Chairman's Introduction to the SJCC Session

Security and privacy in computer systems

by WILLIS H. WARE
The RAND Corporation
Santa Monica, California

INTRODUCTION

Information leakage in a resource-sharing computer system

With the advent of computer systems which share the resources of the configuration among several users or several problems, there is the risk that information from one user (or computer program) will be coupled to another user (or program). In many cases, the information in question will bear a military classification or be sensitive for some reason, and safeguards must be provided to guard against the leakage of information. This session is concerned with accidents or deliberate attempts which divulge computer-resident information to unauthorized parties.

Espionage attempts to obtain military or defense information regularly appear in the news. Computer systems are now widely used in military and defense installations, and deliberate attempts to penetrate such computer systems must be anticipated. There can be no doubt that safeguards must be conceived which will protect the information in such computer systems. There is a corresponding situation in the industrial world. Much business information is company-confidential because it relates to proprietary processes or technology, or to the success, failure, or state-of-health of the company. One can imagine a circumstance in which it would be profitable for one company to mount an industrial espionage attack against the computer system of a competitor. Similarly, one can imagine scenarios in which confidential information on individuals which is kept within a computer is potentially profitable to a party not authorized to have the information. Hence, we can expect that penetrations will be attempted against computer systems which contain non-military information.

This session will not debate the existence of espionage attempts against resource-sharing systems. Rather, it is assumed that the problem exists, at least in principle if not in fact, and our papers will be devoted to discussing technological aspects of the problem and possible approaches to safeguards.

First of all, clarification of terminology is in order. For the military or defense situation, the jargon is well established. We speak of "classified information," "military security," and "secure computer installations." There are rules and regulations governing the use and divulgence of military-classified information, and we need not dwell further on the issue. In the non-military area, terminology is not established. The phrase "industrial security" includes such things as protecting proprietary designs and business information; but it also covers the physical protection of plants and facilities. For our purposes, the term is too broad. In most circles, the problem which will concern us is being called the "privacy problem."

The words "private" and "privacy" are normally associated with an individual in a personal sense, but *Webster's Third New International Dictionary* also provides the following definitions:

Private:... intended for or restricted to the use of a particular person, or group, or class of persons; not freely available to the public

Privacy:... isolation, seclusion, or freedom from unauthorized oversight or observation.

We are talking about restricting information within a computer for the use of a specified group of persons; we do not want the information freely available to the public. We want to isolate the information from unauthorized observation. Hence, the terminology appears appropriate enough, although one might hope that new terms will be found that do not already have strongly established connotations. For our purposes today, "security" and "classified"

