251	345'078	89.55%	2018.09.03	Found Left View

https://hashes.org/leaks.php

- At this site with password databases, only a few percent of the passwords have been cracked when the database has passwords hashed with a slow password hashing algorithm such as bcrypt
- Where weak hashing is used (MD5), over 90% of the passwords have been cracked
- Lesson: use slow password hashing algorithms

Offline Attacks:
Brute-force Attacks – 1

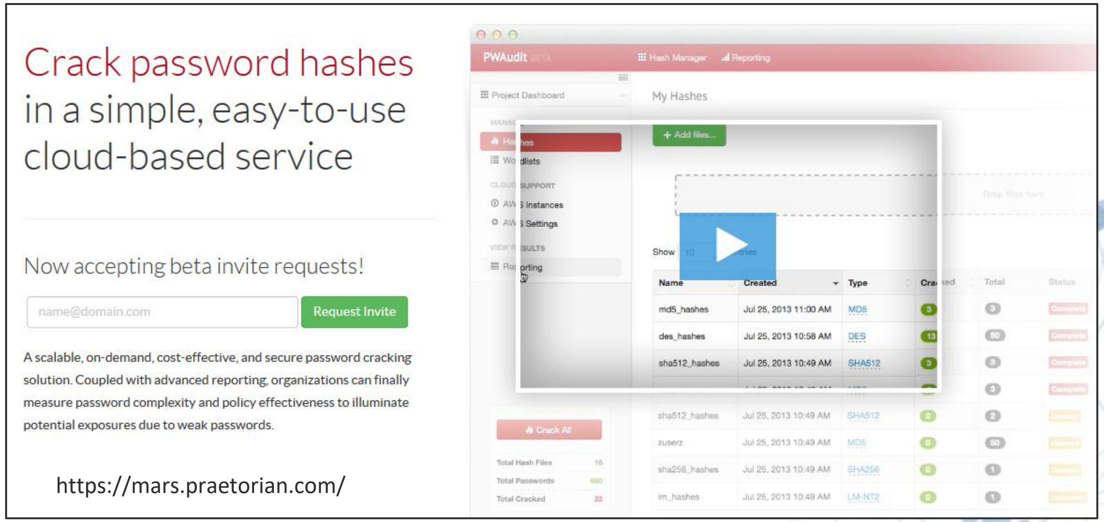
- A **brute-force** attack is where an attacker keeps trying various **character combinations** against each hash in the database until:
  - The account is cracked
  - Too much time has been expended
- Using automated tools, an attacker has a fairly good chance of “guessing” a shorter, less-complex static password
  - This is referred to as “cracking”
- Brute-force attacks are capable of cracking random and complex passwords
  - A hybrid attack (dictionary + brute-force) attack is most commonly done
    - Start with a dictionary of common/stolen passwords
    - Next try common permutations of those words (reversed, with numbers at the beginning/end, etc.)
    - Finally try various character combinations (brute-force)

SECURITY INNOVATION

Offline Attacks: Brute-force Attacks – 2

- Brute-force attacks can be very time-consuming, depending on the size of the character set used
  - A – Z, a – z
  - A – Z, a – z, 0 – 9
  - A – Z, a – z, 0 – 9, top row of the keyboard
  - A – Z, a – z, 0 – 9, top row of the keyboard, other symbols

- These attacks become **very difficult** to do in a reasonable time with longer passwords and more characters in the character set
  - Attack optimization techniques increase the speed of the attack
    - **Distributed cracking** is when attackers use multiple systems simultaneously to crack the password hashes
    - Rainbow tables (see next slide)



Crack password hashes
in a simple, easy-to-use
cloud-based service

Now accepting beta invite requests!

name@domain.com

Request Invite

A scalable, on-demand, cost-effective, and secure password cracking solution. Coupled with advanced reporting, organizations can finally measure password complexity and policy effectiveness to illuminate potential exposures due to weak passwords.

https://mars.praetorian.com/

My Hashes

Name	Created	Type	Cracked	Total	Status
md5_hashes	Jul 25, 2013 11:00 AM	MD5	3	3	[illegible]
des_hashes	Jul 25, 2013 10:58 AM	DES	13	50	[illegible]
sha512_hashes	Jul 25, 2013 10:49 AM	SHA512	3	3	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	3	[illegible]
sha512_hashes	Jul 25, 2013 10:49 AM	SHA512	0	2	[illegible]
zuserz	Jul 25, 2013 10:49 AM	MD5	0	50	[illegible]
sha256_hashes	Jul 25, 2013 10:49 AM	SHA256	0	1	[illegible]
lm_hashes	Jul 25, 2013 10:49 AM	LM-NT2	0	1	[illegible]

Total Hash Files 16
Total Passwords 660
Total Cracked 22

Offline Attacks:
Pre-computed Hashes/Rainbow Tables

- A **rainbow table** is an enormous, pre-computed table of hash values for every possible combination of characters in a given character set

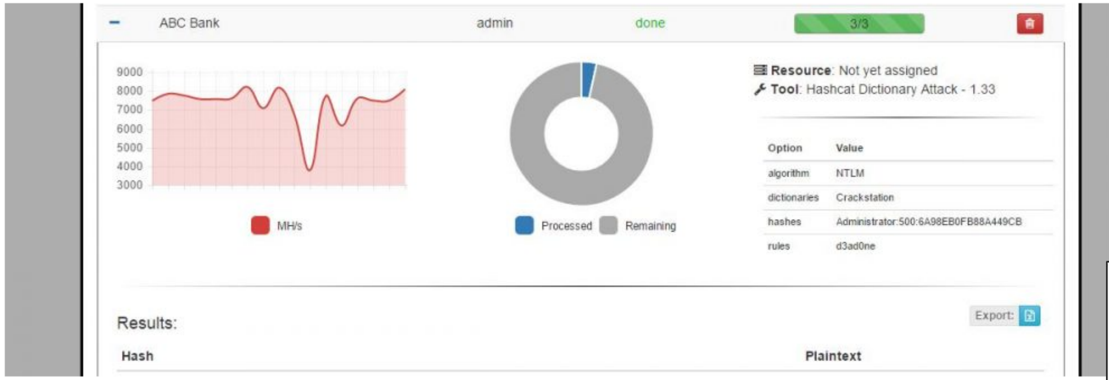
```
FOR every permutation of the char set (A-Z, 0-9, special chars)
    Compute HASH(string)
    Store the hash and the string in the rainbow table
FOR each hash in the stolen PASS_DB
    IF hash from PASS_DB == HASH(string) from the rainbow table
    THEN we know what the password is!
    ELSE try the next hash from the stolen PASS_DB
```

- The attacker compares a hash from the password database to a pre-computed hash
  - A match means that the attacker knows the password used to generate the hash that matched
- Pre-computing hashes takes a significant amount of time and requires a huge amount of disk space
  - All possible Windows LM hashes requires 166 Tb of disk space

Offline Attacks: Distributed Password Cracking

Distributed Password Cracking: CrackLord

CyberPunk » Password Attacks



ABC Bank admin done 3/3

Resource: Not yet assigned
Tool: Hashcat Dictionary Attack - 1.33

Option	Value
algorithm	NTLM
dictionaries	Crackstation
hashes	Administrator:500:6A98EB0FB88A449CB
rules	d3ad0ne

Results: Export:

Hash	Plaintext

Over the past several years the world of password cracking has exploded with new tools and techniques. These new techniques have made it easier than ever to reverse captured password hashes. With the addition of powerful techniques, from GPGPU cracking to rainbow tables, it is easier than ever to access the plaintext for fun and profit. Furthermore, during our day jobs we have found that many other processes can benefit from distributed access to the resources of high powered systems. With many people requesting access to this, CrackLord was born.

