

PASSWORD  
SECURITY  
PART 1

# AGENDA

- \* big leaks
- \* password strength
- \* weak & strong passwords
- \* passphrases
- \* guidelines

BIG LEAKS

# BIG LEAKS: ROCKYOU

- \* RockYou (2009)  
32 mln, plain text
- \* game changer in password cracking

# BIG LEAKS: LEAKEDIN

- \* LinkedIn (2012)  
8 mln, unsalted SHA1
- \* SHA1 is very fast,  
50-60% hashes cracked within hours

# BIG LEAKS: ADOBE

- \* Adobe (2013)  
152 mln, encrypted 3DES/ECB
- \* hints in plain text
- \* the greatest crossword puzzle in the history of the world (XKCD)

# ' ; -- HAVE I BEEN PWNED?

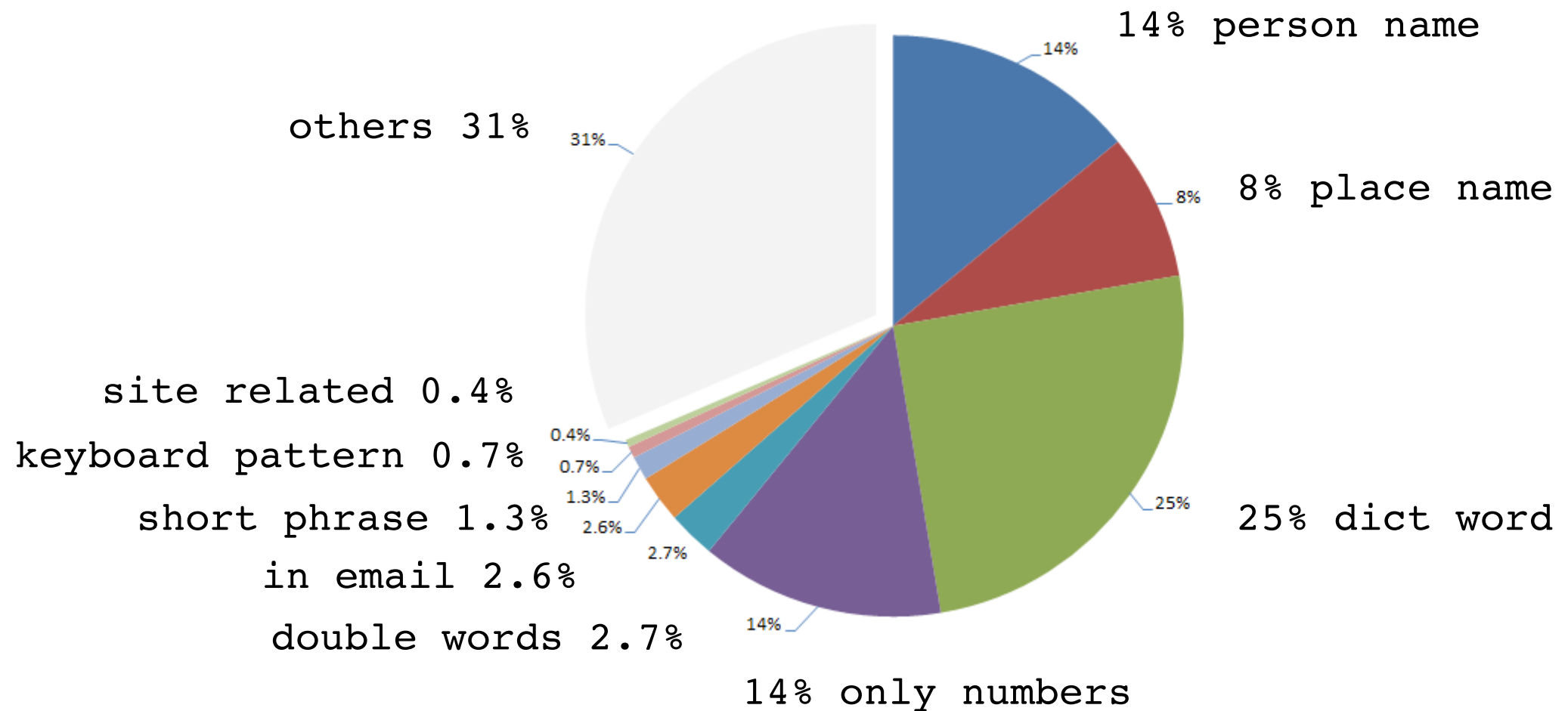
- \* so have I been pwned?  
probably yes ;)
- \* check [haveibeenpwned.com](http://haveibeenpwned.com)  
made by Troy Hunt

