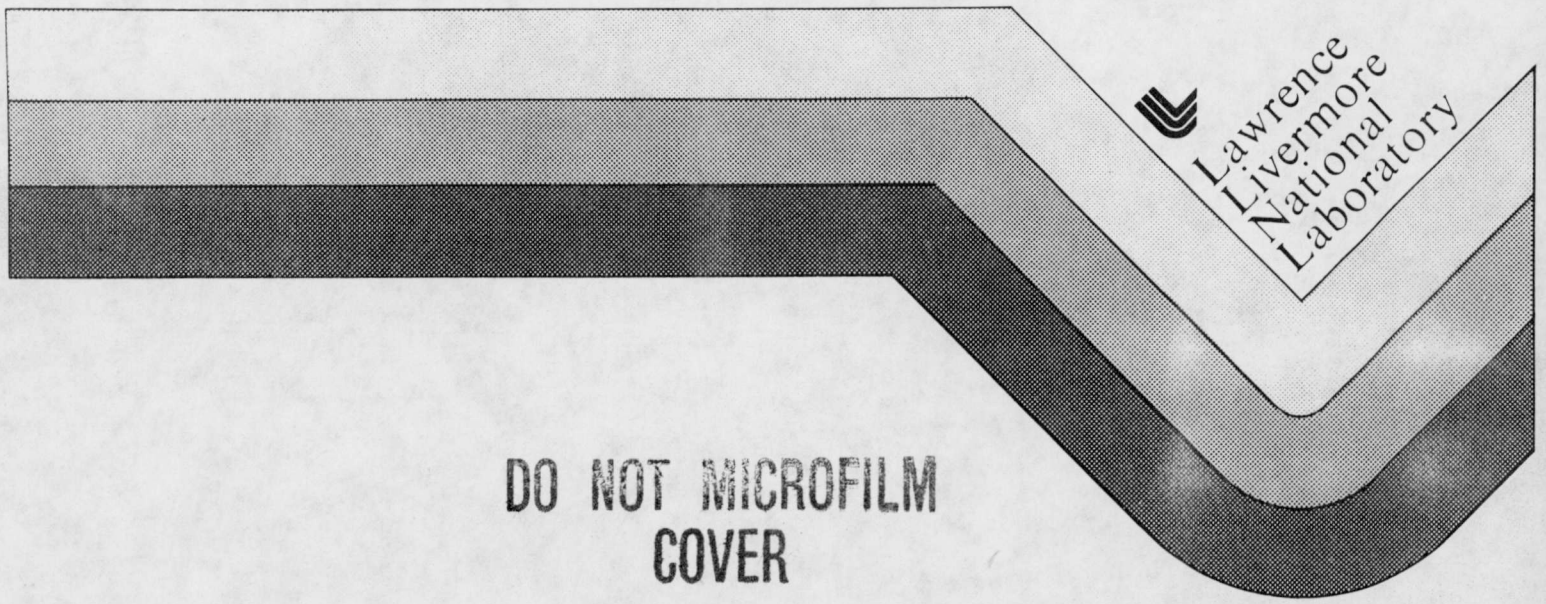
UCRL-MA-103421

SPI/VMS User Manual
Version 1.0

Marianne E. King
Computer Communications and Security Group
Lawrence Livermore National Laboratory
P.O. Box 808, L-619
Livermore, CA 94550
(415) 423-4116

March 12, 1990



DO NOT MICROFILM
COVER

DISCLAIMER

UCRL-MA--103421

DE90 010693

SPI/VMS User Manual
Version 1.0

March 12, 1990

Marianne E. King
Computer Communications and Security Group
Lawrence Livermore National Laboratory
P.O. Box 808, L-619
Livermore, CA 94550
(415) 423-4116

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract No. W-7405-ENG-48.

SPI/VMS User Manual

1. Introduction

1.1. Purpose

This manual explains how to use the Security Profile Inspector for the VMS operating system (SPI/VMS) to detect potential security problems. This manual provides different levels of information for different levels of understanding of SPI/VMS and the VMS operating system.

