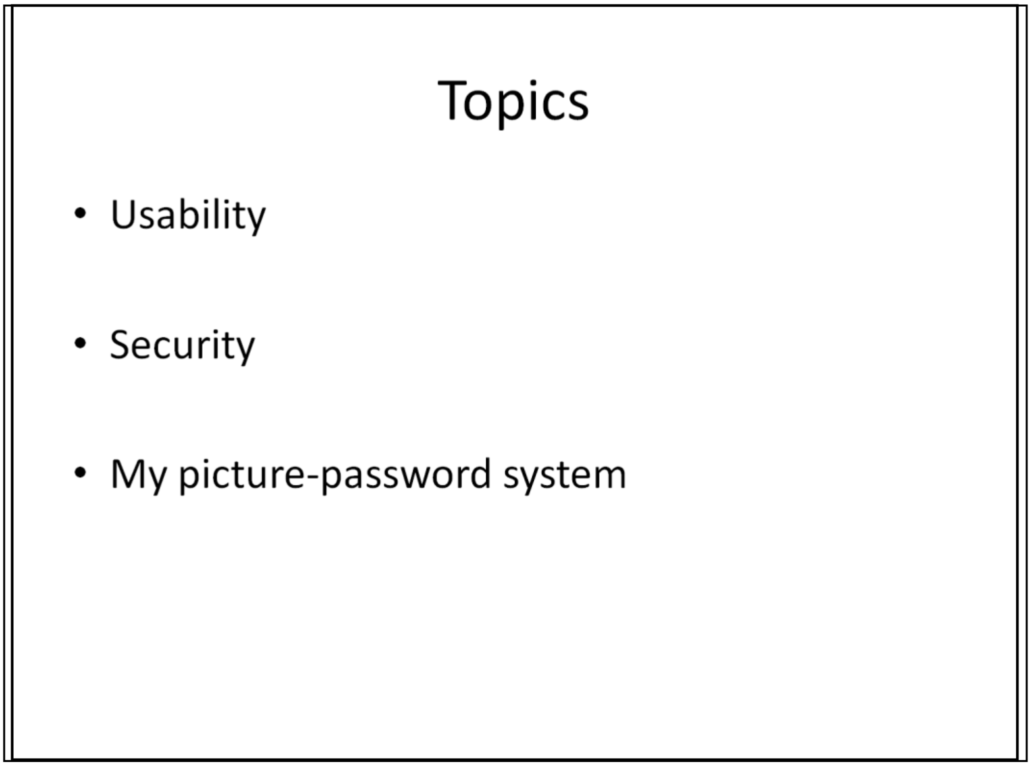
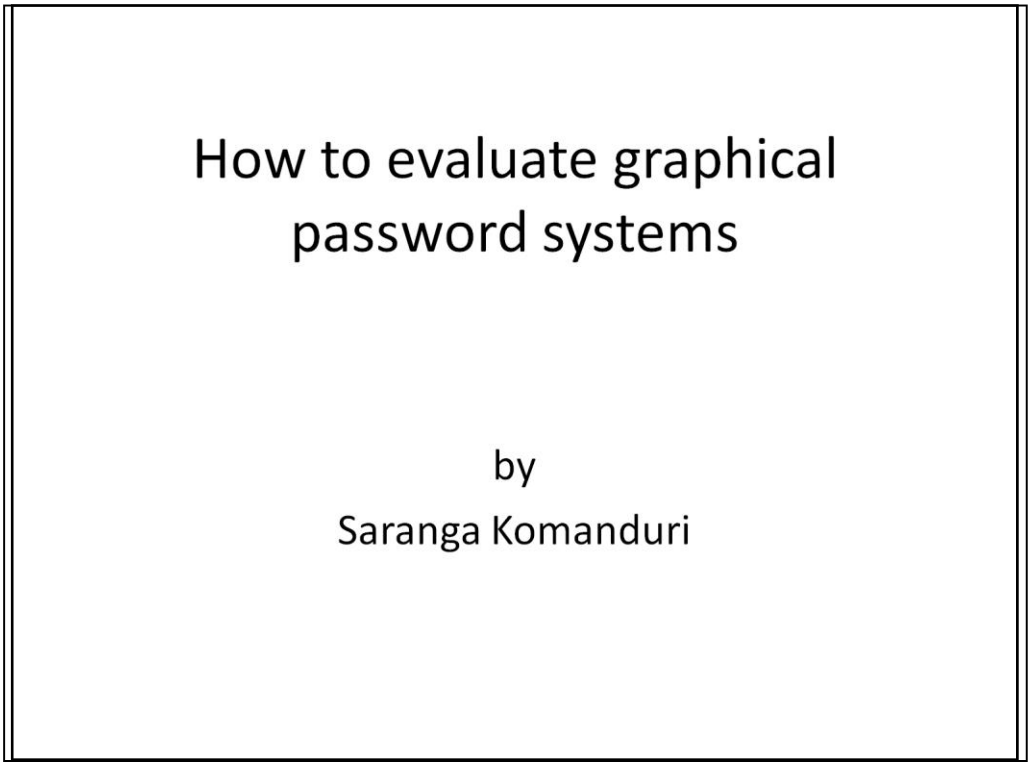


Topics

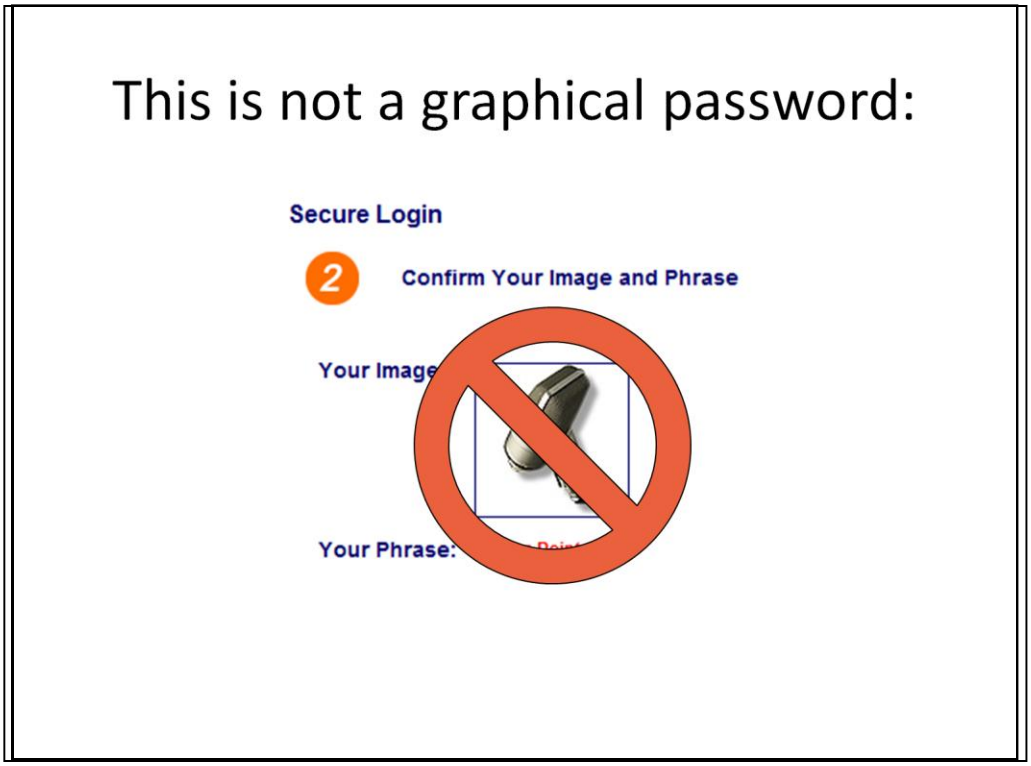
- Usability
- Security
- My picture-password system



How to evaluate graphical password systems

by

Saranga Komanduri



This is not a graphical password:

Secure Login

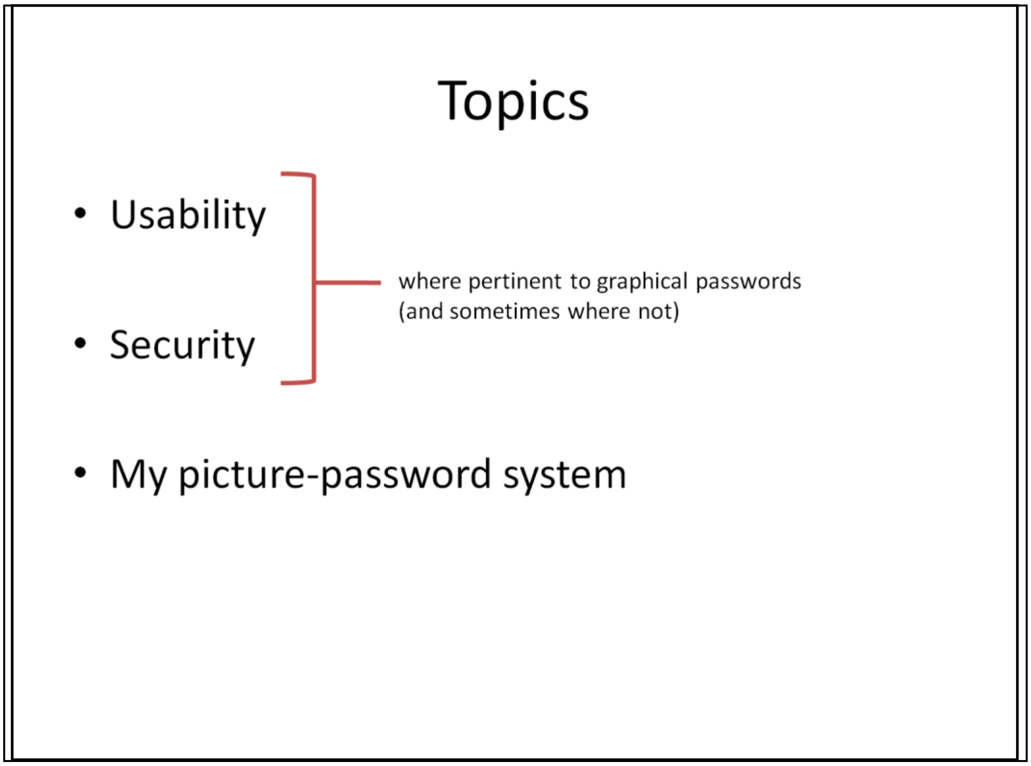
2 Confirm Your Image and Phrase

Your Image

Your Phrase:

The SiteKey. This is not a graphical password system.

...and I'm pretty sure it doesn't work.



Topics

- Usability
- Security

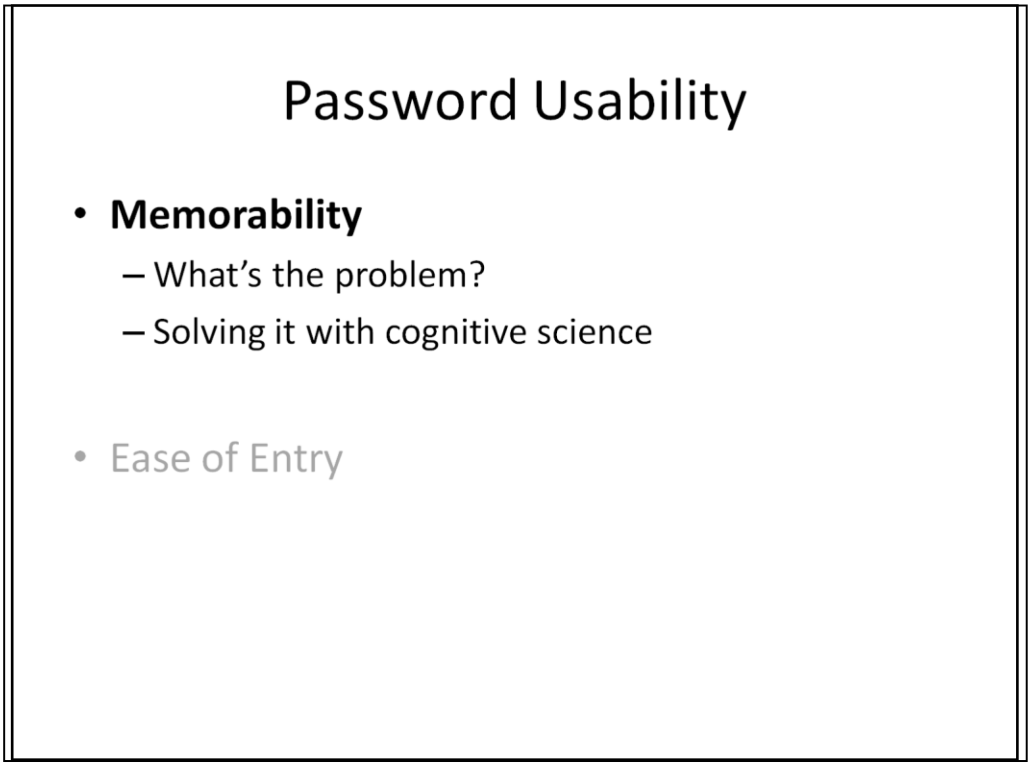
where pertinent to graphical passwords
(and sometimes where not)

- My picture-password system

- basic outline
-problems with password usability and security
-how various graphical password systems address them.

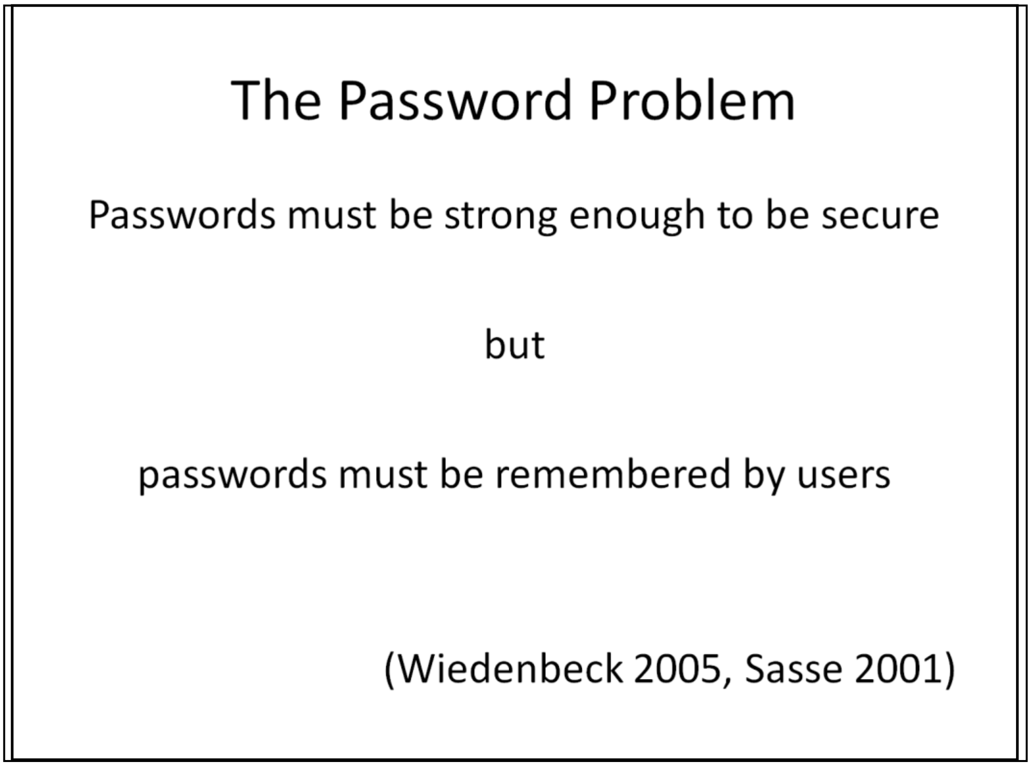
Usability includes memorability – this will be a big chunk of my talk – and ease of entry
Security includes issues of social engineering, cracking, and shoulder-surfing

-The picture-password system I developed for my master's thesis



Password Usability

- **Memorability**
  - What’s the problem?
  - Solving it with cognitive science
- Ease of Entry



The Password Problem

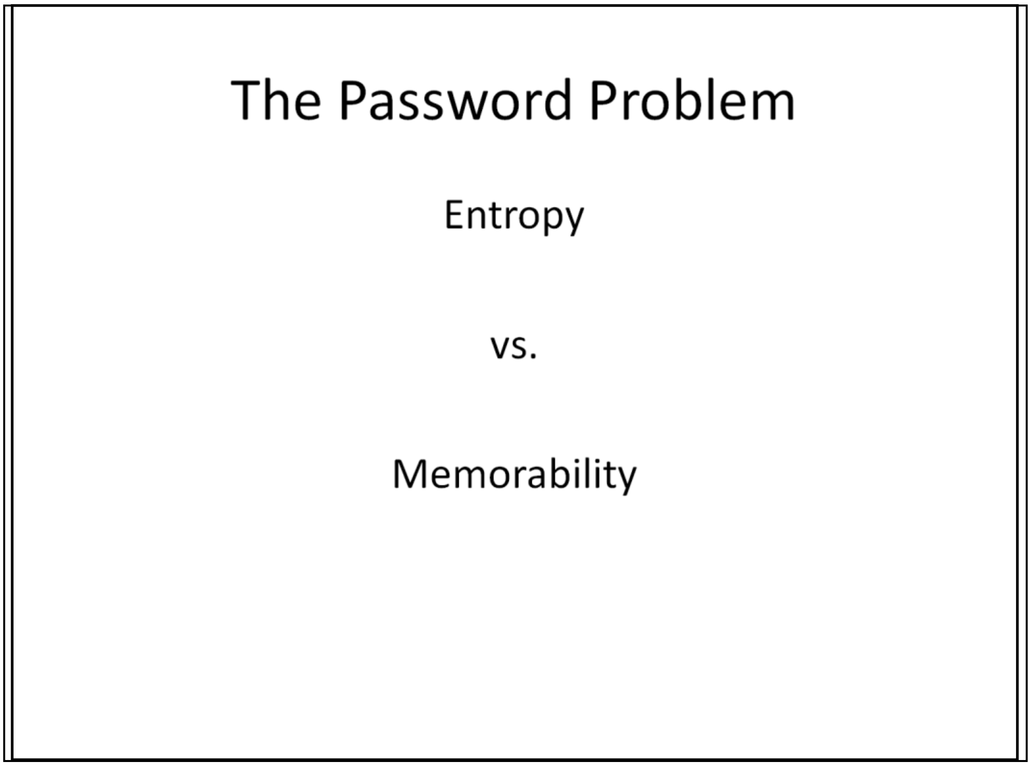
Passwords must be strong enough to be secure

but

passwords must be remembered by users

(Wiedenbeck 2005, Sasse 2001)

It in our terminology this boils down to…

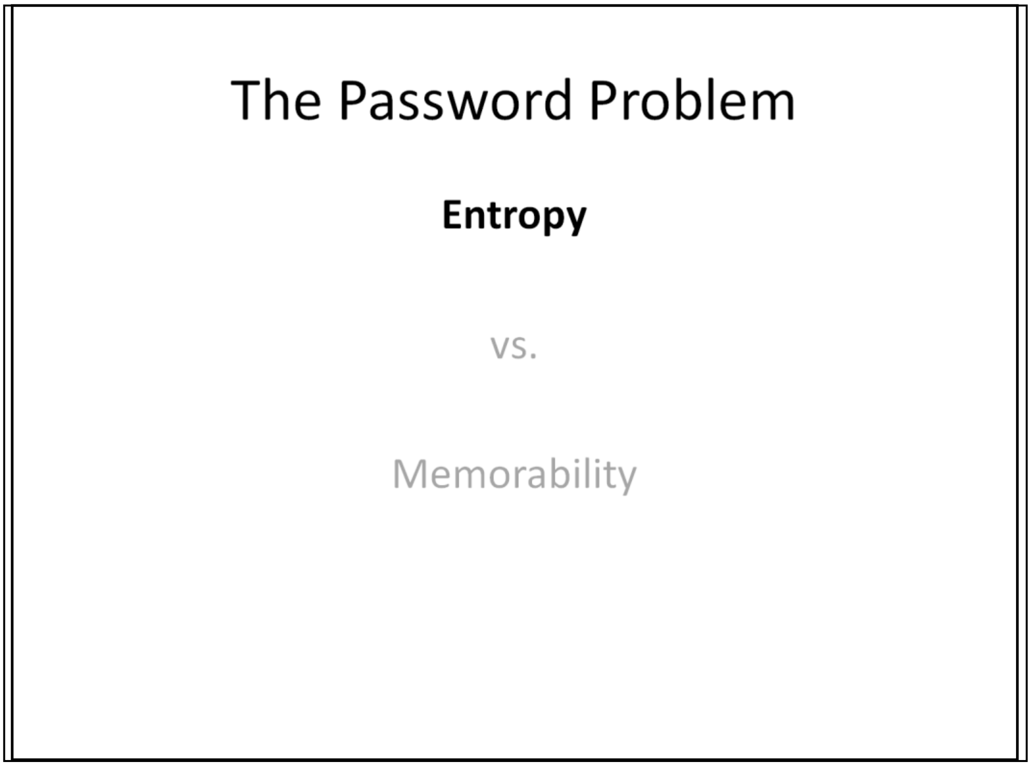


The Password Problem

Entropy

vs.

Memorability

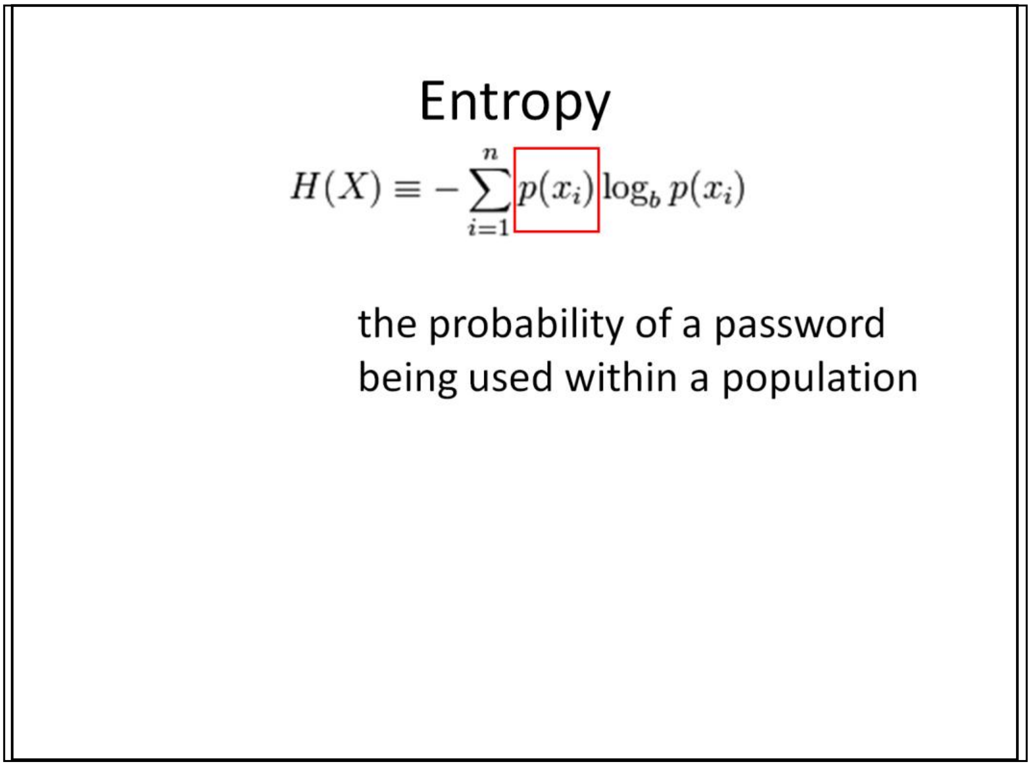


The Password Problem

Entropy

vs.

Memorability



Entropy

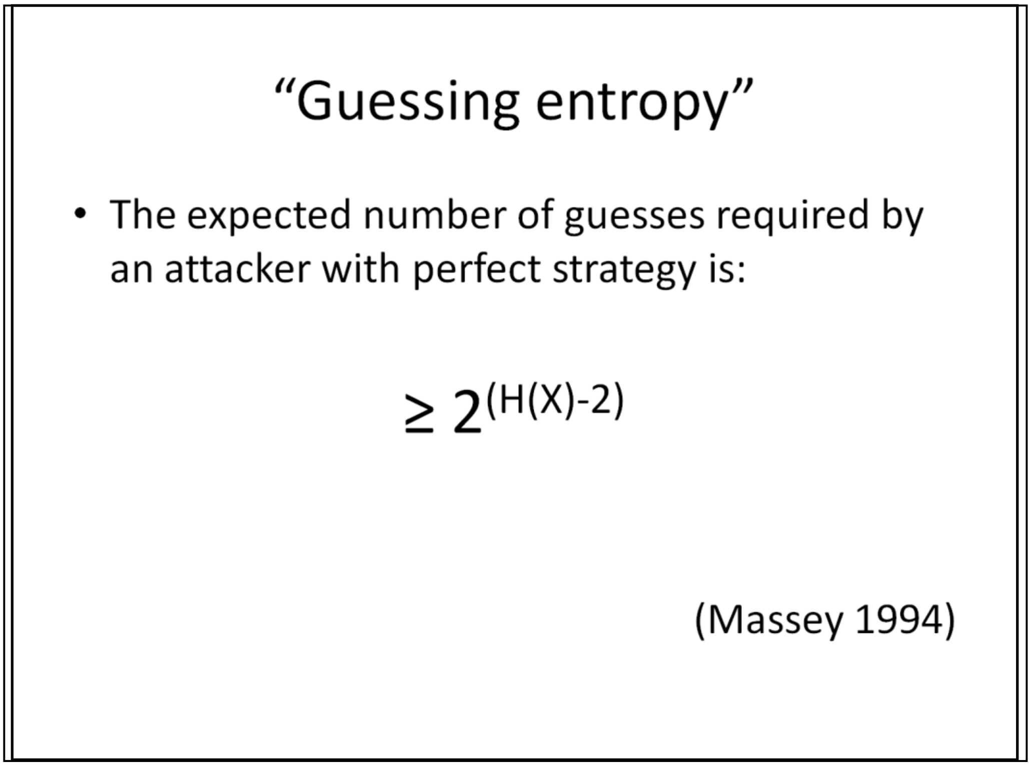
$$H(X) \equiv -\sum_{i=1}^{n} \boxed{p(x_i)} \log_b p(x_i)$$

the probability of a password being used within a population

Rich covered this in our last student presentation, and I’ll be building on that a little bit.

