To: MTB Distribution

From: C. D. Tavares

Date: April 3, 1979

Subject: Son of Authentication vs. Resource Management

This MTB results from a design review held to discuss problems with the current authentication algorithm for volumes. Your comments and suggestions are solicited; however, be advised that they must be received NO LATER THAN Monday, April 9, since the contractual delivery date for this product is actually April 1.

MOTIVATION

During the discussion of the current algorithm's unappreciated practice of "canonicalizing" volume names into a "standard" format, it became apparent that the proposed solution (discarding canonicalization entirely), beside being philosophically distasteful to some of those attending, suffered from the flaw that certain I/O modules must obey standards that demand that the volumes they manipulate have names in some canonical form of their own choosing. Some of these forms, in fact, conflict; making it very hard for Resource Management to choose a canonical form of a volume name that will satisfy every such format. This MTB will describe some of the problems and suggest a proposed solution. Although the problems described exist for many volume types managed by Resource Management, we will be primarily discussing tapes, since they represent the immediate problem, and since their problems are largely typical of those which will be experienced in the future when managing other volume types.

WHAT IS AUTHENTICATION?

When RCP is asked to mount a tape, part of its job is to make sure that the actual volume mounted is the correct volume. This process is known as authentication. Typically RCP tries to do this automatically by reading the magnetic label, interpreting it in various character codes, densities, and formats until it has a recognizable label in some standard format (e.g., ANSI, IBM, GCOS, Multics) and then comparing the contents of the tape name field with the volume name the user supplied in his mount request. If these match, no operator intervention is required.

However, in some cases, either the names do not match, the tape label (if any) is in no known standard format, or the tape is blank, such that there is no automatic authentication possi-

ble. In this case, RCP requests manual authentication from the operator. This is performed using a three-letter authentication code on a paper sticker which is permanently affixed to the reel. This three-letter code is a non-intuitive hash function of the official name of the tape, which is also present on a sticker. A standard program exists that will generate the proper authentication code and create both labels described. The algorithm used to generate the codes is designed so that authentication codes for volumes with very similar names will be wildly different, to minimize the chance of the operator mounting a volume of the wrong, but similar, name. However, this feature makes the authentication algorithm very sensitive to minor discrepancies in similar forms of the "official" volume name; e.g., the names U323, u323, and U-323 will normally generate quite different authentication codes unless otherwise instructed.

TYPICAL REASONS FOR CANONICALIZATION

For historical reasons, tapes have been typically treated as if their names were numbers. Resource management makes no such assumption, preferring to treat the names of all resource as 32-character entities. But other systems and standards (e.g., IBM, ANSI, GCOS) have sets of rules that were originally designed for standard tape "numbers" which include common practices such as right-adjusting the tape number and inserting leading zeroes to fill it to the correct number of character positions. Because there are many users who would be upset if they couldn't use alphabetic characters at all, some provisions are made in each of these formats for alphabetics. Typically a tape "number" with alphabetics is left-adjusted and padded with trailing spaces to the proper number of character positions. These are simple forms of volume name canonicalization. Another operation that is required by certain formats is the translation of lower case letters to upper case (e.g., ANSI and IBM formats; GCOS format uses BCD which is effectively upper case since there is no lower case BCD.) Some formats do not allow certain characters (e.g., tildes) to be used in labels at all. Last, some I/O modules implementing defined standards become upset if they are asked to manipulate volumes whose names are longer than allowed for in the label field (e.g., IBM and ANSI standards refuse to manipulate a volume with a name longer than six characters.) We will refer to labels produced by such a restrictive standard as "highly-processed" labels, because of the number of steps necessary to transform arbitrary resource names into a form that is acceptable to them.

Although in the case of names (e.g., segment names) it is usually the Multics philosophy that "what you see is what you get", we are forced by considerations of these standards to perform some type of canonicalization of volume names at some stage in the processing of mounts in order to perform volume authentication. Presently this canonicalization is performed on both the user-supplied name and the name read from the magnetic label, and the results compared.

