APAR OA43999 – RACF password security enhancements

Note: This document also contains information for the corresponding System Authorization Facility (SAF) APAR OA43998.

Summary of Changes		
Version	**Date**	**Nature of Change**
1.0	11/03/14	Initial version
1.1	11/11/14	Correction to PWCLEAN/PWCONVERT bit values in ALTUSER SMF mapping for “Flags for additional keywords ignored because of processing error”
1.2	03/10/15	Clarify that a phrase-only user can be authenticated using a PassTicket (OA47205)

Table of Contents

1 Introduction

There are many different elements of password security, including password length, password quality, encryption strength, access control of the password database, system-wide policy controls, and most important, user education. All of these aspects should be considered together to provide a multi-layered and interdependent set of defenses.

Policy controls, such as an invalid password revoke count, prevents repeated guesses against a user's password. Careful protection of the password database ensures that

nobody can start an offline attack against encrypted passwords. Should a copy of the database be inappropriately accessed, then longer and stronger passwords are harder to compromise. Strong encryption requires the attacker to use time and resources in a brute-force attack. A relatively short password change interval lessens the chance that a password can be cracked while it is still valid. Password history keeps users from reusing the same password so that an attacker does not essentially have an infinite time with which to crack it. However, password history does provide a cracker with a set of values that the user could be reusing. A small revoke count should be sufficiently low such that even if all history values were cracked, the chances of guessing which one might be the current password would be reduced. Users should be trained that even if they can use a previous value, they should not. Despite all of these controls, if a user has the same password in RACF and in their favorite gaming website, and that website is successfully compromised, then your careful work is wasted if a cracker has enough personal information to correlate that user with a user ID on your corporate network.

There is a lot more to password security than mere technical controls. Technical controls are necessary, but alone are not sufficient. They do, however, provide an aspect of defense in depth in a multi-layered security strategy.

With this enhancement, RACF provides more controls to help you maintain an effective password security policy:

- A stronger encryption algorithm for passwords and password phrases, which is easily activated with the SETROPTS command. A function is provided to convert existing DES passwords to the new format without requiring the passwords to be changed.
- You can demonstrate to an auditor that passwords and password phrases are encrypted under the new algorithm with the help of new fields created by the IRRDBU00 utility. A sample query is included.
- Support for 14 additional special characters in passwords. This significantly increases the password space, and thus the work factor that is involved in cracking a password. It also allows users to choose passwords that are less likely to exist in a list of frequently used passwords that are the first ones that are attempted in a cracking effort.
- A new password syntax control that requires a password to contain at least one character from each of four different categories: uppercase letters, lowercase letters, numeric digits, and symbolic characters (which include the existing national characters and the newly supported special characters). This can help prevent users from choosing weak passwords.
- The ability for a user to have a password phrase without a password. This allows for a much longer authenticator, without the effort and exposure of also maintaining a password, or having to write, install, and maintain an exit.
- An ALTUSER command function to mark a user's password and password phrase as expired, without having to change its value. You can force a password change without needing to generate and communicate a random temporary password.
- An ALTUSER command function to clean up password history after changing the SETROPTS PASSWORD(HISTORY(*n*)) value. This replaces the need for the CUTPWHIS utility available as a download on the RACF website.

The new encryption algorithm is KDFAES (Key Derivation Function with AES). The key

derivation function appends random data to the password or password phrase, and then iteratively hashes it with SHA256 to derive a 256-bit encryption key. This key is then used to AES-encrypt the user ID appended with other data. The result is the password hash. This hash is stored in the RACF database along with the random data, and other parameters, that were used to derive it.

2 Overview of new functions

Before using the new functions, read the planning considerations below. A brief overview of the functions is included here to provide context for the subsequent documentation. Additional information is in the Updated RACF publications section below.

To activate special character support, use the SETROPTS command:

```
SETROPTS PASSWORD(SPECIALCHARS)
```

To define a password syntax rule that requires one of each character type, use the new MIXEDALL content-keyword. For example:

```
SETROPTS PASSWORD(RULE1(LENGTH(8) MIXEDALL(1:8)))
```

If SETROPTS PASSWORD(NOMIXEDCASE) is in effect, a lowercase letter is not required. If SETROPTS PASSWORD(NOSPECIALCHARS) is in effect, then the special character requirement can be satisfied with a national character (these are **not** grouped with uppercase letters as they are when using the existing content-keywords ALPHANUM and MIXEDNUM).

To activate the new encryption algorithm, use the SETROPTS command:

```
SETROPTS PASSWORD(ALGORITHM(KDFAES))
```

SETROPTS LIST displays the setting of the algorithm and special characters options. Note that the word LEGACY is displayed as the encryption algorithm when KDFAES is not active.

When KDFAES is active, legacy passwords continue to be evaluated without requiring them to be changed (unless you are using the masking algorithm. See the ICHDEX01 considerations below).

If you are currently using DES, and want to convert existing passwords (and password history entries) to KDFAES format without requiring them to be changed, you can use the new conversion function provided with the ALTUSER command:

```
ALTUSER userID PWCONVERT
```

This does not convert password phrases or password phrase history. New fields that are unloaded by the IRRDBU00 utility can be used to identify KDFAES versus legacy format

passwords, phrases, and history entries.

The SEARCH command with the CLIST option can be used to create a bulk conversion utility:

```
SEARCH CLASS(USER) CLIST('ALTUSER ' ' PWCONVERT')
```

A password phrase may now be assigned to a user without requiring a password. To remove a password, use the existing NOPASSWORD keyword of ALTUSER:

```
ALTUSER userID NOPASSWORD
```

The LISTUSER command identifies a phrase-only user by the attributes NOPASSWORD and PASSPHRASE.

Note: A phrase-only user can be authenticated using a PassTicket.

To expire a user's password and password phrase without changing them, use the existing EXPIRED keyword of ALTUSER without also specifying PASSWORD or PHRASE:

```
ALTUSER userID EXPIRED
```

Use SEARCH with CLIST, as described above, to perform this in bulk.

To clean up a user's password and password phrase history after changing the SETROPTS PASSWORD(HISTORY(*n*)) value, use the new PWCLEAN keyword of ALTUSER:

```
ALTUSER userID PWCLEAN
```

Use SEARCH with CLIST, as described above, to perform this in bulk.

Restriction: The ISPF panels and TSO helps are not updated for the new command operands with OA43999.

3 Planning

While the new algorithm and special character support can be activated using the SETROPTS command, consider the following before activating them.

- Create a backup copy of your RACF database.
- Apply OA43998 and OA43999 to all systems sharing the RACF database.
- Make sure all necessary PTFs are applied to other products that are affected by this support.
- Check programs you have written to ensure that they can tolerate the new function.
- Determine if the new function affects RACF exits, if present on your system.
- If you are using RACF downloads, determine if they are affected.
- Consider effects on the performance and space usage of your RACF database.

- Activate and test the new functions in your application test environment before activating them in your production environment.

Note: The new encryption algorithm uses the Central Processor Assist For Cryptographic Function (CPACF) to perform SHA-256 operations. SHA-256 is not supported in the CPACF on the z890, z990, z800 and z900 processors, or their predecessors. When the CPACF is not available, SHA-256 is performed in software. Therefore, consider planning a performance test to ensure that the increased computational complexity does not create a performance issue on these older processors.

3.1 Create a backup copy of your RACF database

Creating a backup of the RACF database is recommended whenever significant changes are being made to RACF and the RACF database.

3.2 Apply OA43998 and OA43999 to all systems that share the RACF database

Make sure that the service is applied on all sharing systems, and that all the ++HOLD documentation has been reviewed.

3.3 Apply service to other affected products

A fix category is an identifier used to group and associate PTFs to a particular category of software service. The following fix categories identify the group of fixes that are required to support these password enhancements.

Category	Description
`IBM.Function.RACF.PasswordCharacters`	Fixes for z/OS Security Server RACF to support additional special characters in passwords, and fixes for other z/OS software to support this enhancement.
`IBM.Function.RACF.PasswordEncryption`	Fixes for z/OS Security Server RACF to support a stronger password encryption algorithm, and fixes for other z/OS software to support this enhancement.

For more information on fix categories, see:

http://www-03.ibm.com/systems/z/os/zos/features/smpe/fix-category.html

Informational APAR II14765 contains pertinent information (e.g. restrictions) for other software products related to this support.

3.4 Determine if programs you have written are affected

If you have any programs that authenticate users with passwords, or otherwise manage passwords, they should be checked to determine if they are affected by this new support.

For special character support, if a program is validating that a password contains only characters that RACF currently allows, then this program does not work with special characters after they are activated. Generally speaking, any program of this type should pass the unmodified string to the security product and the security product determines its validity.

