



# UniVerse

## **10.2 New Features**

Version 10.2  
September, 2006

IBM Corporation  
555 Bailey Avenue  
San Jose, CA 95141

Licensed Materials – Property of IBM

© Copyright International Business Machines Corporation 2006. All rights reserved.

AIX, DB2, DB2 Universal Database, Distributed Relational Database Architecture, NUMA-Q, OS/2, OS/390, and OS/400, IBM Informix®, C-ISAM®, Foundation.2000™, IBM Informix® 4GL, IBM Informix® DataBlade® module, Client SDK™, Cloudscape™, Cloudsync™, IBM Informix® Connect, IBM Informix® Driver for JDBC, Dynamic Connect™, IBM Informix® Dynamic Scalable Architecture™ (DSA), IBM Informix® Dynamic Server™, IBM Informix® Enterprise Gateway Manager (Enterprise Gateway Manager), IBM Informix® Extended Parallel Server™, i.Financial Services™, J/Foundation™, MaxConnect™, Object Translator™, Red Brick® Decision Server™, IBM Informix® SE, IBM Informix® SQL, InformiXML™, RedBack®, SystemBuilder™, U2™, UniData®, UniVerse®, wIntegrate® are trademarks or registered trademarks of International Business Machines Corporation.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

Windows, Windows NT, and Excel are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names used in this publication may be trademarks or service marks of others.

This product includes cryptographic software written by Eric Young (eay@cryptosoft.com).

This product includes software written by Tim Hudson (tjh@cryptosoft.com).

Documentation Team: Claire Gustafson, Shelley Thompson

US GOVERNMENT USERS RESTRICTED RIGHTS

Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

# Table of Contents

<b>Chapter 1</b>	<b>Transaction Logging Enhancements</b>	
	UniVerse Command Enhancements . . . . .	1-3
	ENABLE.RECOVERY . . . . .	1-3
	SHUTDOWN.RECOVERY . . . . .	1-4
	SUSPEND.RECOVERY . . . . .	1-5
	RECOVER.FILE . . . . .	1-6
	File-level Operations . . . . .	1-9
	New uvconfig Parameter . . . . .	1-10
	System Administration Menu Changes . . . . .	1-11
	Media Recovery Rollforward Screen . . . . .	1-11
	Media Recovery Rollforward From Tape Screen . . . . .	1-13
	Rollforward a File Screen . . . . .	1-15
	Rollforward a File From Tape Screen . . . . .	1-19
 <b>Chapter 2</b>	 <b>Automatic Data Encryption</b>	
	Encrypted File Types . . . . .	2-3
	Encryption With UniVerse Replication . . . . .	2-3
	Key Store . . . . .	2-4
	How Encryption Works . . . . .	2-5
	Defining a Master Key . . . . .	2-7
	Changing a Master Key After Data is Encrypted . . . . .	2-7
	UniVerse Encryption Algorithms . . . . .	2-8
	Encryption Commands . . . . .	2-9
	CREATE.ENCRYPTION.KEY . . . . .	2-9
	DELETE.ENCRYPTION.KEY . . . . .	2-9
	LIST.ENCRYPTION.KEY . . . . .	2-10
	GRANT.ENCRYPTION.KEY . . . . .	2-10
	REVOKE.ENCRYPTION.KEY . . . . .	2-11
	ENCRYPT.FILE . . . . .	2-12
	DECRYPT.FILE . . . . .	2-16

LIST.ENCRYPTION.FILE . . . . .	2-21
ACTIVATE.ENCRYPTION.KEY . . . . .	2-21
DEACTIVATE.ENCRYPTION.KEY . . . . .	2-22
DISABLE.DECRYPTION . . . . .	2-22
ENABLE.ENCRYPTION . . . . .	2-23
UniVerse BASIC Encryption Commands . . . . .	2-24
ACTIVATEKEY . . . . .	2-24
DEACTIVATEKEY . . . . .	2-24
DISABLEDEC . . . . .	2-25
ENABLEDEC . . . . .	2-26
STATUS Function Changes . . . . .	2-26
The encman Utility . . . . .	2-28
Viewing Audit Trail Information . . . . .	2-28
Generating a Key Store . . . . .	2-29
Deleting the Key Store . . . . .	2-30

## Chapter 3      **IBM U2 Web Services Developer**

IBM U2 Web Services Developer. . . . .	3-2
Minimum Requirements . . . . .	3-2

## Chapter 4      **Licensing Changes in UniVerse 10.2**

Authorizing UniVerse . . . . .	4-4
Authorizing a New UniVerse License . . . . .	4-5
Obtain Configuration Code . . . . .	4-6
Obtain Authorization Code . . . . .	4-6

---

# Transaction Logging Enhancements

UniVerse Command Enhancements . . . . .	1-3
ENABLE.RECOVERY . . . . .	1-3
SHUTDOWN.RECOVERY . . . . .	1-4
SUSPEND.RECOVERY . . . . .	1-5
RECOVER.FILE . . . . .	1-6
File-level Operations . . . . .	1-9
New uvconfig Parameter . . . . .	1-10
System Administration Menu Changes . . . . .	1-11
Media Recovery Rollforward Screen . . . . .	1-11
Media Recovery Rollforward From Tape Screen . . . . .	1-13
Rollforward a File Screen . . . . .	1-15
Rollforward a File From Tape Screen . . . . .	1-19

This chapter describes enhancements to Transaction Logging. These enhancements include changes to UniVerse BASIC commands, UniVerse commands, the UniVerse System Administration menu, and UniAdmin.

---

# UniVerse Command Enhancements

This section describes enhancements to UniVerse commands.

## ENABLE.RECOVERY

Use ENABLE.RECOVERY to enable the transaction logging system. You must be a UniVerse Administrator logged in to the UV account to use this command.

### *Syntax*

**ENABLE.RECOVERY** { YES | NO } { INFORM }

### *Parameters*

The following table describes each parameter of the syntax.

Parameter	Description
YES	Retains the current logging info file. This is the default.
NO	Removes the current logging info file.
INFORM	Displays messages during the startup process.

**ENABLE.RECOVERY Parameters**

### *Description*

ENABLE.RECOVERY starts the log daemon *uvlogd*.

You should fully back up your UniVerse files before enabling transaction logging for the first time with ENABLE.RECOVERY or after SHUTDOWN.RECOVERY.

## Example

This example starts the log daemon and retains the current logging info file *uvlogd.info*:

```
>ENABLE.RECOVERY YES
Request to Enable Logging Subsystem made at 12:51:52 on 01 OCT
1996. You can use the 'Display logging state' menu to verify the
current state of the logging subsystem.
```

## SHUTDOWN.RECOVERY

Use SHUTDOWN.RECOVERY to disable the transaction logging system if it is enabled or suspended. You must be a UniVerse Administrator logged in to the UV account to use SHUTDOWN.RECOVERY.

### Syntax

#### SHUTDOWN.RECOVERY { INFORM }

The following table describes the parameter of the syntax.

Parameter	Description
INFORM	Displays messages during the shutdown process.

#### SHUTDOWN.RECOVERY Parameter



**Warning:** Because updates to recoverable files are not logged after you execute *SHUTDOWN.RECOVERY*, the point of disablement is the latest point to which you can consistently recover your UniVerse files.

If transaction logging is currently full or suspended, SHUTDOWN.RECOVERY starts the log daemon to disable logging properly. While transaction logging is in the disabled state, programs that request writes to the log file fail.

## SUSPEND.RECOVERY

Use SUSPEND.RECOVERY to suspend the transaction logging system if it is currently enabled. You must be a UniVerse Administrator logged in to the UV account to use SUSPEND.RECOVERY.



*Syntax*

**SUSPEND.RECOVERY { INFORM }**

The following table describes the parameter of the syntax.

Parameter	Description
INFORM	Displays messages during the suspend process.

**SUSPEND.RECOVERY Parameter**

*Description*

SUSPEND.RECOVERY suspends the transaction logging system. In this state, updates to recoverable files are prohibited, except by means of the roll-forward utility, and any attempt to update a recoverable file waits until the state changes. You can reenale transaction logging with ENABLE.RECOVERY.

While transaction logging is in the suspended state, programs that request writes to the log file wait until the state changes.

*Example*

This example suspends transaction logging:

```
>SUSPEND.RECOVERY  
Request to Suspended Logging Subsystem made at 12:47:59 on 01 OCT  
1996. You can use the 'Display logging state' menu to verify the  
current state of the logging subsystem.
```

**RECOVER.FILE**

Use the RECOVER.FILE command to recover files activated for Transaction Logging. You must be a UniVerse Administrator logged on to the UV account to use this command.

