CSE P 590 / CSE M 590 (Spring 2010)

Computer Security and Privacy

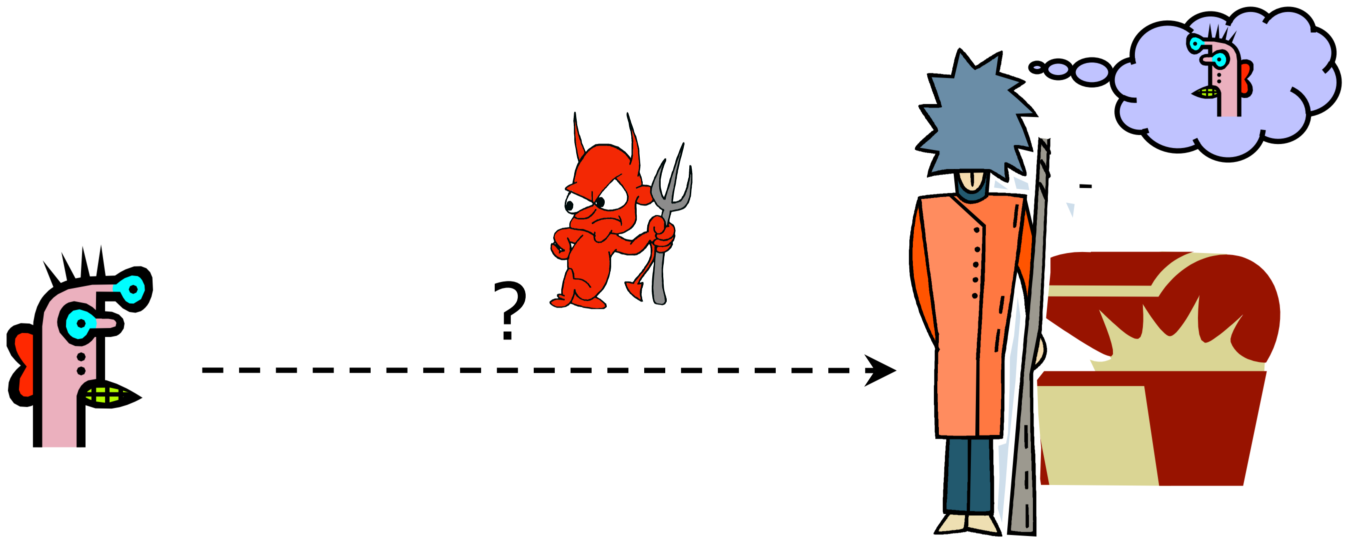
Tadayoshi Kohno

Thanks to Dan Boneh, Dieter Gollmann, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

Goals for Today

- Lab 1 (Announcement)
- User authentication
- Physical security

Basic Problem



How do you prove to someone that
you are who you claim to be?

Any system with access control must solve this problem

Many Ways to Prove Who You Are

- What you know
  - Passwords
  - Secret key
- Where you are
  - IP address
  - Physical location
- What you are
  - Biometrics
- What you have
  - Secure tokens
- All have advantages and disadvantages

Why Authenticate?

- To prevent an attacker from breaking into our account
  - Co-worker, family member, ...
- To prevent an attacker from breaking into any account on our system
  - Unix system
    - Break into single account, then exploit local vulnerability or mount a "stepping stones" attack
  - Calling cards
  - Building
- To prevent an attacker from breaking into any account on any system

Also Need

- Usability!
  - Remember password?
  - Have to bring physical object with us all the time?
- Denial of service
  - Stolen wallet
  - Try to authenticate as you until your account becomes locked
  - What about a military or other mission critical scenario
    - Lock all accounts - system unusable

Password-Based Authentication

- User has a secret password.
  System checks it to authenticate the user.
  - May be vulnerable to eavesdropping when password is communicated from user to system
- How is the password stored?
- How does the system check the password?
- How easy is it to remember the password?
- How easy is it to guess the password?
  - Easy-to-remember passwords tend to be easy to guess
  - Password file is difficult to keep secret

Common usage modes

Amazon = t0p53cr37

UWNetID = f0084r#1

Bank = a2z@m0$;

Image from http://www.interactivetools.com/staff/dave/damons_office/

Common usage modes

- Write down passwords
- Share passwords with others
- Use a single password across multiple sites
  - Amazon.com and Bank of America?
  - UW CSE machines and Facebook?
  - GMail and Facebook?
- Use easy to remember passwords
  - Favorite <something>?
  - Name + <number>?
- Other "authentication" questions
  - Mother's maiden name?

