Text Passwords

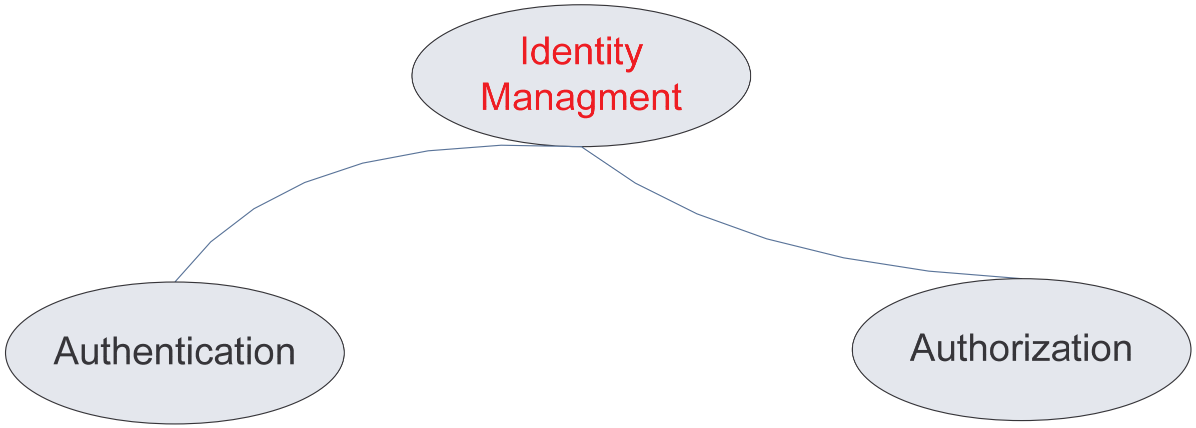
Sasha Romanosky

March 30, 2006

Carnegie Mellon
Information Security Policy & Management
Heinz School

Mind Map

- Mind Maps show the associations between components of a system
- Our system is Text passwords, but let's start with Identity Management…

