6.875 Computer and Network Security **Fall Term, 1997**

Lecture 13 : October 16, 1997

Lecturer: Ron Rivest *Scribe: Todd Parnell*

1 Topics Covered

- User Authentication
- Passwords
- Tokens
- Biometrics
- Recommended Reading

2 User Authentication

2.1 Purposes

There are several reasons why user authentication is necessary:

Computational Power. We have been assuming in our previous discussions that people can do complicated calculations. Encryption, hashing, ZK proofs, etc. all require a user to perform difficult computations. But people don't do computations, machines do. How do we, as people, identify ourselves to a machine?

Access control. The process of of enforcing access rights for network resources. Access control grants or denies permission to a given user for accessing a resource and protects resources by limiting access to only authenticated and authorized users. Computers may be required to control access to computer logins, buildings, or restricted databases.

Authorization. The process of assigning access rights to a user. The access rights include specification, such as whether the user is permitted to read, write, or update a given file. Transactions such as electronic payments require the user to be authorized to prevent fraudulent transactions.

Auditing. It is often useful to log who did what, even if no controls are in place. Such logs facilitate accountability. If any malicious activity occurs, the pertretrator can be held responsible. This is fairly important, especially since most attacks on a computer system are from authorized insiders.

The symmetric case is also of interest. How do we identify electronics? There have been several cases of fake ATM machines put into malls. Users walk up the the ATM, give their card and PIN, and are told that the machine is unable to serve them. At the end of the day, the database of card information together with PINs is removed from the machine and the crooks can use the cards at their leisure.

2.2 Principles

Authentication systems can be based on any of the following three basic principles:

1. Something the user *KNOWS*. This is a *password*. Often credit card companies require you to provide you SSN or mother's madien name to provide verification over the phone.

2. Something the user *POSSESSES*. A device serves as a *token*. Tokens are often things, either physical or electronic. Ordinary keys are examples of tokens.

3. Something the user *IS* (or how he behaves). This principle is known as *biometrics*, or the measurement of some biological property of the user.

Biometrics can be either static (measurements) or dynamic (handwriting recognition, voice analysis, etc).

2.3 Considerations

Designing or choosing an effective user authentication system requires weighing many considerations:

Resistance to deceit, counterfeiting, circumvention. Reasonably determined attackers should not be able to fool the authentication system.

Time to authenticate. The user should not be required to spend too long interacting with the system. Passwords take only a couple seconds, while signing the user's name long-hand or using a retinal scanner take somewhat longer. Doing a DNA analysis today is infeasible. (Though such an authentication scheme is at the heart of a new movie, Gattaca.)

Cost of devices, interfaces, distribution, protection. These change a lot with technology. Currently passwords are very cheap, but DNA analyzers are not. Technology is rapidly decreasing in cost, and more sophisticated methods are becoming possible.

Updating complexity. Keeping the user's authentication information up to date, or adding a new user's authentication information, can be time-consuming. This is especially true of systems like voice recognition which may require many training runs to be able to identify the user's voice.

Processing required. Often CPU time to process the information can be a problem. An authentication technique that took a minute to complete the computations would be disliked by users. Passwords are good in this respect, voice recognition is harder, requiring a FFT.

Reliability; maintainability. Many techniques suffer from reliability problems. For example, a finger print reader might get a smudge on it and become unable to read finger prints correctly, thereby denying an authorized used access.

User acceptability, ease of use, psychological factors. It is important for the users to feel comfortable with the authentication system. This includes factors such as difficulty, effort, and time required to authenticate. It also includes psychological factors, such as a general uneasiness to use their finger prints, since fingerprinting is generally associated with criminal activity. People are also wary to stick their fingers into a mysterious box placed before them. People might also be wary of shining light in their eyes to perform a retinal scan.

Transferability. Passwords are easily transferable, while retinal scans are not. The desired transferability properties depend largely on the system, as giving out passwords can be a security risk, though may also save administrative overhead to grant other users the appropriate privileges. Transferability also suggests more potential for theft.

Cooperation. Most systems require user cooperation. However, prisoners or others not likely to want to be identified. Whether or not this is a concern depends on the application, though for certain systems it is of utmost importance.

Accuracy. Accuracy is usually measured as two quantities, the *false accept rate* (FAR) and the *false reject rate* (FRR). As their names imply, the false accept rate measures the percentage of unauthorized users that get past the security, while false reject rate measures the percentage of authorized users that are denied their proper access. Accuracy is usually measured at the crossover point between the FAR and FRR (see fig. 1).