1.2. Scope

This manual is for system managers or computer system security officers (CSSOs), who are responsible for the security of a computer system. It is assumed that SPI/VMS users have a basic understanding of the DEC Command Language (DCL) concepts (e.g., file specifications, commands, and logical names) and know how to perform basic system management tasks (e.g., setting up new user accounts).

1.3. Overview

The remainder of this document is organized as follows:

- Section 2 describes the SPI/VMS commands. This section discusses each security problem that SPI/VMS commands detect.
- Section 3 is a tutorial on how to run SPI/VMS. This section is for novice VMS and SPI/VMS users.
- Section 4 is the reference section for SPI/VMS commands.
- Section 5 describes the advanced techniques for SPI/VMS commands.
- Appendix A contains the conventions used in this manual.
- Appendix B contains the definition of terms, acronyms, and abbreviations.
- Appendix C provides the references for this document.

2. What is SPI/VMS?

System managers and CSSOs are responsible for the security of their computer systems. Effective security measures can prevent some of the following situations:

- The denial of computer resources to legitimate users (e.g., A malicious user crashes the system).
- The delay of computer resources to legitimate users (e.g., An unauthorized user steals CPU cycles).
- The disclosure of computer data (e.g., A malicious user steals company proprietary computer data).
- The destruction of computer data (e.g., A malicious user deletes all the files on the system).
- The distortion of data (e.g., A malicious user changes the system value of pi).

Whenever a breach in computer security occurs, it costs a company time and money to assess and repair the damage. Also, computer security for government installations is a matter of national security.

System managers and CSSOs need to set up proper security measures to prevent unauthorized users from gaining access to their computer systems. Once malicious users have access to a machine, they can exploit weaknesses in the operating system or other software to damage the computer system. Note that a malicious user can be an unauthorized user or an authorized user who is dishonest or disgruntled with the company. Therefore, the system managers and CSSOs must protect their systems from any computer user.

SPI/VMS is a collection of security tools or commands which reports potential security problems of a computer system. It is up to the system manager or CSSO to use this information to secure their computer systems.

Currently, SPI/VMS consists of 3 commands which are described in the next section. For future SPI/VMS commands read the *SPI/VMS Requirements Specification* document.

2.1. PASSWORD_CHECKER

Passwords are one of the most vulnerable areas of computer security. An unauthorized user can gain access to a computer system by guessing the password of a valid user account. Once an unauthorized user has access to a machine, the user can exploit weaknesses in the operating system to damage the system or steal computer data. The unauthorized user's job is not too difficult because computer users often pick easy to guess passwords such as their own name, the names of their friends, the names of their pets, the names of famous people or places, or words in the English dictionary. The purpose of the PASSWORD_CHECKER is to detect easy to guess passwords so that the system manager can notify the user to change his password to a better one.

At a minimum, the PASSWORD_CHECKER checks for passwords equal to the user's name and for null passwords. The system manager can also provide the

PASSWORD_CHECKER with dictionaries, which are files containing a list of possible passwords. SPI/VMS is delivered with 3 dictionaries. TRIVIAL.PASS contains about 9,000 commonly used passwords such as common first and last names, historical places, and names of characters from popular books and movies. DICT.WORDS contains about 30,000 commonly used English words. WEBSTERS.WORDS contains about 250,000 words, which are all the words listed in the *Webster's Dictionary*. It is also recommended that the system manager come up with a list of possible passwords common to the company such as company name and project names.

In addition to the dictionaries providing possible passwords, there are 2 other ways to specify possible passwords. The REVERSE_DICTIONARY qualifier uses the words in an input dictionary with the order of their letters reversed as possible passwords. The NAME_PERMUTATION qualifier uses permutations of the names and initials in the account and owner fields as possible passwords. The owner and account fields usually contain information about a user.

Other qualifiers of the PASSWORD_CHECKER allow the system manager to check the passwords of only those users who have at least one of the specified PRIVILEGES and those users who have changed their passwords SINCE or BEFORE the specified time.

