

How I performed automated/unattended cloning of Oracle Application R11i using Rapid Clone in RedHat.

Abstract:

Cloning of Oracle Application R11i using rapid clone needs human intervention to clone to other instance. This case study outlines how I avoided this human intervention using standard Linux 'expect' utility.

Background:

We are running single node Oracle Application R11i production system on RedHat AS3 for our telecom sector business. I have to often clone the production instance (11.5.10.2) to test and development instance. The Apps rapid clone is wonderful tools to clone. Still I need to run the Perl script and answer a series of questions, one by one manually from keyboard, each time I clone, (for each tier and each node). The questions & answers are static/fixed as long as we are using same node to clone. Only an apps-dba who has to clone on regular interval into multiple nodes knows how boring is it. Moreover one single miss-spelled character during cloning process can waste my hours, under pressure.

So I was looking for some automated/unattended solution to clone R11i using Rapid Clone without any human interaction. Basically I was looking for 'something' to whom I can say 'Do database clone' or "Do apps clone" and it will (a.) run clone scripts (b.) answer all question for both db & apps tier correctly (c.) do the clone (e.) report me with a smile 'Cloning is complete'.

I searched Metalink and Google for a solution. I noticed several peoples were also looking for similar solution. But there is no known solution so far! Later I came up with an idea which meets my requirement. This is extremely simple and straight forward approach. I simply used the 'autoexpect' and 'expect' command line utility to do this, which is shipped with standard RedHat Linux distribution.

'autoexpect' and 'expect' are two Linux command line utility. Quoting form Linux man page '...Expect is a program that "talks" to other interactive programs according to a script. Following the script, Expect knows what can be expected from a program and what the correct response should be....'. See Linux man page for detail by 'man expect' and 'man autoexpect'.

Solution:

Here are the details steps I follow to perform unattended Apps R11i cloning for both db & apps tier on Linux:

- a) First I checked if 'expect' is installed in my target node. '\$rpm -qa expect' returns 'expect-5.38.0-92'. Otherwise I had to download and install expect rpm.
- b) I simply performed dbTier clone process within autoexted command '\$autoexpect -f db.exp'. autoexpect saves the whole process into the db.exp file.

- c) Now the db.exp file contains all questions and my answer (specific to my setup, server, display, directory structure etc) to rapid clone. The script was unnecessary large as this saves all output of rapid clone. So edit it in vi editor and removed unnecessary lines like context file, log file location, start time, progress indicator etc.
- d) During next cloning, I will simply execute above script. To execute it slowly, so that human eye can follow, I change the set force_conservative 0 to 1 and 'sleep' inside db.exp file. It gives me the chance to follow the question and answer during automated clone process.
- e) Now my automated dbTier clone wizard is ready. For the next cloning, I simply open a terminal, issue '\$expect db.exp' at shell prompt and seat back. Its done!!
- f) To automate apps tier cloning, I repeated steps 'b' to 'e' for appsTier. I save the autoexpect file as 'apps.exp' and run using '\$expect apps.exp' from shell prompt after necessary modification. This clone the appsTier on this node without any human interventions.
- g) Now, whenever I need to clone without any human intervention, I simply run these two scripts at shell. Its now feels like, 'Let there be clone and it is cloned' ☺

Security/Extensibility/Compatibility:

- a) These scripts store my apps password as plain text. Keep it safe.
- b) If I had multi node installation, probably I might have to prepare separate script for each node (two scripts per node).
- c) As this is now one simple linux command, I am wondering to schedule it in cron or at 'at', at my suitable time (say after the offline copy is complete).
- d) This approach should work for all Oracle Applications running on *NIX as long as 'expect' and 'rapid clone' is there. Simply someone have to prepare his own expect script based on their setup and execute those.
- e) I was reading in internet 'expect' is also available for windows. So this same approach might work for Apps running on Windows.

Conclusion:

This is not a perfect solution for automated/unattended clone, rather a workaround that meets my requirement. The good thing about it is, nobody EVER needs to run these expect scripts on production instance (as you are cloning). That certainly minimizes most of the risk.

DISCLAIMER:

The information in this article is the opinion of the author, not of Oracle Corporation. Any content, materials, information or software downloaded or otherwise obtained through the use of the site is done at your own discretion and risk. Oracle shall have no responsibility for any damage to your computer system or loss of data that results from the download of any content, materials, information or software.

About the Author:

Name: Iqbal H. Khondker

Title: Manager, Enterprise Applications

Company: Banglalink- an ORASCOM TELECOM Company

Email: khondker@gmail.com