CONF-780910--1

LA-UR-78-1660

TITLE: EVENT AND EFFECTIVENESS MODELS FOR SIMULATING COMPUTER SECURITY

AUTHOR(S): Edward P. Schelonka

MASTER

SUBMITTED TO: IEEE Computer Society Conference, Washington, DC, September 5-8, 1978

NOTICE MN ONLY

PORTIONS OF THIS REPORT ARE ILLEGIBLE. It has been reproduced from the best available copy to permit the broadest possible availability.

By acceptance of this article for publication, the publisher recognizes the Government's (license) rights in any copyright and the Government and its authorized representatives have unrestricted right to reproduce in whole or in part said article under any copyright secured by the publisher.

The Los Alamos Scientific Laboratory requests that the publisher identify this article as work performed under the auspices of the USERDA.

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87545

An Affirmative Action/Equal Opportunity Employer

NOTICE
This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Department of Energy, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

Form No. 836
St. No. 2629
1/75

UNITED STATES
ENERGY RESEARCH AND
DEVELOPMENT ADMINISTRATION
CONTRACT W-7405-ENG. 36

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

EVENT AND EFFECTIVENESS MODELS FOR SIMULATING COMPUTER SECURITY

Edward P. Schelonka

Los Alamos Scientific Laboratory
P.O. Box 1663
Los Alamos, New Mexico 87545

This paper describes the development and application of a series of simulation codes designated SECSIM that are used for computer security analysis and design. Individual barrier characteristics are incorporated into generalized architectural reduction algorithms providing numerical indices in selected subcategories and for the system as a whole.

INTRODUCTION

As computer technology has expanded and proliferated evermore intimately into individual everyday life, the leading edge has been followed by a wake of concerns for computer security that have emerged from throughout the government and private sector. These concerns are reinforced by reports of actual cases in which computer security has been breached resulting in damage, theft, and fraud.

With time, computers have become more widely accepted; however, nagging fears persist that computers will fail to perform essential data processing functions whenever required, will contribute to violations of individual or organizational privacy, and will make organizational assets more vulnerable to fraud. The purpose of this paper is to identify the conditions under which these concerns might exist, and to provide systematic design criteria by which improvements may be selected and judged. It should be stated at the onset that the traditional manual and semi-automated methods for implementing data-processing, assuring privacy, and controlling fraud, while more familiar, are inherently less secure than those implemented on a computer. The current tendency is to perceive the computer environment as a means by which new, very powerful and potentially damaging acts can occur. In reality they are dwarfed by losses occurring daily in other less sophisticated situations within the broad security threat spectrum.

Within this computer security simulation designated SECSIM, a decision framework is established to analyze the security risk in terms of the actual or perceived value of what is to be protected. This is followed by examining the threat environment and incorporating countermeasures to reduce the probability of penetration in each threat area. Conceptual designs for potential systems architectures having improved protection mechanisms are then incorporated into SECSIM security models. The modules for analysis and their interrelationships are shown in Fig. 1. From these a security simulation profile is generated that indicates the strengths and weaknesses of the protective systems. Parameters are changed and the process is repeated until the desired requirements are met. An in-depth reliability simulation code (RELSIM) for computer systems and associated network architectures having options for incorporating nonidentical redundancy generates projections of reliability performance. Availability and down time per year are computed at each functional level and for the sytem as a whole. The models in this study that characterize the security behavior of candidate systems are comprehensive in that they accommodate a wide range of threats and protection levels. They are tailored to each facility by weighting input parameters consistent with the design configuration and environment selected. This method of analysis is applicable to computer centers in the conceptual design, detailed design, construction, and operational stages of evolution. Although most security criteria are common to all facilities, individual specific facility vulnerability data are purposely suppressed.

