

id: fsconfigref title: FileServer Configuration File Reference allowDiscussion: None subject: description: A reference for the Puppet fileserver configuration file. contributors: creators: luke effectiveDate: 2005/08/29 18:38:39.469 GMT-5 expirationDate: None language: en rights: creation_date: 2005/08/29 18:08:28.942 GMT-5 modification_date: 2005/08/29 18:38:39.471 GMT-5 layout: document_view Content-Type: text/x-rst

FileServer

Puppet comes with both a client and server for copying files around. The file serving function is provided as part of the central Puppet daemon, `puppetmasterd`, and the client function is used through the `source` attribute of `file` objects:

```
# copy a remove file to /etc/sudoers
file { "/etc/sudoers":
  mode => 440,
  owner => root,
  group => root,
  source => "puppet://server/module/sudoers"
}
```

As the example implies, Puppet's fileserving function abstracts local filesystem topology by supporting fileservice "modules". You specify a path to serve and a name for the path, and clients request it by name instead of by path. This provides the ability to conceal from the client unnecessary details like the local filesystem configuration.

File Format

The default location for the file service is `/etc/puppet/fileserver.conf`; this can be changed using the `--fsconfig` flag to `puppetmasterd`. The format of the file is almost exactly like that of `rsync`, although it does not yet support nearly the functionality of `rsync`. The configuration file resembles INI-style files, but it is not exactly the same:

```
[module]
  path /path/to/files
  allow *.domain.com
  deny *.wireless.domain.com
```

These three options represent the only options currently available in the configuration file. The module name somewhat obviously goes in the brackets. While the path is the only required option, the default security configuration is to deny all access, so if no `allow` lines are specified, the module will be configured but available to no one.

Security

There are two aspects to securing the Puppet file server: Allowing specific access, and denying specific access. By default no access is allowed. There are three ways to specify a class of clients who are allowed or denied access: By IP address, by name, or a global allow using `*`.

Priority

All `deny` statements are parsed before all `allow` statements, so if any `deny` statements match a host, then that host will be denied, and if no `allow` statements match a host, it will be denied.

Host Names

Host names can be specified using either a complete hostname, or specifying an entire domain using the * wildcard:

```
[export]
  path /export
  allow host.domain1.com
  allow *.domain2.com
  deny badhost.domain2.com
```

IP Addresses

IP address can be specified similarly to host names, using either complete IP addresses or wildcarded addresses, but you can also use CIDR-style notation:

```
[export]
  path /export
  allow 127.0.0.1
  allow 192.168.0.*
  allow 192.168.1.0/24
```

Global allow

Specifying a single wildcard will let anyone into a module:

```
[export]
  path /export
  allow *
```