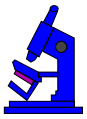
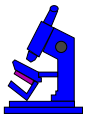



Chapter 9

Lecture 9 - System (in)security


Admininstration

- ✔ Results are out - please check.
- ✔ Assignment 1
  - ✔ hardcopy to tutor or me,
  - ✔ email softcopy to me.
- ✔ Assignment 2 ... better get going.

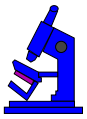


Last session

- Kerberos
- Voting
- Contract signing

This session

- Ethics and computing
- Organizations and standards
- UNIX passwords
- NT passwords



Stranger danger...

*One of my sons was taught stranger-danger at his school. We were asked to quiz him afterwards, so we asked him if he should accept a lift in a car with a stranger. He immediately replied "**No way!**". We then asked: "**What if he offered you sweets?**", but he still replied "**No way!**". Finally we asked: "**Why not?**", to which he replied "**Because you might not get any!**"*

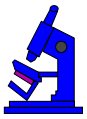
Ethics

Moral development stages:

Stage 1: *Obedience and punishment*

...

Stage 6: *Individual principles of conscience* - an orientation not only toward existing social rules, but also toward the conscience as a directing agent, mutual trust and respect, and principles of moral choice involving logical universalities and consistency. If one acts otherwise, self-condemnation and guilt result.



Ethics

- ✔ It is my expectation, and requirement, that you are able to maturely evaluate rights and wrongs.
- ✔ In these sections of the course, I will be outlining systems which demonstrate poor cryptographic techniques, and as a result, can be defeated.
- ✔ A more cynical view might be that *I am teaching hacking*

...this is not my intent...

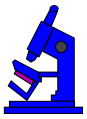
Ethics and computing

No new ethical dilemmas... Perhaps the only significant difference is that the computer crimes are so easy.

Software duplication: = *theft*.

Using information: = *insider trading.*

E-mail abuse: = *abuse.*



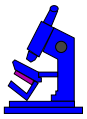
Network administrator's dilemma

- ✔ Network administrators often come to learn things about their 'clients'
- ✔ Without asking the client, they should not make use of that information.
- ✔ The network administrator's dilemma: How to control bad-guys without trampling over rights.

Professional codes of ethics

- ✔ Most professional bodies[13] have formal written codes of ethics
- ✔ The computer industry has yet to develop a standard code of conduct
- ✔ If computer crime continues to rise, codes may be imposed on it.

[13] For example: Medical boards.



ACS code of ethics

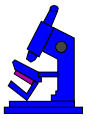
1. I will serve the interests of my clients and employers, my employees and students, and the community generally, as matters of no less priority than the interests of myself or my colleagues.
   ...

Within a general framework of ethical and moral responsibility, codes such as this one can help clarify *grey* areas of concern.

Insecurity - threats are real

For example:

- Pentagon machines were repeatedly corrupted by unknown intruders during the Gulf war. The intruders appeared to be doing it as part of a contest.
- German hackers demonstrated on TV a method of transferring money into their own accounts using ActiveX controls downloaded to an unsuspecting person's machine.
- Estimates of computer theft in the US range from 1 to 30 $billion/year - most of which goes unreported.



Taxonomy of insecurity?

Each new attack adds new levels to the structure:

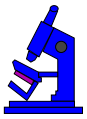
- physical insecurity, and
- password insecurity

Some of the security of modern systems is provided through cryptographic techniques (particularly password storage), the subject today.

Non-cryptographic cracking

Misconfiguration: If excessive permissions exist on certain directories and files, these can lead to gaining higher levels of access. For example, on a UNIX system, if /dev/kmem is writable it is possible to rewrite your UID to match root's.

Poor SUID: Sometimes there are scripts (shell or Perl) that perform certain tasks and run as root. If the scripts are writable by you, you can edit it and run it.



Non-cryptographic cracking

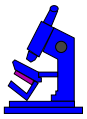
Buffer overflow: Buffer overflows are typically used to spawn root shells from a (server) process running as root.

Race conditions: A race condition is when a program creates a short opportunity for attack by opening a small window of vulnerability. For example, a program that alters a sensitive file might use a temporary backup copy of the file during its alteration.