These simulation techniques provide top management, security officers, system operators, system programmers, and other center staff with a set of evaluation criteria with which to assess their facility for its security vulnerability. They also provide a basis for determining the

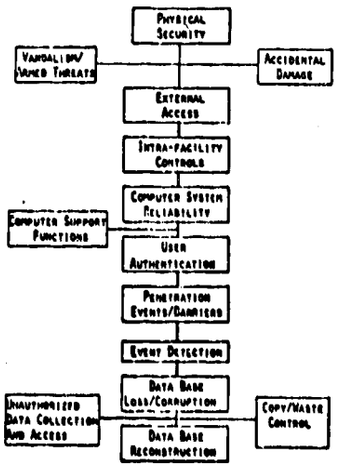


Fig. 1. COMPUTER SECURITY SIMULATION AREAS

effectiveness of design changes for upgrading and improvement. Finally, interfacility comparisons can be made using a common set of references.

RISK ANALYSIS

A thorough risk analysis is the fundamental first step in developing an organizational computer security policy. It requires joint upper management and center staff action. The analysis must clearly identify what is to be protected and its relative value. Primary areas of consideration are: equipment destruction or theft, lack of computer processing capacity, loss of data files, and misuse of information.

Costs for hardware replacement include debugging the system after installation and leasing interim facilities. The cost to the organization for the loss of data processing capability is dependent upon the duration of the loss and its position within the sequence of processing tasks. Initially, all low priority jobs are set aside. Extended loss can disrupt computerized record maintenance. Other automated internal management functions will also be inhibited.

In a similar manner the replacement cost and value of data files to organizational operation must be established, and an analysis should be made of the potential damage caused by release to a competitive firm. Personal and proprietary privacy and other sensitivity attributes must be carefully evaluated in terms of some reasonable monetary bounds. Both legal and outside authoritative consultant assistance will improve objectivity. Top management must be included in these determinations since proprietary data is close to the core of organizational goals. Possible losses due to fraud involving insiders, outsiders, or both must then be determined. A primary measure of the maximum monetary value is derived from the ability of the machine to control assets such as equipment and funds.

The next requirement is to estimate the probability of occurrence of each loss. This necessitates using all possible prior information, including history, experience, and trends. For the case of hardware failures, reliability data are obtained from the manufacturer. Somewhat less precise but more meaningful are estimates of the computer system reliability as a whole, including software, peripherals, and terminals. Figure 2 shows the dependence of availability and corresponding down time per year as a function of the mean time to repair (MTTR) with the mean time between failures (MTBF) as a range of input parameters fixed by equipment choice. Estimates of the probability of equipment damage, theft, data loss, and fraud are obtained from history, experience, national averages, and insurance authorities.

Finally, the magnitudes and probabilities are combined as products to give weighted losses in dollars per year for each category. This type of data provides basic guidelines for allocating the amount of money and effort to be spent in each respective area. Several cycles of analysis involving proposed designs, simulations, and evaluations are normally required to converge to an acceptably low level of risk.

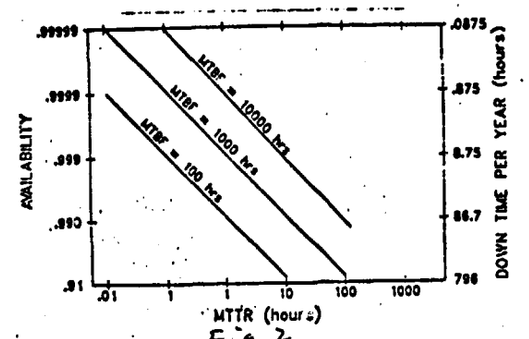


Fig 2

THE THREAT ENVIRONMENT

Threat here is defined as any act or omission that will enhance the likelihood of or allow for losses to occur. The more pertinent areas are mentioned here. Subsequently, a summary of the threat versus countermeasure encounters is provided.