# ROCKYOU ANALYSIS

- \* most passwords are short,  
6-10 characters
- \* capital letters  
mostly come at the beginning
- \* numbers and punctuation  
mostly show up at the end
- \* strong tendency to use  
first names followed by years



# LULZSEC RELEASES 2011



# TOP 25 PASSWORDS 2013

123456	123123	password1
password	Admin	princess
12345678	1234567890	azerty
qwerty	letmein	trustno1
abc123	photoshop	000000
123456789	1234	
111111	monkey	
1234567	shadow	
iloveyou	sunshine	
adobe123	12345	

PASSWORD STRENGTH

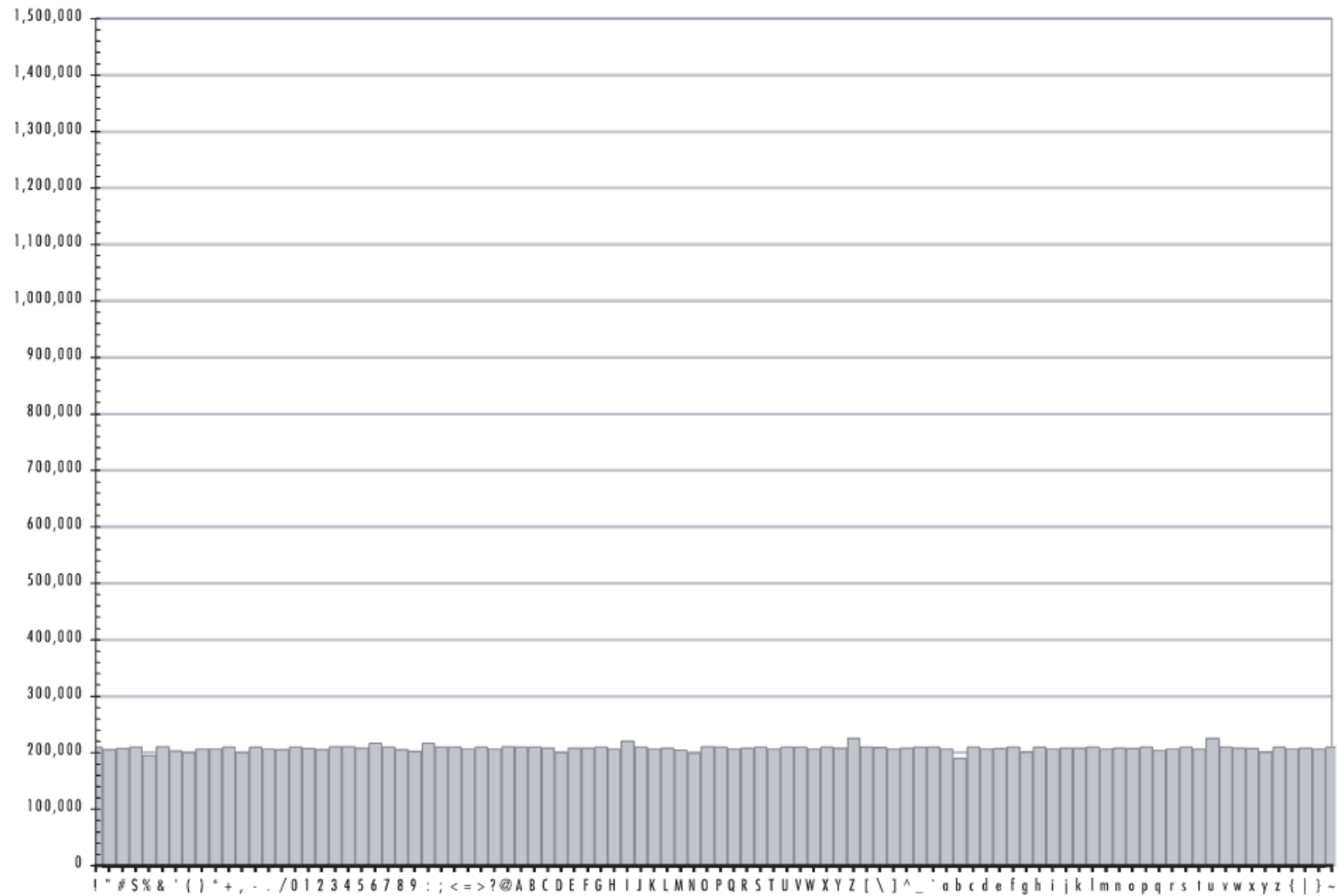
# HOW TO MEASURE

- \* entropy, measured in bits
- \* strength of random passwords can be estimated
- \* passwords generated by people are difficult to estimate