Non-cryptographic cracking

Poor temporary_files: Many programs create temporary files while they run. If a program runs as root and is not careful about where it puts its temporary files and what permissions these files have, it might be possible to use links to create root-owned files.

Attacks using these methods can be launched locally on the target machine, or often remotely, by exploiting *services* with loopholes.



Protection

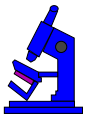
Can you protect yourself against attacks?

- **Hack/crack yourself:**
- **Be vigilant:**
- **Reduce reliance:**
- **Use more secure systems:**
- **Update systems:**

Finally: "*Its not the end of the world!*"

This session

- Ethics and computing
- Organizations and standards
- UNIX passwords
- NT passwords

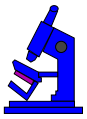


Computer Emergency Response Team

The CERT Coordination Center is the organization that grew from the computer emergency response team formed by the Defense Advanced Research Projects Agency (DARPA) in November 1988 in response to the needs identified during the Internet worm incident. The CERT charter is to work with the Internet community to facilitate its response to computer security events involving Internet hosts, to take proactive steps to raise the community's awareness of computer security issues, and to conduct research targeted at improving the security of existing systems.

CERT

- ✔ If you are ever involved in a computer security incident it is useful to get in touch with CERT.
- ✔ They provide incident reports and advisories, and can liaise with other system administration people if the attack on your system comes from outside your organization.



CERT Incident Note IN-99-04

Here is an excerpt from an incident report:

Similar Attacks Using Various RPC Services

Thursday, July 22, 1999

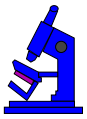
Overview

We have recently received an increasing number of reports that intruders are using similar methods to compromise systems. We have seen intruders exploit three different RPC service vulnerabilities; however, similar artifacts have been found on compromised systems.

...

SIGINT

- ✔ *Signals Intelligence* (SIGINT) broke the Japanese military code and learned of plans to invade Midway Island.
- ✔ In 1943 they began the VENONA project to examine encrypted Soviet diplomatic communications.
- ✔ The messages were double-encrypted and were extremely difficult to crack.
- ✔ Almost all of the US KGB messages in 1944 and 1945 were broken between 1947 and 1952.

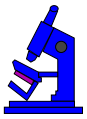


NSA - National Security Agency

- ✔ Successor of SIGINT
- ✔ *The National Security Agency is the USA's cryptologic organization.*
- ✔ *It coordinates, directs, and performs highly specialized activities to protect U.S. information systems and produce foreign intelligence information.*

NSA - National Security Agency

- ✔ *NSA employs the country's premier codemakers and codebreakers.*
- ✔ *It is said to be the largest employer of mathematicians in the United States and perhaps the world.*



Rainbow documents

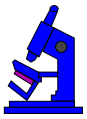
✔ The NSA created various documents describing the criteria for evaluating the security behaviour of machines.

✔ These criteria were published in a series of documents with brightly coloured covers, and hence became known as the *Rainbow* series. (red book, yellow book...)

C2 security

DOD 5200.28-STD - "Department of Defense Trusted Computer System Evaluation Criteria":

- To provide a standard to manufacturers (for security features related to confidentiality)...
- To provide DoD components with a metric with which to evaluate the degree of trust...
- To provide a basis for specifying security requirements in acquisition specifications.



C2 security example

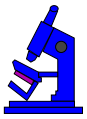
- ✔ *The TCB[14] shall require users to identify themselves to it before beginning to perform any other actions that the TCB is expected to mediate.*
- ✔ *Furthermore, the TCB shall use a protected mechanism (e.g., passwords) to authenticate the user's identity.*

[14] Trusted Computing Base.

Microsoft and C2

Windows NT Workstation vs 3.5 with U.S. Service Pack 3 was the first Microsoft product that has completed C2 testing, and is only certified if using the same hardware, and installed software, and does not include any network connection. The NT utility ***c2config.exe*** sets up an NT system to pass the C2 tests.

The 1998 attacks on the Pentagon involved theft and modification of data, as well as denial-of-service. The attacked machines were C2-secure Windows NT machines.



