Malware Evaluation Based on Behavioural Characteristics

Panagiotis Michalopoulos

University of Patras
Department of Electrical & Computer Engineering

July 20, 2017

The Problem

- Each day large amounts of new malware appear
  - not exactly new: variants of existing malware families
- To defend against it we need to analyse and classify malware samples
- Two methods of analysis: static and dynamic

The Problem

Static Analysis

Performed on the binary of the sample in order to create its profile. We collect static features (file size and type, entropy etc.) and signatures (YARA).

Pros

- Very fast
- Resource friendly

Cons

- Vulnerable to obfuscation techniques (metamorphic, polymorphic malware)
- Cannot detect new variants of a family (zero-day attacks)

The Problem

Dynamic Analysis

The sample is executed in a safe environment (sandbox) and we collect the behavioural artifacts it leaves (files opened, connections established)

Pros

- Can detect obfuscated malware and new variants of a family
- Gives more information about the sample

Cons

- Much slower than static analysis
- Resource intensive (need for VMs)

Solution

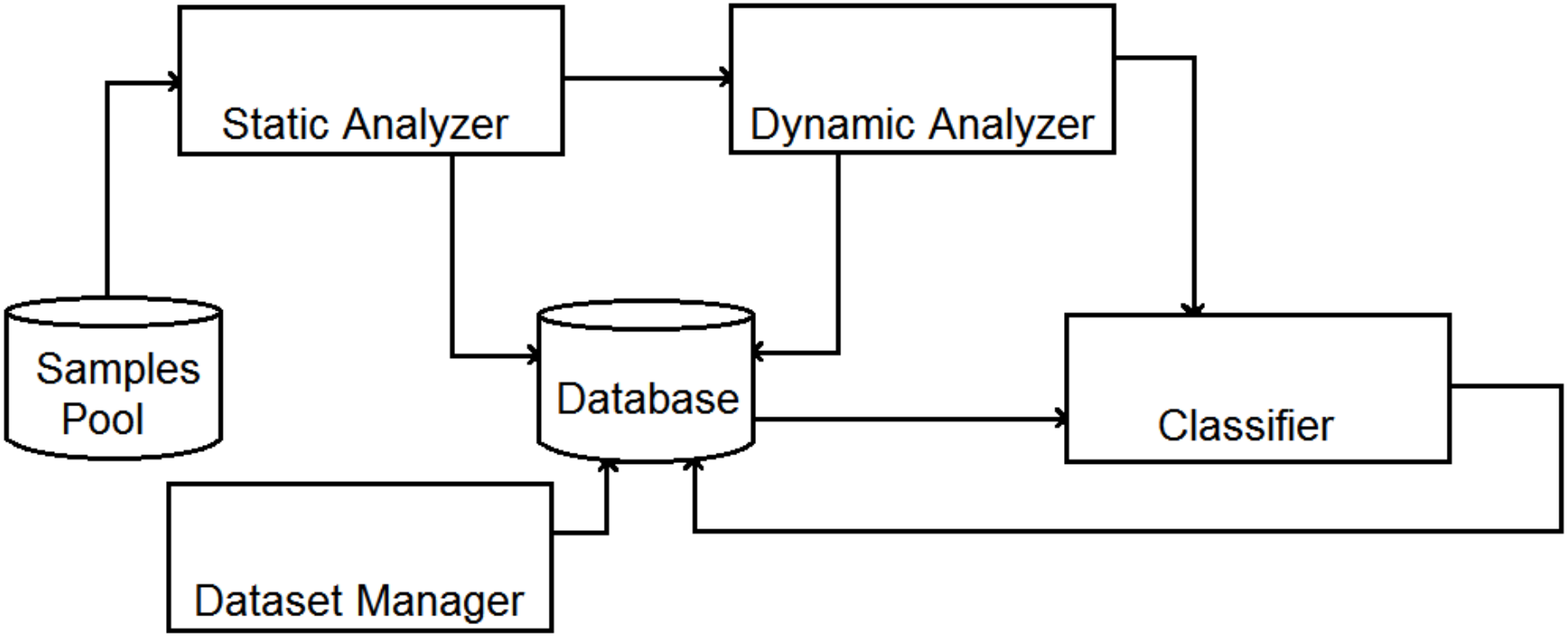
- Build a platform that combines the two analysis techniques
  - Open source: most of the existing solutions are closed source
  - Scalable
  - Able to perform both multiclass and binary classification
- Use static analysis at first
- Employ dynamic analysis when the static falls short (new variants)
- Use machine learning for the classification

