Timeline Analyzer

Social network timeline analysis tool

User Guide

Radek Burget

TARZAN Project Documentation
Faculty of Information Technology, Brno University of Technology

Last changed: November 4, 2019

TimelineAnalyzer User Guide

TimelineAnalyzer consists of a central RDF respository, *local client applications* that are used for acquiring the event data from social networks and local browser profiles and a *web client application* used for browsing and analyzing the data. The installation of the individual components is described in the Installation Guide. All the tools described below expect that the RDF repository server is properly configured and running.

Local Client Applications

The following three local applications are provided for acquiring the data:

- **DownloadFB** -- obtains events from public Facebook profiles
- **DownloadTwitter** -- downloads events from Twitter profiles
- **DecodeProfiles** -- analyzes the browser profiles on a local filesystems and extracts the events

The applications are controlled via command line interface and may share a common configuration.

Common Configuration

All the command-line tools try to load a common configuration in a file called `config.properties` in the current working directory. The structure of the file is the following:

```
# An example TimelineAnalyzer client configuration
# The repository URL is required
repo.url=http://localhost:8080/rdf4j-server/repositories/timeline
# Optionally, download start date, end date and limit may be specified
download.startDate=2019-01-01
download.endDate=2019-08-25
download.limit=500
```

The repository URL must point to an existing repository on a running RDF4J server. Optionally, the `download.startDate`, `download.endDate` properties may be used for specifying the date of the first event, last event extracted from the corresponding data source. For social network sources, the `download.limit` specifies the maximal number of posts downloaded via the API.

Alternatively, the configuration options may be specified in the command line using the `-D` option, such as

```
java -Drepo.url=http://localhost:8080/rdf4j-server/repositories/ta -jar DecodePr
```

Facebook Downloader

The Facebook downloader extracts the public posts from a Facebook profile and stores the corresponding events to the configured RDF repository. The tool is executed as follows:

```
java -jar DownloadFB.jar <username> [<username> ...]
```

where is a Facebook user name.

Twitter Downloader

The Twitter downloader extracts the posts from a Twitter profile and stores the corresponding events to the configured RDF repository. The tool is executed as follows:

```
java -jar DownloadTwitter.jar <username> [<username> ...]
```

where is a Twitter user name without the leading @.

Browser Profile Decoder

The browser profile decoder extracts the posts from locally stored browser profiles (such as Firefox browser profiles) and stores the corresponding events to the configured RDF repository. The tool is executed as follows:

```
java -jar DecodeProfiles.jar <homedir> [<homedir> ...]
```

where is a path to a user home directory (e.g. /home/john).