Human actions, both those of the intruder and the center employee are subject to error (at levels established by human reliability studies). This factor adds to the unpredictability of their behavior. Between 1 and 9 per 1000 actions involving simple tasks will be done incorrectly. Actions on the part of center staff could take the form of incorrect cards input, tapes loaded, switches set, command keys depressed and data input. These errors include effects ranging from marginal operation to complete operating systems erase, and the effects can propagate into the data base and data store.[1]

The account number-password system entry is penetrable.[2] One method used is that of replacing the user with a small computer and attempting to LOGIN with randomly selected characters. Experience shows that this requires approximately one week of effort.[3] Another method uses a small computer inserted into the line between the terminal and main computer. The small machine is programmed to respond to a LOGIN; however, it covertly records the account number and password. A somewhat simpler approach involves collecting the LOGIN signals through line taps, line electromagnetic coupling, or by emanations from the terminal itself. Private data can also be compromised if the individual at the terminal does not execute a complete LOGOUT.

Data being processed can be read out remotely as well. This is done by collecting radiated energy from the main frame and peripherals, particularly printers. A remote collection system with a minicomputer can interpret the bit streams and reveal not only the data being processed but also the characteristics of the operating systems software.

A substantial number of computer security breaches have occurred because of inadvertent data releases. These may occur when several incremental releases are brought together and under scrutiny reveal private information, whereas each release taken separately would not.

These frequently reflect a lack of aggressive reviews by the security officer and management. Other methods are associated with waste in the form of cards, printout, carbons, and printer ribbon. Even more potentially damaging is the casual discard of systems manuals. Private information can also reach unauthorized persons by accessing memory, storage registers, disk space, and tapes that have not been cleared or purged prior to user access.

Breaks in normal operating patterns, such as those that exist during maintenance, provide additional opportunities for penetration. Intruders masquerading as repairmen can be given wide ranges of freedom and access well beyond those of normal operation. Disastrous immediate or latent effects can result.

Allowing broad single individual responsibilities or job latitude encompassing the functions of systems programming, application programming, machine operator, and data store custodian point toward potential threats to security. The problem is further compounded by an open policy allowing for off-shift work. This environment is characteristic of those that have spawned past computer-assisted frauds.[4]

SECURITY COUNTERMEASURES

Certifiably secure operating systems do not exist at the present time; however, there are a number of methods for improving existing systems. These are given in Table I.

An inaccurate or corrupt data base can be more damaging than simply loss of the data. Errors can propagate unchecked through successive processing stages before they are identified. The slightest amount of false data entered in a data-bank system, whether by accident or by intent, can make much of the systems operation worthless.[5] A fundamental step required is input data editing. A substantial effort invested in a comprehensive set of reasonableness checks will provide direct return in filtering input errors. These errors originate in manually introduced data from cards and terminals and from malfunctioning interfaced instruments. Also, it should be verified that security controls are uniformly consistent from initial entry, edit-to-correct-errors entry, and longer-term data-base correction or update.

Classical auditing methods are revised extensively when applied to data processing systems. Auditors must be aware of how computers are integrated into the organizational structure and its information system. It is further necessary to evaluate the machines in terms of their ability to run standard auditing programs brought in by the auditor to determine the integrity of the internal accounting system. Initially, benchmark programs are run to verify that the machine is functioning normally. Raw and processed facility data are then scanned and checked for outliers and unusual trends. After determining the size of the files to be audited, subroutines provide estimates of the number of samples to be taken for a given confidence level and actually choose the samples in a random fashion. Full manual verification is then used to insure that all steps in establishing that segment of the data base were correct. This increases the level of assurance that internal audit controls are in effect.

The checkpoint/restart routine provides dumps of program status data and intermediate results so that in the event of a systems crash the program can be reinitiated at the last checkpoint. This is contrasted with a complete restart in which files in edit may be lost. Sensitive data in the checkpoint dump can be altered or used in an unauthorized manner upon reinitiation.

Tables II through IX are examples of structured sets of threats and counterforce actions for use at the design, evaluation, and operating levels.

