

# DSNT2PRM Utility

**Author: Jun Ogata**

**z/OS Security Server RACF**

**ogata@us.ibm.com**

<u>Change Date</u>	<u>Change Description</u>
08/17/2017	Original documentation.
11/27/2017	Updated for comments provided.

## **Overview:**

The DSNT2PRM utility is a REXX program that converts an existing ICHRDSNT (Data Set Name Table) and/or ICHRRNG (RACF Range Table) load modules into a RACF parmlib member. It can also be used to inspect the in-storage RACF data set descriptor table (ICHPDSDT) and RACF database range table (RNG) and convert them into a parmlib member.

The generated parmlib member can then be used with the RACF parmlib member support introduced in z/OS V2R3 to enable the system to IPL with the generated member.

## **Package Contents:**

This package contains:

- This PDF file with detailed information about DSNT2PRM.
- DSNT2PRM REXX program.

## **Installation Instructions:**

Download and save the DSNT2PRM REXX program in a cataloged partitioned data set.

## **Syntax:**

DSNT2PRM can be used in the following ways:

1. **Reading the in-storage data set name table and range table.** If an output data set is provided, the results are stored there. If no output data set is specified, or if it is an asterisk (\*), the output is displayed to the user's screen.

```
EX "some.dataset(DSNT2PRM) " "[<output.dataset(member)> | *]  
[- SCREENOUT]"
```

2. **Processing only the provided ICHRDSNT member.** If an output data set is provided, the results are stored there. If the output data set is an asterisk (\*), the output is displayed to the user's screen.

```
EX "some.dataset(DSNT2PRM) " "<output.dataset(member)> | *  
<input.ichrdsnt(member)>  
[- SCREENOUT]"
```

3. **Processing only the provided ICHRRNG member.** If an output data set is provided, the results are stored there. If the output data set is an asterisk (\*), the output is displayed to the user's screen. Note that the 2<sup>nd</sup> parameter must be an asterisk (\*) to skip the processing of the ICHRDSNT.

```
EX "some.dataset(DSNT2PRM) " "<output.dataset(member)> | *  
*  
<input.ichrrng(member)>  
[- SCREENOUT]"
```

4. **Processing both the provided ICHRDSNT & ICHRRNG members.** If an output data set is provided, the results are stored there. If the output data set is an asterisk (\*), the output is displayed to the user's screen.

```
EX "some.dataset(DSNT2PRM) " "<output.dataset(member)> | *  
<input.ichrdsnt(member)>  
<input.ichrrng(member)>  
[- SCREENOUT]"
```

## **Where:**

**<output.dataset(member)>** is the data set member that you want to save the generated parmlib statements in. If this value is not specified (only valid in case 1), or is an asterisk (\*), the generated output ONLY goes to the user's screen and a warning message will be issued. If a data set member is provided, the generated output is not displayed to the user's screen, unless the SCREENOUT optional parameter is provided.

**<input.ichrdsnt(member)>** is the ICHRDSNT load module that you want to convert. The member name must be provided. DSNT2PRM makes no assumption for the member name. If this value is an asterisk (\*), DSNT2PRM skips the processing of the ICHRDSNT load module.

<*input.ichrrng(member)*> is the ICHRRNG load module that you want to convert. The member name must be provided. DSNT2PRM makes no assumption for the member name. If this value is an asterisk (\*), DSNT2PRM skips the processing of the ICHRRNG load module.

Anything after the dash (-) is for the optional parameter used by DSNT2PRM.

SCREENOUT           Generates output to the user's screen when an output data set is provided. If an output data set is not provided, the generated output always goes to the screen.

### **Return Codes:**

00     - Successful.  
04     - Warning situation encountered, but otherwise successful.  
12     - Terminating error was encountered and DSNT2PRM execution ended early.

### **Usage Notes:**

1. All data sets are assumed to exist and are cataloged.
2. All data set names must be fully qualified. The utility does not assume any name prefix.
3. None of the data sets used can be a PDSE. If they are, unpredictable results can occur.
4. If an asterisk (\*) is provided for both input values (the <*input.ichrdsnt(member)*> and <*input.ichrrng(member)*> load module values) or both are not specified, DSNT2PRM creates the parmlib member based on what is in storage.