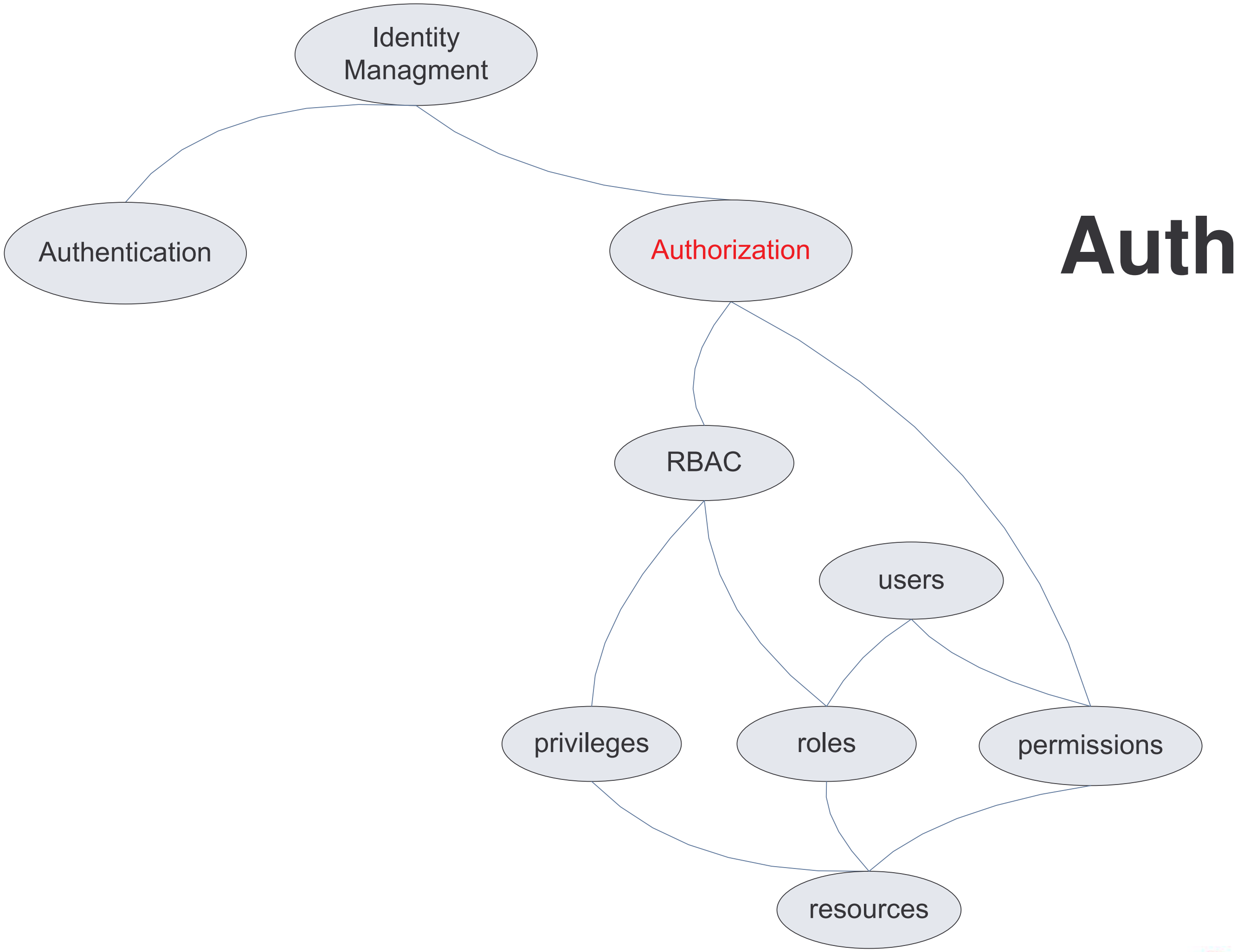


Identity Managment

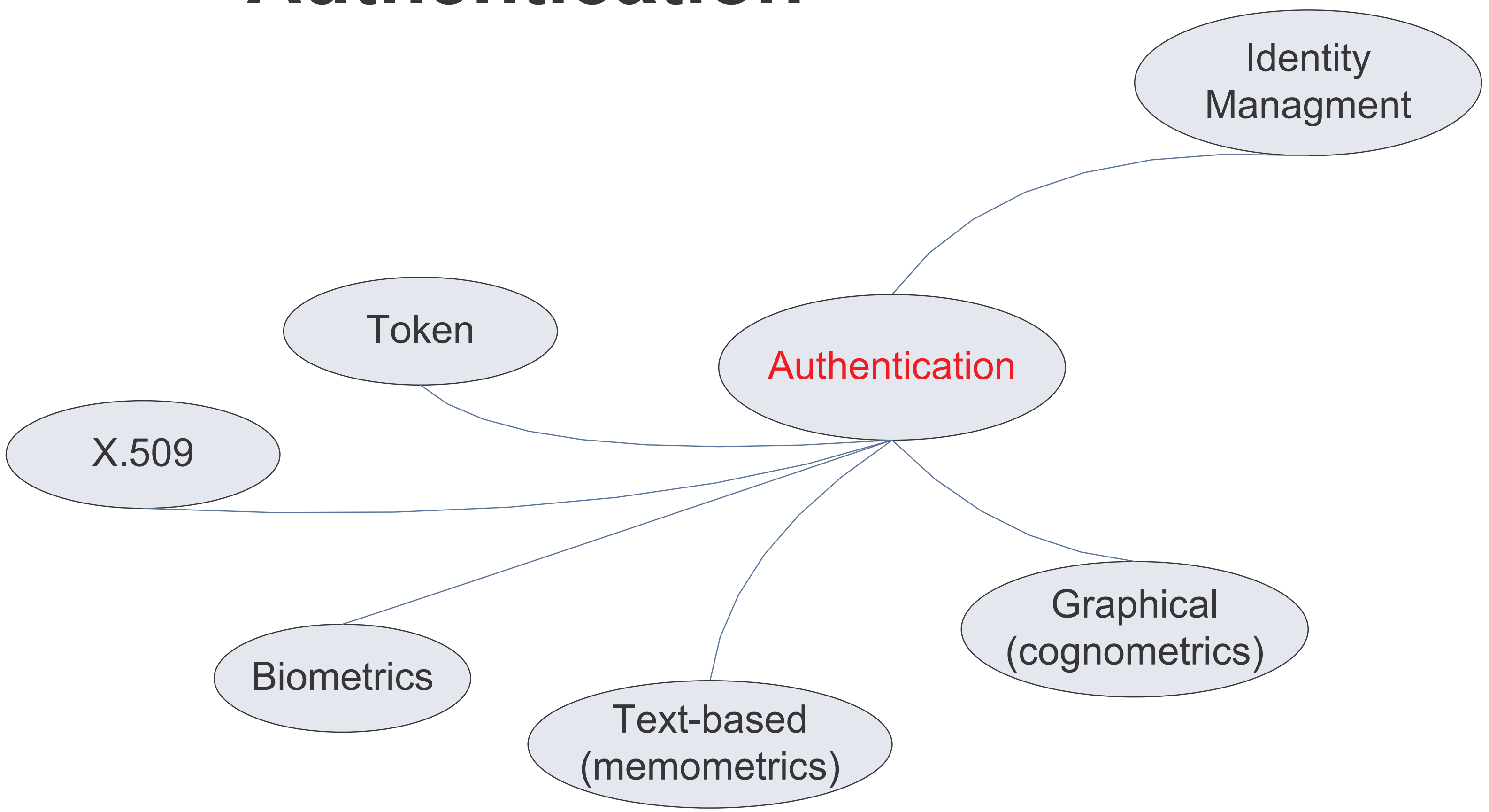
Authentication

Authorization

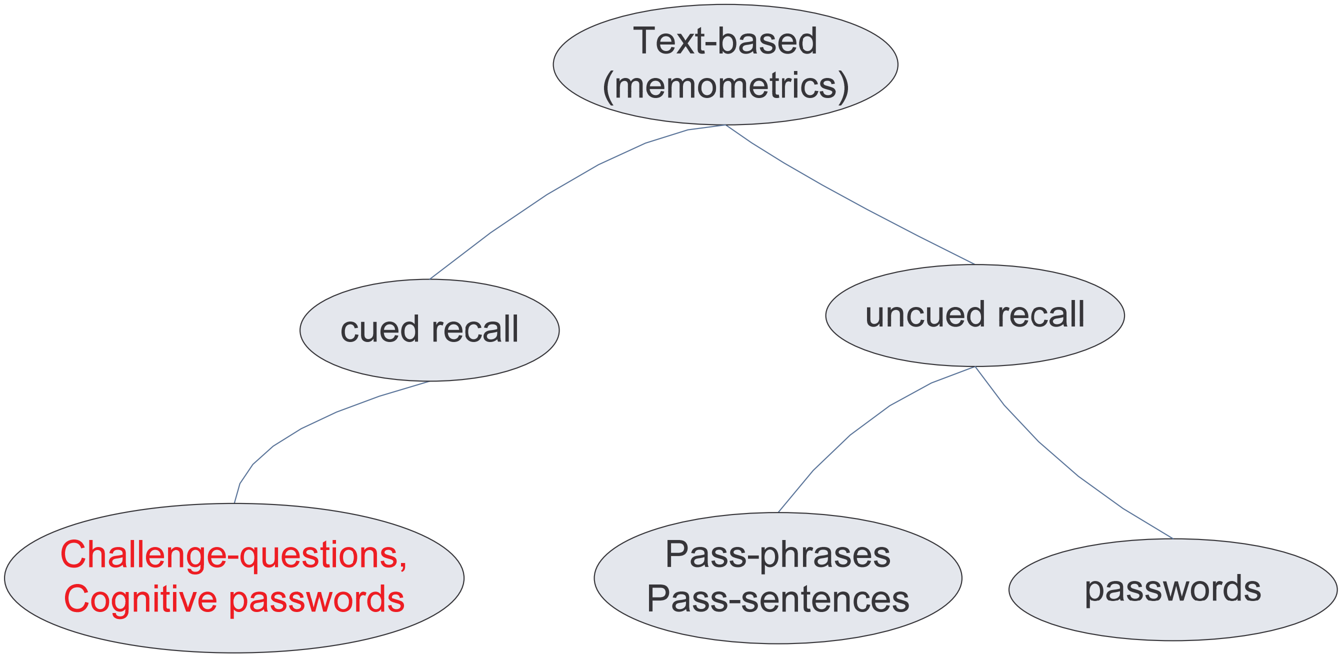
Authorization



Authentication



Challenge Questions