Some anecdotes [Dhamija and Perrig]

- Users taught how to make secure passwords, but chose not to do so

- Reasons:
  - Awkward or difficult
  - No accountability
  - Did not feel that it was important

University of Sydney Study [Greening '96]

- 336 CS students emailed message asking them to supply their password
  - Pretext: in order to "validate" the password database after a suspected break-in
- 138 students returned their password
- 30 returned invalid password
- 200 changed their password
- (Not disjoint)

- Still, 138 is a lot!

Awkward

- How many times do you have to enter your password before it actually works?
  - Sometimes quite a few for me! (Unless I type extra slowly.)
- Interrupts normal activity
  - Do you lock your computer when you leave for 5 minutes?
  - Do you have to enter a password when your computer first boots? (Sometimes it's an option.)
- And <u>memorability</u> is an issue!

- Untrusting of friends
  - Locking computer every time you get up

Memorability [Anderson]

- Hard to remember many PINs and passwords
- One bank had this idea
  - If pin is 2256, write your favorite 4-letter word in this grid
  - Then put random letters everywhere else

1	2	3	4	5	6	7	8	9	0
	b								
	l								
				u					
					e				

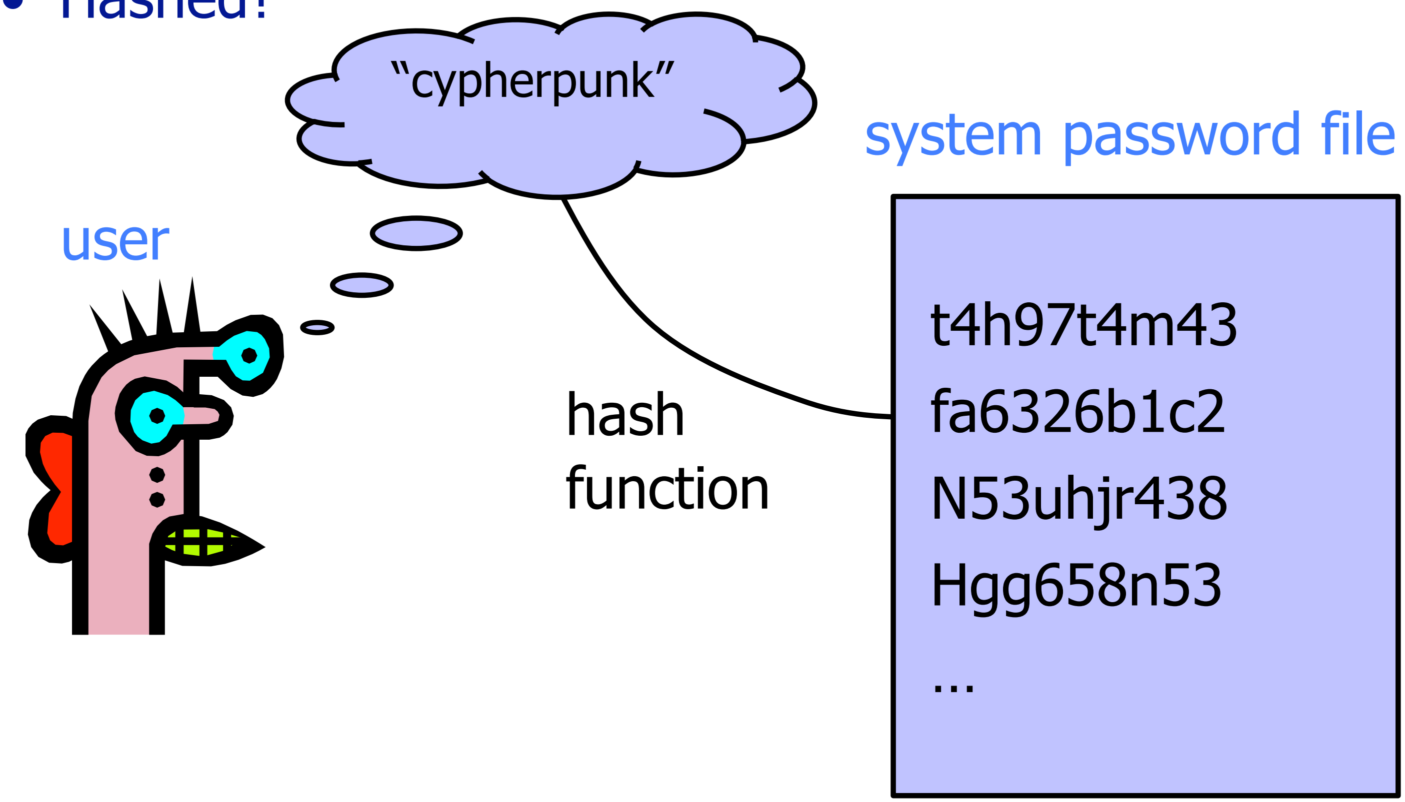
Memorability [Anderson]