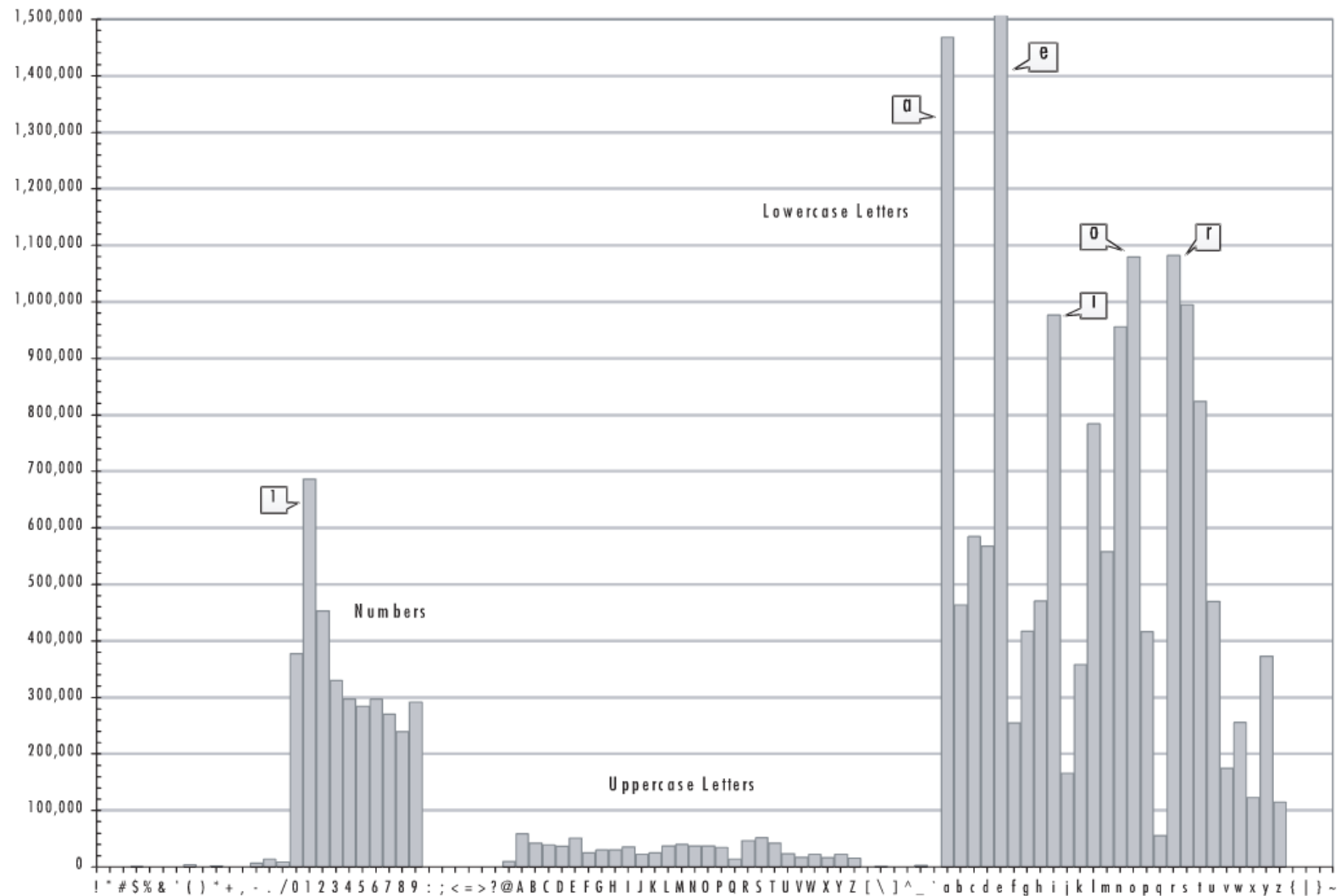
# RANDOMNESS

- \* humans tend to follow patterns
- \* are unable to achieve sufficient entropy
- \* rarely make full use of key space

# RANDOM DISTRIBUTION



# HUMAN RANDOMNESS



# DEPENDS ON USER

- \* key space:  
length and complexity
- \* can be controlled up to certain  
level by password policy
- \* reusing passwords is dangerous