You can talk about the entropy of individual characters, but they have to be entered in the right order, so you end up needing to know the probabilities of entire passwords.

When this concept is applied to passwords it is also called the “guessing entropy”



“Guessing entropy”

- The expected number of guesses required by an attacker with perfect strategy is:

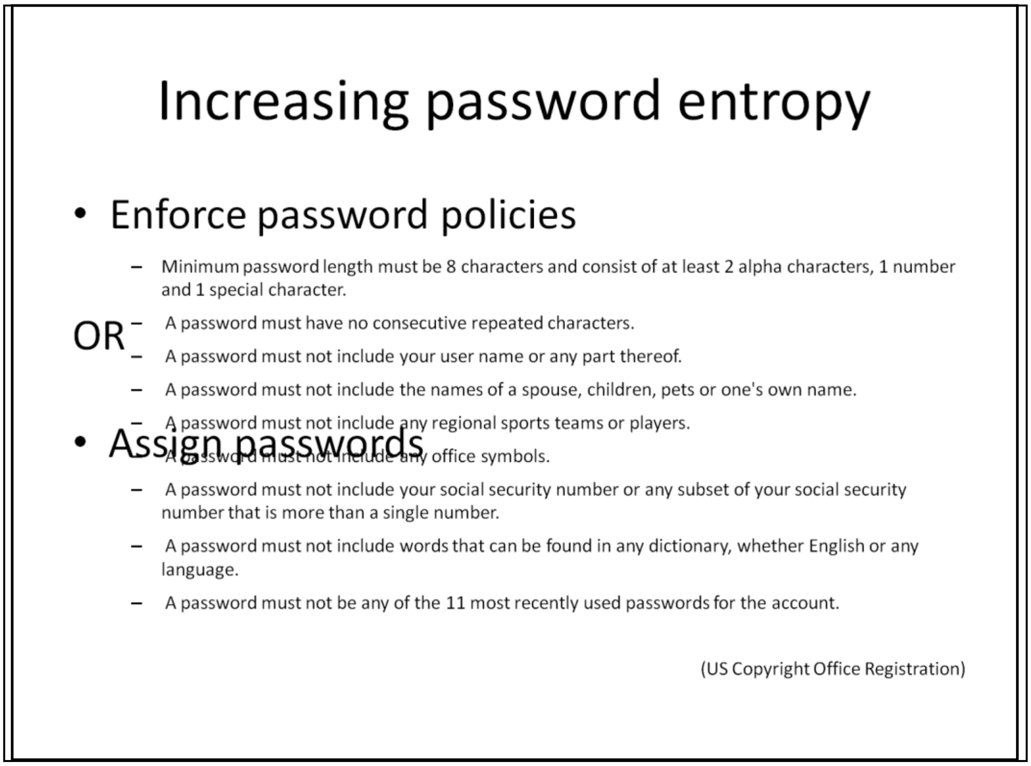
$$\geq 2^{(H(X)-2)}$$

(Massey 1994)

…which is well-explained in chapter 9 of our book.

So, the more entropy in our passwords the harder they are to guess.

Looking at this, if you are in charge of a password system, you’d want to



Increasing password entropy

- Enforce password policies
  - Minimum password length must be 8 characters and consist of at least 2 alpha characters, 1 number and 1 special character.
  - A password must have no consecutive repeated characters.
  - A password must not include your user name or any part thereof.
  - A password must not include the names of a spouse, children, pets or one's own name.
  - A password must not include any regional sports teams or players.
  - A password must not include any office symbols.
  - A password must not include your social security number or any subset of your social security number that is more than a single number.
  - A password must not include words that can be found in any dictionary, whether English or any language.
  - A password must not be any of the 11 most recently used passwords for the account.

OR

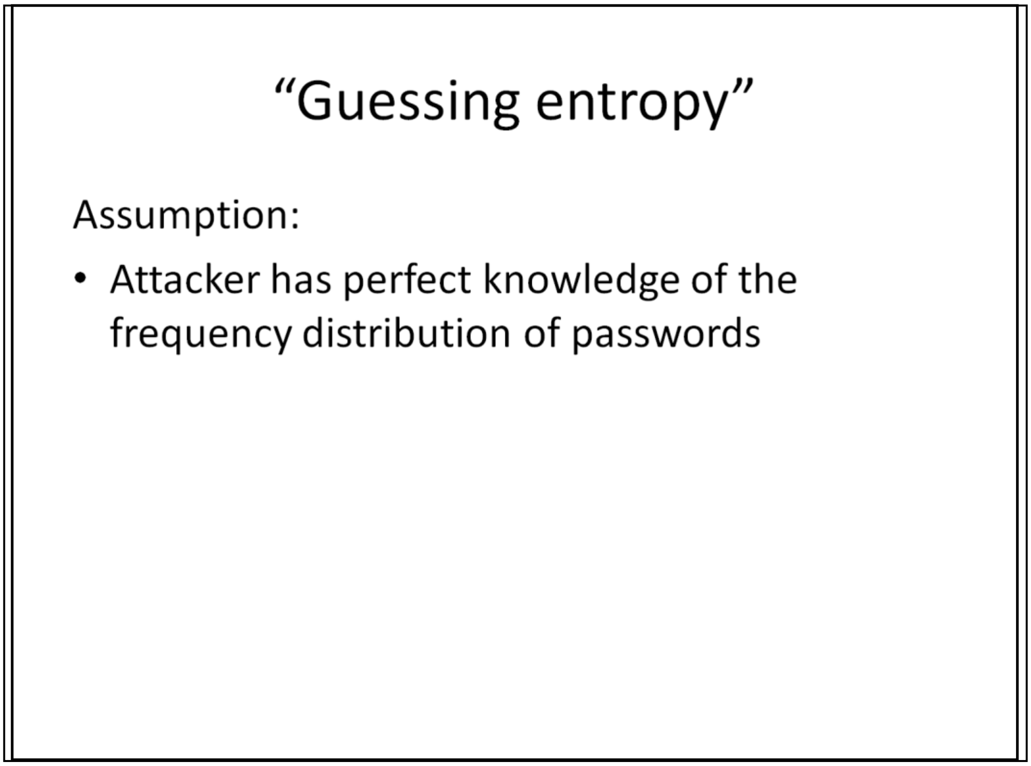
- Assign passwords

(US Copyright Office Registration)

…increase the entropy of your passwords.

Here are some ways to increase entropy…

(discuss) One way to increase entropy (check if passwords match)



“Guessing entropy”

Assumption:

- Attacker has perfect knowledge of the frequency distribution of passwords

There is an assumption being made here, that the attacker has perfect strategy, but this assumes the above.

Something I want you to think about: Is this a good assumption?

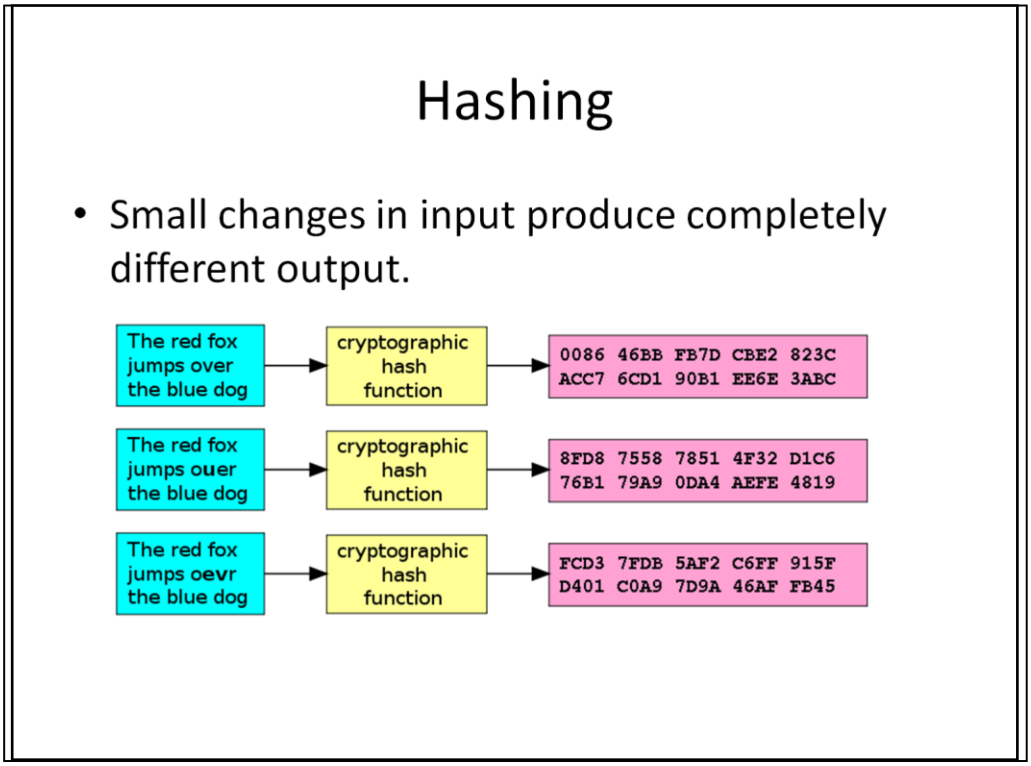
And I want you to think about that as I talk about

Hashing

- Convert any input to a string of random characters

Ex. SHA-512(“security”) =
f2a46a9101d3b65c419c98a9ffe73c154196bc3
e87379491746cf5a70ee0b5e4d308b27b28f77
960582d8ff88ab7c3c4930860436bf05d6d551
7c8e3f9efb8e5

- Consistent but non-reversible



Hashing

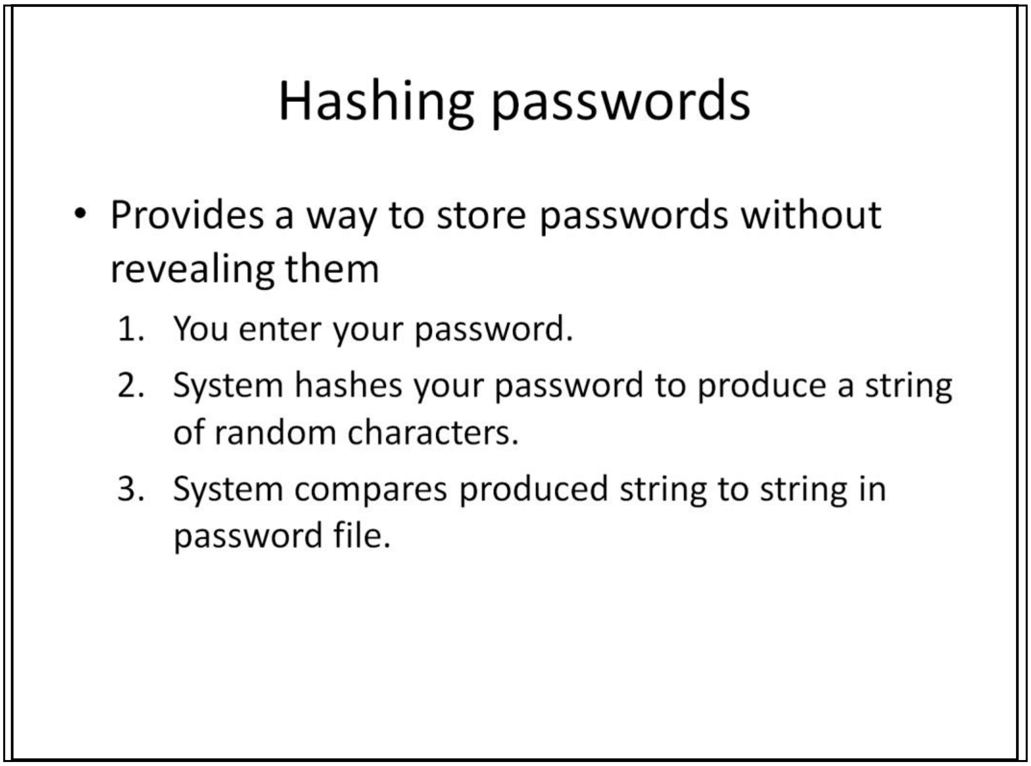
- Small changes in input produce completely different output.

The red fox jumps over the blue dog → cryptographic hash function → 0086 46BB FB7D CBE2 823C ACC7 6CD1 90B1 EE6E 3ABC

The red fox jumps ouer the blue dog → cryptographic hash function → 8FD8 7558 7851 4F32 D1C6 76B1 79A9 0DA4 AEFE 4819

The red fox jumps oevr the blue dog → cryptographic hash function → FCD3 7FDB 5AF2 C6FF 915F D401 C0A9 7D9A 46AF FB45

Hashing is used in almost all standard password systems.



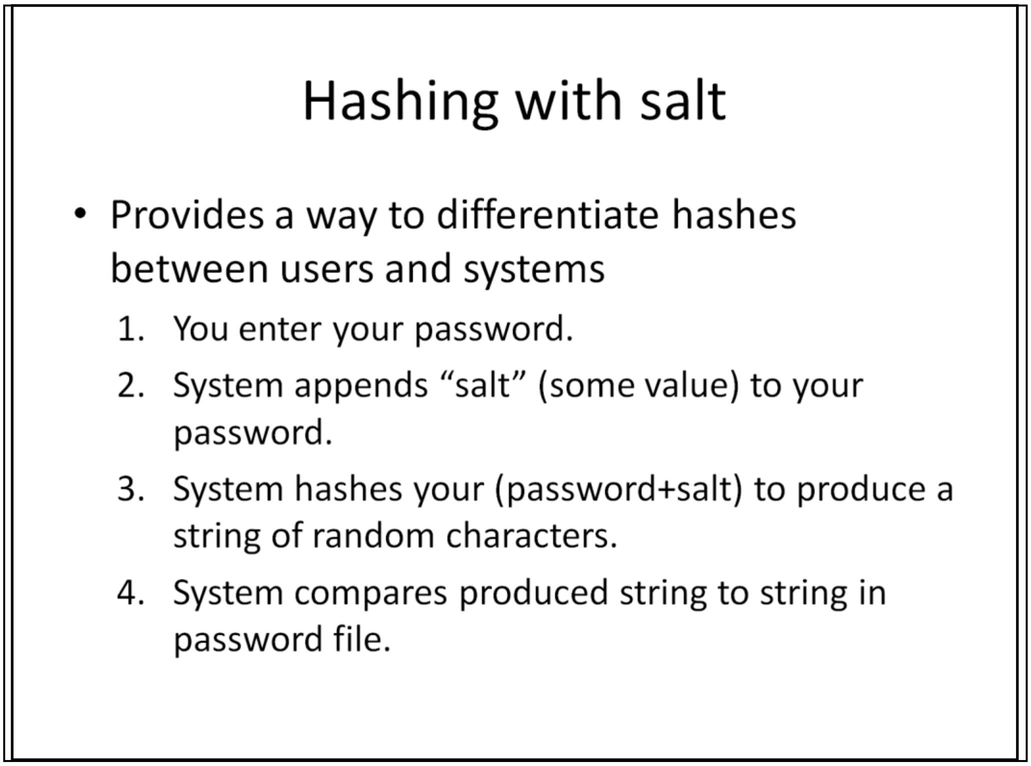
Hashing passwords

- Provides a way to store passwords without revealing them
  1. You enter your password.
  2. System hashes your password to produce a string of random characters.
  3. System compares produced string to string in password file.

Hashing is used in almost all standard password systems.

-System doesn’t even know your password. An admin looking at the password file doesn’t know your password…

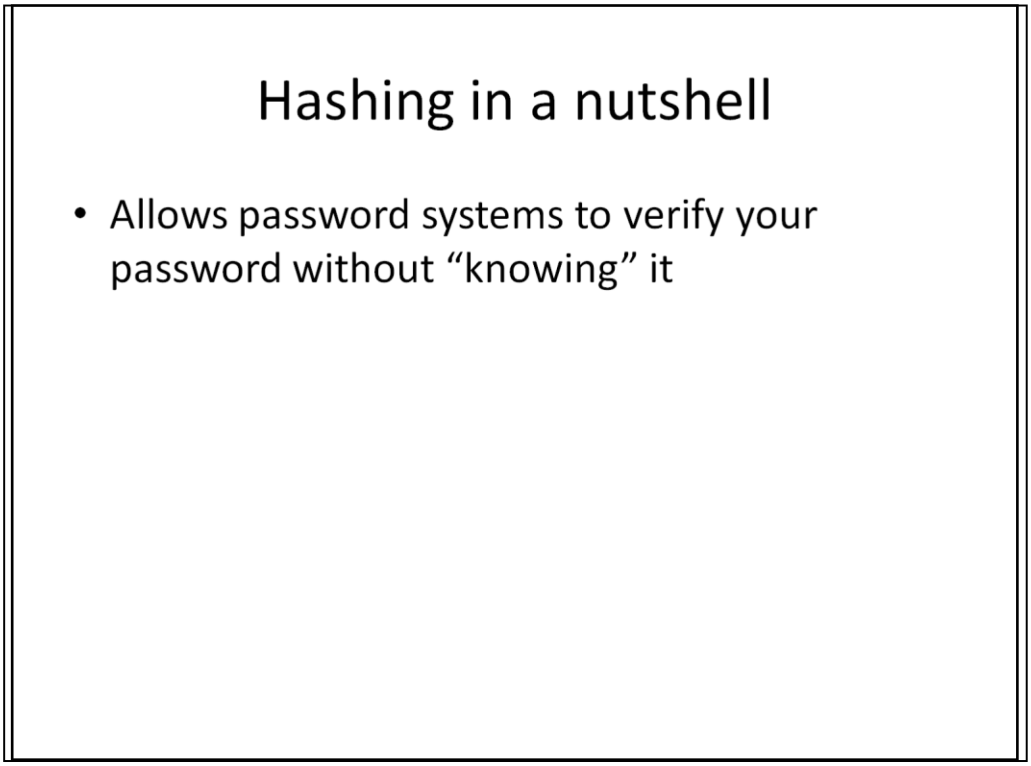
- But if an admin sees two hashes in the file that are the same…



Hashing with salt

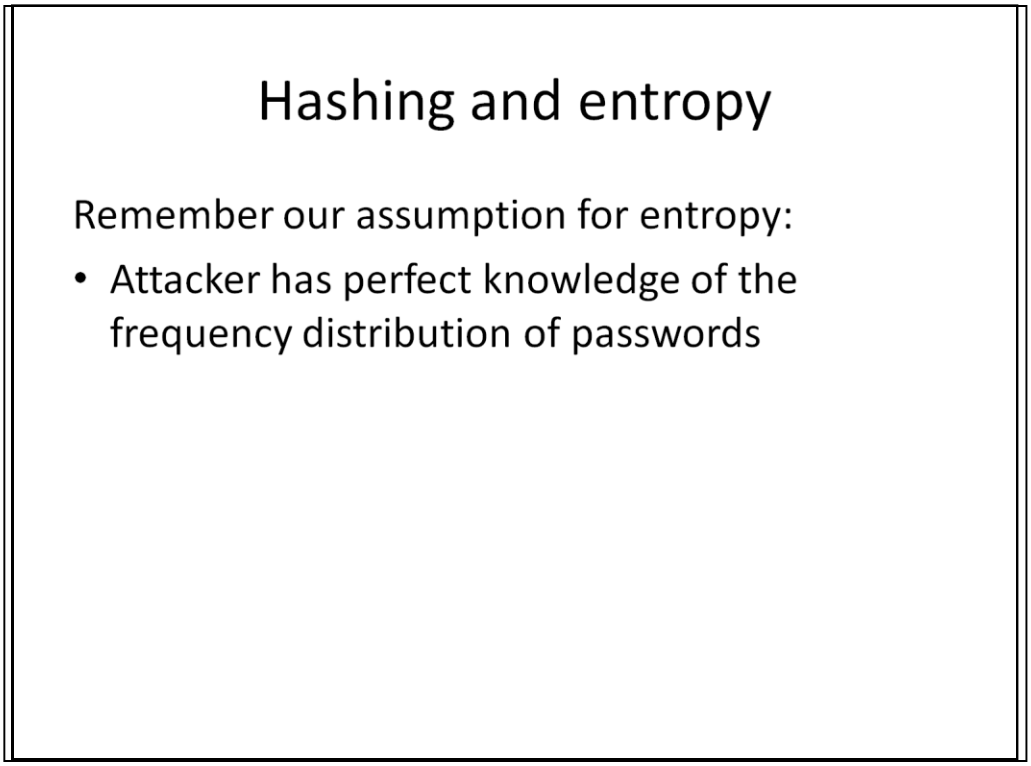
- Provides a way to differentiate hashes between users and systems
  1. You enter your password.
  2. System appends “salt” (some value) to your password.
  3. System hashes your (password+salt) to produce a string of random characters.
  4. System compares produced string to string in password file.

So we add salt. This makes the hashes different between users and even across systems, so you can use the same password on multiple systems or two people could have the same password and no one will know, even if they know the salts (which are typically stored in the password file).



Hashing in a nutshell

- Allows password systems to verify your password without “knowing” it



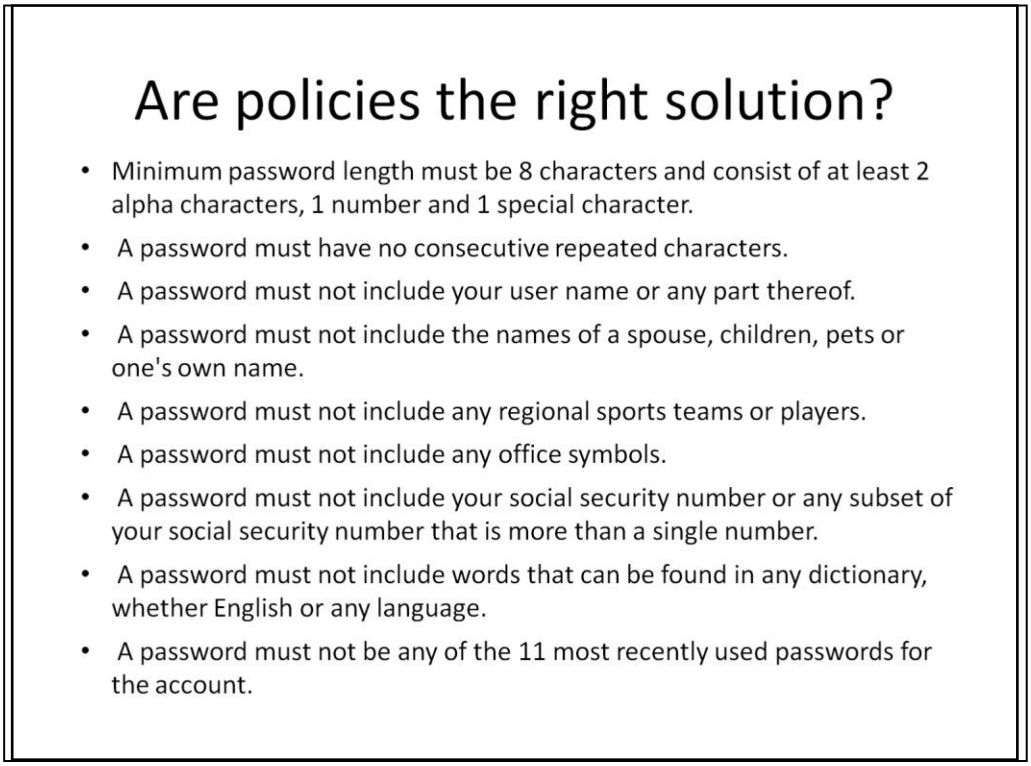
Hashing and entropy

Remember our assumption for entropy:

- Attacker has perfect knowledge of the frequency distribution of passwords

Given that we can hash passwords to hide this information, is entropy the right way to think about passwords?

