BQ,0

TO: R. C. Daley
F. J. Corbato
J. H. Saltzer
T. VanVleck
J. Grochow
E. Stone
J. Gintell
C. Clingen
A. Evans

FROM: M. J. Spier

DATE: December 8, 1969

SUBJECT: System Control Documentation

Attached is the latest draft of the System Control overviews we intend to publish as soon as possible. Your comments are solicited.

1. Good
2. Too much detail of commands
3. Too much on how it now is - tell it like it will be.
4. I think per-shift maxima are very bad.
5- How does a proj. admin. create a non-registered user? Must sys. admin be told?
6. Who can change passwords?

0. Very good
1. How do I change my password?
2. Last two sections are mostly repetition of earlier material - cross reference or overview would be better.
3. More comments "inside"
4. How Art E. Evans seen it?

BQD

DRAFT
12.5.69

Identification

Overview of System Control

Michael J. Spier

Purpose

This overview presents the general design framework of System Control, i.e., definitions, design objectives, policies, etc. It attempts to remain as independent as possible of the current implementation.

Definitions

This section defines the terms most commonly used in the area of system control. It is important that such terms as "person," "user," "project," "account," etc. be properly defined, in order to achieve an appropriately modular design.

A person is a human being, known to System Control by his unique person-ID. A person is also associated with at least one secret password known only to himself and to System Control. The combination person-ID/password uniquely and positively identifies a person.

A project, known by its unique project-ID, is a vehicle for grouping persons working towards a common goal and (perhaps) sharing common data. The File System's access control mechanism is designed to respect the notion of project, so that access to the file system may be granted by person, by project, or by any combination thereof. We name the association of person/project user, and identify it with a unique user-ID which,

by agreement, is derived from the corresponding person-ID and project-ID. A person may assume several user identities, i.e., be associated with several projects. By virtue of the project's special purpose (goal), its associated users may also share common attributes, such as special-purpose sub-systems, special privileges, party-group line, etc. A user is also associated with one or more accounts, each known by its unique account-ID. An account specifies the amount of credit that a user (or group of users) have with the system. A user may use the system only as long as he is "solvent," i.e., as long as the account on which he currently draws is not depleted. One or more accounts may be grouped into an account group, known by its unique account-group-ID.

We name the period of time during which a user is using the system user session; it corresponds to the period of time during which there exists at least one user process (i.e., a process dedicated to the user's computation). A user session may, in fact, comprise several user processes, which may exist serially and/or concurrently. A user process enjoys the access privileges associated with the user's project. It may, however, be desired sometimes to restrict the access privileges of a user process beyond the restrictions imposed upon the associated project. This is accomplished by associating each user process with the concept of process group, which defines lesser "degrees of trust" within the domain of a single project. Every process group, consisting of one or more user processes which belong to a single user, is associated with a unique process-group-ID which, by agreement, is derived from the appropriate user-ID. Every Multics process is associated with a unique process-ID which is maintained by, and meaningful to, the Traffic Controller (see MSPM sections BJ). The process-group-ID is the most elementary criterion by which access to the File System may be granted, i.e., access may not be defined by a value that is "finer" than the process-group-ID.

A user session may be viewed as a "job," consisting of one or more "tasks" which are normally executed one at a time; control information to specify the "task" (and perhaps data to be manipulated by it) is inputted through the user session's input stream. Results may be outputted

through the session's output stream. An input/output (I/O) stream is an abstraction; it is an idealized I/O device providing standard read/write interfaces for the user regardless of the actual device with which his process is currently associated. An I/O stream may thus be attached to a large variety of devices, ranging from processes to standard I/O devices to files. A user session whose I/O control streams are attached to a console (typewriter, teletype, display-tube) is known as interactive computation, i.e., a computation which is capable of interacting with a human being. If its I/O streams are attached to any other device, the user session is said to be an absentee computation.

Introduction

The main purpose of System Control is to protect the system's integrity by a) positively identifying any user who wishes to be granted access to the system, and b) the ability to preempt any user and/or reclaim any system resources that are currently available to a user. To this end, System Control maintains databases which contain all the necessary information to a) identify and describe every one of its logical components (person, user, project account etc.), and b) to describe the current configuration of system resource allocations and usage.

