

TABLE OF CONTENTS

Introduction	1
Overview.....	1
Data Encryption Algorithms	1
Details about Dr. Crypto Usage.....	1
Installing Dr. Crypto	3
Implementing Dr. Crypto	4
Native Implementation Method	4
DRCRYPTO Macro.....	4
Optional BIM-EPIC Implementation Method.....	10
Messages	12
Index	14

Introduction

Overview

Dr. Crypto is a data encryption tool that provides security to files created using the Dr. D disk utility. Dr. Crypto protects your company's electronic data asset from unauthorized access when, for example, backups are taken off site.

Dr. Crypto operates on any machine that supports the VSE operating system. It requires Dr. D 6.8 or higher as well as SSL for VSE Release 1.5E or higher.

Dr. Crypto provides two implementation methods. The first, a native method, requires coding, assembling, and cataloging a module that defines all the possible seed phrases for the encryption/decryption process. The second is an optional method that requires BIM-EPIC Release 7.1 or higher.

Data Encryption Algorithms

Dr. Crypto supports encryption using the following standard algorithms as defined by the U.S. Department of Commerce National Institute of Standards and Technology's Federal Information Processing Standards Publications (also referred to as FIPS-PUBs).

- **Data Encryption Standard.** Data Encryption Standard (DES), defined in 1977 in FIPS PUB-46, is the standard cipher for protecting sensitive, but unclassified, computer data. The cipher uses 16 rounds with a single 8-byte key in its algorithm to encrypt data.
- **Triple Data Encryption Standard.** Triple Data Encryption Standard (DES3), defined in 1978 in FIPS PUB 46-3, is a cipher similar to DES with the exception of using 48 rounds with a 24-byte key in the algorithm. It provides additional layers of encryption security to the data compared to DES.
- **Advanced Encryption Standard.** Advanced Encryption Standard (AES), defined in 2001 in FIPS PUB 197, supersedes DES encryption. It uses 14 rounds with a 32-byte key in its encryption algorithm.

Details about Dr. Crypto Usage

When planning your backup strategy using Dr. Crypto, consider the following.

- Dr. Crypto can encrypt any file created by a Dr. D SAVE function and decrypt any save file used in a Dr. D RESTORE command, including save files created on tape or disk.
- Dr. Crypto neither encrypts nor decrypts save files created or restored with the data tape (DT=) feature.
- Never encrypt your standalone backups. The decryption routines function only in a VSE environment and not in the standalone environment.
- The functions that Dr. Crypto performs extend run times and increase CPU usage. The amount of overhead varies from file to file depending on the amount of data and the selected encryption algorithm. If your CPU has the hardware encryption features

installed, Dr. Crypto uses them. These hardware features help keep the overhead to a minimum.

Always consider the potential increase in overhead when deciding which backups to encrypt. For example, you may want to encrypt only those backups to be sent off-site. For volume backups, you may want to encrypt only SAVE=CURRENT backups, but not SAVE=ALL. (SAVE=ALL backups also encrypt empty tracks.)

Installing Dr. Crypto

As mentioned, Dr. Crypto requires the following:

- Dr. D Release 6.8 (or higher)
- SSL for VSE Release 1.5E (or higher)

The SSL for VSE routines may be those that IBM distributes with the operating system or those that CSI distributes with TCP/IP for VSE. You must obtain authorization for at least one set of routines from the appropriate vendor.

Once you install the above products, Dr. Crypto's components automatically install. To activate Dr. Crypto, you need only add an appropriate product key to your PRODKEY phase.

Dr. Crypto functions on any machine supporting the VSE operating system.

If you are running a machine that has the hardware crypto-assist features installed, Dr. Crypto uses those features.

Implementing Dr. Crypto

As mentioned, Dr. Crypto provides two implementation methods:

- A native implementation method that requires you to code, assemble, and catalogue a module that defines all possible seed phrases for the encryption/decryption process. This method requires no additional software other than Dr.D and SSL for VSE as described in the section Installing Dr. Crypto.
- An optional implementation method requires BIM-EPIC release 7.1 or higher.

Native Implementation Method

To implement Dr. Crypto in Native Mode, you must:

- Create a phase using the DRCRYPTO Macro as the source. With the DRCRYPTO Macro, you define the encryption/decryption algorithm as well as the seed phrases to be used when generating the actual encryption/decryption key.
- Add the ENCRYPTKEYS= operand to the SAVE and RESTORE control statements that use encryption.

DRCRYPTO Macro

To use Dr. Crypto in Native Mode, you must code, assemble, and catalog the DRCRYPTO Macro. There are 3 types of DRCRYPTO statements:

- The **BEGIN** statement must be the first statement coded. Use it to define the default seed phrase and the encryption algorithm.
- The **SEED** statement defines a seed phrase that generates the actual encryption/decryption key. You may define have up to 256 key phrases.
- The **END** statement must be the last statement coded.