DEFICIENCIES OF THE CURRENT APPROACH

One problem with the current approach is that the authentication character algorithm works on the built-in assumption that tape names are never more than six characters; are always leading-zero filled if numeric; are always trailing-blank-filled if not; and, should any tape name be longer than six characters, the extra characters are not worthy of any consideration. The implications of these assumptions are worrisome. For instance, two tape_mult_ tapes (e.g., backup tapes) named "COMPDUMP3" and "COMPDUMP8", although easily confused, carry the same authentication code!

Second, with the advent of Resource Management, most sites will be choosing to use the automatic registration facility, as opposed to having to preregister all tapes known to the system and preacquire to the proper owners. Automatic registration works as follows: A user requests a mount of a tape. RCP asks Resource Management whether this user has access to perform the mount. Resource Management replies that it hasn't any knowledge of any such tape at all. RCP checks to see if automatic registration is in order (via a flag in installation_parms.) If so, RCP prints a mount message to the operator. Then, whether or not the label matches, the operator is required to manually authenticate the tape, and is warned that if he agrees to authenticate it, the requesting user will be given full ownership of that tape. If the operator agrees, RCP calls the registration entry-point of Resource Management to register the tape and acquire it to the user.

Under this scenario, it is possible for an innocent user to accidentally (or a malicious user to purposely) acquire ownership of a tape that in fact has already been registered and is already owned by another, simply by asking for a mount via a similar name. For instance, if tape "foobar" is already registered and acquired, a user may request it via the name "FOOBAR". There is a good chance that the operator will manually authenticate the request, since the labels are "close enough", with the result that there are now two entries in the registry for the same tape, with different owners, different methods of determining access control, and so on.

HOW CAN WE DO IT CORRECTLY?

One way to prevent this happening at a site (besides the obvious one of not allowing automatic registration) would be to change the current authentication algorithm to not perform ANY canonicalization of the tape name, so that the authentications *generated* from "foobar" and "FOOBAR" are different (thus disabling the operator from authenticating the "wrong" request even if he wants to). This has some ramifications-- one being that at

a site such as MIT, where the "official" tape names all contain leading zeroes for historical reasons, all users would always be forced to type the leading zeroes when requesting tapes. Although this procedure is secure, some Multicians feel it to be morally repugnant. Another ramification is that some translation will still have to be performed on the magnetic tape labels at mount time, since they themselves might represent some standard's idea of the "proper" representation of the official name of the tape, that does not exactly match it. Therefore, it seems that we can never completely do away with canonical conversions EVERYWHERE.

However, when and how this canonicalization is done is very important. Canonicalization performed at an improper time in the authentication sequence directly affects security. Every canonicalization translation is a many-to-one translation (e.g., names with upper case and names with lower case are both translated into names with upper case; the same goes for names with leading zeroes and names without). Every such translation one uses to create a highly-processed label multiplies the possible number of reels with different "official" resource names that may possess this label. Any canonicalization occuring to the user-supplied resource name after the time that Resource Management gets hold of it serves to destroy the security of automatic registration. It seems clear that any solution based in increasing, rather than decreasing the absolute amount of canonicalization performed at this point in the sequence will increase, rather than decrease, our problems.

A second possible method of preventing this problem would be to enforce some type of "system standard" resource name. All resource names provided by the user would be canonicalized by some algorithm BEFORE ever being passed to Resource Management and RCP. Proponents of this have argued that a site would be foolish to have two tapes in their libraries named, for example, u301 and U301; while others insist that this should be a manually-enforced site concern. However, there is the consideration that by enforcing some arbitrary "Multics standard"" canonicalization, we are preventing one site from having all their tapes named consistently in upper case, and another site from having a consistent library of lower case volume names. (And without value judgement on the wisdom of a site's having tape names differing only in case, it was pointed out that at least one such site does in fact exist!) It would be possible to provide to sites a mechanism by which a site-specific canonicalization procedure could be used by Resource Management. A "sample" standard canonicalization routine could then be shipped as part of the product, and used as a default routine.

However, does this solve the problem? The first obvious design feature of such a default is that it should satisfy ALL label constraints imposed by all implemented and/or known highly-processed label formats, to the purpose of approaching the ideal of one-name/one-label correspondence that will make auto-

matic authentication efficient and secure, and the necessity of manual authentication rare. Although at the design review not much stock was given to arguments that it is likely that some day we may choose to implement a standard for which the label format is unreconcilable with some already known standard format, in fact this has already happened! One such example is that ANSI tapes require a six-character label name of which the first character is nonblank, and if entirely numeric, the last character is also nonblank; while GCOS tapes require either (depending on how you interpret the format) a five-character label name or a six-character label name of which the first character is ALWAYS blank. Clearly we cannot create a "standard" Multics label to satisfy all tape formats without re-introducing canonicalization at the tape-label level.

