Adding a DNS Update Step to a Recovery Plan

VMware vCenter Site Recovery Manager 4.x

VMware vCenter Site Recovery Manager 4.0 includes tools that enable scripted updates of DNS records for recovered virtual machines. Because recovered virtual machines typically acquire new IP addresses, hostnames, or both, DNS records for the recovered virtual machines must be updated before applications running on other virtual machines can contact them. You can update a recovered virtual machine by adding a recovery plan step that runs one of the DNS update tools supplied with SRM. There are two DNS update tools: one that updates a single DNS entry from command-line parameters, and another that updates multiple entries using information that you enter in an expanded CSV file created by the SRM `dr_ip_customizer` utility.

Using the DNS Update Command

You can use `dns_update.cmd` to update a single DNS entry directly from the command line. This command is installed in the SRM `scripts` directory (typically `C:\Program Files\VMware\VMware vCenter Site Recovery Manager\scripts\callouts`). It is intended for use by SRM administrators who are familiar with DNS servers, DNS records, and how hosts on the recovery network use DNS for name resolution.

DNS Update Command Requirements

The DNS update command depends on vendor-supplied commands to update records that DNS servers manage.

- For a Windows DNS server, you need the following vendor-supplied commands:

  - `dnscmd.exe` – An executable file available from the Microsoft Windows 2000/2003 Support Tools package. You must install this tool on the DNS server host in a directory that is included in the system PATH.

  - `psexec.exe` – Part of the Microsoft PSTools package. You must install this command on the SRM server host.

    Read the psexec End User License Agreement (EULA) before you run `dns_update.cmd`. To do this, run `psexec.exe` with no parameters. The `dns_update.cmd` command uses the `/accepteula` option when it calls `psexec.exe`. This option explicitly accepts the EULA. If you cannot or do not want to accept the psexec license agreement, do not run `dns_update.cmd`.

- For a Linux DNS server, you need `nsupdate.exe`, a tool from the Windows binary distribution of BIND. For more information, see https://www.isc.org/software/bind/. You must install this command on the SRM server host.

DNS Update Command Options and Arguments

Table 1 lists the command-line options and arguments that `dns_update.cmd` accepts. Some options are optional. Some arguments must be delimited by double quotation marks.

Table 1. dns-update command-line options and arguments

Option	Description
−srv *addr*	*addr* specifies the IP address of the DNS server to update.
−u "*name*"	(Optional) *name* specifies the name of a Windows user account to use when updating the Windows DNS server. Requires specification of −p option.
−p "*password*"	(Optional) Password of Windows account specified with the −u option. If this option is used, the password is visible to anyone with permission to test, run, or modify a recovery plan.
−srvtype *type*	(Optional) DNS server type. *type* must be one of the following choices: ■ bind – Specifies a BIND server running on Linux ■ windns – Specifies a DNS server running on Windows If −srvtype is not specified, the Windows type is tried, then the Linux type.
−cmd *command*	*command* must be one of the following DNS commands: ■ add – Adds or modifies DNS records for the given hostname or IP addresses. New records are added to the forward lookup zone and, if −rzone is specified, to the reverse lookup zone. If duplicate records exist in either zone, they are overwritten. ■ del – Deletes existing DNS records for the given hostname and IP addresses. All matching records in the specified forward zone are deleted. If a reverse lookup zone (−rzone) is specified, all matching IP addresses in that zone are deleted. Use this option to clean up DNS records before running the tool in add mode.
−fzone *domain*	Specifies the forward lookup zone *domain* name.
−rzone *addr*	(Optional) Specifies the IP address of the reverse lookup zone. *addr* must be specified in normal (not reversed) IP address order. The 192.168.0.*x* network is represented as 192.168.0, not 0.168.192. The IP addresses that the −ip option supplies are appended to the given reverse zone. For example, if you specify −rzone 192.168.0 −ip "50", the IP address 192.168.0.50 is created for the specified hostname. If you omit the −rzone option, the −ip option must specify one or more complete IP addresses.
−hostname *name*	Use *name* as the hostname part of the updated DNS record.
−ip "*addr [, addr] ...*"	*addr* specifies an IP address to use for the updated DNS record. Use commas to separate multiple IP addresses. If a reverse lookup zone (−rzone) is specified, *addr* can consist of one, two, or three digits. The digits are appended to the address specified in −rzone to form complete IP addresses. For example, if you specify −rzone 192.168.0 −ip "1,21,121", the IP addresses 192.168.0.1, 192.168.0.21, and 192.168.0.121 are created. If you omit −rzone, all IP addresses contained in *addr* must be complete. If you include −rzone and also specify complete IP addresses in *addr*, the specified addresses are used.
−fkey "*key*"	(Optional) Use only if −srv specifies a BIND server that uses transaction signatures security. *key* specifies the shared secret of the forward lookup zone for BIND in *keyname:secret* format. For example: −fkey "example.com.key:zHMF4YNS5D1LWnDXnUk7oQ=="
−rkey "*key*"	(Optional) Use only if −srv specifies a BIND server that uses transaction signatures security. *key* specifies the shared secret of the forward lookup zone for BIND in *keyname:secret* format. For example: −rkey "example.com.key:zHMF4YNS5D1LWnDXnUk7oQ=="
−toolspath "*path*"	(Optional) Comma-separated list of directories to append to the PATH environment variable while the script runs. Required if the vendor-supplied tools listed in "DNS Update Command Requirements" on page 1 are not in the system PATH.
−ttl *seconds*	(Optional) A time-to-live value, in seconds, for the new or changed DNS record. The default value is 86400 (1 day).
−skipmode *mode*	(Optional) Do not execute for a recovery plan run in the specified mode (one of test or recovery). If this option is not specified, the command runs in both recovery modes.