*Syntax*

**RECOVER.FILE** [-F *file\_path*] [-M [*list\_name* | -A]] [-C | -U *starting\_log\_number*  
*ending\_log\_number*] [-S | -R] [-I *record\_ID* | -N *record\_list\_name*] [-L *log\_path*] [-V  
*verbosity*] [-B *start\_time*] [-E *end\_time*] [-T *tape\_device*]

## Description

Use the RECOVER.FILE command to start the roll forward process.

You should use the RECOVER.FILE command only when the system is not in use. Use the SUSPEND.RECOVERY command to suspend transaction logging, or SHUTDOWN.RECOVERY to stop transaction logging.

## Parameters

The following table describes each parameter of the syntax.

Parameter	Description
<i>-F file_path</i>	Specifies a single file to recover. You must specify fully-qualified path, and activate the file to recover. Use a space between the -F option and <i>file_path</i> . Do not use with the -M option.
<i>-M list_name</i>   <i>-A</i>	Specifies a <i>list_name</i> stored in the &SAVEDLISTS& file. If you do not specify a list name, you must specify -A to specify all active files. Use a space between the -M parameter and <i>list_name</i> , or the -M parameter and -A. Do not use with -F option.
<i>-C starting_log_number ending_log_number</i>	Instructs the roll forward process to begin processing at <i>starting_log_number</i> and stop at the end of <i>ending_log_number</i> . During the recovery process, UniVerse verifies the log numbers. Use a space between the -C and <i>starting_log_number</i> and <i>ending_log_number</i> . Do not use with the -U option.
<i>-U starting_log_number ending_log_number</i>	Instructs the roll forward process to begin processing at <i>starting_log_number</i> and stop at the end of <i>ending_log_number</i> . During the recovery process, UniVerse does not verify the log numbers. Use a space between the -C and <i>starting_log_number</i> and <i>ending_log_number</i> . Do not use with the -C option.
<i>-S</i>	Specifies to output messages to the screen. If you do not specify -S, UniVerse writes output messages to the uvrolf.info file. Do not use with the -R option.
<i>-R</i>	Specifies to retain the uvrolf.info file. UniVerse appends output messages to the existing uvrolf.info file. Do not use with the -S option.

### RECOVER.FILE Parameters

Parameter	Description
-I <i>record_ID</i>	Specifies a single record for UniVerse to recover. The record must exist in the file you specify with the -F option. Use a space between the -I parameter and <i>record_ID</i> . If you do not specify a record ID, UniVerse displays an error and the program terminates. Do not use with the -N option.
-N <i>record_list_name</i>	Specifies a list of records stored in <i>record_list_name</i> in the &SAVEDLISTS& file to recover. UniVerse recovers the records from the file you specify with the -F option. Use a space between the -N parameter and <i>record_list_name</i> . If you do not specify <i>record_list_name</i> , UniVerse displays an error and the program terminates. Do not use with the -I option.
-L <i>log_path</i>	Specifies the directory for log_path. If you do not specify the -L option, UniVerse uses the default log path. Do not enter a space between -L and <i>log_path</i> .
-B <i>start_time</i>	Specifies the starting date and time within <i>starting_log_number</i> you specify with the -C or -U option from which to begin restoring the logs. Enter <i>start_time</i> in the yyyy-mm-ddThh:mm:ss or UTC format. The hour must be 00 - 23. Use a space between -B and <i>start_time</i> .
-E <i>end_time</i>	Specifies the ending date and time within <i>ending_log_number</i> you specify with the -C or -U parameter to stop restoring the logs. Enter <i>end_time</i> in the yyyy-mm-ddThh:mm:ss or UTC format. The hour must be 00 - 23. Use a space between -E and <i>end_time</i> .
-V <i>level</i>	Specifies the amount of messaging to output by the rollforward process. <i>level</i> must be 0, 1, 2, or 3. 0 is the minimal message output, while 3 is the maximum message output. If you do not specify <i>level</i> , UniVerse displays an error and the process terminates. Do not enter a space between -V and <i>level</i> .
-T <i>device</i>	Specifies the logs to recover that exist on <i>device</i> . <i>device</i> must exist in the &DEVICE& file. To specify multiple tape devices, use a separate -T option followed by <i>device</i> . You must enter a space between -T and <i>device</i> . Do not use with the -L option.

---

**RECOVER.FILE Parameters (Continued)**

---

---

## File-level Operations

Beginning at UniVerse 10.2, the following file-level operations are now logged:

- `CLEAR.FILE` (on a recoverable file)
- `CREATE.FILE` (when you specify the `RECOVERABLE` keyword)
- `DELETE.FILE` (on a recoverable file)
- `CNAME` (on a recoverable file)
- `RESIZE` (of a recoverable file)

In addition, the rollforward process has been enhanced to perform these file-level operations.

---

## New uvconfig Parameter

Prior to UniVerse 10.2, when the transaction data reached the physical log file size, the log file was marked as full, and UniVerse opened the next available log file for processing.

Beginning at UniVerse 10.2, you can specify a time frame in which to close the log file and move to the next available log file, even if the log file is not full. You define this time frame the UVLOGSWITCH uvconfig parameter.

The UVLOGSWITCH parameter determines the time, in seconds, that the log file forces a switch to the next available log file if the current log file does not fill during the interval you specify. If you set this value to 0, UniVerse does not switch to the next log file until the current log file is full.

**Note:** *If the amount of time you specify expires and no logging activity has occurred (the log file is empty), UniVerse resets the timer for the currently empty log file, ensuring that a completely empty log file is never marked as full.*

---

## System Administration Menu Changes

This section describes changes to the Media Recovery Rollforward screen and the Media Recovery Rollforward from Tape screen.

### Media Recovery Rollforward Screen

The following example illustrates the Media Recovery Rollforward screen:

```
+----- Media Recovery Rollforward -----+
|                                     |
|      File              Action      Help      |
|-----|-----|-----|-----|
| File path or name of select list  ?      |
| Logging Directory                  : /stanley1/uv102/tlogs |
| First Log Number To Use           : 1      |
| Last Log Number To Use            : 1      |
| Verify Log Numbers                 : NO     |
| Verbosity Level (0, 1, 2, 3)      : 0      |
| Start time (YYYY-MM-DDThh:mm:ss)  :         |
| End time   (YYYY-MM-DDThh:mm:ss)  :         |
| Single record ID                   :         |
| Record ID list name                :         |
|                                     |
|                                     |
|----- Help Region -----|
| Press <F1> for longer help about any particular entry, or <ESCAPE> to exit |
| PROGRAM                             |
| Enter file path, name of list or ALL for all files in UV.TRANS file         |
|                                     |
+-----+-----+-----+-----+
```

The following table describes the Media Recovery Rollforward fields:

Field	Description
File path or name of select list	Enter the fully-qualified path to a single file to recover, or the name of a saved list that exists in the &SAVEDLISTS& file in the UV account.
Logging Directory	Enter the path where the transaction log files reside. UniVerse displays the default path, if any.
First Log Number to Use	Enter the number of the oldest log file to roll forward. If you want the transaction logging subsystem to locate the oldest log file, enter 0.

---

#### Media Recovery Rollfoward Fields

Field	Description
Last Log Number to Use	Enter the number of the newest log file to roll forward. If you want the transaction logging subsystem to locate the newest log file, enter 0.
Verify Log Numbers	Enter YES if you want the roll forward process to verify the log numbers. Enter NO if you do not require verification. The default is NO.
Verbosity Level	The verbosity level determines the amount of information to record in the roll forward information file. The highest verbosity level is 3, with the lowest being 0. The default value is 0.
Start Time	Enter the starting date and time in the universal time code (UTC) format, or in the <i>yyyy-mm-ddThh:mm:ss</i> format. This setting instructs the roll forward process to apply log records from the date and time you specify in the first log number. To roll forward from the beginning of the first log file, leave this field blank.
End Time	Enter the ending date and time in the universal time code (UTC) format, or in the <i>yyyy-mm-ddThh:mm:ss</i> format. This setting instructs the roll forward process to stop applying log records from the date and time you specify in the last log number. To roll forward to the end of the last log file, leave this field blank.
Single Record ID	Enter the record name to roll forward. UniVerse uses this entry to restore a specific record from a single file path you specify in the File path field. You cannot use this option if you do not specify a single file path in the File path field. If you want to roll forward more than one record, leave this field blank.
Record ID list name	Enter the name of saved list of records to restore. This list name must exist in the &SAVEDLIST& file. UniVerse ignores any records contained in the saved list that are not in the file path you specify in the File path field. This option is unavailable if you specify a Single Record ID. If you want to restore all records in one or more files, leave this field blank.