The DRCRYPTO Macro is in the Dr. D Sublibrary as DRCRYPTO.A. Observe standard macro conventions when coding.

BEGIN Statement

The format for the BEGIN statement is:

```
name DRCRYPTO BEGIN ,
      DEFLTCI=ACSDDES | ACSDES3 | ACSAES |
      DEFLTKEY=n
```

Where:

name is the name of the resulting phase. This name is used as the *name* on the ENCRYPTKEY operand to be used on the SAVE or RESTORE control statement. This operand is required.

BEGIN defines this statement as the BEGIN statement. This operand is required.

DEFLTCI defines this encryption algorithm to be used. The following table lists the valid values for this parameter are as follows:

Value	Description
ACSDDES	Use the DES algorithm
ACSDES3	Use the DES3 algorithm
ACSAES	Use the AES algorithm

This operand is optional. The default is DEFLTCI=ACSDDES.

DEFLTKEY defines the default key phrase to use when the ENCRYPTKEYS operand does not specify a key number. This key equates to the value associated with the KEY operand of the DRCRYPTO SEED statement. This operand is optional. The default is DEFLTKEY=1.

SEED Statement

The format for the SEED statement(s) is:

```
DRCRYPTO  SEED ,  
           KEY=n,  
           PHRASE='character string',  
           XPHRAS='hexadecimal string'
```

Where:

SEED	defines this statement as an encryption/decryption seed phrase statement. This operand is required.
KEY	defines an identifier for the seed phrase. The value coded for this operand must be any number from 1 – 256. You may use this key value on the ENCRYPTKEYS operand to specify the seed phrase used to equate the DEFLTKEY operand on the BEGIN statement when denoting the default phrase. This operand is required.
PHRASE	specifies a seed phrase that generates the actual encryption/decryption password. Specify a character string up to 253 characters enclosed in single quotes. This operand is mutually exclusive with the XPHRAS operand, but either PHRASE or XPHRAS must be specified on each SEED statement.
XPHRAS	specifies a seed phrase that generates the actual encryption/decryption password. Specify a hexadecimal string up to 253 characters enclosed in single quotes. This operand is mutually exclusive with the PHRASE operand, but either PHRASE or XPHRAS must be specified on each SEED statement.

END Statement

The format for the END statement is:

```
DRCRYPTO  END
```

Example: Creating and Using DRCRYPTO Seed Phrase

This example creates 4 possible seed phrases. The first statement is the BEGIN statement, which indicates the following:

- Use of the AES encryption cipher
- The seed phrase to use when the ENCRYPTKEYS operand on a SAVE or RESTORE command has no specified SEED KEY. In this example, DEFLTKY=4 refers to the DRCRYPTO SEED,KEY=4 statement.

Once the generated phase is link-edited, it is cataloged with the phase name SEEDSAMP.

```
// LIBDEF PHASE,CATALOG=lib.sublib
// LIBDEF SOURCE,SEARCH=lib.sublib
// OPTION CATAL
// EXEC ASSEMBLY,SIZE=200K
SEEDSAMP DRCRYPTO  BEGIN,DEFLTCTI=ACSAES,DEFLTKY=4
          DRCRYPTO  SEED,KEY=1,                      X
                    PHRASE='A PHRASE FOR SENDING TAPES TO COMPANY A'
          DRCRYPTO  SEED,KEY=2,                      X
                    PHRASE='ANOTHER PHRASE COULD BE ANOTHER COMPANY'
          DRCRYPTO  SEED,KEY=3,                      X
                    PHRASE='THIS COULD BE A PHASE WHEN TESTING THE ACSAES ENX
                    CRYPTION ALGORITHM'
          DRCRYPTO  SEED,KEY=4,                      X
                    XPHRAS='04222594A8E0C4C58681E4D3A3A1C9D58885A7'
          DRCRYPTO  END
          END
/*
// EXEC LNKEDT
/*
```

Once the above DRCRYPTO phase has been cataloged, it has the name SEEDSAMP.