And remember what entropy analysis does



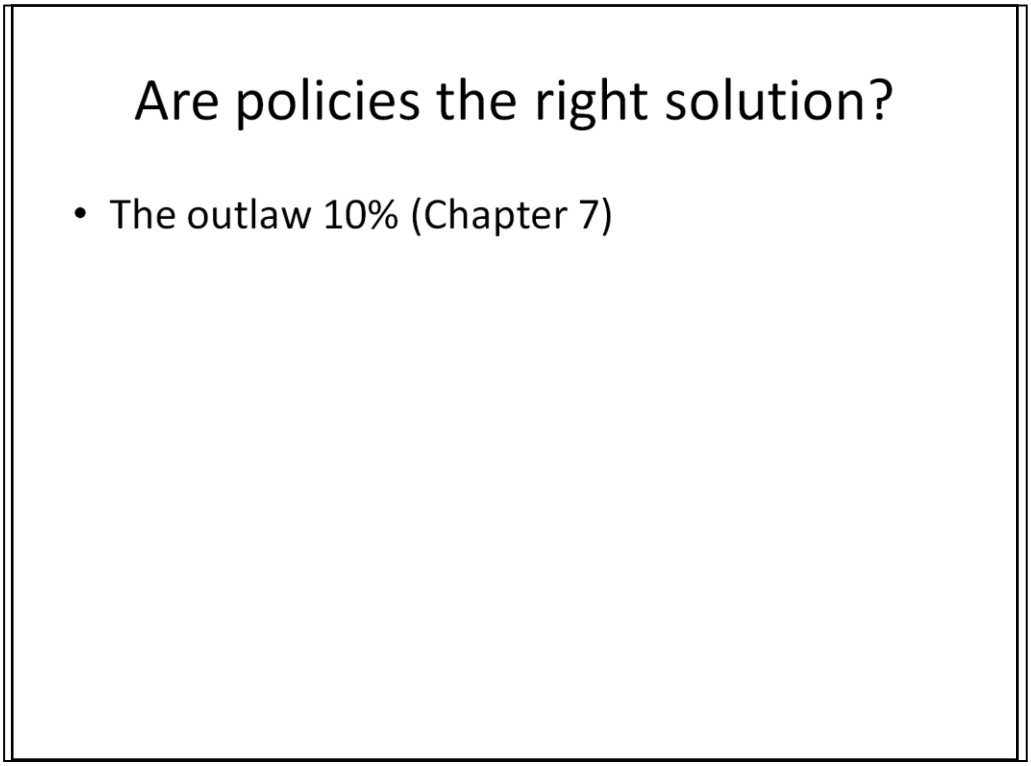
Are policies the right solution?

- Minimum password length must be 8 characters and consist of at least 2 alpha characters, 1 number and 1 special character.
- A password must have no consecutive repeated characters.
- A password must not include your user name or any part thereof.
- A password must not include the names of a spouse, children, pets or one's own name.
- A password must not include any regional sports teams or players.
- A password must not include any office symbols.
- A password must not include your social security number or any subset of your social security number that is more than a single number.
- A password must not include words that can be found in any dictionary, whether English or any language.
- A password must not be any of the 11 most recently used passwords for the account.

…it produces policies like this.

(discuss) Given what you've learned so far, do policies like this make sense?

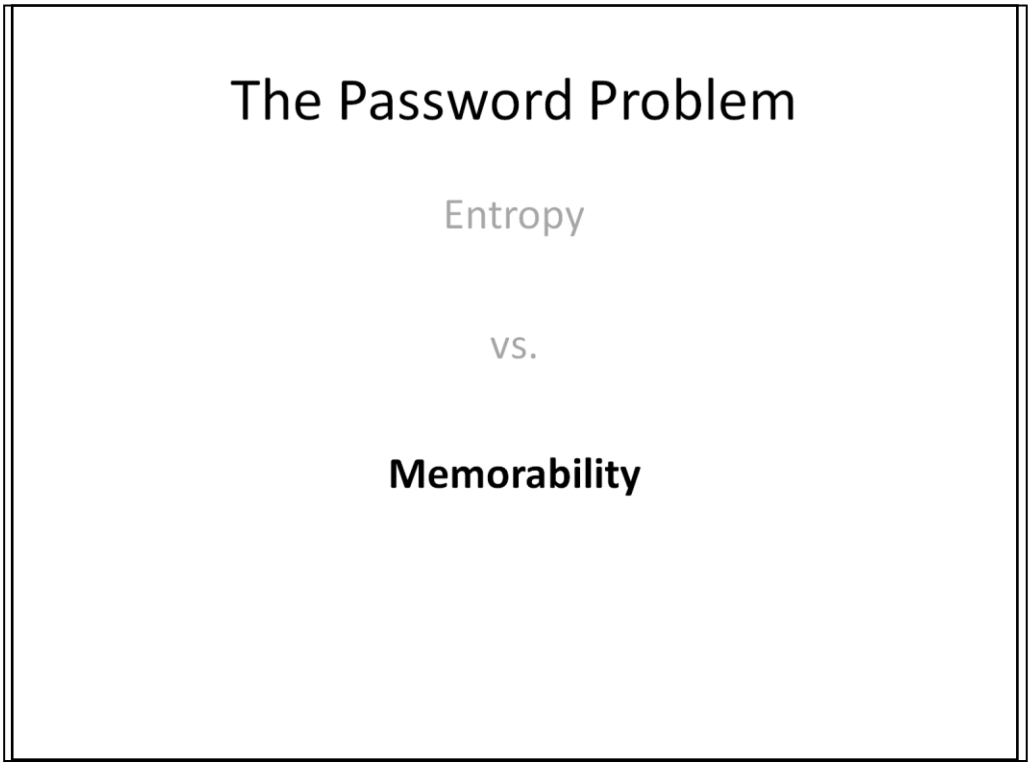
-Does anything on this list seem unnecessary?
-Does anything seem necessary?



Are policies the right solution?

- The outlaw 10% (Chapter 7)

Jeff Yan in chapter 7 and in other papers says that about 10% of a population will always be non-compliant…

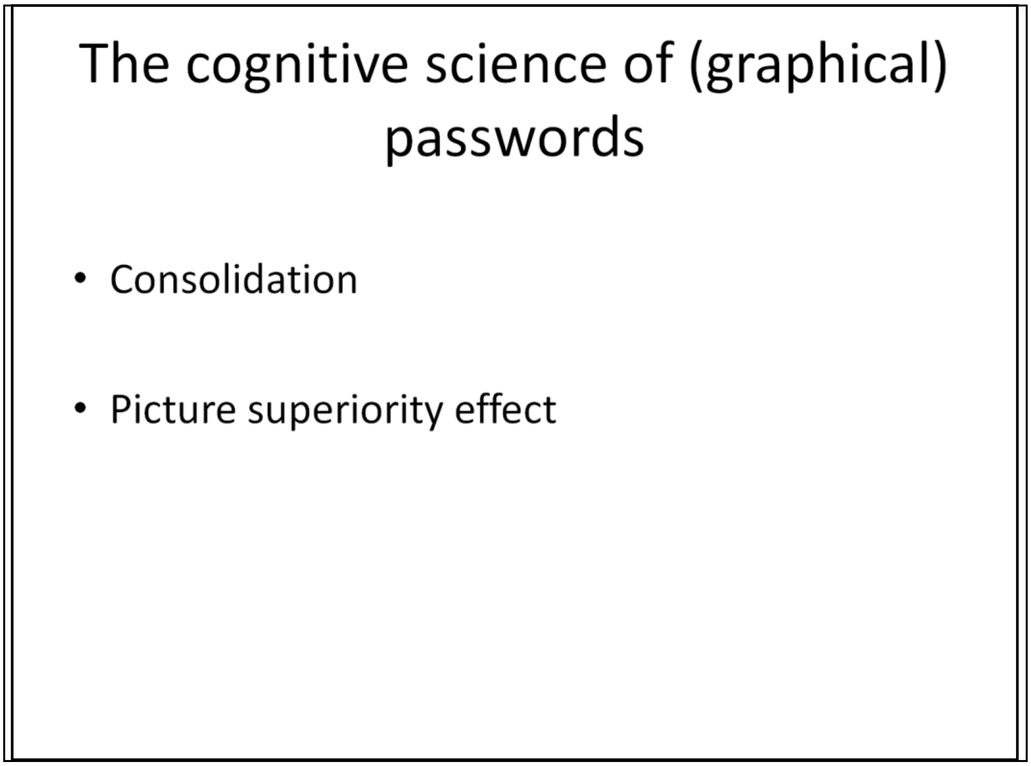


The Password Problem

Entropy

vs.

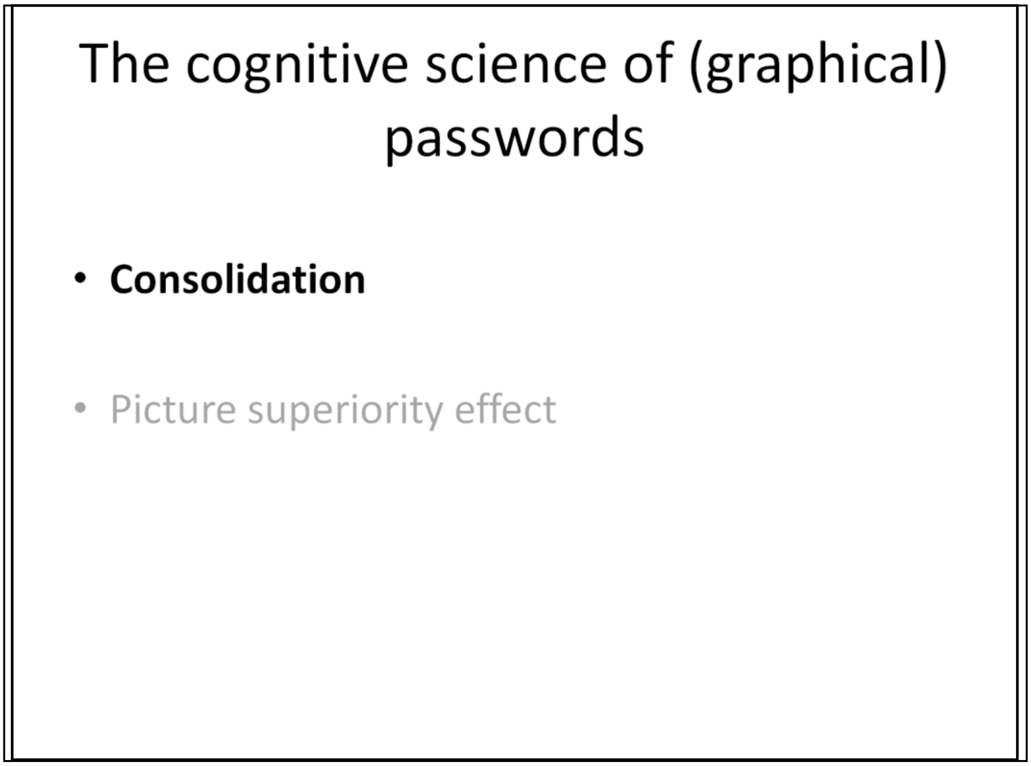
Memorability



The cognitive science of (graphical) passwords

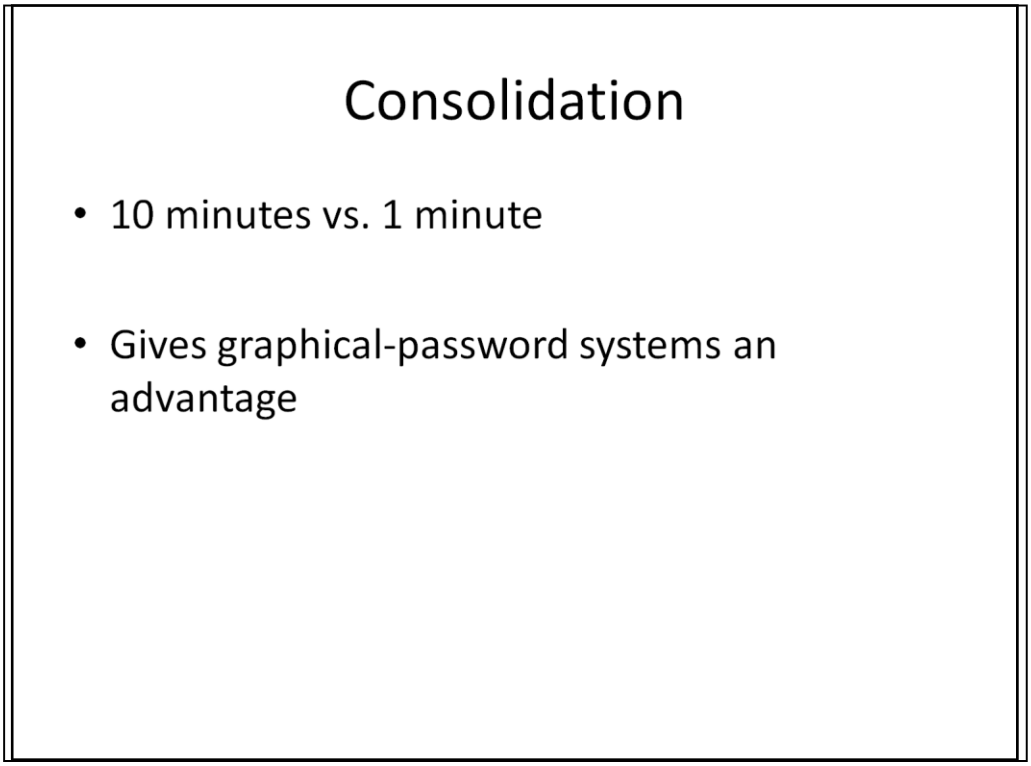
- Consolidation

- Picture superiority effect



The cognitive science of (graphical) passwords

- **Consolidation**
- Picture superiority effect



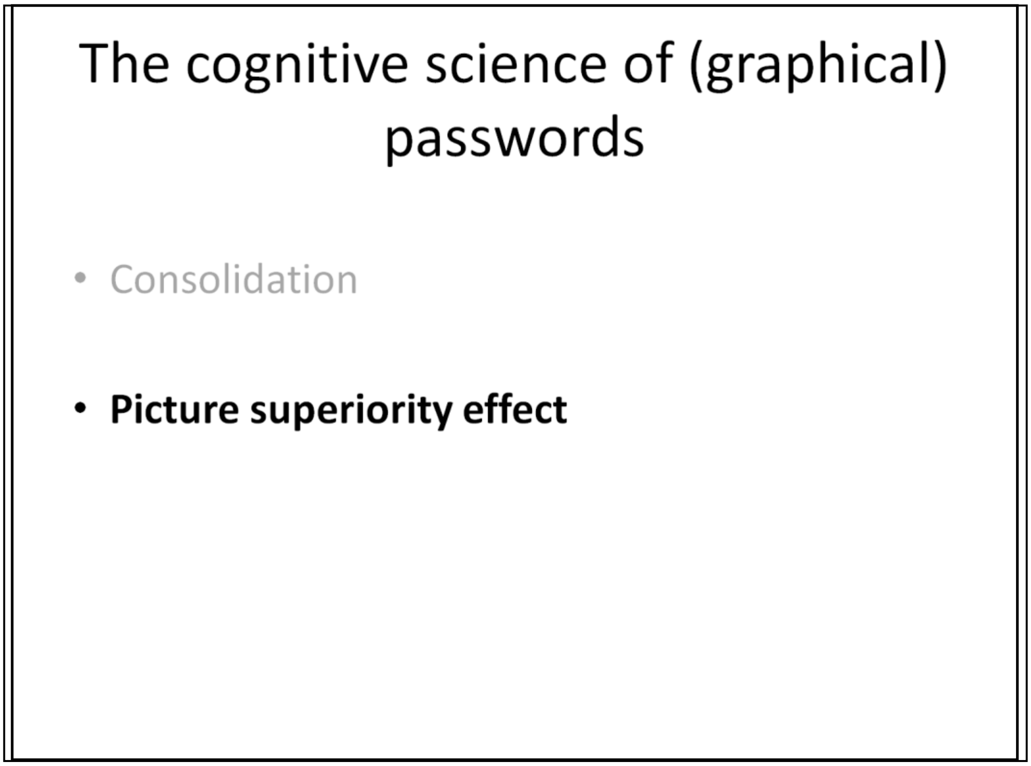
Consolidation

- 10 minutes vs. 1 minute
- Gives graphical-password systems an advantage

Consolidation is a term from neuroscience that describes how memories can be strengthened over time...

-graphical passwords often have training interfaces
-holding all else equal, 10 minutes will always win

This is straightforward.

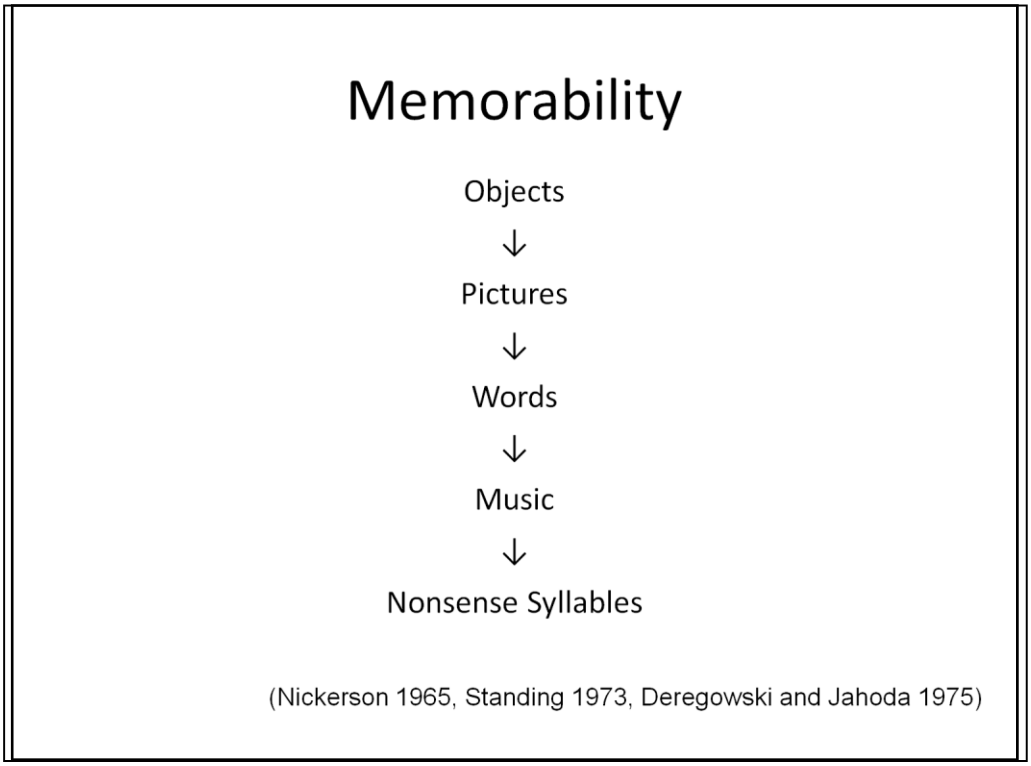


The cognitive science of (graphical) passwords

- Consolidation

- **Picture superiority effect**

Now lets talk about picture superiority. Most graphical-password systems use pictures because of this effect.



Memorability

Objects

↓

Pictures

↓

Words

↓

Music

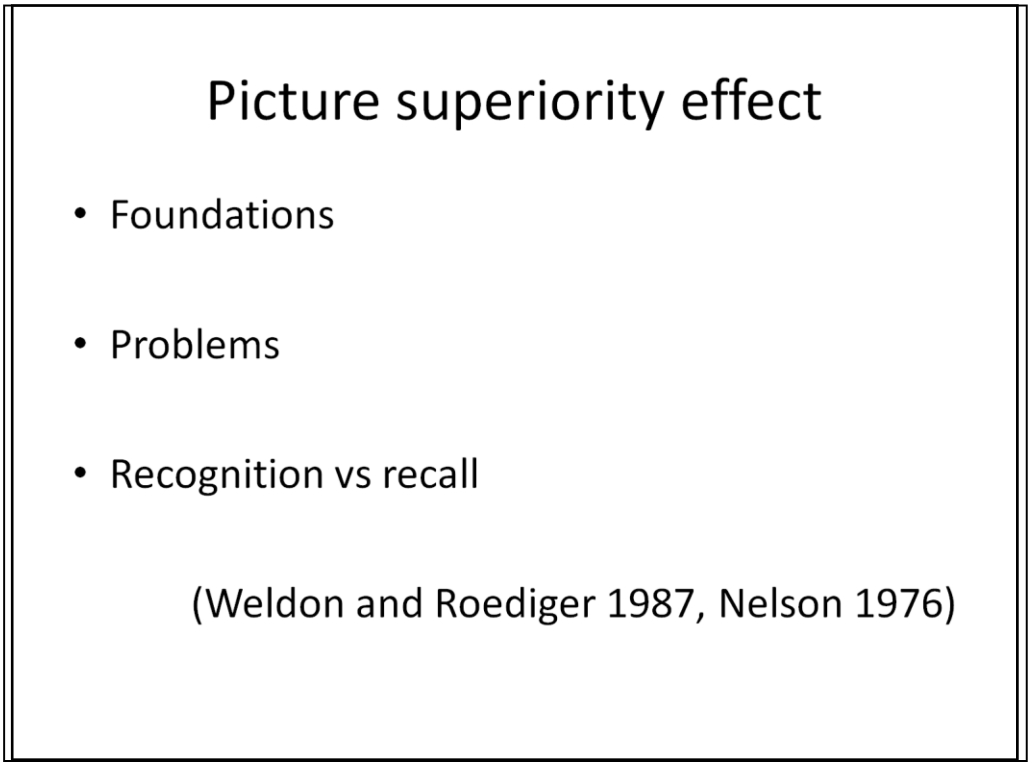
↓

Nonsense Syllables

(Nickerson 1965, Standing 1973, Deregowski and Jahoda 1975)

The PSE is a heavily studied and verified phenomenon in psychology which states that pictures are remembered better than words.

-continuum (transitivity)
-what makes items memorable?

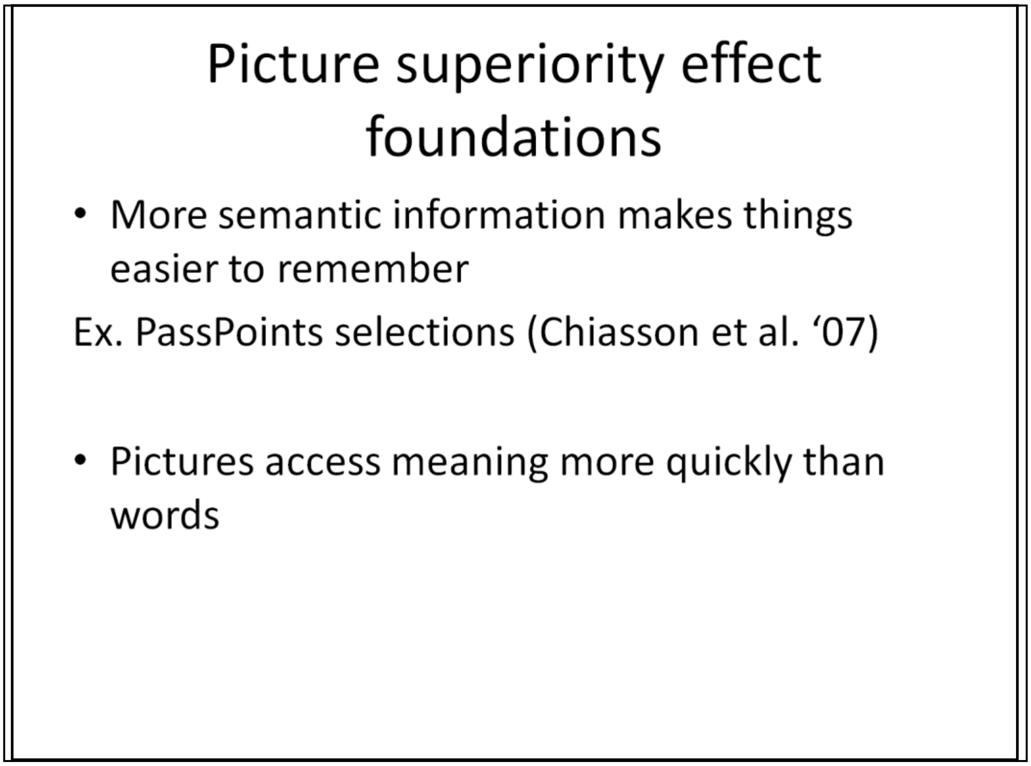


Picture superiority effect

- Foundations
- Problems
- Recognition vs recall

(Weldon and Roediger 1987, Nelson 1976)

There are many facets to the PSE that have been experimentally verified. I am going to run through them now because they all impact memory for pictures in different ways.