2.2. FILE_ACCESS

VMS provides different mechanisms for specifying file access. UIC-based file protection determines access by the user identification code (UIC) of the owner of a file, the UIC of the user requesting access to that file, and the UIC protection code assigned to the system, owner, group, and world user categories. ACL-based file protection determines access by access control lists (ACLs) whose access control entries (ACEs) can grant or deny access to individual users. The SYSPRV, BYPASS, GRPPRV, and READALL privileges can override the UIC and ACL-based file protection mechanisms. For a more detailed discussion on file protections, read the *Guide to VMS System Security*.

A system manager tries to protect sensitive files by placing a combination of UIC-based and ACL-based protections on a file. However, it is possible to make mistakes, especially with ACLs. The system manager manager may inadvertently grant access to an undesired user. Below are some examples:

- The system manager is unaware that an undesired user holds a particular identifier and an ACE grants access to the holders of that identifier.
- The system manager is unaware that an undesired user has a particular UIC and an ACE grants access to the holder of that UIC.
- The system manager is unaware that a user has privileges which will override the UIC and ACL-based protections.

The FILE_ACCESS reporting command reports all the users who have access to the specified file, their type of access, and the reason for their access. The reasons for granting access would be the result of UIC-based protection, an ACE, or a privilege. Now a system manager can know all the users who have access to a sensitive file.

2.3. FIND_IDENTIFIER

The *Guide to VMS System Security* suggests that system managers remove occurrences of a user identifier or a general identifier from ACLs before removing the identifiers from the system. If the system managers fail to do this, hexadecimal identifiers and numeric UICs in ACLs will be created. Hexadecimal identifiers in ACLs occur when a general identifier has been removed from the rights database. Numeric UICs in ACLs occur when a user has been removed from the system. There is a danger that a new user may be assigned the value of a numeric UIC, which would give the new user the same access rights as the previous user.

Therefore, whenever a system manager is about to delete a general identifier from the rights database or a user from the system, he should check to see if any ACL contains the identifier to be deleted by using the FIND_IDENTIFIER command. The system manager can also use FIND_IDENTIFIER to check for numeric identifiers.

3. SPI/VMS Tutorial

This section provides novice SPI/VMS users information on how to use SPI/VMS. It gives a description and an example of each SPI/VMS command and its qualifiers.

This section assumes that you have a basic understanding of VAX/VMS concepts. As a note to novice VMS users, some command examples in this section are longer than 1 line. A command line is extended by a hyphen (-) as the last character of the line. The command is continued on the next line. For more information on the command line continuation read the *VMS DCL Concepts Manual.*

3.1. Getting Started

To get started with SPI/VMS log in to the account in which you installed SPI/VMS. Then run the command file, STARTUP.COM, to set up the command table and logical names necessary to run SPI/VMS commands. You can do this by typing the command below. If STARTUP.COM is not in your current directory, then you must type in the appropriate pathname as well.

```
$ @STARTUP
```

You can also insert the above command line into your LOGIN.COM file so that the SPI/VMS environment is set up every time you log in to the system.

3.2. Privileges

At a minimum the SPI/VMS commands require the SYSPRV and SECURITY privileges. If a certain command needs other privileges, it will be stated in the documentation. In order to set up the minimum privileges necessary to run SPI/VMS type in the following command.

```
$ SET  PROC/PRIV=(SYSPRV, SECURITY)
```

3.3. PASSWORD_CHECKER

At a minimum, the PASSWORD_CHECKER checks for passwords equal to the user's name and for null passwords. However, you can have the PASSWORD_CHECKER try more passwords by specifying files which contain a list of possible passwords. These files are often referred to as dictionaries. The TRIVIAL.PASS, DICT.WORDS, and WEBSTERS.WORDS dictionaries are delivered

with SPI/VMS and can be found in your SPI/VMS installation directory. TRIVIAL.PASS contains about 9,000 commonly used passwords such as common first and last names, historical places, and names of characters from popular books or movies. DICT.WORDS contains about 30,000 commonly used English words. WEBSTERS.WORDS contains about 250,000 words, which are all the words in the *Webster's Dictionary*. It is recommended that you use TRIVIAL.PASS AND DICT.WORDS as your input dictionaries. As a warning use the WEBSTERS.WORDS dictionary with extreme caution. Since it contains so many words, it takes the PASSWORD_CHECKER a very long time to process the dictionary (7 to 8 hours per user depending on the system load).