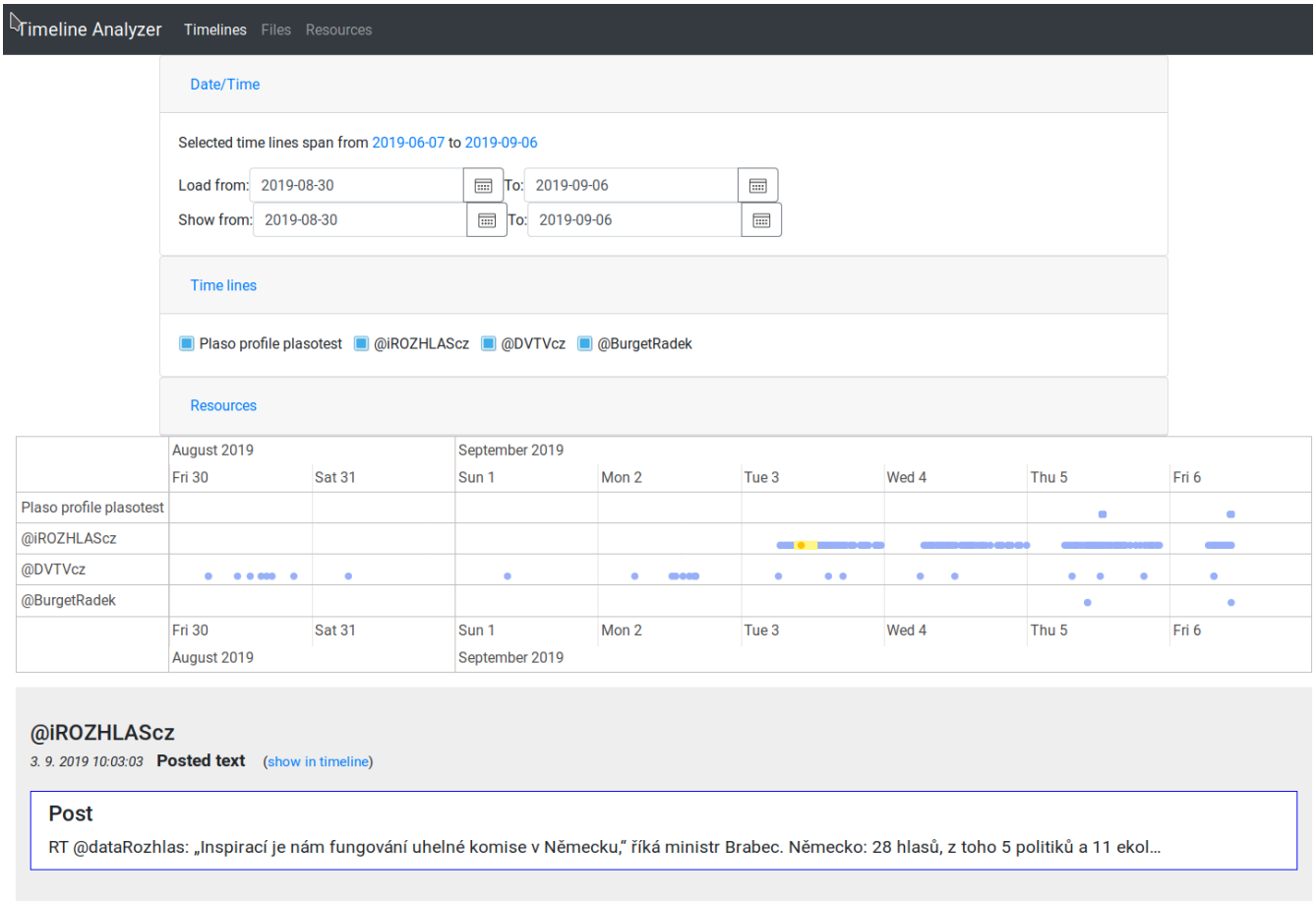
Web Client Application

The web client is accessible with any web browser by entering its configured URL as described in the Installation Guide. It connects the RDF repository and provides a main view

for browsing the whole timeline contained in the repository and two additional views for examining the individual local files and web URLs referenced in the events.

Timeline View

The main timeline view allows to choose the desired event sources such as Twitter profiles, Facebook profiles or local browser profiles to be examined and the desired time span. The events are displayed on an interactive graphical time line as shown below:

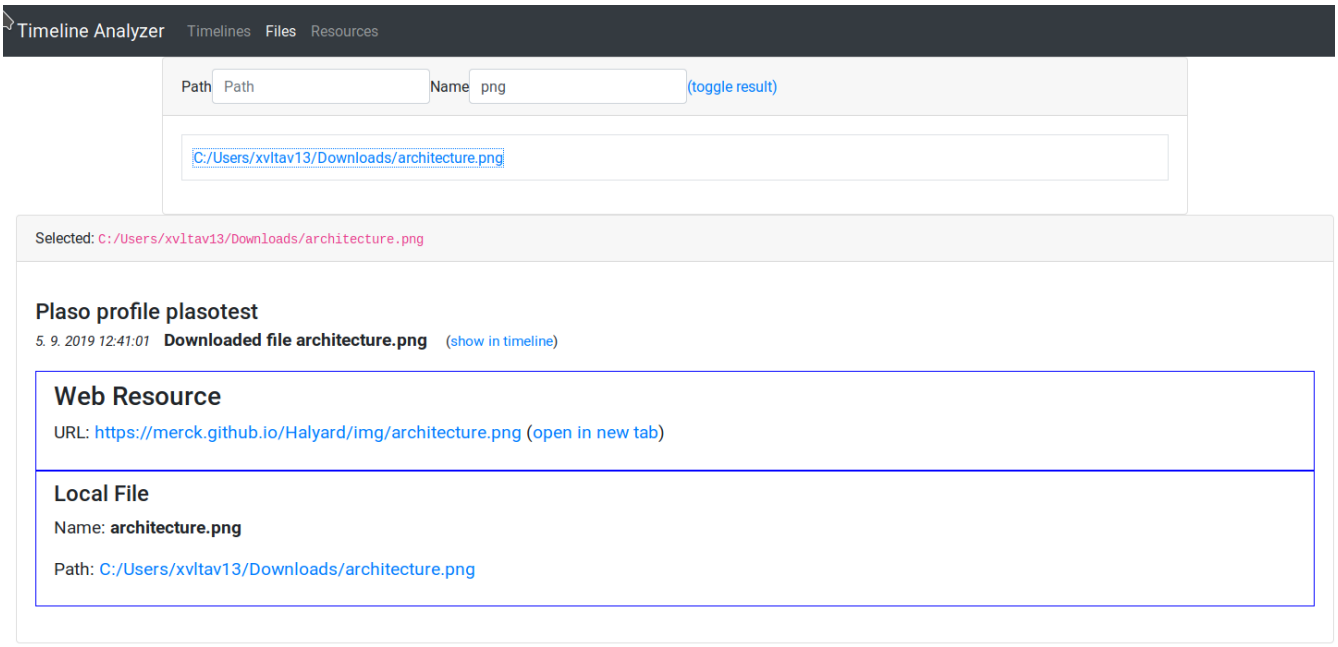


After clicking any event in the timeline, event details are displayed below the timeline including the referenced resources such as local files and web URLs.

Files View

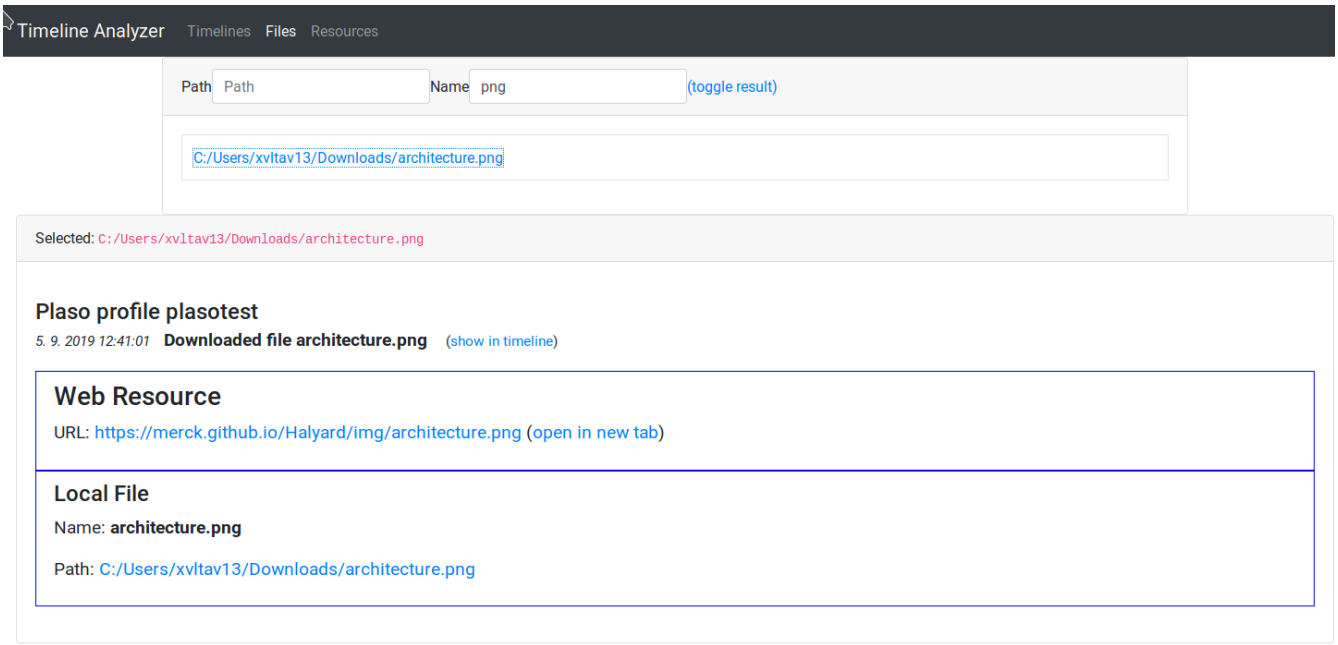
The *Files* view allows to search for local resources referenced in the events. The file may be filtered by their names and paths (even partial). Once a file is clicked, a list of related events is displayed below. By clicking the *Show in timeline* link, the event is shown in the Timeline view. This allows the user to investigate the context of the event (e.g. the related events in the given time).

The screenshot below shows that a PNG file has been discovered and a corresponding download event was found:



Resources View

The *Resources* view is used to search for web resources by their URL. The functionality and the way of use is analogical to Files.



Acknowledgements

This work was supported by the Ministry of the Interior of the Czech Republic as a part of the project Integrated platform for analysis of digital data from security incidents VI20172020062.