CrackLord provides a scalable, pluggable, and distributed system for both password cracking as well as any other jobs we need. Better said, CrackLord is a way to load balance the resources, such as CPU and GPU, from multiple hardware systems into a single queueing service across two primary services: the Resource and Queue.

https://n0where.net/distributed-password-cracking-cracklord

Cracking in the Cloud with CUDA GPUs

February 28, 2017 r0kh Kali Linux News

Due to increasing popularity of cloud-based instances for password cracking, we decided to focus our efforts into streamlining Kali's approach. We've noticed that Amazon's AWS P2-Series and Microsoft's Azure NC-Series are focused on Windows and Ubuntu. The corresponding blog posts and guides followed suit. Although these instances are limited by the ities, the ability to quickly deploy a Kali instance with CUDA support is appealing.

rs has always been a source of frustration; fortunately, improvements in packaging seamless. Although we've done the work for you in the cloud offerings, we'd like to n use.

https://www.kali.org/news/cloud-cracking-with-cuda-gpu/

25-GPU cluster cracks every standard Windows password in <6 hours

All your passwords are belong to us.

by Dan Goodin - Dec 10, 2012 12:00 am UTC

HACKING PRIVACY VIRTUALIZATION 262

Welcome to Radeon City, population: 8. It's one of five servers that make up a high-performance password-cracking cluster.

Jeremi Gosney

http://arstechnica.com/security/2012/12/25-gpu-cluster-cracks-every-standard-windows-password-in-6-hours/

Offline Attacks
Attack Success Factors

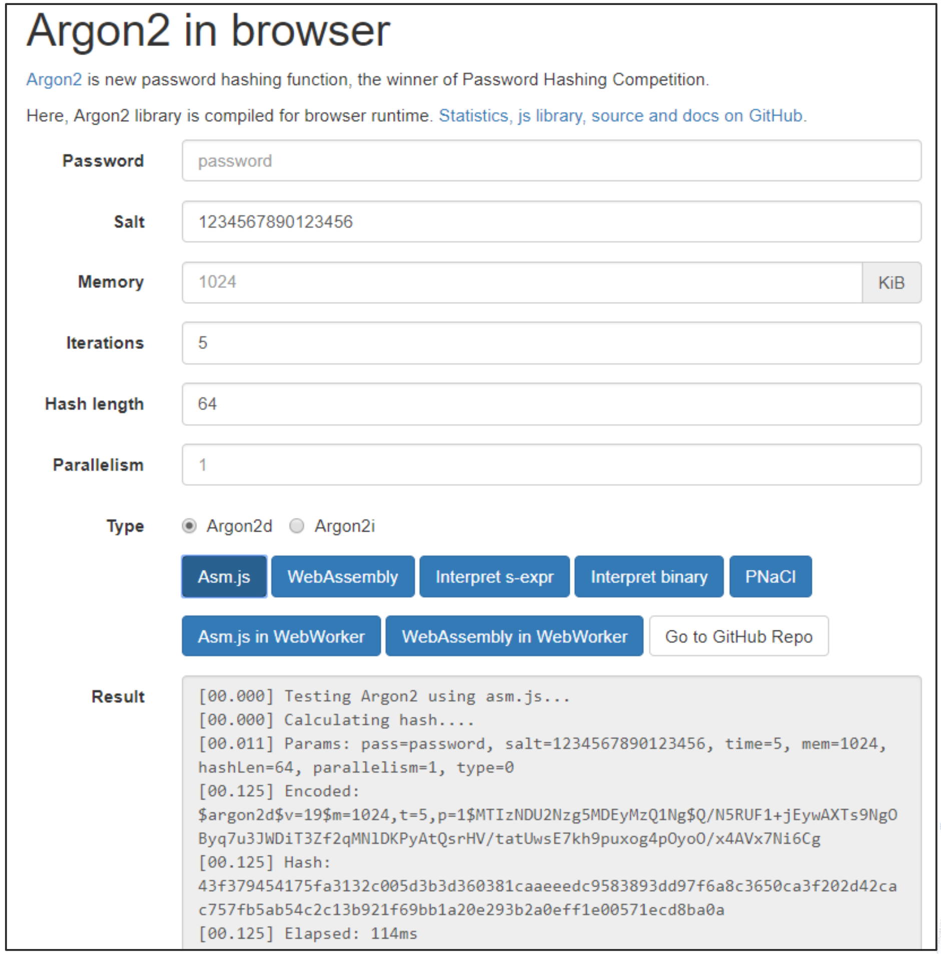
- Offline attacks are successful due to many of the previous issues mentioned…
  - Unchanged default passwords
  - Dictionary words used as passwords
  - No or weak password policies
  - Users can’t remember longer, harder-to-crack passwords
  - Password crackers can readily crack most, if not all, passwords that users can be reasonably expected to remember
  - Poor physical security
- System security is weak, allowing attackers to access the password database
- Weak and/or fast, general-purpose hashing algorithms are used to hash passwords
- Passwords are not salted or are salted with weak salts
- Attacker tools are very sophisticated and scale up

Offline Attack Countermeasures – 1

- All of the previous recommendations and…
- Protect the password file
  - Ensure the system itself is well-protected to help protect the password database
- Do not store clear-text and reversibly encrypted passwords in a database
- Don’t invent your own hashing algorithm – use something that has been well-vetted in the industry
- Use hashes created with a **strong, slow password hashing algorithms**
  - Argon2, PBKDF2, scrypt, bcrypt

Offline Attack Countermeasures – 2
Hash Passwords with a Cryptographic Salt

- Store **hashed passwords** instead of clear text or encrypted passwords
- To make it much harder for the attacker to pre-compute hashes, concatenate a “**salt**”, a sufficiently long random string to the password, and then calculate the hash
  - An attacker would have to spend a very long time calculating the hashes for every word in the dictionary or every possible character combination with every salt
  - Salting makes it **less feasible** to run a dictionary or brute-force attack since the attacker would have to re-hash the entire dictionary for **every possible salt**
    - Rainbow tables would be incredibly large
- Since every user should have their own unique, random salt, two users that have the same password will have different hashes ☺



Argon2 in browser

Argon2 is new password hashing function, the winner of Password Hashing Competition.

Here, Argon2 library is compiled for browser runtime. Statistics, js library, source and docs on GitHub.

