Massachusetts Institute of Technology

Project MAC

Computation Structures Group Memo No. 59

# Surveillance Mechanisms in a Secure Computer Utility

Leo J. Rotenberg

March 1971

This memo will be published in Computers and Society, the ACM SIGCAS newsletter, in March 1971.

Work reported herein was supported (in part) by Project MAC, an M.I.T. research program sponsored by the Advanced Research Projects Agency, Department of Defense, under Office of Naval Research Contract Number Nonr-4102(01).

# Surveillance Mechanisms in a Secure Computer Utility

## Information Explosion

There is a growing trend in America to use computers to keep better records and collect more data related to the normal activities of persons in our society. Since it is not reasonable to expect this trend to reverse itself, our society is faced with the necessity of embracing the use of computers to handle personal data. Computer networks will no doubt be used to collect, store, collate, analyse, and disseminate data. This usage poses a threat to privacy, to the extent that information is disclosed without the consent of the individuals identified by the data.

The formation of several computer networks has already begun, and multi-access time-sharing computers are among those included. Any data base maintained on-line to such a system, if it contains information of value to an intruder, will present itself as a target for penetration. A would-be intruder could pretend to be a normal user of the system, and attempt a penetration from within.

Hardware and programming errors will surely create some holes in a computer system's defenses, and they will be found unpredictably but probably by someone looking for them. Some of the surveillance mechanisms presented here will help to deter such a search, while others are better described as societal tools whose purpose is to tame the power created by the computer.

## A Model

A secure computer Utility is one which will not allow unauthorized release of data. It must include an authorization mechanism for computing objects which permits controlled sharing, and a protection mechanism for running programs which protects them from other programs. The protection mechanism must also protect the supervisor program, and the input/output operations it coordinates.

The model of a secure computer Utility on which this discussion is based is developed fully in (2). The model includes elements from Multics (1), particularly the hierarchical file system in which tree names are used to name directories, files, and sphere gateways. Access control lists (ACLs) implemented in directories are used to specify where and how files and directories may be accessed. While any user can name any file with a tree name, ACLs restrict access to authorized users. Files may be transformed into segments in spheres of protection, where programs and data are brought together. A computation is a collection of spheres (of protection), together with at least one process. Each process must be in some sphere, and a process in one sphere cannot access segments of other spheres, except shared segments. A process is allowed to call from one sphere to another by directing a call to the called sphere's gateway. Each process has a stack segment with a protected section which is adjusted on calls and returns between spheres, so that the portion of the stack used by each sphere may

not be examined or modified by any sphere it calls. The supervisor program in our model is implemented in a collection of spheres, to create "firewalls" in it.

## Auditing Responsibility

Networks of computer Utilities are new arenas of action in our society. The forces of law and order deserve a place there, and each user should bear, to a reasonable extent, some responsibility for keeping order.

We model individual privilege and responsibility with a set of abstract legal entities called principals. Any unit of our society which shoulders responsibility, e.g. an individual, a corporation, or an agency of government, can be assigned a principal. Principals are represented by unique character-string names in the Utility.

When a user logs into the Utility, he associates himself with one of the principals and thereby becomes able to exercise the privileges of that principal. His association with the principal, and his identity as established by the login ritual, render him responsible for his actions.

Every sphere of protection, process, and program in the Utility is associated with some principal. The principal associated with a sphere is held responsible for the contents of the sphere, particularly the set of programs there. The principal of a program is responsible for what the program makes a process do. The principal of a process is responsible for the creation and initial direction of the process. Responsibility

for complex events in the Utility can be factored into combinations of these simple elements.

## General Surveillance

We postulate that the supervisor contains a program called the Log Manager which continually writes records on a system log tape. The log's existence is well known and appreciation of it serves to deter wrongful action by users. Tape output is used to insure the integrity of the log in the moments just before a system crash.

Whenever a program in the Utility requests something from the supervisor, a record might be made in the log. The origin of the request can be adequately described by the tree name of the sphere the request came from, and its principal; the tree name of the program making the request, and its principal; the principal associated with the process making the request; and the date and time.

Whenever an access control list is modified, the Log Manager is called, creating a log record containing the tree name of the branch whose ACL is being changed, the old and new states of the ACL, and the origin data of the request. ACL change records serve to record accurately for the Utility administration the access authorizations it has agreed to enforce. This is a precaution which allows the Utility to defend its access control decisions in a court, if need be, by showing who set up the relevant ACL.

Whenever a user identification ritual fails, a record is entered in the log. The Utility administration will examine these records (presumably with a program) to identify threats against passwords, in those cases where the intruder knows most characters and is searching for the rest.