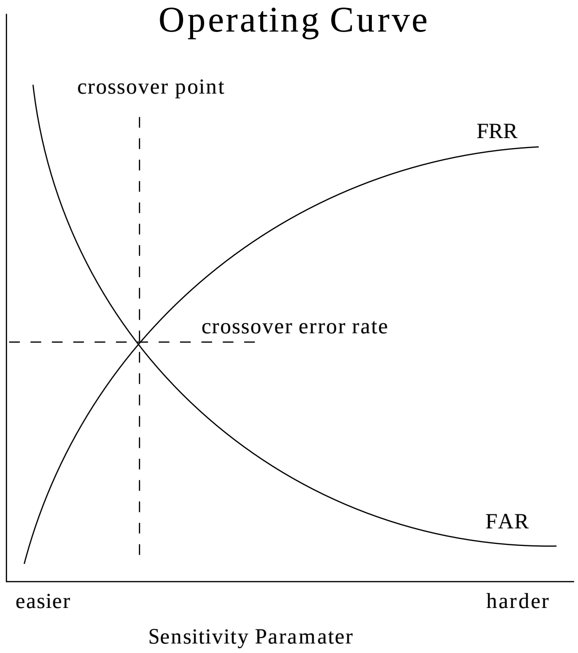

Figure 1: Accuracy Operating Curve

Note that this is often a trade-off: Making false accepts rarer makes false rejects more common, and vice-versa. We can parameterize how easy or difficult is easy for the system to accept/reject. For military applications, we can make the system very hard to get into, but for commercial applications we must back off on the operating curve. Banks with high FRR's would be unlikely to keep annoyed customers denied access to their records.

2.4 History

- The human species depends on recognizable identity. What would society be like if we were all isomorphic sardines? In fact, some theories of evolution emphasize the complexity of social situation, especially involving deceit, as an explanation for the evolution of the human brain.

- King's messengers used to carry signet rings to identify themselves.

- Some evidence has been found that the Chinese used fingerprints 2000 years ago. (See *Computer Security Handbook*, page 38.)

- The Bertillon system, which dates back to 1870, was used in the United States for criminal identification. This system identifies a person by a composite of many measurements such as height, trunk height, arm span (both arms outstretched), forearm length, finger length, foot length, and head width. However, a famous case in the U.S. in 1903 in which two "Willie West"'s with identical Bertillon measurements were found. The day after that, fingerprints won as the dominant form of criminal recognition. (See *Computer Security Handbook*, page 38.)

- Passwords have long been used in war-time scenarios.

3 Passwords

Passwords are cheap, easy, and reasonably accurate, and consequently, are in wide use. Note: much of this material is carry-over from previous years lecture, and was not covered in this depth.

3.1 Properties

Passwords should have two basic properties:

1. Hard to guess.
2. Easy to remember.

These requirements conflict! Something which is easy to remember has small entropy, and is consequently easy to guess.

3.2 Vulnerabilities

Morris and Thompson did a survey (CACM, Nov. 1979, 594-597) of 3289 passwords, and found 2831 (86%) vulnerable for one of the following reasons:

15	were a single ASCII character
72	were a string of two ASCII characters
464	were a string of three ASCII characters
477	were a string of four alphanumeric
706	were a string of five letters, all the same case (upper or lower)
605	were a string of six letters, all lower case. Note that $26^6 = 309M$, which is not too large

Other vulnerabilities to passwords include: Generic:

Common words. English or other languages.

Common names. TV, musicians, significant other, pet, friends, family members, nicknames.

User-specific:

Easily obtained information. Birthdays, license number, phone numbers, social security number, make of car, number or street where user lives.

Keyboard patterns. Something like “qwerty”.

Passwords on other systems.

Permutations of these. Especially backwards.

The Internet Worm (November 1988) tried the following attacks:

- No password
- User name
- User name appended to itself
- Nickname

- Last name
- Last name spelled backwards
- Personalized 432-word dictionary attack

3.3 Choosing a password

Some good ideas for creating passwords:

- Mix upper and lower case
- Use something unprounceable, such as “stowishy.”
- Include non-alphanumeric characters
- Mix numbers and letters
- Perform a systematic substition, such as $o \rightarrow 0$ or $l \rightarrow 1$
- Pick letters from a longer passphrase or sentence
- Computer generation. This generates hard-to-remember combinations. Also, program might have a bug. For example, Professor Rivest’s wife, Gail, discovered that the lock given to her by her gym had the same combination as many others in the gym by accidentally opening the wrong locker with the combination she was given.