Picture superiority effect foundations

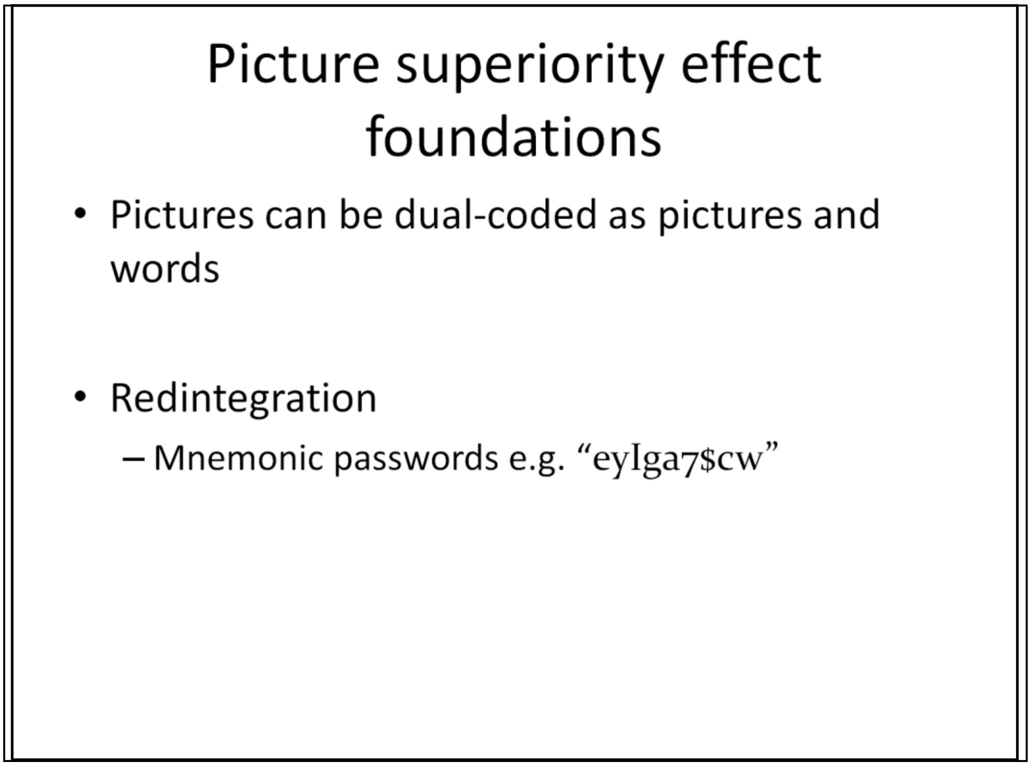
- More semantic information makes things easier to remember

Ex. PassPoints selections (Chiasson et al. ‘07)

- Pictures access meaning more quickly than words

This can be seen in the paper assigned for today (Passpoints)…

A picture of an apple is easier to remember than the word “apple” (but only if you try to remember the word “apple” and not the thing “apple”)

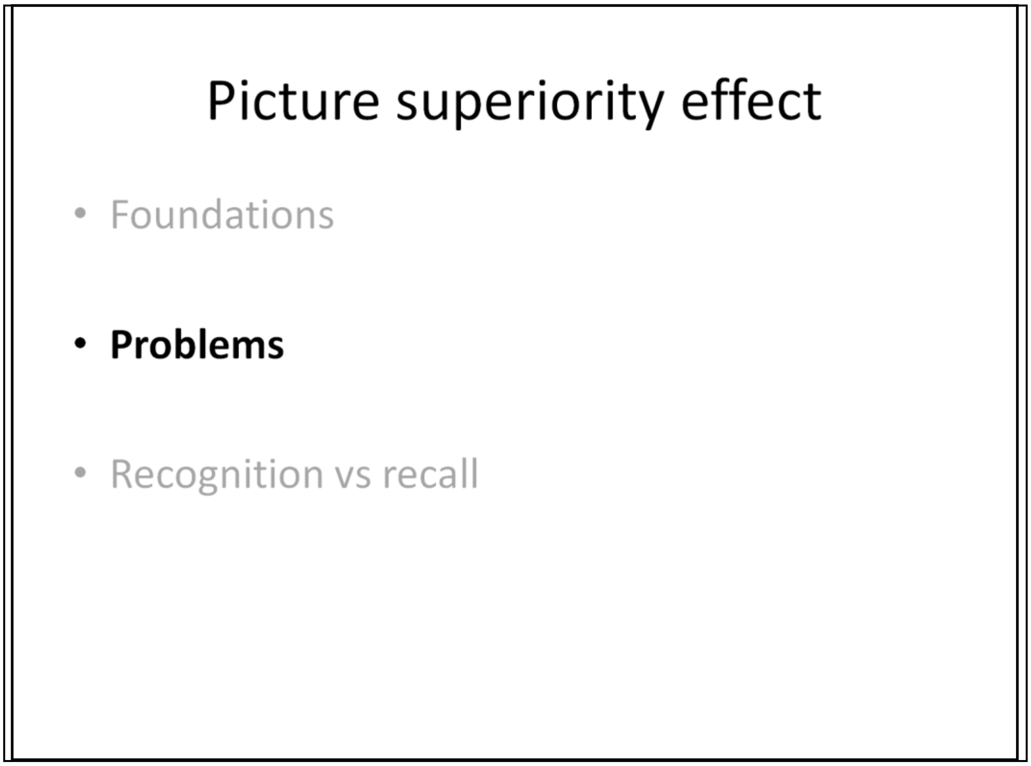


Picture superiority effect foundations

- Pictures can be dual-coded as pictures and words
- Redintegration
  - Mnemonic passwords e.g. "eyIga7$cw"

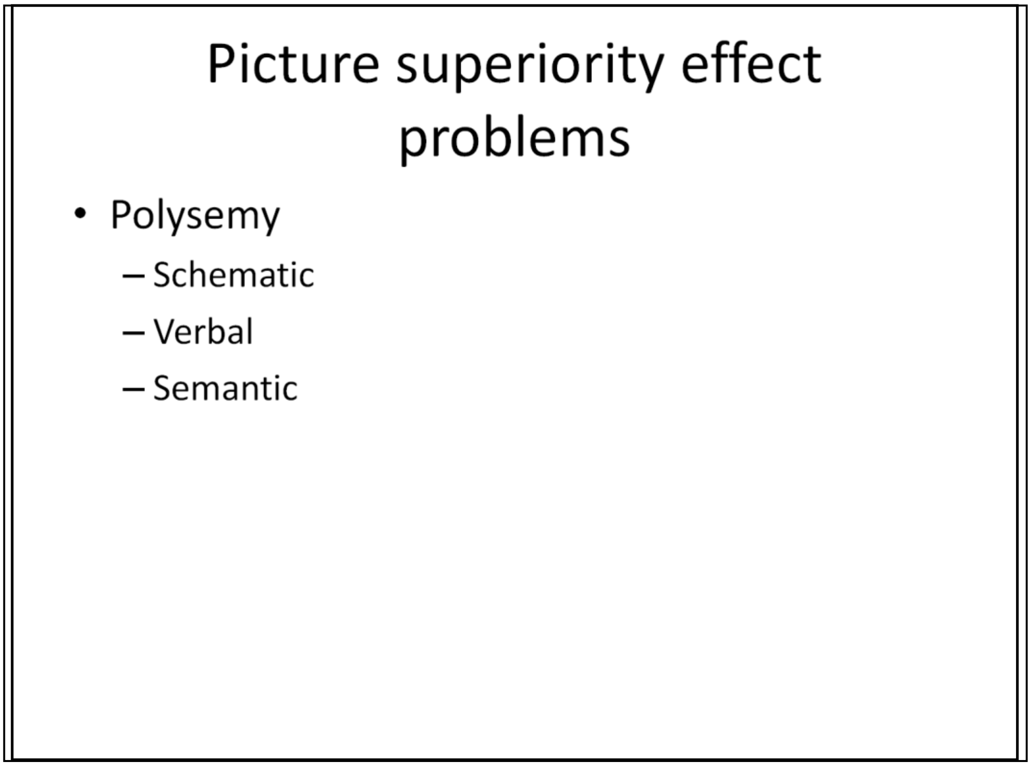
This was a popular theory during the 70s and 80s but has since been mostly refuted…

… but there is evidence that multiple encodings encourage redintegration…
- mnemonic passwords and muscle memory



Picture superiority effect

- Foundations
- **Problems**
- Recognition vs recall

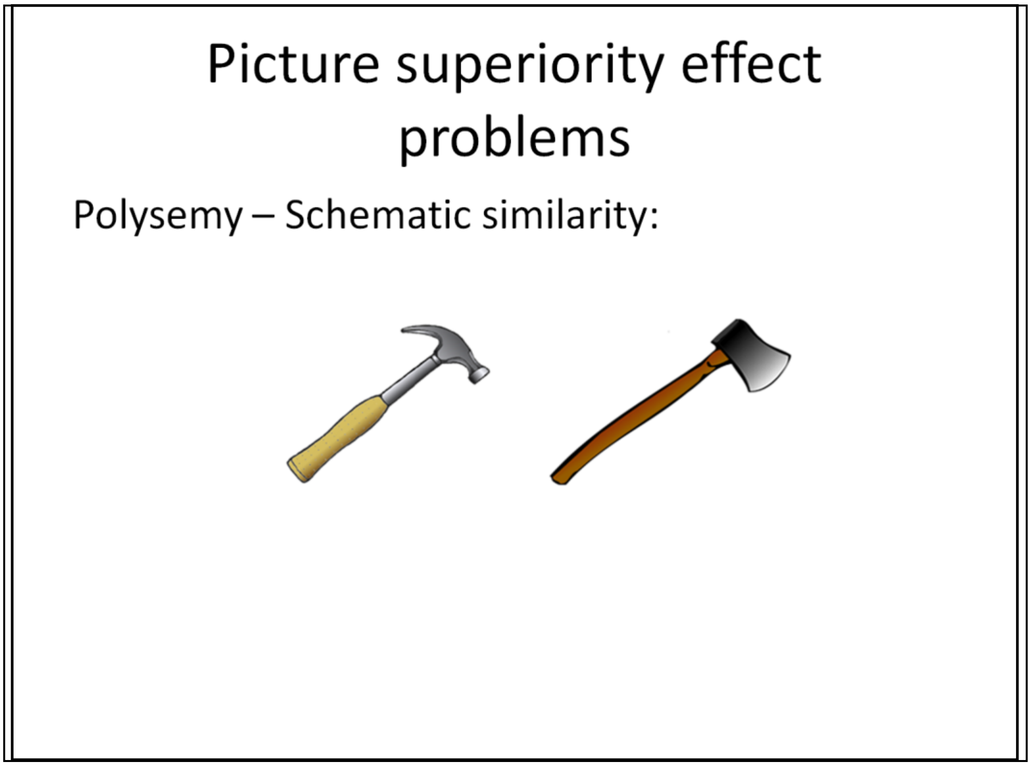


Picture superiority effect problems

- Polysemy
  - Schematic
  - Verbal
  - Semantic

Polysemy is a major problem with pictures and it has to do with similarity. It is hard to remember things that don’t stand out.

Or, if you have to remember a subset of items in a larger set, the items can be confused on these three levels.

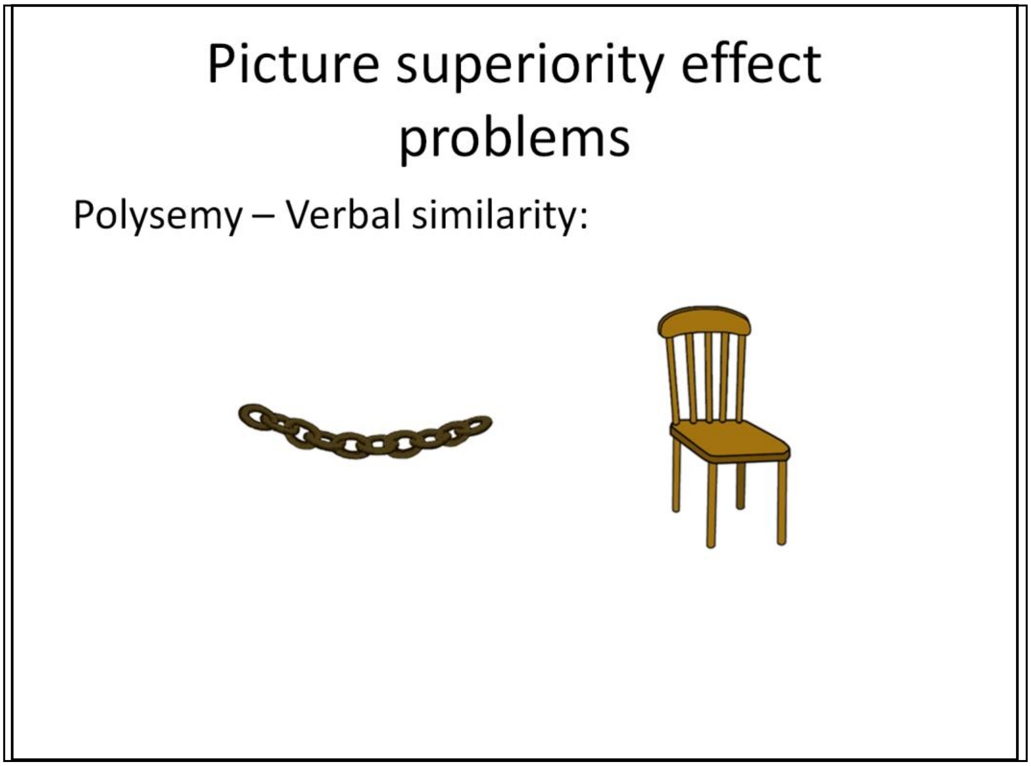


Picture superiority effect problems

Polysemy – Schematic similarity:

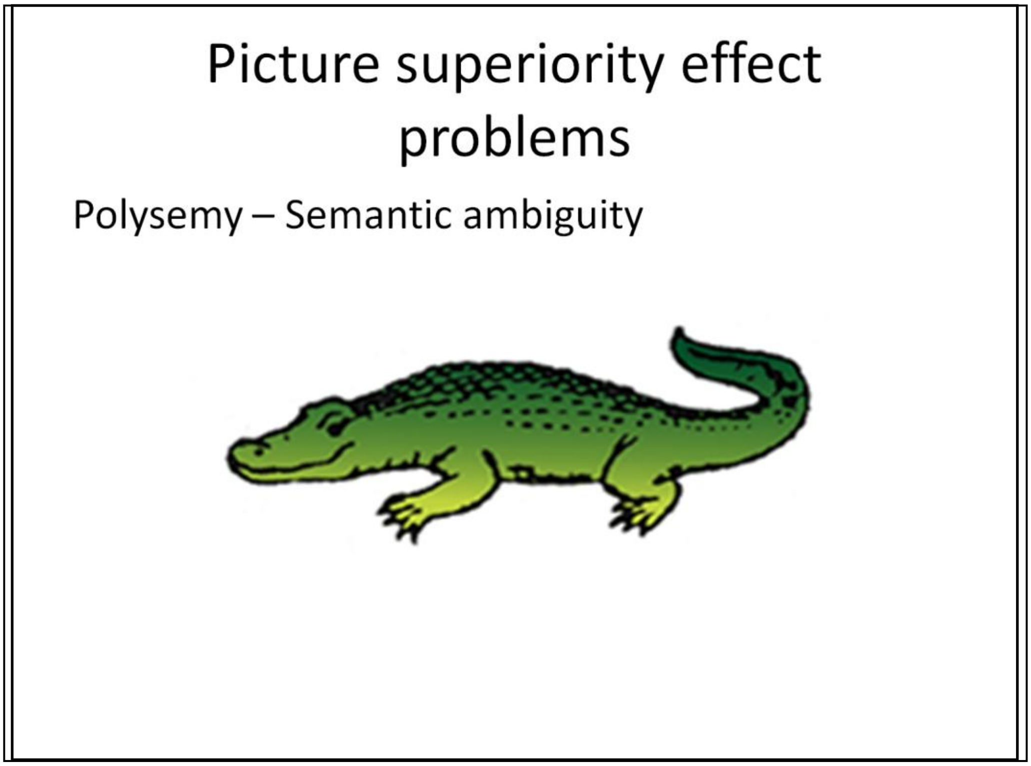
This is a major problem with pictures and it has to do with similarity.

This is an example of schematic similarity



Picture superiority effect problems

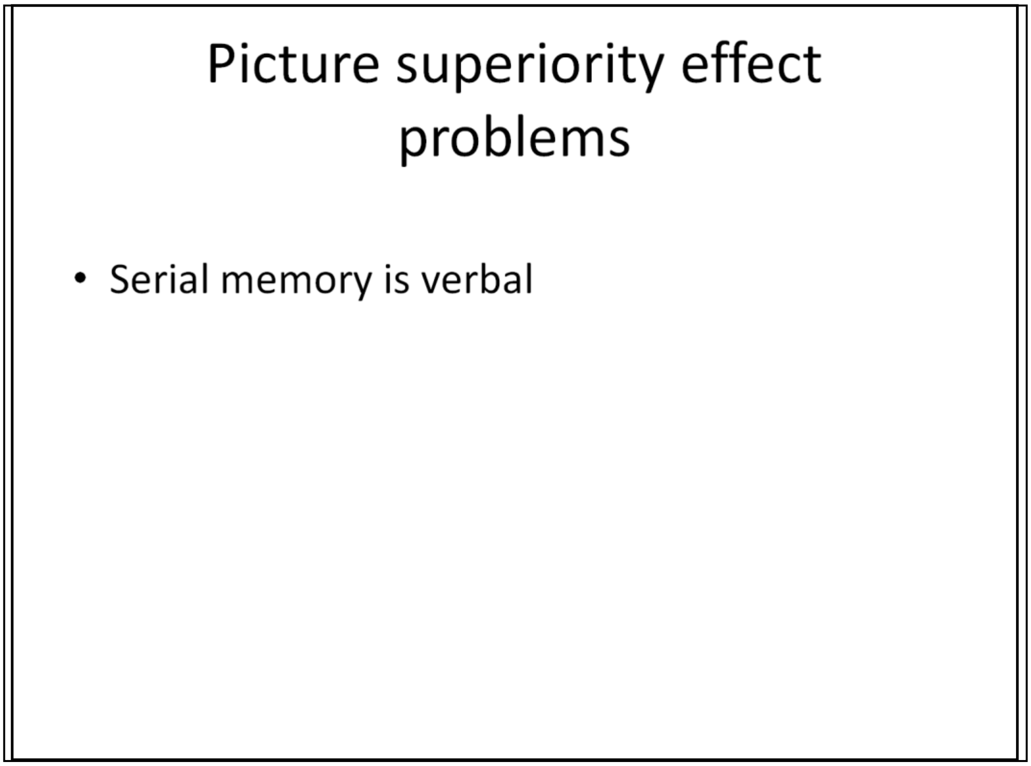
Polysemy – Verbal similarity:



Picture superiority effect problems

Polysemy – Semantic ambiguity

What is this a picture of? How many think it’s a crocodile? How many think it’s an alligator? How many are not sure?...



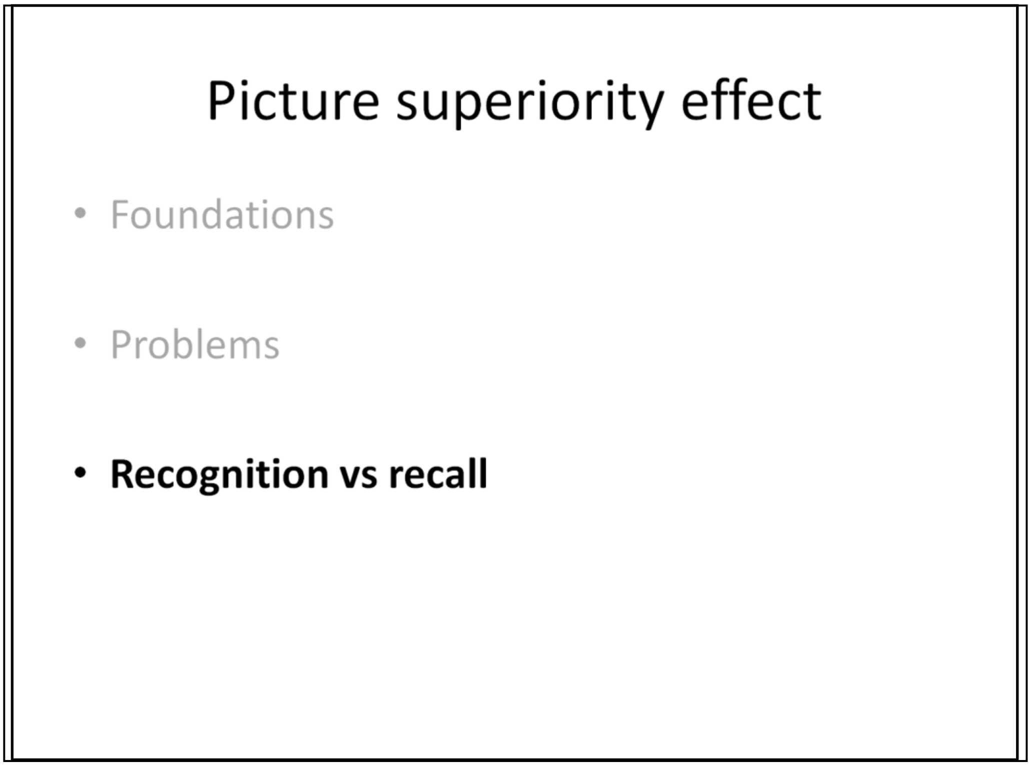
Picture superiority effect problems

- Serial memory is verbal

Even though pictures have all these features that can make them easy to remember, when you apply them to passwords you can run into a problem.

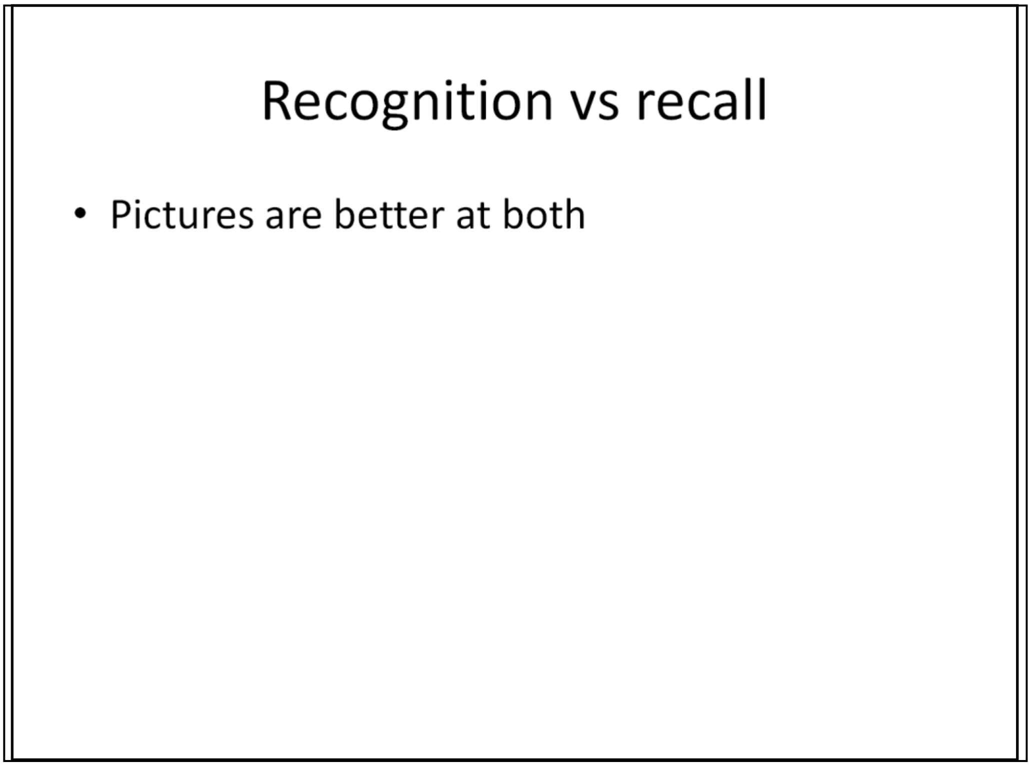
If you want to remember pictures in a specific order you have to work with serial memory (native ASL-signers story).