There are five major databases to define System Control's logical components:

1) The Person Name Table (PNT), which is a single per-system table with an entry per person-identity.
2) The System Administrator's Table (SAT) which is a single per-system table with an entry per project.
3) The Master Account Table (MAT) which is a single per-system table with an entry per account-group.
4) The Project Definition Table (PDT) which is a table per project with an entry per user.
5) The Account Group Table (AGT) which is a table per account group with an entry per account.

All the above data bases contain static information (excepting a single item in the SAT) and reflect the respective administrators' definition of

the system. Any of those tables may be updated on-line by its respective administrator without either requiring any system pause or interfering with any currently-active user session.

There are four major databases to describe the current configuration of system resource allocation:

1) The User Overseer Table (UOT) which is a single per-system table containing an entry per active user-session.
2) The Answer Table which is a single per-system table dedicated to the management of all consoles and which contains an entry per known console. [Note: In the initial implementation, due to the fact that all user-sessions are interactive, and for historical reasons of implementation, both above tables are integrated into a single Answer Table which features the UOT information in the entries corresponding to "dialed-up" consoles]
3) The Absentee Computation Table (ACT) which is a single per-system table with an entry per known absentee computation (not in initial implementation).
4) The Account Usage Table (AUT) which is a table per account group and contains an entry per account.

Also, there is the System Log, a table containing an entry per user session and identifying the user, his login time, and the resources used during his session.

Associated with the above databases are dedicated system functions to maintain and consult those databases:

1) System Control, to maintain the PNT, SAT and MAT.
2) User Control to maintain PDT in conjunction with PNT and SAT.
3) Accounting Control to maintain the AGT and AUT.
4) Login Control to consult the PNT, SAT, MAT, PDT and AGT.
5) User Overseer to maintain UOT, Answer Table and ACT

System Control

The system administrator has complete authority over the system and its resources. He has the ability to delegate such authority to projects and account group administrators which in turn have authority over users. The system administrator's identity (or at least one of his identities) must be "wired" into the system, i.e., the very initial system administrator identity may not be established dynamically, on line. Further system administrator identities (we talk of the system administrator, in fact that position may be held, for administrative, clerical or other reasons, by more than a single person) as well as all project administrator and user identities may then be dynamically defined, using the tools provided by the system- and user-control modules.

The system administrator defines projects, account groups and registered person identities.

User Control

Access to Multics depends upon a person's association with a pre-defined quality, namely his project; it is this association which establishes his user identity, and hence his range of access privileges in the system. It is the responsibility of User Control to maintain all the information that is necessary in order to enable the system to positively identify a user. User Control and its data bases are protected from the normal user; only a high privilege user, the project administrator, has the ability to modify User Control's data bases, and that only within his own project and subject to certain restrictions.

The project administrator has the ability to add users to his project. Normally, such users are persons which are already known to the system ("registered persons"). However, with the system administrator's approval, the project administrator may also have the privilege of establishing new (and temporary) person identities, so as to allow non-registered persons to become users of his project. Also, a project administrator may choose to admit (at his own cost and risk) any person as a valid user of his project, by waiving his right for the system-provided service of positive person identification.

Users of such a project are said to have "unauthenticated person identities". Also, subject to the system administrator's approval, a project administrator may confer upon his users certain privileges, such as membership in a party line group (assuring a certain degree of guaranteed access to the system), immunity to preemption (i.e., to autmmatic logouts) etc.

Accounting Control

It is the responsibility of Accounting Control to charge all system usage against some appropriate account. An account is an entry in our Account Group Table, consisting of an amount of credit (dollars) that may be spent through system usage. Any number of users may draw on a single account; also, a user may have the option of drawing on one of several accounts. The user/account(s) association is established by the account administrator and maintained by User Control. A user session may progress only as long as the associated account is not depleted. Account depletion ("out of funds") is one of the conditions that the User Overseer recognizes to imply user session termination. Accounting Control's responsibility is to maintain the account data and to maintain the appropriate account-ID/user-ID cross-reference control information.