Password password

Salt 1234567890123456

Memory 1024 KiB

Iterations 5

Hash length 64

Parallelism 1

Type ● Argon2d ○ Argon2i

Result

```
[00.000] Testing Argon2 using asm.js...
[00.000] Calculating hash....
[00.011] Params: pass=password, salt=1234567890123456, time=5, mem=1024,
hashLen=64, parallelism=1, type=0
[00.125] Encoded:
$argon2d$v=19$m=1024,t=5,p=1$MTIzNDU2Nzg5MDEyMzQ1Ng$Q/N5RUF1+jEywAXTs9NgO
Byq7u3JWDiT3Zf2qMNlDKPyAtQsrHV/tatUwsE7kh9puxog4pOyoO/x4AVx7Ni6Cg
[00.125] Hash:
43f379454175fa3132c005d3b3d360381caaeeedc9583893dd97f6a8c3650ca3f202d42ca
c757fb5ab54c2c13b921f69bb1a20e293b2a0eff1e00571ecd8ba0a
[00.125] Elapsed: 114ms
```

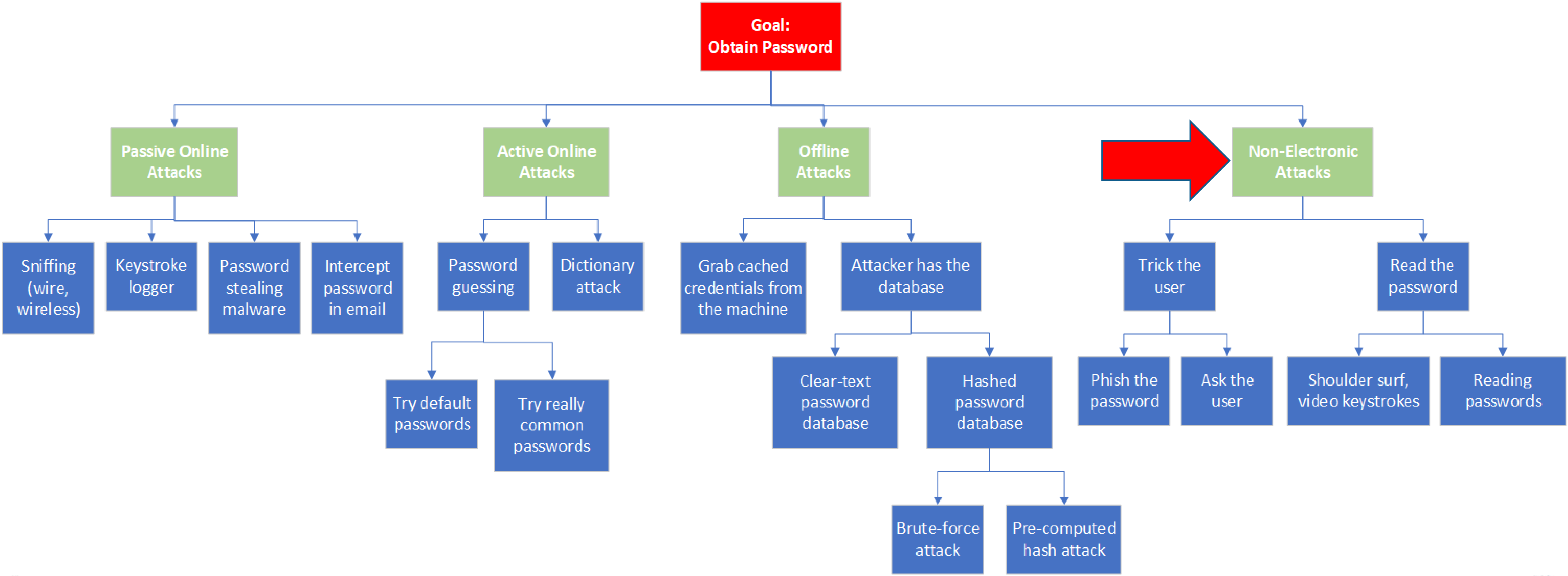
http://antelle.net/argon2-browser/

SECURITY INNOVATION

Offline Attack Countermeasures – 3
Use Slow Hashing Algorithms

- **General-purpose** cryptographic hashes are designed to be very fast
  - This helps attackers conduct faster cracking attacks
- MD-5 and SHA-1 hashes are broken and really should not be used for anything
- Instead, use slow **password hashing** algorithms
  - Argon2, PBKDF2, bcrypt, scrypt
  - They are designed to take **much** longer than general-purpose hashes, making brute forcing and calculating rainbow tables very slow
  - These password hashes include parameters that can be tweaked to increase the amount of work (time) required to hash
- Don’t reuse salts; use a different salt for every user
- Generate salts using a Cryptographically Secure Pseudo-Random Number Generator (CSPRNG)
- Store salts in a different database from the hashed passwords
  - Or at least store salts in an encrypted column

Password Attack Tree



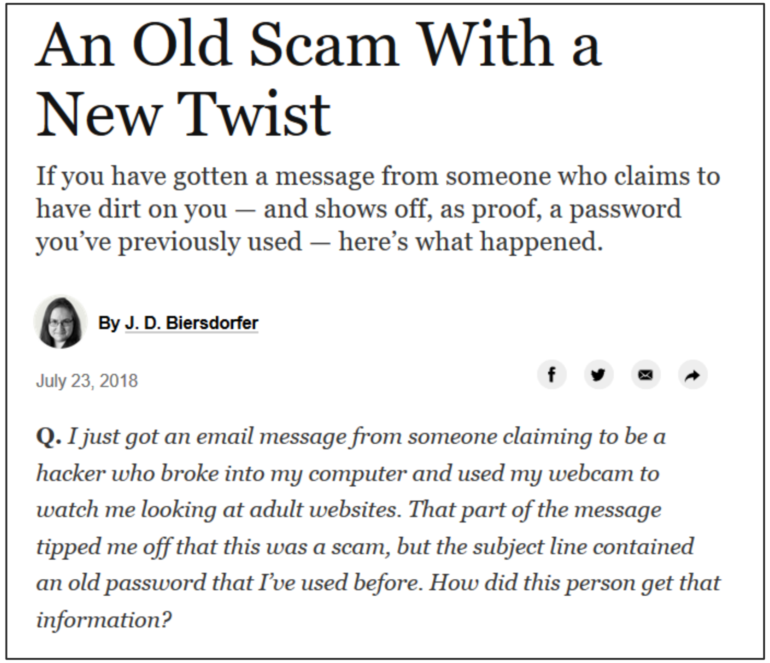
Non-Electronic Attacks

- These are attacks at layer 8 of the OSI model ☺ (the human)
- Once the attacker has the authentication credentials, he can log into the system and impersonate the victim

https://nakedsecurity.sophos.com/2017/12/18/cryptocoins-robbed-at-gunpoint/

Non-Electronic Attacks

- Phishing
  - Users are tricked into handing over their passwords to an attacker as a result of clever social engineering techniques such as phishing



An Old Scam With a New Twist

If you have gotten a message from someone who claims to have dirt on you — and shows off, as proof, a password you've previously used — here's what happened.

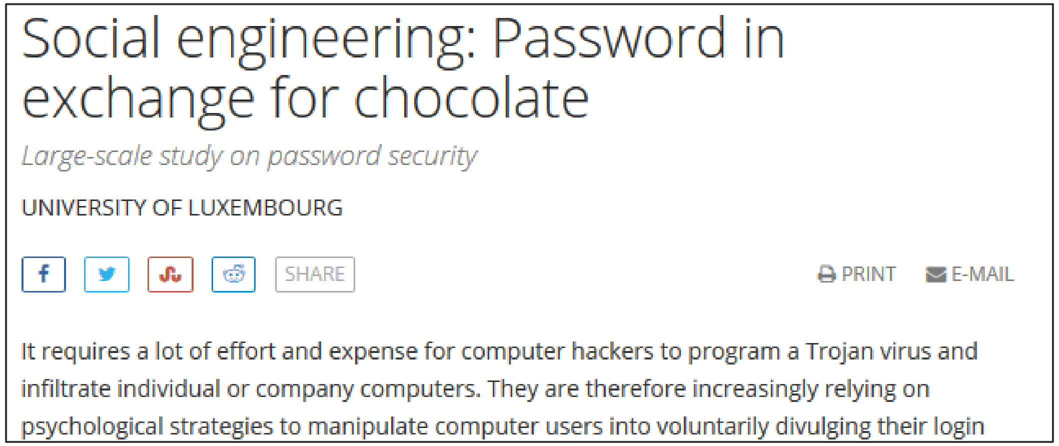
By J. D. Biersdorfer

July 23, 2018

Q. *I just got an email message from someone claiming to be a hacker who broke into my computer and used my webcam to watch me looking at adult websites. That part of the message tipped me off that this was a scam, but the subject line contained an old password that I've used before. How did this person get that information?*

https://www.nytimes.com/2018/07/23/technology/personaltech/phishing-password-email.html

