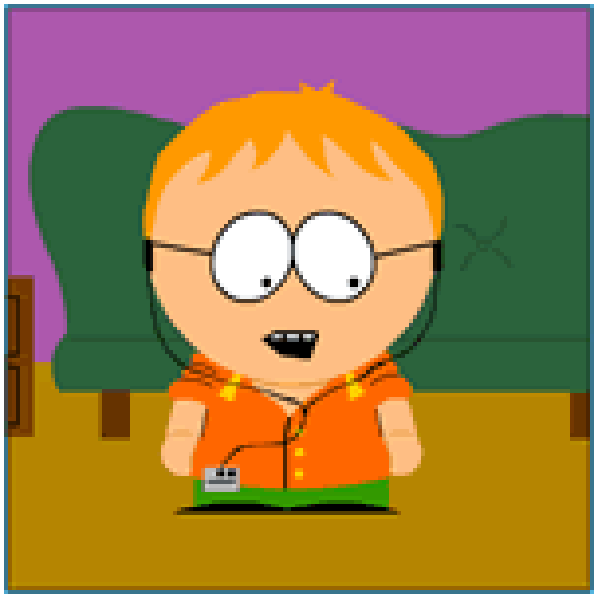
2FA War Stories

David Robinson
ChCon October 2017

whoami

- Dave, Karit, @nzkarit
- Security Consultant at ZX Security
- Enjoy Radio Stuff
- Enjoy Picking Locks and other physical security things



Today

- What is 2FA
- Why should you use it
- Some stories of some implementation issues

What is 2FA

- Two Factor Authentication
  - A - Something you know – Password
  - B - Something you have – Token, Card, Phone, etc
  - B - Something you are – Biometric
- Most common password and something you have
- Biometric issue
  - Hard to change your fingerprints or iris when there is a data breach

Why do we need 2FA?

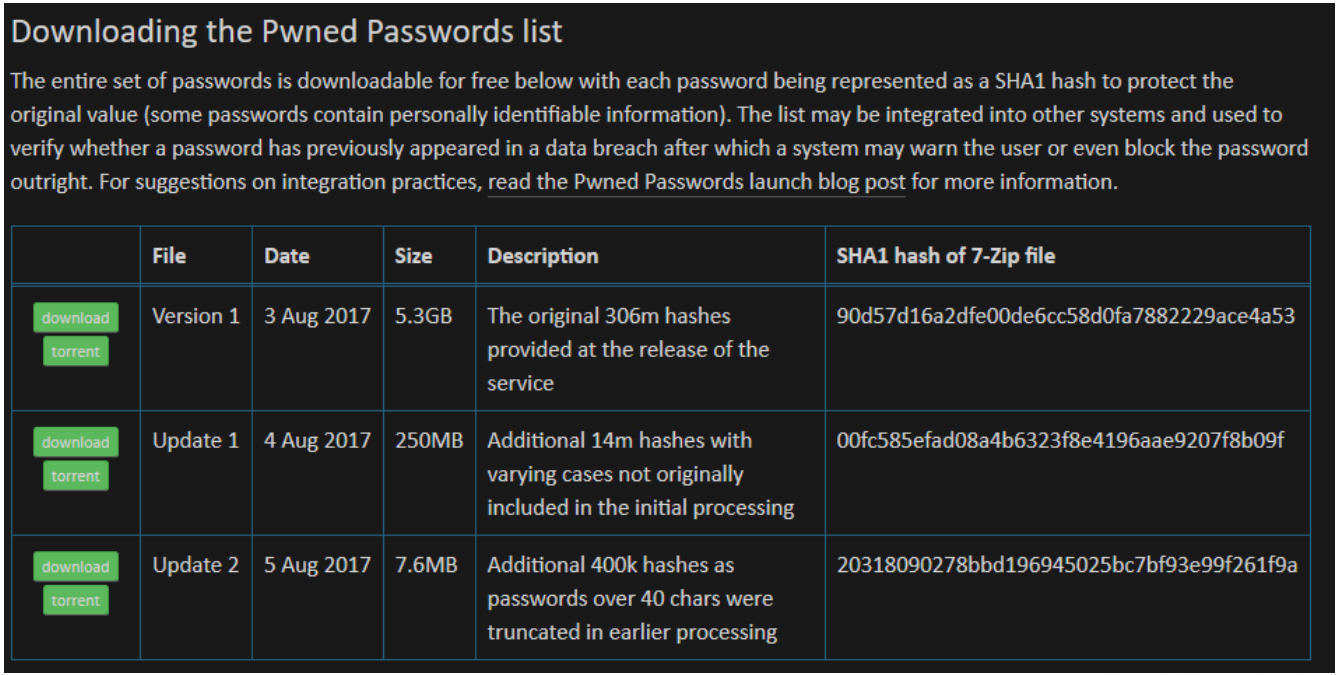
- Password Reuse
  - People use the same password on multiple sites
- During 2016 there was cases of breaches used to get into other sites
  - Logmein forced reset after LinkedIn, MySpace & Tumblr

Don’t allow passwords from breach corpus

- NIST gone as far as saying:
  - 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.

Need a list of passwords from breaches?

- https://haveibeenpwned.com/Passwords
- Has 5.5ish GB worth



Downloading the Pwned Passwords list

The entire set of passwords is downloadable for free below with each password being represented as a SHA1 hash to protect the original value (some passwords contain personally identifiable information). The list may be integrated into other systems and used to verify whether a password has previously appeared in a data breach after which a system may warn the user or even block the password outright. For suggestions on integration practices, read the Pwned Passwords launch blog post for more information.