Architecture of the Platform

- Two modes of operation
  - training: collect features from know malware in order to train the classifiers
  - classification: collect features from unknown samples and classify them (using SVM)
- Basic components
  - Static analyzer (Laika BOSS)
  - Dynamic analyzer (Cuckoo Sandbox & VirtualBox)
  - Database (MongoDB)
  - Classifiers (scikit-learn)

Architecture of the Platform

The pipeline



Implementation

General overview

- Python is used as the development language
  - All the tools are written in it
  - Fast prototyping
  - Large collection of modules
  - Quality of the documentation
- Installation and configuration of the components
- Custom scripts for the interconnection of the platform's parts
  - Folder monitoring
  - Pipelining
  - Scanning management
  - Dataset management

Implementation

Basic components

- LaikaBOSS:
  - Developed in-house by Lockheed Martin
  - Abundance of information for each sample
  - Support for large scale deployment
  - *Lack of documentation*
- Cuckoo Sandbox
  - Created at Google Summer of Code
  - Highly modular, expandable
  - Only decent open source option
- VirtualBox
  - Integrates well with cuckoo
  - Easy to use, yet powerful
  - Previous experience

Implementation

Basic components

- MongoDB
  - No-SQL database centred around "documents"
  - Allows for increased flexibility
  - More suitable for this kind of application
- scikit-learn
  - Python module for machine learning
  - Wide variety of classifiers
  - Evolves into a de facto standard
  - Previous experience

Showcase

Main loop

- training mode:

```
user@server:/home/framework/src$ sudo ./start.py
Starting laika...
Success!


Processing file:a313d1092c5245da1c20ac05915a3d11
Waiting for submissions... Seems that the file a313d1092c5245da1c20ac05915a3d11
already exists in our database classified as:desert, malware.
 Do you want to resubmit? (y/n)y
Starting static analysis for a313d1092c5245da1c20ac05915a3d11
Static analysis for a313d1092c5245da1c20ac05915a3d11 completed succesfully
Submitting a313d1092c5245da1c20ac05915a3d11 for cuckoo scan...
Submission successful. Now we wait for the results...
Waiting for submissions... Dynamic analysis for a313d1092c5245da1c20ac05915a3d11
 terminated.
Waiting for submissions...
```

- classification mode:

```
user@server:/home/framework/src$ sudo ./start.py
Preparing the classifiers...
Classifiers trained sucsessfully!
Starting laika...
Success!


Processing file:4a0ef41272210f41b987224ff57f6280Waiting for submissions...
Seems that the file 4a0ef41272210f41b987224ff57f6280 already exists in our datab
ase classified as:desert, malware.
 Do you want to resubmit? (y/n)y
Starting static analysis for 4a0ef41272210f41b987224ff57f6280
Static analysis for 4a0ef41272210f41b987224ff57f6280 completed succesfully
Submitting 4a0ef41272210f41b987224ff57f6280 for cuckoo scan...
Submission successful. Now we wait for the results...
Waiting for submissions... Dynamic analysis for 4a0ef41272210f41b987224ff57f6280
 terminated.
Classified as: [2] desert
```

Showcase

Dataset creation

the command...

```
user@server:/home/framework/src$ ./utils.py create -d "This a demo dataset" bcla
ss_demo both malware, benign
user@server:/home/framework/src$
```

Showcase

Dataset creation

the command...

```
user@server:/home/framework/src$ ./utils.py create -d "This a demo dataset" bcla
ss_demo both malware, benign
user@server:/home/framework/src$
```