- Asking the user for the password

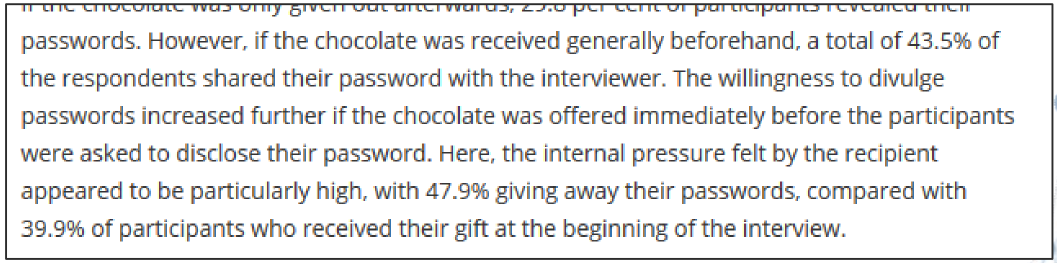


Social engineering: Password in exchange for chocolate

Large-scale study on password security

UNIVERSITY OF LUXEMBOURG

It requires a lot of effort and expense for computer hackers to program a Trojan virus and infiltrate individual or company computers. They are therefore increasingly relying on psychological strategies to manipulate computer users into voluntarily divulging their login

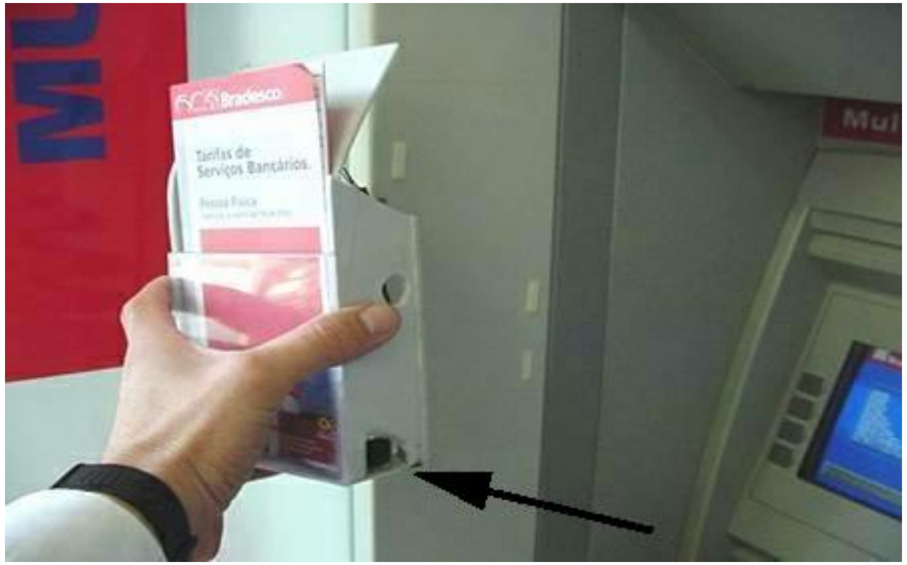


If the chocolate was only given out afterwards, 29.8 per cent of participants revealed their passwords. However, if the chocolate was received generally beforehand, a total of 43.5% of the respondents shared their password with the interviewer. The willingness to divulge passwords increased further if the chocolate was offered immediately before the participants were asked to disclose their password. Here, the internal pressure felt by the recipient appeared to be particularly high, with 47.9% giving away their passwords, compared with 39.9% of participants who received their gift at the beginning of the interview.

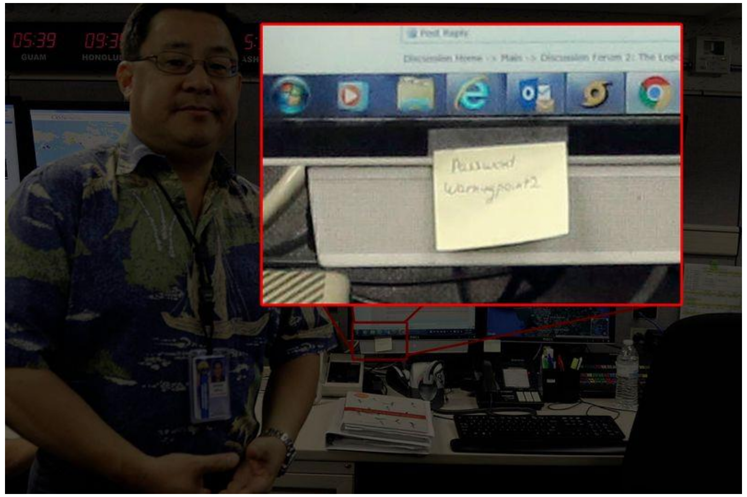
https://www.eurekalert.org/pub_releases/2016-05/uol-sep051216.php

Non-Electronic Attacks
Reading Passwords

- Shoulder surfing or videoing the keystrokes as the user types is a very common way of stealing passwords
  - Frequently done at ATM machines with hidden cameras
- People may write passwords down and leave them exposed



https://www.makeuseof.com/tag/3-danger-signs-look-time-use-atm/



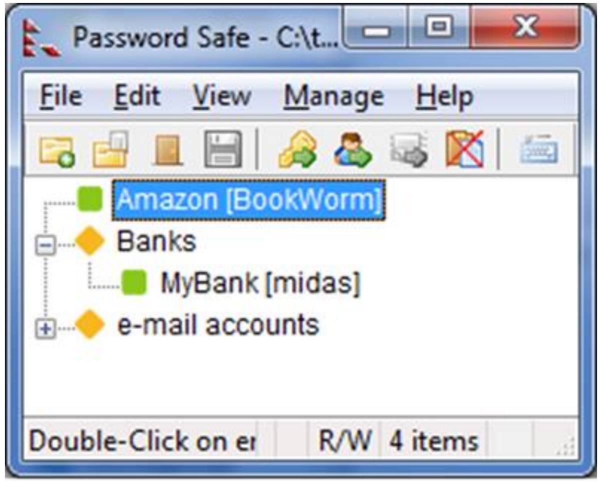
http://www.hawaiinewsnow.com/story/37279882/yes-that-is-a-password-stuck-to-a-screen-at-hawaiis-emergency-management-hq

Non-Electronic Attacks
Attack Success Factors

- Non-electronic attacks against passwords are **extremely** successful
- Victims frequently have poor security hygiene/habits
    - Reusing the same weak password on multiple sites
    - Picking easily guessed passwords
    - Exposing passwords where an attacker can see them
- Attacks are easy to do
    - Don’t require any fancy technology
- Victims are remarkably gullible/easily tricked

Non-Electronic Attacks Countermeasures

- Implement good security hygiene
  - Do not use the same password anywhere else
  - Use a password manager to securely maintain all your passwords
    - Password managers create and store unique, strong passwords for every application/site
      - Generate long, random strings of characters for every password
    - Use a good passphrase to protect the password manager!
- Be wary of phishing attacks
- Cover the keypad when entering passwords/PINs