If you type the command below, you will check all the user accounts on the system using the recommended TRIVIAL.PASS and DICT.WORDS dictionaries. This example assumes that the files, TRIVIAL.PASS and DICT.WORDS, are in your current directory. If they are not in your current directory, then you must also specify the appropriate pathname. A star (*) is used as a wildcard to specify all the users on the system. The results are printed out to your terminal screen.

```
$ SPI  PASSWORD_CHECKER  /DICTIONARY=(TRIVIAL.PASS, DICT.WORDS)  *
```

Below is an example of the output output from the PASSWORD_CHECKER. It outputs the names of the user accounts examined, the password of the account if it was found, and whether or not the password was found. It also prints out the number of accounts checked, the number of passwords found, and the percentage of passwords found.

```
SPI/VMS 1.0     PASSWORD_CHECKER     6-OCT-1989 08:52:30.78

Username                Password                    Found?
========                ========                    ======
YOUNGER                 COLE                        YES
DALTON                                              NO
JAMES                   JESSE                       YES
CASSIDY                                             NO
COAL                    CHARLES                     YES
----------------------
5 user accounts checked.
3 passwords found.
60.00 % of passwords found.
```

If the results show that some users have easy to guess passwords, then you must make sure that these users change their passwords. In the previous example the

system manager should make sure that the users, YOUNGER, JAMES, and COAL change their passwords.

3.3.1. Specifying Users

When specifying users, you can either specify a user by name or use wildcards. You can also specify more than one user on a command line by separating the user specifications by commas. The example below specifies user accounts with names which start with "A", "B", and "C".

```
$ SPI  PASSWORD_CHECKER  /DICTIONARY=TRIVIAL.PASS  A*, B*, C*
```

3.3.2. Reverse Dictionary

When a file is specified with the REVERSE_DICTIONARY qualifier, then the PASSWORD_CHECKER uses the letters of a word in reverse order as a possible password. The example below checks the passwords of all user accounts on the system using TRIVIAL.PASS as a reverse dictionary.

```
$ SPI  PASSWORD_CHECKER  /REVERSE_DICTIONARY=TRIVIAL.PASS  *
```

3.3.3. Name Permutation

When you specify the NAME_PERMUTATION qualifier, the PASSWORD_CHECKER uses the permutations of the names and initials found in a user's owner and account fields. These fields usually contain information about the user. For example, suppose the owner string of the account RNIXON is "Nixon, Richard Milhous" and the account field is blank. The PASSWORD_CHECKER would use combinations of the three names and initials. Possible passwords would be RMN, RICHARD, RMNIXON, etc.

```
$ SPI  PASSWORD_CHECKER  /NAME_PERMUTATION  *
```

3.3.4. Privileges

The PRIVILEGE qualifier allows you to only check those users who have at least one of the specified privileges. In VMS you should be most concerned with user accounts which have privileges such as SETPRV, SYSPRV, CMKRNL, SYSNAM, PHYIO, ACNT, SECURITY, OPER, READALL, and BYPASS. If a malicious user gains access to a privileged account, the user has the privileges to seriously damage your system. You can specify a privilege by its name You can specify all the privileges by specifying "ALL". You can unspecify a privilege by using the negative of a privilege name (e.g., NOSYSPRV). See Table A-1 in the *DCL Concepts Manual* for the list of VMS privileges. To examine the user accounts which have privileges other than the minimum NETMBX and TMPMBX type the following command:

```
$ SPI  PASSWORD_CHECKER  /DICTIONARY=TRIVIAL.PASS -
  /PRIVILEGES=(ALL, NOTMPMBX, NONETMBX)  *
```