DNS Update Command Examples

Example 1 illustrates a command line that updates the DNS record for a host with a single IP address. The command does not run when a test recovery is performed. Because no user name is specified, the command updates the Windows DNS server host at 192.168.0.1 using the Windows local system account. It adds a new record, pointing to the IP address 192.168.0.50, for the host named `mailer` in the `example.com` zone. It also adds a new record, pointing to the IP address 192.168.0.50, for the host named `mailer` in the `0.168.192.in-addr.arpa` reverse zone. A `-toolspath` is included to specify the locations of the `dnscmd.exe` and `psexec.exe` files.

Example 1. Update a DNS Record for a Host With a Single IP Address

```
dns_update.cmd -srv 192.168.0.1 -srvtype windns -cmd add -fzone example.com -rzone 192.168.0
              -hostname mailer -ip "50" -toolspath "C:\Program Files\PsTools;C:\Program
              Files\Windows 2000 Support Tools" -skipmode test
```

Example 2 modifies the command in Example 1 to add multiple IP addresses for the host named `mailer`, with forward and reverse lookup information.

Example 2. Update a DNS Record for a Host With Multiple IP Addresses

```
dns_update.cmd -srv 192.168.0.1 -srvtype windns -cmd add -fzone example.com -rzone 192.168.0
              -hostname mailer -ip "50,51,52" -toolspath "C:\Program Files\PsTools;C:\Program
              Files\Windows 2000 Support Tools" -skipmode test
```

The three address fragments supplied in the argument to the `-ip` option are combined with the address fragment supplied in the `-rzone` option to form new IP addresses in the forward and reverse lookup zones. The new IP addresses are 192.168.0.50, 192.168.0.51, and 192.168.0.52.

Creating a Log File

`dns_update.cmd` stops running the first time it encounters a nonzero status return from any operation. The the output of the failed command is printed to standard output and `dns_update.cmd` exits. You can redirect the diagnostic output of `dns_update.cmd` to a log file and review it to obtain more information about the details of the failure.

The following items are common reasons for `dns-update.comd` to stop running:

- Incorrect username or password
- Inability to connect to the specified DNS server because it is not a Windows DNS or BIND server
- Nonexistent forward or reverse zone
- Invalid hostname or IP address
- Incorrect key
- Inability to add a new record or to delete an existing record

After you review the log file, correct any failure conditions and clean up as needed. `dns_update.cmd` does not perform any automated clean up after a failure.

Using the DNS Update Script with dr_ip_customizer Output

SRM includes the `dns_updates.pl` PERL script that automates the execution of `dns_update.cmd` for multiple hosts. The script reads an expanded version of the CSV file that the `dr_ip_customizer` command generates. You use the expanded CSV file and the `dns_updates.pl` DNS update script in your recovery plan to manage IP customization and DNS updates for recovered virtual machines.

`dns_updates.pl` is installed in `C:\Program Files\VMware\VMware vCenter Site Recovery Manager\scripts\callouts`. It is intended for use by SRM administrators who are familiar with DNS servers, DNS records, and how hosts on the recovery network use DNS for name resolution. For more information about `dr_ip_customizer`, see the SRM online Help.

DNS Update Script Requirements

`dns_updates.pl` requires that PERL 5.8.8 or later be installed on the SRM server host in a location that is in the system PATH.

Because `dns_updates.pl` calls `dns_update.cmd`, your computing environment must meet all the prerequisites for running `dns_update.cmd`. For more information, see "DNS Update Command Requirements" on page 1.

The SRM `dr_ip_customizer` utility generates a special form of the CSV file. The `dns_updates.pl` script requires this file. To generate this CSV file, run `dr_ip_customizer` with the `-extradnscols` option, which creates a CSV file that includes additional columns. You can edit the CSV file to populate the new columns with values that `dns_updates.pl` passes as command-line arguments to `dns_update.cmd`. Table 2 lists the column names and describes how `dns_update.cmd` uses the data in each column. For more information, see "DNS Update Command Options and Arguments" on page 2.