- Problem!
- Normally 10000 choices for the PIN --- hard to guess on the first try
- Now, only a few dozen possible English words --- easy to guess on first try!

1	2	3	4	5	6	7	8	9	0
	b								
	l								
				u					
					e				

UNIX-Style Passwords

- How should we store passwords on a server?
  - In cleartext?
  - Encrypted?
  - Hashed?



Password Hashing

- Instead of user password, store H(password)
- When user enters password, compute its hash and compare with entry in password file
  - System does not store actual passwords!
  - System itself can't easily go from hash to password
    - Which would be possible if the passwords were encrypted
- Hash function H must have some properties
  - One-way: given H(password), hard to find password
    - No known algorithm better than trial and error

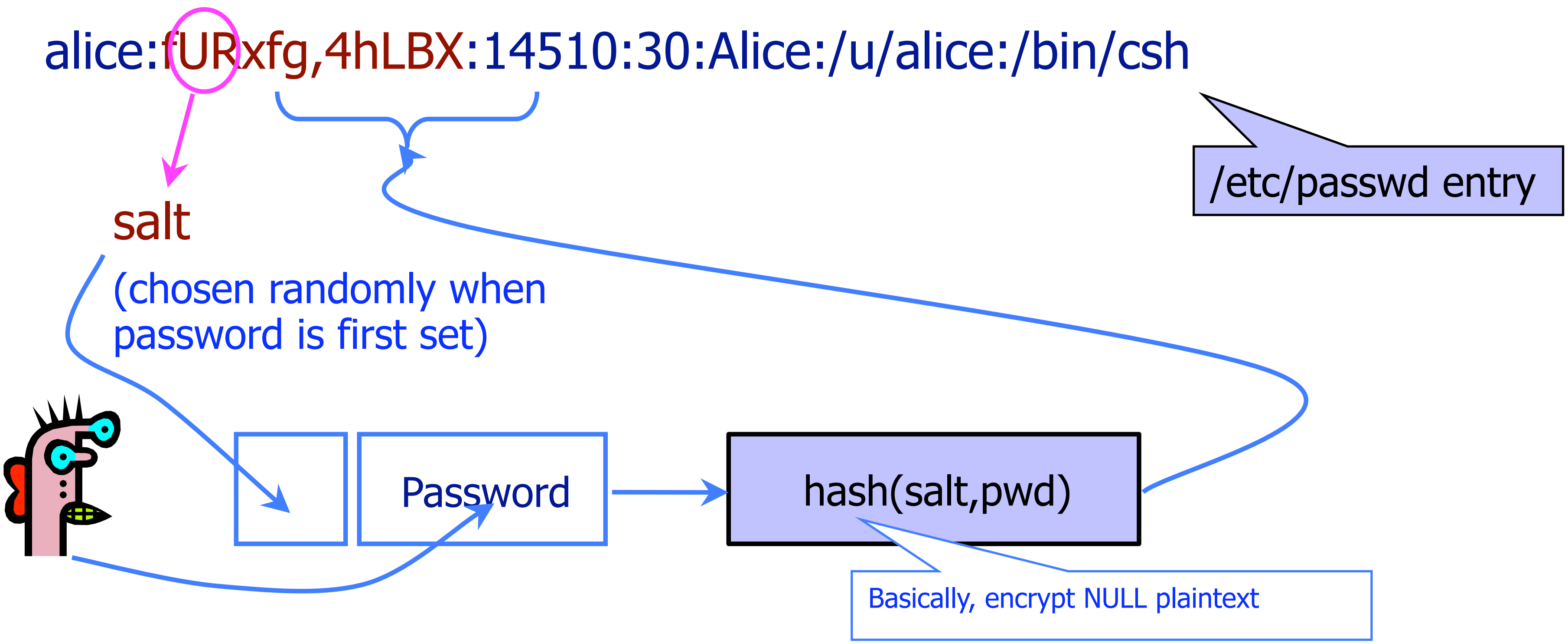
(Early) UNIX Password System

- Uses DES encryption as if it were a hash function
  - Encrypt NULL string using password as the key
    - Truncates passwords to 8 characters!
  - Artificial slowdown: run DES 25 times
    - Why 25 times? Slowdowns like these are important in practice!
  - ("Don't use DES like this at home.")
  - Can instruct modern UNIXes to use MD5/SHA1 hash function