# DEPENDS ON DEVELOPER

- \* password storage:
  - type of hash function
  - key stretching
- \* other system vulnerabilities

# DEPENDS ON ATTACKER

- \* hardware involved
- \* identified password limitations
- \* knowledge and quality of tools
- \* identified other system vulnerabilities

WEAK PASSWORDS

# WEAK PASSWORDS

- \* single or doubled dictionary words
- \* words with appended numbers or symbols
- \* obfuscated words, leet speak
- \* common keyboard sequences

# WEAK PASSWORDS

- \* any purely numeric passwords
- \* anything related:  
identifiers, usernames, emails,  
license plate, phone numbers,  
addresses, dates, birthdays,  
names, nicknames, initials,  
content of WHOIS db!

# EXAMPLES

kitty	1Kitty	1Ki77y
susan	Susan53	.Susan53.
jellyfish	jelly22fish	J3lly22Fish
smellycat	sm3llycat	\$m3llycat.
allblacks	AllBlacks!	A11Black\$!
jackbauer	jAckBauer	jA(kBauer
doctorhouse	Doct0rH0use	.Doct0rH0use.
adamsandler	adamSandler	#adamS@ndler
ilovemypiano	ILoveMyPiano	ILov3MyPi@no

# MORE EXAMPLES

k1araj0hns0n  
Sh1a-labe0uf  
Apr!1221973  
Qbesancon321  
DG091101%  
@Yourmom69  
ilovetofunot  
windermere2313  
tmdmmj17

BandGeek2014  
all of the lights  
i hate hackers  
allinedislove  
ilovemySister31  
iloveyousomuch  
Philippians4:13  
Philippians4:6-7  
qeadzcwrsfxv1331

STRONG PASSWORDS



# STRONG PASSWORDS

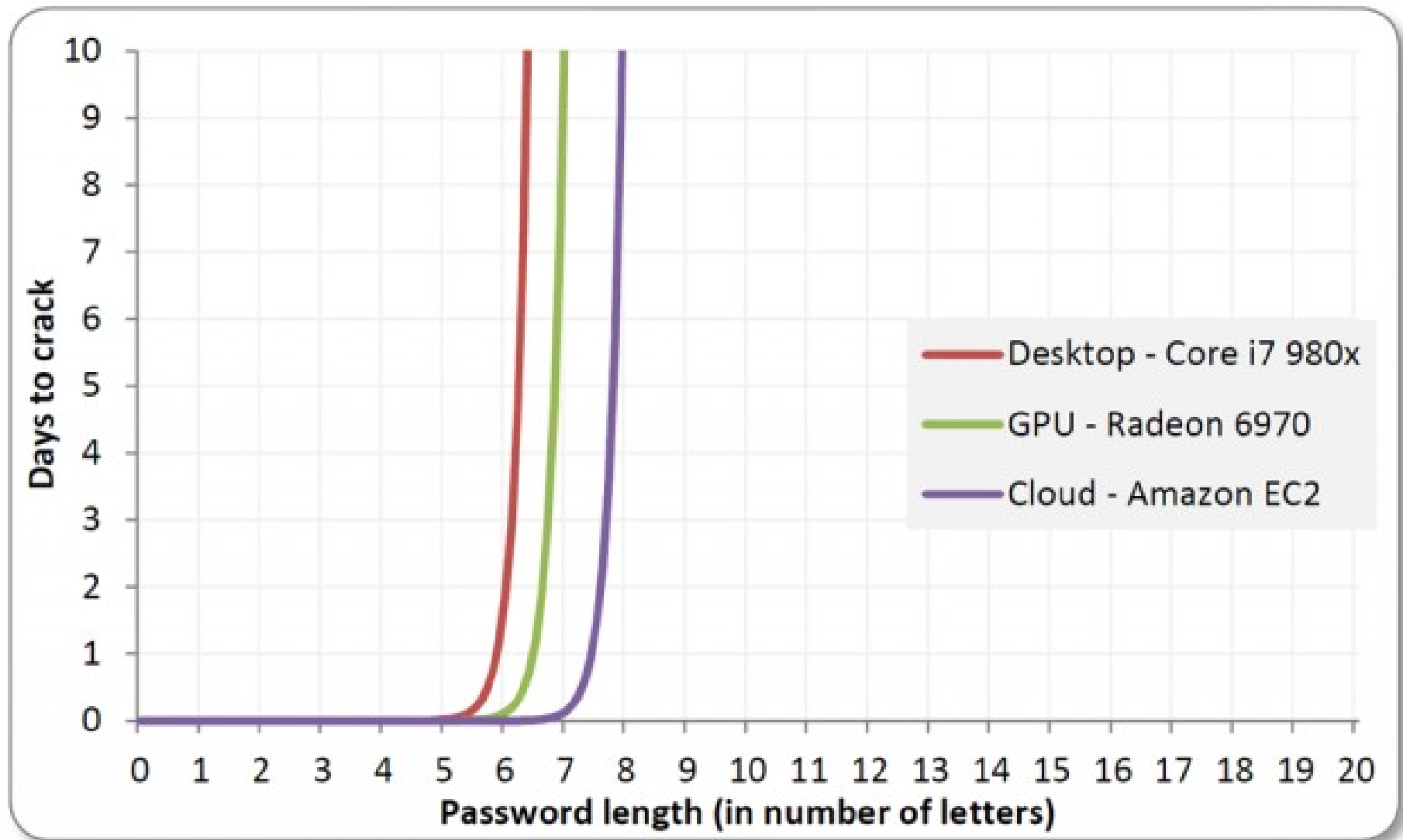
```
$pwgen -n 15 -sy
```

```
*?/#C"*:8lq1:jV  
.n'rUXJ+jcZ\%D9  
7qvmh*O.q$:P$\M  
oO8seLzCUbN}h#p  
#-5L=UBd6!%vH4G
```

```
$pwgen -n 15 -s
```

```
cn9KgidMrOD0zjh  
Xc4dXxuZpImQFOp  
NvC0xBPt60VRMmk  
FgUwSOsJl5Prw8V  
VE2zQM02gQaoiQL
```