-impact on PassPoints
-unordered passwords



Picture superiority effect

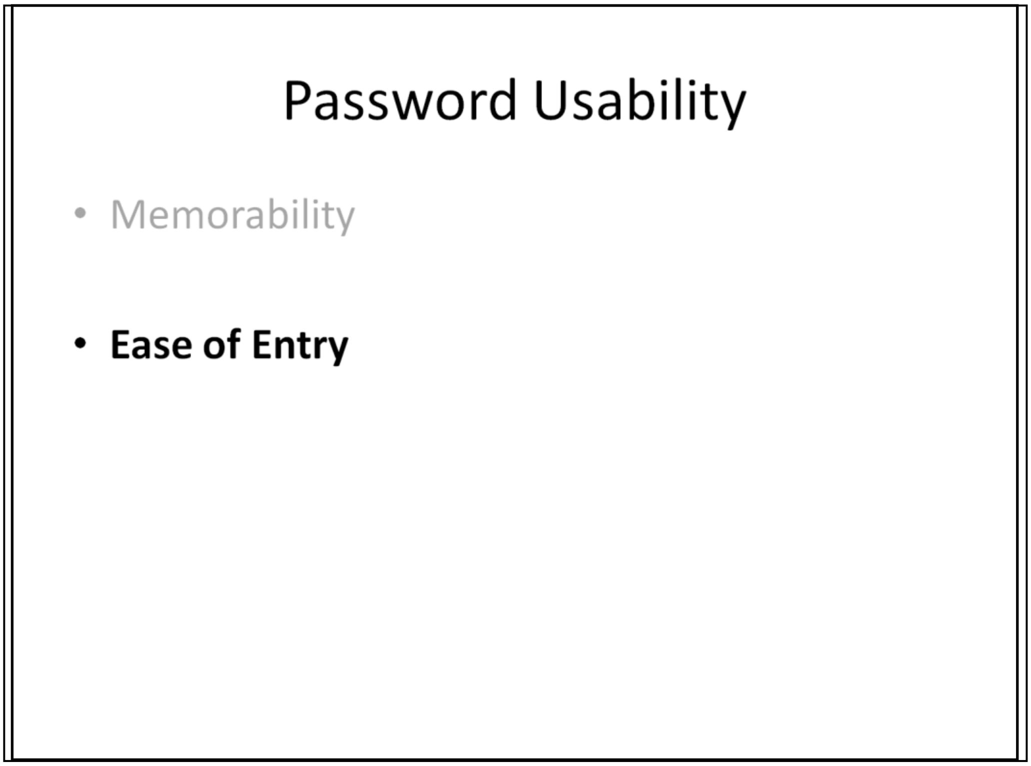
- Foundations
- Problems
- **Recognition vs recall**



Recognition vs recall

- Pictures are better at both

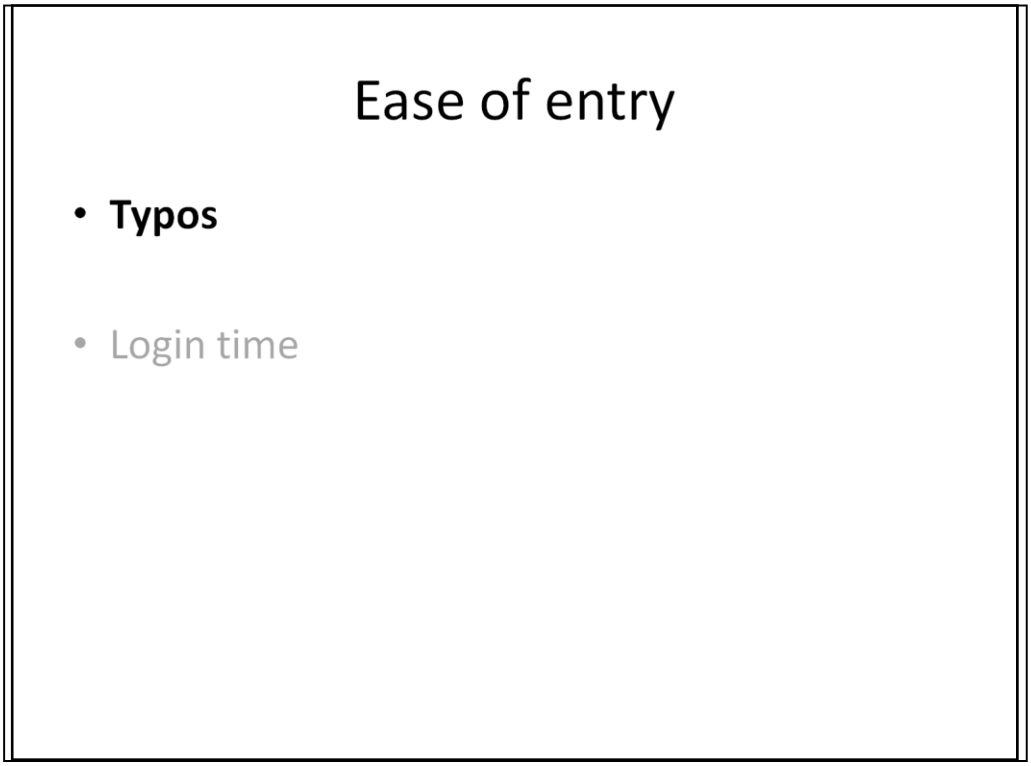
A lot of graphical password systems rely on recognition, but pictures are actually better than text at both. In fact, the relative advantage of pictures in recall tasks is greater than their advantage for recognition tasks (though recognition always performs better than recall).



Password Usability

- Memorability

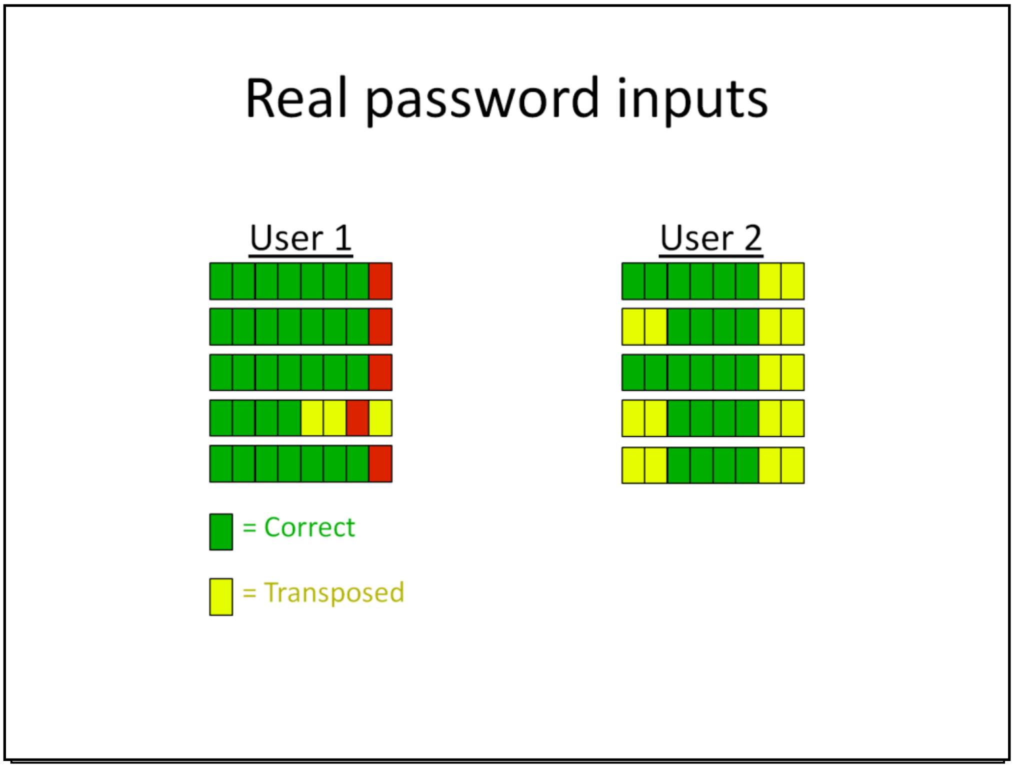
- **Ease of Entry**



Ease of entry

- **Typos**
- Login time

-Passwords and lack of feedback.
-Repeated input problem



Real password inputs

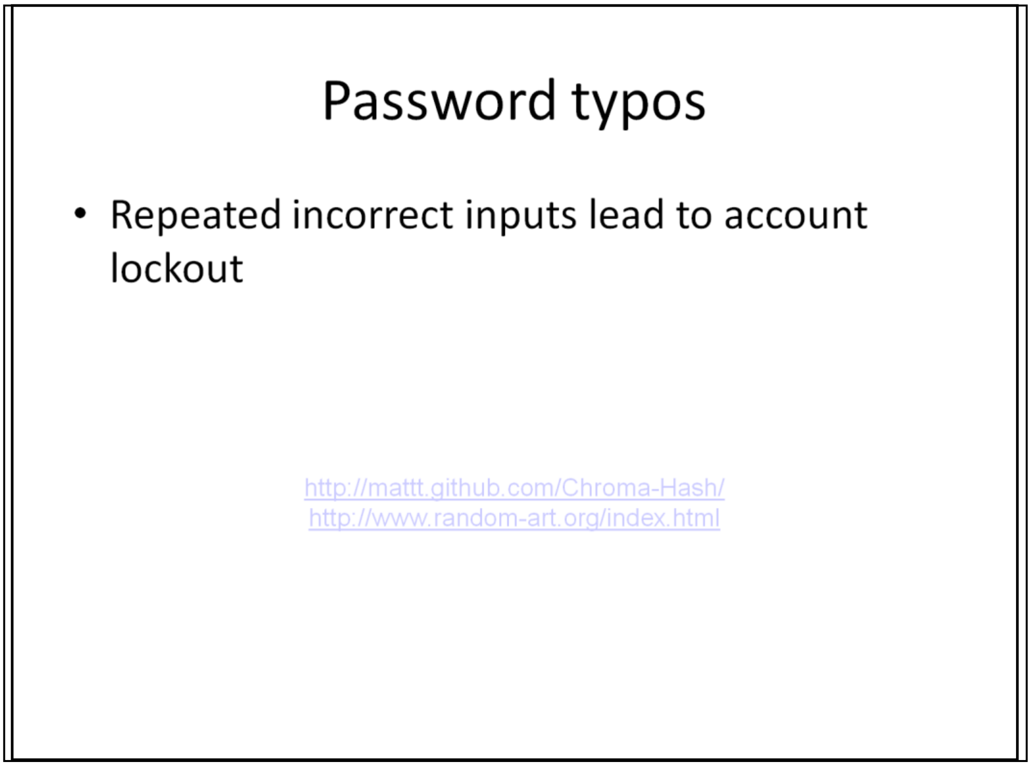
User 1

User 2

🟩 = Correct

🟨 = Transposed

-inputs from study
-best example
-Why is this a problem?

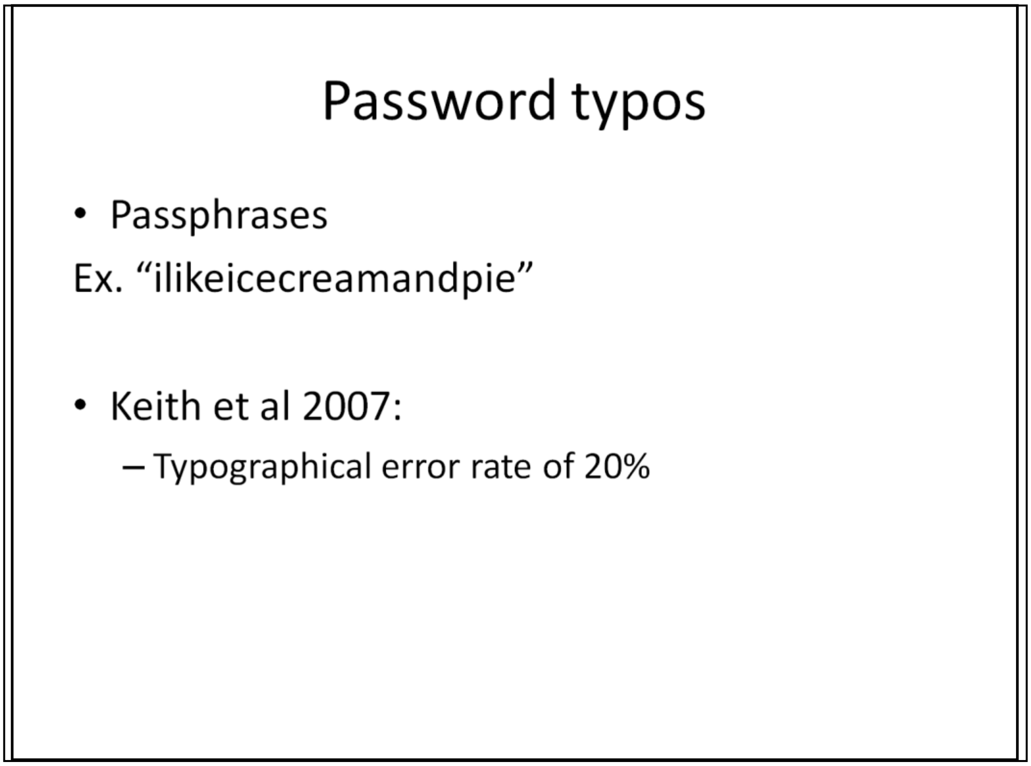


Password typos

- Repeated incorrect inputs lead to account lockout

http://mattt.github.com/Chroma-Hash/
http://www.random-art.org/index.html

-attacker inputs vs innocent user



Password typos

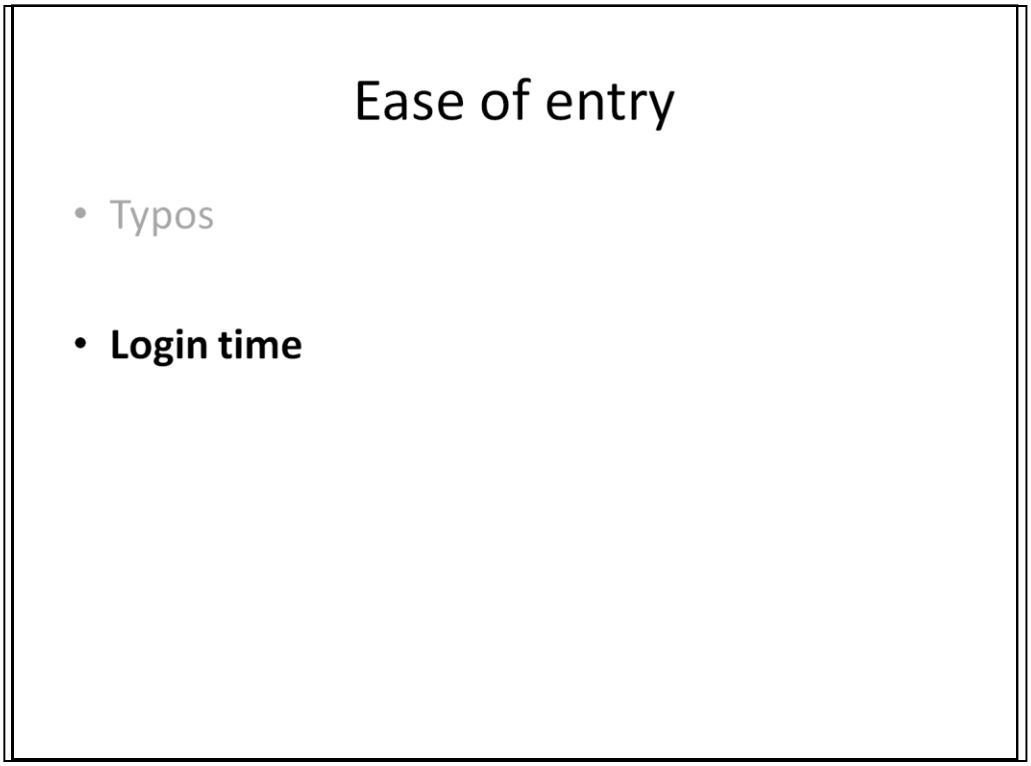
- Passphrases

Ex. “ilikeicecreamandpie”

- Keith et al 2007:
  - Typographical error rate of 20%

In Chapter 7 of our book, the authors use the term passphrase to refer to something that I call a mnemonic password, but I don’t think that is typical. Here I’m referring to a password that is composed of several words strung together.

-Passphrase length vs brute force
-Passphrase and semantic units
-Passphrases and entropy
-Passphrases and typos (typographical error rate of 20% for 15-character passphrases)



Ease of entry

- Typos
- **Login time**

Login time is time to a successful login.

Login time is relevant to passphrases because more characters takes longer to input. It’s also relevant because typos mean the user has to try again and this increases login time.

Graphical password systems often have novel input methods and several screens.

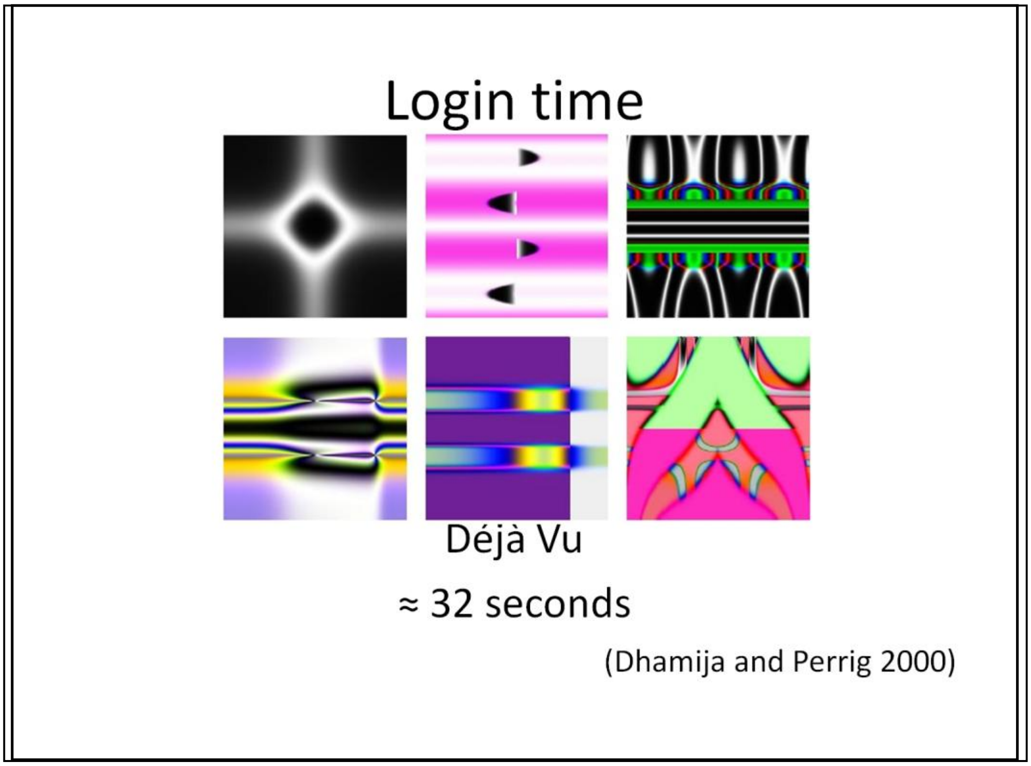
Login time

PassPoints

≈ 14 seconds

(Wiedenbeck et al. 2005)

PassPoints has the user click five points in order on a single image.



Login time

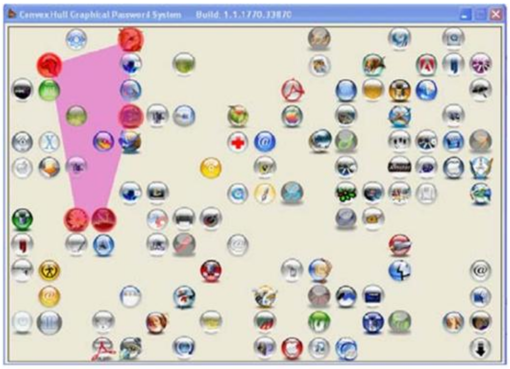
Déjà Vu

≈ 32 seconds

(Dhamija and Perrig 2000)

The Déjà vu system has the user select 5 images from their portfolio (I’ll talk about that later) from a set of 25 that are presented. The other 20 are “decoys”.

Login time



Convex Hull Click

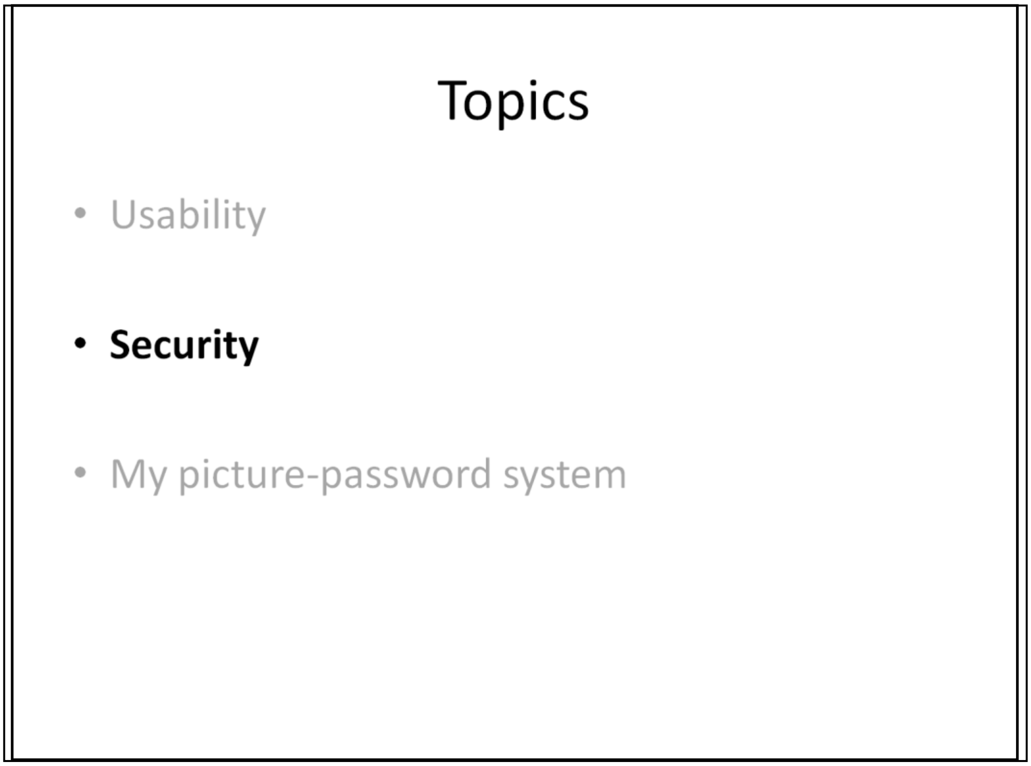
≈ 72 seconds

(Wiedenbeck 2006)

Click 5 times.

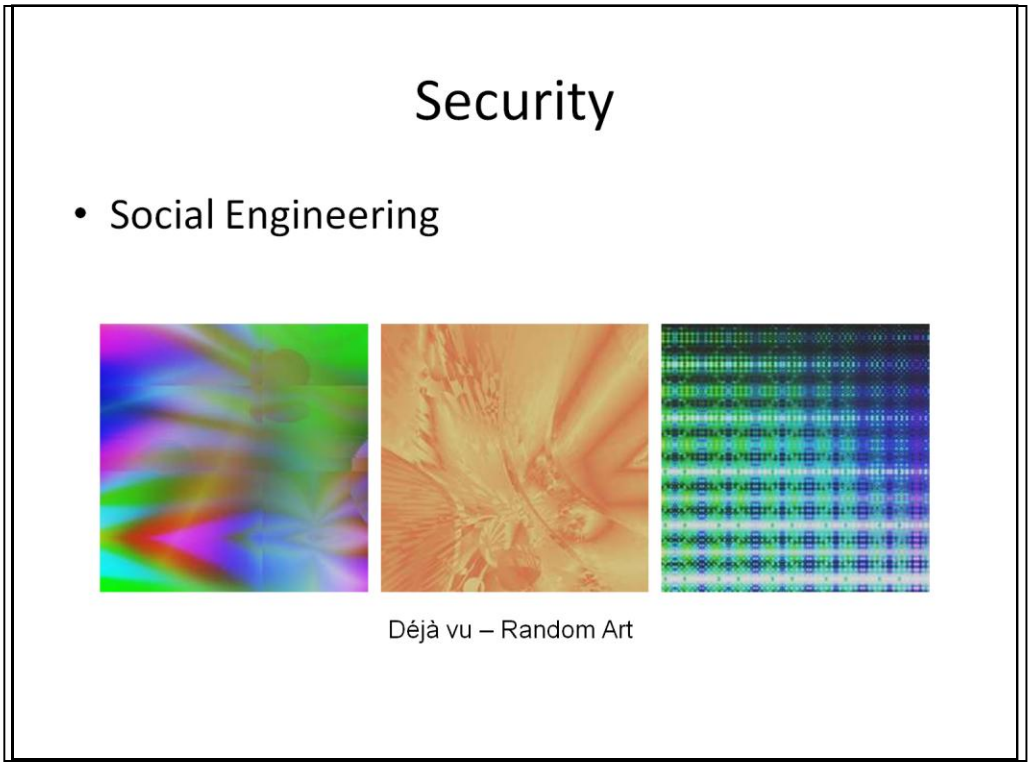
-password is 5 icons, systems shows 3-5 per round, 5 rounds to authenticate.

-game-like system, animation



Topics

- Usability
- **Security**
- My picture-password system



Security

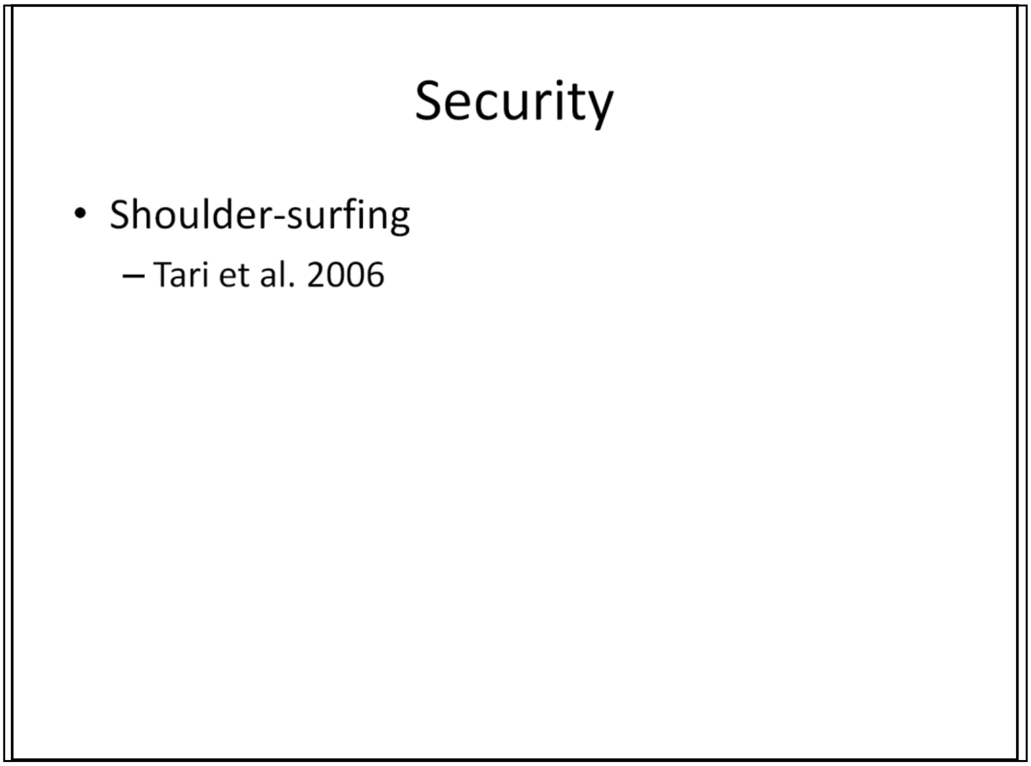
- Social Engineering

Déjà vu – Random Art

The déjà vu system uses “random art” (an algorithmic way to generate nonrepresentational art images)

A benefit of using images like this is that, seemingly, they cannot be written down.