- Problem: passwords are not truly random
  - With 52 upper- and lower-case letters, 10 digits and 32 punctuation symbols, there are $94^8 \approx$ 6 quadrillion possible 8-character passwords (around 2^{52})
  - Humans like to use dictionary words, human and pet names $\approx$ 1 million common passwords

Dictionary Attack

- Password file /etc/passwd is world-readable
  - Contains user IDs and group IDs which are used by many system programs
- Dictionary attack is possible because many passwords come from a small dictionary
  - Attacker can compute H(word) for every word in the dictionary and see if the result is in the password file
  - With 1,000,000-word dictionary and assuming 10 guesses per second, brute-force online attack takes 50,000 seconds (14 hours) on average
    - This is very conservative. Offline attack is much faster!
  - As described (H(word)), could just create dictionary of "word to H(word)" mapping once -- for all users!!

Salt



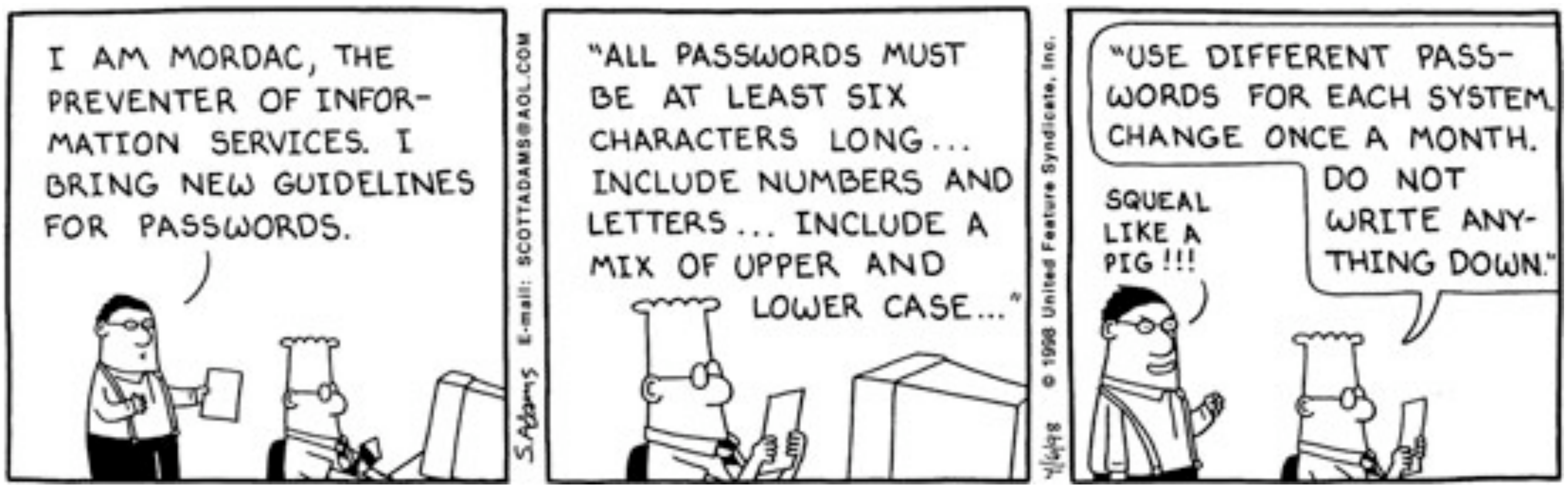
- Users with the same password have <u>different</u> entries in the password file
- Online dictionary attack is still possible! (Precomputed dictionaries possible too -- but significantly more expensive.)