3.4 Storing Passwords

Passwords must be stored somewhere. The user must remember the password, either by memorizing it or by writing it down, which some argue is a bad idea, though is it really that bad? In addition, the computer must remember the password so that it can be checked when the user presents it.

There are several applicable techniques:

In the clear on file system. This is not a very good alternative, as a user that gains access to the file has all of the passwords.

On a dedicated authentication server. This is somewhat better, though a compromise of the authentication server will still reveal users' passwords.

Encrypted. This way, compromising the password file will not reveal users' passwords. But, we don't need to be able to decrypt, so...

Hashed. Use a one-way hash function, such as MD5. When the user presents the password, it is hashed and compared against the stored value. Knowledge of the hashed password is inadequate to authenticate oneself to the machine. Another useful trick that can be layered onto a hash is *salting*. Salting involves storing a value $hash(pwd, salt), salt$, where *salt* is a per-user (possibly randomly chosen) value. Salting prevents pre-computation of hashes by an adversary, which makes breaking more common passwords at least a little more difficult. Furthermore, reuse of passwords, either by two users with the same password or one user with the same password on two systems, will not be evident from the salted hashes.

With access controls. It is often useful for all users to be able to access some of the information in the password file, but not have access to the actual passwords. Many systems break the password file into two pieces, one with useful user information, such as the user's default UNIX shell, and another "shadow" password file that is stored in a secret place that contains the actual passwords.

3.5 Changing passwords

Changing passwords frequently improves security; however, it makes passwords harder to remember. And, if very good, hard-to-remember, hard-to-guess passwords are used, there will be even more resistance to changing passwords. DoD says that passwords should be changed at least once a year. It is also generally not a good idea to reuse passwords, especially recent ones. Some systems do not allow a user to change their password to one recently used. Also, some systems restrict the frequency of password changes, while other require periodic changes. It is also generally advisable to change a password after using it in a way which might have compromised it. For example, Jeff Schiller, who works for MIT DCNS, always changes his password when he returns from vacation, since he might have compromised his password by logging in remotely. As usual, a risk analysis should be performed to find the optimal compromise.

3.6 One-time passwords

The fundamental idea behind one-time passwords is to provide the user with a list of passwords, each of which works exactly once. Since the passwords are only used once, eavesdropping is much less of a problem. The user will compromise his password when he types it, but the password will no longer be valid, so this is not a problem.

4 Tokens

4.1 Types

Physical key. These are normal keys. Certain key systems in use in Europe have a double key system, where one key opens and another closes.

Magnetic strip cards. Examples include credit and ATM cards.

Smart Card. Same size and shape as a mag strip card, but it contains an IC. This type of token is the subject of much research and debate. Among the questions, though, is “Who’s security concerns are we representing with a smart card?” With a smart card, there are three parties in any transaction (user, smart card, foreign system.) This also presents difficulties.

4.2 Reproduction

How hard is it to reproduce a token?

Keys. Ordinary keys are very easy to duplicate. Newer high-security keys have pits and groves on all sides, though, increasing the security offered at least some.

Mag strip cards. More security than a key, but reproducible with very limited funds and a bit of time. Possible to copy magnetic domains from one card to another using an iron.

Smart Cards. Much more difficult to reproduce than mag strip cards.

5 Biometrics

Biometrics is the use of a person's physical characteristics for authentication. Biometrics are generally not as good as passwords since error rates under 1% are hard to achieve. Passwords, while they can be compromised, do not have any false rejects. Among the biometrics we will look at are:

- Voice recognition
- Signature dynamics
- Fingerprints
- Hand geometry
- Retinal scan
- Iris Scan
- Facial recognition
- Typing characteristics

When evaluating biometrics, we are concerned with the time it takes to measure each metric, the equipment expense, the user acceptability of the method, and false-acceptance and false-rejection rates. When looking at false-acceptance and false-rejection rates, we often look at the equal-error error rate. This is the point on a graph of percent error vs. threshold set for a match, where the false-acceptance and false-rejection rates are equal.

5.1 Voice Recognition

The first problem when designing a voice recognition system is how to recognize the voice. Pitch doesn't work too well so the method often used is modeling the characteristics of the resonant system of the vocal tract using linear predictive coding (LPC). Voice recognition has many similarities to work in speech recognition.

There are many problems with speech recognition that make it unpopular with people who use it. It requires a tedious enrollment process for the computer to learn users' voices. It's effectiveness depends upon ambient noise levels. For example, if a train goes by while you are trying to authenticate yourself, the authentication will probably fail. Also, voices tend to change both over time and when users are sick, increasing false-rejection rates. Finally, many people just do not like talking to a computer.

