An application of Machine Learning to Intrusion Detection and Prevention in Web Applications

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djibrilla.amadoukountche@gmail.com

15/03/2018

Agenda

Data, Data, Data

Intrusion detection in Web applications

IDPS Detection Method: Behavioral Intrusion detection

RoCaWeb

Conclusion

Reading list

DATA (as in a Datalake)

Data, Data, Data

Suppose, as a Data scientist you have been given:

- Logs from web servers
- Logs from developpers (functional tests logs)
- Logs from applications
- Packet captures from your Network admin (or maybe NSA and BND)

And you are asked to get usefull information from these data for intrusion detection and prevention in web applications.

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djib-rilla.amadoukountch

Web application

Our first problem will be a definition problem. What is a Web application ?

Web application

Our first problem will be a definition problem. What is a Web application ?

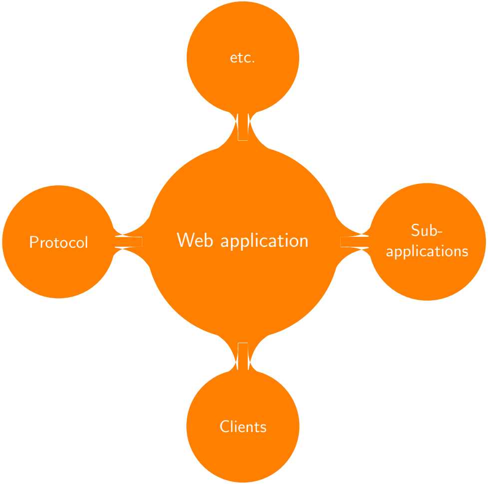


Figure: An attempt to represent a web application

Clients

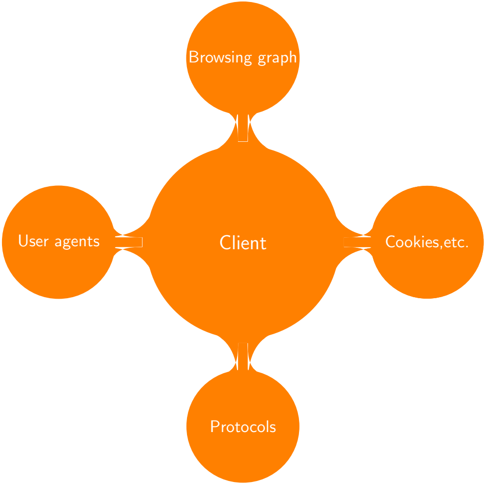


Figure: An example of client model

HTTP protocol

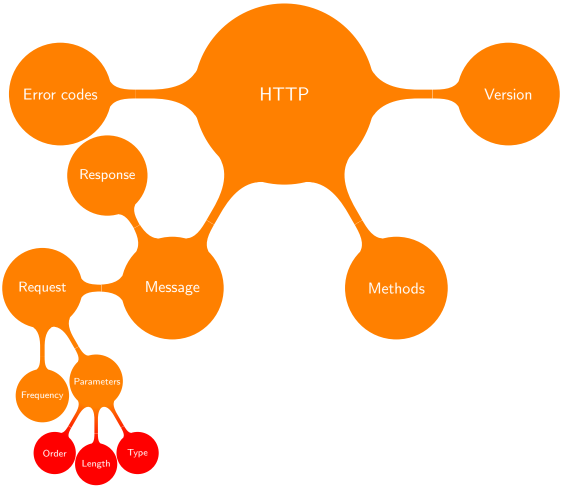


Figure: Data obtained from the HTTP protocol

Message

```
generic-message = start-line
                  *message-header
                  CRLF
                  [ message-body ]

start-line      = Request-Line | Status-Line
```


Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djibrilla.amadoukountch

Request

```
POST / HTTP/1.1
Host: meinesite.de

/path/code?param1=value1&param2=value2
```

Response

```
HTTP/1.1 200 OK

...
(trocated)
```


Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djibrilla.amadoukountche

Parameters

Consider captured values from a POST Request

$/\text{path}/\text{code}?param_1 = value_{11} \& param2 = value_{21}$
$/\text{path}/\text{code}?param_1 = value_{12}$

$$\ldots$$

$/\text{path}/\text{code}?param_1 = value_{1n} \& param_2 = value_{2n}$
Example

The problem

Web intrusion: Origin of the problem

How many of you follows Secure Development Life cycles ?