### **Example 1**

The following is an example of converting the in-storage data set name table and range table, and generating output only to the screen.

```
EX 'RACFDEV.CONVERT(DSNT2PRM) '  
  
/* -----  
--  
-- This PARMLIB member was generated on 09/07/17  
-- by the DSNT2PRM utility on system MYSYS.  
--  
-- In-Storage version of ICHRDSNT & ICHRRNG were used  
-- to generate this PARMLIB member.  
--  
----- */  
DATABASE_OPTIONS  
  
DCU004I Generate PARMLIB data based on ICHRDSNT data.  
      # of DS = 1  
/* ----- */  
SYSPLEX(NOCOMMUNICATIONS)  
DATASETNAMETABLE  
  ENTRY  
    PRIMARYDSN('RACFDRVR.RACF323')  
    BACKUPDSN('RACFDRVR.RACF322')  
    UPDATEBACKUP(ALL)  
    BUFFERS(255)
```

```

DCU005I Generate PARMLIB data based on ICHRRNG data.
      Ranges = 1
/* ----- */
RANGETABLE
  START('00' HEX)
  ENTRYNUMBER(1)

DCU103I INFO:      No OUTPUT data set was specified.
DCU104W WARNING: Using current in-storage Data Set Name Table
      values.  These values may NOT match what you IPLed with.

DCU002W Successful execution of DSNT2PRM, with WARNINGS!  Return code = 4

```

## **Example 2**

The following is an example of generating a parmlib member for a provided data set name table and range table.

```

EX 'RACFDEV.CONVERT(DSNT2PRM)' 'OGATA.PARMLIB(IRRPRMAA) SYS1.LINKLST(IXXRDSNT)
SYS1.LPALIB(IXXRRNG)'

```

```

DCU004I Generate PARMLIB data based on ICHRDSNT data.
      # of DS = 2

```

```

DCU005I Generate PARMLIB data based on ICHRRNG data.
      Ranges = 4

```

```

DCU105I INFO:      Process generated output through RACPRMCK.
DCU001I Successful execution of DSNT2PRM.

```

The generated IRRPRMAA member appears as follows:

```

/* -----
--
-- This PARMLIB member was generated on 09/07/17
-- by the DSNT2PRM utility on system MYSYS.
--
-- ICHRDSNT used, was from
--     SYS1.LINKLST(IXXRDSNT)
-- ICHRRNG used. was from
--     SYS1.LPALIB(IXXRRNG)
-- to generate this PARMLIB member.
--
----- */
DATABASE_OPTIONS
/* ----- */
SYSPLEX(DATASHARING)
DATASETNAMETABLE
  ENTRY
    PRIMARYDSN('RACF.OGATA.RACFDB1')
    BACKUPDSN('RACF.OGATA.RACFBK1')
    UPDATEBACKUP(ALL)
    BUFFERS(50)
  ENTRY
    PRIMARYDSN('RACF.OGATA.RACFDB2')

```

```

BACKUPDSN ('RACF. OGATA. RACFBK2 ')
UPDATEBACKUP (ALL)
BUFFERS (50)
/* ----- */
RANGETABLE
  START ('00' HEX)
    ENTRYNUMBER (1)
  START ('CO' CHAR)
    ENTRYNUMBER (2)
  START ('SABCDEFghi' CHAR)
    ENTRYNUMBER (1)
  START ('E2E8E2C9D6C601C9D6C6C1C3C34BC3C2' HEX)
    ENTRYNUMBER (2)

```

### **Code Page considerations:**

It is possible that the code page that is used to create the ICHRRNG varies from location to location. When DSNT2PRM is processing the ICHRRNG to determine whether to use the CHAR or HEX format, a CHAR string is only generated if the RANGE value consists of non-variant EBCDIC characters.