Advantages of Salting

- Without salt, attacker can pre-compute hashes of all dictionary words once for <u>all</u> password entries
  - Same hash function on all UNIX machines
  - Identical passwords hash to identical values; one table of hash values can be used for all password files
- With salt, attacker must compute hashes of all dictionary words once for <u>each</u> password entry
  - With 12-bit random salt, same password can hash to 2^{12} different hash values
  - Attacker must try all dictionary words for each salt value in the password file
- Pepper: Secret salt (not stored in password file)

Other Password Issues

- Keystroke loggers
  - Hardware
  - Software / Spyware
- Shoulder surfing
  - It does happen!
- Online vs offline attacks
  - Online: slower, easier to respond
- Multi-site authentication
  - Share passwords?


I AM MORDAC, THE PREVENTER OF INFORMATION SERVICES. I BRING NEW GUIDELINES FOR PASSWORDS.
S. Adams
E-mail: SCOTTADAMS@AOL.COM
"ALL PASSWORDS MUST BE AT LEAST SIX CHARACTERS LONG... INCLUDE NUMBERS AND LETTERS... INCLUDE A MIX OF UPPER AND LOWER CASE..."
© 1998 United Feature Syndicate, Inc.
"USE DIFFERENT PASSWORDS FOR EACH SYSTEM. CHANGE ONCE A MONTH. DO NOT WRITE ANYTHING DOWN."
SQUEAL LIKE A PIG!!!

“Improving” Passwords

- Add biometrics
  - For example, keystroke dynamics or voiceprint
  - Revocation is often a problem with biometrics
- Graphical passwords
  - Goal: increase the size of memorable password space
- Password managers

Graphical Passwords

- Images are easy for humans to process and remember
  - Especially if you invent a memorable story to go along with the images
- Dictionary attacks on graphical passwords are difficult
  - Images are believed to be very "random" (is this true?)
- Still not a perfect solution
  - Need infrastructure for displaying and storing images
  - Shoulder surfing

Graphical Password Systems

- *Cognometric schemes*
  - present a set of images,
  - authentication requires selection of correct images
- *Locimetric Schemes*
  - presents a single image, with authentication requiring clicking on regions of the image
- *Drawmetric Schemes*
  - require drawing figures or doodles to authenticate.