TABLE I

PENETRABLE OPERATING SYSTEM

Threat	Countermeasures
Obtain unauthorized privileges from a penetrable operating system	*Design and implement a provably secure operating system
	*System security improvement techniques
	Use a virtual machine architecture
	Place a reference monitor and security kernel in separate machine spaces
	Use control lists for user resource verification
	Prevent user access to supervisory program space, including program status words and halting instructions
	Control user access to input/output and direct memory access
	Conduct alternate as well as primary logical path analysis with complete parameter checking and caller access
	In hardware, monitor commands and calls for each use transaction in batch or interactive modes
	Implement a watchdog timer to generate an interrupt in the event that the executive program is changed
	Minimize the number of hardware trapdoors by keeping the number of special switches for computer mode changes as small as possible and under control at all times
	Use read-only memory for the system executive program
	Eliminate software trapdoors that allow system programmer access to generally forbidden computer resources during debugging and checkout
	Use hardware memory lockout to prevent alteration of core memory outside of assigned program areas
	Use a hardware bootstrap to rapidly reload the operating system and key programs in the event of a crash
	Protect the logical section of the memory via base and bounds registers
	Secure the checkpoint/restart routine
	Do parity checking in main core store
	Erase memory after each run and eliminate a core dump possibility by the next program user
	Eliminate user access to the active job queue

SECURITY ANALYSIS

Barrier evaluation is similar to security evaluation in general in that subjective as well as objective judgments must be made to determine effectiveness. The range of 0 to 1 is chosen, where 1 represents a totally effective barrier. The individual facility threat environmental factors are incorporated as weighted inputs along with the corresponding attributes of each barrier, and a numerical value or range of values is assigned. These are then introduced into the appropriate segment of the model and a composite effectiveness is computed for that hierarchical sublevel. This is repeated for the next layer until an index is generated for that respective category. A two-dimensional horizontally layered and vertically structured methodology is used. Also event/effectiveness diagrams aid in establishing the interrelationships between computer

TABLE II

FRAUD CONTROL

Threat	Countermeasures
Opportunities for fraud	*Screen background of personnel before hiring and regularly during employment Alcoholism/drugs Debts Family problems/divorce Overall mental attitude *Limit the following job functions to specific categories of individuals with no overlapping responsibility or access to assets Machine operator System program development System program testing Application programming Data/tape librarian *Write programs in higher-order languages if possible *Have systems programmers work in pairs *Conduct computer security training and stimulate awareness *Provide incentives to staff for finding security weaknesses *Require two man enforcement of operating system changes and supervisory approval *No off-shift work or open access to the machine *Rotate job assignments Programmers change application areas Operators change shifts *Two operators on duty at all times *Provide insurance coverage and bonding for all personnel.

TABLE III

DATA COLLECTION AND INTERPRETATION

Threat	Countermeasures
Unauthorized data collection and interpretation	*Physically secure wireline with sealed cabinets, junction boxes, and conduit *Tap detection methods, including time domain reflectometry *Dual channel fiber optics links *Encryption *Electromagnetically shielded (including near field effects) wire lines *Increase monitoring distance with physical barriers to minimize detection probability *Mask signals with electromagnetic noise generators *Use "TEMPEST" enclosures and overall room shielding *Random character stuffing

TABLE IV

WASTE CONTROL

Threat	Countermeasures
Information compromise due to inadequate waste control	*Destroy paper, waste-shredder Operator console User terminals System programs Application programs Cards Ribbons Carbons Revised operating manuals *Electronic storage Registers cleared Allocated memory cleared Disk overwrite Tape degauss

TABLE V

USER AUTHENTICATION

Threat	Countermeasures
Inadequate user authentication	*Mechanical and electronic keys *Personal characteristic verification *Credential authorization *LOGIN procedures Limit number of LOGINS per unit time Limit length of time to LOGIN Allow user to establish next LOGIN time window *Password - account number check Control characters in password Algorithmic passwords Change passwords for each LOGIN Passwords previously sent to user