The *Reference* variable, which is defined within the DSNT2PRM program, contains a list of characters that DSNT2PRM uses to determine if the CHAR format can be used. If DSNT2PRM finds a character value that is not in this list, the RANGE is saved as a hexadecimal string.

If the user wants to add additional characters to avoid their ranges from being saved in hexadecimal format, they can add these characters to the end of *Reference* variable.

## **Messages:**

### **End of execution messages:**

DCU001I Successful execution of DSNT2PRM.

DCU002W Successful execution of DSNT2PRM, with WARNINGS! Return code = *<return\_code>*

DCU003E Unsuccessful execution of DSNT2PRM. Return code = *<return\_code>*

### **Informational processing messages:**

DCU004I Generate PARMLIB data based on ICHRDSNT data.

# of DS = *<entries>*

DCU005I Generate PARMLIB data based on ICHRRNG data.

Ranges = *<ranges>*

### **Exiting informational and warning messages:**

DCU101W WARNING: Currently in READ-ONLY mode.

Assuming SYSPLEX option is supposed to be Datasharing mode.

**Explanation:** The system is currently in READ-ONLY mode, which is not a normal state to IPL in. DSNT2PRM assumes that the user wants to be in data sharing mode (and not READ-ONLY mode) and sets that as the SYSPLEX option.

**User Action:** Verify that SYSPLEX(DATASHARING) is the sysplex mode that you want to IPL with. If not, update the generated parmlib member with the SYSPLEX option that you want to IPL with.

DCU102W WARNING: Currently in a SYSPLEX transition state.

Assuming SYSPLEX option is supposed to be Datasharing mode.

**Explanation:** The system is currently in a transition state between sysplex modes, which is not a normal state to IPL in. DSNT2PRM assumes that the user wants to be in data sharing mode and sets that as the SYSPLEX option.

**User Action:** Verify that SYSPLEX(DATASHARING) is the sysplex mode that you want to IPL with. If not, update the generated parmlib member with the SYSPLEX option that you want to IPL with.

DCU103I INFO: No OUTPUT data set was specified.

**Explanation:** No output data set was specified on execution of the DSNT2PRM program, and all generated information will ONLY go to the user's screen.

**User Action:** If the user's intention was not to save the output, then no further action is required.

If the user expected the generated output to be saved, the user will need to re-execute DSNT2PRM and provide an output data set.

DCU104W WARNING: Using current in-storage Data Set Name Table values. These values may NOT match what you IPLed with.

**Explanation:** It is possible that the in-storage data set name table that is used by DSNT2PRM does not match the ICHRDSNT provided at IPL time. This can be due to various reasons, for example:

- an RVAR SWITCH of the RACF data sets,
- an RVAR DATASHARE/NODATASHARE,
- in a SYSPLEX, the SYSPLEX group data set name table and range table are used, and not the one that is provided at IPL time.

**User Action:** Verify that the generated parmlib member has the desired data set name table values in it.

DCU105I INFO: Verify the generated output using the RACPRMCK command.

**Explanation:** It is recommended that after execution of DSNT2PRM, the generated output is validated though RACPRMCK (the parmlib syntax checker). This is to avoid the unlikely event that DSNT2PRM did not correctly process the input ICHRDSNT or ICHRRNG.

**User Action:** Save the final copy of the generated output in a data set that is part of the parmlib concatenation, then issue the RACPRMCK command.

```
RACPRMCK MEMBER (<member_name>)
```

DCU106W WARNING: DSNT2PRM running on V2R2 release or lower.

**Explanation:** DSNT2PRM noticed that it was executed on a Version 2 Release 2 system or lower where the RACPRMCK command does not exist.

**User Action:** To run the RACPRMCK command, the user must go to a Version 2 Release 3 system or higher and execute it from there.

DCU107W WARNING: Unsupported value for <field>.

**Explanation:** DSNT2PRM noticed that the value provided for the given field is unsupported. This causes DSNT2PRM to generate a value for field in the parmlib member that the user will need to later modify and sets the exit return code to 4.

