Sharing and Privacy in a Computer Utility

ILO Symposium

4/14/69

Talk will be in two parts

~~3.~~ 1. Innerview	(Saltzer)	20 m.	25
2 Outerview	(Corbató)	20 m.	25
			50

Innerview: Technical aspects of Controlling Information Sharing

Outerview: System-wide implications; coupling to social aspects

Two-part structure is to emphasize the need to look at this area from two sides.

Groundrules for this Talk (to provide some common structure)

1. Subject is the computer utility: it stores information for many users; it permits remote (interactive) access to stored information.

2. Information sharing is a key service of the computer utility. (Our whole discussion centers on how to control this service.)

~~At the risk of sounding negative about things, we will promise that~~

View: there are simple ways of providing "all-or-nothing" sharing but they do not provide the necessary gradations of control.

☞ Control introduces complexity. ☜

4/14/69

Without belaboring why, need merely mention proposals for

1. National Data Bank
2. On-line medical system
3. Automated Stock Exchange

to evoke visions of need to control the sharing abilities

Let us delve into the technical aspects by dividing the problem arbitrarily into three general areas:
(not sacred)

1. Authentication (identify the user)
2. Protection (keep him under control)
3. Certification (are you sure 1 + 2 work)

We will split off (and not discuss) two equally important areas:

1. Communications Security
2. Equipment radiation

We will concentrate on the ordinary user, at his remote terminal.

Simplest area is <u>authentication</u>

✯ "Challenge - Response"
(e.g., give me your password)

~~Eys~~
Psychology
Dynamics

opportunity to explore
1. psychological interface

2. Dynamic usage;
special considerations

Observations.

1. For purposes of log keeping, and ease of changes,
password is per-<u>person</u>
<u>not</u> per-project
or per-information item

Why?
1. You want a record of who logged in and when
2. If you ~~change~~ decide to change a person's access to a file do you have to tell everyone (else) the new password?
3. Widespread knowledge of a password makes control harder.
- increases probability of ~~loss~~ exposure
- lower ability to figure out who leaked it
- lower record of who logged in.

~~Mattis has a proposed <u>proxy</u> scheme -~~

2. Techniques
1. Turn off printer (psychological)
2. Allow user to change his own (Avoids [illegible]

~~2½ Should not be user-generated~~

~~3. One-time password helps keep user covered.~~

~~- less vulnerable~~

~~- if user can't log in he knows that password file has leaked.~~

~~4. Keep password list decentralized... and only from user list.~~

Other proposals:

I D card — (encourages passing it around)

thumbprint reader —

Observe typing pattern —

they tend to lack the appeal of simplicity —

5. Console identification helps localize user.

CTSS can limit the consoles you may log in from.

(N.B. flaw in Telephone network)

★ Specification

★ Enforcement

<u>Protection:</u>

2 general points

1. Specify who (be sure to make it open ended (groups)) or the whole thing falls of its own weight.

2. Enforce it: List of techniques suggests wide ranging nature of the problem. [Four highlights illustrate range]

→ a. Compare user name with access list ~~#~~ on every reference to information. Requires hardware help; very complex when dynamically changing access is taken into account.

→ b. Hardware "protection state variable" in CPU to limit what it can do when user has control.

c. Core and drum areas must be cleared when reallocated

d. Duplicate copies of files (for reliability) must be protected

e. Hardware instruction are all decoded completely. All "undefined" operations cause defined traps.

f. I/O locations must be verified

→ g. Local memories in multiple CPU's must be clearable, e.g., when an access control list is changed.

h. Multiple ring, hardware protected supervisor, to minimize extent of potential exposure to an accident

→ i. (Presume system is self maintaining) Source files to generate system must be equally well protected.

j. Decentralize password file

k. Clear all storage before returning to maintenance group.

l. Ability to lower mainframe must be controlled

m. time delay on mistakes to discourage methodical probing.

At this point we begin to shade into system issues –

so let's switch to Corbató