# EXPOTENTIAL WALL OF BF



PASSPHRASES

# PASSPHRASES

- \* strong and memorable secrets
- \* short phrases will be cracked
- \* avoid popular phrases, quotes, lyrics, things from Wikipedia
- \* introduce some variation:  
mixed case, digits, specials
- \* diceware method

# EXAMPLES

- \* correct horse battery staple
- \* sensible shark rubs own belly
- \* never\_seats\_ghost\_main\_97
- \* slam,rust,armor,gg,spire

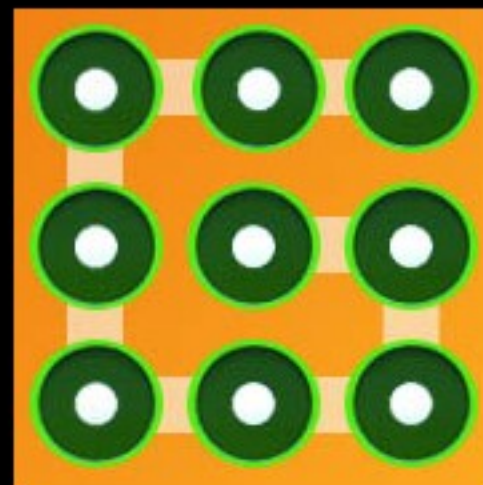
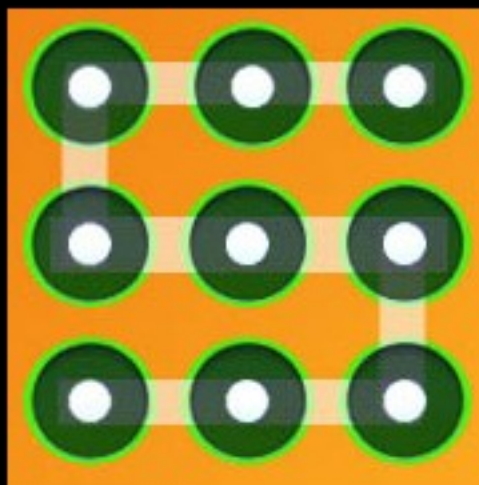
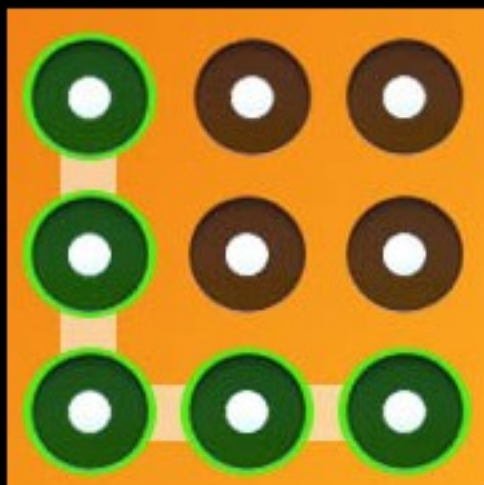
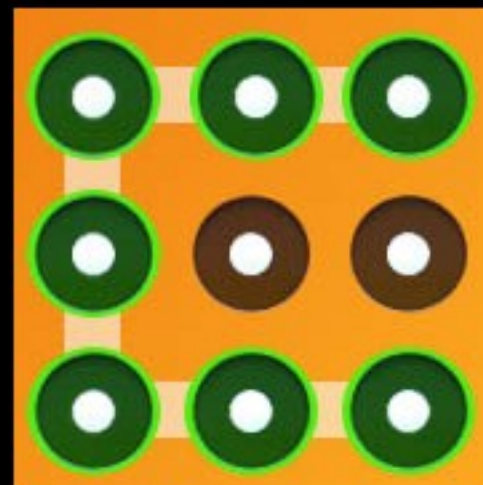
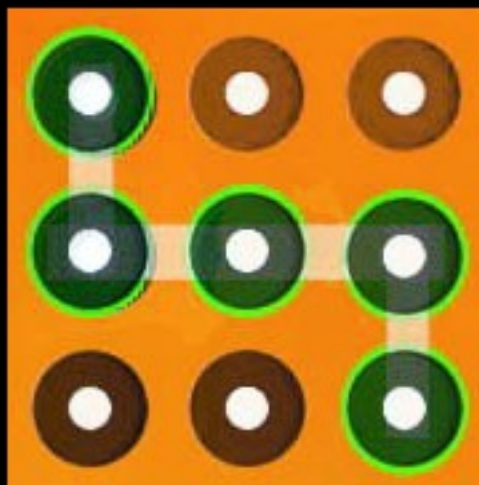
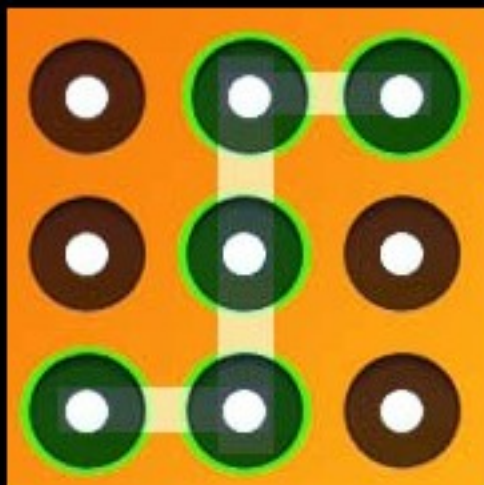
# PASSWORD MANAGERS

- \* people can not remember so many good passwords, too many accounts
- \* the only way to securely store secrets and password managers
- \* LastPass, Passpack, KeePassX

# 2 FACTOR AUTH

- \* use second factor is possible
- \* secure recovery codes
- \* lock your phone:  
clean screen, hide pattern

# COMMON PATTERNS





# GUIDELINES

# GUIDELINES 4 USERS

- \* generate random passwords or use memorable and long passphrases
- \* do not reuse
- \* store in password manager
- \* backup your passwords (offline/offsite)