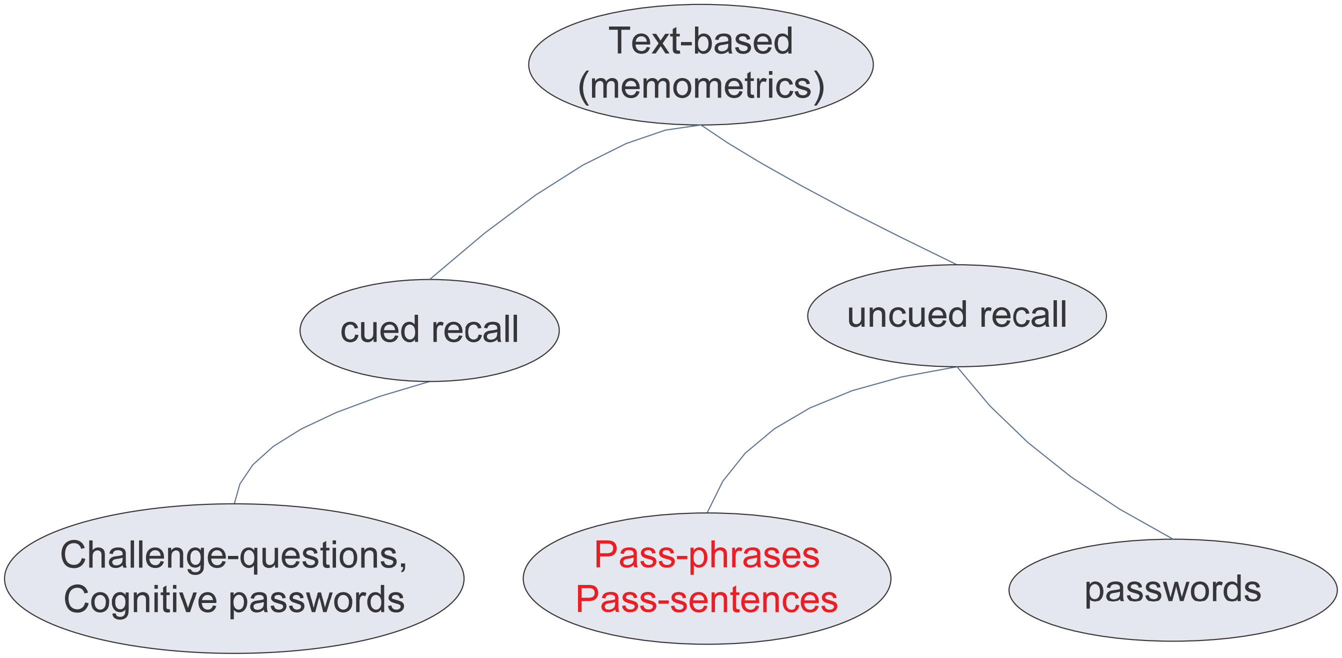


Challenge Questions

- Also known as cognitive passwords [Zivran and Haga 1990]
- Personal questions that either the system poses or the user is able to self-select
- Advantages:
  - Increased memorability
  - Difficulty for others to guess
- Fact-based
  - Constant, factual information
  - E.g. “what is your shoe size?”
- Opinion-based
  - can change over time due to beliefs
  - E.g. “how do you like your eggs?”