TABLE VI

RECONSTRUCT DATA BASE

Threat	Countermeasures
Inability to reconstruct lost data base	*Loss detection *On site storage of first, second, and third generation transaction tapes *On site machine availability with sufficient capacity *Off-site transaction tape storage *Alternate processing center Trained staff Transaction records on hand Current software Operating system documentation Application programs machine capacity

TABLE VII

CENTER DAMAGE CONTROL

Threat	Countermeasures
Damage and disable machine and support functions	*Intruder access denial Foreign-object detector Personal characteristics check Credential check Mechanical barrier enable *Intruder recognition Escort Badges *Intrafacility controls Partitions Vault Guard check points *Vandalism protection Center location in building middle floors Entry barricades Interior barricades Siting in benign neighborhood *Protection from accidental damage Manmade Natural *Support function protection Primary and backup power Primary and backup air conditioning

TABLE VIII

RELIABILITY

Threat	Countermeasures
Computer system unavailability due to unreliability	*High-reliability hardware *Software verification and validation *Incorporate redundant architectures *Immediate failure detection *System diagnostic programs *Aggressive repair program

TABLE IX

EVENT DETECTION

Threat	Countermeasures
Penetration event is undetected	*Maintain an audit trail showing the source, transaction, privileges, and exceptions to privileges *Auditors are trained in data processing techniques *Allow for unscheduled internal and external audits including sampling and end-item verification *Sequential numbering of each file access *Date of last access given *Observe file changes *Observe abnormal resources consumed *Log all security authorizations *Monitor jobs in progress *Run code comparisons and benchmark calibration programs *Record all LOGINS

security factors. Each is accompanied by a list of computed input and output effectiveness ratios.

Physical security is of primary concern in protecting the center and its assets. The overall contributions to effectiveness are shown in Fig. 3 with the results of the sublevel and summary computation listed Table X.

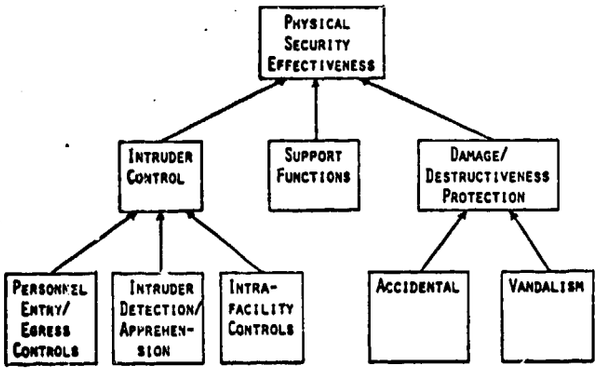
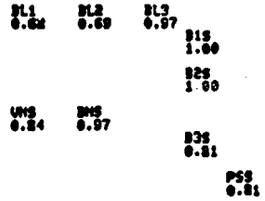


Fig. 3. PHYSICAL SECURITY COMPOSITE

TABLE X

PHYSICAL SECURITY SIMULATION OUTPUT



BL1=PHYSICAL ENTRY EGRESS CONTROLS
BL2=INTRUDER DETECT,APPREHEND
BL3=INTRAFACILITY CONTROLS
B1S=INTRUDER CONTROLS
B2S=SUPPORT FUNCTIONS
UNS=VANDALISM SUMMARY
DNS=DAMAGE-NATURAL CAUSES
B3S,DSP=SUMMARY,IMMUNITY FROM DAMAGE/DESTRUCTION
PSS=PHYSICAL SECURITY SUMMARY

The next major layer of protection relates to the computer system itself. The user authentication techniques are shown in Fig. 4. With the broad range of methods available in this simulation, the model computations (Table XI) show that it is essentially impossible to defeat these authentication barriers in a normal environment. Thus it is necessary for an intruder to obtain other means such as inside assistance to penetrate at that level. The criterion established in this simulation model is that if at least one of three authentication controls are successful