For password encryption support, certain types of applications are affected when the new algorithm is activated. Any application that authenticates a user or stores into the RACF database using clear text is **not** affected.

Applications that are affected are those that have dependencies on the current content or format of the PASSWORD field in the RACF database. For example, if the password field is extracted (using ICHEINTY or RACROUTE REQUEST=EXTRACT,TYPE=EXTRACT) and compared against a password that the application has encrypted itself (possibly by using RACROUTE REQUEST=EXTRACT,TYPE=ENCRYPT), this no longer works when the password on the RACF database is encrypted under KDFAES.

Another example is an application that passes a pre-encrypted password to RACROUTE REQUEST=VERIFY/X with ENCRYPT=NO. If the application supplies a current password extracted from the current password field, it fails when the password on the RACF database is encrypted under KDFAES. If the application supplies a *new* password that is encrypted using RACROUTE, this might not work after KDFAES is activated. The encryption method is specified on RACROUTE REQUEST=EXTRACT,TYPE=ENCRYPT,ENCRYPT=(*data-addr*,*method*). If the method specified is DES, this continues to work. If the method specified is STDDES or HASH, this does not work. If the method is INST, and there is no ICHDEX01 exit active, or the exit directs RACF to use DES, then this continues to work. However, if the exit directs RACF to use the masking algorithm, or if the exit performs its own encryption algorithm, this does not work.

Finally, any application that copies a password or manages the password history is affected when the password is encrypted using KDFAES. That application then requires awareness of the active encryption algorithm and must take into account new password extension fields that are used for a KDFAES password.

3.5 RACF exit considerations.

3.5.1 ICHDEX01 – Password authentication exit

The ICHDEX01 exit can be used to implement an encryption algorithm, or to instruct RACF to use DES or masking. These three algorithms are collectively referred to as “legacy”, to differentiate them from KDFAES. If there is no ICHDEX01 exit, RACF encrypts new passwords using DES, and uses DES and masking while evaluating passwords. After KDFAES is activated, RACF continues to call ICHDEX01 to evaluate a legacy password. When a password is changed under KDFAES, ICHDEX01 is no longer called for that password.

The considerations for activating KDFAES depend on the legacy algorithm currently active.

DES

If you have no ICHDEX01 exit, or your ICHDEX01 exit directs RACF to use the DES algorithm (by setting return code 8 or 16), then there are no issues with activating KDFAES, or performing a KDFAES conversion.

Note that when KDFAES is activated, RACF never uses the masking algorithm, regardless of the presence or function of ICHDEX01. If your only purpose for having an ICHDEX01 exit is to instruct RACF to use only DES (return code 8), then the exit is no longer necessary after activating KDFAES. You might want to leave it active for a period of time until you are certain that you do not need to deactivate KDFAES.

Installation-defined encryption method

If you have an ICHDEX01 exit that performs its own encryption algorithm, it needs to stay active until all passwords have been changed under KDFAES and until all non-KDFAES history entries have been replaced in the history. **Do not perform a KDFAES conversion.** The conversion function assumes that the input password hash is created with DES. If you do perform a conversion, a user is no longer able to log on with their existing password, and an administrative password change is required. Also, history entries are not usable. (A user is able to reuse any installation-encrypted password value that is contained in history).

You can determine when no more legacy passwords, phrases, and history values exist by using the new fields unloaded by IRRDBU00.

Masking

Masking is only active for encrypting passwords when you have an ICHDEX01 exit that sets return code 4. Since masking is never used when KDFAES is activated, there is more work to do on your part. **Do not perform a KDFAES conversion.** The conversion function assumes that the input password hash is created with DES. If you do perform a conversion, a user can no longer log on with their existing password, and an administrative password change is required. Instead, either:

- Convert to DES before activating KDFAES. This would entail changing ICHDEX01 to set a return code of 16, which causes RACF to encrypt new passwords using DES, and to try masking during evaluation if the DES evaluation fails. The password expiration function can be used to force users to change their password at their next log on

attempt, however, this does not address existing history entries. It might not be easy to know when all user passwords have changed, as there is no way to differentiate a masked password from a DES password on the RACF database.

- Implement a bulk password change on a test system, and sacrifice your history entries. Follow these steps:
  1. Make a copy of the your RACF database.
  2. Activate this copy on a test system.
  3. Activate KDFAES on the test system.
  4. Perform a bulk password change, notifying users of their pending new password.
  5. Activate this copy on your production system.
  6. Remove your ICHDEX01 exit, or change it to set return code 8 to provide a safer default in case you need to deactivate KDFAES.

Note that this results in unusable history entries (a user can reuse any previously masked password value contained in history).

3.5.2 ICHPWX01 – New password exit

The ICHPWXP parameter list contains the address to a password history structure (the PWXPWHST field). This parameter is passed to the exit by RACROUTE REQUEST=VERIFY/X and the PASSWORD command. When KDFAES is enabled, the PWXPWHST field is always 0, even if SETROPTS PASSWORD(HISTORY(*n*)) is active.

If you are using the sample exit available on the RACF web site, note that it requires a change if you plan to enable special character support. The exit, as written, does not allow any of the special characters to be specified. An update to the download is available. However, you might choose to make the update yourself. Replace the following line in IRRPWREX:

```
special = '$@#'
```

with

```
special = '$@#.<+|&!*-%_>?:='
```

If you have products or applications that have restrictions on the permissible character set for passwords, you can delete the characters that are not supported. Refer to informational APAR II14765 for a list of known restrictions.

Note that the “Pwd_req_types” check can now be fully implemented in a SETROPTS password syntax rule using the new MIXEDALL content-keyword.

3.5.3 ICHPWX11 – New password phrase exit

When KDFAES is not active, the ICHPWX11 exit must approve a password phrase value

between 9 and 13 characters, inclusive. In the absence of an ICHPWX11 exit, RACF rejects any password phrase less than 14 characters in length. A REXX-based sample exit is provided in SYS1.SAMPLIB.

When KDFAES is active, RACF allows a password phrase value of 9 characters or higher without an ICHPWX11 exit being active. You can remove the ICHPWX11 exit after enabling KDFAES if the only purpose of the exit is to allow a password phrase less than 14 characters.

If you use the sample exit for the additional checking it provides, note that the IRRPHREX exec requires a change if you plan to enable the new encryption algorithm and take advantage of the 9-character minimum password phrase length. The exit, as written, fails a password phrase less than 14 characters in length. An update to the sample is included in this APAR.

3.6 RACF download considerations.

The following downloads are available on the RACF web site at:

http://www-03.ibm.com/systems/z/os/zos/features/racf/downloads/index.html

3.6.1 CUTPWHIS

This is a utility that removes extraneous passwords from the RACF password history. This function is now provided by the PWCLEAN keyword of the ALTUSER command.

There is no update planned for CUTPWHIS.

Attention: Continued use of CUTPWHIS after activating the new encryption algorithm results in damaged password history and allows users to reuse password values.

3.6.2 IRRXUTIL

This is a set of sample REXX programs which illustrate the power of IRRXUTIL, the REXX interface to the R_admin callable service. The XSETRPWD sample is a program that displays only the password-related SETROPTS options, and indicates whether password and password phrase enveloping is active. This program is updated to report on the new settings and to support the new RACF password syntax rule enhancements.

3.6.3 PWDCOPY

This is a utility that copies passwords from one RACF database to another RACF database. Monitor the RACF website for availability of an update.

Attention: Continued use of the old version of PWDCOPY after activating the new encryption algorithm results in users being unable to log on on the target system of the copy.

3.6.4 REXXPWEXIT

This is a REXX-based sample new password exit.

The considerations are documented above under ICHPWX01 exit considerations.

Note that REXXPWEXIT is also being updated with a number of new quality checks that are not directly related to the RACF password enhancements.

3.7 Performance and space considerations

Consider two different aspects regarding performance, described below:
1. An increase in CPU required to evaluate and change passwords.
2. Fragmentation of the RACF database that can reduce performance of RACF database look-ups.

3.7.1 CPU consumption

The new encryption algorithm intentionally uses a higher amount of CPU than DES. It can be difficult to predict the effect this algorithm has on user response time and overall system performance, as it depends on the frequency of password evaluations.

This also affects password and password phrase changes, especially when history is active. Beyond the encryption that is required to verify the current value, the new value must also be encrypted. When history is active, the new value needs to be encrypted separately against each KDFAES history value to perform the comparison. If you have a high history value, and your users all tend to change their passwords on the same day, consider using the new password expiration function to evenly distribute password expiration dates among your user population. Also, consider using the PWCLEAN function of ALTUSER if you have lowered your history value in the past.