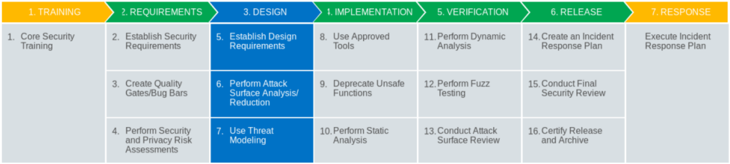


Figure: Microsoft SDL. Source

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djib-rilla.amadoukountche

Web intrusion: Origin of the problem

Complex Web applications ?

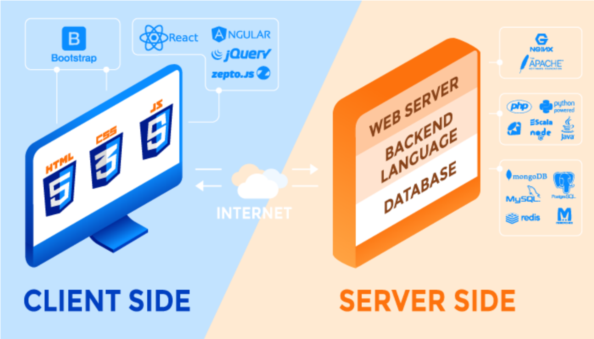


Figure: Example of the diversity of web technologies. Source

Web intrusion: Origin of the problem

An attacker has many opportunities...

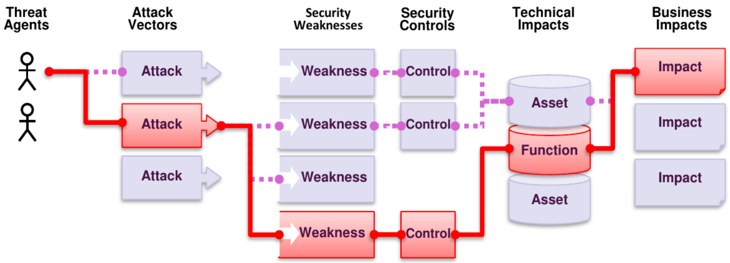


Figure: An attacker only needs the weakest link. Source

Web intrusion: Origin of the problem

An attacker only needs the weakest point.

Figure: An attacker only needs the weakest link.

How to solve the problem

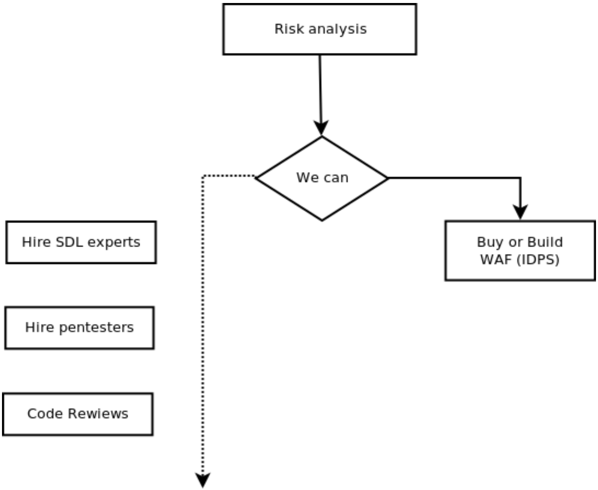


Figure: An approach to solve the problem

A solution ?

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djibrilla.amadoukountche

What is an IDPS ?

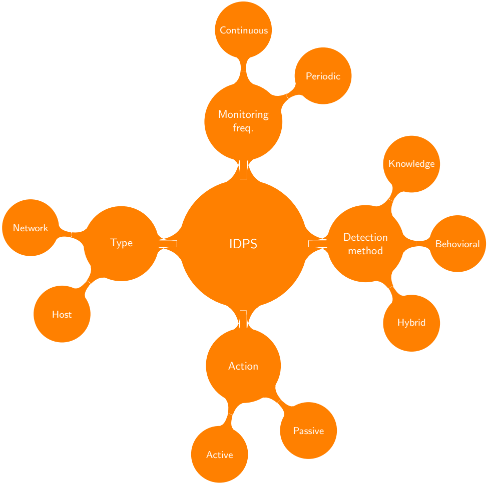


Figure: Components and different types of IDPS

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djib-rilla.amadoukountch

IDPS Detection Method: Knowledge-based

Figure: Knownledge-based intrusion detection. Source

IDPS Detection Method: Knowledge-based

Issues with the oracle:

- Frequency of updates of the databases
- Polymorphic malicious programs
- *Zeroday* attacks