- Uses: ING Direct, often used for password recovery

Pass-Sentences, Pass-phrases

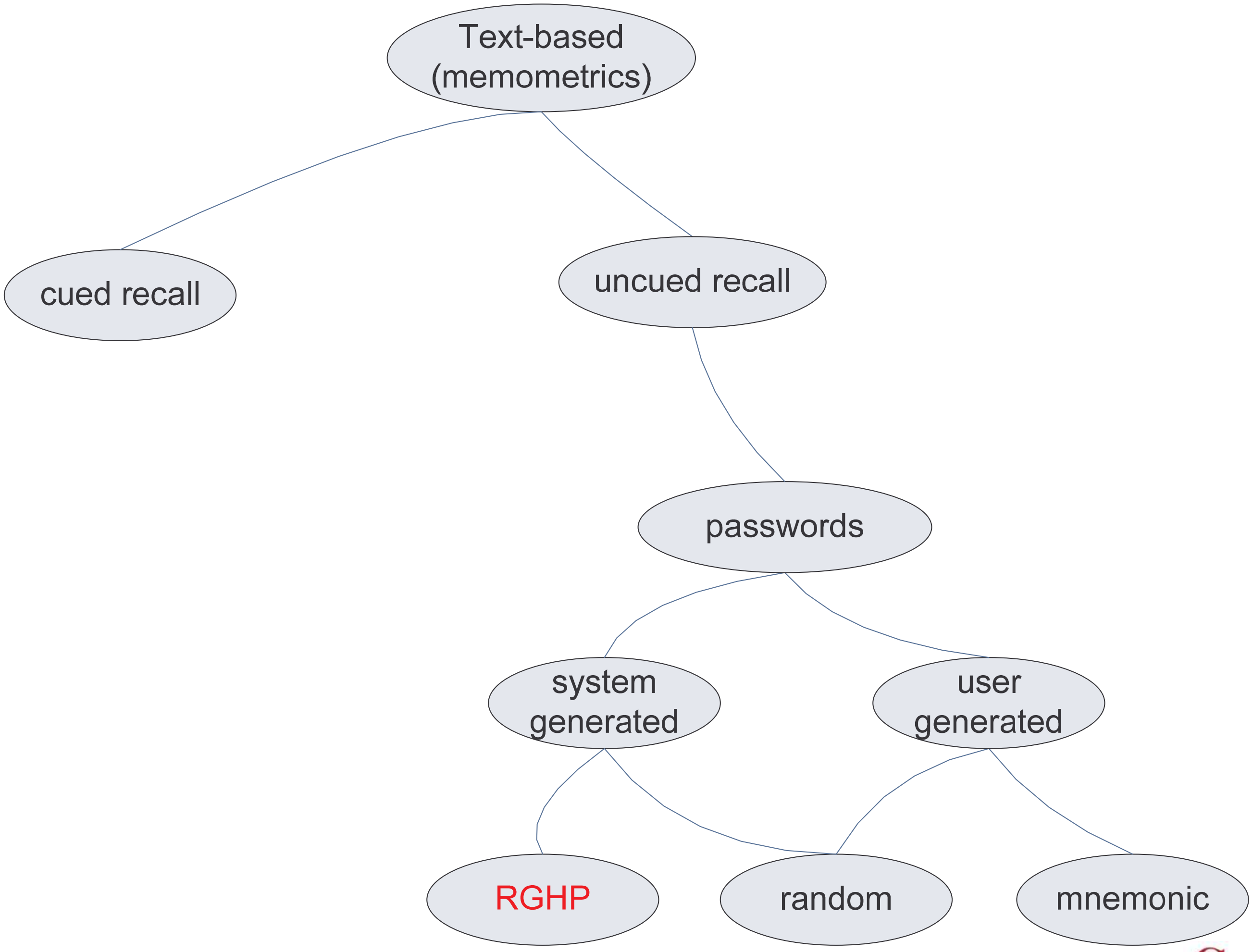


Pass-Sentences, Pass-phrases

- Essentially really long passwords. Full or partial phrase.
- Advantages:
  - Increased memorability
  - Difficult for others to guess
  - Difficult for software to crack
  - Still vulnerable to theft (interception and file) and social engineering like any other text password

- Some propose a semantic implementation called Pass-sentence [Spector and Ginzberg, 1994]
- User is granted (incremental) access based on a compliance score
- e.g. compare actual phrase: "Jack bought a pizza at Martin's for half a dollar." Based on a threshold, the system might accept the following, "Jack got some food at Martin's place."

Randomly generated, human pronounceable



Randomly generated, human pronounceable

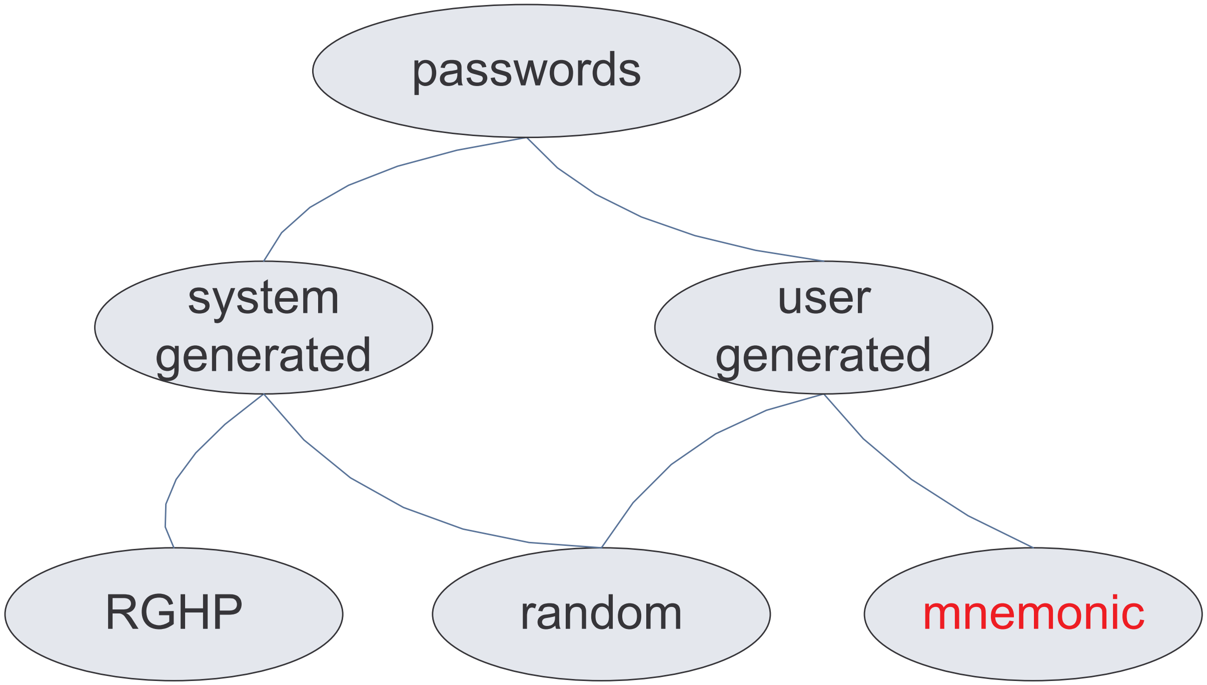
- System-generated passwords constructed from concatenating syllables from a given language [Morrie Gasser 1977, FIPS Pub 181, 1993]