Attention: IBM strongly recommends that you implement RACF ACEE caching using the IRRACEE class in VLF. See the z/OS Security Server RACF System Programmer's Guide for more information.

3.7.2 RACF database performance and space utilization

Passwords, password phrases, and history encrypted under the new algorithm occupies more space in USER profiles than they do under DES. A password extension field is defined in the RACF database templates for each of these entities to contain the additional information that does not fit within the current field. For password and password phrase history, a parallel field contains the extension information, and requires a corresponding set of generation numbers by which to index the elements. The size of a password, password history entry, and password phrase history entry increases by approximately 40 bytes. The current password phrase field is variable length. Under KDFAES, a password phrase is stored in the same manner as a password. Using your SETROPTS PASSWORD(HISTORY(*n*)) value, you can determine approximately how much more space is required. Ensure that your RACF database contains enough unused space to accommodate this, and increase the size of your database if necessary.

As passwords and password phrases change over time, and a user's profile grows in size, it might need to be split across data blocks in the RACF database. When this happens, references to this profile require additional I/O. IRRUT400 can be used to reorganize databases to bring the blocks back within close proximity of each other, including the index block that references them.

Consider performing a KDFAES conversion (for passwords and password history) up front to get the user profiles to their maximum size, and then perform a reorganization. Keep in mind, however, that conversion does not affect password phrases. If you use password phrases, the profile grows slowly over time, as they are changed.

3.8 Test the change

After activating KDFAES and changing your password, you can then log on to all applications that provide an authentication dialog. Try this with special characters if you choose to enable that option. Then, change your password or password phrase with all such applications and verify that you can authenticate with the new value.

Using test workloads that are indicative of your production workloads, you can estimate performance impacts. You can consider performing a KDFAES conversion to ensure that your database can accommodate the extra space utilization. You could also use IRRUT400 to reorganize the database, and then activate that copy.

4 Updated RACF publications

Chapters of the following RACF publications are affected by the new function:

Publication Name	Publication Number	
	Version 1	Version 2
z/OS Security Server RACF Security Administrator's Guide	SA22-7683	SA23-2289
z/OS Security Server RACF Command Language Reference	SA22-7687	SA23-2292
z/OS Security Server RACF System Programmer's Guide	SA22-7681	SA23-2287
z/OS Security Server RACROUTE Macro Reference	SA22-7692	SA23-2294
z/OS Security Server RACF Callable Services	SA22-7691	SA23-2293
z/OS Security Server RACF Macros and Interfaces	SA22-7682	SA23-2288
z/OS Security Server RACF Data Areas	GA22-7680	GA32-0885
z/OS Security Server RACF Diagnosis Guide	GA22-7689	GA32-0886
z/OS Security Server RACF Messages and Codes	SA22-7686	SA23-2291
z/OS Security Server RACF General User's Guide	SA22-7685	SA23-2298

The documentation that follows has been enhanced to reflect the new functions provided. The documentation that follows does not constitute all of the updates that will be made to the RACF publications listed above in cases where the additional modifications contain information that has already been included in this document. This includes references that:

- Assume DES is used.
- State that an ICHPWX11 exit is required to allow a password phrase minimum length as low as nine.
- State that a user must also have a password when assigned a password phrase.
- State that old password history entries cannot be removed when the history value is lowered.
- Do not include special characters when describing allowable password content.

In the following sections, highlighting is used to denote changed information in existing documentation. Sections, tables, messages, command keywords, etc. without highlighting contain new information.

4.1 z/OS Security Server RACF Security Administrator's Guide

This information supplements the information in Chapter 5, **Specifying RACF options**, under the following topics:

- Allowing special characters in passwords (PASSWORD option)
- Specifying the encryption method for user passwords

4.1.1 Allowing special characters in passwords (PASSWORD option)

If you have the SPECIAL attribute, you can allow a set of special characters to be specified in passwords for all users on this system and on all systems that share the RACF database. Use the SETROPTS PASSWORD(SPECIALCHARS) option to allow special characters in passwords at your installation.

```
SETROPTS PASSWORD(SPECIALCHARS)
```

Restriction: The ISPF panels do not support the SETROPTS option to activate and deactivate special character support. For this, you must use the SETROPTS command with the PASSWORD option.

Enabling special characters allows the following characters to be specified in RACF passwords.

Hexadecimal value	Symbol (using the EBCDIC 1047 code page)
4B	.
4C	<
4E	+
4F	|
50	&
5A	!
5C	*
60	-
6C	%
6D	_
6E	>
6F	?
7A	:
7E	=

By default, NOSPECIALCHARS is in effect and special characters are not supported. If you want to allow special characters, be sure that they are permitted by your password syntax rules. Syntax rules can be created to require special characters.

The new password exit (ICHPWX01) can be used to further restrict this set when you have characters that are known to present problems with applications that you use.

User considerations: When you activate the SPECIALCHARS option, be aware of the following considerations.

- Special characters can make passwords more secure and harder to guess. Users are encouraged to select special characters.
- Certain characters might pose problems for certain applications. Avoid using such characters when possible.
- Certain characters have different character representations in different code pages. This might present problems when logging in with a different terminal than you normally use, for example, while traveling internationally. Avoid the use of such characters, when necessary.

RRSF considerations for mixed-case passwords: Be careful when RRSF nodes do not have the same settings in effect for the special characters option of the SETROPTS PASSWORD command. This can occur when one of the nodes is a downlevel system that does not have support for APAR OA43999 applied, or when the nodes have differing settings in effect for the SPECIALCHARS option of the SETROPTS PASSWORD command. When this is the case, message IRRI006I is issued when the RRSF connection is established between the nodes.

The following rules apply when RRSF nodes do not have the same special character setting in effect and a password with special characters is propagated to a system that does not have support for APAR OA43999 applied, or on which special characters are not enabled:

1. The propagation fails when it occurs by using automatic command direction with the ADDUSER, ALTUSER, and PASSWORD commands.
2. The propagation succeeds when it occurs by using automatic password direction (with RACROUTE REQUEST=VERIFY/X, RACROUTE REQUEST=EXTRACT,TYPE=REPLACE, or ICHEINTY).
    - The user continues to be able to LOGON with this password.
    - The user cannot continue to be able to change the password using the PASSWORD command if support for APAR OA43999 is not applied, but is able to if the support is applied, even if SPECIALCHARS is not enabled.

- The user is able to change the password during LOGON.

Guideline: Apply support for APAR OA43999 where necessary and enable SPECIALCHARS at the same time on all RRSF nodes.

4.1.2 Specifying the encryption method for user passwords

RACF keeps user authentication data secure when it is stored in the RACF database. This authentication data can be a password, password phrase, or operator identification card (OIDCARD) data. In this section, passwords and password phrases are referred to collectively as “passwords”.

RACF supports several different methods to encrypt authentication data:

1. Masking
2. The data encryption standard (DES) algorithm.
3. An installation-defined method implemented using the ICHDEX01 exit.
4. The KDFAES (key derivation function with AES256) algorithm (for passwords).

Encoding functions performed by RACF are:

- Data encoding
- Data comparison

Encoding means that, given data in clear text and given an encryption key (which RACF constructs), the equivalent data is produced in encrypted form. RACF provides a "one-way" encoding. That is, data encrypted by RACF can only be decoded if the data is already known. For more information, see z/OS Security Server RACF System Programmer's Guide.

Comparison means that, given authentication data as entered by a user (in clear text form) and given that data as stored in the RACF database in encoded form, an indication whether they are equal or not is returned.

By default, RACF uses the DES algorithm to encrypt authentication data. By default, RACF uses DES and masking to compare authentication data. That is, a first attempt is made using DES and if that comparison fails, a second attempt uses masking. This is a historical behavior that eased the transition from masking to DES.

The DES, masking, and installation-defined methods are collectively referred to as “legacy” methods in some contexts. When KDFAES is not enabled, the presence and function of the ICHDEX01 exit determines which of these algorithms is active.

The SETROPTS command is used to enable KDFAES.

```
SETROPTS PASSWORD(ALGORITHM(KDFAES))
```

Restriction: The ISPF panels do not support the SETROPTS option to activate and deactivate KDFAES. For this, you must use the SETROPTS command with the PASSWORD option.

KDFAES is the strongest algorithm and should be used when possible. This algorithm is resilient against offline brute-force password attacks if your RACF database or one of its copies is compromised.

KDFAES is used for passwords, but not for OIDCARD data. When KDFAES is active, OIDCARD data continues to be protected by a legacy algorithm. When KDFAES is enabled, existing DES and installation-encoded passwords continue to evaluate correctly. When they are changed, they are encrypted using KDFAES. When KDFAES is active, the masking algorithm is no longer used to evaluate legacy passwords, and masked passwords must be changed by the administrator after KDFAES is enabled.