Behavioral Intrusion detection

Figure: Is something abnormal with Mr. Zuckerberg ? Source

Behavioral Intrusion detection

Issues with the normal behavior:

- Is it really possible to model (cover) the behavior ?
- What about new behaviors (new trends) ?
- False positive rate

Algorithmic approaches used for Behavioral Intrusion detection

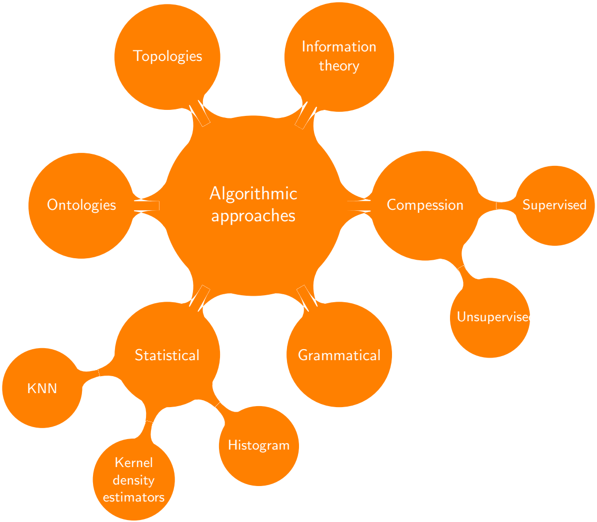


Figure: Algorithmic approaches

What we consider as Learning

Imporant remark

Learning is the process where we build a model representing the normal behavior of the application.

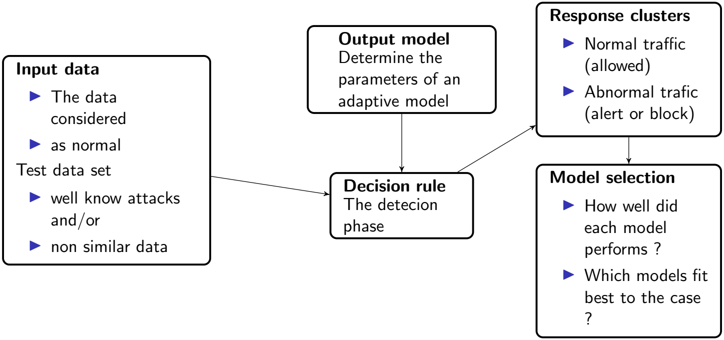


Figure: The overall view of the learning phase

How well are we learning

Performance measures

- Classification error rate
- False positive rate
- False negative rate
- Execution time

	True class	
	Normal	Abnormal
Normal	TP	FP
Abnormal	FN	TN

Librairies

Environment for DeveLoping KDD-Applications Supported by Index-Structures

ELKI

The most interesting libraries for anomaly detection

- Implemented in Java
- License GPLv3
- Anomaly detection algorithms:
  - Distance-based
  - Local factor family
  - Angle-based
  - clustering based
  - etc.

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djibrilla.amadoukountch

Librairies

Lingpipe

- Implemented in Java
- Free and Paid License
- Anomaly detection algorithms:
  - General machine learning algorithms
  - Sentiment analysis algorithms (LDA)

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djib-rilla.amadoukountch

Librairies

Torch

- Implemented in LuaJIT C, C++
- BSD license
- GPU support
- Anomaly detection algorithms:
  - Statistical libraries such as cephes
  - General purpose machine learning algorithms
- Integration in Modsecurity

Author: Djibrilla Amadou Kountche
Email: djib-rilla.amadoukountch

RoCaWeb

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djib-rilla.amadoukountche

RoCaWeb

About

- A research project funded by a RAPID grant
- Aimed building a WAF using Behavioral mehtods
- Implemented in Java, Lua, Scala
- Open source GPLv3 licence
- https://github.com/dakountche/RoCaWeb

