Whitepaper

RACADM Error Codes

Abstract
This white paper provides details on RACADM error codes. It serves as a guide that will help you identify various error codes that are displayed while executing RACADM commands.

February 2020

Revisions

Date	Description
February 2020	Initial release

Acknowledgements

This paper was produced by the following:

Authors: Rehan Hudlikar, Sai Suryanarayan, A Anis Ahmed

The information in this publication is provided “as is.” Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Use, copying, and distribution of any software described in this publication requires an applicable software license.

Table of contents

Executive summary

An error code or return code is an integer value that represents the status of a command run. You can identify the failure type based on the error code returned. It is very important to know about the command execution status, whether it was ~~e~~xecuted successfully or failed and in case it fails, the error code will help you identify the reason.

1 Prerequisites

Error codes are supported from both remote and local RACADM starting iDRACTools v9.4.0 along with iDRAC firmware version 4.00.00.00 or later.

2 RACADM Error codes:

Each RACADM command will return an error code. Based on the error codes, you can identify if the command is successful or failed and the reason for the failure.

The following table lists the error codes for RACADM commands:

Error Code	Description
0	RACADM command executed successfully
1	Generic Failure E.g. iDRAC has an internal or unknown error
2	The value specified is out of range. The length of the argument (E.g. Filename/path) is larger or shorter than allowed.
3	Invalid or incorrect RACADM command is entered. Any command or option entered is not supported on the current interface/platform.
4	Syntax of the command is not correct
5	You do not have required privileges to run the command
6	iDRAC does not have a required license installed
7	iDRAC does not have enough resources (E.g. Memory)
8	iDRAC is busy in running similar job/operation
9	Failures (Permission issues, invalid share details, mount failures, etc.) related to network shares (CIFS/NFS/FTP/TFTP/HTTP/HTTPS).
10	File export failure to local shares using local RACADM.
11	iDRAC lockdown mode is enabled or a dependent feature is not enabled, or dependent attributes are invalid/not configured.
12	Unable to connect to iDRAC remotely using remote RACADM
13	All IPMI related failures in local RACADM
14	File transfer failure from remote RACADM to iDRAC.
15	A session related issue
16	Failures due in invalid keys, certificate, and signing error.
17	Argument passed are invalid/incorrect.

2.1 Error codes using remote RACADM:

2.1.1 Windows

After running any remote RACADM command from Windows, run the “echo %errorlevel%” command to get the error code. Following is an example where a get version command was executed and then the command echo %errorlevel% was run to determine whether it was successful or failed.

```
C:\Users>racadm -r <idrac_ip> -I <idrac_username> -p <idrac_password> getversion
Bios Version                        = 2.1.8
iDRAC Version                       = 4.20.20.20
Lifecycle Controller Version        = 4.20.20.20
IDSDM Version                       = NA
C:\Users>echo %errorlevel%
0
C:\Users>
```

In the above example, 0 indicates that the command was successful.

2.1.2 Linux

After running any remote RACADM command from Linux run the “echo $?**”** command to get the error code. Following is an example where a get version command was executed and then the command echo $? was run to determine whether it was successful or failed.

```
[root@localhost~]#racadm -r <idrac_ip> -I <idrac_username> -p <idrac_password>
getversion
Bios Version                     = 2.1.8
iDRAC Version                    = 4.20.20.20
Lifecycle Controller Version     = 4.20.20.20
IDSDM Version                    = NA
[root@localhost~]#echo $?
0
[root@localhost~]#
```

In the above example, 0 indicates that the command was successful.

2.2 Error codes using Local RACADM:

2.2.1 Windows

After running any remote RACADM command from Windows, run the “echo %errorlevel%” command to get the error code. Following is an example where a get version command was executed and then the command echo %errorlevel% was run to determine whether it was successful or failed.

```
C:\Users>racadm getversion
Bios Version                     = 2.1.8
iDRAC Version                    = 4.20.20.20
Lifecycle Controller Version     = 4.20.20.20
IDSDM Version                    = NA
C:\Users>echo %errorlevel%
0
C:\Users>
```

In the above example, 0 indicates that the command was successful.

2.2.2 Linux

After running any remote RACADM command from Linux run the “echo $?” command to get the error code. Following is an example where a get version command was executed and then the command echo $? was run to determine whether it was successful or failed.

```
[root@localhost~]#racadm getversion
Bios Version                     = 2.1.8
iDRAC Version                    = 4.20.20.20
Lifecycle Controller Version     = 4.20.20.20
IDSDM Version                    = NA
[root@localhost~]#echo $?
0
[root@localhost~]#
```

In the above example, 0 indicates that the command was successful.

3 Error Codes in-detail with usage examples:

3.1 Error code 1:

This error code is returned when iDRAC has an internal error or unknown failures.