KDFAES passwords require more space in the RACF database than the legacy methods require. This changed format requires updates from some other software applications for them to keep functioning under the new algorithm. Be sure that you have all the available updates before enabling KDFAES.

If you need to disable KDFAES, the SETROPTS command can also be used.

```
SETROPTS PASSWORD(NOALGORITHM)
```

After KDFAES is disabled, all legacy rules apply, however, KDFAES passwords continue to evaluate correctly. When they are changed, they are encrypted using the legacy algorithm in effect, determined by the presence and function of ICHDEX01.

Guideline: Run your system with an ICHDEX01 exit that directs RACF to use only the DES algorithm. Keep this exit active after enabling KDFAES until you are sure that KDFAES never needs to be disabled.

For more information about the ICHDEX01 password authentication exit, see the z/OS Security Server RACF System Programmer's Guide.

Performing a password conversion

After enabling KDFAES, you might want to strengthen passwords immediately, rather than wait for users to change them on their normal schedule. For example, you might want to prove compliance with a security policy that requires the strongest possible algorithm. The PWCONVERT operand of the ALTUSER command can be used. PWCONVERT converts a DES password and DES password history entries to KDFAES format without requiring the password to be changed.

A sample DB2 query against IRRDBU00 output is provided to report users that have a legacy format current password or phrase. See the RACDBUQR member of

SYS1.SAMPLIB.

See the z/OS Security Server RACF Command Language Reference for more information about the PWCONVERT function.

Notes:

1. Conversion assumes that the existing password and password history are in DES format. Do not perform the conversion if you have masked or installation-encoded passwords.
2. Conversion does not affect password phrases or phrase history.

4.2 z/OS Security Server RACF Command Language Reference

This information supplements RACF commands.

4.2.1 ADDUSER

KERB

ENCRYPT

…
When a principal's password changes, a key of each allowed type is generated and stored in the principal's user profile. The use of each key is based on the z/OS Network Authentication Service configuration.
...

NOPASSWORD

Specifies that the new user cannot supply an initial log on password when first entering the system. If you specify NOOIDCARD (or you allow this option to default), do not specify PHRASE, and you specify NOPASSWORD, you define a protected user ID that cannot be used to enter the system by any means that requires a password to be specified, such as a TSO log on, CICS signon, or batch job that specifies a password on the JOB statement. Therefore, user IDs that you assign to z/OS UNIX, UNIX daemons, started procedures, applications, servers or subsystems can be protected from being revoked when an incorrect password is entered. If the user attempts to enter the system with a password, the attempt fails. Note that the protected user ID is not revoked because of the failed password attempts even if the SETROPTS PASSWORD(REVOKE) option is in effect.

...

PHRASE('password-phrase')

Specifies the user's initial password phrase. The password phrase you define is a text string of up to 100 characters and must be enclosed in single quotation marks. The password phrase is always set expired, thus requiring the user to change it on initial use.

The following syntax rules apply to all password phrases. You cannot alter these syntax rules but you can specify additional syntax rules if your installation tailors the new-password-phrase exit (ICHPWX11).

Syntax rules for password phrases:

- Maximum length: 100 characters
- Minimum length:
  - 9 characters, when the encryption algorithm is KDFAES or ICHPWX11 is present and allows the new value
  - 14 characters, when ICHPWX11 is not present and the encryption

algorithm is not KDFAES

- Must not contain the user ID (as sequential uppercase or sequential lowercase characters)
- Must contain at least 2 alphabetic characters (A - Z, a - z)
- Must contain at least 2 non-alphabetic characters (numerics, punctuation, or special characters)
- Must not contain more than 2 consecutive characters that are identical
- If a single quotation mark is intended to be part of the password phrase, you must use two single quotation marks together for each single quotation mark.

If the new-password-phrase exit (ICHPWX11) is present, it can reject the specified password phrase. RACF allows password phrases greater than 8 characters when the encryption algorithm is KDFAES, however, ICHPWX11 can enforce any minimum length greater then 8.

If the specified password phrase is accepted, it is made the user's current password phrase and, when SETROPTS PASSWORD(HISTORY) is in effect, it is added to the user's password phrase history.

If you omit PHRASE, no password phrase is assigned. If you enter PHRASE without a password-phrase value, you are prompted for a value unless your TSO session is in NOPROMPT mode.

4.2.2 ALTUSER

Authorization required:

The PWCLEAN and PWCONVERT operands require the system SPECIAL attribute.

The EXPIRED and NOPASSWORD operands can be issued by a user with the SPECIAL attribute at the system or group level, and by the owner of the user's profile.

Password reset authority provided by granting access to the IRR.PASSWORD.RESET resource, the IRR.PWRESET.OWNER.owner resource, or the IRR.PWRESET.TREE.owner resource, does **not** allow the ability to add a password for a user that does not have one.

EXPIRED

Specifies that the new password or password phrase (specified with the PASSWORD or PHRASE keyword) or the new password defaulted by the PASSWORD keyword is marked as expired. Specifying the EXPIRED keyword requires the user to change their new password or password phrase at the next log on or job start.

When specified without PASSWORD and PHRASE, specifies that the existing password and password phrase (if they exist) are to be marked as expired.

...

KERB

ENCRYPT

…
When a principal's password changes, a key of each allowed type is generated and stored in the principal's user profile. The use of each key is based on the z/OS Network Authentication Service configuration.
...

NOOIDCARD

Specifies that the user is not required to supply an operator identification card.

If NOPASSWORD is specified or the user ID already has the NOPASSWORD attribute, and NOPHRASE is specified or the user ID already does not have a password phrase, specifying NOOIDCARD causes this user ID to become a protected user ID. Protected user IDs cannot be used to enter the system by any means that requires a password to be specified, such as TSO log on. If the user attempts to enter the system with a password, the attempt fails.

Protected user IDs can be used for the user IDs associated with the started tasks in ICHRIN03 or the STARTED class.

NOPASSWORD

Specifies that the user cannot supply a password when entering the system. If NOOIDCARD is specified, or the user ID has the NOOIDCARD attribute, and NOPHRASE is specified or the user ID does not have a password phrase, and you specify NOPASSWORD, you change the status of the user ID to protected. Protected user IDs cannot be used to enter the system by any means that requires a password to be specified, such as a TSO log on, CICS signon, batch job that specifies a password on the JOB statement. Therefore, user IDs that you assign to z/OS UNIX, UNIX daemons, started procedures, applications, servers or subsystems can be protected from being revoked when an incorrect password is entered. If the user attempts to enter the system with a password, the attempt fails. Note that the protected user ID is not revoked because of the failed password attempts even if the SETROPTS PASSWORD(REVOKE) option is in effect.

...

PHRASE | NOPHRASE

PHRASE('password-phrase')
See the ADDUSER command for changes.

NOPHRASE
Specifies that the user cannot use a password phrase for authentication. If a password phrase was previously set, the password phrase is cleared. The date of the last password phrase change is also cleared from the user's profile. If NOOIDCARD is specified, or the user ID has the NOOIDCARD attribute, and NOPASSWORD is specified or the user ID has the NOPASSWORD attribute, and you specify NOPHRASE, you change the status of the user ID to protected. See NOPASSWORD for more information.

PWCLEAN | PWCONVERT

PWCLEAN
Performs the following functions:

- Removes residual password and password phrase history entries resulting from lowering the SETROPTS PASSWORD(HISTORY(*n*)) value.
- Reorganizes the history so that an increase in the SETROPTS PASSWORD(HISTORY(*n*)) value takes immediate effect.
- Removes any password history and password phrase history from a PROTECTED user.

When the SETROPTS PASSWORD(HISTORY(*n*)) value is lowered, the residual history entries continue to be used by RACF. PWCLEAN removes these entries.

If the SETROPTS PASSWORD(HISTORY(*n*)) value is **raised**, that the higher number does not immediately take effect, depending on how many times a user has changed their password or password phrase in the past. PWCLEAN reorganizes the history so that the history change takes effect immediately after using PWCLEAN.

PWCLEAN should be used against all user IDs whenever the SETROPTS PASSWORD(HISTORY(*n*)) value is changed. The SEARCH command with the CLIST option provides a way of creating a 'utility' to do this.

PWCONVERT
Performs the following functions:

- Performs the PWCLEAN function.
- If KDFAES is active:
  - If the current password is in legacy format, converts it to KDFAES format.
  - Converts any legacy-format password history entries to KDFAES.
- If KDFAES is not active:
  - Deletes any password and password phrase history entries that are in KDFAES format.

PWCONVERT does nothing with the current password phrase. After KDFAES is enabled, the phrase must be changed before it is encrypted with the new algorithm. Likewise, PWCONVERT does nothing with phrase history entries. They remain in their legacy form until they are replaced in the history.