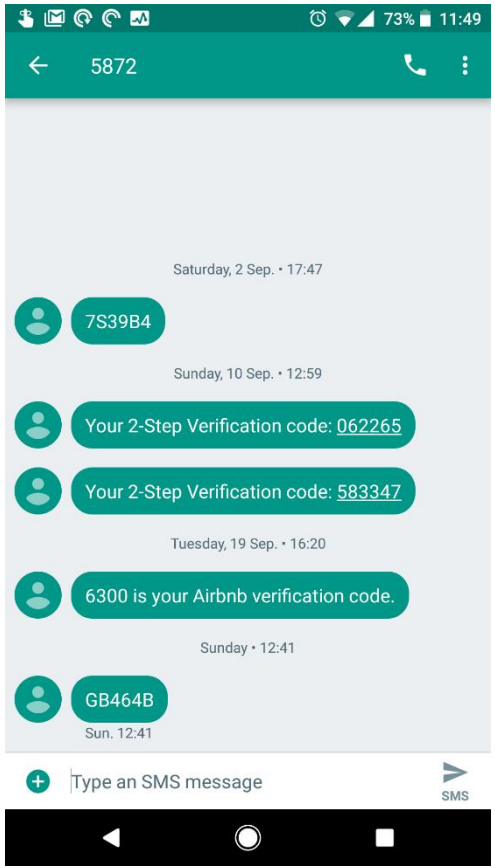
	File	Date	Size	Description	SHA1 hash of 7-Zip file
download torrent	Version 1	3 Aug 2017	5.3GB	The original 306m hashes provided at the release of the service	90d57d16a2dfe00de6cc58d0fa7882229ace4a53
download torrent	Update 1	4 Aug 2017	250MB	Additional 14m hashes with varying cases not originally included in the initial processing	00fc585efad08a4b6323f8e4196aae9207f8b09f
download torrent	Update 2	5 Aug 2017	7.6MB	Additional 400k hashes as passwords over 40 chars were truncated in earlier processing	20318090278bbd196945025bc7bf93e99f261f9a

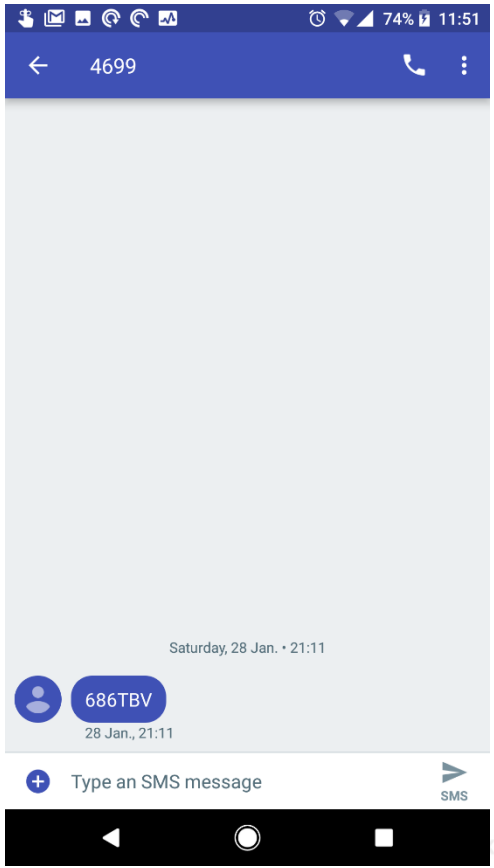
Why are breaches an issue?

- Why does a breach of another site affect my site?
- Users reuse passwords between sites
- Other sites don't securely store passwords correctly. They:
  - Don't hash
  - Don't salt or use the same salt
  - Don't use a password hashing algorithm
- So are "easily" crackable
- If storing passwords follow the OWASP Cheat Sheet on password storage
  - https://www.owasp.org/index.php/Password_Storage_Cheat_Sheet
  - https://www.owasp.org/index.php/Authentication_Cheat_Sheet

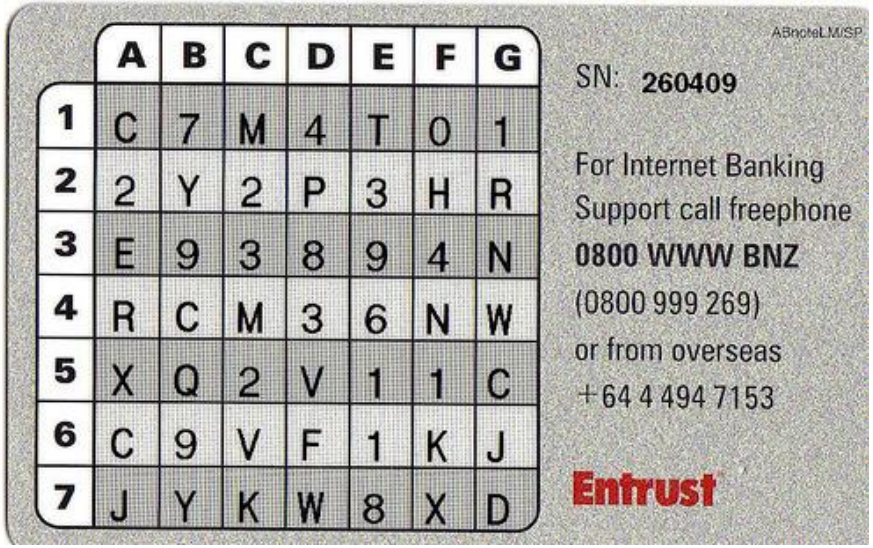
So what methods do we have for 2FA?