#### Media Recovery Rollforward Fields (Continued)

## Media Recovery Rollforward From Tape Screen

The following example illustrates the Media Recovery Rollforward from Tape Screen:

File	Action	Help
File path or name of select list	?	
Device list to restore from	:	
First Log Number To Use	: 1	
Last Log Number To Use	: 1	
Verify Log Numbers	: NO	
Verbosity Level (0, 1, 2, 3)	: 0	
Start time (yyyy-mm-ddThh:mm:ss)	:	
End time (yyyy-mm-ddThh:mm:ss)	:	
Single record ID	:	
Record ID list name	:	

Help Region

Press <F1> for longer help about any particular entry, or <ESCAPE> to exit

PROGRAM

Enter file path, name of list or ALL for all files in UV.TRANS file

The following table describes each field of the screen::

Field	Description
File path or name of select list	Enter the fully-qualified path to a single file to recover, or the name of a saved list that exists in the &SAVEDLISTS& file in the UV account.
Device list to restore from	Enter one or more device names, separated by spaces. The devices must exist in the &DEVICE& file.
First Log Number to Use	Enter the number of the oldest log file to roll forward. If you want the transaction logging subsystem to locate the oldest log file, enter 0.
Last Log Number to Use	Enter the number of the newest log file to roll forward. If you want the transaction logging subsystem to locate the newest log file, enter 0.
Verify Log Numbers	Enter YES if you want the roll forward process to verify the log numbers. Enter NO if you do not require verification. The default is NO.

### Media Recovery Rollforward From Tape Fields



Field	Description
Verbosity Level	The verbosity level determines the amount of information to record in the roll forward information file. The highest verbosity level is 3, with the lowest being 0. The default value is 0.
Start Time	Enter the starting date and time in the universal time code (UTC) format, or in the <i>yyyy-mm-ddThh:mm:ss</i> format. This setting instructs the roll forward process to apply log records from the date and time you specify in the first log number. To roll forward from the beginning of the first log file, leave this field blank.
End Time	Enter the ending date and time in the universal time code (UTC) format, or in the <i>yyyy-mm-ddThh:mm:ss</i> format. This setting instructs the roll forward process to stop applying log records from the date and time you specify in the last log number. To roll forward fto the end of the last log file, leave this field blank.
Single Record ID	Enter the record name to roll forward. UniVerse uses this entry to restore a specific record from a single file path you specify in the File path field. You cannot use this option if you do not specify a single file path in the File path field. If you want to roll forward more than one record, leave this field blank.
Record ID list name	Enter the name of saved list of records to restore. This list name must exist in the &SAVEDLIST& file. UniVerse ignores any records contained in the saved list that are not in the file path you specify in the File path field. This option is unavailable if you specify a Single Record ID. If you want to restore all records in one or more files, leave this field blank.

---

**Media Recovery Rollfoward From Tape Fields (Continued)**

---

# Rollforward a File Screen

The following example illustrates the Rollforward a File Screen:

```

+----- Rollforward a File -----+
| File Action Help |
+-----+-----+-----+
| File path or name of select list ? |
| Logging Directory : /stanley1/uv102/tlogs |
| First Log Number To Use : 1 |
| Last Log Number To Use : 1 |
| Verify Log Numbers : NO |
| Verbosity Level (0, 1, 2, 3) : 0 |
| Start time (yyyy-mm-ddThh:mm:ss) : |
| End time (yyyy-mm-ddThh:mm:ss) : |
| Single record ID : |
| Record ID list name : |
|
|
+----- Help Region -----+
| Press <F1> for longer help about any particular entry, or <ESCAPE> to exit |
| PROGRAM |
| Enter file path, name of list or ALL for all files in UV.TRANS file |
|
|
+-----+

```

The following table describes each field in the Rollward a File screen::

Field	Description
File path or name of select list	Enter the fully-qualified path to a single file to recover, or the name of a saved list that exists in the &SAVEDLISTS& file in the UV account.
Logging Directory	Enter the path where the transaction log files reside. UniVerse displays the default path, if any.
First Log Number to Use	Enter the number of the oldest log file to roll forward. If you want the transaction logging subsystem to locate the oldest log file, enter 0.
Last Log Number to Use	Enter the number of the newest log file to roll forward. If you want the transaction logging subsystem to locate the newest log file, enter 0.
Verify Log Numbers	Enter YES if you want the roll forward process to verify the log numbers. Enter NO if you do not require verification. The default is NO.

## Rollforward a File Fields

Field	Description
Verbosity Level	The verbosity level determines the amount of information to record in the roll forward information file. The highest verbosity level is 3, with the lowest being 0. The default value is 0.
Start Time	Enter the starting date and time in the universal time code (UTC) format, or in the <i>yyyy-mm-ddThh:mm:ss</i> format. This setting instructs the roll forward process to apply log records from the date and time you specify in the first log number. To roll forward from the beginning of the first log file, leave this field blank.
End Time	Enter the ending date and time in the universal time code (UTC) format, or in the <i>yyyy-mm-ddThh:mm:ss</i> format. This setting instructs the roll forward process to stop applying log records from the date and time you specify in the last log number. To roll forward fto the end of the last log file, leave this field blank.
Single Record ID	Enter the record name to roll forward. UniVerse uses this entry to restore a specific record from a single file path you specify in the File path field. You cannot use this option if you do not specify a single file path in the File path field. If you want to roll forward more than one record, leave this field blank.
Record ID list name	Enter the name of saved list of records to restore. This list name must exist in the &SAVEDLIST& file. UniVerse ignores any records contained in the saved list that are not in the file path you specify in the File path field. This option is unavailable if you specify a Single Record ID. If you want to restore all records in one or more files, leave this field blank.

---

**Rollforward a File Fields (Continued)**

The following table describes each field in the Rollforward a File From Tape screen::

Field	Description
File path or name of select list	Enter the fully-qualified path to a single file to recover, or the name of a saved list that exists in the &SAVEDLISTS& file in the UV account.
Device list to restore from	Enter one or more device names separated by spaces. An entry for each device must exist in the &DEVICE& file.
First Log Number to Use	Enter the number of the oldest log file to roll forward. If you want the transaction logging subsystem to locate the oldest log file, enter 0.
Last Log Number to Use	Enter the number of the newest log file to roll forward. If you want the transaction logging subsystem to locate the newest log file, enter 0.
Verify Log Numbers	Enter YES if you want the roll forward process to verify the log numbers. Enter NO if you do not require verification. The default is NO.
Verbosity Level	The verbosity level determines the amount of information to record in the roll forward information file. The highest verbosity level is 3, with the lowest being 0. The default value is 0.
Start Time	Enter the starting date and time in the universal time code (UTC) format, or in the yyyy-mm-ddThh:mm:ss format. This setting instructs the roll forward process to apply log records from the date and time you specify in the first log number. To roll forward from the beginning of the first log file, leave this field blank.
<b>Rollforward a File From Tape Fields</b>	

Field	Description
End Time	Enter the ending date and time in the universal time code (UTC) format, or in the <i>yyyy-mm-ddThh:mm:ss</i> format. This setting instructs the roll forward process to stop applying log records from the date and time you specify in the last log number. To roll forward fto the end of the last log file, leave this field blank.
Single Record ID	Enter the record name to roll forward. UniVerse uses this entry to restore a specific record from a single file path you specify in the File path field. You cannot use this option if you do not specify a single file path in the File path field. If you want to roll forward more than one record, leave this field blank.
Record ID list name	Enter the name of saved list of records to restore. This list name must exist in the &SAVEDLIST& file. UniVerse ignores any records contained in the saved list that are not in the file path you specify in the File path field. This option is unavailable if you specify a Single Record ID. If you want to restore all records in one or more files, leave this field blank.