The IRRDBU00 utility reports on the algorithm that is used to encrypt a user's current password and password phrase, including the number of legacy password history entries. This information can be used to determine the exact user IDs needing an update. This allows for a more efficient conversion than SEARCH with CLIST.

Attention:

1. The existing current password and password history entries are assumed to be encrypted with DES when PWCONVERT encrypts them using KDFAES. If you use masking, or an installation-defined encryption method by use of an ICHDEX01 exit, do not use PWCONVERT. This results in the user being unable to log on until the password is changed. In addition, it results in unusable history entries. That is, a user is able to reuse a password value that is contained in the password history.

2. When password history entries are converted, they can never be converted back to the legacy format. Thus, they are always more expensive to evaluate, and they are not recognized by any systems not containing KDFAES support.

4.2.3 RACLINK

DEFINE(*[node].userid2[/password]* ...)

Specifies that a user ID association is to be formed between *userid1* at the node where the command is issued, and *userid2* at *node*. If you specify more than one *node.userid2* operand, an association is established between *userid1* and each

node.userid2 specified. A user ID association enables RACF users to use command direction and password synchronization.

If the password starts with an asterisk, the entire string (*[node].userid2[/password]*) must be enclosed in single quotation marks.

4.2.4 SETROPTS

The SETROPTS PASSWORD keyword contains new sub-keywords.

ALGORITHM(KDFAES) | NOALGORITHM

ALGORITHM(KDFAES)

Indicates that RACF should start using the KDFAES algorithm to encrypt user passwords and password phrases. After enablement, the existing algorithm continues to be used to evaluate a user's password or password phrase until the user's password or password phrase is changed. The first time a user's password or password phrase is changed, the new algorithm is used from that point forward.

The KDFAES algorithm is more secure than DES, but is more computationally intensive, by design. Carefully review the planning considerations before enabling this option.

The PWCONVERT keyword of ALTUSER can be used to convert a user's password from DES to KDFAES format without requiring the password to be changed.

If ALGORITHM is specified without a sub-operand, it is ignored.

NOALGORITHM

Indicates that the legacy algorithm is used to encrypt passwords. This is the default setting. In this case, the algorithm in effect is determined by the ICHDEX01 exit, with DES being the default if there is no exit installed.

If you deactivate KDFAES after some set of passwords have been encrypted using KDFAES, each password continues to be evaluated using KDFAES. When the password is changed, the legacy algorithm is used from that point forward. Any

history entries that were created with KDFAES continue to be evaluated using KDFAES. The PWCONVERT keyword of ALTUSER can be used to delete KDFAES history entries, if you want, after reverting to DES.

RULE*n*

The content-keyword section adds two new keywords.

SPECIAL

Includes the special characters documented under SETROPTS PASSWORD(SPECIALCHARS) including the national characters # (X'7B'), $ (X'5B'), and @ (X'7C').

MIXEDALL

Includes all allowable password characters that are separated into the following categories. There are either three or four "active" categories, depending on whether SETROPTS PASSWORD(MIXEDCASE) is enabled.

1. The national characters, and special characters if SETROPTS PASSWORD(SPECIALCHARS) is in effect.
2. Numeric characters.
3. Uppercase alphabetic characters (not including the national characters).
4. Lowercase alphabetic characters, if SETROPTS PASSWORD(MIXEDCASE) is in effect.

MIXEDALL is intended to force a mixture of character types that can include special characters. MIXEDALL requires a character from as many different active categories as there are MIXEDALL positions that are specified, in any combination:

- When one MIXEDALL position is specified, any character from any active category may be specified in that position. This is equivalent to not specifying a content-keyword in this position.
- When two MIXEDALL positions are specified, two characters from any two different active categories must be specified in the designated positions.
- When three MIXEDALL positions are specified, three characters from any three different active categories must be specified in the designated positions.

- When four or more MIXEDALL positions are specified, and SETROPTS PASSWORD(MIXEDCASE) is enabled, then at least one of every category must be specified anywhere across the designated positions. If MIXEDCASE is not enabled, then there is no change in behavior from having three MIXEDALL positions, other than in the number of positions over which the three active categories might be spread.

NOVOWEL

Includes characters that are not vowels, such as

- Uppercase alphabetic characters that are consonants, not vowels.
- National and special characters.
- Numeric characters.

SPECIALCHARS | NOSPECIALCHARS

SPECIALCHARS

Indicates that all applications on this system and those that share the RACF database support additional special characters in passwords. For a list of the characters, and other considerations, see the z/OS Security Server RACF Security Administrator's Guide.

This option is inactive by default.

NOSPECIALCHARS

Indicates that special characters are not allowed in passwords. This is the default setting. If NOSPECIALCHARS is specified after you started using special characters in passwords, you can continue to log on with your existing password, but cannot include special characters in the new password when the password is changed.

LIST

The following sample output demonstrates how the new settings are displayed. If KDFAES is not enabled, the displayed algorithm is “LEGACY”.

```
PASSWORD PROCESSING OPTIONS:
  THE ACTIVE PASSWORD ENCRYPTION ALGORITHM IS KDFAES
```

```
PASSWORD CHANGE INTERVAL IS  30 DAYS.
PASSWORD MINIMUM CHANGE INTERVAL IS   0 DAYS.
MIXED CASE PASSWORD SUPPORT IS IN EFFECT
SPECIAL CHARACTERS ARE ALLOWED.
NO PASSWORD HISTORY BEING MAINTAINED.
USERIDS NOT BEING AUTOMATICALLY REVOKED.
PASSWORD EXPIRATION WARNING LEVEL IS  15 DAYS.
INSTALLATION PASSWORD SYNTAX RULES:
  RULE 1  LENGTH(1:8)    xxs***xx
 LEGEND:
  A-ALPHA C-CONSONANT L-ALPHANUM N-NUMERIC V-VOWEL W-NOVOWEL *-ANYTHING
  c-MIXED CONSONANT m-MIXED NUMERIC v-MIXED VOWEL $-NATIONAL s-SPECIAL
  x-MIXEDALL
```

HISTORY | NOHISTORY

HISTORY(*number-of-previous-values***)**

…

If you increase the HISTORY number, RACF saves and compares that number of passwords and password phrases to the new intended value. If you subsequently reduce the HISTORY number, any previous passwords and password phrases stored in the user profile in excess of the newly specified HISTORY number are not deleted and continue to be used for comparison.

For example, if you specify 12 for your HISTORY number and subsequently reduce it to 8, RACF compares the old passwords and password phrases 9 - 12 with the new intended value.

Attention: You should use ALTUSER PWCLEAN to clean up history entries for all users any time you change the HISTORY value.

If you specify HISTORY, INITSTATS must be in effect.

NOHISTORY

Specifies that new password and password phrase values are only compared with the current password or password phrase. If prior history information exists in the user profile, it is neither deleted nor changed. ALTUSER PWCLEAN can be used to delete history from USER profiles when NOHISTORY is in effect.

NOHISTORY is in effect when RACF is using a newly initialized database.

4.3 z/OS Security Server RACF Messages and Codes

This information supplements RACF messages.

4.3.1 RACF processing messages

ICH408I LOGON/JOB INITIATION - INVALID PASSWORD

Explanation: A user attempted to log on or submit a job using a password that is not valid or belongs to a user without a password (for example, a protected user ID or a phrase-only user ID).

System Action: RACF prevents the user from logging on or the job from executing.

User Response: Correct any spelling errors in the password and try again. If you cannot remember your password, ask your RACF security administrator to provide you with a new password.

IRR420I ERROR *error-code* DURING *operation-name* OF *field-name* FOR *userID*. DIAG CODE 1=*diag-code1*, DIAG CODE 2=*diag-code2*. OCCURRENCES *error-count*.

Explanation: An internal error occurred during an attempt to encrypt a password or password phrase for user *userID*. Operation-name is either “VALIDATE” or “CREATE”. Field-name indicates the name of the RACF database template field being referenced. RACF only issues IRR420I once per minute. The variable error-count indicates how many times an encryption error occurred since IPL. This count is cumulative for all user IDs and all diagnostic codes.

System action: RACF stops processing the request.

System programmer response: If SETROPTS PASSWORD(ALGORITHM(KDFAES)) is not active, and *error-code* is 132, then you might be able to fix the error by issuing the following two commands:

```
SETROPTS PASSWORD(ALGORITHM(KDFAES))
SETROPTS PASSWORD(NOALGORITHM)
```

Otherwise, report this message to the IBM support center.

Destination: Descriptor code is 4. Routing code is 9.

4.3.2 ADDUSER command messages

ICH01024I User *userID* is defined as PROTECTED.