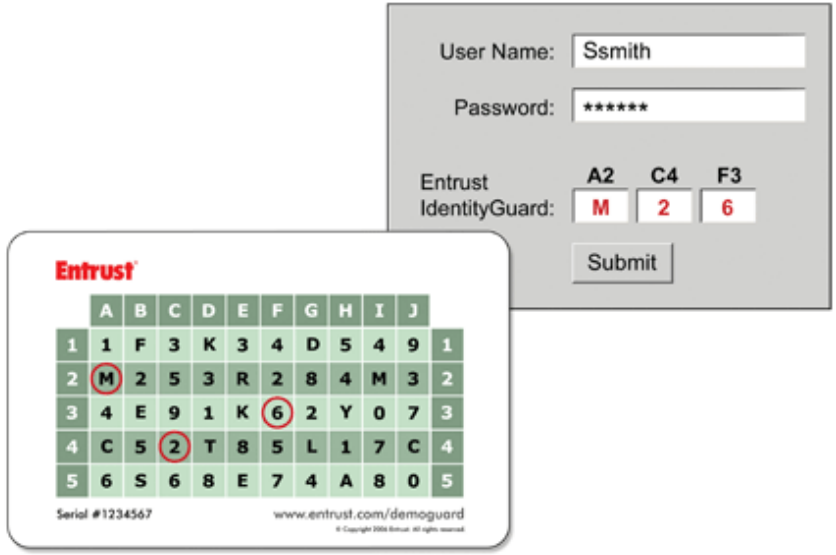
SMS



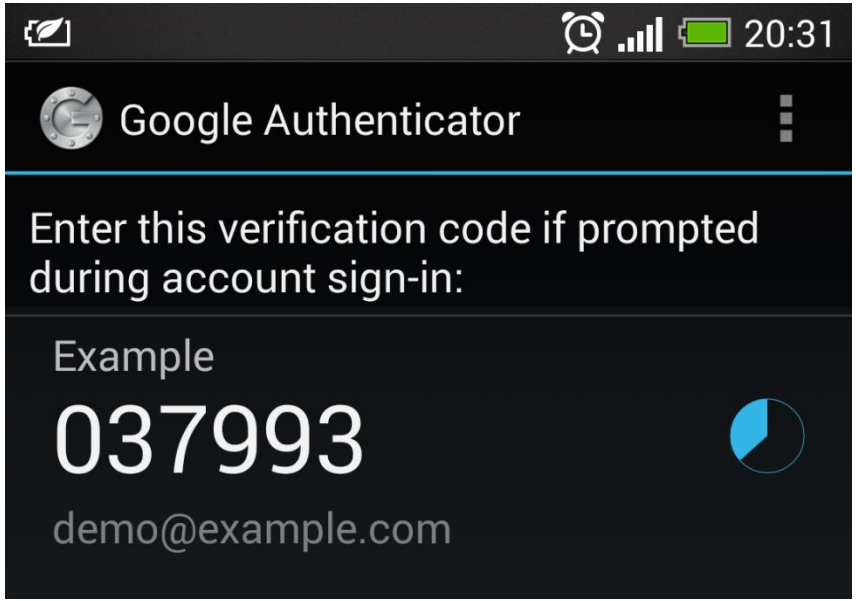


Battleship Cards





Time Based One Time Passwords (TOTP)



HMAC based One Time Passwords (HTOP)

- Looks same as TOTP

Universal Second Factor (U2F)

2FA Method Weaknesses

SMS

- User has a phone number associated with account
- Text them a code when they login to type in
- Pros:
  - Cellco worries about device enrolment, lost phone, etc
- Cons:
  - Have to pay for each text or block of texts
  - Text messages don’t roam well through text message gatways
  - Cellco responsible for your security

SMS Roaming

- What do you think the Aussie Govt recommends for travellers?
- https://my.gov.au/mygov/content/html/securitycodes.html

Aussie – myGov

Travelling overseas with myGov security codes

If you are planning on travelling overseas you will still be able to receive a code as long as:

- you take your registered mobile phone number with you
- your telephone service provider has enabled you to receive SMS while outside of Australia
- you are connected to a mobile phone network that your telephone service provider is compatible with

If you will not be able to receive myGov security codes overseas, consider switching them off before your departure. Access to your account will not be possible if you are unable to receive the SMS myGov sends you when you sign in.

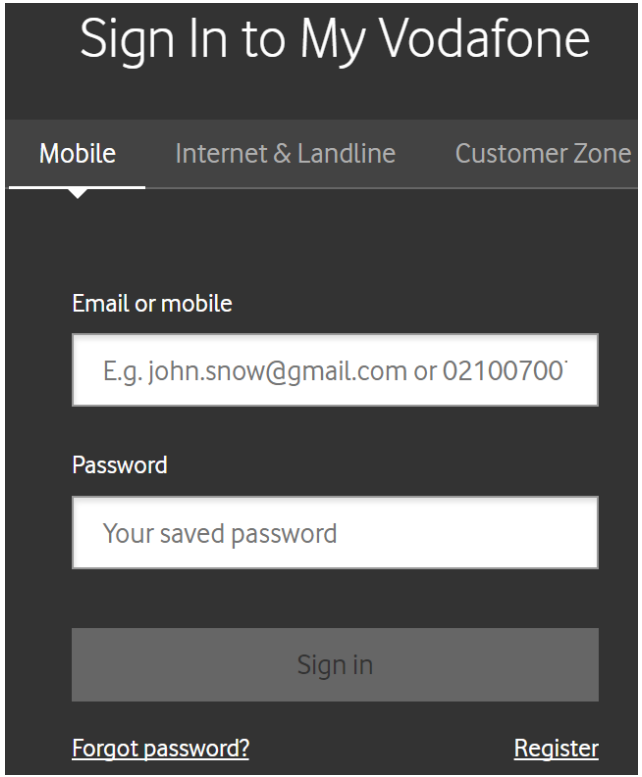
Cellco responsible for you security

- SIM Card Swap
  - Did try to social engineer but Spark, Vodafone and 2 Degrees all asked for photo ID
  - But you know that thing about password reuse? Well Vodafone hasn't got that message

