mitmproxy-HTTPolice Documentation

Release

Vasiliy Faronov

August 13, 2016

Contents

1 Installation 3

2 Usage 5

mitmproxy is an advanced HTTP debugging tool that can intercept TLS-encrypted connections, supports HTTP/2, and many more.

mitmproxy-HTTPolice is an inline script for mitmproxy that will check intercepted exchanges and produce an HTTPolice report. It also works with mitmproxy's command-line tool mitmdump.

For recent changes in mitmproxy-HTTPolice, see the changelog.

CHAPTER 1

Installation

```
$ pip install mitmproxy-HTTPolice
```

If this is giving you trouble, see mitmproxy docs and HTTPolice docs for more detailed instructions.

Note that mitmproxy requires a specific version of Python (mitmproxy 0.17 requires Python 2.7).

CHAPTER 2

Usage

To run HTTPolice together with mitmproxy, use a command like this:

```
$ mitmdump -s "`python -m mitmproxy_httpolice` -o html report.html"
```

Note the backticks. Also, you can replace `mitmdump` with `mitmproxy` if you wish.

`-s` is an option for mitmproxy that specifies an inline script to run, along with arguments to that script.

`python -m mitmproxy_httpolice` is a sub-command that prints the path to the script file:

```
$ python -m mitmproxy_httpolice
/home/vasiliy/.local/lib/python2.7/site-packages/mitmproxy_httpolice.py
```

`-o html` tells HTTPolice to produce HTML reports (omit it if you want a plain text report). Finally, `report.html` is the name of the output file.

Now, mitmproxy/mitmdump starts up as usual. Every exchange that it intercepts is checked by HTTPolice. When you stop mitmdump (Ctrl+C) or exit mitmproxy, HTTPolice writes an HTML report to `report.html`.

You can use the `-s` option to silence unwanted notices, just as with the `httpolice` command-line tool:

```
$ mitmdump -s "`python -m mitmproxy_httpolice` -s 1089 -s 1194 report.txt"
```

mitmproxy/mitmdump itself has many interesting options. One of the more useful features is the ability to dump traffic into a file. If you do this, you can then "replay" it as many times as you wish:

```
$ mitmdump --wfile flows.dat
$ mitmdump --no-server --read-flows flows.dat \
>     -s "`python -m mitmproxy_httpolice` /dev/stdout"
```