

Advanced Configuration Options

The purpose of this document is to elaborate on the advanced configuration options offered by pyClowder2.

This list should be expanded as new features are discovered.

- [Logging](#)
- [Mounted Paths](#)
- [Customizing SSL Certificate Bundle](#)
- [Local Processing using Input Path / Output Path](#)

Logging

More information needed on what this does and acceptable/proper configuration values.

Mounted Paths

When running an extractor within a container, it is sometimes necessary or beneficial to map from one an absolute path on the host to a different absolute path within the container.

For example: if Clowder stores its uploaded files at `/Users/myname/clowder/data`, but you want this data mounted into the extractor container at `/home/clowder/data` to avoid embedding an arbitrary username into the path.

To configure such a scenario, pass a stringified JSON map of host absolute path => container absolute path into the `MOUNTED_PATHS` environment variable in the container:

```
docker run -it -v /Users/lambert8/clowder/data:/home/clowder/data/ -e MOUNTED_PATHS '{ "/Users/lambert8/clowder/data":"/home/clowder/data" }' ...
```

The `-v` flag above tells Docker where to map our files, and the `MOUNTED_PATHS` environment variable tells pyClowder about this mapping as well.

Now, when the `process_message` call comes through, the `local_paths` property of the resource metadata passed by the function stores the substituted path(s).

NOTE: In order to leverage the `MOUNTED_PATHS` configuration option, your extractor's `check_message` function must NOT return `CheckMessage.bypass`

Customizing SSL Certificate Bundle

Python `requests` allows the user to specify which TLS bundle is used for communicating via HTTP. This can be helpful for performing deep-packet inspection on each request in a particular environment.

By default, requests will install its own certificate bundle. You can tell requests to use a different bundle by setting the `REQUESTS_CA_BUNDLE` environment variable to the path of the target bundle.

For example, the path to the system certificate bundle for **CentOS** is `/etc/pki/ca-trust/extracted/pem/tls-ca-bundle.pem`. NOTE: On Ubuntu, this path is instead somewhere in `/etc/ssl`.

You can mount this file into the container by passing `-v /etc/pki/ca-trust/extracted/pem/tls-ca-bundle.pem:/etc/pki/ca-trust/extracted/pem/tls-ca-bundle.pem`.

Finally, tell requests to use this bundle by also passing `-e REQUESTS_CA_BUNDLE="/etc/pki/ca-trust/extracted/pem/tls-ca-bundle.pem"`.

Now Python requests will use your custom certificate bundle!

Local Processing using Input Path / Output Path

More information needed on when to use this scenario and acceptable/proper configuration values.