---

**Rollforward a File From Tape Fields (Continued)**

## Rollforward a File From Tape Screen

The following example illustrates the Rollforward a File From Tape screen:

File	Action	Help
File path or name of select list	?	
Device list to restore from	:	
First Log Number To Use	: 1	
Last Log Number To Use	: 1	
Verify Log Numbers	: NO	
Verbosity Level (0, 1, 2, 3)	: 0	
Start time (yyyy-mm-ddThh:mm:ss)	:	
End time (yyyy-mm-ddThh:mm:ss)	:	
Single record ID	:	
Record ID list name	:	

Help Region
Press <F1> for longer help about any particular entry, or <ESCAPE> to exit
PROGRAM
Enter file path, name of list or ALL for all files in UV.TRANS file

# Automatic Data Encryption

Encrypted File Types . . . . .	2-4
Encryption With UniVerse Replication . . . . .	2-4
Key Store . . . . .	2-5
How Encryption Works. . . . .	2-6
Defining a Master Key . . . . .	2-8
Changing a Master Key After Data is Encrypted . . . . .	2-8
UniVerse Encryption Algorithms. . . . .	2-9
Encryption Commands . . . . .	2-10
CREATE.ENCRIPTION.KEY . . . . .	2-10
DELETE.ENCRIPTION.KEY . . . . .	2-10
LIST.ENCRIPTION.KEY . . . . .	2-11
GRANT.ENCRIPTION.KEY . . . . .	2-11
REVOKE.ENCRIPTION.KEY . . . . .	2-12
ENCRYPT.FILE . . . . .	2-13
DECRYPT.FILE . . . . .	2-17
LIST.ENCRIPTION.FILE . . . . .	2-21
ACTIVATE.ENCRIPTION.KEY . . . . .	2-22
DEACTIVATE.ENCRIPTION.KEY . . . . .	2-22
DISABLE.DECRYPTION . . . . .	2-23
ENABLE.ENCRIPTION . . . . .	2-24
UniVerse BASIC Encryption Commands . . . . .	2-25
ACTIVATEKEY . . . . .	2-25
DEACTIVATEKEY . . . . .	2-25
DISABLEDEC . . . . .	2-26
ENABLEDEC . . . . .	2-27
STATUS Function Changes . . . . .	2-27
The encman Utility . . . . .	2-29

*Beta Beta*

Viewing Audit Trail Information . . . . . 2-29  
Generating a Key Store . . . . . 2-30  
Deleting the Key Store . . . . . 2-31



At this release, automatic data encryption is introduced. With this feature, you can encrypt specified fields or entire records, and UniVerse automatically decrypts the data when accessed by UniVerse or UniVerse BASIC commands. This enhancement includes the following features:

- Defining which fields in the UniVerse file to encrypt
- Automatically encrypt the data you specify when writing the record to the UniVerse file
- Automatically decrypt the data you specify when reading the record from the file
- Key management support
- Audit trail for operations on keys and encrypted files
- Support of Federal Information Processing Standards (FIPS) encryption algorithms, which include popular encryption algorithms DES and AES.

**Note:** *When using automatic data encryption, performance may degrade due to encryption operations, and more disk space may be required.*



## Encrypted File Types

At this release, UniVerse only encrypts hashed files. UniVerse does not encrypt directory files, system log files, dictionary files, or system temporary files. However, UniVerse does encrypt the transaction log file, which contains encrypted data for files that are encrypted.

## Encryption With UniVerse Replication

If you are using UniVerse Replication, care must be taken when adding automatic data encryption. If a file that is encrypted is also being replicated, UniVerse transfers encrypted data to the subscribing system. Encryption does not occur on the subscribing system. IBM highly recommends that the encryption configuration be the same on both the publishing and subscribing systems, including the master key, encryption key, encryption file definitions, and the algorithms you specify for encryption. If the configurations are not identical, the replicated data may not be synchronized with the source data, and will not be usable when failover is required.

---

## Key Store

The most important part of an encrypted system is key management. To ensure a fully secure system, UniVerse maintains a key store, with an interface to create keys and reference keys. Keys can be protected through a user-name based access control, and also protected by a password.

The UniVerse key store is protected by a master key. This master key is known only to UniVerse, and is also used in deriving all other keys. After you install UniVerse, you should define a master key, either providing one of your own, or using the UniVerse default.

UniVerse stores the master key and loads it into memory each time UniVerse starts. UniVerse uses the master key to open the key store, and loads keys in the UniVerse work space. UniVerse can also use this master key to recover a key password if it is lost.

---

## How Encryption Works

This section gives an overview of how encryption works on a UniVerse database:

After installing UniVerse, you define a master key. You can define your own master key, or use a UniVerse default. IBM recommends that you define your own master key. UniVerse uses the master key in all operations related to encryption.

When you create a new encryption key, you can choose to protect the key with a password, or rely on the operating system-level user name to control access to the key. You can grant access to the encryption key to other users or groups based on the OS-level account name.

When you create an encrypted file, you must associate a key and an encryption algorithm for each object to encrypt. You can encrypt an entire record or a just a field or fields in the record. UniVerse checks if the user has access permission to the key based on the OS-level user or group ID, then asks for the password if the key is password protected.

During the UniVerse read or write operation, either from UniVerse BASIC, Retrieve, or UniVerse SQL, UniVerse locates the key ID associated with an encrypted field and checks if the key is active. The key is considered active if the user has permission to the key, the key is not password protected, or the key is password protected and the correct password has been provided through the `ACTIVATE.ENCRYPTION.KEY` command or the UniVerse BASIC `ACTIVATEKEY` statement.

If the operation you specify is a read operation and the key is not active, UniVerse returns an error in the UniVerse BASIC `STATUS` command, then presents encrypted data. However, if you disable encryption through the `DISABLE.DECRYPTION` command, UniVerse does not attempt to decrypt the data.

If the operation you specify is a write operation and the key is not active, the encrypted field keeps the original cipher text value, and no new encryption occurs. If the data in the encrypted field is in clear text, the write operation fails.

If you provide your own master key, the encrypted data can only be decrypted on the installed system. If you moved the encrypted data to another system, you must set up the same master key, and the same encryption key(s) with the same password, before you can read the encrypted data.

If you choose to use the UniVerse default master key, if you move the encrypted data and the key store to another UniVerse system, you must set up the same encryption keys with the same passwords before you can decrypt the data.

The following table shows the combination of the master key and the key password and their impact on security level and file portability.

System Master Key / File Encryption Key	No Password	With Password
Default	Minimum Protection. Data can be accessed on another UniVerse system with default master key and encryption key.	Strong Protection. Data can be accessed on another UniVerse system with the default master key and the same encryption key with the same password.
System-Specific (user-defined)	Strong Protection. Data can be accessed on another UniVerse system with the same user-defined master key and encryption key.	Maximum Protection. Data can be accessed on another UniVerse system with the same user-defined master key and the same encryption key and password.

**Master Key and Key Password Impact**

---

## Defining a Master Key

When you initially install UniVerse, each installation has the same default master key. For a new UniVerse installation, UniVerse displays a message at the end of the installation process to remind you to establish a site-specific master key. For an upgrade installation, UniVerse does not change your master key.

Use the `uvregen` command to define a new master key, as shown in the following example:

```
C:\IBM\UV>uvregen -m new_master_key  
Changing UV master key is DANGEROUS!!!  
Do you really want to change it [No]?Yes
```

If you specify `SYSTEM` for the master key, UniVerse changes the master key to the system default. In order to revert to the system default, you must provide the current master key.

Use `@/full_path` to indicate that the master key is stored in a file, as shown in the following example:

```
@/mysecure/mymaster
```

We recommend that the key file is strongly protected, or removed from the system after the installation is complete and stored in a safe place.

The maximum length of a master key is 64 characters. The master key should be long and difficult to guess.

## Changing a Master Key After Data is Encrypted

Once a master key has been used in file encryption, we recommend that you do not change it. All aspects of UniVerse data encryption involves the master key, and changing it makes all previously encrypted data, existing keys, and audit records inaccessible.

If you decide to change the master key, you must first decrypt all encrypted data, save a text copy of your existing audit records, and make sure you can re-create existing encryption keys. If you do not follow these steps, your data will not be accessible after you change the master key.

---

# UniVerse Encryption Algorithms

UniVerse supports the following encryption algorithms:

- AES (AES128, AES192, AES256)
- DES (DES, DES3)
- RC2
- RC4

AES and DES are Federal Information Processing Standards (FIPS) compliant encryption algorithms. Within each group, with the exception of RC4, there are multiple chaining modes (CBC, ECB, OFB, and CFB).

When you encrypt a file, you must specify a specific algorithm to use in encryption. The following table describes valid algorithms for UniVerse decryption:

Type of Encryption Desired	Algorithm to Specify
56-bit key DES encryption	des, des-cbc, des-ecb, des-cfb, or des-ofb
112-bit key ede DES encryption	des_ede, des-ede-cbc, des-ede, des-ede-cfb, or des-ede-ofb
168-bit key ede DES encryption	des3, des_ede3, des_ede3-cbc, des_ede3-cfb, or des_ede3-ofb
128-bit key R2 encryption	rc2, rc2-cbc, rc2-ecb, rc2-cfb, or rc2-ofb
128-bit key RC4 encryption	rc4
128-bit key AES encryption	aes128, aes-128-cbc, aes-128-cfb, or aes-128-ofb
192-bit key AES encryption	aes192, aes-192-cbc, aes-192-cfb, aes-192-ofb
256-bit key AES encryption	aes256, aes-256-cbs, aes-256-ecb, aes-256-cfb, or aes-256-ofb

---

## UniVerse Encryption Algorithms

***Note:** The algorithm specification is case-insensitive.*



---

## Encryption Commands

This section lists commands you can use for encrypting and decrypting your data.