(discuss) Can they be written down?
(discuss) Does this solve the social engineering problem?



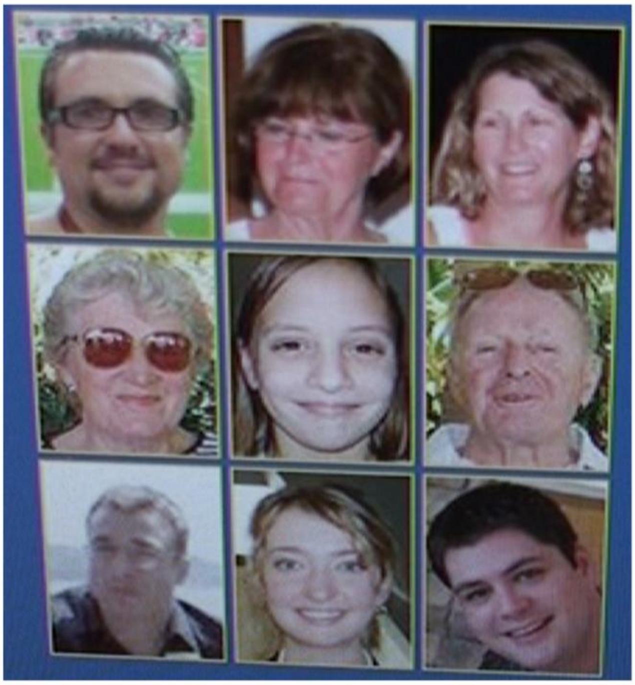
Security

- Shoulder-surfing
  - Tari et al. 2006

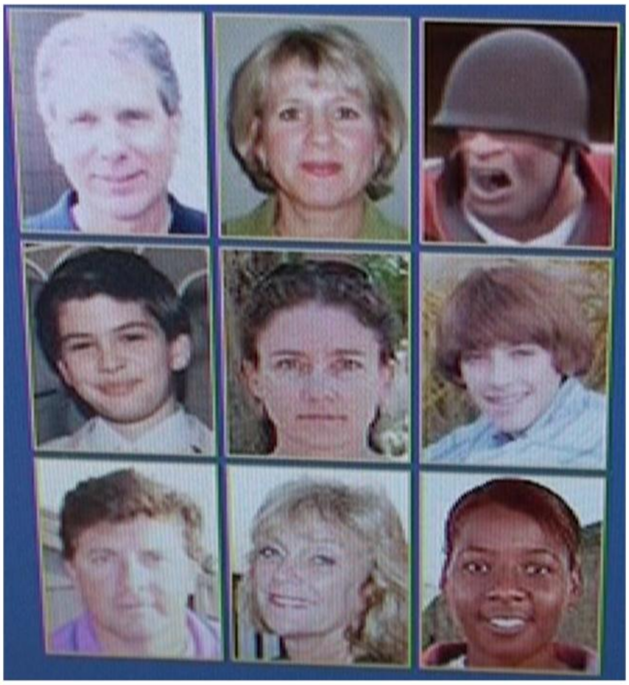
The contest.

Dictionary word = 1 point
Strong password = 2 points
Passfaces password = 2 points
Cristian’s password = 5 points

- Paper found PassFaces extremely hard to surf but the current version of PassFaces required inclusion of my own pictures which should make it easier.

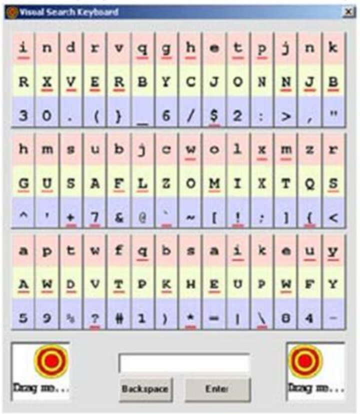


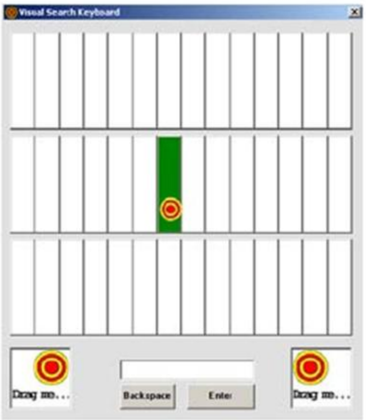
Pictures of PassFaces screens



Shoulder surfing

The spy-resistant keyboard

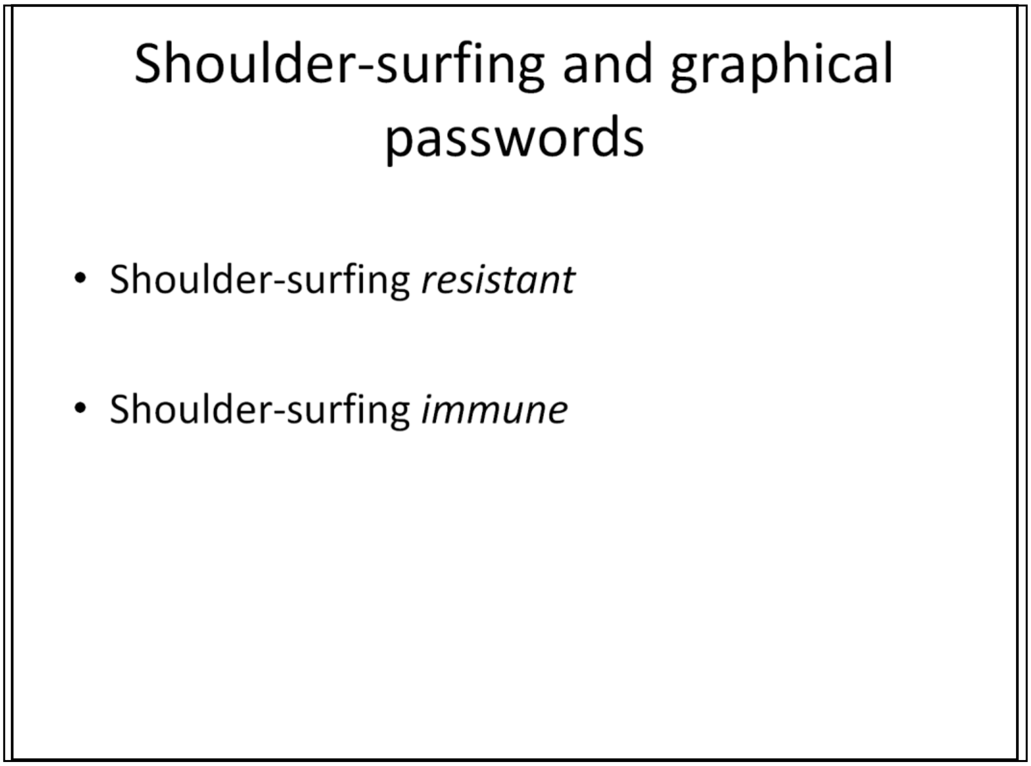

Visual Search Keyboard
i n d r v q g h e t p j n k
R X V E R B Y C J O N N J B
3 0 . ( ) _ 6 / $ 2 : > , "
h m s u b j c w o l x m z r
G U S A F L Z O M I X T Q S
^ ' + 7 & @ ` ~ [! ;] { <
a p t w f q b s a i k e u y
A W D V T P K H E U P W F Y
5 9 % ? # 1) * = | \ 8 4 -
Drag me...
Backspace
Enter
Drag me...



(Tan et al. 2005)

This is the spy-resistant keyboard.

-same principle as Passfaces
-meant for Microsoft Surface and large touchscreen displays

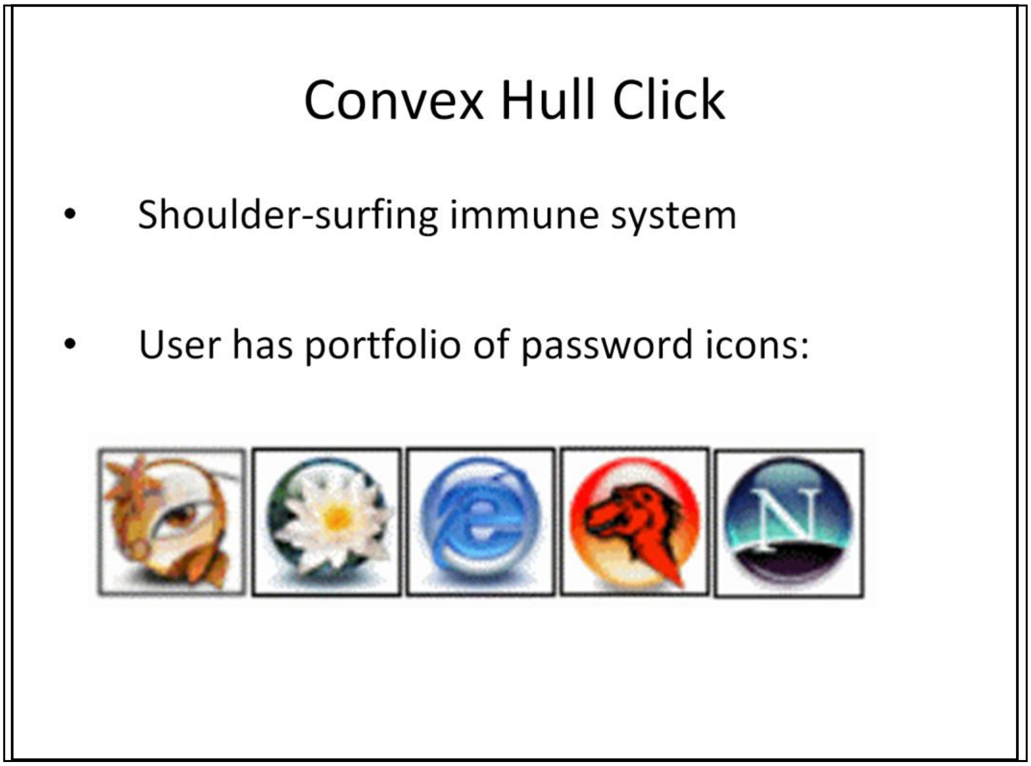


Shoulder-surfing and graphical passwords

- Shoulder-surfing *resistant*
- Shoulder-surfing *immune*

So the spy-resistant keyboard is what I would call a shoulder-surfing resistant system. If you record the authentication, you can figure out the password.

There are other systems which I would call shoulder-surfing immune. These are systems which, even if you record the authentication process, you won't be able to figure out the password.



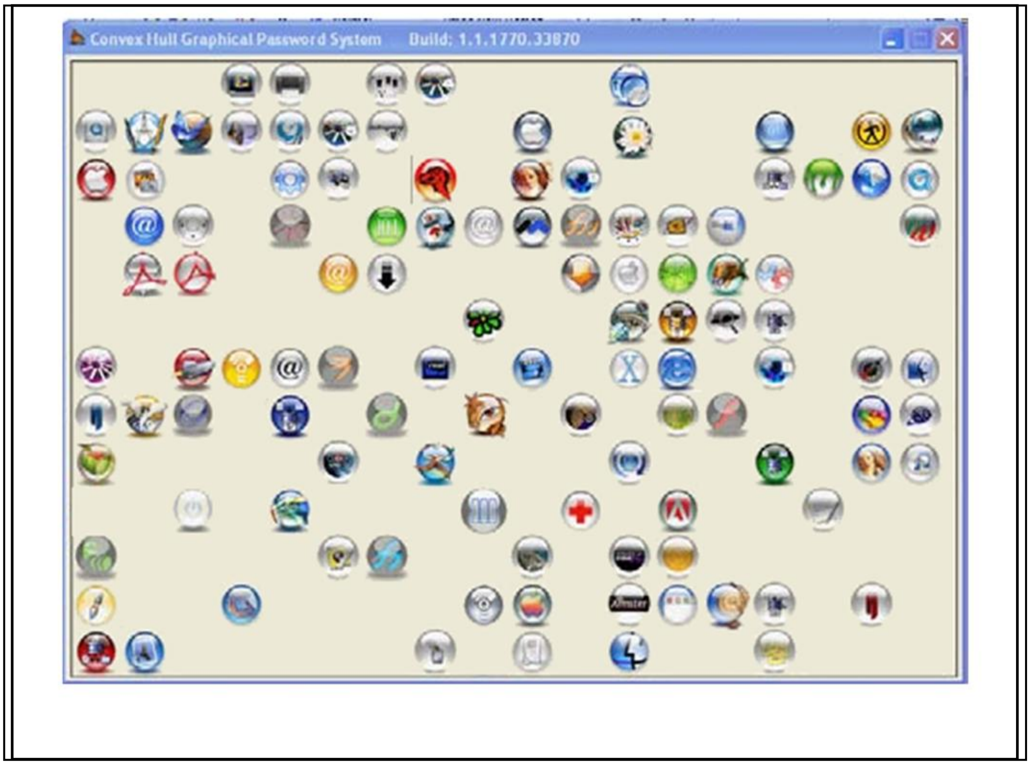
Convex Hull Click

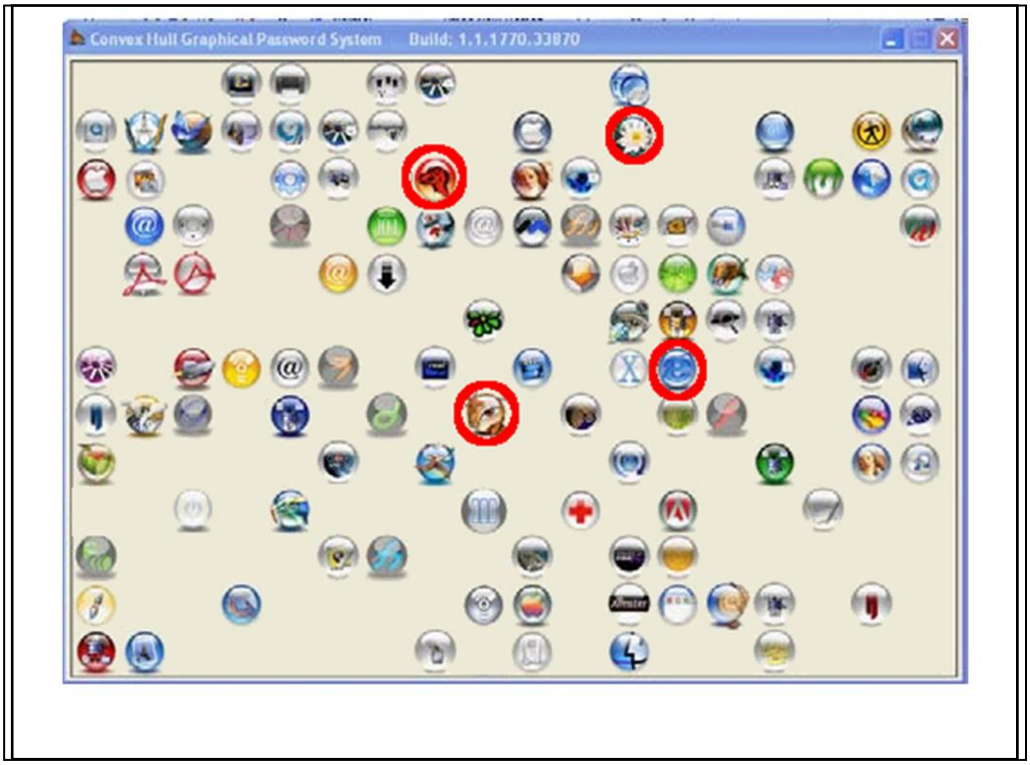
- Shoulder-surfing immune system
- User has portfolio of password icons:

This is such a system. It's calledthe Convex Hull Click system.

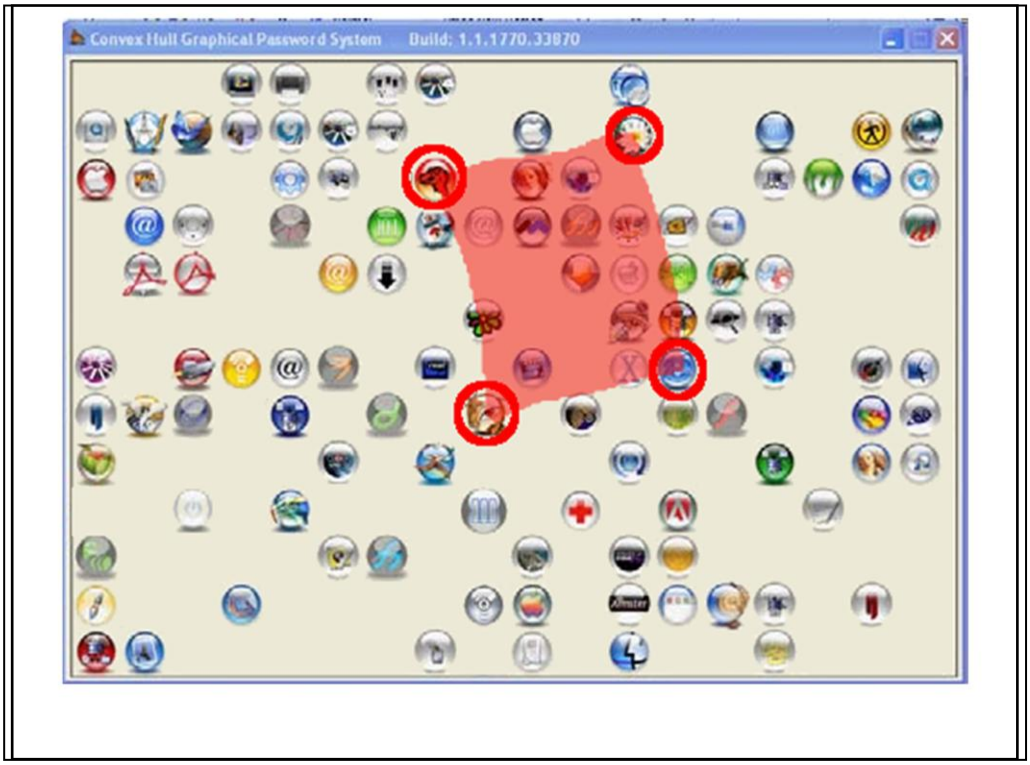
-passicons

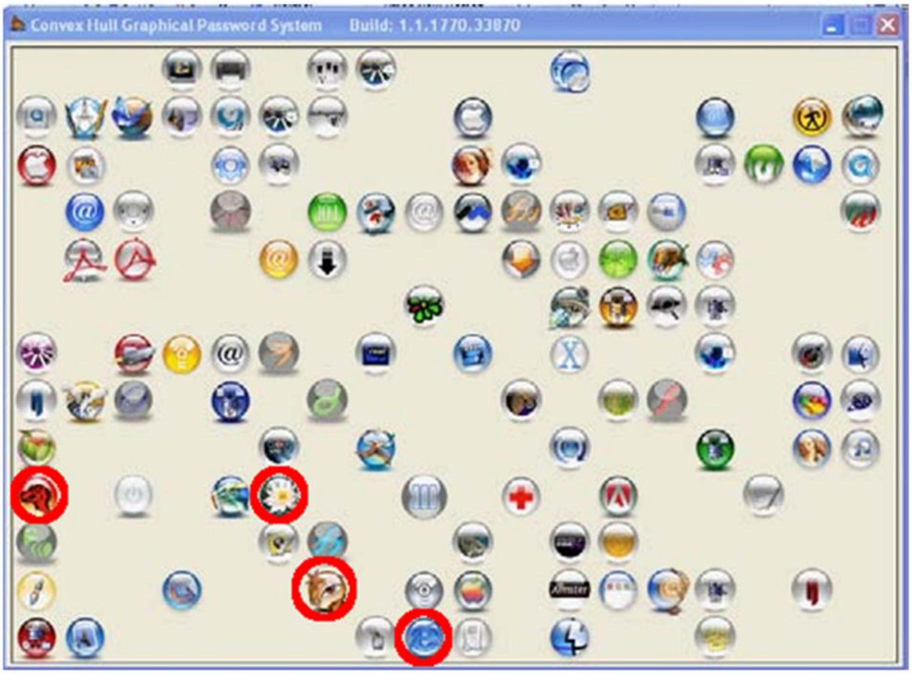
-5 screens

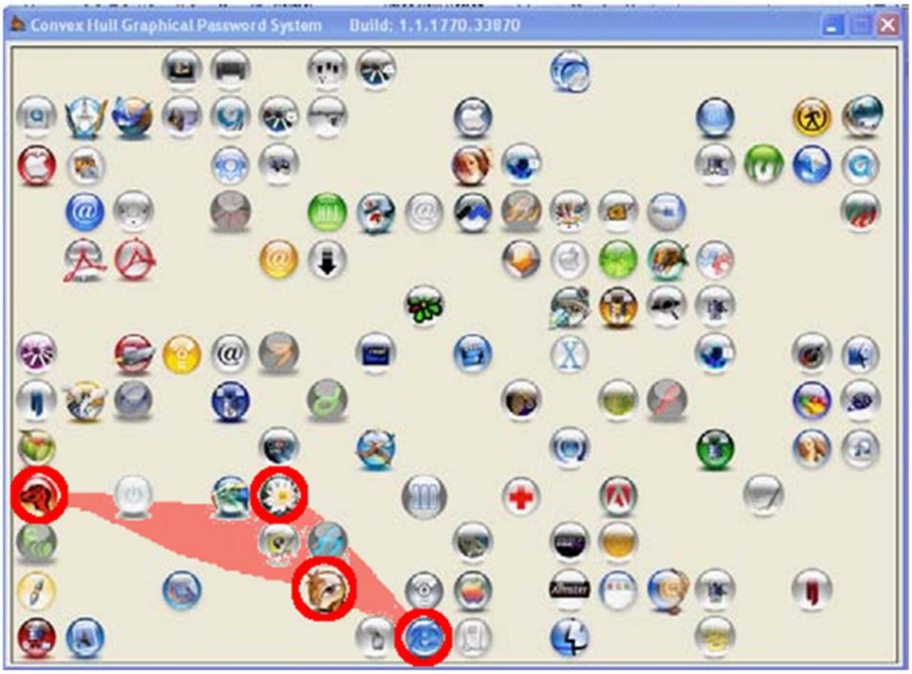

Convex Hull Graphical Password System   Build: 1.1.1770.33870



Convex Hull Graphical Password System Build: 1.1.1770.33870

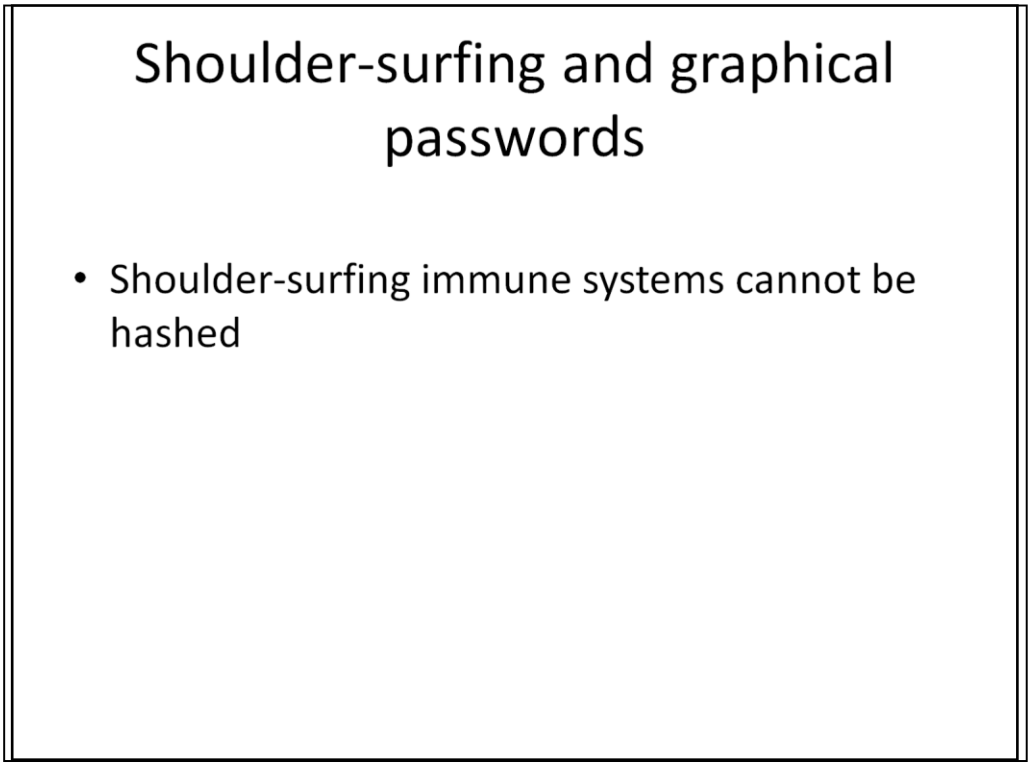

Convex Hull Graphical Password System
Build: 1.1.1770.33870


Convex Hull Graphical Password System Build: 1.1.1770.33870



Convex Hull Graphical Password System Build: 1.1.1770.33870

Success!