3.3.5. Since

The PASSWORD_CHECKER has the SINCE qualifier which checks only the accounts of users who have changed their passwords since the time specified. This qualifier is useful when several new users are added to the system, and you would like to make sure they select hard to guess passwords. To check for passwords changed since October 4, 1989 type in the following command:

```
$ SPI  PASSWORD_CHECKER  /DICTIONARY=TRIVIAL.PASS -
  /SINCE=4-OCT-1989  *
```

3.3.6. Before

The PASSWORD_CHECKER has the BEFORE qualifier which checks only the accounts of users who have changed their passwords before the time specified. This qualifier is the opposite of the SINCE qualifier. To check for passwords changed before October 4, 1989 type in the following command:

```
$ SPI  PASSWORD_CHECKER  /DICTIONARY=TRIVIAL.PASS -
  /BEFORE=4-OCT-1989  *
```

3.3.7. Output

In order to save the output of the PASSWORD_CHECKER in a file you must specify a file with the OUTPUT qualifier. The example below writes the results out to the file, RESULTS.LOG.

```
$ SPI  PASSWORD_CHECKER  /DICTIONARY=TRIVIAL.PASS -
  /OUTPUT=RESULTS.LOG  *
```

3.3.8. Report All Accounts

The default for SPI/VMS is to print out all the user account names whose passwords were examined. If you wish to print out only those user account names which have easy to guess passwords, then use the NORPTALL_ACCOUNTS qualifier. Below is an example of a command whose results only print out the accounts of users with easy to guess passwords:

```
$ SPI  PASSWORD_CHECKER  /DICTIONARY=TRIVIAL.PASS -
  /NORPTALL_ACCOUNTS  *
```

3.3.9. Report Password

The default of SPI/VMS is to print out the bad password. If you do not want the password to be printed out, then use the NORPTPASSWORD qualifier. Below is an example of a command whose results do not print out the passwords that were found.

```
$ SPI  PASSWORD_CHECKER  /DICTIONARY=TRIVIAL.PASS  /NORPTPASSWORD  *
```

3.4. FILE_ACCESS

In addition to the SECURITY and SYSPRV privileges, the FILE_ACCESS command needs the BYPASS privilege. Type in the following command to make sure you have the necessary privileges.

```
$ SET  PROC/PRIV=(BYPASS,SECURITY,SYSPRV)
```

In order to run the FILE_ACCESS command you must specify the file or files you want to examine. You may use wildcards in your file specifications. Below is an example of how to find out all the users who have access to a specified file. The results are printed out to your terminal.

```
$ SPI  FILE_ACCESS  CONTRAS.DAT
```

Below is the output from the FILE_ACCESS command from the previous example.

```
SPI/VMS 1.0     FILE_ACCESS     6-OCT-1989 08:52:30.78

DUA1:[NORTH]CONTRAS.DAT
  CASEY           R       (IDENTIFIER=[STAFF,CASEY],ACCESS=READ)
  NORTH           RWED    UIC
  POINDEXTER      R       UIC
  SYSTEM          RWEDC   SYSPRV
  USERP           RWEDC   SYSPRV
```

3.4.1. Output

In order to save the output of the FILE_ACCESS command in a file you must specify a file with the OUTPUT qualifier. The example below writes the results out to the file, RESULTS.LOG.

```
$ SPI  FILE_ACCESS  /OUTPUT=RESULTS.LOG  CONTRAS.DAT
```

3.5. FIND_IDENTIFIER

In order to run the FIND_IDENTIFIER command you must specify the file or files you want to examine and the general identifier you want to find. You may use wildcards in your file specifications. Below is an example of how to check for the identifier, PRILEY, in the files in the current directory. The results are printed out to your terminal.

```
$ SPI  FIND_IDENTIFIER/IDENTIFER=PRILEY  *.*
```

Below is the output from the FIND_IDENTIFIER command from the previous example.

```
SPI/VMS 1.0     FIND_IDENTIFIER     6-OCT-1989 08:52:30.78

DUA1:[LAKERS]PLAYERS.DAT
  (IDENTIFIER=[STAFF,PRILEY],ACCESS=READ)

DUA1:[LAKERS]DEFENSE.DAT
  (IDENTIFIER=[STAFF,PRILEY],ACCESS=READ)
```

