
server-admintools

Release 0.1.1

Jun 06, 2018

CONTENTS

1	server_admintools package	3
1.1	Submodules	3
1.2	server_admintools.gitlab module	3
1.3	server_admintools.rsync module	3
1.4	server_admintools.sudo module	4
1.5	Module contents	4
2	server_admintools	5
3	Indices and tables	7
	Python Module Index	9
	Index	11

Contents:

SERVER_ADMINTOOLS PACKAGE

1.1 Submodules

1.2 server_admintools.gitlab module

`server_admintools.gitlab.backup_repos()`
Clones all repositories from a gitlab instance for a user and tars the directories together :return: None

`server_admintools.gitlab.create_backup()`
Performs the Gitlab Backup. Requires sudo permissions. After creating and renaming the backup, changes the ownership to the specified user :return: None

`server_admintools.gitlab.create_config_tarball(destination_file)`
Creates a tarball containing the gitlab instance's secrets and config :return: None

`server_admintools.gitlab.execute_gitlab_rake_backup()`
Executes the gitlab-rake command that creates a backup file in the location specified in the gitlab.rb config file :return: None

`server_admintools.gitlab.fetch_gitlab_clone_script()`
Downloads the latest version of the gitlab-cloner script :return: None

`server_admintools.gitlab.parse_backup_args()`
Parses the arguments for the gitlab-backup script :return: The arguments provided via the CLI

`server_admintools.gitlab.parse_clone_repo_args()`
Parses the arguments for the git repository cloner :return: The parsed CLI arguments

1.3 server_admintools.rsync module

`server_admintools.rsync.execute_ssh_rsync_backup(user, source_server, source_path, destination, port=22)`
Executes an rsync backup over SSH :param user: The user that logs on using SSH :param source_server: The source server name/IP address :param source_path: The path on the source machine :param destination: The path on the destination machine :param port: The port over which to connect, defaults to 22 :return: None

`server_admintools.rsync.parse_ssh_rsync_backup_args()`
Parses the CLI for the information required for an Rsync backup :return: None

`server_admintools.rsync.run_ssh_rsync_backup()`
Runs the Rsync backup using the CLI data :return: None

1.4 server_admintools.sudo module

`server_admintools.sudo.change_ownership` (*path, user*)

Changes the ownership of a file or directory :param path: The path of the file or directory
whose owner should be changed

Parameters `user` – The new owner of the file or directory

Returns None

`server_admintools.sudo.is_sudo` ()

Checks if the current user has root privileges :return: True if the user has the privileges, False otherwise

`server_admintools.sudo.quit_if_not_sudo` ()

Quits the program in case the user does not have root privileges :return: None

1.5 Module contents

SERVER_ADMINTOOLS

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

S

server_admintools, 4
server_admintools.gitlab, 3
server_admintools.rsync, 3
server_admintools.sudo, 4

INDEX

- B**
backup_repos() (in module server_admintools.gitlab), 3
server_admintools.rsync (module), 3
server_admintools.sudo (module), 4
- C**
change_ownership() (in module server_admintools.sudo),
4
create_backup() (in module server_admintools.gitlab), 3
create_config_tarball() (in module
server_admintools.gitlab), 3
- E**
execute_gitlab_rake_backup() (in module
server_admintools.gitlab), 3
execute_ssh_rsync_backup() (in module
server_admintools.rsync), 3
- F**
fetch_gitlab_clone_script() (in module
server_admintools.gitlab), 3
- I**
is_sudo() (in module server_admintools.sudo), 4
- P**
parse_backup_args() (in module
server_admintools.gitlab), 3
parse_clone_repo_args() (in module
server_admintools.gitlab), 3
parse_ssh_rsync_backup_args() (in module
server_admintools.rsync), 3
- Q**
quit_if_not_sudo() (in module server_admintools.sudo), 4
- R**
run_ssh_rsync_backup() (in module
server_admintools.rsync), 3
- S**
server_admintools (module), 4
server_admintools.gitlab (module), 3