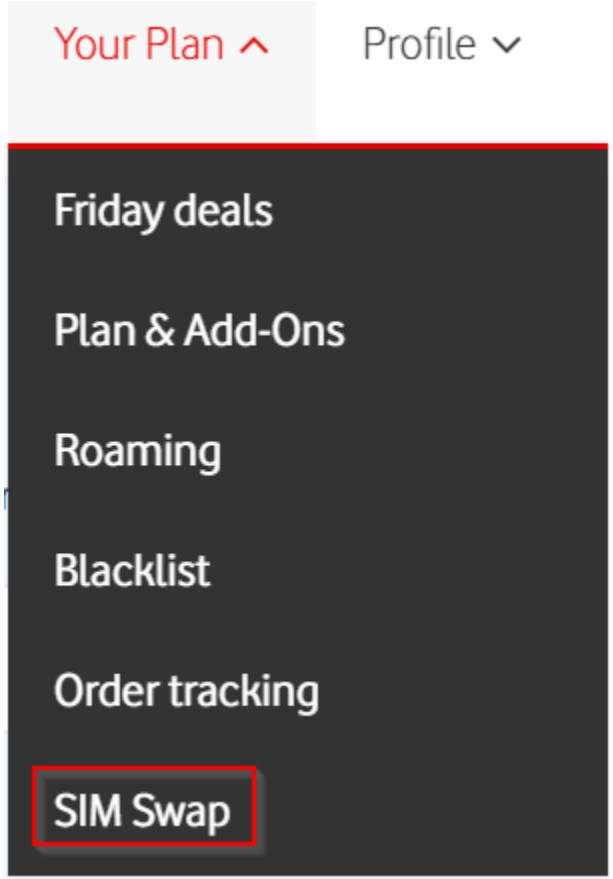
- In past have had luck with no photo ID
- It is policy to ask for photo ID but depends on the person

SIM Swap Process #1

- Target uses Vodafone
- They use the same password on site and Vodafone
- Vodafone has no 2FA on their system



SIM Swap Process #2



This is a 3-step process, taking a few minutes, where you will enter your SIM swap details; confirm they are correct; then follow onscreen instructions for completing the swap.

Enter your details

[redacted] - Prepay Pay & Go

This is the number you want to move to your new SIM card

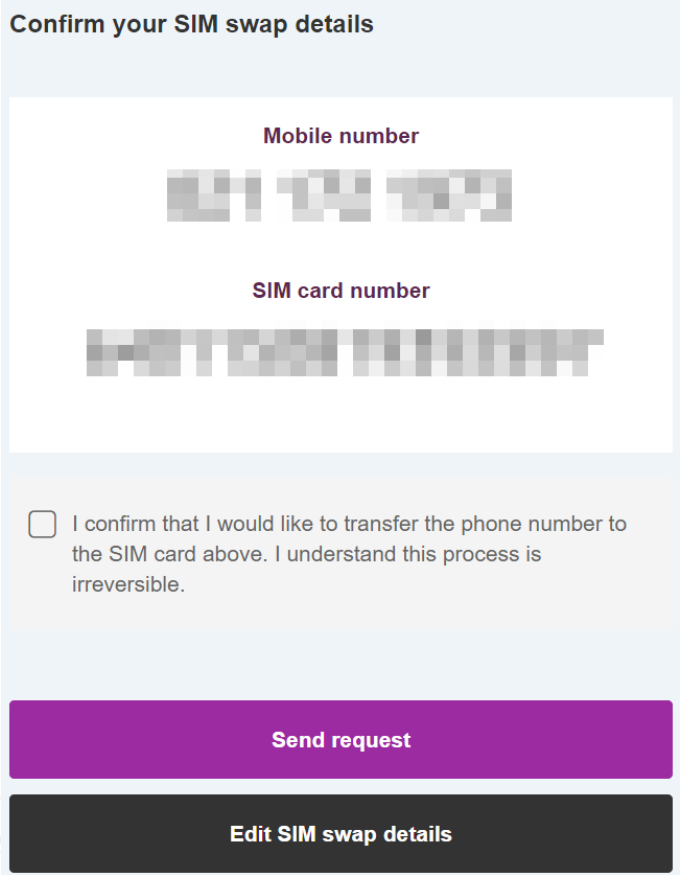
[redacted]