### CREATE.ENCRYPTION.KEY

Use the CREATE.ENCRYPTION.KEY command to create an encryption key in the UniVerse key store. We recommend that you create a password for the key.

#### *Syntax*

**CREATE.ENCRYPTION.KEY** *key.id* [*password*]

#### *Parameters*

The following table describes each parameter of the syntax.

Parameter	Description
<i>key.id</i>	The encryption key ID.
<i>password</i>	The password for <i>key.id</i> .

#### **CREATE.ENCRYPTION.KEY Parameters**

*Note: We suggest that the password you create is a phrase that is hard to guess, but easy to remember, using a combination of ASCII characters and digits. If a password contains a space ( " "), you must use quotation marks to enclose the password.*

### DELETE.ENCRYPTION.KEY

Use the DELETE.ENCRYPTION.KEY command to delete a key from a key store. You must be the owner of the file or logged on as root or a UniVerse Administrator to delete an encryption key, and you must provide the correct password. If the key is referenced by any encrypted field or file, deleting the key will fail, unless you specify FORCE.



Syntax

DELETE.ENCRIPTION.KEY [FORCE] *key.id* [*password*]

Parameters

The following table describes each parameter of the syntax.

Parameter	Description
FORCE	Forces the encryption key to be deleted, even if it is referenced by an encrypted record or field.
<i>key.id</i>	The encryption key to delete.
<i>password</i>	The password for the encryption key to delete.

DELETE.ENCRIPTION.KEY Parameters

LIST.ENCRIPTION.KEY

Use the LIST.ENCRIPTION.KEY command to list the existing keys in the key store. You can also list records in the key store using UniVerse Retrieve commands, such as LIST, LIST.ITEM, SORT, SORT.ITEM, and so forth.

***Note:** The name of the key store file is &KEystore&. Although you can view records from this file using UniVerse Retrieve commands, other UniVerse commands, such as DELETE.FILE and CLEAR.FILE will fail. The ED command will only display encrypted data.*

GRANT.ENCRIPTION.KEY

Use the GRANT.ENCRIPTION.KEY command to grant other users access to the encryption key. When a key is created, only the owner of the key has access. The owner of the key can grant access to other users.

Syntax

GRANT.ENCRIPTION.KEY {PUBLIC | *grantee* {,*grantee*...}}





*Parameters*

The following table describes each parameter of the syntax.

Parameter	Description
PUBLIC	Grants access to the encryption key to all users on the system.
<i>grantee</i>	<p>Grants access to the encryption key to the <i>grantee</i> you specify. <i>grantee</i> can be a user name, or a group name. If you specify a group name, prefix the name with an asterisk (“*”). When you specify a group name, UniVerse grants access to all users belonging to the group.</p> <p>On Windows platforms, a group name can be a local group or a global group (specified in the form of *Domain\global-group). A user can also be a domain user, specified in the form of Domain\user. In the case of “\” appearing in a group or user name, you should use quotation marks to enclose the name.</p> <p>Grantees cannot grant access to the encryption key to other users.</p> <p><i>Note: To grant access to global users or groups, you must log on as a domain user to creat keys and perform the GRANT operation.</i></p>

**GRANT.ENCRIPTION.KEY Parameters**

You must grant access to an encryption key even if it does not have password protection if you want other users to use the key. On the other hand, even if you have the correct password for the key, you cannot access it without being granted access.

**REVOKE.ENCRIPTION.KEY**

Use the REVOKE.ENCRIPTION.KEY command to revoke access to the encryption key from other users. When a key is created, only the owner of the key has access. The owner of the key can revoke access from other users.

*Syntax*

```
REVOKE.ENCRIPTION.KEY {PUBLIC | grantee {,grantee...}}
```

Parameters

The following table describes each parameter of the syntax.

Parameter	Description
PUBLIC	Revokes PUBLIC access to the encryption key from all users on the system. For example, if “PUBLIC” access is granted, itis removed. However, this does not revoke individual user or group access that had been granted.
grantee	Revokes access to the encryption key from the <i>grantee</i> you specify. <i>grantee</i> can be a user name, or, on UNIX platforms, a group name. If you specify a group name, prefix the name with an asterisk (“*”). When you specify a group name, UniVerse revokes access from all users belonging to the group.  On Windows platforms, a group name can be a local group or a global group (specified in the form of *Domain\global-group). A user can also be a domain user, specified in the form of Domain\user. In the case of “\” appearing in a group or user name, you should use quotation marks to enclose the name.  Grantees cannot revoke access to the encryption key from other users.

REVOKE.ENCRYPTION.KEY Parameters

ENCRYPT.FILE

Use the ENCRYPT.FILE command to create a file in which each record is encrypted.

*Note:* You cannot encrypt an index file.

Syntax

```
ENCRYPT.FILE {<filename> <type> <modulo> <separation> | <30 |
dynamic> parameter [value]...} <USING partition> < { WHOLERECORD
| fieldname },alg,key[,pass] [fieldname,alg,key[,pass]]...>
```



*Parameters*

Most of the ENCRYPT.FILE parameters are the same as the RESIZE command parameters. If the file you are encrypting is empty, you do not need to specify any of the RESIZE parameters. If the file you are encrypting is not empty, and you know that the file needs resizing because encrypting the file will increase the record size, you should specify the RESIZE parameters.

The following table describes each parameter of the syntax.

Parameter	Description
<i>filename</i>	The UniVerse file name. If you do not specify <i>filename</i> , ENCRYPT.FILE prompts for the name. <i>filename</i> must follow the UniVerse naming conventions. For more information about naming conventions, see “File Naming Conventions” in <i>UniVerse User Reference</i> .
<i>type</i>	The UniVerse file type for the file you are encrypting. Type 1 or type 19 files are not hashed and are usually used to store text files such as BASIC programs. Types 2 through 18 are hashed files. Type 25 is a balanced tree file.
<i>modulo</i>	The modulo for the file you are encrypting. The modulo should be an integer from 1 through 8,388,608 defining the number of groups in the file. UniVerse ignores <i>modulo</i> if you specify a nonhashed or dynamic file type.
<i>separation</i>	The separation for the file you are encrypting. The separation should be an integer from 1 through 8,388,608, specifying the group buffer size is 512-byte blocks. UniVerse ignores <i>separation</i> if you specify a nonhashed or dynamic file type.
30	Encrypts a dynamic file.
dynamic	Encrypts a dynamic file.
USING <i>partition</i>	Specifies the path of the work area that ENCRYPT.FILE will use for creating the necessary temporary files. For example, the following command encrypts SUN.MEMBER as a dynamic file, and creates the temporary files it needs in the partition <i>/u4</i> :  <b>&gt;ENCRYPT.FILE SUN.MEMBER DYNAMIC USING /u4</b>  ENCRYPT.FILE moves the files back into the correct directory after encrypting the SUN.MEMBER file.

**ENCRYPT.FILE Parameters**

Parameter	Description
WHOLERECORD	Specifies to fully encrypt every record in the file.
<i>fieldname,alg,key,pass</i>	<p>Specifies the field name to encrypt, and the algorithm, key, and password to use. You can use a different algorithm and key for each field.</p> <p>If you do not specify a password, but created the key using password protection, UniVerse prompts for the password. If several fields use the same password, you only have to specify it once, at the first field that uses that key.</p>
<i>fieldname</i>	The name of the field to encrypt.
<i>alg</i>	The algorithm to use for encryption. See “ <a href="#">UniVerse Encryption Algorithms</a> ” on page 8 for a list of valid values.
<i>key</i>	The key ID to use for the field encryption.
<i>pass</i>	The password corresponding to the <i>key</i> .