UNIX and C2

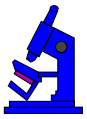
Many UNIX systems have also got C2 certification, and come configured this way from the manufacturer.

> *There are numerous examples of hacked UNIX systems found on the Internet. In 1996, a site I managed in New Zealand was the target of a malicious attack by intruders from Australia and Belgium.*

Given all this, C2 certification is probably not a good guide as to the security of your system.

This session

- Ethics and computing
- Organizations and standards
- UNIX passwords
- NT passwords



Password security

- ✔ Morris and Thompson article:

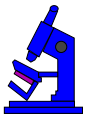
  http://citeseer.nj.nec.com/morris79password.html

- ✔ Computer generated passwords more predictable than user ones...

UNIX password security

- ✔ UNIX systems are traditionally open systems, given their background in university environments.
- ✔ As such, the security on them is often minimal.
- ✔ It is common for UNIX accounts to be made available relatively freely.
- ✔ For example, at the MIT Media lab[15] all computers have been password-free until recently.

[15] MIT - home of Kerberos!



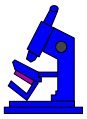
UNIX password security

- ✔ UNIX systems are vulnerable to a wide range of attacks, particularly internal attacks.
- ✔ All Unix systems have a ***root*** account.
- ✔ This account has a UID and GID of zero, and once root access is obtained on a UNIX system, there is very little that *cannot* be done.

UNIX accounts

Account passwords are constructed to meet the following requirements:

- Each password has at least six characters.
- Only the first eight characters are significant.



UNIX accounts

There are many other accounts found on Unix systems, not just those for clients:

sysadm - A System V administration account, and

daemon - A daemon process account, and

uucp - The UUCP owner, and

lp - The print spooler owner.

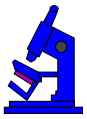
When protecting a UNIX system, we must protect all these accounts - not just root.

UNIX password file

✔ Account information is kept in a file called /etc/passwd.

✔ It normally consists of seven colon-delimited fields, and may look like the following:

```
hugo:aAbBcJJJx23F55:501:100:Hughs Account:/home/hugo:/bin/tcsh
```



/etc/passwd fields

hugo: The account or user name.

aAbBcJJJx23F55: A one-way encrypted (hashed) password

501: The UID - unique user number

100: The GID - group number for user.

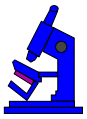
Hughs Account: Account information.

/home/hugo: The account's home directory

/bin/tcsh: A program to run when you log in

UNIX passwords

- ✔ When you log in with your account name and password, the password is encrypted and the resulting hash is compared to the hash stored in the password file.
- ✔ If they are equal, the system accepts that you've typed in the correct password and grants you access.

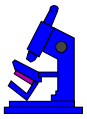


UNIX passwords

- ✔ UNIX uses a DES-like algorithm to calculate the encrypted password.
- ✔ The password is used as the DES key (eight 7-bit characters make a 56 bit DES key) to encrypt a block of binary zeroes.
- ✔ The result of this encryption is the hash value.
- ✔ Note: the password is not encrypted, it is the key used to perform the encryption!

UNIX salt

- ✔ A strengthening feature of UNIX is that it introduces two random characters in the alogrithm (the salt).
- ✔ This ensures that two equal passwords result in two different hashes.
- ✔ From viewing the UNIX password file you can not tell if two persons have the same password.



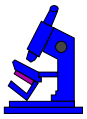
UNIX salt

- ✔ To prevent crackers from simply encrypting an entire dictionary and then looking up the hash, the salt was added to the algorithm to create a possible 4096 different hashes for a particular password.
- ✔ This lengthens the cracking time because it becomes a little harder to store an encrypted dictionary online as the encrypted dictionary now would have to take up 4096 times the disk space.
- ✔ This does not make password cracking harder, just more time consuming.