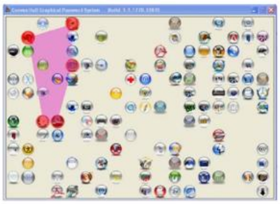
Shoulder-surfing and graphical passwords

- Shoulder-surfing immune systems cannot be hashed

-because the system needs to know the password

Hashing revisited

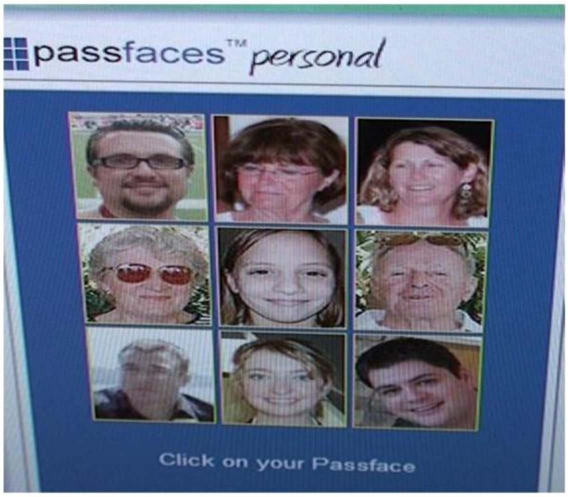
- Systems which are unhashable
  - Convex Hull Click



- Systems which don't employ hashing
  - PassPoints

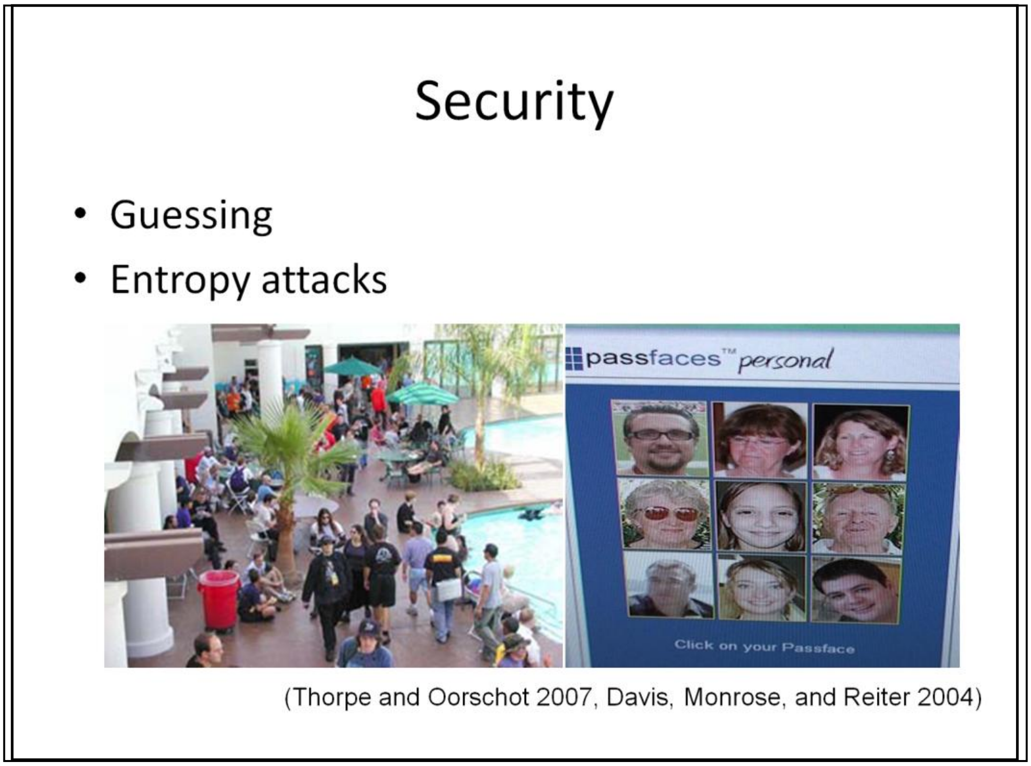
…and you have systems like convex hull click which are not hashable and some, like PassPoints that don't employ hashing.

Hashing revisited



Can a PassFaces password be hashed?

And by “hashed” I mean stored as a hash and not storing the password explicitly.

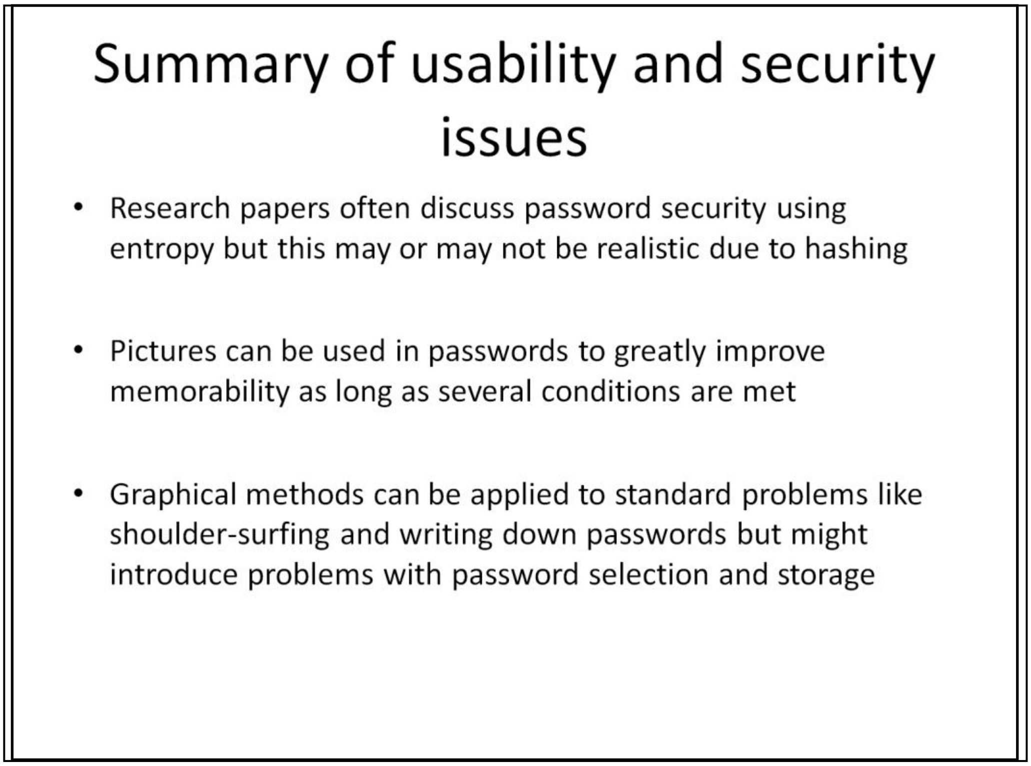


Security

- Guessing
- Entropy attacks

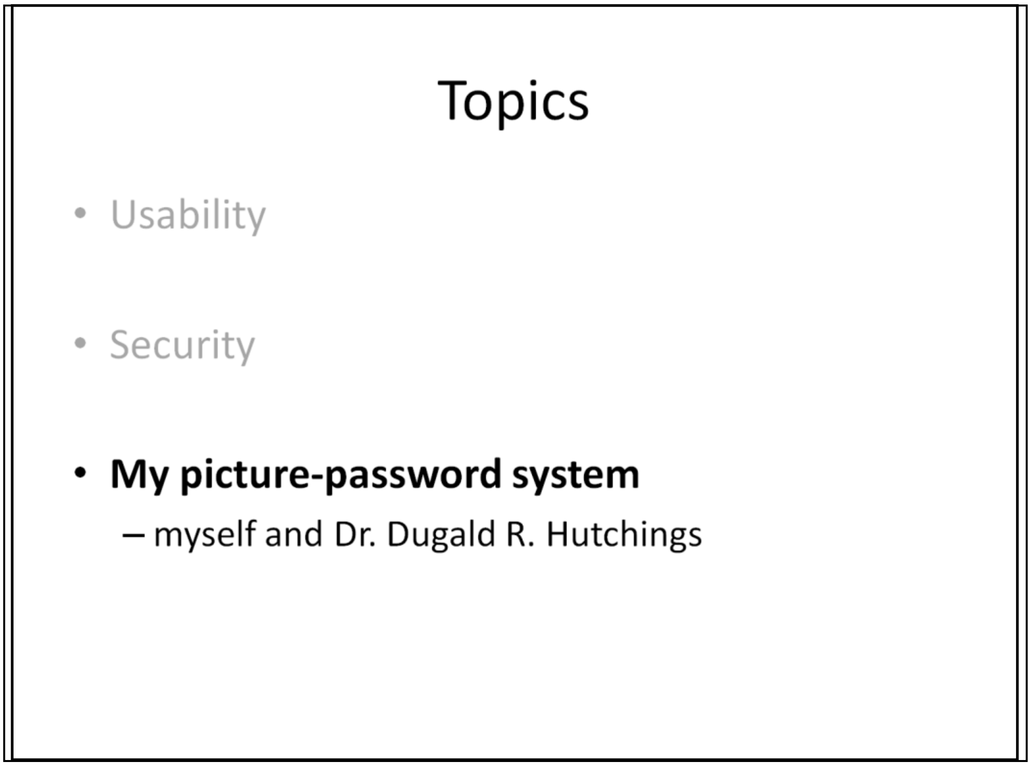
(Thorpe and Oorschot 2007, Davis, Monrose, and Reiter 2004)

Both PassFaces and PassPoints have been studied from a security perspective and in both cases, user-selected passwords were easily guessed.



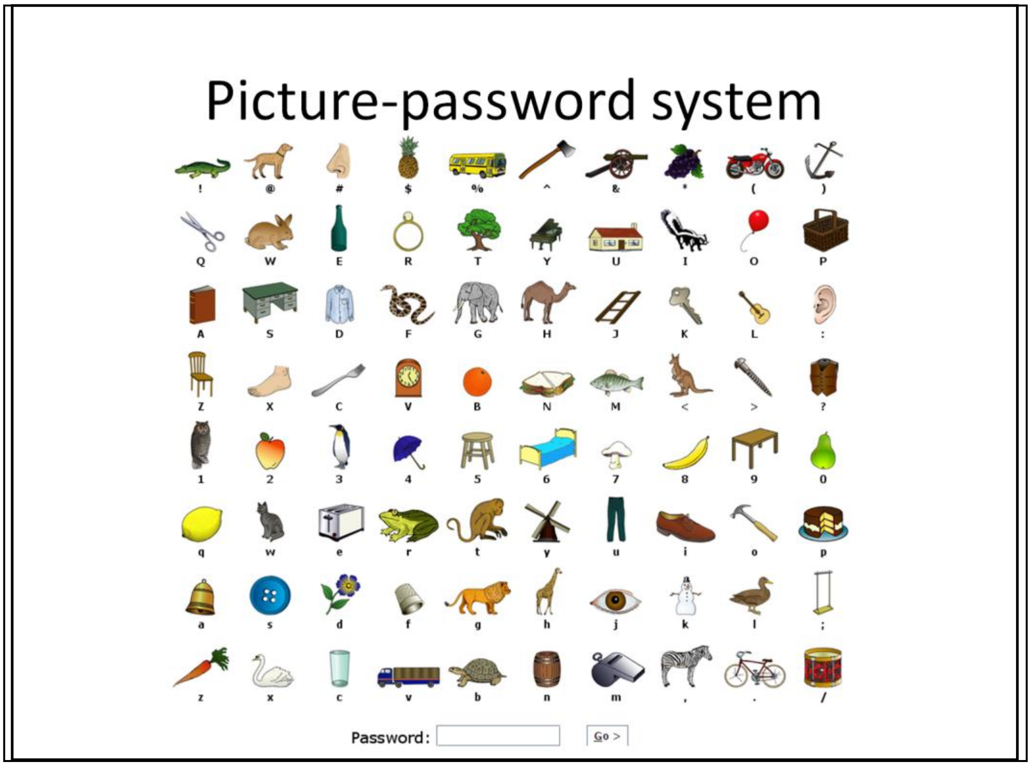
Summary of usability and security issues

- Research papers often discuss password security using entropy but this may or may not be realistic due to hashing
- Pictures can be used in passwords to greatly improve memorability as long as several conditions are met
- Graphical methods can be applied to standard problems like shoulder-surfing and writing down passwords but might introduce problems with password selection and storage



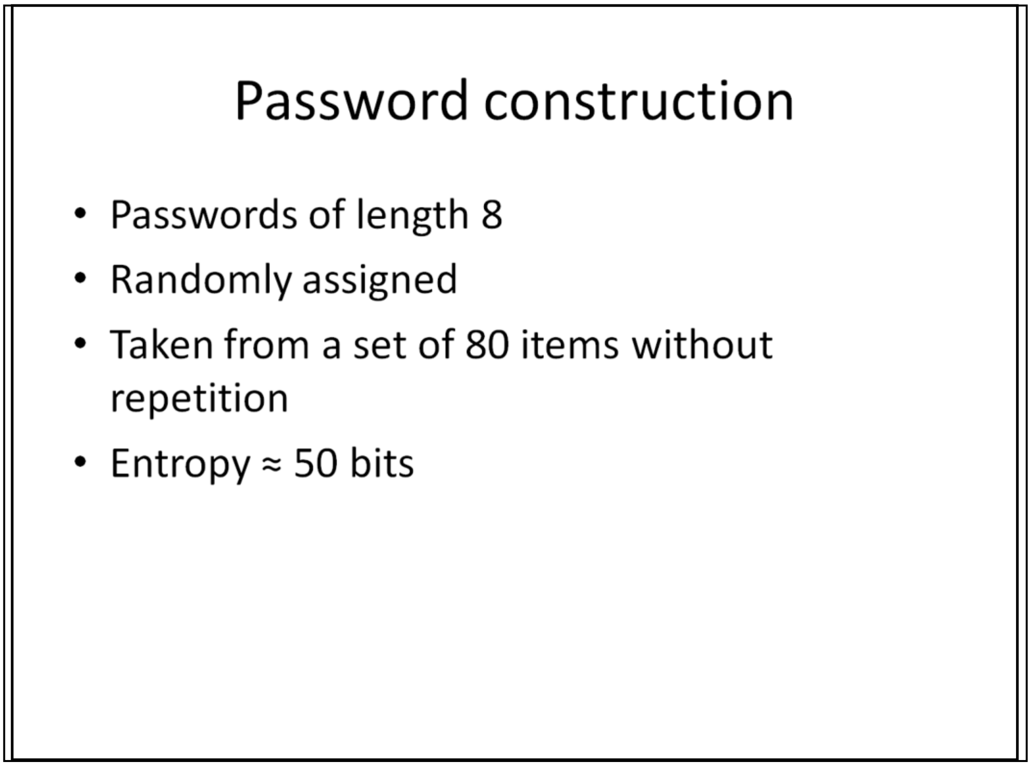
Topics

- Usability
- Security
- **My picture-password system**
  - myself and Dr. Dugald R. Hutchings



Picture-password system

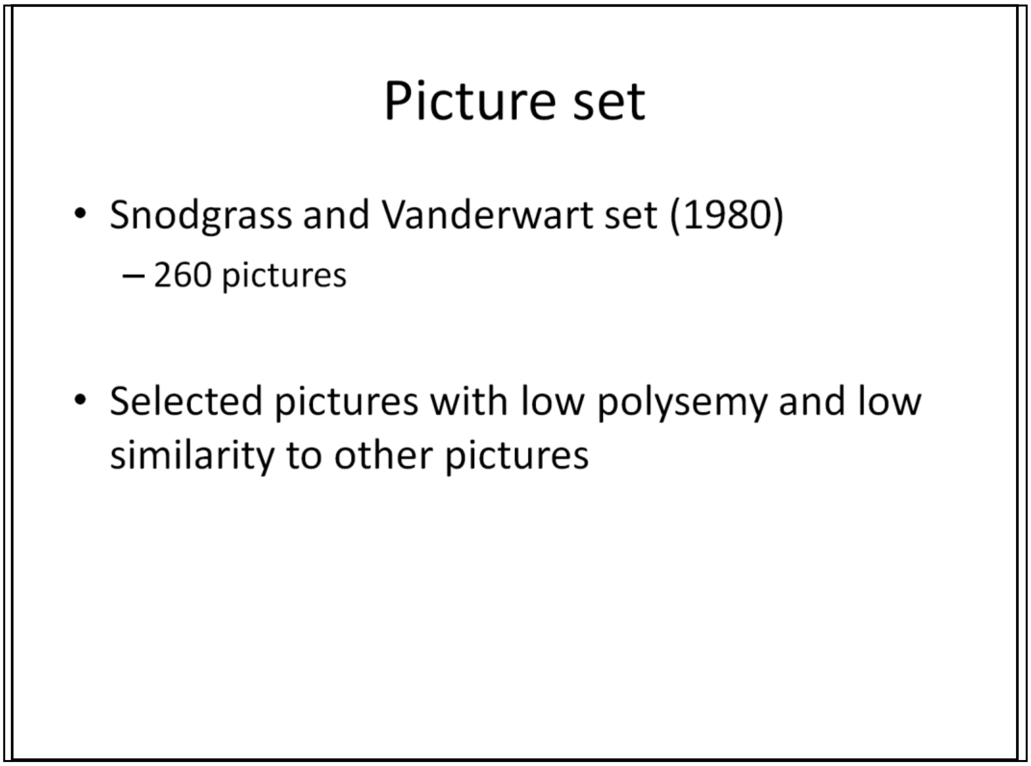
This is the login screen for my picture password system. My goal was to try to design the best password system.



Password construction

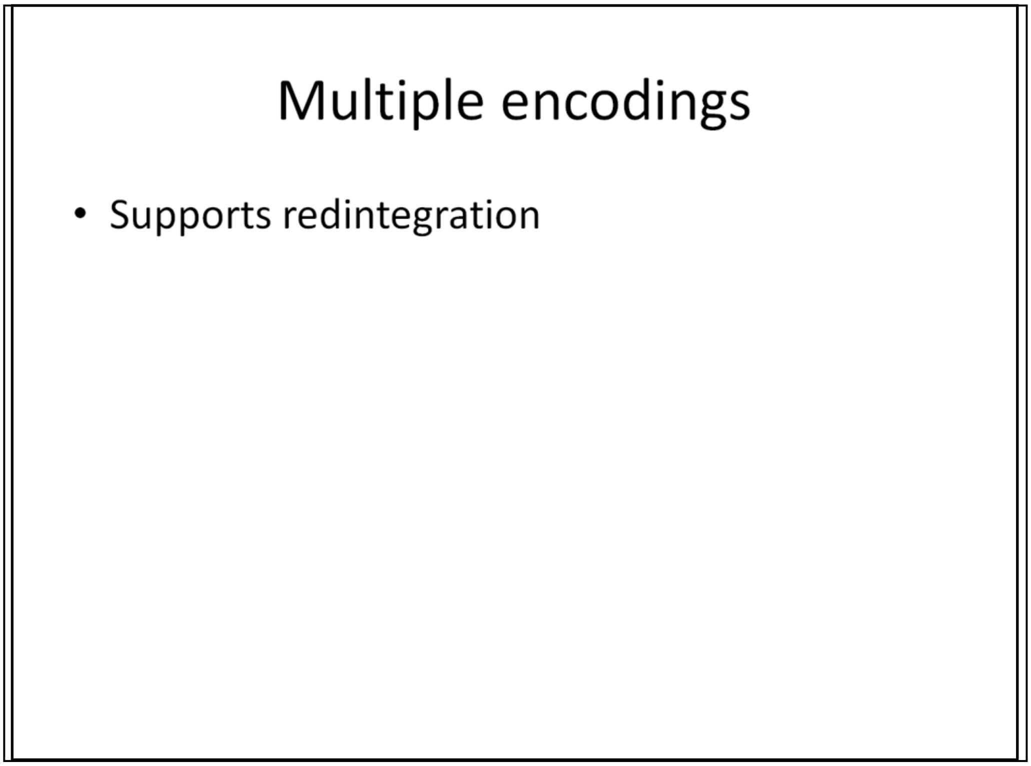
- Passwords of length 8
- Randomly assigned
- Taken from a set of 80 items without repetition
- Entropy ≈ 50 bits

-emphasize random assignment



Picture set

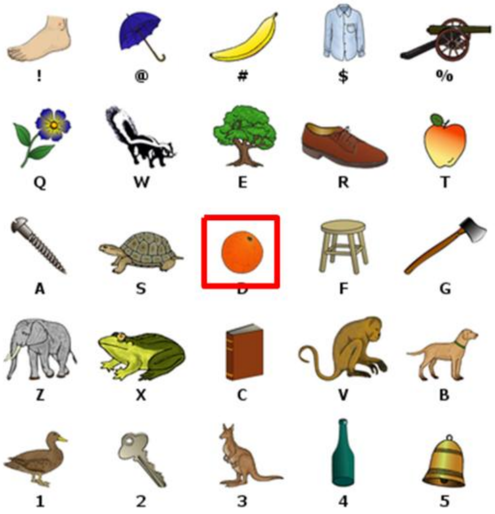
- Snodgrass and Vanderwart set (1980)
  - 260 pictures
- Selected pictures with low polysemy and low similarity to other pictures