3.5.1. Numeric Identifiers

In order to search files for ACLs containing numeric identifiers. Below is an example of a command that searches all the files in the current directory for numeric identifiers.

```
$ SPI  FIND_IDENTIFIER/NUMERIC  *.*
```

3.5.2. Output

In order to save the output of the FIND_IDENTIFIER command in a file you must specify a file with the OUTPUT qualifier. The example below writes the results out to the file, RESULTS.LOG.

```
$ SPI  FIND_IDENTIFIER  /IDENTIFIER=PRILEY  /OUTPUT=RESULTS.LOG  *.*
```

3.6. Executing SPI/VMS in Batch Mode

You may find that some SPI/VMS commands take a long time to complete. Instead of running SPI/VMS interactively at your terminal, you may want to run SPI/VMS in the batch mode. To do this create a file which contains the desired SPI/VMS command. The command below is an example of using the PASSWORD_CHECKER command. Note than when you execute a command in the batch mode, the current directory is your login directory. Therefore, if you reference any files that are not in your login directory, you should reference these files by specifying their complete pathnames. Also, note that you must run the SPI/VMS STARTUP.COM file to set up the SPI/VMS environment. This example assumes that you have set up the SPI/VMS environment in you LOGIN.COM file.

SPI_BATCH.COM

```
$ SPI  PASSWORD_CHECKER  /DICTIONARY=(TRIVIAL.PASS, DICT.WORDS)  *
```

To submit a batch job to the batch queue type in the following command.

```
$ SUBMIT/NOTIFY  SPI_BATCH
```

3.7. HELP

To get online help for SPI/VMS use the VAX/VMS help utility. To get help type in the help command:

```
$ HELP @SPI
```

A list of SPI/VMS commands will be displayed. To display the help text for a command, type in the name of that command at the "@SPI Topic" prompt. For example the command below will display the help text for the FILE_ACCESS command.

```
@SPI Topic? FILE_ACCESS
```

4. SPI/VMS Reference Section

4.1. PASSWORD_CHECKER

The PASSWORD_CHECKER checks the specified user accounts for easy to guess passwords.

This command requires the SYSPRV and SECURITY privileges.

FORMAT: SPI PASSWORD_CHECKER *username-specifier[,...]*

PARAMETER: username-specifier[,...]

The parameter specifies one or more users whose passwords will be checked. You can use wildcards in the username-specifier.

DESCRIPTION:

The PASSWORD_CHECKER checks the specified user accounts for easy to guess passwords. At a minimum the PASSWORD_CHECKER checks for passwords equal to a user's name and null passwords. Certain qualifiers allow you to specify lists of possible easy to guess passwords, what type of information to output, and what type of user accounts to check.

QUALIFIERS:

/BEFORE=[time]

Selects only the accounts whose passwords have changed before the specified time. The time can be specified as absolute time, a combination of absolute and delta times, or as one of the following keywords: TODAY, TOMORROW, or YESTERDAY. If a time is not specified, the default is TODAY.

/DICTIONARY=(file-spec[,...])

Provides the PASSWORD_CHECKER with a list of possible passwords. Wildcard characters are supported in the file specification. The file must have the format of one word per line.

/NAME_PERMUTATION

Uses permutations of the names and initials of the names found in the account and owner fields of SYSUAF.DAT as possible passwords.

/OUTPUT[=file-spec]

Controls where the output of the command is sent. By default, the display

is written to the current SYS$OUTPUT device. No wildcard characters are allowed.

/PRIVILEGES=(privilege-spec[,...])
Specifies that only the user accounts with at least one of the specified privileges are checked.

/RPTALL_ACCOUNTS(default)
/NORPTALL_ACCOUNTS
Controls whether the names of all accounts checked are printed.

/RPTPASSWORD(default)
/NORPTPASSWORD
Controls whether the "guessed" passwords are printed.

