

Identification and Authentication

Authentication is the binding of an identity to a subject

Identification and Authentication

Based on one or more of four things

- What you have (token, key)
- What you know (password, pin)
- What you are (fingerprint, retinal scan)
- Where you are (secure building, parking lot)

Klein sample of 15,000 passwords

8%	Dictionary words
4%	Common names
3%	User/account name
2%	Phrases, patterns
1%	Male names
1%	Female names
1%	Uncommon names
1%	Machine names
1%	Place names
1%	King James Bible

Categories of Easily Guessed PWs

- Based on an account
 - account name followed by number
 - account name surrounded by delimiters
- Based on a user's name
 - Initials repeated 0 or more times
 - All letters upper (or lower) case
 - Name reversed
 - First initial followed by last name reversed
- Dictionary word
- Dictionary with words spelled backwards
- Dictionary word with all or some letters capitalized

- Reversed dictionary word with all or some letters capitalized
- Dictionary word with arbitrary letter turned into a control character
- Pattern from the keyboard
- Contains only digits
- Looks like a license plate number
- Acronyms (e.g., UCSB, DOD, IEEE)
- Concatenation of dictionary words
- Dictionary words with all vowels deleted

Gramp and Morris

Unix Operating System Security (1984)

- If login is abc
 Try abc, cba, abcabc
- Comments field
- Finger
- 20 most common female names each followed by a single digit yielded at least one password on every system tried

Gramp and Morris Suggestions

- Make it difficult for an outsider to get a copy of the password file
- Remove encrypted passwords from the password file and put in a parallel file unreadable to the public and UUCP
- Remove comment field
- Modify password program to check for and prevent weak passwords
- Educate users about good and bad passwords or assign passwords

Proactive Password Checking

- Analyze proposed password for “goodness”
 - Always invoked
 - Can detect, reject bad passwords for an appropriate definition of “bad”
 - Discriminate on per-user, per-site basis
 - Needs to do pattern matching on words
 - Needs to execute subprograms and use results
 - Spell checker, for example
 - Easy to set up and integrate into password selection system

Storage

- Store as cleartext
 - If password file compromised, *all* passwords revealed
- Encipher file
 - Need to have decipherment, encipherment keys in memory
 - Reduces to previous problem
- Store one-way hash of password
 - If file read, attacker must still guess passwords or invert the hash

One-way Cipher

An irreversible function \mathbf{f} from plaintext to ciphertext. It is computationally infeasible to determine a plaintext message \mathbf{M} from the ciphertext $\mathbf{C} = \mathbf{f}(\mathbf{M})$

Unix Crypt() Algorithm

- DES with first eight characters of password as the key
- Iterated 25 times on constant 0
- E-table modified to depend on 12-bit salt
 - This is so that DES chip cannot be used for fast encryption
- final 64 bits are unpacked into a string of 11 printable characters

Salt

12-bit random number that is appended to the password supplied by a user

Thus, 4096 encrypted versions of each password

Increases the work of checking a given string against a list of encrypted passwords

Example

3hWG9n3yrXmw6

nBRlcGosLI76w

glaTgrS8galyw

Shadow Password File

- Shadow file contains encrypted passwords
- Shadow file is protected from being read
- Password field in `/etc/passwd` is blank or contains a special symbol or random value

LDAP Server

- Lightweight Directory Access Protocol
- Internet protocol that email and other programs use to look up information from a server
- Used at UCSB for “single signon” to many services

Computer Generated Passwords

3sed77yo

TT6sw45

xc_56RR

Passphrase

- User friendly passwords generated by combining symbols where each symbol is a pronounceable word
- Based on phoneme
 - cv, vc, cvc, vcv
(e.g., **helgoret** or **juttelon**)
- Number of pronounceable passwords of length n is considerably less than number of random passwords of length n

Tunable Passwords

System gives the user part of a password

User constructs new password according to specified rules

Bishop Advice for Strong Passwords

Password should contain at least one of each of the following

- digit
- letter
- punctuation symbol
- control character

Writing Down Passwords

- If you absolutely need to write down a password you should use some transformation scheme such that what is written down is transformed to the actual password
- Bishop gives the following scheme:
 - Capitalize the third letter
 - Add 2 to the end
(e.g., **TuzLeb7** is transformed to **TuZLeb72**)

Password Aging

- The idea is that in the time it takes to guess a password it will no longer be valid
- Force users to change passwords after some time has expired
 - How do you force users not to re-use passwords?
 - Record previous passwords
 - Block changes for a period of time
 - Give users time to think of good passwords
 - Don't force them to change before they can log in
 - Warn them of expiration days in advance

Dictionary Attacks

- Trial-and-error from a list of potential passwords
 - *Off-line*: know *one-way function* and *enciphered passwords*, and repeatedly try different guesses until the list is done or passwords guessed
 - Examples: *crack*, *john-the-ripper*
 - *On-line*: have access to system functions and try guesses until some *guess* succeeds
 - Examples: trying to log in by guessing a password

Type 2 Dictionary Attacks

- The attacker uses the actual login, su, etc. function to try the guess
- Defense is to slow down the attack through
 - backoff
 - disconnection
 - disable
 - jailing

Password Sniffers

- Monitor all information sent over a local area network
- Record the first 20, 50, 128 characters sent over each network connection

