mirrorchecker Documentation

Release 0.2

Nadav Goldin

November 02, 2016

Contents

CHAPTER 1

Introduction

mirrorchecker is a small tool that checks if mirror sites are synchronized with a source site. It uses `python3.5` and `asyncio`. It was inspired by mirmon, but unlike mirmon, runs as a daemon in the background, optionally in a container, and adds some more features.

1.1 Notable Features

1. Check mirror sites freshness by comparing timestamps automatically.
2. Retrieve each mirror site status individually using a simple GET request.
3. Retrieve all mirror sites status using a GET request.
4. Serve the `mirrorlist` file, filtered by mirror synchronization status suited for use with RPM repositories(with `yum` or `dnf`).
5. Allows to easily build monitoring alerts(Icinga/Nagios plugin provided).

1.2 How does it work?

mirrorchecker assumes you have SSH access to the *source site* (the site which mirror sites connect to, in order to pull updates). *mirrorchecker* does 4 things:

1. Send periodically timestamp files to pre-specified locations in the *source site* over SFTP.
2. Pull periodically the timestamps from the mirror sites URLs.
3. Exposes a small async http server, which can be queried to check the status of the mirroring sites, see nagios_alert as an example.
4. Optionally, the web server can also serve the `mirrorlist` URL in yum based repositories. This allow to filter mirror sites by the time they were last syncronized.

At its base, it uses the simple concept that if mirroring is working properly, the timestamps ‘planted’ in the *source site* would reach to the mirror sites.

1.3 How does mirrorchecker determines if a mirror site is synchronized?

By default, all timestamps are fetched from the mirror sites URLs and compared to the local time, the maximal difference is set as the last synchronization time. This is, of course, not accurate, assuming the synchronization is done with `rsync`, it could be that the timestamps files were downloaded but the synchronization did not finish. However, from my experience this is usually sufficient. If what you are looking for is an accurate and atomic synchronization verification, this is not the tool for you.

CHAPTER 2

Installation

1. Make sure you are using `python3.5` (in venv or global) and run:

```
pip install git+http://gerrit.ovirt.org/mirrorchecker.git
```

2. Running:

```
mirror_checker.py
```

3. See mirrors.yaml for a working example of the configuration file. See mirrors.txt for a working example of separating the mirror sites lists to a different file(referenced in mirrors.yaml).

2.1 Examples

1. See this for an example of deploying *mirrorchecker* in a Docker container, as used in the oVirt project.
2. For building a monitoring alert: see the nagios plugin: mirrorchecker_nagios_plugin

Development

1. *mirrorchecker* works with `python 3.5`, and uses `asyncio` and `aiohttp`, other than that its only major depdency is paramiko.
2. The git repository is at: https://gerrit.ovirt.org/mirrorchecker.git
3. For general information on how to use gerrit, see here.
4. Patches are welcome!

CHAPTER 4

Source code

CHAPTER 5

Indices and tables

- genindex
- search