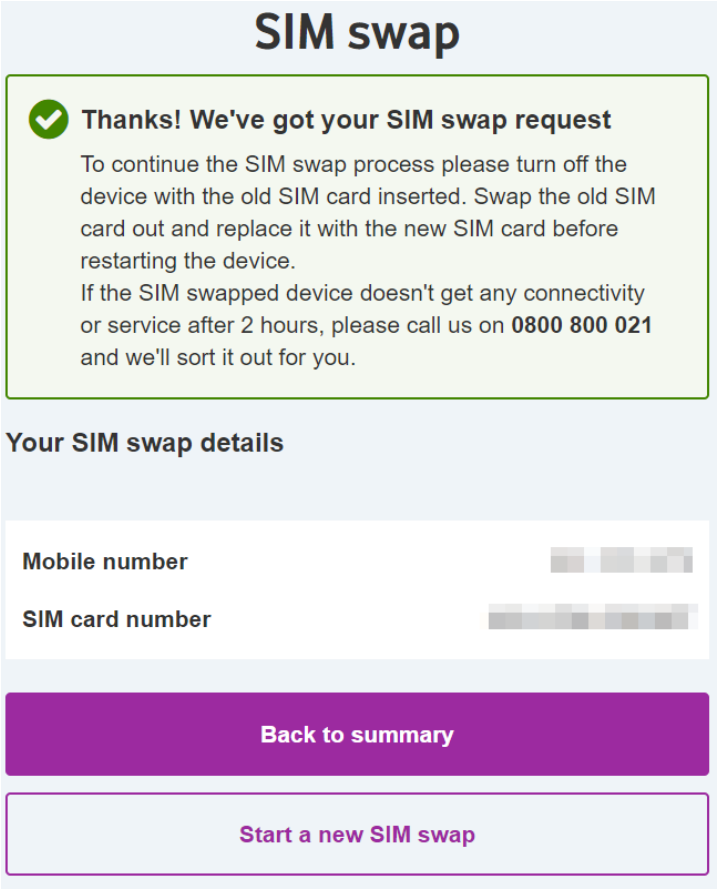
This is the 16 digit number on the back of your SIM card starting with 6401

[redacted]

Next

SIM Swap Process #3





SS7

Real-World SS7 Attack — Hackers Are Stealing Money From Bank Accounts

Wednesday, May 03, 2017 Swati Khandelwal

Tweet Share Share 44 Share Share Share

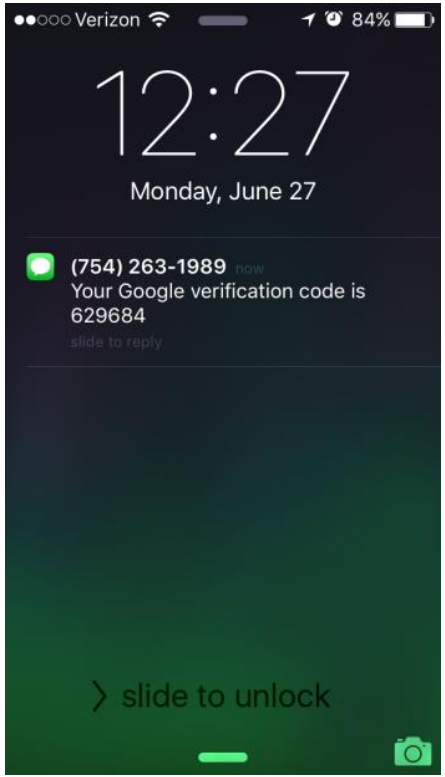
Security

After years of warnings, mobile network hackers exploit SS7 flaws to drain bank accounts

O2 confirms online thefts using stolen 2FA SMS codes

By Iain Thomson in San Francisco 3 May 2017 at 20:02 48 SHARE ▼

Or phone will just display to everyone



Issues with Battleship Cards

- People are worried about losing them, so they make copies
- During Red Teams and Pen Tests, search for files that in file name contain:
  - access card
  - <battleship card vendor name>
  - Etc
- Collect images, docs, spreadsheets, etc containing the card
  - Maybe next to passwords.txt
  - (I have come across very few orgs that have a Password Manager installed as standard)

TOTP


Device Status
Live Video
Preset
Set 1 Go
Resolution 640*480
Mode Outdoor
Brightness 9
Contrast 4
refresh camera params
refresh video
Snapshot
Device Management


RSA
SecurID

U2F

- Uses a hardware Security Module, hard to clone
- Hard for user to misuse
- Has proof of presence

- Chrome Supports
- Firefox – currently in Nightly builds

Implementing 2FA

- Some stories to learn from

Think about new phones

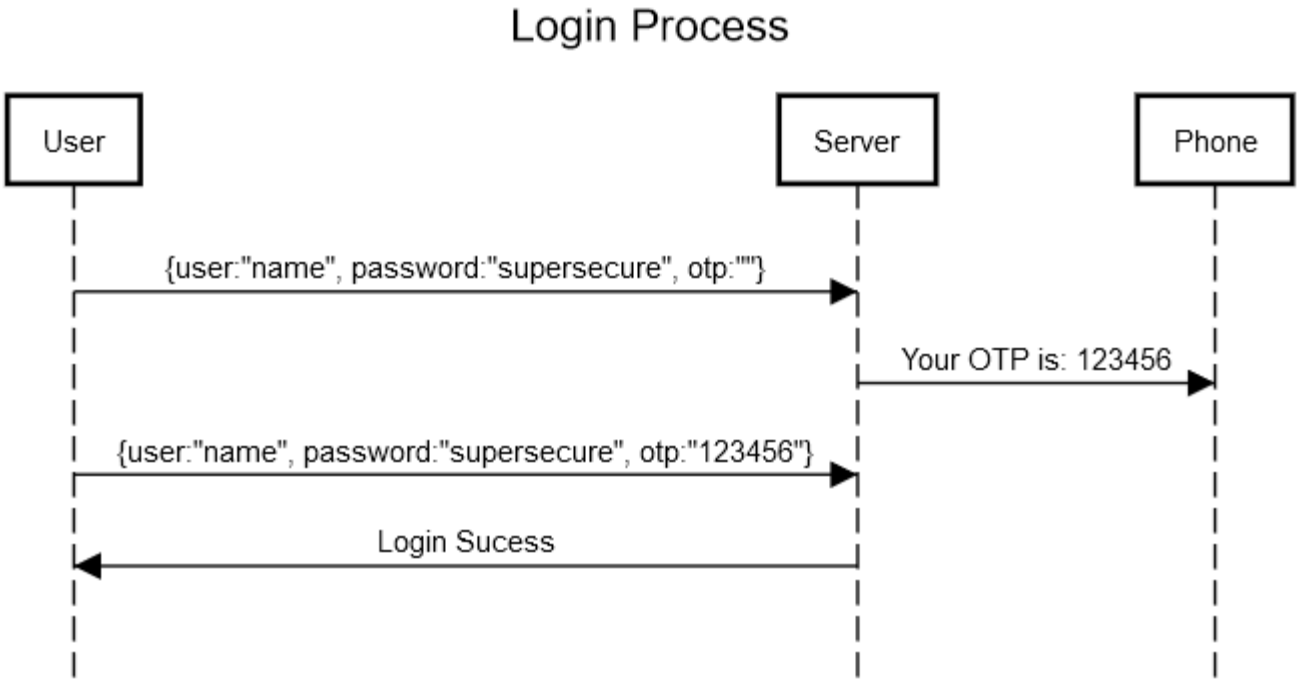
- Sites that don’t support multiple 2FA methods on one account
- Sometimes have to turn 2FA off to set up a new phone with TOTP

