

# Puppet on Windows

Systems Administration

Otago Polytechnic  
Dunedin, New Zealand

# YOU CAN USE PUPPET TO MANAGE WINDOWS

- ▶ The puppetmaster must run Linux.
- ▶ The puppet agents can be most modern versions of Windows.
- ▶ The puppet agent version can be lower than the master (e.g., a 2.7 agent can talk to a 3.0 master), but not the reverse.
- ▶ Some Puppet resources can be specified in a cross platform way, but
  - ▶ some need to be specified in a platform specific way,
  - ▶ some are platform specific.

# WRITING MANIFESTS FOR WINDOWS

Puppet can manage the following resource types on Windows

- ▶ file
- ▶ user
- ▶ group
- ▶ package
- ▶ service
- ▶ exec
- ▶ host
- ▶ scheduled\_task

# INSTALLING THE WINDOWS AGENT

- ▶ Download the puppet-3.8.5-x64 msi from <http://downloads.puppetlabs.com/windows>.
- ▶ The msi includes all of the dependencies.
- ▶ Edit the hosts file before you install. The agent needs to be able to resolve the hostname of the puppetmaster.

# WINDOWS FILE PATHS

Forward slash or back slash?

Use forward slashes for

- ▶ file resource titles and paths
- ▶ package resource paths
- ▶ the command attribute of an exec resource (with exceptions)

You *must* use back slashes for

- ▶ the command of a scheduled\_task
- ▶ the install\_options of a package resource

# INSTALLING PACKAGES

Puppet can install MSI and executable packages, but the file must be available on the agent system. You can either

- ▶ Manage the installer as a file resource, or
- ▶ Place the installer on a mapped network drive, or
- ▶ Specify a UNC path.

# INSTALLING PACKAGES

## Examples

```
package { 'mysql':  
  ensure      => '5.5.16',  
  source      => 'N:/packages/mysql-5.5.16-winx64.msi',  
  install_options => ['INSTALLDIR=C:\mysql-5.5'],  
}  
  
package { "Git version 1.8.4-preview20130916":  
  ensure      => installed,  
  source      => 'C:\\code\\puppetlabs\\temp\\windowsexample\\  
  install_options => ['/VERYSILENT']  
}
```

# USERS AND GROUPS

Puppet's native user and group resources can only directly manage local users and groups. To manage domain users you will need to use a Powershell script. There are also some 3rd party Puppet modules that can help with this.



# LINE ENDINGS

Remember that Unix/Linux and Windows systems denote line endings differently. If you create a file resource on a \*nix systems that will be used on Windows hosts, you may need to modify the line endings.