Whenever a request for access to a file or a request to examine or manipulate a directory or sphere gateway fails, a record is entered in the log. The Utility administration will examine these records (with another program), to find threatening patterns. Such patterns include several refused requests for one object, or several refused requests from a particular sphere or program.

Whenever a user logs in or out, a record is made in the log. The login records serve to identify the persons who are associated with principals by the login ritual.

## Privacy Protection

Some uses of statistical or information retrieval programs can result in an invasion of privacy. Suppose a user is accessing a data bank, which contains personal information, through an information retrieval program; and the group of individuals satisfying all the criteria of one of his requests is rather small. In such a case the program should refuse to reveal the number of individuals to the user, and in addition it should call the Log Manager to notify the data bank's owner of the incident.

The owner of the data bank can use this tool to protect the

privacy of the individuals named in his data bank. To do so he would have to investigate the situations brought to his attention by the log records.

## Private Surveillance Mechanisms

A user who owns a particularly sensitive or valuable file can make it a marked file. Whenever a marked file is made a segment in any sphere, the Log Manager is called, creating a log record containing the tree name of the file, the origin data of the request, and the name of the user to whom the record should be delivered. Also if a request for access to a marked file fails, a similar record is written in the log.

The owner of a marked file will be notified by log records if his file is accessed in a way which fails the test of his audit. The owner could implement his audit with a computer program. This mechanism serves to assure owners of marked files that their files are made segments only in proper spheres. If surveillance information is needed for individual fields in a file, it must be gathered by programming in the sphere where the file is accessed. If the owner wants more protection, he can use the Log Manager to back up his private log.

## Police Surveillance

It is likely that some criminal activity will go on within the Utility. A criminal could use the Utility for his everyday data processing, or to spy in some way on another user. As a counterforce to criminal activity, the Utility should include

spying mechanisms to be monopolized by government; e.g. to be used by police agencies with the approval of a competent court. The courts which examine police requests for access to computing objects will have to develop tests to decide which requests are reasonable. The following mechanisms can be made available.

First, users' console sessions can be recorded for later examination or tapped and displayed at another console. Second, undisclosed police inspection of directories and files can be arranged with secret terms on access control lists. Police would use the facilities of the Utility to examine files and console sessions for which they had obtained warrants. Secret terms on ACLs could be created by the Utility administration or the court where the warrant was granted.

Our third mechanism allows users' computations to be saved for inspection after the user logs out. Also, snapshots can be taken of a user's processes' stacks at moments of maximal extension. From these the exact sequence of calls between programs, the arguments passed, and some states of the internal variables, may all be observed. A more complete snapshot mechanism could be used: the snapshot can be taken on every call and return between programs.

This set of mechanisms is offered from a technical point of view: they can be implemented. But from a civil libertarian's point of view, each requires careful study to assess the potential for harmful effects. For example, a console tap, like a wiretap, is something of a "fishing expedition" which could be

an unreasonable search in the Constitutional sense. American law requires search and seizure warrants to name specifically the things being searched for. So any mechanisms made available to police must automatically restrict police in ways determined by the courts' interpretation of Constitutional safeguards.

## A Balance

The courts, and any authority with the power to release information, must protect privacy to a reasonable extent, balancing the need for privacy of individuals and groups against the society's need for disclosure and surveillance. This conflict of values, between privacy and surveillance, is unavoidable.

## Power for the Utility Administration

The Utility administration needs some power to cope with emergencies, such as a process gone wild. They could be given the power forcibly to log in as any principal, with appropriate safeguards. A forced login would require the consent of a committee of the Utility administration, to prevent its use by one man acting alone; and somebody representing the principal being logged into ought to be notified, observe the console session, and be given a copy of it.

Perhaps some other power would more appropriately serve the needs of the Utility administration. But any power given the Utility staff should be circumscribed and balanced against the rights of the Utility's users, and of society.

## Conclusion

Surveillance mechanisms in a computer Utility are intended to curb anti-social computer usage by placing knowlege and power in the hands of information owners, and the state. This is an example of a general phenomenon: the generation and distribution of power (e.g. power over wealth, organizations, or people) by computers. We hope the means can be found to check and balance all this power. As such means are agreed upon and implemented in laws, appropriate enforcement mechanisms, of which surveillance mechanisms are an example, will be programmed into the nation's information systems.

## References

1) The Multiplexed Information and Computing Service: Programmer's Manual. Massachusetts Institute of Technology, 1969, 1970.

2) Rotenberg, Leo J. Privacy and Data Security in a Multi-Access Computer System. In preparation.