Example: Using DRCRYPTO Default Seed Phrase

This Dr. D example uses the SEEDSAMP phase that the DRCRYPTO Macro generated in the previous example. It shows how to set up Dr. D to encrypt the data during a SAVE and how to decrypt it during a RESTORE. The example uses the following default seed phrase from the DRCRYPTO Macro:

```
DRCRYPTO SEED,KEY=4,XPHRAS='04222594A8E0C4C58681E4D3A3A1C9D58885A7'
```

Adding the ENCRYPTKEYS operand to the SAVE and RESTORE commands identifies the name of the DRCRYPTO phase. In the RESTORE function, ENCRYPTKEYS= is abbreviated to EN=.

```
* Backup UCAT1 to an encrypted tape
// ASSGN SYS007, cuu
// TLBL TAPEOUT, 'UCAT1.BACKUP'
// DLBL UCAT1, 'VSAM.USER.CATALOG.ONE' , , VSAM
// EXEC DRD, SIZE=300K
SAVE=LOGICAL, SC=UCAT1, ASSOCIATIONS=YES, ENCRYPTKEYS=SEEDSAMP
SAVE=NO, TAPE=REW
/*
* Restore UCAT1 from an encrypted tape
// ASSGN SYS008, cuu
// TLBL TAPEIN, 'UCAT1.BACKUP'
// DLBL UCAT1, 'VSAM.USER.CATALOG.ONE' , , VSAM
// EXEC DRD, SIZE=300K
RESTORE=NO, TAPE=REW
RESTORE=LOGICAL, SC=UCAT1, RC=UCAT1, FD=D, EN=SEEDSAMP
RESTORE=NO, TAPE=REW
/*
```

Example: Using DRCRYPTO with Specific Seed Phrase

This Dr. D example uses the SEEDSAMP phase that the DRCRYPTO Macro generated in the earlier example above. It shows how to set up Dr. D to encrypt the data during a SAVE and how to decrypt it during a RESTORE. The example uses the following seed phrase from the DRCRYPTO Macro:

```
DRCRYPTO SEED,KEY=2,PHRASE='ANOTHER PHRASE COULD BE ANOTHER COMPANY'
```

Adding the ENCRYPTKEYS operand to the SAVE and RESTORE functions identifies the name of the DRCRYPTO phase. In the RESTORE function, ENCRYPTKEYS= is abbreviated to EN=.

```
* Backup UCAT1 to an encrypted tape
// ASSGN SYS007,cuu
// TLBL TAPEOUT,'UCAT1.BACKUP'
// DLBL UCAT1,'VSAM.USER.CATALOG.ONE',,VSAM
// EXEC DRD,SIZE=300K
SAVE=LOGICAL,SC=UCAT1,ASSOCIATIONS=YES,ENCRYPTKEYS=SEEDSAMP/2
SAVE=NO,TAPE=REW
/*
* Restore UCAT1 from an encrypted tape
// ASSGN SYS008,cuu
// TLBL TAPEIN,'UCAT1.BACKUP'
// DLBL UCAT1,'VSAM.USER.CATALOG.ONE',,VSAM
// EXEC DRD,SIZE=300K
RESTORE=NO,TAPE=REW
RESTORE=LOGICAL,SC=UCAT1,RC=UCAT1,FD=D,EN=SEEDSAMP/2
RESTORE=NO,TAPE=REW
/*
```

Optional BIM-EPIC Implementation Method

Dr. Crypto has a built-in interface with BIM-EPIC releases 7.1 and above that allows Dr. Crypto to run “behind the scenes”. When using this interface for implementation, the Dr. D job streams require no changes. Simply set up your SAVE and RESTORE job streams according to the “Dr. D User Guide”.

In BIM-EPIC, you must add encryption information to the appropriate dataset definition(s) (EDD) by performing the following steps.

Step	Action
1	<p>You must associate the EDD with a dataset password by cataloging/updating the EDD with the PWD operand.</p> <p>The following example shows how to catalog an EDD:</p> <pre data-bbox="560 766 1166 793">CAT 'ENCRYPT.BACKUP',CYC=1,PWD=ENCRYPT</pre>
2	<p>Update the EDD to specify the encryption algorithm. Use the ENC operand, which is valid only with the UPDATE command. Valid ENC values include:</p> <ul data-bbox="516 961 1230 1094" style="list-style-type: none"> • DES for the DES algorithm • DE3 for the DES3 algorithm • AES for the AES algorithm, which requires the hardware crypto-assist feature <p>The ENC operand requires the DPW operand to specify the dataset password, which must match the PWD operand password. Non-matching passwords trigger an EP112 PASSWORD VIOLATION and cancels the job.</p> <p>This example shows how to update an EDD with a DES3 algorithm:</p> <pre data-bbox="560 1360 1198 1388">UPD 'ENCRYPT.BACKUP',DPW=ENCRYPT,ENC=DE3</pre>

Once the EDD has been updated with the encryption information in the BIM-EPIC catalog, the output is encrypted according to the defined encryption algorithm each time Dr. D uses that EDD for output.