Explanation: When the ADDUSER command is specified with NOPASSWORD and PHRASE to define a phrase-only user, but the specified phrase value is not valid, the user is defined as a PROTECTED user ID.

System Action: RACF creates a user without a password or phrase. The user cannot log on using a password or phrase unless one is then assigned.

User Response: Use the ALTUSER command to assign a valid password phrase.

4.3.3 ALTUSER command messages

ICH21007I EXPIRED/NOEXPIRED OPERAND IGNORED

Explanation: One of the following situations occurred:

1. You specified the NOEXPIRED operand but, the PASSWORD or PHRASE operand is specified on the command. NOEXPIRED is valid only if specified with the PASSWORD or PHRASE operand.
2. You specified the EXPIRED operand with both the NOPASSWORD and NOPHRASE operands.

System Action: RACF ignores the operand and continues command processing with the next operand.

ICH21043I {PWCLEAN|PWCONVERT} REJECTED FOR USER *userID* DUE TO A CONCURRENT PASSWORD CHANGE BY ANOTHER TASK.

Explanation: The ALTUSER command attempted to clean or convert the password or password phrase history for user *userID*. Between the time the history is read from the user's profile and the time the modified history is to be written to the profile, the user's password or password phrase is changed by another user. Storing the cleaned or converted history would result in the loss of the new history entry that the other change created.

System action: The history is not updated in the user's profile.

User response: Reissue the ALTUSER command.

ICH21044I PWCONVERT encountered internal error *error-code* and diagnostic code1=*diag-code1* and code2=*diag-code2* while processing user *userID*.

Explanation: An internal error occurred during an attempt to process the PWCONVERT keyword.

System action: RACF ignores the operand and continues processing with the next operand.

System programmer response: Report this message to the IBM support center.

User response: Report this message to your system programmer.

4.3.4 PASSWORD command messages

ICH08008I *userid* NOT DEFINED TO USE A PASSWORD [PHRASE].

Explanation: The indicated user ID is defined to RACF but does not have a password or password phrase, as indicated in the message.

System Action: No command processing is performed.

4.3.5 IBM health checker for z/OS and sysplex messages

IRRH293E KDFAES encryption is not enabled on this system.

Explanation: The RACF_ENCRYPTION_ALGORITHM check verifies that only the KDFAES or DES encryption algorithm is used for password protection. When KDFAES encryption is not enabled, an exception is raised.

See the z/OS Security Server RACF System Programmer's Guide for more information about the ICHDEX01 exit.

System Action: The check continues processing. There is no effect on the system.
Operator Response: Report this problem to the system security administrator.
System Programmer Response: None.
Problem Determination:
Reference Documentation:
z/OS Security Server RACF System Programmer's Guide
Automation: None.

IRRH294I KDFAES encryption is enabled on this system. If present, ICHDEX01 is used only for password history.

Explanation: The RACF_ENCRYPTION_ALGORITHM check verifies that only the

KDFAES or DES encryption algorithm is used for password protection. When KDFAES encryption is enabled, the ICHDEX01 exit is used only for password history and message IRRH296I is issued.

See the z/OS Security Server RACF System Programmer's Guide for more information about the ICHDEX01 exit.

System Action: The check continues processing. There is no effect on the system.
Operator Response: None.
System Programmer Response: None.
Problem Determination:
Reference Documentation:
z/OS Security Server RACF System Programmer's Guide
Automation: None.

4.4 z/OS Security Server RACF RACROUTE Macro Reference

This information supplements RACROUTE macros.

4.4.1 RACROUTE REQUEST=EXTRACT: Replace or retrieve fields and RACXTRT macro (standard form)

...

Programming interface information:

1. Use caution when using these interfaces as general programming interfaces for product code because they might not be supported by security products other than RACF.
2. For REQUEST=EXTRACT, the following functions are the only suggested programming interfaces:
    a) Retrieving or updating fields in any other product segment (including WORKATTR) in the user, group, and resource profiles.
    b) Retrieving the following installation-reserved fields:
        - USERDATA
        - USRCNT
        - USRDATA
        - USRFLG
        - USRNM
    c) Retrieving the current or a specified user's default group or password when the password is in legacy format (encoded with DES, masking, or an installation-defined method).
    d) Retrieving the member list from a SECLABEL profile.
3. The following functions of RACROUTE REQUEST=EXTRACT are programming interfaces, but are not suggested.
    - Retrieving or updating fields, other than the APPLDATA field, in the BASE segment of a user, resource, or group profile. The APPLDATA field can be retrieved and updated.

...

4.4.2 RACROUTE REQUEST=EXTRACT (standard form)

,ENCRYPT=(*data addr***,DES)**
,ENCRYPT=(*data addr***,HASH)**
,ENCRYPT=(*data addr***,INST)**

,ENCRYPT=(*data addr***,STDDES)**
specifies the user-authentication key and authentication method.

A new note is added:

Note: If SETROPTS PASSWORD(ALGORITHM(KDFAES)) is active and a password is being encrypted for subsequent input to RACROUTE REQUEST=VERIFY or RACROUTE REQUEST=VERIFYX with ENCRYPT=NO, then the password must be encoded using the DES method to be evaluated successfully. If a password is being encrypted for comparison with a password extracted using RACROUTE REQUEST=EXTRACT,TYPE=EXTRACT, the comparison fails if the password is encrypted using the KDFAES algorithm, even if the clear text is correct.

4.4.3 RACROUTE REQUEST=VERIFY (standard form) and RACROUTE REQUEST=VERIFYX (standard form)

ENCRYPT=NO

A new note is added:

Note: If a RACF password is encrypted using KDFAES, then the data that is specified by the PASSWRD= keyword must be encoded using the DES method to be evaluated successfully. If SETROPTS PASSWORD(ALGORITHM(KDFAES)) is active, then the data that is specified by the NEWPASS= keyword must be encoded using the DES method to create a new password that is correctly evaluated.

4.4.4 RACINIT (standard form)

ENCRYPT=NO

A new note is added:

Note: If a RACF password is encrypted using KDFAES, then the data that is specified by the PASSWRD= keyword must be encoded using the DES method to be evaluated successfully. If SETROPTS PASSWORD(ALGORITHM(KDFAES)) is active, then the data that is specified by the NEWPASS= keyword must be encoded using the DES method to create a new password that is correctly evaluated.

4.5 z/OS Security Server RACF Callable Services

This information supplements RACF callable services.

4.5.1 R_Admin (IRRSEQ00)

The field name table for SETROPTS administration in Appendix A is updated with new fields:

Field name	Flag byte value	SETROPTS keyword reference
PWDALG	'Y'	PASSWORD (ALGORITHM (xx))
	'N'	PASSWORD (NOALGORITHM)
PWDSPEC (boolean)	'Y'	SPECIALCHARS
	'N'	NOSPECIALCHARS
RULE*n* ...		
NOTE: When specifying the 'Y' flag, the date supplied in the RULE*n* field consists of a length field and a character sequence, separated by a blank. The length field can be either a single numeric value, or two numeric values separated by a colon (:) to denote a minimum and maximum length. The character sequence conforms to the format of the output of the SETROPTS LIST command. It is a string of 1 to 8 characters, where each position of the string contains a character that indicates the valid characters that can occupy that position: • A - Alphabetic • C - Consonant • c - Mixed consonant • L - Alphanumeric • m - Mixed numeric • N - Numeric • V - Vowel • v - Mixed vowel • W - Non-vowel • * - Any character • $ - National • s – Special character • x – Mixed all For example, if the RULE1 field is specified with field data of "3:6 A*NV*A", the resulting SETROPTS PASSWORD keyword would be RULE1(LENGTH(3:6) ALPHA(1 6) NUMERIC(3) VOWEL(4)). See the z/OS Security Server RACF Command Language Reference for more information about SETROPTS.		

4.5.2 R_kerbinfo (IRRSMK00)

The final sentence in the Parameters section is changed to:

Only the keys for the encryption types allowed for this profile are returned (for example, those indicated in the ENCTYPE field).

4.5.3 R_Password (IRRSPW00): Evaluate or encrypt a clear-text password or password phrase

Function

The R_Password service provides:

1. Evaluate a clear-text password or password phrase (regardless of the encryption algorithm in effect).
2. Encrypt a clear-text password or password phrase under the KDFAES algorithm.

See usage notes for important considerations.

Requirements

Authorization:	Any PSW key in supervisor state
Dispatchable unit mode:	Any task
Cross memory mode:	PASN = HASN or PASN not = HASN
AMODE:	31
RMODE:	Any
ASC mode:	Primary or AR mode
Recovery mode:	ESTAE. Caller cannot have an FRR active
Serialization:	Enabled for interrupts
Locks:	No locks held
Control parameters:	The parameter list and the work area must be in the primary address space. ALETs must be passed for all parameters except the work area. The words containing the ALETs must be in the primary address

	space.