A couple of technological problems also make voice identification difficult. It requires a lot of complicated processing including procedures such as Fourier transforms. Additionally, replays are fairly easy because all one has to do is record a person saying the pass phrase and replaying it to the computer. One way to protect a system against such replay attacks is the computer can display a different pass phrase that the user must read for each authentication. This makes replays more difficult.

5.2 Signature Dynamics

As with voice identification, the simple solution for signature recognition doesn't work very well. Just comparing signatures for exact copies has problems because they are easy to forge and most people have lots of variation in their signatures. To solve this, more complicated dynamics are used such as pressure, time, velocity and acceleration in addition to the x and y coordinates of the signature. The processing is similar to voice identification where you compare the characteristics of each segment in the sequence that constitutes the signature.

Signature dynamics is not too expensive to implement, but does have some problems. While it allows for some variation in individuals' signatures, it won't accept all variations. It is also prone to forgery. One way to help reduce the forgery problem is to use a similar method of changing pass phrases as described above.

5.3 Fingerprints

Fingerprint analysis is a fairly straight forward comparison of the characteristics of a user's fingerprints such as arches, loops, whorls, etc. (see page 59 of the *Computer Security Handbook*). While it has a fairly simple interface, consisting of placing a hand on a scanner, it also has several problems. The transducer needs to be well done to be accurate, though it is not perfect and getting dirt on it can lower the acceptance rate. It's accuracy is also dependent upon the finger that is being identified. The technology required to implement the scanner is relatively expensive. Additionally, there is a social stigma against using fingerprints since they are associated with their use for criminal identification.

5.4 Hand Geometry

Hand geometry involves measuring the physical characteristics of the user's hand to determine identity. Measurements such as the lengths of the fingers, width of the hand, and finger thickness are used. This is one of the best biometric methods in terms of accuracy and user acceptance. It is questionable about how easy it is to fake out this method.

ID3D system - 9 byte template, false accept=false reject = 0.2%, 1.2s verify time and costs $3,000

5.5 Retinal Scans

A 1935 study by Simon and Goldstein showed that the pattern of blood vessels in the retina, like fingerprints, are unique to every individual. This means that by taking an infra-red scan of a person's eye one can match the pattern to that of a known template using a technique called Fourier transform cross-correlation. This technique is not popular with users since it is not comfortable and requires a light to be shined into one's eye. Another problem with this method is the alignment of the head in relation to the scanner is critical to make a positive identification.

Eyedentify Inc. - 35 byte template, false accept= 0% false reject is low, and costs $4,000

5.6 Iris scan

Similar concept to fingerprinting, but looks for patterns in the iris. This method gets potentially six times more information than a fingerprint.

Iriscan Inc. - 256 byte template, false accept=false reject = 1/131,000 and 100ms verify time

5.7 Facial Recognition

Work on developing facial recognition techniques is currently being done by Tommy Poggio's group at the AI Lab. It is modelled after how we recognize each other–by sight. The computer takes a picture of the user's face and compares it to a picture it has stored on disk. The technology for this is still being developed, but probably won't be ready for use for another ten to twenty years.

Neuromatic Vision Systems, Inc. - neural network and database, costs $30,000 and 20 people per second.

5.8 Typing and Mouse Characteristics

A final technique is to recognize a user by the characteristics of his typing or use of the mouse. This was commonly used during World War I to determine which operator was typing the German's Morse code. This method can have problems if the user hurts his hand or his typing style changes.

5.9 Sandia Study of Biometrics

A study done by Sandia National Labs compared the effectiveness of commercially available biometric equipment. The crossover error rates for the methods studied are summarized in the table below:

Technique	**Error Rate**
Voice (Alpha)	3%
Voice (ECCO)	2%
Signature	2%
Retinal scan	0.4%
Hand geometry	0.1%
Fingerprint	9% false reject, no false accepts

They also compared other characteristics of the techniques:

Characteristic	**Best**	**Worst**
User acceptability	Hand	Voice
False reject	Hand	Fingerprint
False accept	Hand, Retina, Fingerprint	Voice
Throughput	Hand, Retina, Fingerprint	Voice, Signature
Template size	Retina	Voice
Difficulty of imitation	Retina	Voice, Signature
Cost	Voice	Retina

As mentioned above, these biometrics are not big improvements over passwords because of their high error rates. They can be used in conjunction passwords, though, so that the system checks both what the user is and what the user knows.

6 Recommended Reading

Levy, Steven. *Heroes of the Computer Revolution: Hackers.* Dell Publishing, 1994.