Example: Adding Encryption Information

This example shows how to define an EDD to use the DES encryption algorithm. It then shows how to backup all the files in the UCAT1 VSAM catalog.

```
* Define EDD for DES Encryption
// EXEC TSIDMNT
CAT 'UCAT1.BACKUP',CYC=3,PWD=ENCRYPT
UPD 'UCAT1.BACKUP',DPW=ENCRYPT,ENC=DES
/*
* Backup UCAT1 - Unload tape at close
// TLBL TAPEOUT,'UCAT1.BACKUP',,,,,,2
// DLBL UCAT1,'VSAM.USER.CATALOG.ONE',,VSAM
// EXEC DRD,SIZE=200K
SAVE=LOGICAL,SC=UCAT1,ASSOCIATIONS=YES
/*
```

Messages

Dr. Crypto issues the following printer and console messages. Each message includes the probable reason for the message, any recommended action, and the disposition. The messages are listed by the three-digit DOCTOR number.

DOCTOR900 INSUFFICIENT GETVIS RC=xx

Explanation: Insufficient getvis memory below the line is available to satisfy a GETVIS request. The job is canceled (RC16).

The value *xx* is the return code from the GETVIS macro.

Action: Resubmit in a larger partition.

DOCTOR901 INTERNAL ERROR RC=xx

Explanation: An internal error has been detected. The job is canceled (RC16).

- 03 Invalid parameter list
- 04 Encryption type unsupported
- 05 Input buffer too large to encrypt

Action: Contact CSI technical support after gathering the JCL, syslst, DRZAP SUM report, and a display of the save file dataset name from the BIM-EPIC catalog.

DOCTOR902 DECRYPTION FAILURE RC=-xx

Explanation: An error was detected during the decryption process.

The value *xx* is the return code from the failing routine.

Action: Contact CSI technical support after gathering the JCL, syslst, DRZAP SUM report, and a display of the input save file dataset name from the BIM-EPIC catalog.

DOCTOR903 ENCRYPTION FAILURE RC=-xx

Explanation: An error was detected during the encryption process.

The value *xx* is the return code from the failing routine.

Action: Contact CSI technical support after gathering the JCL, syslst, DRZAP SUM report, and a display of the save file dataset name from the BIM-EPIC catalog.

DOCTOR904 KEY INIT FAILURE RC=-xx

Explanation: An error was detected during the creation of an encryption/decryption key.

The value *xx* is the return code from the failing routine

Action: Insure the modules for SSL for VSE release 1.5E or higher are in the LIBDEF search chain. If they are, contact CSI technical support after gathering JCL, syslst, DRZAP SUM report, DRCRYPTO assembly source and a LIBR LIST of the seed phase that was generated.

DOCTOR905 KEY NOT FOUND

Explanation: The specific key requested by the ENCRYPTKEYS operand or the default key was not found in seed phrase module created by the DRCRYPTO Macro assembly.

Action: Insure the proper DRCRYPTO seed phrase is in the search chain and that the proper key is being requested. If it is, contact CSI technical support after gathering JCL, syslst, DRZAP SUM report, DRCRYPTO assembly source, and a LIBR LIST of the seed phase that was generated.

DOCTOR906 SSL INIT FAILURE RC=x RE=-yy

Explanation: An error was detected during the encryption process.

The value *x* is the return code from the failing routine. Valid values can be any of the following:

- 1 Invalid DSN name
- 2 Invalid EXTENT list
- 4 SSL routine error

If RC=4, the RE=-yyy will indicate the SSL reason code.

Action: Insure the modules for SSL for VSE release 1.5E or higher are in the LIBDEF search chain. If they are, contact CSI technical support after gathering JCL, syslst, DRZAP SUM report and a display of the save file dataset name from the BIM-EPIC catalog.

DOCTOR907 Dr. Crypto v.r ACTIVATED

Explanation: Informational message when Dr. Crypto has been activated.

- The value *v* is the version number.
- The value *r* is the release level.

Action: None.

DOCTOR908 DECRYPTUION INFORMATION NOT SUPPLIED

Explanation: The input save file is encrypted but it cannot be read because the cipher key information was not supplied. .

Action: Rerun with the appropriate ENCRYPTKEYS value

Index

- .Specifying the encryption algorithm, 10
- Adding encryption information, 10
 - example, 11
- Advanced Encryption Standard, 1
- Cataloging an EDD, 10
- Data Encryption Algorithms
 - Description, 1
- Data Encryption Standard, 1
- Details about Dr. Crypto Usage, 1
- DRCRYPTO Macro, 4
 - BEGIN Statement, 5
 - SEED Statement, 6
 - Creating and Using Example, 7
 - Using Default Example, 8
 - Using with Specific Seed Phrase Example, 9
- Implementing Dr. Crypto, 4
 - Native Implementation Method, 4
 - Optional BIM-EPIC Implementation Method, 10
- Installing Dr. Crypto, 3
- Messages, 12
- Native Implementation Method, 4
- Optional BIM-EPIC Implementation Method, 10
- Overview, 1
- Triple Data Encryption Standard, 1