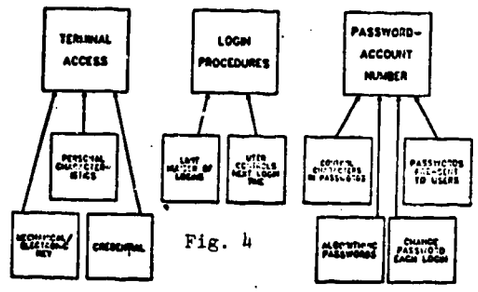
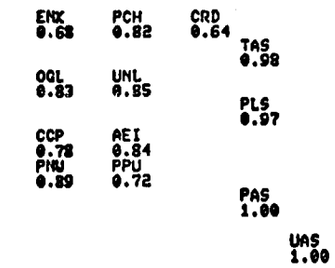


Fig. 4

TABLE XI

USER AUTHENTICATION SIMULATION OUTPUT



EMK=ELECTRONIC/MECHANICAL KEY
PCH=PERSONAL CHARACTERISTICS
CRD=CREDENTIAL
TAS=TERMINAL ACCESS SUMMARY
OGL=LIMIT NUMBER OF LOGINS
UNL=USER DETERMINES NEXT LOGIN TIME
PLS=LOGIN PROCEDURE SUMMARY
CCP=CONTROL CHARACTERS IN PASSWORDS
AEI=ALGORITHMIC PASSWORDS
PNU=NEW PASSWORD FOR EACH LOGIN
PPU=PASSWORDS PRE-SENT TO USERS
PAS=SUMMARY PASSWORD/ACCOUNT NUMBER
UAS=USER AUTHENTICATION SUMMARY

the intruder is denied permission for further access. In this instance the effectiveness was so high that the summary was rounded off to 1.0. Other simulations are done in a like manner. A penetration event detection diagram is shown in Fig. 5. If detection is pursued to the level of dedication shown, the simulation results of Table XII predict that detection will be virtually certain.

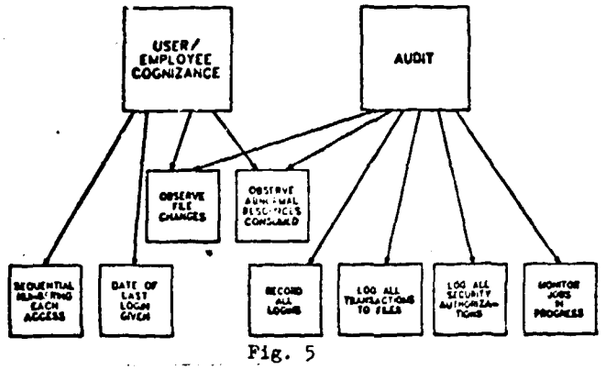


Fig. 5

TABLE XII

PENETRATION DETECTION SIMULATION OUTPUT

SNA 0.62	DLL 0.65					
OFC 0.55	RCD 0.49					
			ECS 0.97			
RAL 0.41	TFL 0.53	SML 0.56				
			RLS 0.88			
CHP 0.72	DFI 0.69	PMV 0.78				
			AUS 1.00			
				PDS 1.00		

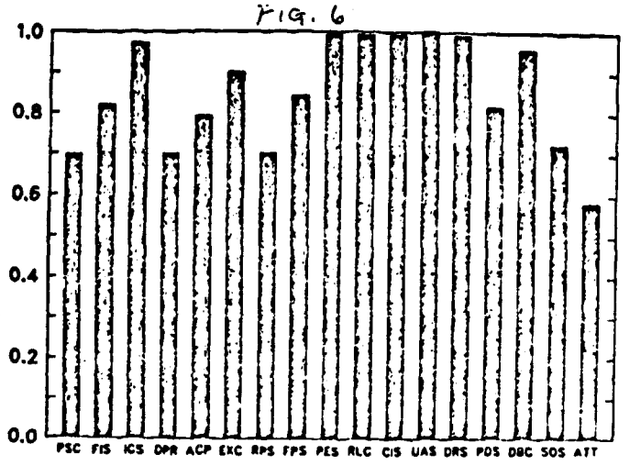
SNA=SEQUENTIAL NUMBERING OF EACH ACCESS
DLL=DATE OF LAST LOGIN
OFC=OBSERVE FILE CHANGES
RCD UNUSUAL QUANTITIES OF RESOURCES CONSUMED
ECS=EMPLOYEE/USER COGNIZANCE SUMMARY
RAL=RECORD ALL LOGIN ATTEMPTS
TFL=LOG ALL TRANSACTIONS TO FILES
SML=LOG ALL MACHINE SECURITY AUTHORIZATIONS
RLS=LOGGING SUMMARY
CHP=CODE CONTAMINATION
DFI=DATA FILE INTEGRITY
MJP=MONITOR JOBS IN PROGRESS
AUS=AUDIT SUMMARY
PDS=PENETRATION DETECTION SUMMARY