One-Time Passwords

- Password that can be used exactly *once*
 - After use, it is immediately invalidated
- Challenge-response mechanism
 - Challenge is number of authentications; response is password for that particular number
- Problems
 - Synchronization of user, system
 - Generation of good random passwords
 - Password distribution problem

Code Book

- List of passwords that are used one time and then never used again
- User looks up next password in the code book
- Can have mathematical algorithm to generate list, then list can be virtual

S/Key

- One-time password scheme based on idea of Lamport
- h one-way hash function (MD5 or SHA-1, for example)
- User chooses initial seed k
- System calculates:

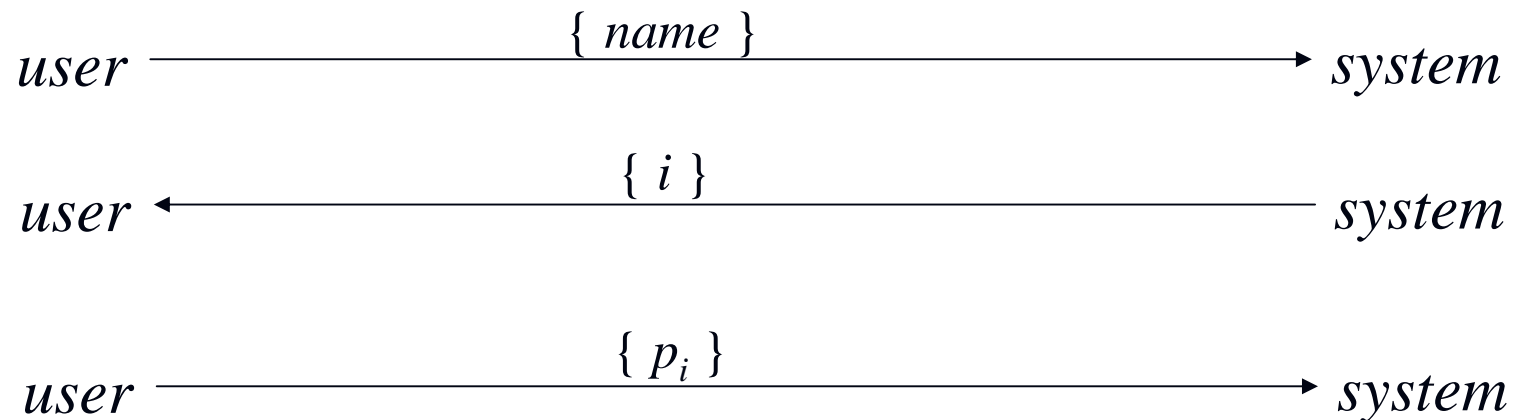
$$h(k) = k_1, h(k_1) = k_2, \dots, h(k_{n-1}) = k_n$$

- Passwords are reverse order:

$$p_1 = k_n, p_2 = k_{n-1}, \dots, p_{n-1} = k_2, p_n = k_1$$

S/Key Protocol

System stores maximum number of authentications n , number of next authentication i , last correctly supplied password p_{i-1}



System computes $h(p_i) = h(k_{n-i+1}) = k_{n-i} = p_{i-1}$. If match with what is stored, system replaces p_{i-1} with p_i and increments i .

Token Cards

- Card has some calculation based on the time and a secret function or serial number
 - Time sensitive
 - Challenge-response
 - May require a user pin

SecurID Card

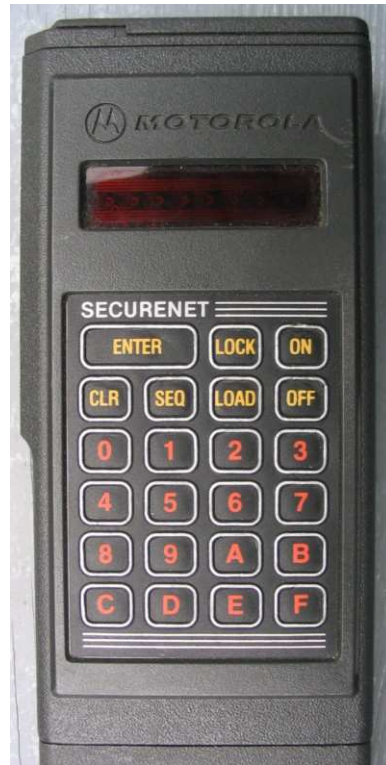
Displays a number that changes every 30-90 seconds



SecureNet Key

- When user logs into a remote machine it displays a number as a challenge
- User enters his/her pin and challenge into the card
- Card returns a response, which the user sends to the remote machine
- Card can be programmed to self-destruct when the wrong pin is entered a certain number of times

SecureNet Key Loader



Biometric and Behavioral Systems

- Depend on unique physiological and behavioral characteristics that can be examined and quantified
- Obtains data from you, converts it to digital representation, and compares it to stored sample
- Should be used in a two-factor authentication system
 - e.g., with passwords

Biometric Systems

- Test actual physical characteristics
- Devices currently available
 - fingerprint
 - handprint
 - retina or iris pattern
 - face pattern

Behavioral Systems

- Test patterns of physiology or behavior
- Devices currently available
 - voice
 - signature
 - keystroke

Ranking in Order of

Effectiveness

- retina pattern
- fingerprint
- handprint
- voice pattern
- keystroke pattern
- signature

Personal Acceptance

- keystroke pattern
- signature
- voice pattern
- handprint
- fingerprint
- retina pattern