Software Product Description

Extended Access Control Facility (EACF)

Executive Summary:

Managing access to data critical to your business using ACL facilities in native VMS can be cumbersome and still is vulnerable to intruders or people acting in excess of their authority.

Want to be sure your critical records can't be accessed save at authorized places, times, and with the programs that are supposed to access them (instead of, say, COPY.EXE)?

Want to have protection against privileged users bypasssing access controls?

Want to be able to password protect individual files?

Want to be able to invisibly hide selected files from unauthorized intruders?

EACF builds in facilities permitting all of these, and is not vulnerable to intruders who disable the AUDIT facility as all other commercial packages which purport to monitor access are.

Description: When your business depends on critical files, or when you are obliged by law or contract to maintain confidentiality of data on your system, in most cases the options provided by VMS for securing this data can be cumbersome and far too coarse-grained.

The problem is that certain kinds of access to data are often needed by people in a shop, but other access should be prevented and audited. Moreover, the wide system access that can come as a result of having system privileges often does not mean that it should be used to browse or disclose data stored on the system. A system manager will in general not, for example, have any valid reason to browse the customer contact file, the payroll database, or a contract negotiation file, save in a few cases where these files need to be repaired or reloaded from backups. Likewise, a payroll clerk may need read and write access to the payroll file, but not in general with the COPY utility, nor from a modem, nor in most cases at 4AM. Finally, a person who must have privileges to design a driver and test it should ordinarily not have the run of the file system as well.

Given examples like these, it is easy to see that simple authorization of user access to files is inadequate. While it is possible to build systems that grant identifiers to attempt some extra control, these can be circumvented by privilege, and create very long ACLs which become impossible to administer over a long period as users come and go.

What is needed is a mechanism that is secure, cannot be circumvented by turning on privileges, and which provides a simple to administer and fine grained control that lets you specify who can get at your critical files, with what images, when, from where, and with what privileges. It is also desirable to be able to control what privileges the images ever see, and to be able to check critical command files or images for tampering before use, so that they cannot be used as back doors to your system. It should be possible to demand extra authentication for particular

files as well, and to prevent a malicious user from even seeing a particularly critical file unless he can be permitted access.

EACF is a VMS add-in security package which provides abilities to control security problems due to intruders, to damage or loss by system "insiders" (users exceeding their authority), and to covert code (worms and viruses). It provides a much easier management interface to handle security permissions than bare VMS and provides facilities permitting control over even privileged file accesses, for cases where there are privileged users whose access should be limited. Unlike systems which only intercept the AUDIT output, EACF can and does protect against ANY file accesses, and can protect files against deletion by unauthorized people or programs in real time as well as against access.

EACF offers the following capabilities:

* ***Files*** can be ***password protected individually.*** If a file open or delete is attempted for such a file and no password has been entered, the open or delete fails.

* Access can be controlled by ***time of day.*** Added EACF protections can be in place only some of the time, access can be denied some times of day, write accesses can be denied at certain times, or various other modalities of access can be allowed.

* You can control ***who*** may access a file, ***where*** they may be (or may not be), ***with what images*** they may or may not access the file, and ***with what privileges*** the file may be accessed. Thus, for instance, it is trivial to allow a clerk access to the payroll file with the payroll programs, but not with COPY or BACKUP, not on dialup lines, and not if they have unexpected privileges. The privilege checks are helpful where there are consultants working on a system who should be denied access to sensitive corporate information but who need privileges to develop programs. With this system you can be sure your proprietary plans or data stay in house, and are available only to those with business reasons to need them, not to everyone needing system privileges for unrelated reasons. Unlike packages using the VMS Audit facility's output (which can be silently turned off by public domain code), *EACF **cannot** be circumvented by well known means.*

* You can ***hide files*** from unauthorized access. If someone not authorized to access a file tries to open it, they can be set to open instead some other file anywhere on the system. Meanwhile, EACF generates alarms and can execute site specific commands to react to the illegal access before it can happen. This can be helpful in gathering evidence of what a saboteur is up to without exposing real sensitive files to danger. Normal access goes through transparently.

* You can arrange that opening a file ***grants identifiers*** to the process that opens it and that closing it revokes these identifiers. Set an interpretive file to do this and set it to be openable only by the interpreter and you have a protected subsystem capability that works for 4GLs which are interpretive. (EACF identifier granting, privilege modification, and base priority alteration is protected by a cryptographic authenticator preventing forging or duplication.)

* You can actively **prevent covert code (*viruses* and *worms*)** from running in two ways. First, EACF can attach a cryptographic checksum to a file such that the file will not open if it has

been tampered with. Second, EACF can attach a privilege mask to a file which will replace all privilege masks for the process that opens it. By setting such a mask to minimal privileges, you can ensure that an untrusted image will never see a very privileged environment, and thus will be unable to perform privilege-based intrusions into your system even if run from a privileged user's account.

* You can ***control base priority*** by image. Thus, a particularly CPU intensive image can be made to run at lower than normal base priority even if it is run interactively.

EACF allows you to exempt certain images (e.g., disk defragmenters) from access checks, and it is possible to put a process into a temporary override mode also where this is needed. EACF facilities are controllable per disk, and impose generally negligible overhead. EACF will work with any VMS file structure using the normal driver interfaces. Also, EACF marking information resides sufficiently in kernel space that it cannot be removed from lower access modes, yet it uses a limited amount of memory regardless of volume size.

Best of all, the EACF protection is provided ***within the file system*** and does not depend on the audit facility. Thus it prevents file access or loss BEFORE it happens, and does not have to react to it afterwards. EACF allows all of its security provisions to be managed together in a simple screen-oriented display in which files, or groups of files, can be tagged with the desired security profiles or edited as desired.

Supported systems:

EACF runs on VAX based VMS systems running VMS 5.5 or later, or AXP based VMS systems running VMS 6.1 or later.

EACF is brought to you by

General Cybernetic Engineering
18 Colburn Lane
Hollis, NH 03049
603 465 9517
Everhart@GCE.mv.Com