This introduces another problem. The routines that match tape labels to resource names have to realize that there may be multiple resource names that translate into a single given tape label. Therefore, to know whether or not a tape label that it has read could have been created by more than one resource name, it has to know something about the properties of the canonicalization algorithm that the site has chosen for use. For instance, if a user requests tape "FOO", and the ANSI label reads "FOOØØØ", this could still represent an ambigous case, if the site's chosen canonicalization algorithm also allowed resource names of "Foo" or "foo" (from which tape_ansi_ would generate exactly the same label). At this point, the idea of site-specifiable canonicalization rapidly loses its attractiveness. We are left with two alternatives: drastic, system standard canonicalization to the greatest common denominator; or designing an authentication system that is not so sensitive to information contained in highly-processed name fields in tape labels.

The first approach is too restrictive. As such standard labels get shorter and shorter, we either have to shorten the canonical label (and five characters is too short already) or handle the security problem of the label not being able to hold possibly significant characters of the volume name. We are back to the second solution; namely, additional authentication information found elsewhere than in the volume name field on the label.

This second solution would be to make use of defined "scratch" spaces in label formats to store further authentication information. The I/O modules responsible for creating "highly-processed" labels would also be responsible for storing into a selected scratch field the authentication code generated from the full, unprocessed name the user supplied. The authentication decision would then be performed in this sequence:

1) Issue the mount request and read the label.
2) Decide what type of label this is (ANSI, GCOS, Multics, etc.)
3) Take the resource name given by the user and perform the same type of translation on it that the I/O module normally does to

produce the given type of label. This might require leading zeroes, trailing blanks, etc. For the case of Multics labels, this step is not required.
4) Compare the labels. If they do not match, ask for manual authentication.
5) If the labels compare, but the label standard is highly-processed, examine the appropriate scratch field in the label. Generate the authentication from the user-supplied resource name. If these do not match, ask for manual authentication.

The fact that the authentication codes are inserted by ring 4 software does not affect the security of the information in any way. (See MTB XXX for a discussion of the principles involved.) Basically, if a user decides to bollix the authentication information on his own tape, the major effect is to get his own tape authenticated more often for himself. A second-order effect is to let someone else have unusual access to that tape, should the operator erroneosly mount that tape for someone else; however this is not only a rare occurrence, but other controls newly introduced in Resource Management itself cut even this probability by orders of magnitude.

WHAT PROBLEMS REMAIN?

First, I/O modules which create highly-processed labels will have to be taught to put authentication codes into fields on the tape. Actually there are only two modules affected: The tape_ansi_/ibm_ module which, for all practical purposes, is one module; and the GCOS encapsulator in which, I assume, someone knows about generating GCOS tape labels. Note that Multics standard tapes need no authentication fields, as their volume names are unprocessed.

Second, tapes with highly-processed labels that are not often written will require constant manual authentication until the next time they are written, with the I/O module that knows about inserting authentication codes. This will also hold true for all stranger tapes, until the first time they are written. This is the most unfortunate drawback of the solution, but one that, because of the very nature of the information loss inherent in highly-processed labels, is inescapable.

PLANNED IMPLEMENTATION

Tape modules that create highly-processed labels will be modified to insert authentication codes into scratch fields in the label. This will affect two modules.

The authentication code algorithm will be modified to make all characters in a resource name significant, and to remove some of the special cases that make it currently dependent on

six-character fields. (This will invalidate all MIT's current labels; however, they have informed us that this will not create a problem for them.)

The tape label generating program will be modified to use the new authentication code generator.

Ring one label validation will be changed to perform only forward canonicalization (resource name to label format) to check volume label fields; and to perform the authentication steps outlined above.

Resource names given to Resource Management and RCP will not be pre-canonicalized, with the exception that leading zeroes will be stripped off for purely aesthetic reasons.