Example:
Perform racreset

```
 >RACADM racreset
RAC reset operation initiated successfully. This may take a few minutes for the
RAC to come online again.
```

Now, execute any basic command

```
>RACADM getsysinfo
ERROR:  Unable to perform the requested operation.
```

Run the following command, to get the error code:

For Window OS:

```
 C:\Users\Administrator>Echo %errorlevel%
 1
```

For Linux OS:

```
 linux-t2x0:~ # echo $?
 1
```

3.2 Error code 2:

This error code is returned when an out of range value is specified for any argument or when length of an argument in filename or path is larger/shorter than allowed limit.

Example:

```
>RACADM get idrac.users.17
  ERROR: Invalid index value. Only index values 1 - 16 are permitted.

 >racadm set idrac.users.17.UserName root17
 ERROR: Invalid index value. Only index values 1 - 16 are permitted.
```

Run the following command, to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
2
```

For Linux OS:

```
linux-t2x0:~ # echo $?
2
```

3.3 Error code 3:

This error code is returned when an invalid RACADM command is passed or when the command is not supported on platform/interface.

Example:

```
>racadm getmytime
ERROR: Invalid command specified.
>racadm remotteimage
ERROR: Invalid command specified.
```

Run the following command, to get the error code

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
3
```

For Linux OS:

```
linux-t2x0:~ # echo $?
3
```

3.4 Error code 4:

This error code is returned when RACADM command is entered with a wrong syntax.

Example:

```
racadm license export -f File.xml
ERROR: The syntax of the command specified is not correct.


racadm remoteimage -c -d
ERROR: The syntax of the command specified is not correct.
```

Run the following command, to get the error code

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
4
```

For Linux OS:

```
linux-t2x0:~ # echo $?
4
```

3.5 Error code 5:

This error is returned when user does not have enough privileges to run RACADM command.

Example:
Create a new user with Read Only privilege and run any command that requires Configure iDRAC privilege using the new account.

```
>racadm traceroute 192.168.0.2

 ERROR: Insufficient user privileges to perform operation.
```

Run the following command, to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
5
```

For Linux OS:

```
linux-t2x0:~ # echo $?
5
```

3.6 Error code 6:

This error code is returned when a required iDRAC license is not installed.

Example:

Run any RACADM command that requires iDRAC Enterprise license on a system with iDRAC Basic or Express license installed.

```
>racadm remoteimage -c -l http://dell.com/Windows_64.iso
 ERROR: SWC0242 : A required license is missing or expired.
 Obtain an appropriate license and try again, or contact your service
 provider for additional details.
 Required License -- Licensing Management
```

Run the following command, to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
6
```

For Linux OS:

```
linux-t2x0:~ # echo $?
6
```

3.7 Error code 7:

This error code is returned when iDRAC does not have enough resources (E.g. Memory) to complete the operation.

Example:

During exceptional cases when iDRAC is running out of memory, running any command might return the following error.

```
>racadm ping 192.168.0.1
 ERROR: Unable to allocate memory needed to perform operation.
```

Run the following command, to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
7
```

For Linux OS:

```
linux-t2x0:~ # echo $?
7
```

3.8 Error code 8:

This error code is returned when iDRAC is busy running a similar operation.

Example:

Execute a valid RACADM command and try to execute same command before the previous command execution is completed.

Example:

```
>racadm update -f firmimgFIT.d9 -l //192.168.12.11/cifs  -u username -p
password
RAC987: Firmware update job for JID_12345678 is initiated. This firmware
update job may take several minutes to complete depending on the component or
firmware being updated. To view the progress of the job, use the "racadm
jobqueue view" command. If the job is scheduled, the system will require a
manual reboot. To reboot the system manually, use the "racadm serveraction
powercycle" command.
```

Re-run the same command immediately.

```
>racadm update -f firmimgFIT.d9 -l //192.168.12.11/cifs  -u username -p
password
ERROR:RAC988:Unable to initiate the firmimgFIT.d9 firmware update.Wait until
the existing firmware update jobs are completed or scheduled before retrying
the operation. To monitor the status of the existing jobs, use the "racadm
jobqueue view" command.
```

Run the following command, to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
8
```

For Linux OS:

```
linux-t2x0:~ # echo $?
8
```

3.9 Error code 9:

This error code is returned when the share location or credentials to access the share are invalid or there is a permission issue with the network share.

Example:

Run any RACADM export/import command with wrong network share details.

```
>racadm update -f firmimgFIT.d9 -u username -p password -l
//100.97.190.11/cifs
ERROR: RAC0904: The remote file location is not accessible or reachable.
Confirm that the path and credentials to remote file are correct and check
that the iDRAC can connect to the network address.
```

Run the following command, to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
9
```

For Linux OS:

```
linux-t2x0:~ # echo $?
9
```

3.10 Error code 10:

This error code is returned when file transfer to local share fails from local RACADM.

