

**CHEF™**

Resources Quick Reference for Windows

In a Chef recipe, a resource is something that exists on a system along with its state. A service to be started, a package to be installed, or a user to be created are all examples of resources.

Resource properties are shown here for illustration only. Read more at <https://docs.chef.io/resources.html>.

windows_package

Use to install, remove, and upgrade Windows MSI packages.

```
windows_package '7zip' do
  source 'C:\7z920.msi'
  action :install
end
```

```
windows_package '7zip' do
  source 'http://www.7-zip.org/a/7z938-x64.msi'
  checksum 'b57cbc304cbfelcfce08...1a7e'
end
```

```
windows_package '7zip' do
  source 'http://www.7-zip.org/a/7z938-x64.msi'
  remote_file_attributes ({
    :path => 'C:\7zip.msi',
    :checksum => 'b57cbc304cbfelcfce08...1a7e'
  })
end
```

template

Use to manage files from the '/templates' directory in a cookbook. Erubis templating processes the file contents after transferring it to a node.

```
template 'c:\inetpub\wwwroot\default.html' do
  group 'Administrators'
  owner 'Administrator'
  source 'index.html.erb'
  variables({ :key => value })
  action :create
end
```

service

Use to manage services.

```
service 'W3SVC' do
  state :manual
  action :delayed_start
end
```

powershell_script

Use to execute a script with the Windows PowerShell interpreter.

```
powershell_script 'cwd-to-win-env-var' do
  cwd '%TEMP%'
  environment {"ENV_VARIABLE" => "VALUE"}
  code <<-EOH
  write-host "hello world"~
  EOH
end
```

user

Use to manage users and passwords.

```
user 'alice' do
  comment 'Alice is a user.'
  action :create
end
```

group

Use to manage a local group.

```
group 'staff' do
  members ['alice', 'bob']
  action :create
end
```

file

Use to manage files directly on a node.

```
file 'c:\windows\example.txt' do
  group 'Administrators'
  owner 'Administrator'
  content 'A string to write.'
  action :create
end
```

cookbook_file

Use to transfer files from the '/files' directory in a cookbook.

```
cookbook_file 'c:\inetpub\wwwroot\bootstrap.css' do
  group 'Administrators'
  owner 'Administrator'
  source 'bootstrap.css'
  action :create
end
```

directory

Use to manage a directory.

```
directory 'C:\data\alice' do
  rights :full_control, 'DOMAIN\User'
  inherits false
  action :create
end
```

log

Use to create log entries. Set *log_location* in the *client.rb* file to *Chef::Log::WinEvt.new* to use the Windows Event Logger.

```
log 'A message to add to the log file.' do
  level :info
  action :write # set log_location in client.rb
end
```

dsc_script

Use to run Desired State Configuration (DSC) resources in Chef.

```
dsc_script 'emacs' do
  code <<-EOH
  Environment 'texteditor'
  {
    Name = 'EDITOR'
    Value = 'c:\emacs\bin\emacs.exe'
  }
EOH
end
```

dsc_resource (requires PowerShell 5.0)

Use to run a Desired State Configuration (DSC) resource in Chef.

```
dsc_resource 'example' do
  resource :archive
  property :ensure, 'Present'
  property :path, 'C:\Users\example.zip'
  property :destination, 'C:\Users\ExtractionPath'
end
```

which corresponds to the DSC resource:

```
Archive ExampleArchive {
  Ensure = "Present"
  Path = "C:\Users\Public\example.zip"
  Destination = "C:\Users\ExtractionPath"
}
```

registry_key

Use to create and delete registry keys in Microsoft Windows.

```
registry_key 'HKEY_LOCAL_MACHINE\...\System' do
  values [{
    :name => 'NewRegistryKeyValue',
    :type => :multi_string,
    :data => ['foo\0bar\0\0']
  }]
  action :create
end
```

batch

Use to execute a batch script using the cmd.exe interpreter..

```
batch 'echo some env vars' do
  environment {"ENV_VARIABLE" => "VALUE"}
  code <<-EOH
  echo %TEMP%
  echo %SYSTEMDRIVE%
  echo %PATH%
  echo %WINDIR%
EOH
end
```

reboot

Use to reboot a node.

```
reboot 'Reboot, then continue Chef client run' do
  delay_mins 2
  action :reboot_now
end
```

```
reboot 'Reboot after Chef client run' do
  delay_mins 2
  action :request_reboot
end
```

Common properties/actions (use w/any resource)

Specify an interpreter:

```
guard_interpreter :powershell_script # or :batch
```

Continue chef-client run even if resource fails:

```
ignore_failure true
```

Notify another resource when this one changes:

```
notifies :action, 'resource[name]'
```

Listen to another resource for changes:

```
subscribes :action, 'resource[name]'
```

Retry the resource:

```
retries 0 # number of times to retry
retry_delay 2 # retry delay, in seconds
```

Wait to be notified by another resource:

```
action :nothing # wait for notification
```

Prevent execution when true:

```
not_if { 'grep bob /etc/motd', :user => 'Alice' }
```

Allow execution when true:

```
only_if { File.exist?('stuff.zip') }
```

Windows file security

The template, file, remote_file, cookbook_file, directory, and remote_directory resources support inheritance and access control lists (ACLs) within recipes.

```
file 'example.txt' do
  rights :read, 'Everyone'
  rights :write, 'domain\group'
  rights :full_control, 'bob'
end
```

or:

```
deny_rights :full_control, ['alice']
```

By default, a file or directory inherits rights from its parent.

```
directory 'C:\company\department' do
  rights :full_control, 'staff\bob'
  inherits false # Only Bob can have access
end
```

Recipe DSL methods for Windows

Find data in a registry key:

```
registry_data_exists?(
  KEY_PATH,
  { :name => 'NAME', :type => TYPE, :data => VALUE },
  ARCHITECTURE
)
```

See all of the Recipe DSL methods for the Windows platform:
https://docs.chef.io/dsl_recipe.html#windows-platform.

Read more:

- * <https://docs.chef.io/resources.html>
- * <https://docs.chef.io/windows.html>



CHEF™