#### ENCRYPT.FILE Parameters (Continued)

Specify the following parameters only for dynamic files:

Parameter	Description
GENERAL	Specifies the general hashing algorithm for a dynamic file. GENERAL is the default.
SEQ.NUM	Specifies a hashing algorithm suitable for sequential numbers for a dynamic file. Use this hashing algorithm only for records with IDs that are mainly numeric, sequential, and consecutive.
GROUP.SIZE { 1   2 }	Specifies the size of each group in the file, either 1 or 2. 1 specifies a group size of 2048 bytes, which is equivalent to a separation of 4. 2 specifies a group size of 4096 bytes, which is equivalent to a separation of 8. A group size of 2048 (GROUP.SIZE 1) is the default.
MINIMUM.MODULUS <i>n</i>	Specifies the minimum modulo of the file, an integer value greater than 1. This value is also the initial value of the modulo of the dynamic file. A minimum modulo of 1 is the default.

#### ENCRYPT.FILE Parameters for Dynamic Files

Parameter	Description
SPLIT.LOAD <i>n</i>	Specifies the level at which the file's modulo is increased by 1. SPLIT.LOAD takes a numeric argument indicating the percentage of space allocated for the file. When the data in the file exceeds the specified percentage of the space allocated for the file, the data in one of the groups is divided equally between itself and a new group, to increase the modulo by 1. The default SPLIT.LOAD is 80%.
MERGE.LOAD <i>n</i>	Specifies the level at which the file's modulo is decreased by 1. MERGE.LOAD takes a numeric argument indicating the percentage of space allocated for the file. When the data in the file is less than the specified percentage of the space allocated for the file, the data in the last group of the file is merged with another group, to decrease the modulo by 1. The default MERGE.LOAD is 50%.

---

**ENCRYPT.FILE Parameters for Dynamic Files (Continued)**

---

Parameter	Description
LARGE.RECORD <i>n</i>	Specifies the size of a record considered too large to be included in the primary group buffer, specified as an integer or a percentage. Specified as an integer, the value is the number of bytes a record must contain to be considered a large record. Specified as a percentage, the value is a percentage of the group size. When the size of a record exceeds the specified value, the data for the record is put in an overflow buffer, but the record ID is put in the primary buffer. This method of large record storage increases access speed. The default LARGE.RECORD size is 80%.
RECORD.SIZE <i>n</i>	Calculates the values for group size and large record size based on the value of the estimated average record size specified. The value is your estimate of the average record size for the dynamic file, specified in bytes. RECORD.SIZE does not limit the size of records. If you specify a value for group size (GROUP.SIZE) or for large record size (LARGE.RECORD), those values override the value calculated by RECORD.SIZE.
MINIMIZE.SPACE	Calculates the best amount of space required by the file (at the expense of access time), using the values for the split load, merge load, and large record size. If you specify values for split load, merge load, or large record size, those values override the value calculated by MINIMIZE.SPACE. If you specify MINIMIZE.SPACE and RECORD.SIZE, the value for large record size calculated by MINIMIZE.SPACE is used above the value calculated by RECORD.SIZE.

#### ENCRYPT.FILE Parameters for Dynamic Files (Continued)

Encrypting a file requires exclusive access to the file, and is very time consuming. During the encryption process, UniVerse creates a temporary file and writes the newly encrypted data to that file. If any errors occur during the encryption process, the command aborts and the original file is left intact.

## DECRYPT.FILE

The DECRYPT.FILE command decrypts data in a file or in the fields you specify.

## Syntax

```
DECRYPT.FILE {<filename> <type> <modulo> <separation> | <30 |  
dynamic> parameter [value]...} <USING partition> < { WHOLERECORD  
| <fieldname> },key[,pass] [fieldname,key[,pass]]...>
```

Most of the DECRYPT.FILE parameters are the same as the RESIZE command parameters. If the file you are decrypting is empty, you do not need to specify any of the RESIZE parameters. If the file you are decrypting is not empty, and you know that the file needs resizing because decrypting the file will change the record size, you should specify the RESIZE parameters.

The following table describes each parameter of the syntax.

Parameter	Description
<i>filename</i>	The UniVerse file name. If you do not specify <i>filename</i> , DECRYPT.FILE prompts for the name. <i>filename</i> must follow the UniVerse naming conventions. For more information about naming conventions, see “File Naming Conventions” in <i>UniVerse User Reference</i> .
<i>type</i>	The UniVerse file type for the file you are decrypting. Type 1 or type 19 files are not hashed and are usually used to store text files such as BASIC programs. Types 2 through 18 are hashed files. Type 25 is a balanced tree file.
<i>modulo</i>	The modulo for the file you are decrypting. The modulo should be an integer from 1 through 8,388,608 defining the number of groups in the file. UniVerse ignores <i>modulo</i> if you specify a nonhashed or dynamic file type.
<i>separation</i>	The separation for the file you are decrypting. The separation should be an integer from 1 through 8,388,608, specifying the group buffer size is 512-byte blocks. UniVerse ignores <i>separation</i> if you specify a nonhashed or dynamic file type.
30	Decrypts a dynamic file.
dynamic	Decrypts a dynamic file.

### DECRYPT.FILE Parameters

Parameter	Description
USING <i>partition</i>	<p>Specifies the path of the work area that DECRYPT.FILE will use for creating the necessary temporary files. For example, the following command decrypts SUN.MEMBER as a dynamic file, and creates the temporary files it needs in the partition <i>/u4</i>:</p> <p><b>&gt;DECRYPT.FILE SUN.MEMBER DYNAMIC USING /u4</b></p> <p>DECRYPT.FILE moves the files back into the correct directory after creating the SUN.MEMBER file.</p>
WHOLERECORD	Specifies to fully decrypt every record in the file.
<i>fieldname,key,pass</i>	<p>Specifies the field name to decrypt, and the key, and password to use. You can use a different key for each field.</p> <p>If you do not specify a password, but created the key using password protection, UniVerse prompts for the password. If several fields use the same password, you only have to specify it once, at the first field that uses that key.</p>
<i>fieldname</i>	The name of the field to decrypt.
<i>key</i>	The key ID to use for the field decryption.
<i>pass</i>	The password corresponding to the <i>key</i> .

#### DECRYPT.FILE Parameters (Continued)

Specify the following parameters only for dynamic files:

Parameter	Description
GENERAL	Specifies the general hashing algorithm for a dynamic file. GENERAL is the default.
SEQ.NUM	Specifies a hashing algorithm suitable for sequential numbers for a dynamic file. Use this hashing algorithm only for records with IDs that are mainly numeric, sequential, and consecutive.
GROUP.SIZE { 1   2 }	<p>Specifies the size of each group in the file, either 1 or 2.</p> <p>1 specifies a group size of 2048 bytes, which is equivalent to a separation of 4. 2 specifies a group size of 4096 bytes, which is equivalent to a separation of 8. A group size of 2048 (GROUP.SIZE 1) is the default.</p>

#### DECRYPT.FILE Parameters for Dynamic Files



Parameter	Description
MINIMUM.MODULUS <i>n</i>	Specifies the minimum modulo of the file, an integer value greater than 1. This value is also the initial value of the modulo of the dynamic file. A minimum modulo of 1 is the default.
SPLIT.LOAD <i>n</i>	Specifies the level at which the file's modulo is increased by 1. SPLIT.LOAD takes a numeric argument indicating the percentage of space allocated for the file. When the data in the file exceeds the specified percentage of the space allocated for the file, the data in one of the groups is divided equally between itself and a new group, to increase the modulo by 1. The default SPLIT.LOAD is 80%.
MERGE.LOAD <i>n</i>	Specifies the level at which the file's modulo is decreased by 1. MERGE.LOAD takes a numeric argument indicating the percentage of space allocated for the file. When the data in the file is less than the specified percentage of the space allocated for the file, the data in the last group of the file is merged with another group, to decrease the modulo by 1. The default MERGE.LOAD is 50%.

---

**DECRYPT.FILE Parameters for Dynamic Files (Continued)**

---

Parameter	Description
LARGE.RECORD <i>n</i>	Specifies the size of a record considered too large to be included in the primary group buffer, specified as an integer or a percentage. Specified as an integer, the value is the number of bytes a record must contain to be considered a large record. Specified as a percentage, the value is a percentage of the group size. When the size of a record exceeds the specified value, the data for the record is put in an overflow buffer, but the record ID is put in the primary buffer. This method of large record storage increases access speed. The default LARGE.RECORD size is 80%.
RECORD.SIZE <i>n</i>	Calculates the values for group size and large record size based on the value of the estimated average record size specified. The value is your estimate of the average record size for the dynamic file, specified in bytes. RECORD.SIZE does not limit the size of records. If you specify a value for group size (GROUP.SIZE) or for large record size (LARGE.RECORD), those values override the value calculated by RECORD.SIZE.
MINIMIZE.SPACE	Calculates the best amount of space required by the file (at the expense of access time), using the values for the split load, merge load, and large record size. If you specify values for split load, merge load, or large record size, those values override the value calculated by MINIMIZE.SPACE. If you specify MINIMIZE.SPACE and RECORD.SIZE, the value for large record size calculated by MINIMIZE.SPACE is used above the value calculated by RECORD.SIZE.

