

# User Authentication and Passwords

## CSS322: Security and Cryptography

Sirindhorn International Institute of Technology  
Thammasat University

Prepared by Steven Gordon on 29 December 2011  
CSS322Y11S2L11, Steve/Courses/2011/S2/CSS322/Lectures/passwords.tex, r2072

# Contents

## Authentication with Humans

### Passwords

# Humans and Computers

*Humans are also large, expensive to maintain, difficult to manage and they pollute the environment. It is astonishing that these devices continue to be manufactured and deployed. But they are sufficiently pervasive that we must design our protocols around their limitations.*

— Kaufman, Perlman, Speciner “Network Security: Private Communication in a Public World”, Prentice Hall 2002

# Authentication for Computers and Humans

## Computer to computer authentication

- ▶ Computers can remember high-quality cryptographic keys and perform cryptographic operations

## Human to computer/human authentication

- ▶ Humans cannot store large keys
- ▶ Humans cannot accurately or efficiently perform cryptographic operations

Need special methods for authenticating people

# Authenticating People

## What You Know

- ▶ Passwords
- ▶ Passphrase
- ▶ PIN

## What You Have

- ▶ Physical keys
- ▶ ATM card
- ▶ ID card

## What You Are

- ▶ Voice recognition
- ▶ Fingerprints
- ▶ Eye scanners

# Contents

Authentication with Humans

Passwords

# What is a Password?

- ▶ Combination of characters (available from keyboard)
- ▶ Should be secure and easy to use
- ▶ Passphrase: sequence of words or text
  - ▶ Spaces allowed
  - ▶ Usually longer than password

# Problems With Passwords

- ▶ Attacker may see the password when it is used
- ▶ Attacker may read the file where computer stores password
- ▶ Your password might be easy to guess at your computer
- ▶ Your password might be crackable off-line (brute force)
- ▶ Your password might be secure, but the system may become too inconvenient to use (e.g. very long password)



# How Are Passwords Used?

- ▶ User identified by unique value (e.g. username)
- ▶ User selects a password
- ▶ System stores username and password
- ▶ To access system:
  1. User submits username/password to system
  2. System compares submitted values with stored values
  3. If match, user is authenticated
- ▶ Issues:
  - ▶ What is a good password?
  - ▶ How to store the passwords?
  - ▶ How to submit the passwords?
  - ▶ How to respond (if no match)?

# Online Password Guessing

- ▶ Try to guess the password while system in use
  - ▶ Attacker has only a limited time
  - ▶ Guesses can be recorded/tracked
- ▶ Security depends on:
  - ▶ Number of incorrect guesses allowed
  - ▶ Consequence of too many incorrect guesses
- ▶ Approaches to make system more secure:
  - ▶ Lock system (e.g. account) if too many guesses
  - ▶ Limit the speed that guesses can be made
  - ▶ Try to find the attacker
  - ▶ Make passwords harder to guess

# Strength of Passwords

- ▶ **Entropy** used as indicator of password strength
  - ▶ Password with entropy of  $n$  bits is equivalent to  $n$ -bit key at withstanding brute force
  - ▶ How many bits needed to represent symbols from symbol set:
    - ▶ Digits, 0 . . . 9: 3.32
    - ▶ English letters, a . . . z: 4.70
    - ▶ Printable ASCII characters (94): 6.55
  - ▶ For 64-bit equivalent strength:
    - ▶ Digits: 20
    - ▶ English letters: 14
    - ▶ Printable ASCII characters: 10
- ▶ Human generated passwords are not random
  - ▶ Difficult to estimate entropy, NIST have approximations

# NIST Estimated Password Strength

Length Char.	User Chosen			Randomly Chosen		
	94 Character Alphabet			10 char. alphabet		94 char alphabet
	No Checks	Dictionary Rule	Dict. & Comp. Rule			
1	4	-	-	3	3.3	6.6
2	6	-	-	5	6.7	13.2
3	8	-	-	7	10.0	19.8
4	10	14	16	9	13.3	26.3
5	12	17	20	10	16.7	32.9
6	14	20	23	11	20.0	39.5
7	16	22	27	12	23.3	46.1
8	18	24	30	13	26.6	52.7
10	21	26	32	15	33.3	65.9
12	24	28	34	17	40.0	79.0
14	27	30	36	19	46.6	92.2
16	30	32	38	21	53.3	105.4
18	33	34	40	23	59.9	118.5
20	36	36	42	25	66.6	131.7
22	38	38	44	27	73.3	144.7
24	40	40	46	29	79.9	158.0
30	46	46	52	35	99.9	197.2
40	56	56	62	45	133.2	263.4

NIST Special Publication 800-63, Electronic Authentication Guideline,  
 April 2006. [http://csrc.nist.gov/publications/nistpubs/  
 800-63/SP800-63V1\\_0\\_2.pdf](http://csrc.nist.gov/publications/nistpubs/800-63/SP800-63V1_0_2.pdf)

# Recommendations for Choosing Passwords

Read online!

# Offline Password Guessing

- ▶ Try to guess the password outside of normal operation of system
  - ▶ Attacker has no restrictions on time or computing resources
  - ▶ Guesses are not recorded
- ▶ Passwords are stored on system
  - ▶ Store cryptographic hash of password
  - ▶ Limit read-access to password file/database

# Storing Passwords in Linux

- ▶ User information stored in file `/etc/passwd` (world-readable)
- ▶ Hash of password stored in file `/etc/shadow` (readable by admin only)

`username:$algorithm$salt$hashedpassword$`

$$h = H(P||S)$$

where:  $P$  = password,  $S$  = salt,  $H()$  = algorithm,  $h$  = hashedpassword

- ▶ Salt: random string chosen by system
  - ▶ Increases effective strength of password (*if salt is secret*)
  - ▶ Harder for attacker to use pre-calculated hash values
  - ▶ Two users with same password have different hashes (minor benefit)