RoCaWeb: Architecture

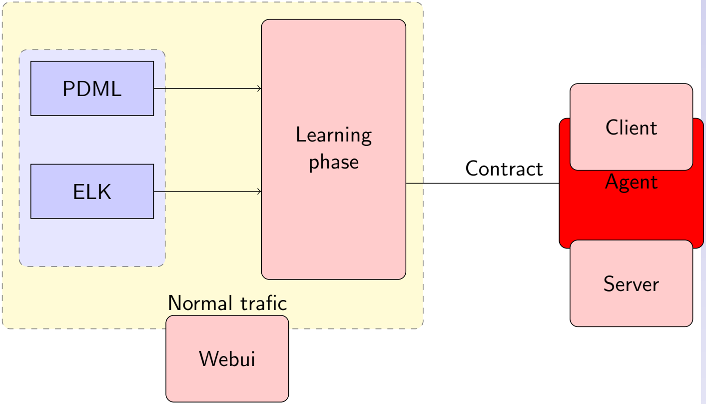


Figure: The RoCaWeb Architecture

RoCaWeb: Learning algorithms

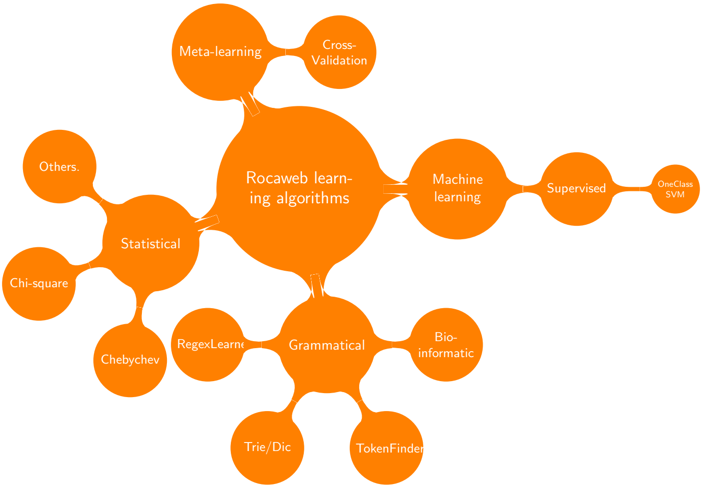


Figure: Learning algorithms considered for RoCaWeb

Grammatical inference

Definition

Let:

- Σ be a finite alphabet of symbols
- Σ^* be the set of all finite strings of symbols from Σ
- *langage* be any sub-set of Σ^*

The learning problem

Given:

- a *langage* L define over Σ
- a sequential or structured data (strings, words) drown form L

Determine the automata or a grammar of L.

Output model

A grammar explaning the data (See the Chomsky Hierarchy)

Grammatical inference: Example

Example

- $A = [a - zA - Z0 - 9]$
- L is a langage define over A
- Observed set: Ich, esse, gern, in, Essen

Infer the grammar of this langage
Is “J’aime manger à Essen” normal given this langage ?

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djib-rilla.amadoukountch

Grammatical inference: RegexLearner

Learning phase

Given:

- S: a set of strings
- R set of Regular expressions
- Find r, the most appropriate regex describing S

Detection phase

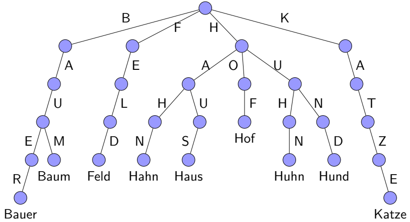
Given a string s:

- s is normal if it validates r
- abnormal otherwire

Grammatical inference: Trie structure

Learning phase

- Given S a set of strings
- Build a trie



Detection phase

Given a string s:

- s is normal if it is in the Trie
- abnormal otherwire

Grammatical inference: Sequence alignment

Origin

- Bio-informatics: computational approach to molecular biology.

Usage

Where given:

- An alphabet: $\alpha = \{A, G, T, C\}$
- And a langage S define on α
- The goal is to compare the sequences from S:
- Aligning these sequences can allow to:
  - Compare species

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djib-rilla.amadoukountch

Grammatical inference: Sequence alignment

Origin

- Bio-informatics: computational approach to molecular biology.

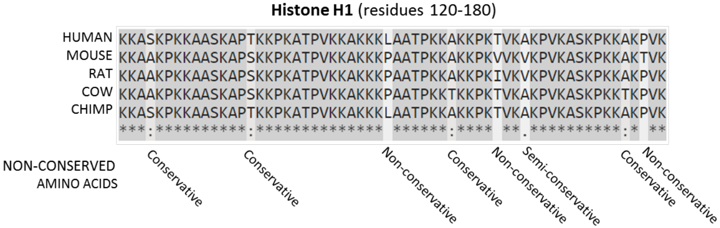


Figure: Example of sequece alginment (From Wikipedia)

Grammatical inference: Sequence Alignment

Definition : Pair alignment

Given:

- $s = s_1 \ldots s_n$ and $t = t_1 \ldots t_m$ two strings defined on an alphabet Σ
- β the gap character, $\beta \ni \Sigma$
- $\Sigma' = \Sigma \cup \beta$