**User Action:** Review the generated output by the DSNT2PRM program, looking for the “unsupported value” string in the output. Determine how that value should be processed and manually update the parmlib member.

**Exiting failing and terminating messages:**

DCU201E ERROR: Input data set '<input.dataset>' allocation failure.  
ALLOCATE command generated return code = <return\_code>  
Terminating execution of DSNT2PRM.

**Explanation:** DSNT2PRM encountered an allocation error when attempting to open the input data set. This causes DSNT2PRM to end its processing with a failing return code of 12.

Note that DSNT2PRM will still attempt to FREE the input file allocation that was attempted to be defined by the ALLOCATE command as a precaution. So a FREE error maybe seen as part of DSNT2PRM's attempt to cleanup.

**User Action:** Review the ALLOCATE return code, as well as any other allocation messages that might have been generated. Fix the specified issue that caused the input data set allocation error.

DCU202E ERROR: Input data set '<input.dataset>' read error.  
EXECIO command generated return code = <return\_code>  
Terminating execution of DSNT2PRM.

**Explanation:** DSNT2PRM encountered a read error when attempting to read the input data set. This causes DSNT2PRM to end its processing with a failing return code of 12.

**User Action:** Review the EXECIO return code, as well as any other messages that might have been generated. Fix the specified issue that caused the input data set read error.

DCU203E ERROR: Output data set '<output.dataset>' allocation failure.  
ALLOCATE command generated return code = <return\_code>  
Terminating execution of DSNT2PRM.

**Explanation:** DSNT2PRM encountered an allocation error when attempting to open the output data set. This causes DSNT2PRM to end its processing with a failing return code of 12.

Note that DSNT2PRM will still attempt to FREE the output file allocation that was attempted to be defined by the ALLOCATE command as a precaution. So a FREE error maybe seen as part of DSNT2PRM's attempt to cleanup.

**User Action:** Review the ALLOCATE return code, as well as any other allocation messages that might have been generated. Fix the specified issue that caused the output data set allocation error.

DCU204E ERROR: Output data set '<output.dataset>' write error.  
EXECIO command generated return code = <return\_code>  
Terminating execution of DSNT2PRM.

**Explanation:** DSNT2PRM encountered a write error when attempting to write to the output data set. This causes DSNT2PRM to end its processing with a failing return code of 12.



**User Action:** Review the EXECIO return code, as well as any other messages that might have been generated. Fix the specified issue that caused the output data set write error.

DCU205E ERROR: Input data not long enough for the # of elements provided  
in the input data set '<input.dataset>'.  
Terminating execution of DSNT2PRM.

**Explanation:** DSNT2PRM noticed that the length of the module processed in the input load module was not long enough to include the number of elements indicated by the beginning of the table. This inconsistency would cause DSNT2PRM to process information past the end of the module. This causes DSNT2PRM to end its processing with a failing return code of 12.

**User Action:** Verify that the input load module is valid.

DCU206E ERROR: Zero address encountered for '<pointer>'.  
Terminating execution of DSNT2PRM.

**Explanation:** DSNT2PRM encountered a zero address pointer when processing the in-storage data set name table or range table. This causes DSNT2PRM to end its processing with a failing return code of 12.

**User Action:** Verify that the system has a valid and active RACF environment. RACF being inactive can generate this failure.

**Disclaimers, etc.**

This program contains code made available by IBM® Corporation on an AS IS basis. Any one receiving this program is considered to be licensed under IBM copyrights to use the IBM-provided source code in any way he or she deems fit, including copying it, compiling it, modifying it, and redistributing it, with or without modifications, except that it may be neither sold nor incorporated within a product that is sold. No license under any IBM patents or patent applications is to be implied from this copyright license.

This software is provided “as-is”, and IBM disclaims all warranties, express or implied, including but not limited to implied warranties of merchantability or fitness for a particular purpose. IBM shall not be liable for any direct, indirect, incidental, special or consequential damages arising out of this agreement or the use or operation of the software.

A user of this program should understand that IBM cannot provide technical support for the program and will not be responsible for any consequences of use of the program.