Amazon [BookWorm]
Banks
MyBank [midas]
e-mail accounts

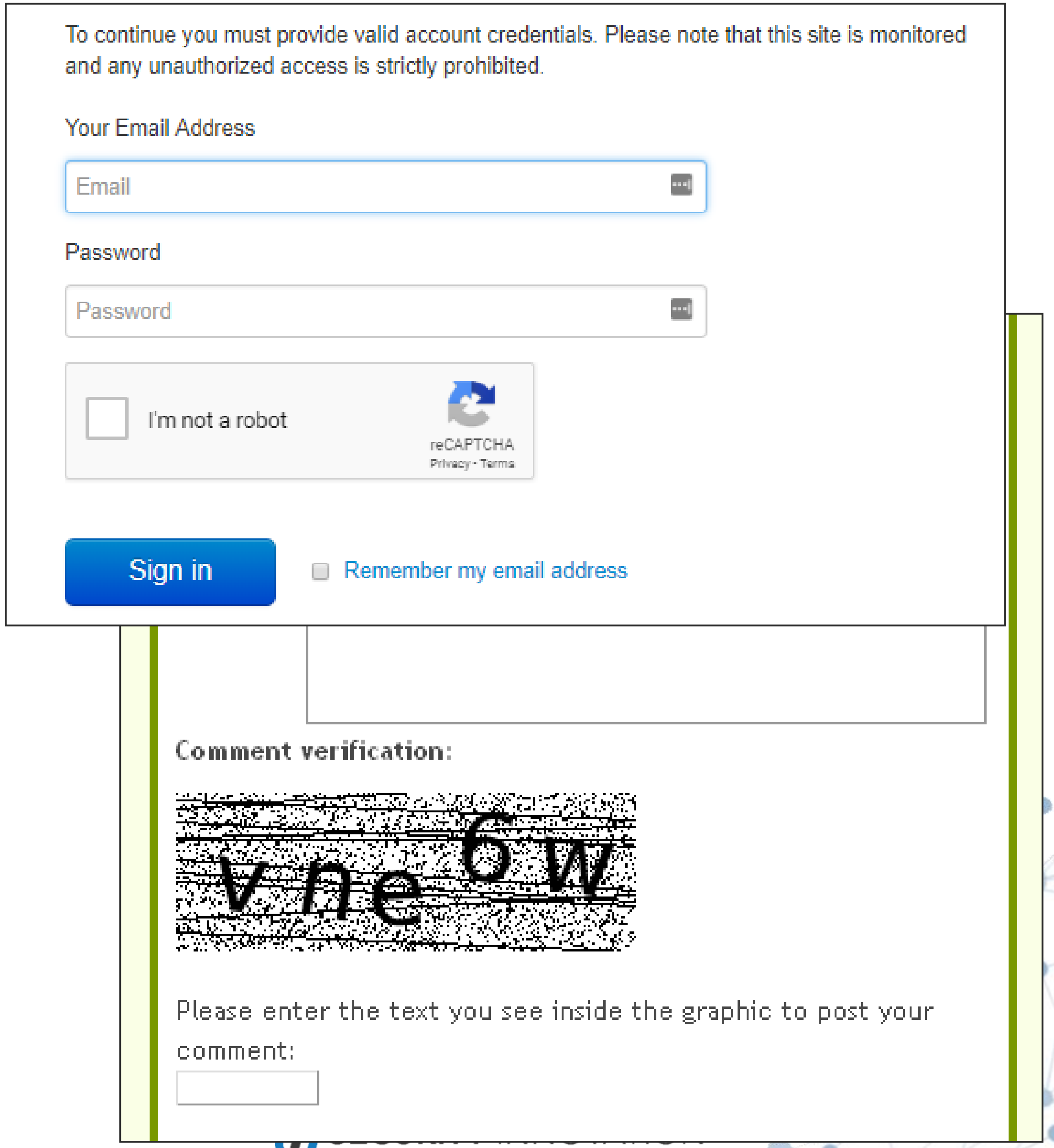
Double-Click on er R/W 4 items

Agenda

- Identification and Authentication
- Common Password Attacks
- ➡ Secure Authentication Practices

Anti-Automation Tools
CAPTCHA

- Captcha is an acronym for "Completely Automated Public Turing test to tell Computers and Humans Apart")
  - It is a type of challenge–response to determine whether or not the entity at the other end of the conversation is a human
    - This is called a Turing Test
- Captchas are designed to prevent machines or bots from automatically doing things
  - Easy for a human to do
  - Very hard for a machine/bot to do
- Captchas authenticate humans rather than machines
- This is an anti-automation technique that can stop brute-force attacks



To continue you must provide valid account credentials. Please note that this site is monitored and any unauthorized access is strictly prohibited.