Table 2. Column Names That dr_ip_customizer Creates

Column Name	Description
Hostname	String to use for the *name* argument of `-hostname` *name*
TTL	String to use for the *seconds* argument of `-ttl` *seconds*
Forward Zone	String to use for the *domain* argument of `-fzone` *domain*
Reverse Zone	String to use for the *addr* argument of `-rzone` *addr*
DNS Server Type	String to use for the *type* argument of `-srvtype` *type*
DNS Server Username	String to use for the *name* argument of `-u` *name*
DNS Server Password	String to use for the *password* argument of `-p` *password*
Forward Zone Key	String to use for the *key* argument of `-fkey` *key*
Reverse Zone Key	String to use for the *key* argument of `-rkey` *key*

Adding Extra DNS Columns to an Existing CSV File

You can extend a CSV file that you created with `dr_ip_customizer` to use with `dns_updates.pl`. You must add extra columns with the correct header strings to the file. You can add this information in the following ways:

- Open the CSV file and append the following text to the first line of the file:

  ```
  ,Hostname,TTL,Forward Zone,Reverse Zone,DNS Server Type,DNS Server Username,DNS Server
              Password,Forward Zone Key,Reverse Zone Key
  ```

  The text must match exactly or `dns_updates.pl` does not generate valid `dns_update.cmd` options. You must also append nine commas to each of the remaining lines in the file.

- Run `dr_ip_customizer` to create a new CSV file. Copy the extra columns from the new file and append them to the CSV file to extend.

DNS Update Script Options and Arguments

Table 3 describes the command-line options and arguments that `dns_updates.pl` accepts. Some options are optional.

Table 3. DNS Update Script Options and Arguments

Option	Description
`––csv` *csv-file*	Pathname to a CSV file generated by using `dr_ip_config` with the `–extradnscols` option
`––toolspath` "*path*"	(Optional) Comma-separated list of directories to append to the PATH environment variable while the script runs. Required if the vendor-supplied tools listed in "DNS Update Command Requirements" on page 1 are not in the default PATH.
`––skipmode` *mode*	(Optional) Do not execute for a recovery plan run in the specified mode (one of test or recovery). If this option is not specified, the command runs in both recovery modes.
`––verify`	(Optional) Reads the CSV file and prints the generated `dns_update.cmd` command lines to the standard output. Does not run any commands.
`––level` *verbosity*	(Optional) Specifies one of the following types of log message *verbosity*: QUIET ERROR WARNING VERBOSE TRIVIA
`––help`	(Optional) Prints a Help message.

DNS Update Script Examples

You can use a spreadsheet program to open the CSV file that `dr–ip–customizer –extradnscols` creates and add rows that provide values for `dns_update.cmd` option arguments, as Table 4 shows. To provide room for the extra columns in this example, some of the default columns are not displayed.

Table 4. CSV File With Extra DNS Columns

VM ID	VM Name	Adapter ID		IP Address		DNS Server(s)	DNS Suffix(es)	Hostname	TTL	Forward Zone	Reverse Zone	DNS Server Type	DNS Server Username	DNS Server Password	Forward Zone Key	Reverse Zone Key
shdw1	srm1	0														
shdw1		1		192.168.0.40		192.168.0.100		dbhost		example.com	192.168.0	windns	admin	passwd		
shdw2	srm2	0														
shdw2		1														
shdw2		1														
shdw3	srm3	0														
shdw2		1														
shdw3		1														

Example 3 shows a command line that uses the CSV file from Table 4 as input to run `dns_updates.pl`. Because the `––verify` option is used, `dns_updates.pl` prints the generated `dns_update.cmd` lines to standard output, but does not run them. The value used for the `–ip` option in the `dns_update.cmd` line is derived from the rightmost component of the IP address column of the CSV file.

Example 3. Running dns_updates.pl With CSV Input

```
dns_updates.pl --csv c:\example.csv --level TRIVIA --verify --toolspath c:\bin
--skipmode test

[2009-04-17 10:39:50 INFO] *** dns_updates.pl ver.1.0.0.7 ***
[2009-04-17 10:39:50 VERBOSE] Scripts Log level set to TRIVIA
[2009-04-17 10:39:50 TRIVIA] csvFile = UseCase1.csv
[2009-04-17 10:39:50 TRIVIA] verification mode enabled
[2009-04-17 10:39:50 INFO] Verification is successful: dns_update.cmd -hostname hostname1 -srv
"192.168.0.100" -srvtype windns -u "admin" -p "passwd" -fzone "example.com" -rzone "192.168.0"
                -ip "40" -toolspath "c:\bin" -skipmode "test"
```

The script runs DNS update commands that update DNS information for all virtual machines in a recovery plan. If your recovery plan includes virtual machines in more than one priority group, create a CSV file for each priority group.

Adding a DNS Update Step to a Recovery Plan

To add a step that runs either dns_updates.pl or dns_update.cmd to a recovery plan, follow the instructions in the SRM online Help for adding commands to a recovery plan. If all virtual machines in the recovery plan are members of the normal priority group, add the new command step immediately preceding the Recover Normal Priority Virtual Machines step. If you moved recovered virtual machines to other priority groups, add a similar step immediately preceding the step in which each of the other priority groups is recovered.