Takeaway – Multiple Phones

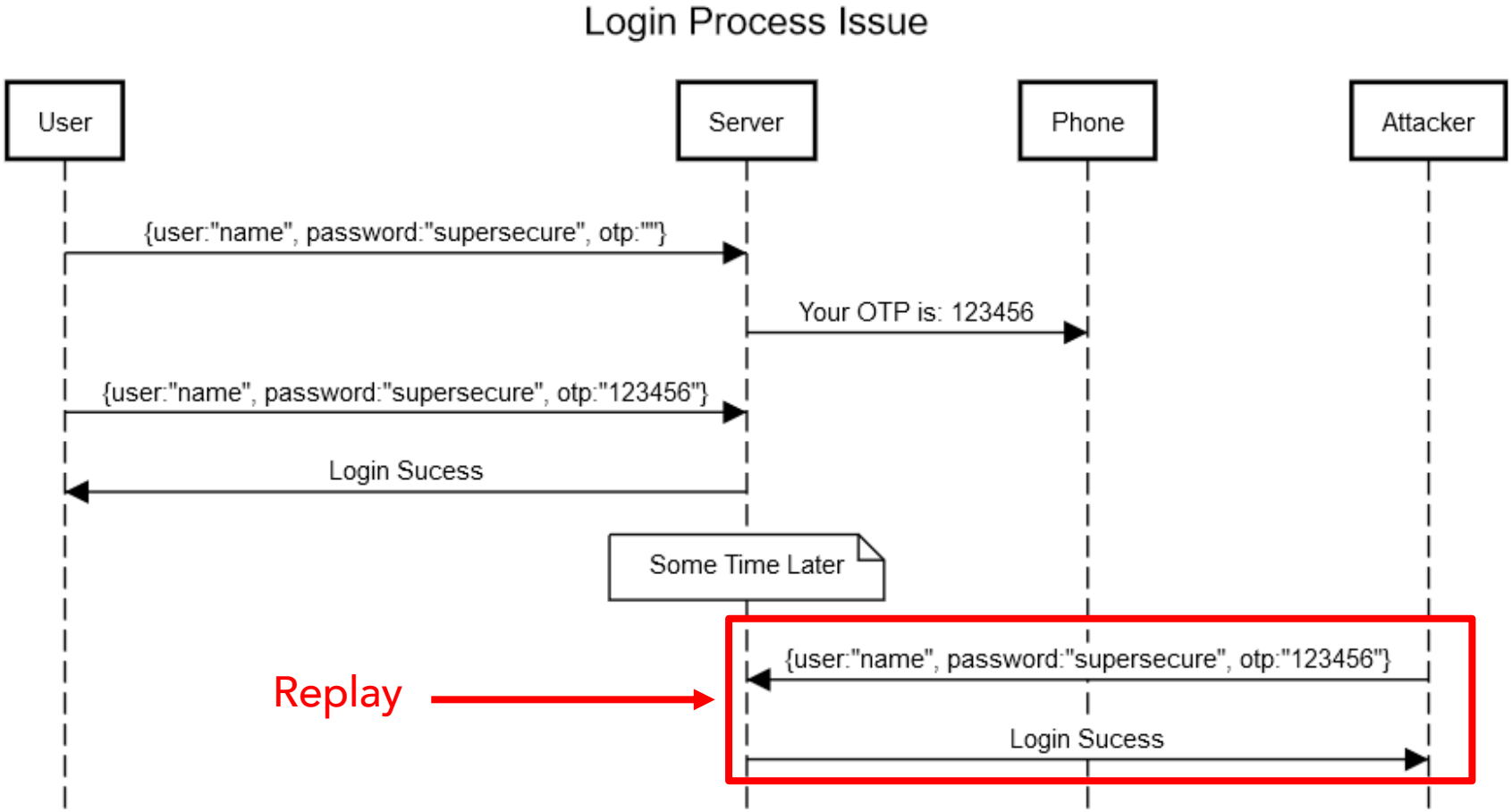
- Allow a user to have more than one TOTP instance/device authorised against the account
- User can subscribe their new phone without disabling 2FA
- Keep their old phone as a backup
- Enrol 2+ U2F tokens