Your Email Address: Email

Password: Password

□ I'm not a robot

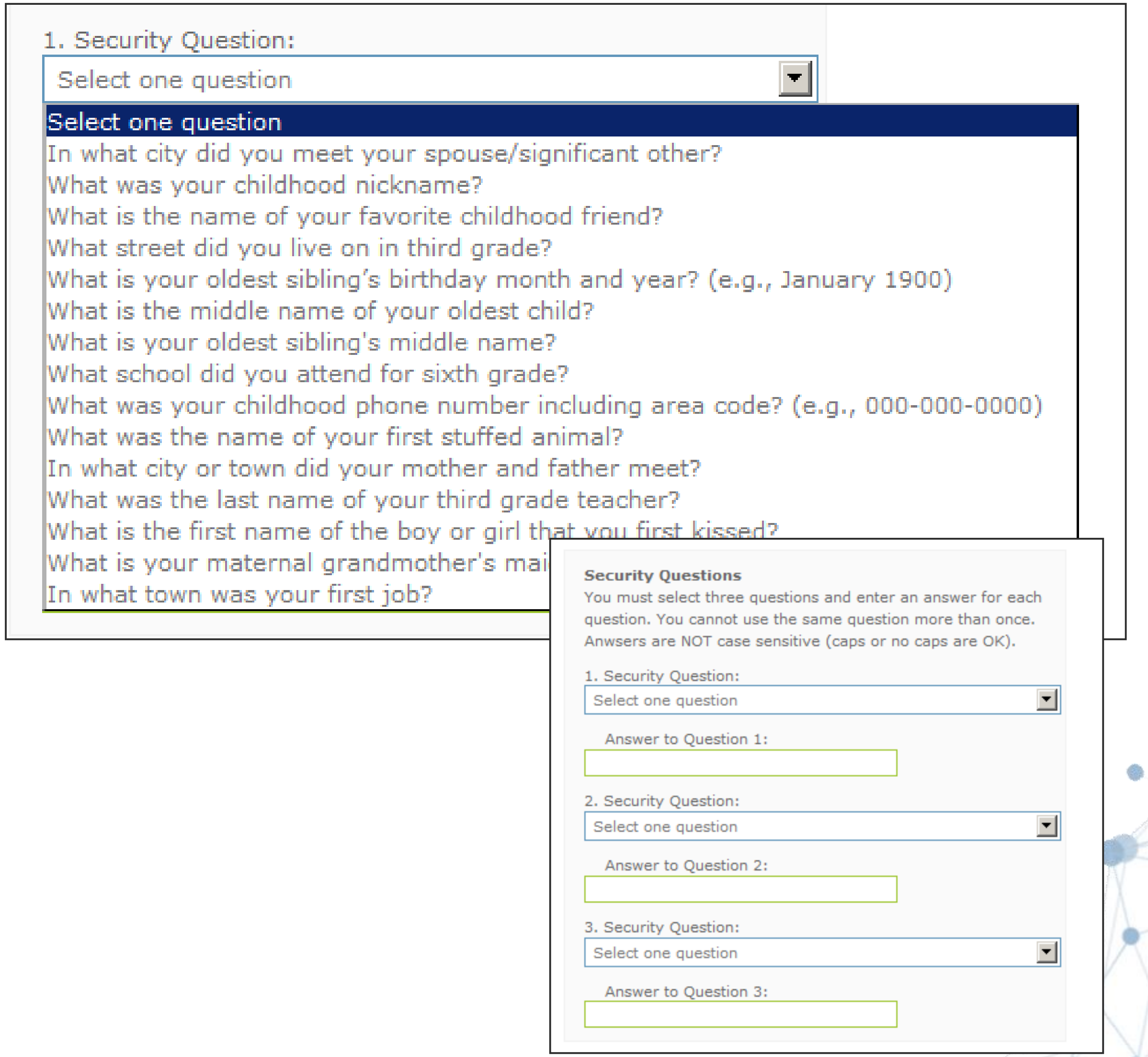
□ Remember my email address

Comment verification:

Please enter the text you see inside the graphic to post your comment:

Anti-Automation Tools
Security Questions

- During the user registration process, the user must answer a set of challenge-response questions that can be then asked of someone claiming to be a specific user
- Security questions are used to prove the identity based on something known only to the user (Knowledge-based Authentication [KBA])
  - Criminals often have access to databases filled with responses to common security questions (mother's maiden name, amount of mortgage payment, etc.)
  - These challenge-response questions have marginal value proving user identity given the number of data breaches
  - They are useful as an anti-automation tool



1. Security Question:

Select one question

Select one question
In what city did you meet your spouse/significant other?
What was your childhood nickname?
What is the name of your favorite childhood friend?
What street did you live on in third grade?
What is your oldest sibling's birthday month and year? (e.g., January 1900)
What is the middle name of your oldest child?
What is your oldest sibling's middle name?
What school did you attend for sixth grade?
What was your childhood phone number including area code? (e.g., 000-000-0000)
What was the name of your first stuffed animal?
In what city or town did your mother and father meet?
What was the last name of your third grade teacher?
What is the first name of the boy or girl that you first kissed?
What is your maternal grandmother's mai
In what town was your first job?

Security Questions

You must select three questions and enter an answer for each question. You cannot use the same question more than once. Anwsers are NOT case sensitive (caps or no caps are OK).

1. Security Question: Select one question

Answer to Question 1:

2. Security Question: Select one question

Answer to Question 2:

3. Security Question: Select one question

Answer to Question 3:

Source: http://goodsecurityquestions.com/designing.htm

Anti-Automation Tools
Security Questions

- Do not let users create their own secret questions because they will generally create bad ones
- The site itself should define a series of secret questions from which the user can **choose**
  - Let the user answer 3 or more secret questions at the time of account registration
    - These questions and answers can then be used as a second channel of identity verification
    - Do not echo the questions and answers in the same session for confirmation
      - During registration either the question or answer may be echoed back to the client, but never both
  - Multiple questions add a higher degree of confidence to the verification process
    - Adds randomness (don’t always display the same question during a password reset)
    - Provides a bit of redundancy in case the legitimate user forgets an answer – they can select a different question

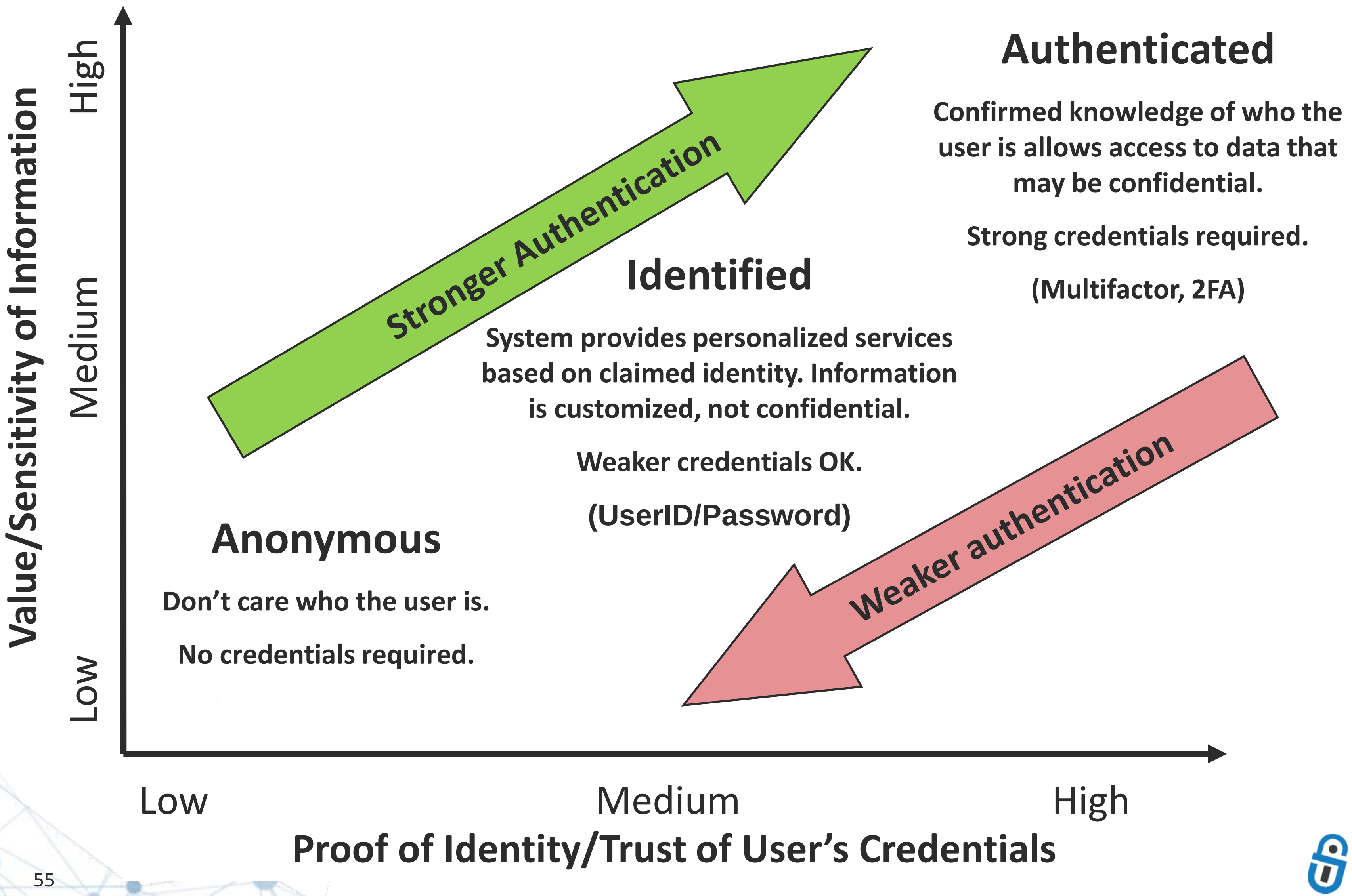
Anti-Automation Tools
Qualities of Better Security Questions

- Cannot be easily guessed or researched/discovered (safe)
  - The answer cannot be found through research (mother's maiden name, birth date, first or last name, social security number, phone number, address, pet's name)
  - The question has many possible answers where the probability of guessing the correct answer is low
  - Answers are unlikely to be known by others such as a family member, close friend, relative, ex-spouse, or significant other
- Don't change over time (stable)
  - "Favorites" change over time
- Are memorable
  - Easy to remember but still not available to others
  - The user should immediately know the answer without doing research or looking up a reference
- Are definitive or simple
  - Definitive or simple; answer has an obvious format; answer is NOT case sensitive
- Do not violate privacy issues
  - Your mother's maiden name probably fails this test
- Have high entropy (many possible responses)
  - Your favorite color probably fails this test, unless your favorite color is "yellowish-brown light chartreuse"
- Are not obvious or stupid
  - "How do you spell 'password'"?
- Won't embarrass a human help desk worker that is trying to authenticate the user
  - "Who did you sleep with at the Halloween party?"