/REVERSE_DICTIONARY=(file-spec[,...])
For each word in a file, the PASSWORD_CHECKER reverses the order of the letters and uses them as possible passwords. Wildcard characters are supported by the file specification. The file must have the format of one word per line.

/SINCE=[time]
Selects only the accounts whose passwords have changed since the specified time. The time can be specified as absolute time, a combination of absolute and delta times, or as one of the following keywords: TODAY, TOMORROW, or YESTERDAY. If a time is not specified, the default is TODAY.

4.2. FILE_ACCESS

The FILE_ACCESS command reports all the users on the system who have access to the specified file.

This command requires the SYSPRV, SECURITY, and BYPASS privileges.

FORMAT: SPI FILE_ACCESS *file-spec*

PARAMETER: file-spec

The parameter specifies the file to be examined. You can use wildcards in the file-spec to examine multiple files. At a minimum you must specify a file name and a file type. However, the wildcards *.* will suffice.

DESCRIPTION:

The FILE_ACCESS command reports the type of access, and the reason for access that users have to the specified file.

QUALIFIERS:

/OUTPUT[=file-spec]

Controls where the output of the command is sent. By default, the display is written to the current SYS$OUTPUT device. No wildcard characters are allowed.

4.3. FIND_IDENTIFIER

The FIND_IDENTIFIER command reports all the files whose ACLs contain the specified identifiers.

This command requires the SYSPRV and SECURITY privileges.

FORMAT: SPI FIND_IDENTIFIER *file-spec*

PARAMETER: file-spec

The parameter specifies the file to be examined. You can use wildcards in the file-spec to examine multiple files. At a minimum you must specify a file name and a file type. However, the wildcards *.* will suffice.

DESCRIPTION:

The FIND_IDENTIFIER command reports the file name and the ACE containing the specified identifier. You can use the qualifiers to specify the identifiers you want to find.

QUALIFIERS:

/IDENTIFIER[=id-spec]

Specifies the general identifier you want to find. A valid general identifier can be up to 32 characters long and consist of the following characters: "A" - "Z", "a" - "z", "0" - "9", and "_".

/NUMERIC

Specifies that the command should look for numeric identifiers.

/OUTPUT[=file-spec]

Controls where the output of the command is sent. By default, the display is written to the current SYS$OUTPUT device. No wildcard characters are allowed.

5. SPI/VMS Advanced Techniques

This section describes some advanced techniques for SPI/VMS.

5.1. PASSWORD_CHECKER

The PASSWORD_CHECKER command checks the passwords of the users in the SYS$SYSTEM:SYSUAF.DAT file. However, if you wish to check the passwords of users in a file other than SYS$SYSTEM:SYSUAF.DAT, then type in the commands shown below.

```
$ SET  PROC/PRIV=SYSNAM
$ DEFINE/TABLE=LNM$PROCESS_DIRECTORY/EXEC  LNM$FILE_DEV -
  LNM$PROCESS, LNM$JOB, LNM$GROUP, LNM$SYSTEM
$ DEFINE/EXEC  SYSUAF  file-specification
$ SET PROC/PRIV=NOSYSNAM
```

The logical assignments above reassign the SYSUAF.DAT file for you user environment. It will not affect other users on the system. If you run the authorize utility or change your password, any modifications will be made to the newly defined SYSUAF. Therefore, if you want to access the original system SYSUAF.DAT file, SYS$SYSTEM:SYSUAF.DAT, you must log out and log back in to reset your logical name environment.

Appendix A
Conventions

Convention	Meaning
[ ]	Brackets indicate that the enclosed item is optional.
{ }	Braces enclose a list from which one element must be chosen.
|	The OR symbol separates alternatives within braces or brackets.
...	A horizontal ellipsis indicates that the preceding items can be repeated one or more times.
-	A command line is extended by typing a hypphen as the last character on the line. The command is continued on the next line.

Unless otherwise noted the following apply:

- All numeric values are represented in decimal notation.
- You terminate a command by pressing the RETURN key.