- Advantages:
  - Presumably more memorable (due to pronounceability)
  - Strength (arbitrary length, non dictionary word)

- Disadvantages
  - Single character class (lower case)
  - Limited password space (entropy)

- “Smallest bucket” attack ” [Ganesan and Davies 1994]. Attacker need only focus on a small subset of the password space

Mnemonic Passwords

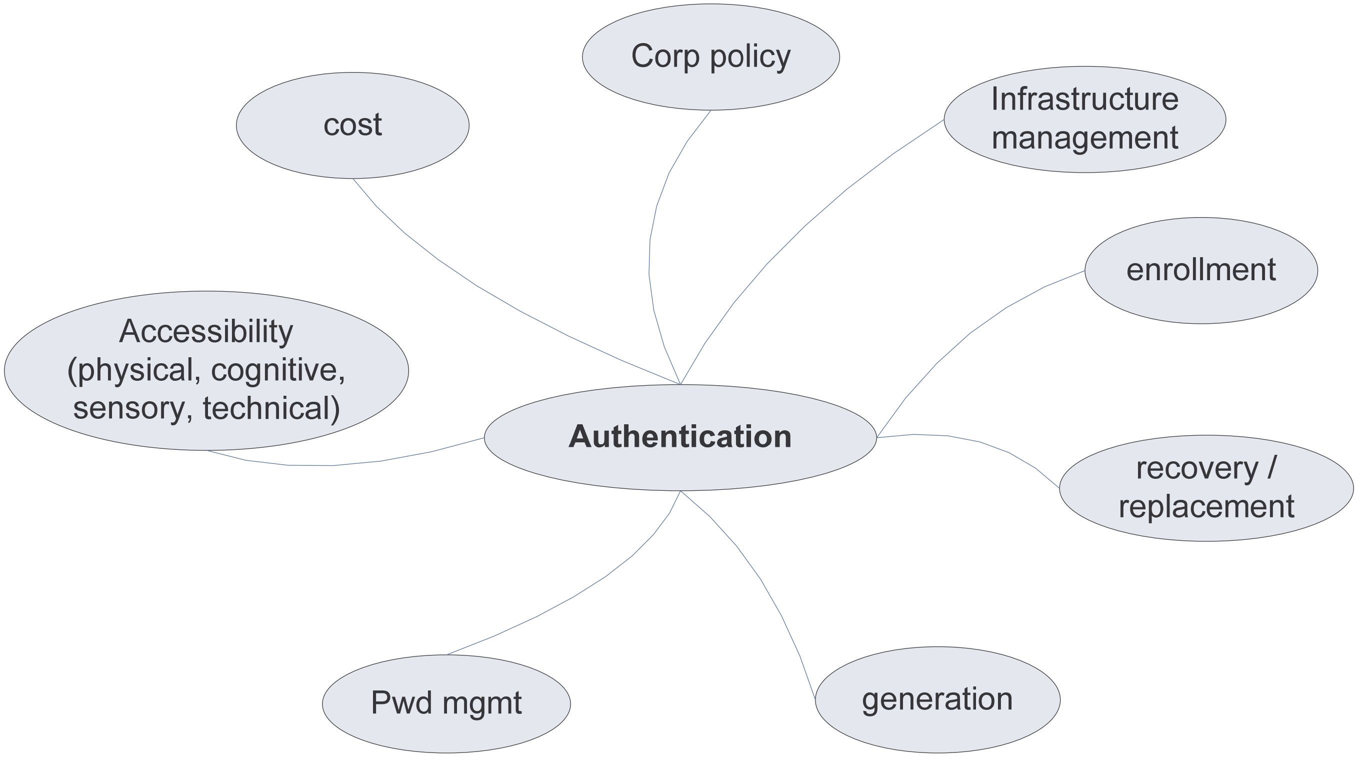


Mnemonic

- Passwords created from taking the first word of a phrase.
  - “I love to ski at Seven Springs!” -> “Ilts@7S!”
  - “Alas, poor Yorick! I knew him, Horatio” -> “A,pY!Ikh,H”
  - “Four score and seven years ago, our Fathers” -> “4s&7yaoF”

- Advantages:
  - Often multiple character classes
  - Easier to remember [Yan et al 2000]
  - Don’t appear in dictionaries – *or do they??*

Underlying Issues


Corp policy
cost
Infrastructure management
enrollment
Accessibility (physical, cognitive, sensory, technical)
Authentication
recovery / replacement
Pwd mgmt
generation

A Word on Memorability

- Sure it’s better if you can always remember your password. (Think of cost savings for help-desks), but is it a necessary condition for better password? *Perhaps not.*
  - Regardless of the randomness (difficulty), if it’s something we use every day, we’ll remember it [Renaud 2005]
  - It’s no longer practical to remember unique passwords, instead use software, file, notebook, etc.
  - Writing down a password is not necessarily a bad thing. People already store confidential documents at home and work.