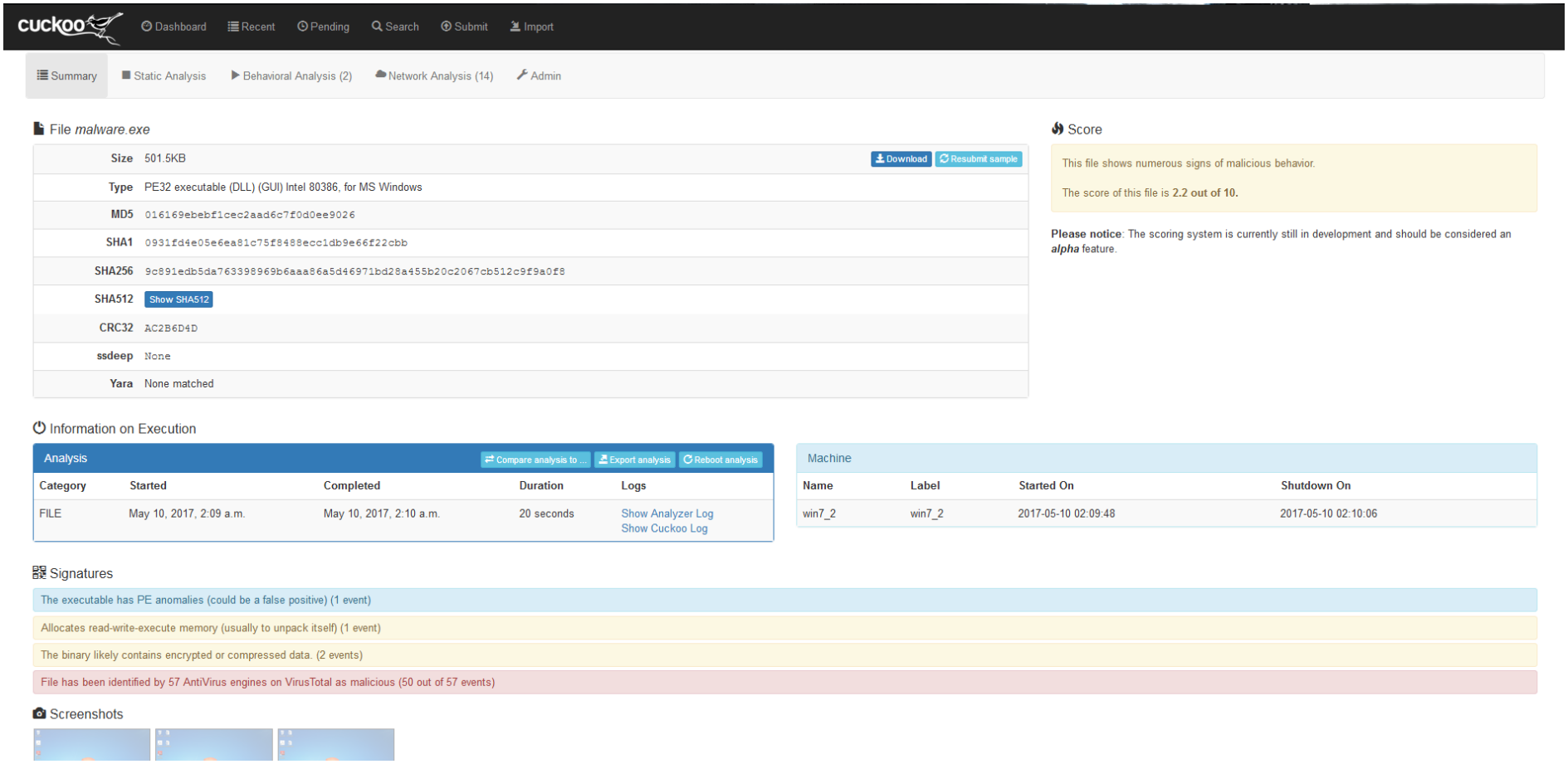
and the proof

58d3233304d6bc49f22c63aa	static	bclass_stat1	test dataset with static for binary classification	Thu Mar 23 2017 03:21:55 GMT+0200 (EET)	58cc779504d6bc20241e5c2b,58cc77b304d6bc20241e5c30,58cc77f004d6bc2
58d324aa04d6bc6bf45a5c18	dynamic	bclass_dyn2	test dataset with dynamic for binary classificati...	Thu Mar 23 2017 03:28:10 GMT+0200 (EET)	58cb115f04d6bc0388e8a1d7,58cb115f04d6bc0388e8a1dc,58cb115f04d6bc03
58d324b604d6bc6c6dabbd28	static	bclass_stat2	test dataset with static for binary classification	Thu Mar 23 2017 03:28:22 GMT+0200 (EET)	58cb115f04d6bc0388e8a1d7,58cb115f04d6bc0388e8a1dc,58cb115f04d6bc03
59068f9a04d6bc08a622b686	both	bclass_demo	This a demo dataset	Mon May 01 2017 04:30:02 GMT+0300 (EEST)	58d3207304d6bc7070727590,58d3209104d6bc7070727595,58d3209104d6bc

Showcase

Web Interface

It is the web interface of cuckoo modified to integrate with the rest of the pipeline (sample submission, static analysis results).



Results

The platform being open source and modular is capable of further expansion and customization.

- Can support a wide range of classifiers and classification schemes
- New modules can be used for extra functionality (android analysis)
- Can be fine tuned for increased performance