Summarizing alignment

- Insertion : the gap characters in the first string;
- Deletion : the gap character in the second string;
- Match or correspondence: the 2 characters are identical
- Mismatch : the two characters are different from each other and gap

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djib-rilla.amadoukountch

Grammatical inference:Example of Pair Alignment

`Andreas Wulfes`	`andreas wulfes`
`An--ja--------`	`an--ja--------`
`Andreas Wulfes`	`andreas wulfes`
`Annika- ----S.`	`annika- ----s.`
`Andreas Wulfes`	`andreas wulfes`
`-----a----rne-`	`a--r-------ne-`
`Andreas Wulfes`	`andreas wulfes`
`-Bernd Bäumler`	`-bernd bäumler`
`Andreas Wulfes`	`andreas wulfes`
`-Chr--------is`	`-chr--------is`
`-Andreas Wulfes`	`-andreas wulfes`
`Christine Gangl`	`christine gangl`
`Andreas Wulfes`	`andreas wulfes`
`----Con ---Con`	`----con ---con`

Pair alignment: Approaches

Dynamic programming

- Exact solution: Dynamic programming approach
- BLAST

Dynamic programming: Score matrix

- The first column and the first row are filled with $(j \times penalty)$ and $(i \times penalty)$
- Any other cell is filled with :

$$M(i,j) \quad = \quad \max \begin{cases} M(i-1,j) + penalty & \text{(i)} \\ M(i,j-1) + penalty & \text{(d)} \\ M(i-1,j-1) + d(s_i, t_j) & \text{(m)} \end{cases} \tag{1}$$

Grammatical inference: Sequence alignment

Definition : Score

$$\delta(s', t') = \sum_{i=1}^{l} d(s'_i, t'_i) \quad (2)$$

$$d(s'_i, t'_i) = \begin{cases} \text{match} & \text{if } s'_i, t'_i \in \Sigma \text{ and } s'_i = t'_j \\ \text{mismatch} & \text{if } s'_i, t'_j \in \Sigma \text{ and } s'_i \neq t'_j \\ \text{penalty} & \text{if } s'_i \text{ or } t'_j \text{ equals } \beta \end{cases} \quad (3)$$

Grammatical inference: Multiple sequence alignment

Multiple sequence alignment

Given:

- n sequences $s_1, \ldots, s_n$ defined on an alphabet Σ having different lengths
- β is the gap character

A multiple alignment of $s_1, \ldots, s_n$ is an $n - uplet(s'_1, \ldots, s'_n)$ of length $l \geq |s_i|, i \in [1 \ldots n]$ on the alphabet Σ' where the following conditions hold :

1. $|s_i| = |s_j|, \bigvee i, j \in [1 \ldots n]$
2. $h(s'_i) = s_i, \bigvee, i \in [1 \ldots n]$
3. there is no row in the $(n \times l)$-matrix where there is only gap characters.

Grammatical inference: Multiple sequence alignment

Solution

- Exact computation (Difficult)
- Combining with pair alignment (Heuristic)

Algorithms

- UPGMA
- T-Coffee
- etc.

Grammatical inference: Example Multiple sequence alignment

Parameters

- Pair alignment algorithm : Needleman Wunsch
- Gap character : $+$, $match = 1.0$, $mismatch = 0.0$, $penalty = -1.0$

Learning Data Set

```
GARFIELD THE LAST FAT CAT
GARFIELD THE FAST CAT
GARFIELD THE VERY FAST CAT
THE FAST CAT
```

Table: The GARFIELD dataset.

Grammatical inference: Multiple sequence alignment

```
GARFIELD THE VERY FAST CAT
+++++++++THE+++++ FAST CAT
GARFIELD THE+++++ FAST CAT
GARFIELD THE LAST FA+T CAT
```

Learning the regular expression

Learning the regular expression

- Alignment does not generate regular expression;
- Sequences of same size;
- Learn the corresponding regular expression.

Result of AMAA																									
G	A	R	F	I	E	L	D		T	H	E		V	E	R	Y		F	A	S	T		C	A	T
+	+	+	+	+	+	+	+	+	T	H	E	+	+	+	+	+		F	A	S	T		C	A	T
G	A	R	F	I	E	L	D		T	H	E	+	+	+	+	+		F	A	S	T		C	A	T
G	A	R	F	I	E	L	D		T	H	E		L	A	S	T		F	A	+	T		C	A	T
[GARFIELD]{0,1}THE[LAST]{0,1} FA[S]{0,1}T CAT																									

Table: Generation of the regular expression for the GARFIELD dataset.