2 Minute Break

The exploding whale

User Study:
Human selection of text passwords

- Survey asked users to generate either a mnemonic or random password
- Conducted for 2 weeks in February, 2006
- 298 respondents: 147 random, 151 mnemonic
- Question: do users generate stronger mnemonic or random passwords?
- In order to test this, we did the following:
  - Scored the strength of each password
  - Tried to crack each password
  - Compared the frequency distribution of characters for both types

Scoring Equation

- Password strength = $\log_{10}(\text{number of characters}^{\text{length}})$

- Where “number of characters” is the sum of characters in the search space. i.e. uppercase = lowercase = 26, numbers = 10, special = 33.
- “length” is the number of characters in the password
- Log function reduces score to an order of magnitude estimate

- “bravo” = $\log_{10}[(26)^5] = 7.1$
- “sUperFl8” = $\log_{10}[(26+26+10)^8] = \log_{10}[(62)^8] = 14.3$
- “ch@rAc!eR5” = $\log_{10}[(26+26+10+33)^{10}] = \log_{10}[(95)^{10}] = 19.8$

Password cracking

- Methods (for both random and mnemonic)
  - Dictionary attack (1.2M words for random, 400k words for mnemonic*)
  - Dictionary with word permutations (a->@, e->3, etc)
  - 62 hour brute force

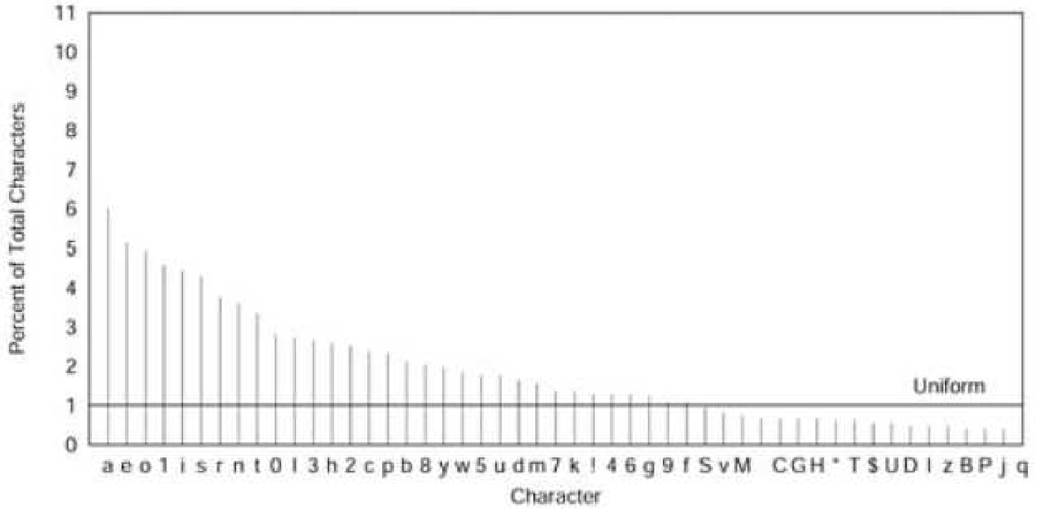
  - *Mnemonic dictionary built from scraping aggregation websites of song lyrics, nursery rhymes, advertising slogans, famous quotations, etc.

Results

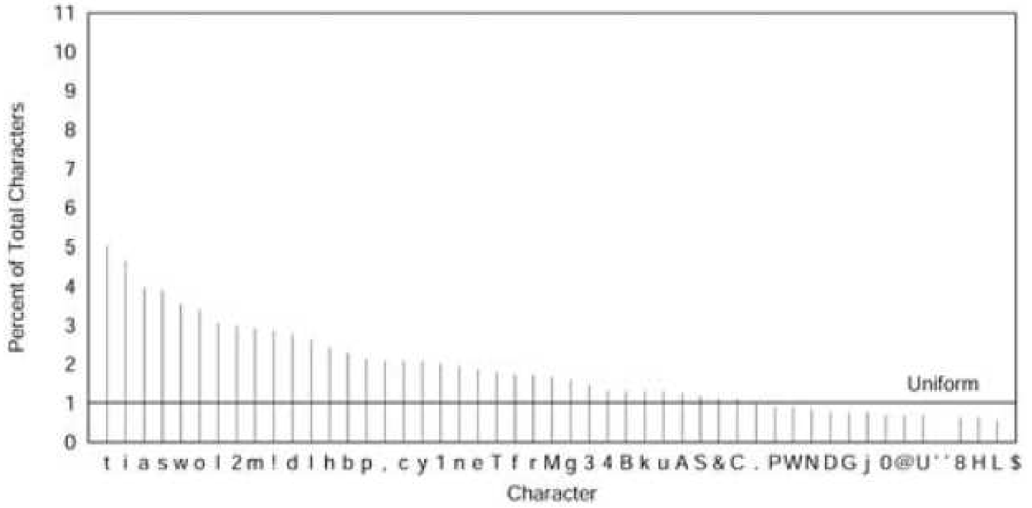
- Most mnemonic passwords are generated from phrases found on the internet – *this means that these passwords **are** susceptible to dictionary attack*
- Random, user-generated text passwords may not be as insecure as previously assumed – *they actually look pretty good!*