Anti-Automation Tools
Security Questions

- Storing the security question responses
  - As plain text in the database (NO! do not do this!)
    - A database breach will immediately reveal the responses
    - This could also put at risk any other (unrelated) applications that depend on the same secret questions
  - Store the responses as you would store any valuable secret
    - As a secure hash (using a password hashing algorithm and cryptographically random salt for each user)
    - Use a different salt for the security questions and for the password hash
- When there is a legitimate reason to make the answer visible in plain text…
  - A typical scenario is when a human operator is verifying an identity over the telephone
    - Responses are hashed – unable to view the response in plain text (preferred solution)
      - Call center personnel will enter the answer the customer provides
        - Be careful of mixed case
        - It may be a good idea to convert to all upper or all lower case before hashing (loses entropy, however)
        - If hashes match, then identity is authenticated
    - Symmetric encryption is reversible and allows the response to be decrypted
      - There are key handling issues that need to be addressed if encryption Is used

Weak vs. Strong Authentication



High
Medium
Low
Value/Sensitivity of Information
Low
Medium
High
Proof of Identity/Trust of User's Credentials
Stronger Authentication
Weaker authentication

Authenticated

Confirmed knowledge of who the user is allows access to data that may be confidential.

Strong credentials required.

(Multifactor, 2FA)

Identified

System provides personalized services based on claimed identity. Information is customized, not confidential.

Weaker credentials OK.

(UserID/Password)

Anonymous

Don't care who the user is.

No credentials required.

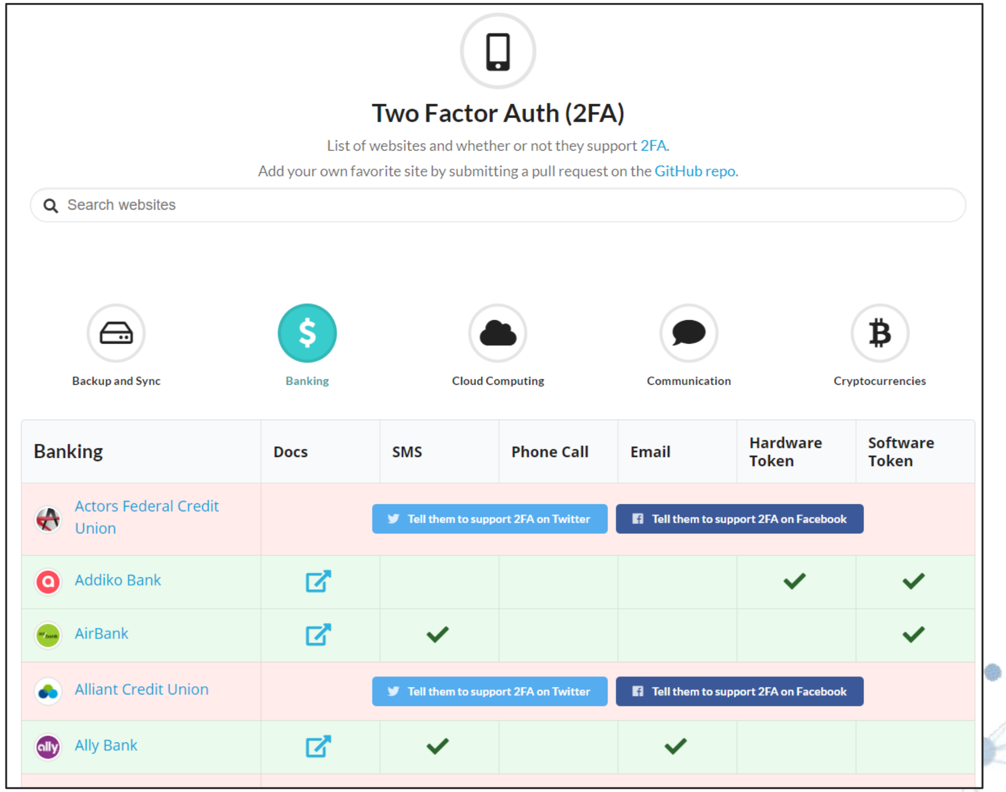
Weak Authentication
Authentication techniques that are **easier** to defeat
Used where data confidentiality is not important
Static password

Strong Authentication
Authentication techniques that are **harder** to defeat
Used where data confidentiality is important
2FA or multifactor authentication

Multifactor Authentication (MFA)

- Multifactor authentication uses **two or more authentication factors** to prove the claimed identity

  - **Something you know** (a secret: typically a password, PIN, answer to security question)
  - **Something you have** (typically a token or other serialized hardware device)
    - The oldest form of access control is a key (an access token)
    - A physical token is used to authenticate the token-holder since we assume that the serialized token is in the control of the legitimate owner
    - Software tokens are also commonly used
  - **Something you are** (typically a biometric – fingerprint, retina scan, etc.)
  - **Where you are located**
- 2FA: Two-factor authentication



Two Factor Auth (2FA)

List of websites and whether or not they support 2FA.

Add your own favorite site by submitting a pull request on the GitHub repo.

Search websites

Banking	Docs	SMS	Phone Call	Email	Hardware Token	Software Token
Actors Federal Credit Union		Tell them to support 2FA on Twitter		Tell them to support 2FA on Facebook		
Addiko Bank	↗				✓	✓
AirBank	↗	✓				✓
Alliant Credit Union		Tell them to support 2FA on Twitter		Tell them to support 2FA on Facebook		
Ally Bank	↗	✓		✓		

https://twofactorauth.org/

FIDO Universal 2nd Factor (U2F) Protocol

- U2F is an open authentication standard that enables internet users to securely access any number of online services with a hardware security token without drivers or client software
- The (USB, BLE, NFC) tokens communicate with the host computer using the HID protocol which mimics a keyboard
  - Allows the application software to directly access the security features of the device without user effort other than possessing and inserting the device
- Once communication is established, the application exercises a challenge–response authentication with the device using public-key cryptography methods and a secret unique device key manufactured into the device

One-Time Passwords (OTP)