This process is repeated for data base loss/corruption, unauthorized data collection and interpretation, copy/waste control, and data base loss/reconstruction. It should be pointed out that to illustrate the latitude of the simulation, a broad range of threat protection countermeasures are incorporated yielding a relatively high security effectiveness. A typical facility will be much more limited in its approach toward security.

A security profile is then generated based on each of the effectiveness indices and displayed in Fig. 6. It indicates the relative strengths and weaknesses of that facility configuration in its threat environment.

The algorithms in the models contain branch points allowing for redundant barriers with generally dissimilar characteristics. A range of conditions can be imposed from at least 1 of N to N of N, or exactly M of N, surviving. A sensi-

Fig. 6



tivity analysis is conducted showing the changes in system effectiveness with changes in individual barrier characteristics. The simulation is capable of being run in real time to give current estimates of security effectiveness as the threat environment and individual barrier effectiveness change.

SYSTEM DESIGN IMPACT

The scope of security threats, countermeasures, and barrier attributes, as well as effectiveness modeling, given in previous sections forms the basis for system design and implementation. Detailed system design and engineering dictate intensive reviews to match hard system requirements with a real-world environment. With respect to security, layered simulations are repeated for the generalized candidate system architectures considered. As the architecture is further defined, vertical event simulations are then conducted as shown in Fig. 7 with the results tabulated in Table XIII.

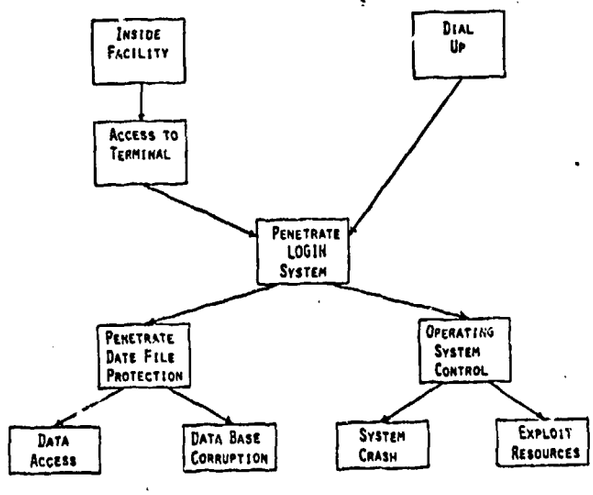


Fig. 7. Machine security penetration events.

TABLE XIII

PENETRATION EVENT SEQUENCE SIMULATION OUTPUT

FIS 0.76	ATT 0.36	FTA 0.85			
	DUP 0.81	B12 0.81			
			BL1 0.69		
	PLS 0.81	B21 0.81			
			BL2 0.81		
	PFP 0.86	B31 0.86			
	OSC 0.74	B32 0.74			
			BL3 0.64		
	DAC 0.73	B41 0.73			
	DBC 0.68	B42 0.68			
	SVC 0.59	B43 0.59			
	EXR 0.81	B44 0.81			
			BL4 0.24		
				PES 0.98	

Penetration paths with probabilities of intruder success are calculated to assist in determining a finer structure of conditions under which penetrations can be made. Appropriate countermeasures are also incorporated and the process is repeated.