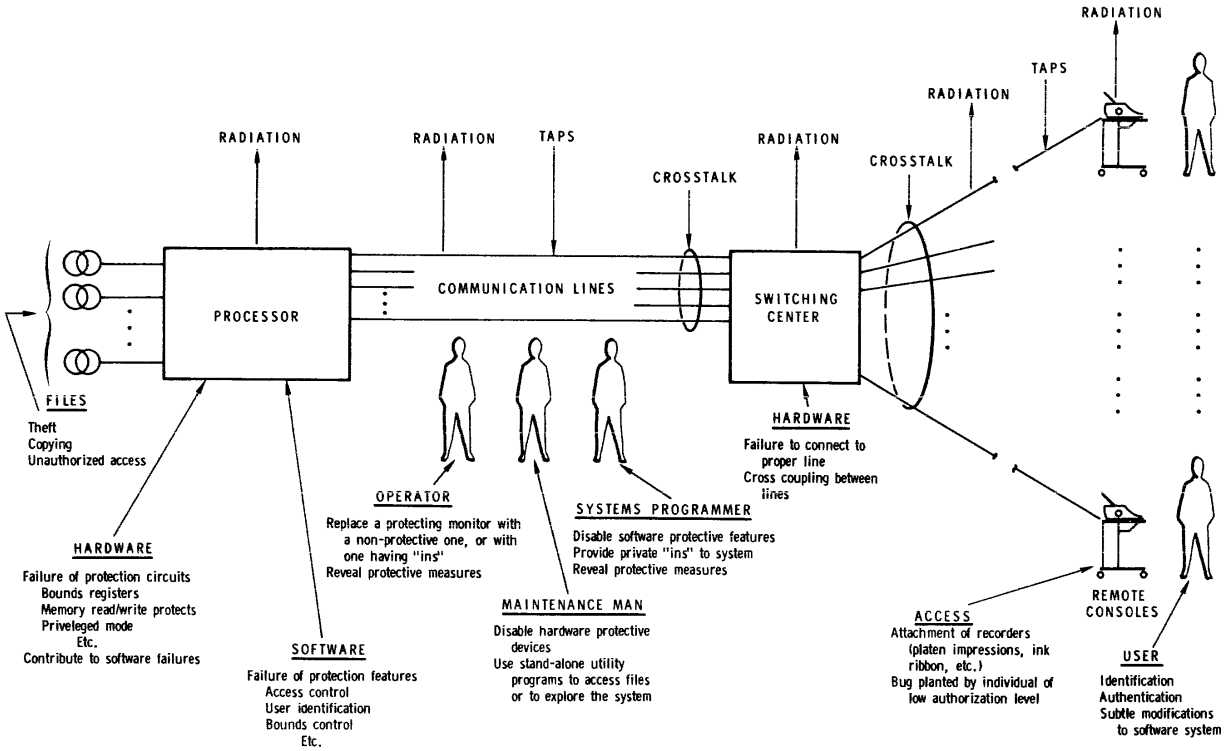

Figure 1 – Typical configuration of resource-sharing computer system

will refer to military or defense information or situations; "private" or "privacy," to the corresponding industrial, or non-military governmental situations. In each case, the individual authorized to receive the information will have "need to know" or "access authorization."

We will do the following in this session. In order to bring all of us to a common level of perspective on resource-sharing computer systems, I will briefly review the configuration of such systems and identify the major vulnerabilities to penetration and to leakage of information. The following paper by Mr. Peters will describe the security safeguards provided for a multi-programmed remote-access computer system. Then I will contrast the security and privacy situations, identifying similarities and differences. The final paper by Dr. Petersen and Dr. Turn will discuss technical aspects of security and privacy safeguards. Finally, we have a panel of three individuals who have faced the privacy problem in real-life systems; each will describe his views toward the problem, and his approach to a solution. In the end, it will fall upon each of you to conceive and implement satisfactory safeguards for the situation which concerns you.

A priori, we cannot be certain how dangerous a given vulnerability might be. Things which are serious for some computer systems may be only a nuisance for others. Let us take the point of view that we will not prejudge the risk associated with a given vulnerability or threat to privacy. Rather, let us try only to suggest some of the ways in which a computer system might divulge information to an unauthorized party in either the security or the privacy situation. We'll leave for discussion in the context of particular installations the question of how much protection we want to provide, what explicit safeguards must be provided, and how serious any particular vulnerability might be.

The hardware configuration of a typical resource-sharing computer system is shown in Figure 1. There is a central processor to which are attached computer-based files and a communication network for linking to remote users via a switching center. We observe first of all that the files may contain information of different levels of sensitivity or military classification; therefore, access to these files by users must be controlled. Improper or unauthorized access to a file can divulge information to the wrong person. Certainly, the file can also be stolen – a rather drastic divulgence of information. On the other hand, an unauthorized copy of a file might be made using the computer itself, and the copy revealed to unauthorized persons.

The central processor has both hardware and software components. So far as hardware is concerned, the circuits for such protections as bound registers, memory read-write protect, or privileged mode might fail and permit information to leak to improper destinations. A large variety of hardware failures might contribute to software failures which, in turn, lead to divulgence. Since the processor consists of high-speed electronic circuits, it can be expected that large quantities of electromagnetic energy will radiate; conceivably an eavesdropping third party might acquire sensitive information. Failure of the software may disable such protection features as access control, user identification, or memory bounds control, leading to improper routing of information.