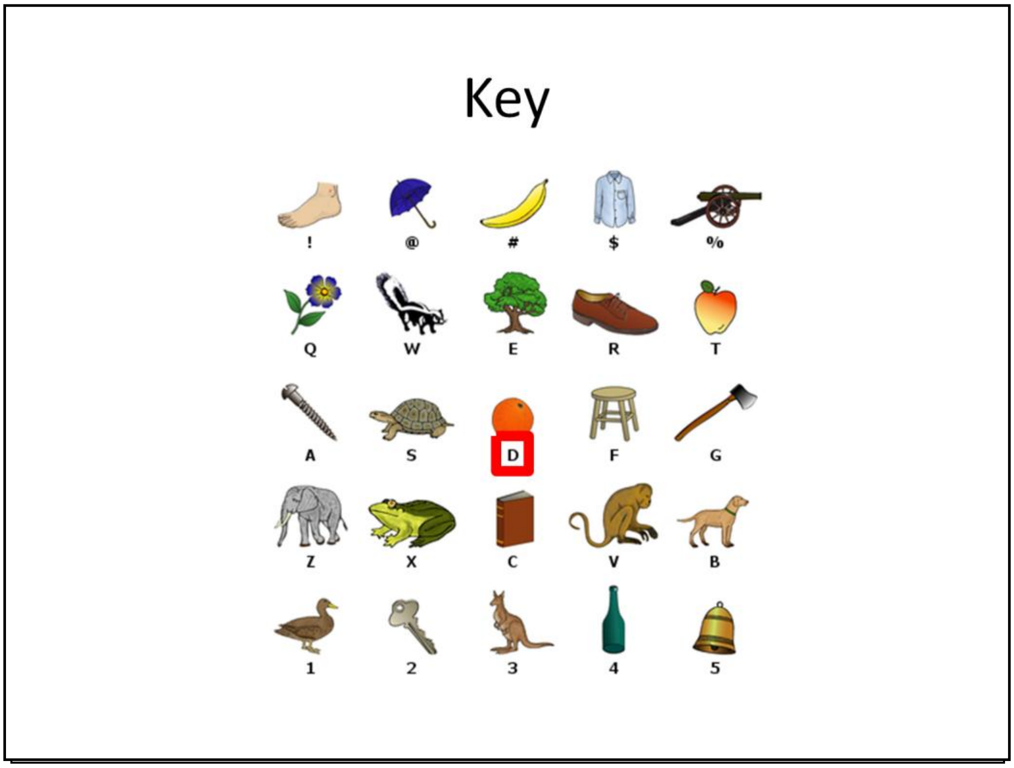


Multiple encodings

- Supports redintegration

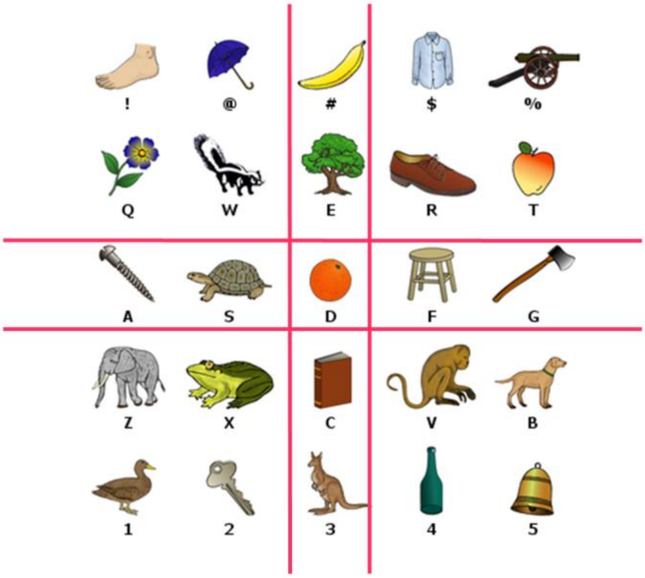
Picture

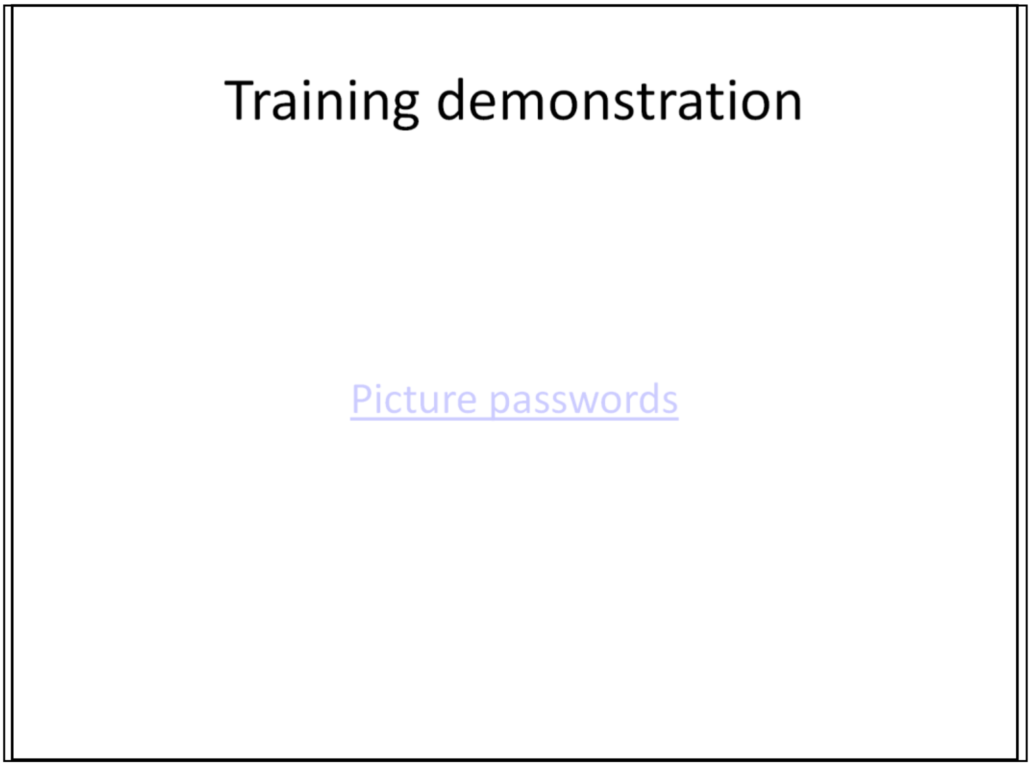




Key

Grid Location

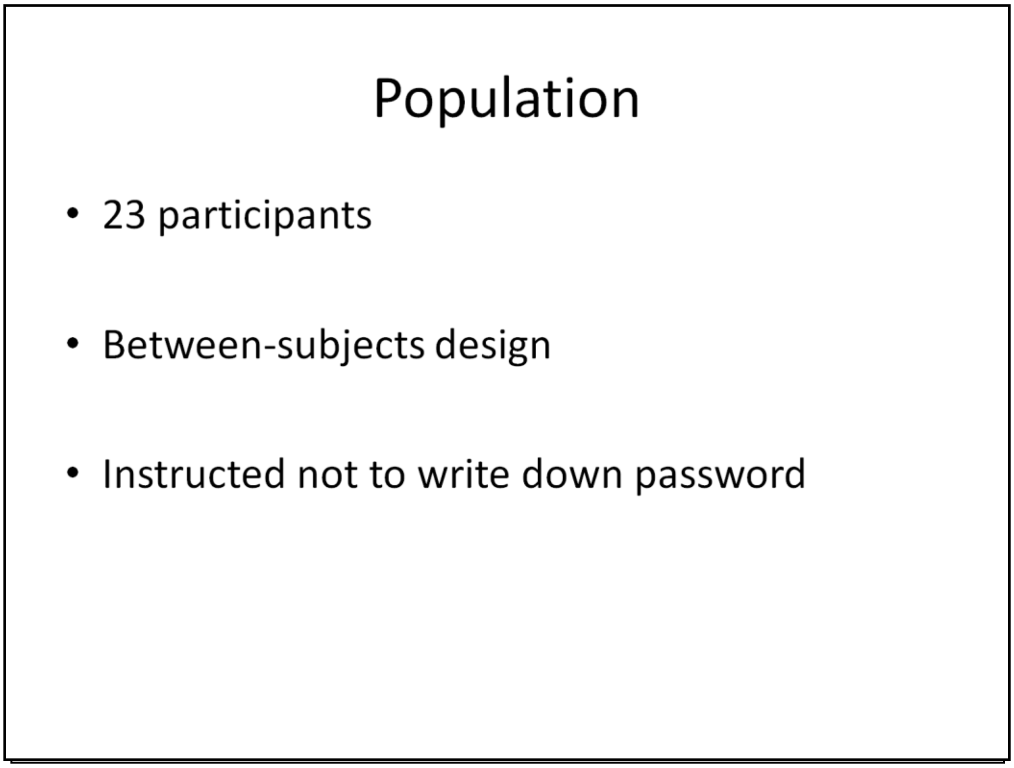




Training demonstration

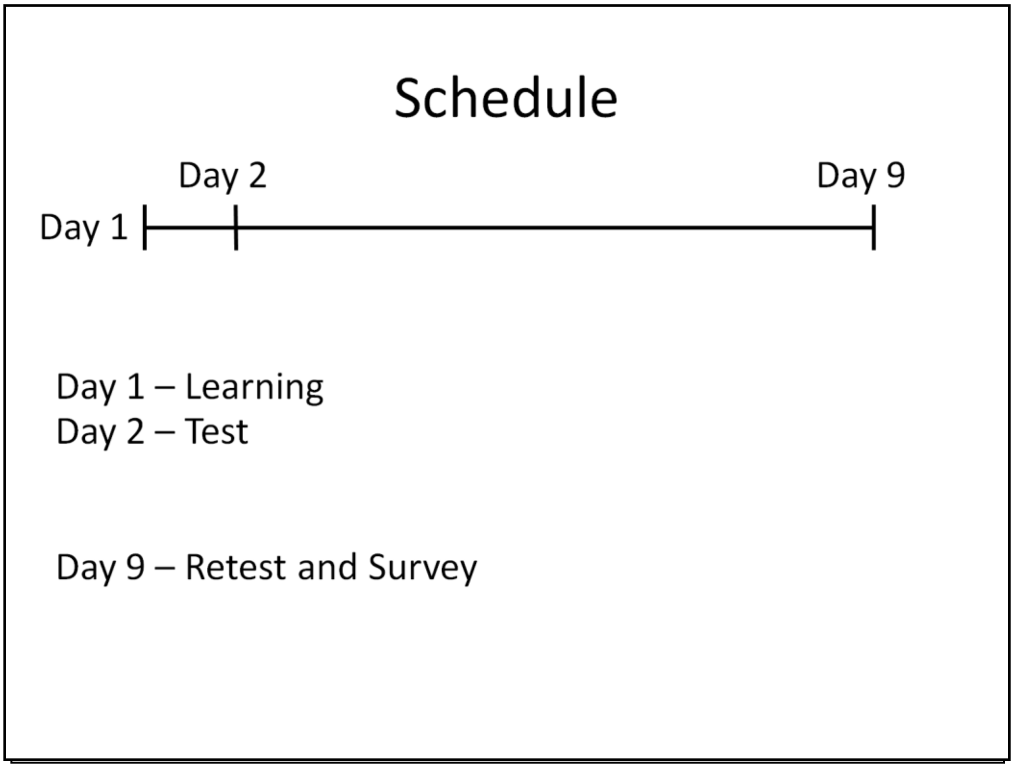
Picture passwords

Interactive training and learning



Population

- 23 participants
- Between-subjects design
- Instructed not to write down password



Schedule

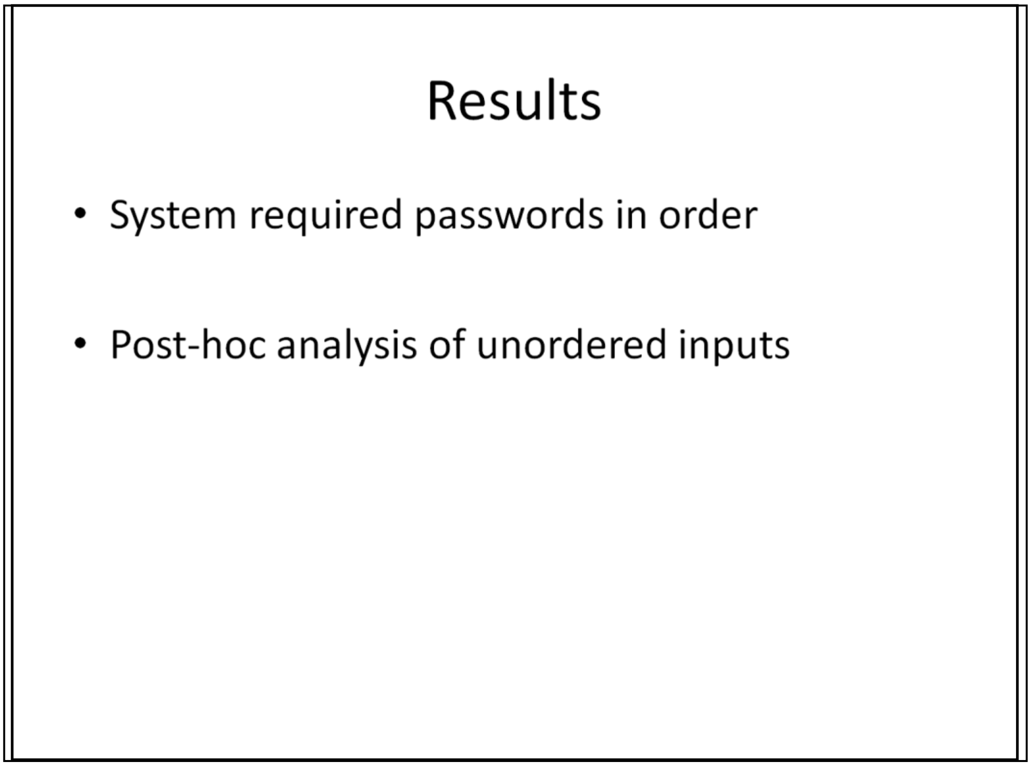
Day 2

Day 9

Day 1

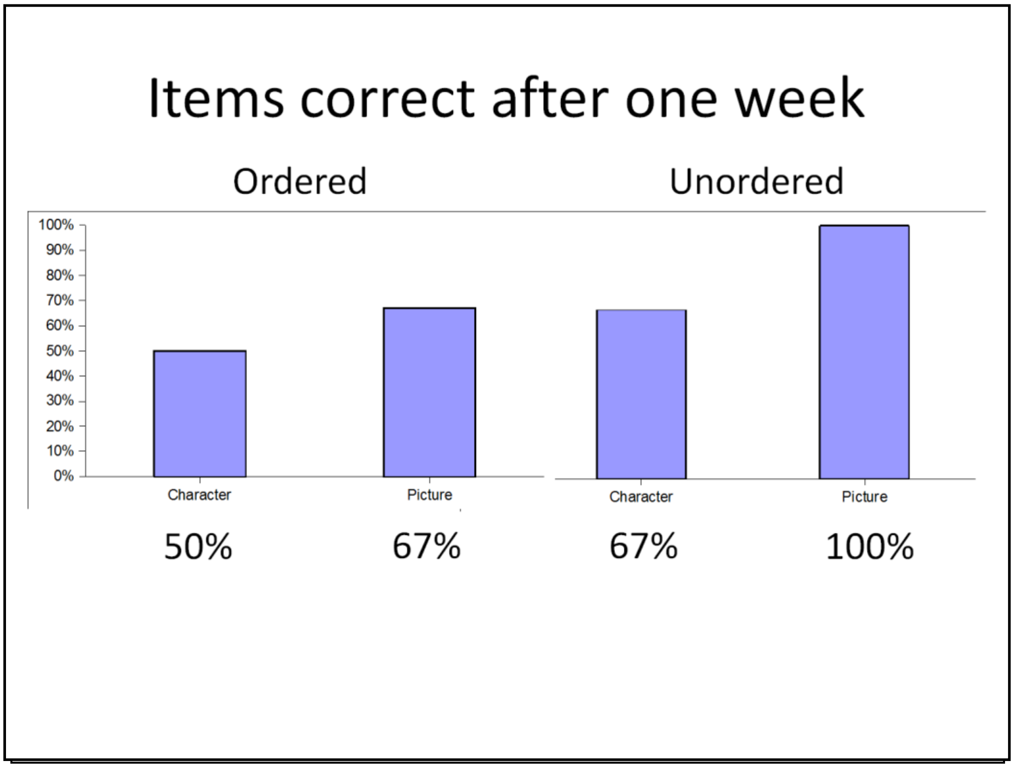
Day 1 – Learning
Day 2 – Test

Day 9 – Retest and Survey



Results

- System required passwords in order
- Post-hoc analysis of unordered inputs



Items correct after one week

Ordered

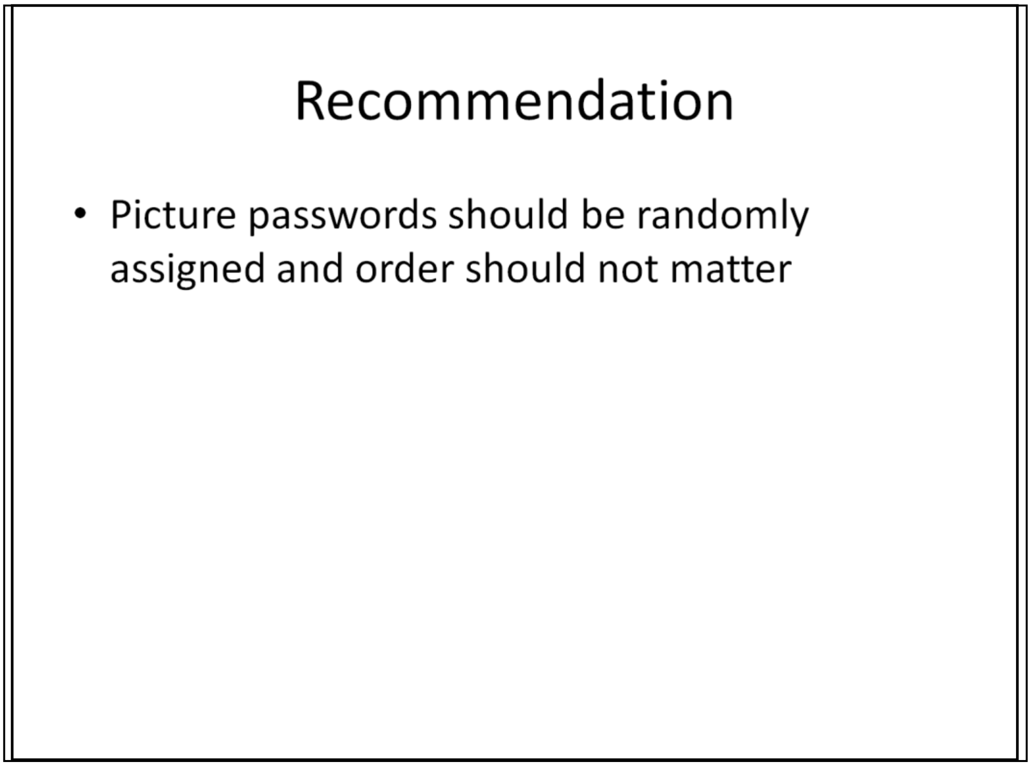
Unordered

50%

67%

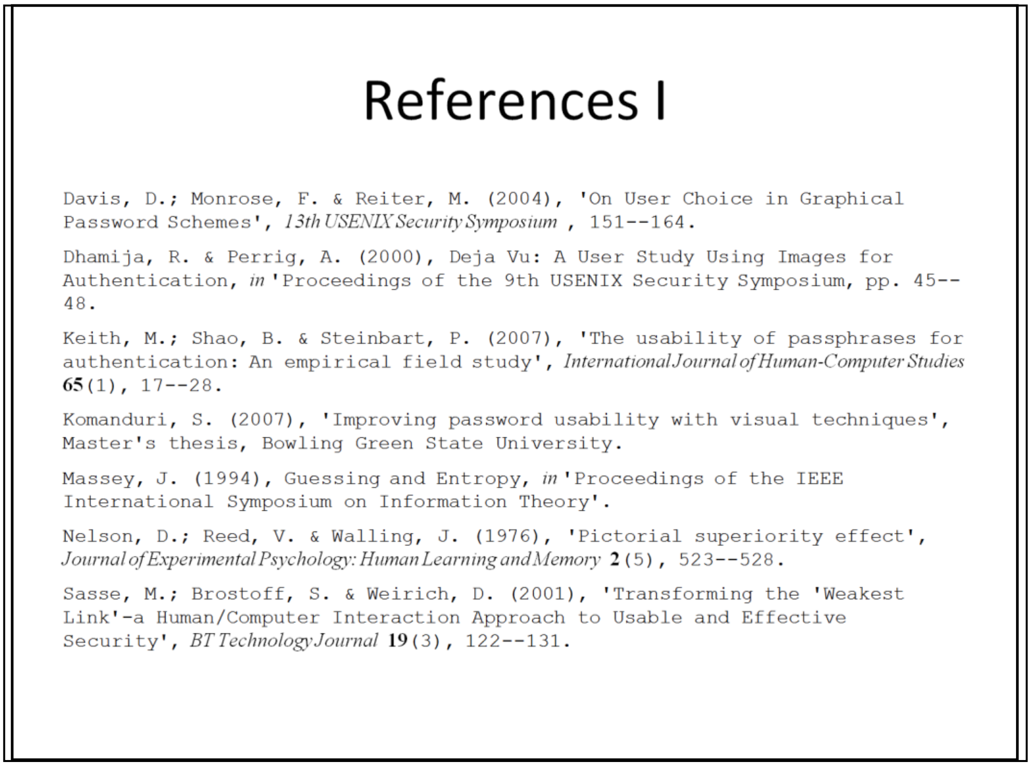
67%

100%



Recommendation

- Picture passwords should be randomly assigned and order should not matter



References I

Davis, D.; Monrose, F. & Reiter, M. (2004), 'On User Choice in Graphical Password Schemes', *13th USENIX Security Symposium* , 151--164.

Dhamija, R. & Perrig, A. (2000), Deja Vu: A User Study Using Images for Authentication, *in* 'Proceedings of the 9th USENIX Security Symposium, pp. 45--48.

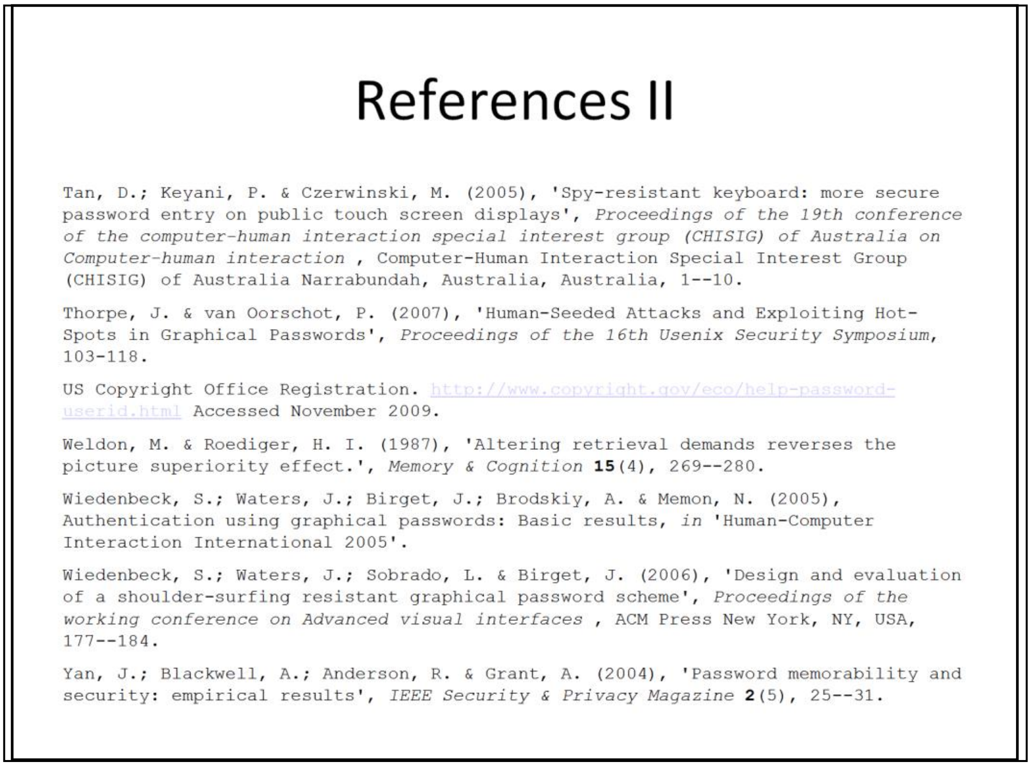
Keith, M.; Shao, B. & Steinbart, P. (2007), 'The usability of passphrases for authentication: An empirical field study', *International Journal of Human-Computer Studies* **65**(1), 17--28.

Komanduri, S. (2007), 'Improving password usability with visual techniques', Master's thesis, Bowling Green State University.

Massey, J. (1994), Guessing and Entropy, *in* 'Proceedings of the IEEE International Symposium on Information Theory'.

Nelson, D.; Reed, V. & Walling, J. (1976), 'Pictorial superiority effect', *Journal of Experimental Psychology: Human Learning and Memory* **2**(5), 523--528.

Sasse, M.; Brostoff, S. & Weirich, D. (2001), 'Transforming the 'Weakest Link'-a Human/Computer Interaction Approach to Usable and Effective Security', *BT Technology Journal* **19**(3), 122--131.



References II

Tan, D.; Keyani, P. & Czerwinski, M. (2005), 'Spy-resistant keyboard: more secure password entry on public touch screen displays', *Proceedings of the 19th conference of the computer-human interaction special interest group (CHISIG) of Australia on Computer-human interaction* , Computer-Human Interaction Special Interest Group (CHISIG) of Australia Narrabundah, Australia, Australia, 1--10.

Thorpe, J. & van Oorschot, P. (2007), 'Human-Seeded Attacks and Exploiting Hot-Spots in Graphical Passwords', *Proceedings of the 16th Usenix Security Symposium*, 103-118.

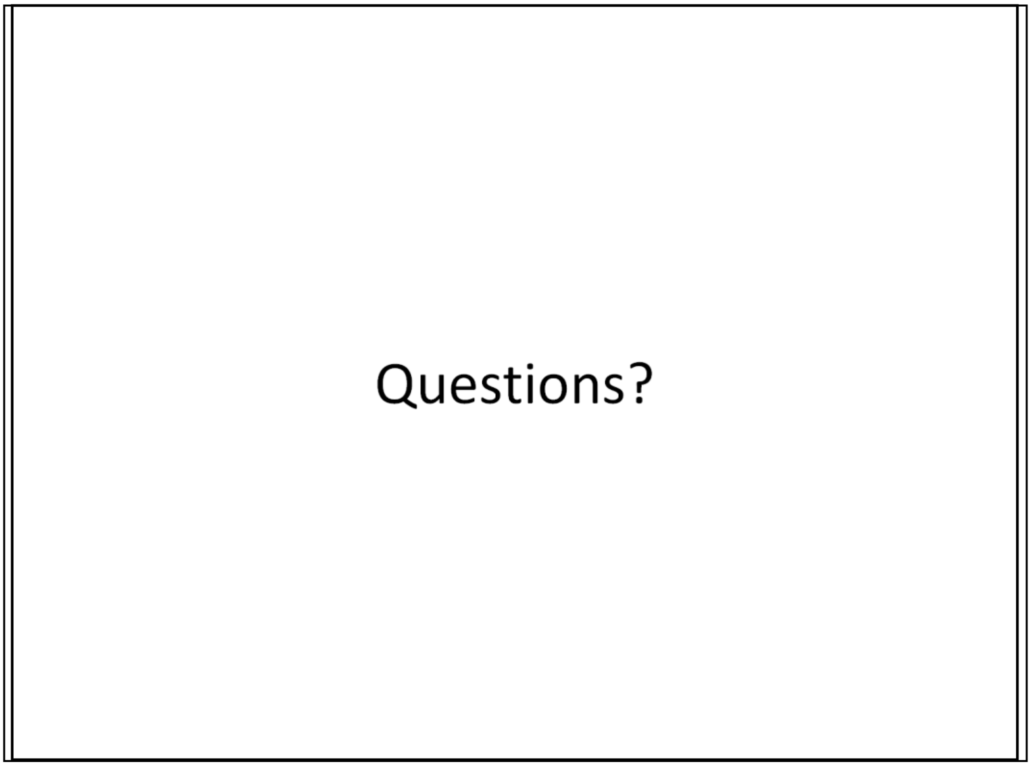
US Copyright Office Registration. http://www.copyright.gov/eco/help-password-userid.html Accessed November 2009.

Weldon, M. & Roediger, H. I. (1987), 'Altering retrieval demands reverses the picture superiority effect.', *Memory & Cognition* **15**(4), 269--280.

Wiedenbeck, S.; Waters, J.; Birget, J.; Brodskiy, A. & Memon, N. (2005), Authentication using graphical passwords: Basic results, *in* 'Human-Computer Interaction International 2005'.

Wiedenbeck, S.; Waters, J.; Sobrado, L. & Birget, J. (2006), 'Design and evaluation of a shoulder-surfing resistant graphical password scheme', *Proceedings of the working conference on Advanced visual interfaces* , ACM Press New York, NY, USA, 177--184.

Yan, J.; Blackwell, A.; Anderson, R. & Grant, A. (2004), 'Password memorability and security: empirical results', *IEEE Security & Privacy Magazine* **2**(5), 25--31.



Questions?