Irrational behavior is not predictable and as such may not be included in the listing of probable penetration strategies. Within reasonable bounds the computer can simulate all possible random combinations of events leading to a penetration objective and rank them according to the opportunities for success. With very large systems, exhaustive testing is not possible; however, representative samples are taken and analyzed for unusual paths.

Security design actions to cope with these threats frequently represent significant investments in hardware, software, and manpower. This cost overlay is matched with countermeasures effectiveness as incremental changes are made through SECSIM. The previously described monetary risk evaluation provides guidance for cost bounds. Another overlay employed at this stage of development is that of reliability. Architectural adjustments are accomplished as required to make system performance fit within reliability and availability limits. Finally an operational overlay is used to determine whether the above constraints limit the primary function of the center in terms of providing computational results. This involves task and time-and-motion studies to identify problem areas that may serve to rule out constraints in subsequent implementation. Modeling provides methods for adjusting security effectiveness to compensate for operational requirements where compromises are indeed possible.

In all of the discussions thus far it has been implicit that security be incorporated into the facility design as early as possible. In this manner costly redesign, retrofits, and rework can be avoided. It is clear, however, that security threats are not static so that some design review and system revisions will have to be made. Simulation provides the methodology for making upgrading decisions.

CONCLUSIONS

A set of automated computer security analysis tools has been designed and presented here with samples of results obtained. These tools provide systematic methods for examining known aspects of computer security and assessing the effectiveness of countermeasures. The input data are incorporated into models that generate effectiveness indices in individual areas and an overall profile. When the security attribute methods given here are adapted to other facilities in a uniform manner, a measure of relative effectiveness is obtained. The resultant facility indices should be considered as extremely sensitive information because they represent the very essence of the facility vulnerability. The information would be particularly useful to a potential intruder in a facility with security holes or weak areas. The input data used in these simulations are typical of the ranges of values observed in practice; however, they have been used in a manner that does not reveal vulnerability of a specific facility.

SECSIM programs are capable of running in real time to give current estimates of the system effectiveness in a dynamic threat/countermeasure environment. As individual protection levels change, perhaps due to failure or other circumstances SECSIM modules can be reconfigured and rerun to generate an updated security profile. The model can also predict very quickly the relative merits of proposals to reestablish or reconfigure protective barriers. This type of information is useful to management for critical decisions. The SECSIM simulation program modules are written in FORTRAN and as such are portable in being able to run on most large main frame machines and minicomputers. They do not depend upon the existence of auxiliary simulation codes.

In the realm of further work, research is continuing on expanding the range of computer security barrier effectiveness attributes and improving the set of reference levels from which numberical inputs to the modules are generated. SECSIM is modular and flexible in being able to accommodate new threat areas that might emerge. Another promising area of future research is in the application of fuzzy sets to the quantification and calculation of barrier effectiveness. This type of mathematics lends itself well to evaluation with human perceptual and subjective, as well as objective measures.

REFERENCES

1. R. L. Thomas and R. H. Courtney, "A Systematic Approach to Data Security," Proceedings of the Privacy and Security Conference, Gaithersburg, Maryland, March 4-5, 1974.
2. National Bureau of Standards, "The Use of Passwords for Controlled Access to Computer Resources, "NBS Special Publication 500-9, May, 1977.
3. W. G. Frickel, D. A. Webb, J. S. Chin, S. S. Lai, K. I. Azuma, C. L. Streeter, and S. Tokubo, "RISOS Analytic Tool Description Manual, Part 1: Program Description" Lawrence

Livermore Laboratory UCRL-51810 Pt. 1, May 19, 1975.
4. H. Weiss, "Computer Security: An Overview, "Datamation, Vol. 20, p. 1, January 1974.
5. H. Feistel, "Cryptography and Computer Privacy," Scientific American, May 1973, p. 15.