Login Control

A user identifies himself to the system by typing

login-word login-name [project] [account]

where the login-word is a character string to identify the type of session which the user wishes to initiate. By agreement, the login-word for the normal user session is "login"(or "Login"). The login-name is the person's unique person-id which is derived from his real life name, and is registered in the PNT by the system administrator. In order to gain access to the system, a person must have at least a single project-associated user identity.

Every person is associated with a default project, and hence every person has a default user identity; if he chooses to assume a different user identity, he must explicitly specify the corresponding project name. Also, a user is associated with a default account that may be overridden by an explicit declaration.

The user further validates his personal identity by typing in his secret password. Login control goes through the following steps:

a) Looks up the Person Name Table (PNT) for a valid person-id/password match. Inability to locate a match causes the login request to be terminated.

b) Finds the Project Definition Table (PDT) corresponding to the user's (explicit or default) project and finds the user's entry in that table. Login attempt terminated if project unknown or if user unknown

c) Finds the Account Group Table (AGT), and the specific entry in it, associated with the user's (explicit or default) account, and validates the user's right to draw on that account. Login attempt terminated if account is inaccessible.

d) Looks up the project's entry in the System Administrator's Table (SAT). If the user is member of a party group line then if this project has such a ("primary") line available, and
there are available lines in the system or somebody could be preempted, then the login attempt is successful. Otherwise, if there is an available line, then the login attempt is successful, or else the login attempt is terminated.

e) Creates a process which comes to life in an initial procedure specified by the user's PDT entry. The Process draws on the account validated in step c. Appropriate entries are made in the User Overseer Table (UOT) to describe the user's console session.

The user preemption rule is that a candidate for a primary line may preempt the oldest standby user; a candidate for a standby line has no preemption privileges. A user who fails to complete his login sequence within 2 minutes, or an interactive user who remains blocked for more than 60 minutes, is automatically logged out.

The User Overseer

The System Control module responsible for the initiation (login) and the termination (logout) of a user session is named User Overseer. In order to fulfill this function, it must be receptive and responsive to all conditions which imply the need to either login (e.g., dialup) or logout (e.g., hangup) a user. The fact that it initiates user sessions implies that it must have the authority to create a user process. More importantly, its being in charge of console session termination implies that it must have the power to force a user process to a halt, and save or destroy it. As the example of a possible "runaway" process (a process that does not respond to normal signals) may suggest, the User Overseer must indeed possess powerful tools to enable it to apply corrective measures in cases where a process may "get out of hand." Moreover, in order to insure the system's integrity, the User Overseer (and indeed the entire System Control process) must be protected from all damages that may be inflicted upon it by a user process.

The User Overseer is also in charge of assigning system resources (e.g., I/O devices) to a user process. For the same reasons as mentioned above, it must be able to enforce seizure of such resources if it wishes to reclaim them. A typical example to illustrate this is the case where an interactive user process (one that interacts with the user over a dedicated console) gets out of hand and must forcibly be terminated. If the User Overseer were to rely on the user process' voluntarily returning the console, then that console would, in this specific case, be irretrievable lost to the system.

As already hinted above, Multics does not restrict the user to a single process per user session. The user is able to swap his current "used" process for a new one, or may engage in a user session consisting of several coexisting and inter-communicating processes. Evidently, the user's process must not be allowed to do any process creation on its own. The user overseer is capable of receiving control communications from the user process; if such a communication implies the user process' wish to have a process created, and if it has the right to request such services, then the User

why not?

Overseer creats a process in its behalf. This feature enables the User Overseer to be always aware of the relationship and possible dependency among processes. Thus, the termination of a user process always implies the termination of all of its concurrently-existing descendents. This is achieved by associating with each user process a list of all of its currently existing descendents. (currently, and in the near future, a user will not be able to possess more than a single process at a time).

The following sections describe the mechanics of system administration (conventions, tools, implementation, usage, etc.) in the following five general areas:

1) System Administration - persons
2) System Administration - projects
3) Project Administration
4) Master Account Administration
5) Account Administration