Intimately involved with the central computer are three types of personnel: operators, programmers, and maintenance engineers. The operator who is responsible for minute-by-minute functioning of the system might reveal information by doing such things as replacing the correct monitor with a non-protecting one of his own, or perhaps with a rigged monitor which has special "ins" for unauthorized parties. Also, he might reveal to unauthorized parties some of the protective measures which are designed into the system. A co-operative effort between a clever programmer and an engineer could "bug" a machine for their own gain in such a sophisticated manner that it might remain unnoticed for an extended period. ("Bug" as just used does not refer to an error in a program, but to some computer equivalent of the famous transmitter in a martini olive.) Bugging of a machine could very easily appear innocent and open.

Operator-less machine systems are practical, and in principle one might conjecture that a machine could be bugged by an apparently casual passerby. There are subtle risks associated with the maintenance process. While attempting to diagnose a system failure, information could easily be generated which would reveal to the maintenance man how the software protections are coded. From that point, it might be easy to rewire the machine so that certain instructions appeared to behave normally, whereas in fact, the protective mechanisms could be bypassed.

While some of the things that I've just proposed require deliberate acts, others could happen by accident.

Thus, so far as the computing central itself is concerned, we have potential vulnerabilities in control of access to files; in radiation from the hardware; in hardware, software, or combined hardware-software failures; and in deliberate acts of penetration or accidental mistakes by the system personnel.

The communication links from the central processor to the switching center, and from the switching center to the remote consoles are similarly vulnerable. Any of the usual wiretapping methods might be employed to steal information from the lines. Since some communications will involve relatively high-frequency signals, electromagnetic radiation might be intercepted by an eavesdropper. Also, crosstalk between communication links might possibly reveal information to unauthorized individuals. Furthermore, the switching central itself might have a radiation or crosstalk vulnerability; it might fail to make the right connection and so link the machine to an incorrect user.

A remote console might also have a radiation vulnerability. Moreover, there is the possibility that recording devices of various kinds might be attached to the console to pirate information. Consideration might have to be given to destroying the ribbon in the printing mechanism, or designing the platen so that impressions could not be read from it.

Finally, there is the user of the system. Since his link to the computer is via a switching center, the central processor must make certain with whom it is conversing. Thus, there must be means for properly identifying the user; and this means must be proof against recording devices, pirating, unauthorized use, etc. Even after a user has satisfactorily established his identity, there remains the problem of verifying his right to have access to certain files, and possibly to certain components of the configuration. There must be a means for authenticating the requests which he will make of the system, and this means must be proof against bugging, recorders, pirating, unauthorized usage, etc. Finally, there is the ingenious user who skillfully invades the software system sufficiently to ascertain its structure, and to make changes which are not apparent to the operators or to the systems programmers, but which give him "ins" to normally unavailable information.

To summarize, there are human vulnerabilities throughout; individual acts can accidentally or deliberately jeopardize the protection of information in a system. Hardware vulnerabilities are shared among the computer, the communications system, and the consoles. There are software vulnerabilities; and vulnerabilities in the system's organization, e.g., access control, user identification and authentication. How serious any one of these might be depends on the sensitivity of the information being handled, the class of users, the operating environment, and certainly on the skill with which the network has been designed. In the most restrictive case, the network might have to be protected against all the types of

invasions which have been suggested plus many readily conceivable.

This discussion, although not an exhaustive consideration of all the ways in which a resource-sharing computer system might be either accidentally or deliberately penetrated for the purposes of unauthorized acquisition of information, has attempted to outline some of the major vulnerabilities which exist in modern computing systems. Succeeding papers in this session will address themselves to a more detailed examination of these vulnerabilities and to a discussion of possible solutions.