The Chebychev inequality applied to the length

Learning phase

Given:

- a parameter A of a request ;
- $A = \{a_i, i = 1...n\}$ values of this parameters are collected

Determine:

- μ : the sample mean of the lengths of a_i;
- σ : the variance ;

The Chebychev inequality applied to the length

The detection phase

Soient :

- a_k: the current value to test
- μ, σ : the mean and variance.

Variante :

$$p(|X-\mu| > |l-\mu|) < p(l) = \begin{cases} \frac{\sigma^2}{(l-\mu)^2} & \text{if l} \geq \mu \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

l the current length. Return $p(l)$

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djibrilla.amadoukountch

Character distribution

Learning phase

- Determine the reference character distribution

Given:

- $A = \{a_1, a_2, ..., a_n\}$;
- Σ an alphabet;
- a_i defined on Σ^*.

A character distribution is défined by :

$$CD = \{n_i, i = 1..k\} \tag{5}$$

Character distribution

Reference character distribution

- For each value of the parameter determine its CD
- $RCD = \{f_i = n_i/k, i = 1...k\}$ is the mean CD
- Where k is the size of the alphabet

Detection phase

Given:

- the CD a value and its length
- the RCD computed earlier

Determine the value of the χ^2 probability

RoCaWeb: Agent

Reserve proxy

- Apache configure as Reverse proxy for each website

Modsecurity

- Contract are formated in the Modsecurity format
- Validation for each method is implemented in Lua
- Modsecurity phases:
  1. Request header
  2. Request body
  3. Response Header
  4. Response body
  5. Logging

RoCaWeb: User interface

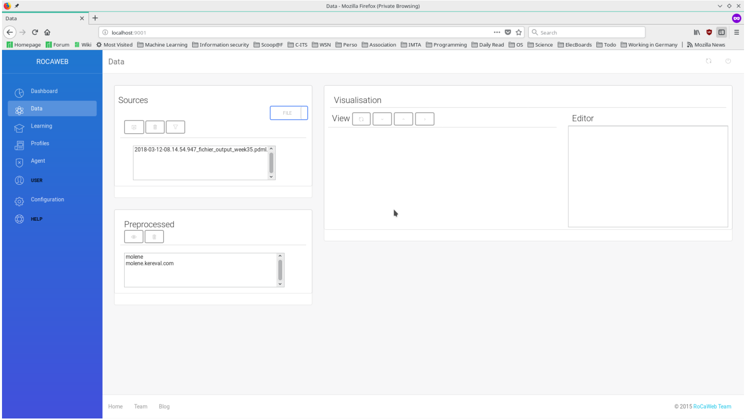


Figure: RoCaWeb User Interface

RoCaWeb: The docker platform

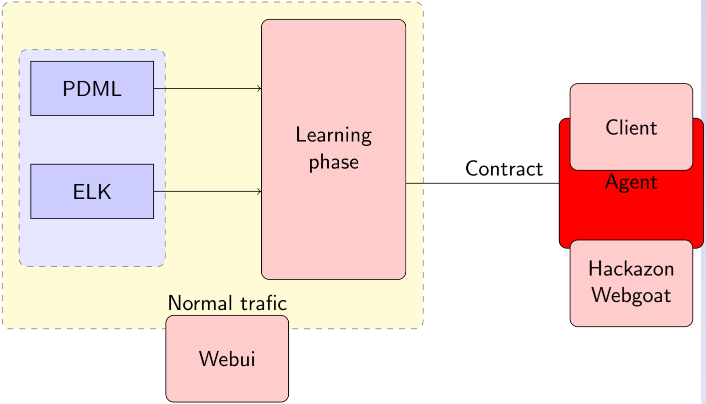


Figure: The RoCaWeb docker platform

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djibrilla.amadoukountch

Similar projects

- Apache metron
- Vulture

Conclusion and perspectives

- Fondamental limitation of Behavioral based

Event:Ruhrgebiet AI & Data Science Meetup
Author: Djibrilla Amadou Kountche
Email: djib-rilla.amadoukountche

Reading list I

Anamaly detection

- Anomaly detection: A survey, Varun Chandola

Grammatical inference

- A bibliographical study of grammatical inference, Colin de la Higuera

Fondament Limitation of Anomaly detection:

- http://all.net