- Though, user does need to be able to revoke a TOTP or U2F

The “ONE” in OTP is important



Because


Login Process Issue
User
Server
Phone
Attacker
{user:"name", password:"supersecure", otp:""}
Your OTP is: 123456
{user:"name", password:"supersecure", otp:"123456"}
Login Sucess
Some Time Later
Replay
{user:"name", password:"supersecure", otp:"123456"}
Login Sucess

Take away – Only use Once

- Expire the OTP after use

Don’t assume time only moves forward

- Some of the TOTP libraries assume time only moves forward
- If the attacker can change time on the server, they can reuse code
- Ensure that library blocks reuse of the TOTP’s value

- Refer to my Defcon talk for more info
  - https://zxsecurity.co.nz/presentations/201707_Defcon-ZXSecurity-GPSSpoofing.pdf

Reuse – Plugins

- Plugins for off the shelf products

Library	Default No Reuse	No Default	Default Reuse
Google Auth libpam		X	
Two Factor Authentication (Wordpress Plugin)	X		
OATHAuth (MediaWiki Plugin)	X		

Reuse - Libraries

- Libraries which you can include in your own code
- Support is a method that does verify with prior context

Library	Support	No Support
Github - pyotp/pyotp		X
Github - mdp/rotp	X	
Github - Spomky-Labs/otphp		X
Github - pquerna/otp		X

Takeaway - Reuse

- Ensure the library being used does noit allow reuse of tokens
  - Store last login time
  - Ensure current_time >= last_login_time + 30seconds

Get the SQL Correct

- SELECT * FROM otp WHERE otp = ?;
- See the problem?

Should really be

```
SELECT *
    FROM otp
    WHERE username = ?
        AND otp = ?
        AND used = FALSE -- if already used the code don't allow used again
        AND current_time() < (generated_time + 5min); -- if the code is more than 5 min
old, send new one
```

Takeaway - SQL

- Ensure that:
  - You match the user with the generated code
  - OTP hasn't been used before
  - OTP isn't too old

SQLi

username	password_hash	totp_seed
some_cool_username	$argon2i$v=19$m=65536,t=2,p=4$c29tZXNhbHQ$RdescudvJCsgt3ub+b+dWRWJTmaaJObG	0123456789ABCDEF0123456789ABCDEF

SQLi Issue

- If I can SQLi the password field I can also SQLi the seed value

Similar

- If user can view the seed after creation
- Or as an admin also an issue
- If attacker can gain access to an existing session they can see the TOTP seed

Take away – SQLi issue

- Consider how the TOTP seeds stored and accessed
- Maybe:
  - Store in a separate table
  - Don’t give the webapp db account access to this table
    - Create a function that returns yes or no which the webapp can access
  - That way attacker can not use SQLi to get the TOTP seed
- Only show seed during TOTP signup

Email is not 2FA

- We have seen organisation who use email as 2FA
- Just so much wrong:
  - Email Clear Text
  - Password Reuse
  - Spam filters

- If an attacker has access to email through password reuse
  - Not going to be having a good day

Takeaway – Email

- Don’t use email for 2FA

the grugq
@thegrugq

Use Tor. Use Signal.

Use it everywhere

- 2FA on Webmail
- But no 2FA on VPN
  - When on VPN can access email without 2FA

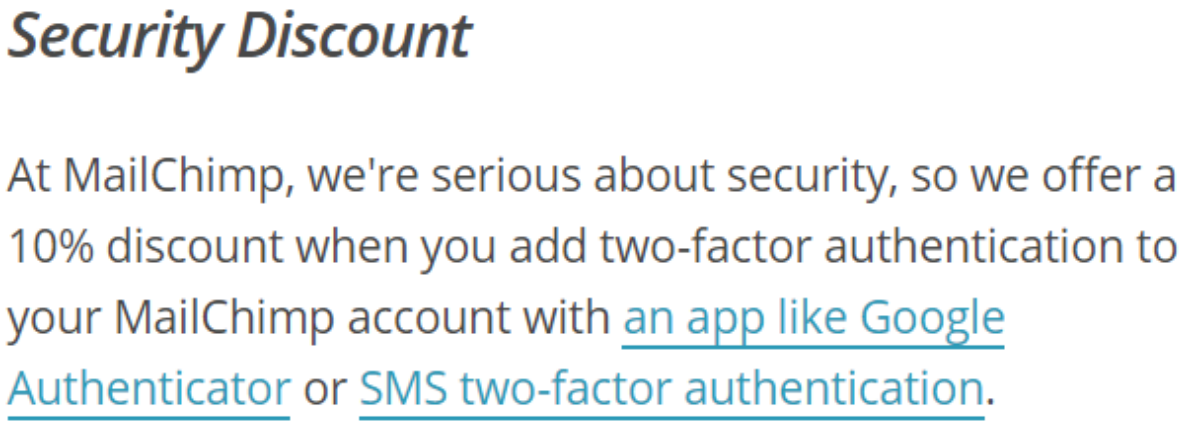
- Also the inverse
- 2FA on VPN but no 2FA on Webmail
  - Then Ruler tool which allows to write Exchange rules to give you reverse shells
  - https://github.com/sensepost/ruler

Takeaway – Use everywhere

- 2FA all access methods

How to encourage users

- Force it on them
  - Enterprise can do
  - User space not so much
- Educate users why they should enable it
- Mailchimp offer 10% discount



