
zget Documentation

Release 0.11.1

Nils Werner

Jul 27, 2017

Contents

1	Command Line Interface	3
1.1	Upload	3
1.2	Download	3
2	Python Interface	5
2.1	zget package	5
3	License	7
4	Indices and tables	9
	Python Module Index	11

Zget is a simple, Zeroconf-based, peer to peer file transfer utility, for situations where you and your peer are sitting next to each other and want to transfer a file quickly (and can shout the filename across the room).

Files and peers are recognized by the filename they want to transfer, not by their hostnames or IPs.

zget uses the fact that the filename is known to both parties as a basic authentication feature: The sender only advertises the sha1 hash of the filename on the network. Since the receiver must also know the filename, it can look for the sha1 using zeroconf.

If a match was found, a simple HTTP request is made: For transferring the file the actual filename is then requested from the sender. If this filename wasn't correct, the process is aborted.

The only third party dependency is [zeroconf](#), a platform independent pure-Python Zeroconf implementation.

```
$ zput file.zip
```

on the sender and

```
$ zget file.zip
```

on the receiver.

Done.

Contents:

CHAPTER 1

Command Line Interface

After installing, this toolkit provides a two command line utilities

```
$ zget
```

and

```
$ zput
```

Upload

`zput` knows a few parameters that allow you to set the behaviour to your needs:

- `-a ADDRESS` lets you manually set the IP address to broadcast and bind to (e.g. `192.168.0.1`)
- `-i INTERFACE` lets you manually select the network card to broadcast on (e.g. `eth0`)
- `-p PORT` lets you manually select the port to open. (e.g. `8080`)
- `-v` (one or more) lets you increase verbosity, to debug failing connections.
- `-q` lets you hide the progress bar, for batch processing.
- `-t SECONDS` lets you set a timeout after which the transfer is aborted and a non-zero exit value is returned
- `-V` Show version number and exit.
- `-h` Show help and exit.

Download

`zget` knows a few parameters that allow you to set the behaviour to your needs:

- `-v` (one or more) lets you increase verbosity, to debug failing connections.

- `-q` lets you hide the progress bar, for batch processing.
- `-t SECONDS` lets you set a timeout after which the transfer is aborted and a non-zero exit value is returned
- `-V` Show version number and exit.
- `-h` Show help and exit.

Note: Each unit provides a help-switch `-h` to investigate its parameters.

zget package

Module contents

<code>zget.get(filename[, output, reporthook, timeout])</code>	Receive and save a file using the zget protocol.
<code>zget.put(filename[, output, interface, ...])</code>	Send a file using the zget protocol.
<code>zget.TimeoutException(msg)</code>	Exception raised when a timeout was hit.

Module reference

`zget.get(filename, output=None, reporthook=None, timeout=None)`

Receive and save a file using the zget protocol.

Parameters **filename** : string

The filename to be transferred

output : string

The filename to save to. Optional.

reporthook : callable

A hook that will be called during transfer. Handy for watching the transfer. See `urllib.urlretrieve` for callback parameters. Optional.

timeout : int

Seconds to wait until process is aborted. A running transfer is not aborted even when timeout was hit. Optional.

Raises **TimeoutException**

When a timeout occurred.

`zget.put` (*filename*, *output=None*, *interface=None*, *address=None*, *port=None*, *reporhook=None*, *time-out=None*)

Send a file using the zget protocol.

Parameters **filename** : string

The filename to be transferred

output : string

The alias to share on the network. Optional. If empty, the input filename will be used.

interface : string

The network interface to use. Optional.

address : string

The network address to use. Optional.

port : int

The network port to use. Optional.

reporhook : callable

A hook that will be called during transfer. Handy for watching the transfer. See `urllib.urlretrieve` for callback parameters. Optional.

timeout : int

Seconds to wait until process is aborted. A running transfer is not aborted even when timeout was hit. Optional.

Raises **TimeoutException**

When a timeout occurred.

exception `zget.TimeoutException` (*msg=u'Timeout.'*)

Bases: `exceptions.Exception`

Exception raised when a timeout was hit.

The code in this package has been licensed using the MIT/BSD license:

Copyright (c) 2015, Nils Werner All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

CHAPTER 4

Indices and tables

- `genindex`
- `modindex`
- `search`

Z

zget, [5](#)

G

`get()` (in module `zget`), [5](#)

P

`put()` (in module `zget`), [5](#)

T

`TimeoutException`, [6](#)

Z

`zget` (module), [5](#)