Example:

```
>racadm lclog export -f lclog.xml
 ERROR: Unable to perform requested operation because of an internal error.
```

Run the following command, to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
10
```

For Linux OS:

```
linux-t2x0:~ # echo $?
10
```

3.11 Error code 11:

This error code is returned when iDRAC lockdown mode is enabled or a dependency feature is disabled

Example A:

Enable iDRAC lockdown mode and run any command that changes iDRAC configuration.

```
>racadm set idrac.lockdown.SystemLockdown Enabled
Object Modified successfully.
```

Now set attribute using set command.

```
>racadm set idrAC.webServer.Enable 1
ERROR: RAC1201: Unable to complete the operation because the server is in the
lockdown mode. Disable the lockdown mode by running the following racadm at
the command line interface (CLI): racadm set
iDRAC.Lockdown.SystemLockdownMode 0.
```

Example B:

Disable Lifecycle Controller and run driverpack getinfo command

```
>racadm set lifecycleController.LCAttributes.LifecycleControllerState   Disabled
Object value modified successfully.

>racadm driverpack getinfo
ERROR: RAC1244: Unable to complete the driverpack operation because the
Lifecycle Controller is disabled.Enable the Lifecycle Controller and retry the
operation.To enable, run the following command at the Command Line Interface
(CLI):"racadm set LifecycleController.LCAttributes.LifecycleControllerState 1".
```

Example C:

Generate a new certificate signing request for SSL type when CsrCommonName attribute is not configured.

```
>racadm get idrac.security.CsrCommonName
[Key=idrac.Embedded.1#Security.1]
CsrCommonName=
>racadm >sslcsrgen -g -t 1
Generating CSR. Please wait...
ERROR: The Common Name (CN) field of the CSR Security group must be
configured before a CSR can be generated.
```

For all the above cases, run the following command to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
11
```

For Linux OS:

```
linux-t2x0:~ # echo $?
11
```

3.12 Error code 12:

This error code is returned when a connection to iDRAC is not authorized using remote RACADM.

Example:

Run any remote RACADM command using invalid iDRAC IP/Username/Password.

```
>racadm -r 192.168.1.1 -u root -p <wrong_password> getnicccfg
 ERROR: Login failed - invalid username or password.
```

Run the following command, to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
12
```

For Linux OS:

```
linux-t2x0:~ # echo $?
12
```

3.13 Error code 13:

This error code is returned from local RACADM when there in an internal IPMI stack failure.

Example:

```
>racadm getversion
 ERROR: Unable to perform requested operation because of IPMI failures.
```

Run the following command, to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
13
```

For Linux OS:

```
linux-t2x0:~ # echo $?
13
```

3.14 Error code 14:

This error code is returned when a file transfer from remote RACADM share to iDRAC fails

Example:

Run any remote RACADM file transfer command and disable http file transfer manually.

```
>racadm 192.168.2.2 -u idracusername -p idracpassword update -f firmimgFIT.d9
 ERROR: Unable to transfer file to RAC.
```

Run the following command to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
14
```

For Linux OS:

```
linux-t2x0:~ # echo $?
14
```

3.15 Error code 15:

This error code is returned for any session related errors.

Example:

Run any remote RACADM command when maximum RACADM sessions are in use.

```
>racadm 192.168.2.2 -u idracusername -p idracpassword getversion
ERROR: Unable to create a session.
```

Run the following command, to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
15
```

For Linux OS:

```
linux-t2x0:~ # echo $?
15
```

3.16 Error code 16:

This error code is returned when connection to remote interface failed due to a certificate error.

Example:

Configure the SEKM with the primary Gemalto server. And run the below command to test the network connection.

```
>racadm set idrac.kms.PrimaryServerAddress 100.97.164.20
 Object value modified successfully
>racadm sekm testserverconnection -p -i 1
ERROR: SEKM011: Unable to connect to the Key Management Server because
either invalid server attributes are entered or invalid SEKM certificate is
uploaded to the iDRAC.
```

Run the following command to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
16
```

For Linux OS:

```
linux-t2x0:~ # echo $?
16
```

Here, 16 indicates the failure due to certificate.

3.17 Error code 17:

This error code is returned when an invalid argument is passed.

Example:

Run any valid RACADM command by providing an invalid input.

```
>racadm license export -f FIL_LIC.xml -t 4 -l 162.198.100.131:/abvt
 ERROR: RAC935: Invalid filename specified
 Filename must be alphanumeric.
 Example filename: hwinv1.xml
```

Run the following command, to get the error code:

For Window OS:

```
C:\Users\Administrator>Echo %errorlevel%
17
```

For Linux OS:

```
linux-t2x0:~ # echo $?
17
```

Technical support and resources

Dell.com/support is focused on meeting customer needs with proven services and support.

- The iDRAC with Lifecycle Controller home page on Dell.com/support, provides access to product documents, technical white papers, how-to videos and more. https://www.dell.com/support/article/sln311300