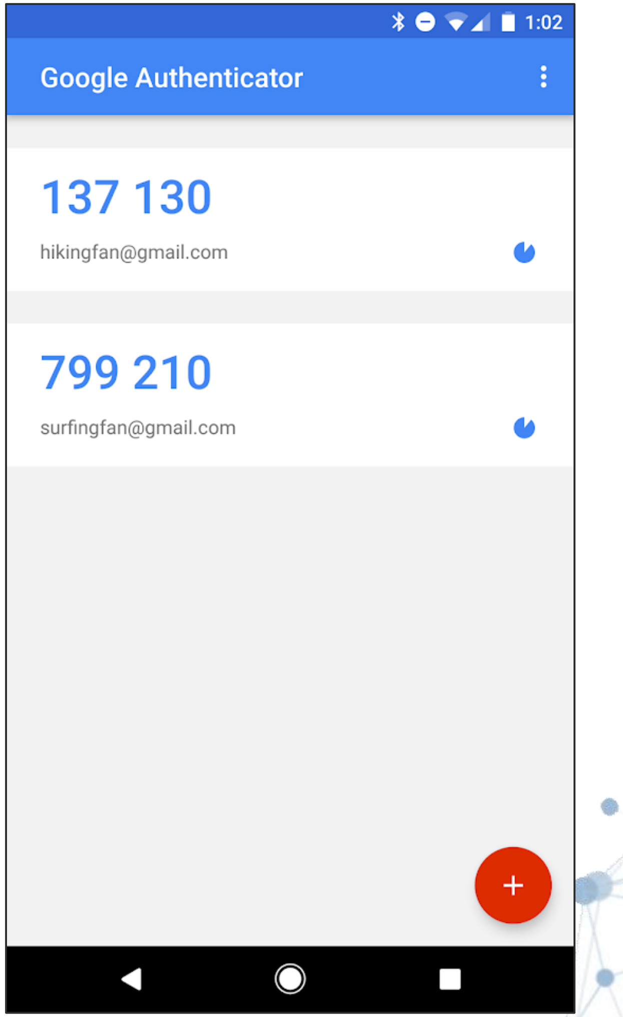
- OTP are more secure than traditional static (unchanging) passwords since the password keeps changing
  - It isn’t vulnerable to replay attacks
- Authentication tokens can be used to implement one-time passwords
  - Hardware tokens
    - Yubikey
  - Software tokens
    - Google authenticator
    - OTPs generated by a smart phone app are more secure than SMS-based authentication because they are not vulnerable to SIM-swapping attacks

HMAC-based One-Time Password (HOTP)

- The HOTP algorithm is an **event-based** OTP algorithm
  - It uses a shared secret and a counter that changes with each request for a new password
- An HMAC-SHA-1 hash of a counter is generated using a shared secret
- When a new OTP is generated, the counter is incremented so that generated passwords will be different each time
- An HOTP password could potentially be valid for a “long time” which could allow and attacker to use a compromised password at their leisure

Time-based One-Time Passcodes (TOTP)

- TOTP is a **time-based** OTP algorithm
  - A one-time password is computed from a shared secret key and the current time
  - An HMAC-SHA-1 hash of the elapsed time since an arbitrary epoch is generated using a shared secret
- A user enters username and password into a website or other server
  - The server generates a secret key which the user enters on to their TOTP application on a smartphone or other device (often by scanning a QR code)
    - To verify that process worked, the user application immediately generates a one-time password to be checked by the server
- On subsequent authentications, the user enters their username, password and the current one-time password
  - The server checks the username and password as normal then also runs TOTP to verify the entered one-time password
- The TOTP passwords keep changing and are only valid for a short period of time (typically 30-60 seconds)



Google Authenticator

137 130

hikingfan@gmail.com

799 210

surfingfan@gmail.com

https://play.google.com/store/apps/details?id=com.google.android.apps.authenticator2&hl=en

OAuth (Open Authorization)

- OAuth is an open standard for token-based authorization on the Internet
- Lets the user authorize one website (the consumer) to access user data from another website (the provider) without exposing the user's password
- OAuth acts as an intermediary on behalf of the end user
  - Provides access to a service with an access token that **authorizes** specific account information to be shared
  - Ensures third-party websites have the right permissions to access a user’s information
- OAuth is about **authorization** (permissions/access control)

OpenID Connect

- OpenID Connect is an interoperable **authentication** protocol based on OAuth 2.0
  - Allows developers authenticate their users across websites and apps without having to own and manage password files
  - Answers the question: “What is the identity of the person currently using the browser or native app that is connected to me?”
- Gives the user one login for multiple sites
- Open ID Connect is about **authentication**

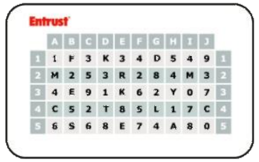
WebAuthn (Web Authentication)

- WebAuthn is a credential management API built into web browsers
- Users can register and authenticate with web applications using an authenticator such as a phone or hardware security token, replacing traditional passwords
    - Strong cryptography is used to authenticate a user by proving possession of a corresponding private key
        - A challenge is signed with the private key, proving the identity of the keyholder

Examples of Strong Authentication Solutions

Machine Auth

Authorized set of workstations



Grid Auth

Grid location challenge and response

Digital Certificate

Knowledge Auth

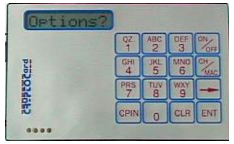
Challenge / response questions

Scratch Pad Auth

One-time password list

One-time Password

Smart Card



Challenge-Response Token

Out-of-Band

TOTP or one-time-passcode to mobile device

- Keep these strong authentication options in mind as you design systems

SMS-Based Authentication is Relatively Weak

- It is fairly easy to highjack a mobile account (SIM Swap Attack)
  - https://krebsonsecurity.com/2018/08/hanging-up-on-mobile-in-the-name-of-security/
- The target’s mobile provider is tricked into transferring the phone number to a device or account under the attacker’s control
  - The attacker intercepts the password reset link or authentication code sent via SMS
- SMS-based authentication is no longer recommended by NIST
  - But SMS-based authentication still provides better security than just a static password

> “ ‘…SMS-based authentication is not nearly as secure as we would hope, and the main attack was via SMS intercept,’ Reddit disclosed.” Reddit Breach Highlights Limits of SMS-Based Authentication (01 Aug 2018)

Practices for Secure Authentication – 1

- Do not put UserID, passwords (or other sensitive information) in…
  - URLs
  - Cookies (unless encrypted or hashed)
  - Hidden variables (unless encrypted or hashed)
- Be sure to have a session time-out (a short period of inactivity)
  - Automatically logout the user
- Lock out the account after <some small number of> consecutive failed login attempts
  - Ideally, the account should remain locked out until reset by an administrator
  - If this is not workable, then lockout the account for at least 1 hour to slow down brute force attacks
- Don’t allow unencrypted credentials
  - Do not just use HTTPS for login, and then do subsequent requests with HTTP
  - Use HTTPS everywhere!

Practices for Secure Authentication – 2

- Implement a logout button that is easy to find on all pages
  - Make it easy for people to log out
  - Destroy client- and server-side state variables
- Don’t create “Remember me” functionality
- Don’t create password reminder functionality – just reset the password
- Don’t allow weak passwords
  - Longer passwords are better
  - 20 or more characters are recommended
    - 8 character passwords are too short for most applications
  - Use passphrases

Recap

- Identification and Authentication
- Common Password Attacks
  - Passive Online Attacks
  - Active Online Attacks
  - Offline Attacks
  - Non-Electronic Attacks

- Secure Authentication Practices
  - Anti-Automation Tools
    - CAPTCHA
    - Security Questions
  - Stronger Authentication
    - Multifactor Authentication (MFA)
    - FIDO Universal 2nd Factor (U2F) Protocol
    - One-Time Passwords (OTP)
    - OAuth (Open Authorization)
    - OpenID Connect

Contact Information

Danny Harris

Senior Security Consultant and Instructor

Security Innovation

dharris@securityinnovation.com

http://www.securityinnovation.com