	Random	Mnemonic
User-generated score	15.7, σ = 7.3	17.2, σ = 8.3
Num. char classes	2.7, σ = 0.9	2.9, σ = 1.0
Length	~ 10	~ 10
Percent cracked	19%	8%

Character frequency of:



<- Random passwords



<- Mnemonic passwords

Character Frequency

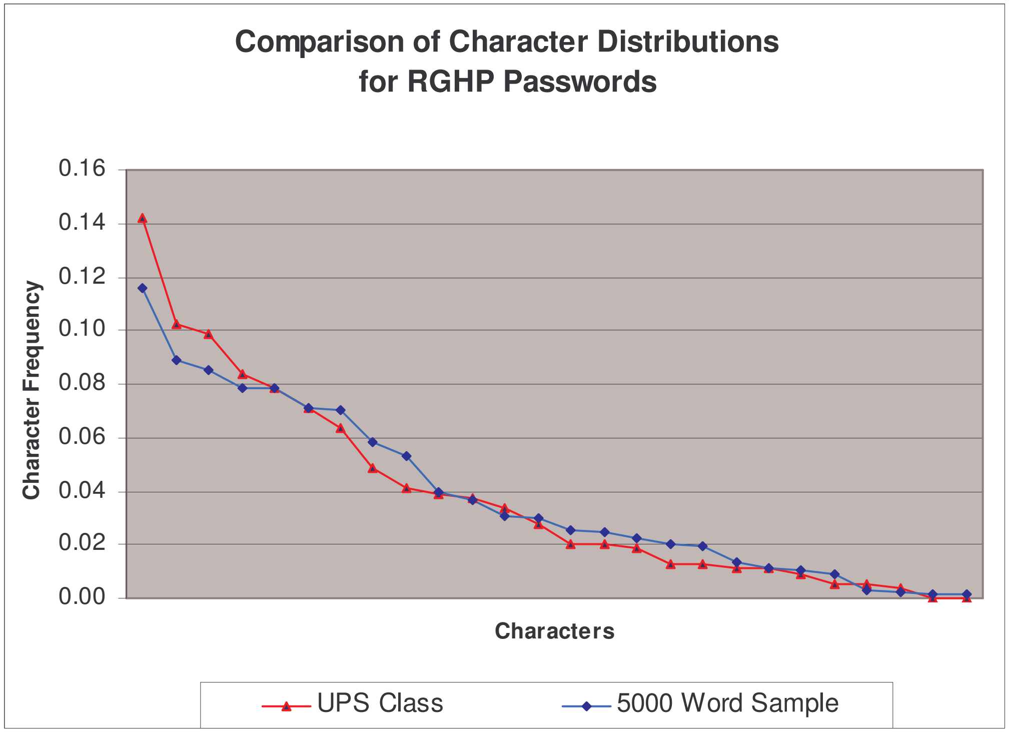
- Why is this interesting?

- Because if it was found that most passwords consisted of a handful of characters, then brute forcing attacks could be significantly improved

Class Exercise: Randomly Generated, Human Pronounceable Passwords

- Visit the web page mentioned in class and enter your selection.
- This page presents you with a list of 10 RGHP passwords. We saw how they were formed and discussed some of their weaknesses in generation.
- Now the question is: will user-selection of RGHP result in a smaller password space?
- i.e. If everyone chooses passwords with the same letters, this will reduce the necessary searchable space and increase the likelihood of compromise
- Let’s see the results!

Results of Class-selected RGHP Passwords



Graph compares the class' selections with a sample of 5000 RGHP passwords.

Result: class' frequency distribution was more skewed.

This means that you password selections were made up of fewer characters and therefore more susceptible to optimized brute force attack.

http://explodingdog.com/rocketguy.gif