
hostlists Documentation

Release 0.7.85

Dwight Hubbard

March 08, 2015

| | | |
|----------|---|----------|
| 1 | Documentation for the hostlists module | 3 |
| 1.1 | Hostlists Plugin Modules | 3 |
| 2 | Indices and tables | 5 |
| | Python Module Index | 7 |

Contents:

Documentation for the hostlists module

A plugin extendable hostlists infrastructure

This module provides functions for getting a list of hosts from various systems as well as compressing the list into a simplified list.

This module uses the `hostlists_plugins` python scripts to actually obtain the listings.

`hostlists.hostlists.cmp_compat(a, b)`

Simple comparison function :param a: :param b: :return:

`hostlists.hostlists.compress(hostnames)`

Compress a list of host into a more compact range representation

`hostlists.hostlists.compress_domain(hostnames)`

Compress a list of hosts in a domain into a more compact representation

`hostlists.hostlists.expand(range_list, onepass=False)`

Expand a list of lists and set operators into a final host lists >>> `hostlists.expand(['foo[01-10]', '-', 'foo[04-06]'])`
`['foo09', 'foo08', 'foo07', 'foo02', 'foo01', 'foo03', 'foo10'] >>>`

`hostlists.hostlists.expand_item(range_list, onepass=False)`

Expand a list of plugin:parameters into a list of hosts

`hostlists.hostlists.get_plugins()`

Wrap the `get_plugins()` function so it can be used by plugins

`hostlists.hostlists.get_setting(key)`

Get setting values from `CONF_FILE` :param key: :return:

`hostlists.hostlists.range_split(hosts)`

Split up a range string, this needs to separate comma separated items unless they are within square brackets and split out set operations as separate items.

1.1 Hostlists Plugin Modules

Hostlists supports expansion using plugins. Programs by default are run through each other recursively.

1.1.1 Host Range Plugin

hostlists plugin to get hosts from a file

1.1.2 File Plugin

hostlists plugin to get hosts from a file

1.1.3 DNS Plugin

1.1.4 DNS IP Plugin

1.1.5 HAProxy Plugin

haproxy host plugin NOTE: This script is primarily intended as a proof of concept of adding a load balancer as the host IP address source.

THEORY OF OPERATION: Since haproxy does not provide an externally accessible means of accessing the backend names and states for hosts it is proxying this module uses ssh to call a helper script on the haproxy server to get the information.

USAGE: This plugin requires a significant amount of pre-configuration on the haproxy server in order to work. This requires a configuration change to enable the stats socket in haproxy, setting permissions, and copying the get_haproxy_phys script into place.

The setup steps are:

1. Enable the haproxy status socket, the script expects it to be /tmp/haproxy by adding:
stats socket /tmp/haproxy
to the global section of the /etc/haproxy/haproxy.cfg file
2. Restart haproxy to create the socket
3. Make sure the socket file is owned by the user that will be connecting to the haproxy server.
4. Copy the get_haproxy_phys from the sshmap hostlists_plugins directory into the root of the home directory of the user that will be connecting to the haproxy server.
5. Set up the user's ~/.ssh/authorized_keys file to allow access via ssh without password.

Indices and tables

- *genindex*
- *modindex*
- *search*

h

- `hostlists.hostlists`, 3
- `hostlists.plugins.dns`, 4
- `hostlists.plugins.dnsip`, 4
- `hostlists.plugins.file`, 4
- `hostlists.plugins.haproxy`, 4
- `hostlists.plugins.range`, 3

C

`cmp_compat()` (in module `hostlists.hostlists`), 3
`compress()` (in module `hostlists.hostlists`), 3
`compress_domain()` (in module `hostlists.hostlists`), 3

E

`expand()` (in module `hostlists.hostlists`), 3
`expand_item()` (in module `hostlists.hostlists`), 3

G

`get_plugins()` (in module `hostlists.hostlists`), 3
`get_setting()` (in module `hostlists.hostlists`), 3

H

`hostlists.hostlists` (module), 3
`hostlists.plugins.dns` (module), 4
`hostlists.plugins.dnsip` (module), 4
`hostlists.plugins.file` (module), 4
`hostlists.plugins.haproxy` (module), 4
`hostlists.plugins.range` (module), 3

R

`range_split()` (in module `hostlists.hostlists`), 3