Slides from Kate Everitt

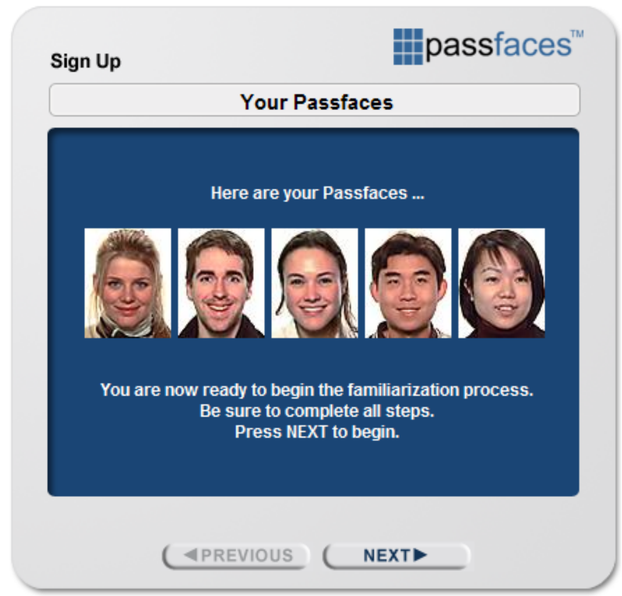
Assumption: Easy to recall faces

Slides from Kate Everitt

How Passfaces Works

Library of Faces

User Interface



Users Are Assigned a Set of 5* Passfaces

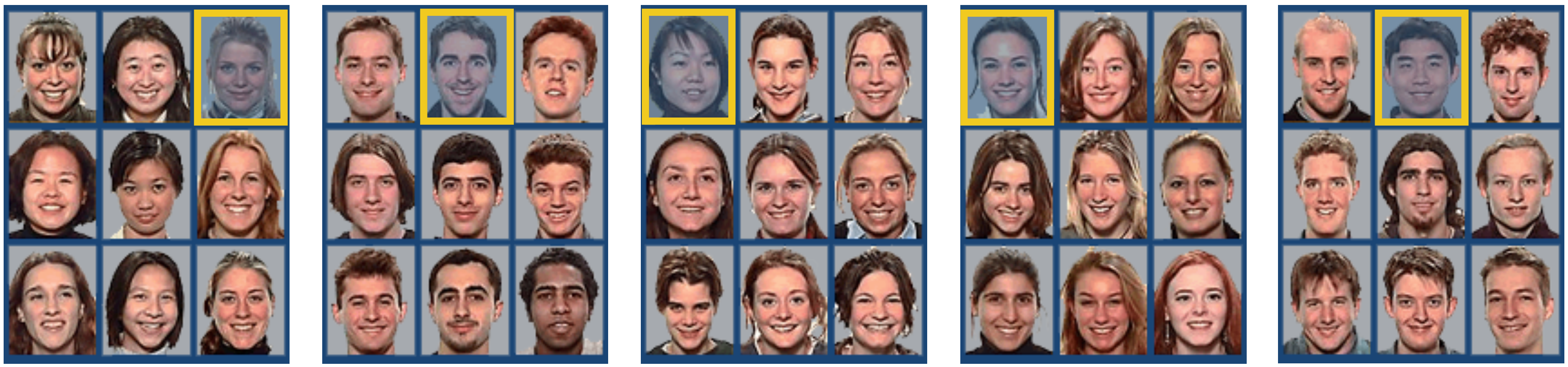
* Typical implementation – 3 to 7 possible as standard

passfaces™
STRONG AUTHENTICATION

The Only Fully Scalable Means to Replace or Reinforce Passwords

How Passfaces Works

- 5 Passfaces are Associated with 40 associated decoys
- Passfaces are presented in five 3 by 3 matrices each having 1 Passface and 8 decoys




passfaces™
STRONG AUTHENTICATION
The Only Fully Scalable Means to Replace or Reinforce Passwords


Demo
passfaces™
You Have Successfully Logged on Using Passfaces
CONGRATULATIONS! You have successfully logged on using Passfaces, the ONLY Fully Scalable means to Replace or Reinforce Passwords. For more information on Passfaces, call 410.224.4848 or e-mail us at sales@passfaces.com
OK

Empirical Results

- Experimental study of 154 computer science students at Johns Hopkins and Carnegie Mellon
- Conclusions:
  - "… faces chosen by users are highly affected by the race of the user… the gender and attractiveness of the faces bias password choice… In the case of male users, we found this bias so severe that we do not believe it possible to make this scheme secure against an online attack…"
- 2 guesses enough for 10% of male users
- 8 guesses enough for 25% of male users

User Quotes

- "I chose the images of the ladies which appealed the most"
- "I simply picked the best lookin girl on each page"
- "In order to remember all the pictures for my login (<u>after forgetting my 'password' 4 times in a row</u>) I needed to pick pictures I could EASILY remember... So I chose beautiful women. The other option I would have chosen was handsome men, but the women are much more pleasing to look at"

More User Quotes

- "I picked her because she was female and Asian and being female and Asian, I thought I could remember that"
- "I started by deciding to choose faces of people in my own race…"
- "… Plus he is African-American like me"

- Recommendation: system picks passfaces
- But is that still memorable? What issues could arise?

What about multiple passwords?

- 109 participants in a 5 week study
- Email-based prompts to access the study website and authenticate
- Study emails were sent on Tuesday, Wednesday, Thursday, and Friday
- Participants were allowed a maximum of three login attempts

Study Conditions

1

A			
	A		
		A	
	A		
			A

2

B	B	B	
	B	B	B
B		B	B
B	B		B
B		B	B

3

B	B	B	A
A	B	B	B
B	A	B	B
B	B	A	B
B	A	B	B

4

A	B	C	D
C	B	A	D
B	D	C	A
D	A	B	C
A	B	C	D

5

A	A	A	A
B	B	B	B
C	C	C	C
D	D	D	D
A	B	C	D

Frequency, interference, and training do play a role in memorability

Slides from Kate Everitt

Variants...

- Plus click-based graphical passwords, drawing-based passwords, ...

Uses of graphical passwords?

- For what applications might graphical passwords be particularly useful?