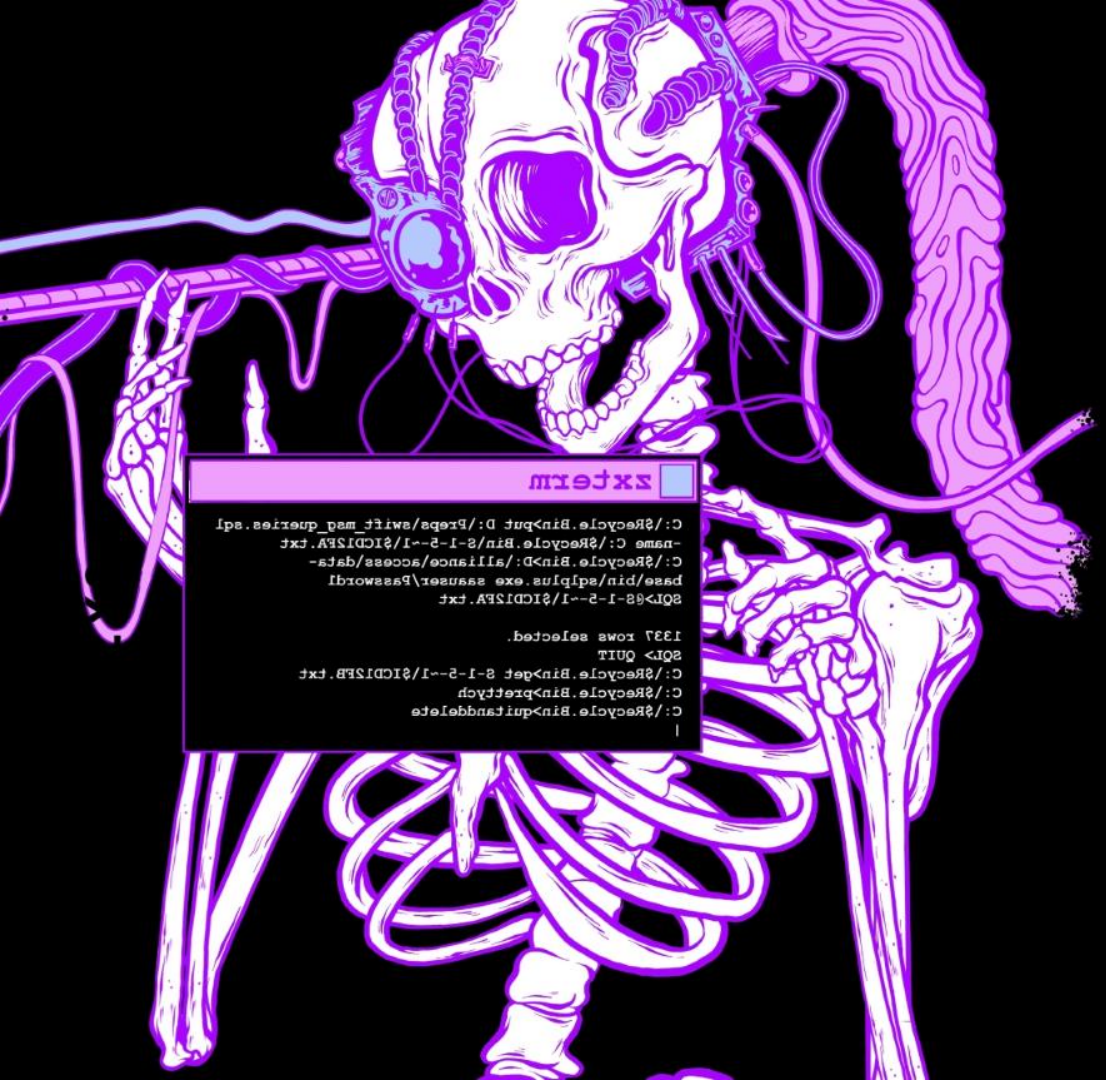
Wrapup

- Please 2FA everything
- Please don’t use SMS
- My preference U2F followed by TOTP
- Also WebAuthN on horizon which also looks interesting
    - Currently going through standardisation

Shameless Plug

- A couple of my colleagues will be doing Bus Factor live on location
  - Outside starting nowish

- https://thebusfactor.party/
- https://twitch.tv/thebusfactor
- https://twitter.com/thebusfactor



Fin.

Thank you for your time

Dave - @nzkarit

References

- https://pages.nist.gov/800-63-3/sp800-63b.html#memsecret
- https://images.avisian.com/idguard_3.gif
- https://farm3.static.flickr.com/2173/1981123327_2834f04f27.jpg
- http://www.tokenguard.com/images/tokens/SID700.gif
- https://blogs.forgerock.org/petermajor/wp-content/uploads/2014/02/totp.png
- https://www.yubico.com/wp-content/uploads/2015/04/Security-Key-by-Yubico-1000-2016.png
- https://images-na.ssl-images-amazon.com/images/I/8103puSbcZL._SX355_.jpg
- https://thehackernews.com/2017/05/ss7-vulnerability-bank-hacking.html
- https://www.theregister.co.uk/2017/05/03/hackers_fire_up_ss7_flaw/
- https://thedailywtf.com/images/remy/robotguys.png
- https://smallhacks.files.wordpress.com/2012/11/camera.jpg

References

- https://kb.mailchimp.com/accounts/billing/about-mailchimp-discounts
- http://jacob.jkrall.net/otpauthexample.png
- https://twitter.com/thegrugq/status/776588609226813440?lang=en
- https://sophosnews.files.wordpress.com/2016/06/gmail-2fa-8a.png?w=350&h=621
- https://support.apple.com/library/content/dam/edam/applecare/images/en_US/appleid/ios10-iphone7-lock-screen-verification-text-notification.jpg