# GUIDELINES 4 USERS

- \* avoid known patterns
- \* change default passwords
- \* change password if compromised
- \* turn on 2-factor authentication

# GUIDELINES 4 DEVS

- \* use BCrypt, SCrypt or PBKDF2 with dynamic salts
- \* require sufficient password complexity, but do not enforce very strict patterns
- \* do not relay on secret hash permutations

# GUIDELINES 4 QA

- \* check hashes, passwords can not be stored encrypted or in plain text
- \* check if hash function is still safe
- \* check if hashes are salted with dynamic salt
- \* check if work factor is still sufficient

# GUIDELINES 4 QA

- \* check if required password complexity is sufficient
- \* check if system allows obvious passwords: empty, same as login, name or email
- \* check if API is rate limited

# REFERENCES: WEB

- \* [sekurak.pl](http://sekurak.pl)
- \* [arstechica.com](http://arstechica.com)
- \* [troyhunt.com](http://troyhunt.com)
- \* [haveibeenpwned.com](http://haveibeenpwned.com)
- \* [splashdata.blogspot.com](http://splashdata.blogspot.com)
- \* [mytrickytricks.blogspot.com](http://mytrickytricks.blogspot.com)
- \* [entima.net/diceware/](http://entima.net/diceware/)

# REFERENCES: BOOKS

- \* Take Control of Your Passwords
- \* Perfect Password