Crypt code

Sample crypt code from LINUX uClibc. The code has the following structure:

```
extern char * crypt(const char *key, const char *salt) {
       /* Are we supposed to be using the MD5 replacement
       /* instead of DES...  */
    if (salt[0]=='$' && salt[1]=='1' && salt[2]=='$')
        return __md5_crypt(key, salt);
    else
        return __des_crypt(key, salt);
}
```

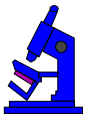


Cracking

- ✔ It is very time consuming, but given enough time, brute force cracking ***will*** get the password.
- ✔ The hashed passwords are compared with the entry in the */etc/passwd* file.
- ✔ BTW - You cannot try to log in using all the possible passwords, as UNIX systems enforce 10 second timeouts after three consecutive login failures.

Dictionary cracking

- ✔ Dictionary password cracking is the most popular method for cracking Unix passwords.
- ✔ The cracking program will take a word list, and one at a time try to crack one or all of the passwords listed in the password file.
- ✔ Some password crackers will filter and/or mutate:
  - ✔ substitute numbers for certain letters,
  - ✔ add prefixes or suffixes,
  - ✔ or switch case or order of letters.



Dictionary cracking

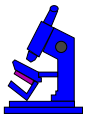
- ✔ A popular cracking utility is called ***Crack*.**
- ✔ Crack can use user-definable rules for word manipulation/mutation to maximize dictionary effectiveness.
- ✔ Crack merges dictionaries, turns the password files into a sorted list, and generates lists of possible passwords from the merged dictionary or from information gleaned about users from the password file.

/etc/shadow passwords

Once the password hashes are moved to the shadow file, its permissions are changed as follows:

```
opo 35# ls -l /etc/shadow
-r--------   1 root     sys      3429 Aug 20 14:46 /etc/shadow
opo 36#
```

These permissions ensure that ordinary users are unable to look at the password hashes, and hence are unable to try dictionary attacks.



This session

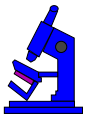
- Ethics and computing
- Organizations and standards
- UNIX passwords
- NT passwords

Microsoft password security

Two one-way password hashes are stored on NT systems:

- a LanManager hash, and
- a Windows NT hash.

The LanManager hash supports the older LanManager protocol originally used in Windows and OS/2. In an all-NT environment it is desirable to turn off LanManager passwords, as it is easier to crack. The NT method uses a stronger algorithm and allows mixed-cased passwords.



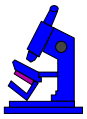
Microsoft password security

- ✔ The database containing these hashes on an NT system is called the SAM (Security Access Manager)
- ✔ If you have administrative access[16], the program ***pwdump*** can extract the hashes.

[16] Originally, *anyone* could extract the hashed passwords from the SAM, as Microsoft believed that "if they didn't tell anyone the algorithms they used, no-one could discover what they had done". Security through obscurity is not a safe strategy, and Jeremy Allison was able to de-obfuscate the SAM entries relatively quickly.

Microsoft salt

- ✔ Microsoft does not *salt* during hash generation, so once a potential password has generated a hash it can be checked against ***all*** accounts.
- ✔ The cracking software takes advantage of this.

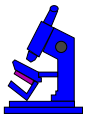


LanManager encryption

- ✔ LanManager encryption is created by taking the user's plaintext password, capitalising it, and either truncating to 14 bytes, or padding to 14 bytes with null bytes.
- ✔ This 14 byte value is used as two 56-bit DES keys to encrypt an eight byte value, forming a 16 byte value which is stored by the server and client.
- ✔ This value is known as the ***hashed password***.

NT encryption

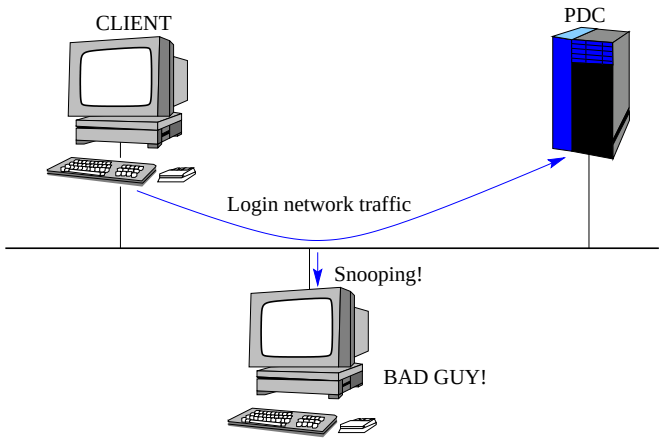
- ✔ Windows NT encryption is a higher quality mechanism, consisting of doing an MD4 hash on a Unicode version of the user's password.
- ✔ This also produces a 16 byte hash value that is non-reversible.