RACF Authorization

None

Format

```
CALL IRRSPW00 (Work_area,
         ALET, SAF_return_code,
         ALET, RACF_return_code,
         ALET, RACF_reason_code,
         Num_parms,
         Parm_ALET,
               Function_code,
               UserID_length,
               UserID,
               Password_length,
               Password,
               Function_parmlist)
```

Parameters

Work_area
The name of a 1024-byte work area for SAF and RACF usage. The work area must be in the primary address space.

ALET
The name of a word containing the ALET for the following parameter. Each parameter must have an ALET specified. Each ALET can be different. The words containing the ALETs must be in the primary address space.

SAF_return_code
The name of a 4-byte area in which the SAF router returns the SAF return code.

RACF_return_code
The name of a 4-byte area in which the service routine stores the return code.

RACF_reason_code
The name of a 4-byte area in which the service routine stores the reason code.

Num_parms
Specifies the name of a 4-byte area that contains the total number of parameters in the parameter list. It must be initialized to 15.

Parm_ALET
The name of a 4-byte area containing the ALET for the remaining parameters in the parameter list and any data areas referenced by parameter list pointers. The word containing the ALET must be in the primary address space.

Function_code
The name of a 2-byte area containing the Function code. The function code must be one of the following values:

- X’0001’ - Verify a user's current password or phrase.
- X’0002’ - Generate an encrypted password or password phrase hash.

UserID_length
The name of a 4-byte area containing the length of the user ID associated with the password. This value must be greater than zero and less than or equal to 8.

UserID
The name of an area containing the user ID.

Password_length
The name of a 4-byte area containing the length of the password. This value must be greater than zero and less than or equal to 100. If the length is less than 9, RACF assumes it is a password. If the length is 9 or greater, RACF assumes it is a password phrase.

Password
The name of an area containing the password or password phrase.

Function_parmlist
Specifies the name of the required function code specific parameter list area for the Function code specified.

Structure for X'0001': Verify a user's current password or phrase.

Offset	Length	Name	Description
0	0	XPWD_VFY_PLIST	Name of structure
0	4	XPW_VFY_OPTIONS	Option flags. Undefined bits must be set to binary zeroes.
		x'80000000'	Perform password expiration and user revocation checking.

Structure for X'0002': Generate an encrypted password or password phrase hash

Offset	Length	Name	Description
0	0	XPWD_GEN_PLIST	Name of structure
0	4	*	Reserved. Must be initialized to zeroes.
4	4	XPWD_GEN_CRYPT1_LEN	On input: Length of the area to hold the first encrypted output value On output: Actual/required length of the first encrypted output value.
8	4	*	Reserved. Must be initialized to zeroes.
12	4	XPWD_GEN_CRYPT1@	Address of pre-allocated output area in which RACF returns the first part of the encrypted value. The area must be at least 8 bytes. This value may be stored in the PASSWORD or PHRASE field of the RACF database using ICHEINTY.
16	4	XPWD_GEN_CRYPT2_LEN	On input: Length of the area to hold the second encrypted output value On output: Actual/required length of the second encrypted output value.
20		*	Reserved. Must be initialized to zeroes.

24	4	XPWD_GEN_CRYPT2@	Address of pre-allocated output area in which RACF returns the second part of the encrypted value. The area must be at least 40 bytes. This value may be stored in the PWDX or PHRASEX field of the RACF database using ICHEINTY.

Return and reason codes

IRRSPW00 returns the following values in the reason and return code parameters:

SAF return code	RACF return code	RACF reason code	Explanation
0	0	0	The service is successful
4	0	0	RACF is not installed
8	8	4	For function code 1, the user profile is not defined to RACF.
8	8	8	For function code 1, the password or password phrase is not authorized
8	8	12	For function code 1, the password or password phrase has expired.
8	8	28	For function code 1, the user's access has been revoked.
8	8	36	For function code 1, the user's access to the default group has been revoked.
8	8	*n*	A RACROUTE REQUEST=VERIFY failed with the indicated return code.
8	12	*n*	Parameter list error. The reason code *n* indicates the problem encountered. Values are:
		8	Invalid number of parameters
		10	Invalid function code
		11	Invalid user ID length
		13	Invalid password length
		15	A reserved area in the function-specific parameter list is not 0
8	16	*n*	An internal error occurred during RACF ICHEINTY processing. The reason code may be useful to IBM service.
8	20	*n*	An internal error occurred during RACF password encryption processing. The reason code may be useful to IBM service.

8	24	4	XPWD_GEN_CRYPT1_LEN too small. The required length is returned in that field.
8	24	8	XPWD_GEN_CRYPT2_LEN too small. The required length is returned in that field.
8	24	16	Recovery could not be established
8	24	20	An abend occurred
8	92 (X'5C')	0483yyyy	An error occurred while RACROUTE REQUEST=VERIFY is accessing the RACF data base. "yyyy" is the RACF manager return code associated with the abend that would have been issued.

Usage notes

1 The password evaluation service is an optimized check to see that the specified password or phrase matches the one stored in the RACF database for the specified user. It also optionally provides password expiration and user revocation checking. When the caller requests the extra checking, and the request fails, or caching does not find a match, a RACROUTE REQUEST=VERIFY is issued. When the extra checking is not requested, no RACROUTE is issued.

2 The encrypted password that is returned by function code 2 cannot be used for subsequent authentication (for example, RACROUTE REQUEST=VERIFY/X or initACEE) of the specified user. It is a means of temporarily encrypting a clear-text password to protect its confidentiality. It can be stored in the RACF database using the ICHEINTY interface.

Related Services

None

4.6 z/OS Security Server RACF Macros and Interfaces

This information supplements RACF macros.

4.6.1 RACF database unload utility (IRRDBU00) records

The user basic data record (0200) has a comment change to the existing USBD_NOPWD field, and is extended to provide the other new fields shown below:

Field Name	Type	Start	End	Comments
...				
USBD_NOPWD	Char	391	394	"YES" indicates that this user ID can log on without a password using OID card. "NO" indicates that this user must specify a password. "PRO" indicates a protected user ID. "PHR" indicates that the user has a password phrase. See also z/OS Security Server RACF Security Administrator's Guide. **Note:** USBD_PWD_ALG and USBD_PHR_ALG are the suggested fields to query to determine what combination of password and password phrase exists for a user.
...				
USBD_PWD_ALG	Char	592	603	Algorithm used to protect password. Values include “LEGACY”, “KDFAES”, and “NOPASSWORD”.
USBD_LEG_PWDHIST_CT	Int	605	607	Number of legacy password history entries
USBD_XPW_PWDHIST_CT	Int	609	611	Number of non-legacy (e.g. KDFAES) password history entries
USBD_PHR_ALG	Char	613	624	Algorithm used to protect password phrase. Values include “LEGACY”, “KDFAES”, and “NOPHRASE”.
USBD_LEG_PHRHIST_CT	Int	626	628	Number of legacy password phrase history entries
USBD_XPW_PHRHIST_CT	Int	630	632	Number of non-legacy (e.g. KDFAES) password phrase history entries

A sample query is provided in the RACDBUQR member of SYS1.SAMPLIB to list the user IDs that have a LEGACY format current password or password phrase.

4.6.2 SMF records

Format of SMF type 80 records: data type 6 command-related data

New fields an flags are added for the ALTUSER and SETROPTS commands.