#### DECRYPT.FILE Parameters for Dynamic Files (Continued)

If the encrypted file was created using the WHOLERECORD keyword, you should specify WHOLERECORD when decrypting the file. If the file was not encrypted using the WHOLERECORD keyword, do not specify WHOLERECORD when decrypting the file.

## LIST.ENCRYPTION.FILE

Use the LIST.ENCRYPTION.FILE command to display encryption configuration data, such as the fields that are encrypted, the algorithms used, and so forth. This command also displays the fields for which decryption is currently disabled.

## Syntax

**LIST.ENCRYPTION.FILE** *filename*

## ACTIVATE.ENCRYPTION.KEY

Use the ACTIVATE.ENCRYPTION.KEY command to activate a key. It is necessary to activate a key if you want to supply a password for key protection.

## Syntax

**ACTIVATE.ENCRYPTION.KEY** *key.id password* [ON *<hostname>*]

## Parameters

The following table describes each parameter of the syntax.

Parameter	Description
<i>key.id</i>	The key ID to activate.
<i>password</i>	The password corresponding to <i>key.id</i> .
ON <i>hostname</i>	The name of the remote host on which you want to activate the encryption key.

### ACTIVATE.ENCRYPTION.KEY Parameters

**Note:** You can activate only keys with password protection with this command. Keys that do not have password protection are automatically activated. Also, you can activate only keys to which you are granted access.

## DEACTIVATE.ENCRYPTION.KEY

Use the DEACTIVATE.ENCRYPTION.KEY command to deactivate one or more encryption keys. This command is useful to deactivate keys to make your system more secure.



### *Syntax*

**DEACTIVATE.ENCRIPTION.KEY** *key.id password* [ON *<hostname>*]

### *Parameters*

The following table describes each parameter of the syntax.

Parameter	Description
<i>key.id</i>	The key ID to deactivate.
<i>password</i>	The password corresponding to <i>key.id</i> .
ON <i>hostname</i>	The name of the remote host on which you want to deactivate the encryption key.

#### **DEACTIVATE.ENCRIPTION.KEY Parameters**

**Note:** *You can deactivate only keys with password protection with this command. Keys that do not have password protection are automatically activated and cannot be deactivated.*

## **DISABLE.DECRYPTION**

Use the DISABLE.DECRYPTION command to turn off decryption on a field you specify.

### *Syntax*

**DISABLE.DECRYPTION** *filename <field\_list>*



## Parameters

The following table describes each parameter of the syntax.

Parameter	Description
<i>filename</i>	The name of the file on which you want to disable decryption.
<i>field_list</i>	A comma-separated list of fields for which you want to disable decryption. Do not enter spaces between the field names.

### DISABLE.DECRYPTION Parameters

## ENABLE.ENCRYPTION

Use the ENABLE.ENCRYPTION command to activate encryption on specific fields in a file.

## Syntax

**ENABLE.ENCRYPTION** *filename* <*field\_list*>

## Parameters

The following table describes each parameter of the syntax..

Parameter	Description
<i>filename</i>	The name of the file on which you want to enable encryption.
<i>field_list</i>	A comma-separated list of fields for which you want to enable encryption. Do not enter spaces between the field names.

### ENABLE.ENCRYPTION Parameters

---

## UniVerse BASIC Encryption Commands

This section describes the UniVerse BASIC commands for use with encryption and decryption.

### ACTIVATEKEY

Use the ACTIVATEKEY command to activate a key. It is necessary to activate a key if you want to supply a password for key protection.

#### *Syntax*

**ACTIVATEKEY** <key.id>, <password> [ON <hostname>]

#### *Parameters*

The following table describes each parameter of the syntax.

Parameter	Description
<i>key.id</i>	The key ID to activate.
<i>password</i>	The password corresponding to <i>key.id</i> .
ON <i>hostname</i>	The name of the remote host on which you want to activate the encryption key.

---

#### **ACTIVATEKEY Parameters**

**Note:** You can activate only keys with password protection with this command. Keys that do not have password protection are automatically activated. Also, you can activate only keys to which you are granted access.

### DEACTIVATEKEY

Use the DEACTIVATEKEY command to deactivate one or more encryption keys. This command is useful to deactivate keys to make your system more secure.



## Syntax

**DEACTIVATEKEY** <key.id>, <password> [ON <hostname>]

## Parameters

The following table describes each parameter of the syntax.

Parameter	Description
<i>key.id</i>	The key ID to deactivate.
<i>password</i>	The password corresponding to <i>key.id</i> .
ON <i>hostname</i>	The name of the remote host on which you want to deactivate the encryption key.

### DEACTIVATEKEY Parameters

**Note:** You can deactivate only keys with password protection with this command. Keys that do not have password protection are automatically activated and cannot be deactivated.

## DISABLEDEC

Use the DISABLEDEC command to turn off decryption on a file or fields you specify.

## Syntax

**DISABLEDEC** <filename> [, <multilevel-filename>], <field\_list>



## ***Parameters***

The following table describes each parameter of the syntax.

Parameter	Description
<i>filename</i>	The name of the file on which you want to disable decryption.
<i>field_list</i>	A comma-separated list of fields for which you want to disable decryption. Do not enter spaces between the field names.

### **DISABLEDEC Parameters**

## **ENABLEDEC**

Use the ENABLEDEC command to activate decryption on a file or fields you specify.

## ***Syntax***

**ENABLEDEC** <*filename*> [, <*multilevel-filename*>], <*field\_list*>

## ***Parameters***

The following table describes each parameter of the syntax.

Parameter	Description
<i>filename</i>	The name of the file on which you want to enable decryption.
<i>field_list</i>	A comma-separated list of fields for which you want to enable decryption. Do not enter spaces between the field names.

### **ENABLEDEC Parameters**

## **STATUS Function Changes**

The following changes have been made to the UniVerse BASIC STATUS function:

- For UniVerse BASIC READ statements, STATUS() returns 5 to indicate that an encryption error occurred during the READ operation.



- For UniVerse BASIC WRITE statements, STATUS() returns -9 to indicate that an encryption error occurred during the WRITE operation.
- When an encryption error occurs, a READ/WRITE statement will execute statements following the ELSE clause, if an ELSE clause is specified.

---

# The encman Utility

The encman utility enables you to manage data encryption. You can view audit trail information, create a key store, or delete a key store through this utility.

## Viewing Audit Trail Information

Use the encman -audit command to view audit trail information.

### Syntax

```
encman [ [-audit] [-b date] [-a date] [-u username] [-o operation] [-f]
[-backup <file>] [-use <file>]]
```

The following table describes each parameter of the syntax.

Parameter	Description
-b <i>date</i>	Displays audit trail data before the date you specify. Enter the date in the mm/dd/yyyy format.
-a <i>date</i>	Displays audit trail data after the date you specify. Enter the date in the mm/dd/yyyy format.
-u <i>username</i>	Displays audit trail data for the user name you specify. You can specify multiple users, for example, -u user1 -u user2.

**encman -audit Parameters**

Parameter	Description
-o <i>operation</i>	Displays audit trail data for the operation you specify. You can specify multiple operations. Valid operations are: <ul style="list-style-type: none"> <li>■ CREATE – Creating encryption key</li> <li>■ DELETE – Deleting encryption key</li> <li>■ GRANT – Granting key access</li> <li>■ REVOKE – Revoking key access</li> <li>■ ACTIVATE – Activating encryption key</li> <li>■ DEACTIVT – Deactivating encryption key</li> <li>■ ENABLE – Enabling encryption key</li> <li>■ DISABLE – Disabling encryption key</li> <li>■ ENCRYPT – Encrypting a file</li> <li>■ DECRYPT – Decrypting a file</li> <li>■ RMKEYSTR – Deleting Key Store</li> </ul>
-f	Displays only failed operations.
-backup < <i>file</i> >	Backs up the current audit file to the < <i>file</i> > you specify, then clears the audit file.
-use < <i>file</i> >	Displays data in the < <i>file</i> > you specify, rather than the current audit file.

**encman -audit Parameters (Continued)**

## Generating a Key Store

To generate a key store, use the -genkeystore option.

### Syntax

**encman** [ [-genkeystore] [-n] ]

The following table describes each parameter of the syntax.

Parameter	Description
-n	Specifies to not create the &ENCINFO& file.