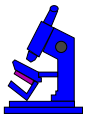


NT Password security

- ✔ Note that the LANManager hash is similar to UNIX level of cyptography
- ✔ The NT hash is better
- ✔ But... neither use strong encryption, and
- ✔ the network login mechanism has some problems.

Challenge response



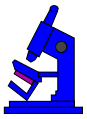


Challenge-response protocol

- ✔ When a client wishes to use a resource, it first requests a connection and negotiates the protocol that the client and server will use.
- ✔ In the reply to this request the server generates and appends an 8 byte, random value - this is stored in the server after the reply is sent and is known as the ***challenge***.
- ✔ It is different for every client connection.

Challenge-response protocol

- ✔ The client then uses the hashed password (16 byte values described above), appended with 5 null bytes, as three 56 bit DES keys, each of which is used to encrypt the challenge 8 byte value, forming a 24 byte value known as the ***response***.
- ✔ This calculation is done on *both* hashes of the user's password, and *both* responses are returned to the server, giving two 24 byte values.



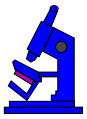
Challenge-response protocol

- ✔ The server then reproduces the above calculation, using its own value of the 16 byte hashed password and the challenge value that it kept during the initial protocol negotiation.
- ✔ It then checks to see if the 24 byte value it calculates matches the 24 byte value returned to it from the client.
- ✔ If these values match exactly, then the client knew the correct password and is allowed access.

Challenge-response protocol

There are good points about this:

- The server never knows or stores the *cleartext* of the users password - just the 16 byte hashed values derived from it.
- The cleartext password or 16 byte hashed values are never transmitted over the network - thus increasing security.



Challenge-response protocol

However, there is also a bad side:

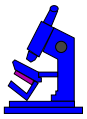
- The 16 byte hashed values are a "password equivalent". You cannot derive the users password from them, but they can be used in a modified client to gain access to a server.

- The initial protocol negotiation is generally insecure, and can be hijacked in a range of ways. One common hijack involves convincing the server to allow clear-text passwords.

Challenge-response protocol

✔ Despite functionality added to NT to protect unauthorized access to the SAM, the mechanism is trivially insecure

✔ Both the hashed values can be retrieved using the network sniffer mentioned before, and they are as-good-as passwords.



Attack

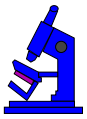
- ✔ Relies on flawed mechanism.
- ✔ Even *without* network access, it is possible by various means to access the SAM password hashes, and *with* network access it is easy.
- ✔ The hashed values are password equivalents, and may be used directly if you have modified client software.
- ✔ The attack considered here is the use of either a dictionary, or brute force attack directly on the password hashes (which must be first collected somehow).

Attack

L0phtCrack is a tool for turning Microsoft Lan Manager and NT password hashes back into the original clear text passwords. It may be configured to run in different ways.

Dictionary cracking: L0phtCrack running on a Pentium Pro 200 checked a password file with 100 passwords against a 8 Megabyte (about 1,000,000 word) dictionary file in under one minute.

Brute force: L0phtCrack running on a Pentium Pro 200 checked a password file with 10 passwords using the alpha character set (A-Z) in 26 hours.



Attack time

Character set size	Size of computation	Relative time taken
26	$8.353 * 10^9$	1.00
36	$8.060 * 10^{10}$	9.65
46	$4.455 * 10^{11}$	53.33
68	$6.823 * 10^{12}$	816.86

So if 26 characters takes 26 hours to complete, a worst-case scenario for 36 characters (A-Z,0-9) would take 250 hours or 10.5 days. A password such as ***take2asp1r1n*** would probably be computed in about 7 days.

Microsoft base security fix

1. Disable the use of LanManager passwords.
2. Don't log in over network as administrator
3. Encrypt all network traffic
4. Use long passwords, and all allowable characters
5. Use an alternative login system
6. Use an unsniffable network cabling system.