Event Code dec (hex)	Command	Data Length	Format	Description	
13 (D)	ALTUSER	* The data for event code 13 is identical to the data for event code 10, with these exceptions.			
: etc :					
		2	Binary	Flags for additional keywords specified:	
				Bit Byte 0	**Keyword specified**
				0	SECLEVEL
				1	NOSECLEVEL
				2	SECLABEL
				3	NOSECLABEL
				4	NOEXPIRED
				5	EXPIRED
				6	RESTRICTED
				7	NORESTRICTED
				Byte 1	
				0	NOREVOKE
				1	NORESUME
				2	PHRASE
				3	NOPHRASE
				4	*PWCLEAN
				5	*PWCONVERT
				6 - 7	Reserved for IBM's Use
		2	Binary	Flags for additional keywords ignored (authorization):	
				Bit Byte 0	**Keyword specified**
				0	SECLEVEL

				1	NOSECLEVEL
				2	SECLABEL
				3	NOSECLABEL
				4	NOEXPIRED
				5	EXPIRED
				6	RESTRICTED
				7	NORESTRICTED
				Byte 1	
				0	NOREVOKE
				1	NORESUME
				2	PHRASE
				3	NOPHRASE
				4	*PWCLEAN
				5	*PWCONVERT
				6 - 7	Reserved for IBM's Use
		2	Binary	Flags for additional keywords ignored because of processing error:	
				Bit Byte 0	**Keyword specified**
				0	SECLEVEL
				1	NOSECLEVEL
				2	SECLABEL
				3	NOSECLABEL
				4	NOEXPIRED
				5	EXPIRED
				6	RESTRICTED
				7	NORESTRICTED
				Byte 1	
				0	*PWCLEAN
				1	*PWCONVERT
				2-7	Reserved for IBM's Use
: etc :					

Event Code dec (hex)	Command	Data Length	Format	Description
24 (18)	SETROPTS			
:				

			etc :
	80	Binary Binary EBCDIC	Password syntax rules (eight rules). Each rule has the following basic format: **Byte Description** **0** Starting length value **1** Ending length value **2-9** Character content rules for each of the eight possible positions. The character values are: L = Alphanumeric A = Alphabetic N = Numeric V = Vowel C = Consonant W = No vowels c = Mixed consonant m = Mixed numeric v = Mixed vowel $ = National s = Special x = Mixed all * = Anything
			: etc :
		4	Binary	Flags for keywords specified: **Bit Keyword specified** 0 Primary language specified 1 Secondary language specified 2 ADDCREATOR specified 3 NOADDCREATOR specified 4 LIST specified 5 KERBLVL specified 6 EIMREGISTRY specified 7 NOEIMREGISTRY specified 8 Password MINCHANGE specified 9 Password MIXEDCASE specified 10 Password NOMIXEDCASE specified 11 Password SPECIALCHARS specified 12 Password NOSPECIALCHARS specified 13 Password ALGORITHM specified 14 Password NOALGORITHM specified 15 Reserved for IBM's use 16 MLFSOBJ(ACTIVE) specified 17 MLFSOBJ(INACTIVE) specified 18 MLIPCOBJ(ACTIVE) specified 19 MLFSOBJ(INACTIVE) specified 20 MLNAMES specified 21 NOMLNAMES specified 22 SECLBYSYSTEM specified 23 NOSECLBYSYSTEM specified 24-31 Reserved for IBM's use

		4	Binary	Flags for keywords specified but ignored because of insufficient authority: same format as flags for keywords specified.
		: etc :		
		1	Binary	Current options **Bit Option** **0** Mixed case passwords are allowed **1** Special characters are allowed in passwords **2-7** Reserved for IBM's use
		1	Binary	Password algorithm in effect: 0 = existing algorithm as indicated by ICHDEX01 (masking, DES, or installation-defined) 1 = KDFAES
		75	EBCDIC	Reserved for IBM's use

Record type 81: RACF initialization record

A new field is added at the end:

Dec	Hex	Name	Length	Format	Description
...					
179	B3	SMF81OP6	1	Binary	Options indicator 6 **Bit Meaning when set** 0 Mixed case passwords 1 New password phrase installation exit is active 2 Field validation exit point (IRRVAF01) for custom fields is active Note: The IRRVAF01 exit point is defined to dynamic exit services. Bit 2 of SMF810P6 indicates that an exit routine is active for this exit point at the time of the last IPL

					when the SMF record is written. The status can change either way multiple times throughout the life of the IPL. See z/OS Security Server RACF System Programmer's Guide for more information. 3 Special characters allowed in passwords 4-7 Reserved for IBM's use
180	B4	SMF81ML2	1	Binary	More SETROPTS options **Bit Meaning when set** 0 MLFSOBJ is active 1 MLIPCOBJ is active 2 MLNAMES is active 3 SECLBYSYSTEM is active 4-7 Reserved for IBM's use
181	B5	SMF81ALG	1	Binary	Password encryption algorithm in effect. 0 indicates LEGACY. 1 indicates KDFAES.
182	B6		73		Reserved for IBM's use

4.6.3 RACF SMF unload utility (IRRADU00) records

The unloaded Type 81 initialization record is extended to provide the following fields:

Field name	Type	Length	Start	End	Comments
RINI_PWD_SPECIAL	Yes/No	4	761	764	Are special characters allowed in passwords?
RINI_PWD_ALG	Char	10	766	777	Algorithm used to encrypt passwords and password phrases. Possible values are “KDFAES” and “LEGACY”.

4.6.4 RACF database templates

The following fields are added to the user template for the RACF database. They are not

considered part of the programming interface:

```
PWDX     100 04 80 00000000 00 Password extension
OPWDXCT  101 10 00 00000004 00 Password history extension: count of entries
OPWDXGEN 102 80 00 00000001 FF Password history extension: generation number
OPWDX    103 84 00 00000000 00 Password history extension: password value
PHRASEX  104 04 80 00000000 00 Password phrase extension
PHRCNTX  105 10 00 00000004 00 Phrase history extension: count of entries
OLDPHRNX 106 80 00 00000001 FF Phrase history extension: generation number
OLDPHRX  107 84 00 00000000 00 Phrase history extension: phrase value

OLDPWDX  000 40 00 102 103 000 000 000  Alias for extended pwd history entry
OLDPHREX 000 40 00 106 107 000 000 000  Alias for extended phrase history entry
```

The definition of the existing two fields is changed:

```
PASSDATE 013 00 A0 00000003 FF DATE OF PASSWORD CHANGE
PHRDATE  088 00 A0 00000003 FF Date the Pass Phrase was last changed
```

4.6.5 ICHEINTY, ICHETEST, and ICHEACTN macros

There is a new return code defined for the ICHEINTY macro:

Hex (Dec) Description
88 (136)
Internal error during encryption of a field.

4.7 z/OS Security Server RACF Data Areas

This information supplements RACF data areas.

4.7.1 PWXP: Password Exit Parameter List

Offset (dec)	Offset (Hex)	Type	Len	Name(Dim)	Description
...					
48	30	ADDRESS	4	PWXPWHST	Password history address: points to an area containing the user's password history. The passwords are in masked or encrypted format, with the oldest password first in the list. The format of the area is: a 2-byte count of the entries in the list, and for each entry a 1-byte reserved field followed by an 8-byte field containing the encrypted password. The SETROPTS PASSWORD(HISTORY(*n*)) option controls the number of past keywords that are kept. **Note:** This address is 0 when RCVTPALG is not 0.
...					

4.7.2 RCVT: RACF Communication Vector Table

RCVTPALG, RCVTPSC, and RCVTXPWD are added to the list of intended Programming Interface fields.

Offset (dec)	Offset (Hex)	Type	Len	Name(Dim)	Description
...					
488	1E8	CHARACTER	8	*	RESERVED
496	1F0	ADDRESS	4	RCVTMXP0	Address of enhanced password routine (IRRMXPW0)
500	1F4	CHARACTER	96	*	RESERVED

...					
633	279	BITSTRING	1	RCVTFLG3	MISCELLANEOUS FLAGS
...					
		…. 1...		RCVTPSC	Special characters are allowed in passwords
		…. .1..		RCVTXPWD	Extended password support provided by OA43999 is available
		…. ..11		*	RESERVED
...					
635	27B	UNSIGNED	1	RCVTPALG	Password algorithm in effect: 0 = DES or the algorithm as indicated by ICHDEX01 (masking, DES, or installation-defined) 1 = KDFAES
636	27C	UNSIGNED	2	RCVTPMEM	Password algorithm memory factor
638	27E	UNSIGNED	2	RCVTPREP	Password algorithm iteration factor
640	280	CHARACTER	56		Reserved
...					

4.8 z/OS Security Server RACF Diagnosis Guide

This information supplements the Inventory Control Block (ICB).

4.8.1 The Inventory Control Block (ICB)

Offset (dec)	Offset (Hex)	Type	Len	Name (Dim)	Description
...					
345	159	BITSTRING	4	ICBMOPT	Miscellaneous options
...					
		…. ..1.		ICBPSC	Special characters are allowed in passwords
		…. ...1			Reserved
...					
2600	A28	UNSIGNED	2	ICBPREP	Password algorithm repetition factor
2602	A2A	UNSIGNED	2	ICBPMEM	Password algorithm memory factor
2604	A2C	UNSIGNED	1	ICBPALG	Password algorithm in effect: 0 = DES or the algorithm as indicated by ICHDEX01 (masking, DES, or installation-defined) 1 = PBKDF2
2605	A2D	CHARACTER		ICBRSVD	RESERVED

4.9 z/OS Security Server RACF General User's Guide

The information for this book is addressed by the general statement at the beginning of section 4.

4.10 z/OS Security Server RACF System Programmer's Guide

The performance- and exit-related information for this book is provided in the Planning section above. Additional changes are addressed by the general statement at the beginning of section 4.

5 Trademarks

IBM®, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.