**encman -genkeystore Parameters**

# Deleting the Key Store

To delete the current key store, use the -delkeystore option.

## Syntax

**encman** [ [-delkeystore] [-f] ]

The following table describes the parameter of the syntax:

Parameter	Description
-f	Deletes the key store without prompting for confirmation. <i>Note: Using this operation is dangerous. If you have encrypted files, data cannot be retrieved unless you recreate the keystore and keys used by these files.</i>
<b>encman -delkeystore Parameter</b>	

---

# IBM U2 Web Services Developer

IBM U2 Web Services Developer . . . . .	3-2
Minimum Requirements . . . . .	3-2

---

## IBM U2 Web Services Developer

The IBM U2 Web Services Developer provides an easy environment to publish your UniVerse database resource as web services.

You can create a web service for the following UniVerse operations:

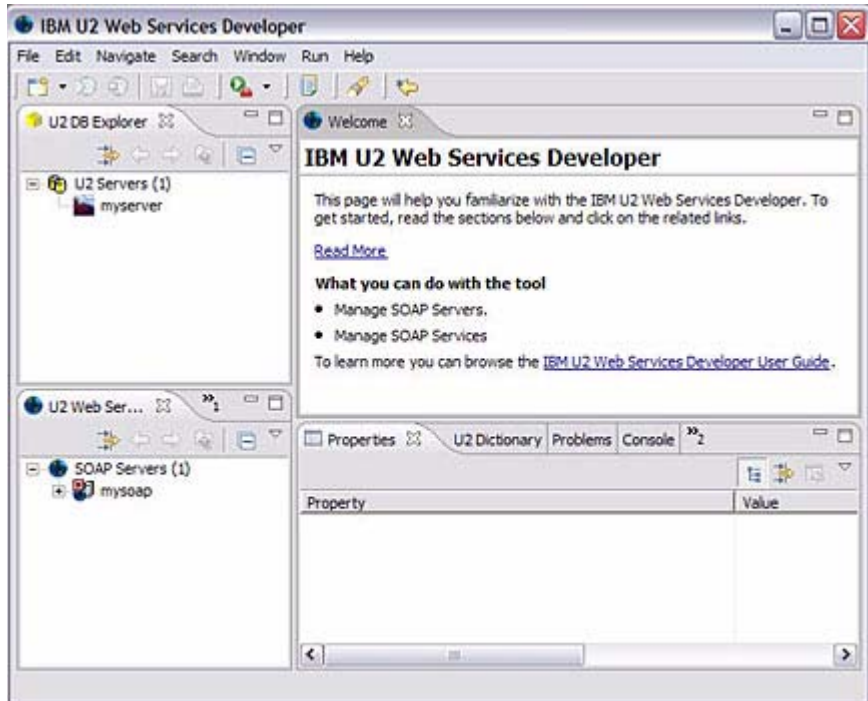
- Retrieve statement
- UniVerse SQL statement
- UniVerse BASIC subroutine

### Minimum Requirements

To access the IBM U2 Web Services Developer, you must have:

- UniVerse 10.1.18 or greater, with connection pooling licensed.
- UniData 7.1 or greater, with connection pooling licensed.

From the **Start** menu, select **All Programs**, select **IBM U2**, select **Web Tools**, then click **IBM U2 Web Services Developer**. A dialog box similar to the following example appears:



From this dialog box, you can connect to or establish a new U2 Server and SOAP Server. After connecting to the servers, you can define web services using Retrieve, UniVerse SQL, or UniVerse BASIC.

For detailed information about The IBM U2 Web Services Developer, see the *IBM U2 Web Services Developer* manual.

---

# Licensing Changes in UniVerse 10.2

Authorizing UniVerse . . . . .	4-4
Authorizing a New UniVerse License . . . . .	4-5
Obtain Configuration Code . . . . .	4-6
Obtain Authorization Code . . . . .	4-6



At UniVerse 10.2, you no longer license the product during the installation. Licensing UniVerse is now a separate process.

---

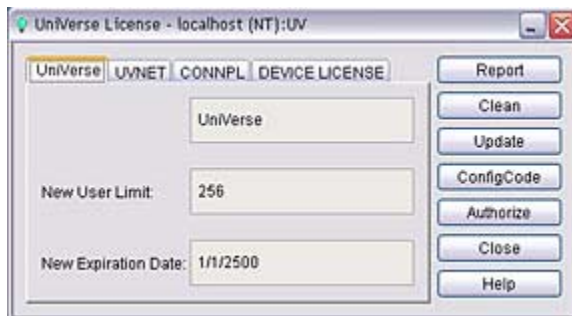
## Authorizing UniVerse

Complete the following steps to authorize UniVerse through UniAdmin:

Select one of the following methods to access the **UniVerse License** dialog box:

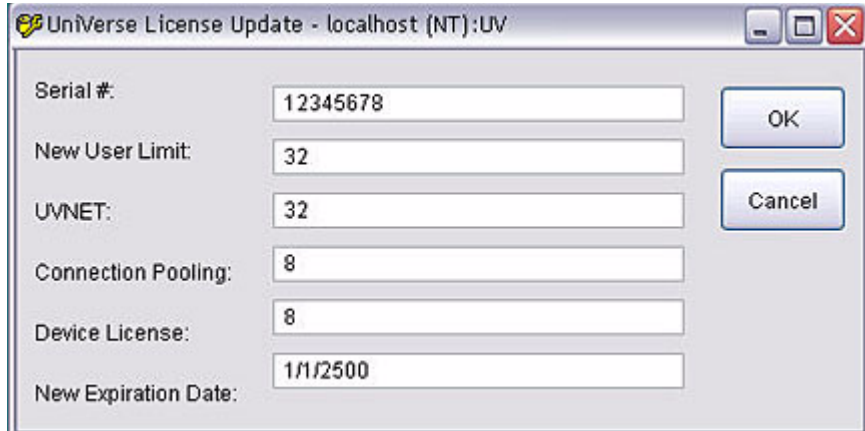
- From the **UniAdmin** window, double-click **License**.
- From the **UniAdmin** menu, click Admin, then click **License**.

The **UniVerse License** dialog box appears, as shown in the following example:



## Authorizing a New UniVerse License

Verify that the number of users and expiration date displayed in the **UniVerse Licensing** dialog box matches the configuration on the Product Configuration sheet shipped with UniVerse. If you need to update any information, click **Update**. The **UniVerse License Update** dialog box appears, as shown in the following example:



UniVerse License Update - localhost (NT):UV

Serial #: 12345678

New User Limit: 32

UVNET: 32

Connection Pooling: 8

Device License: 8

New Expiration Date: 1/1/2500

OK

Cancel

1. Enter your UniVerse serial number in the **Serial #** box.
2. Enter the number of users for which you are licensed in the **New User Limit** box.
3. Enter the number of UVNet users for which you are licensed in the **UVNET** box. If you are not licensed for any UVNet users, enter 0.
4. Enter the number of Connection Pooling licenses in the **Connection Pooling** box. If you are not licensed for any connection pools, enter 0.
5. Enter the number of device licenses for which you are authorized in the **Device License** box.
6. If the expiration date of you license is incorrect, enter the correct date in the **New Expiration Date** box.

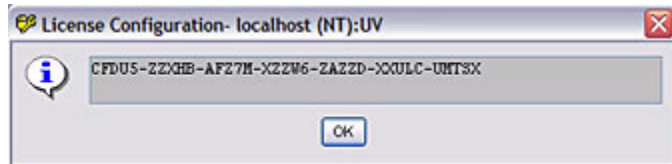
When the license information is correct, click **OK**.



**Note:** If you are using UV/NET, you must authorize both the UniVerse database and UV/NET.

## Obtain Configuration Code

Click **ConfigCode** to obtain the configuration code you will need to authorize UniVerse. A window similar to the following example appears with the configuration code:



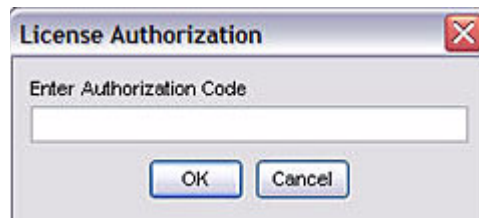
## Obtain Authorization Code

To obtain your authorization code, go to:

<http://www-306.ibm.com/software/data/u2/universe/>

Click **Authorize Products**. Follow the instructions on the website to obtain your authorization code.

Once you have your authorization code, click **Authorize** from the **UniVerse License** dialog box. The **License Authorization** dialog box appears, as shown in the following example:



Enter your authorization code, then click **OK**.