Continuous location validation of cloud service components

Philipp Stephanow, Mohammad Moein, Christian Banse

Fraunhofer AISEC, Germany

13th December 2017, CloudCom 2017, Hong Kong

Introduction

Who we are and what we do

The Authors

- **Fraunhofer**-Institute for **A**pplied and **I**ntegrated **SEC**urity
- Research institute solely focused on IT security (~ 100 employees)
- Located in Munich (main office) and Berlin
- Part of the Fraunhofer Society, biggest applied research organization in Europe (~ 20.000 employees)

Philipp Stephanow, Senior Researcher in Cloud Service Certification

Mohammad Moein, Student Researcher

Christian Banse, Senior Researcher in Cloud and Network Security and Deputy Head of Department

Motivation

Fraunhofer AISEC

- Service or data location is regarded as one of the key decision criteria for companies in choosing cloud providers
- It is incorporated into many certificates and regulations, especially in Europe (BSI C5, EU GDPR, …)
- Depending on the service model, a change of location is not in the control of the customer
- Service location might not always be transparent, especially if using SaaS

Main Contributions

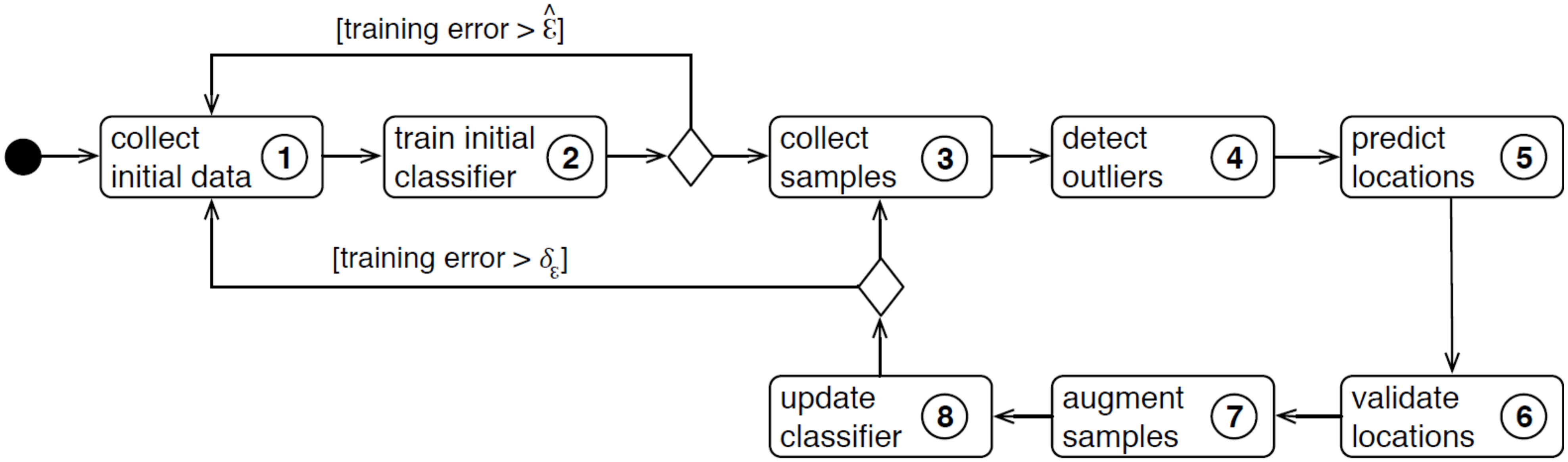
- Design of a process to classify geographical locations of virtual resources using Machine Learning ("location fingerprint")
- Continuous execution of process including measures to counter the "concept drift"
- Experimental evaluation of the process and method using 14 locations of Amazon Web Services (AWS)

Adaptive Location Classification

Designing the process

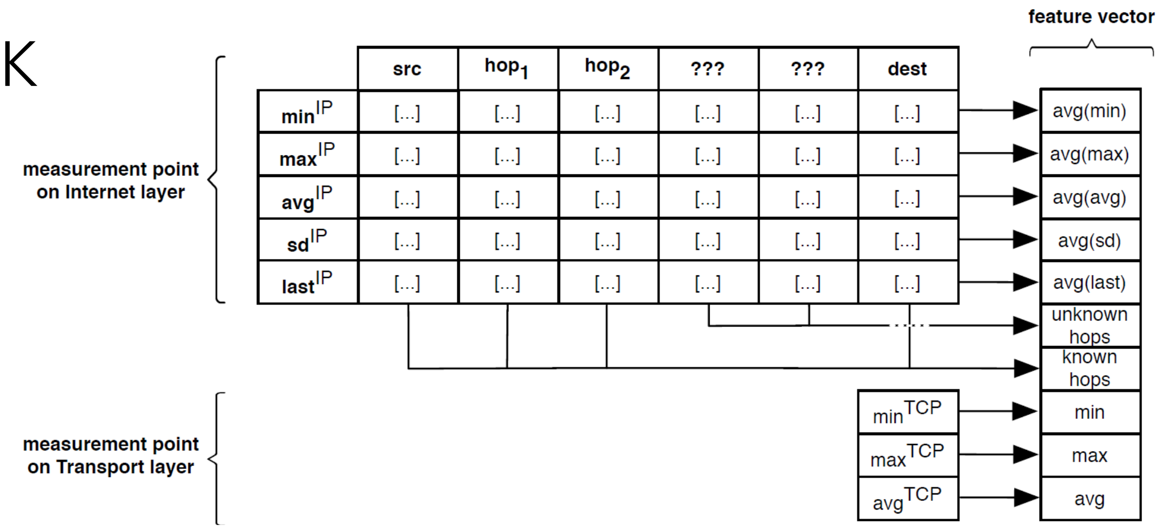
The process

- Goal: detect **changes** in a resource location
- Target: virtual resource with a (public) IPv4 address



Data Collection (Step 1)

- Internet layer
  - IPv4 traceroute (path + delay of hops)
  - Measurement is executed multiple times; *min*, *max*, *sd* are recorded
- Transport layer
  - Delay between SYN and SYN-ACK of the TCP three-way handshake
- Application layer
  - Not in scope of this paper; however we working on it



Training (Step 2)

- Input is the feature vector collected in the first step
- An appropriate supervised learning algorithm needs to be selected, i.e. k-NN or SVM (Linear SVM works good)
- We can calculate the training error ε to adjust parameters of the data collection, i.e. number of measurements (10 is good)
- Output: prediction model

Detection (Steps 5 and 6)

- To classify locations at a latter stage
  - Collect samples again (same as in the first step)
  - Apply the training model to let the classifier classify a location
- We do not want to rely on a single classification because of training errors
- Solution: Consider a sequence of location detections within a time interval by introducing an invalidation window size $|w_l^-| \geq \frac{\log v_l^-}{\log \varepsilon}$
  - Can be configured by a parameter v_l^-
  - Depends on the training error ε

Updating (Steps 4, 7 and 8)

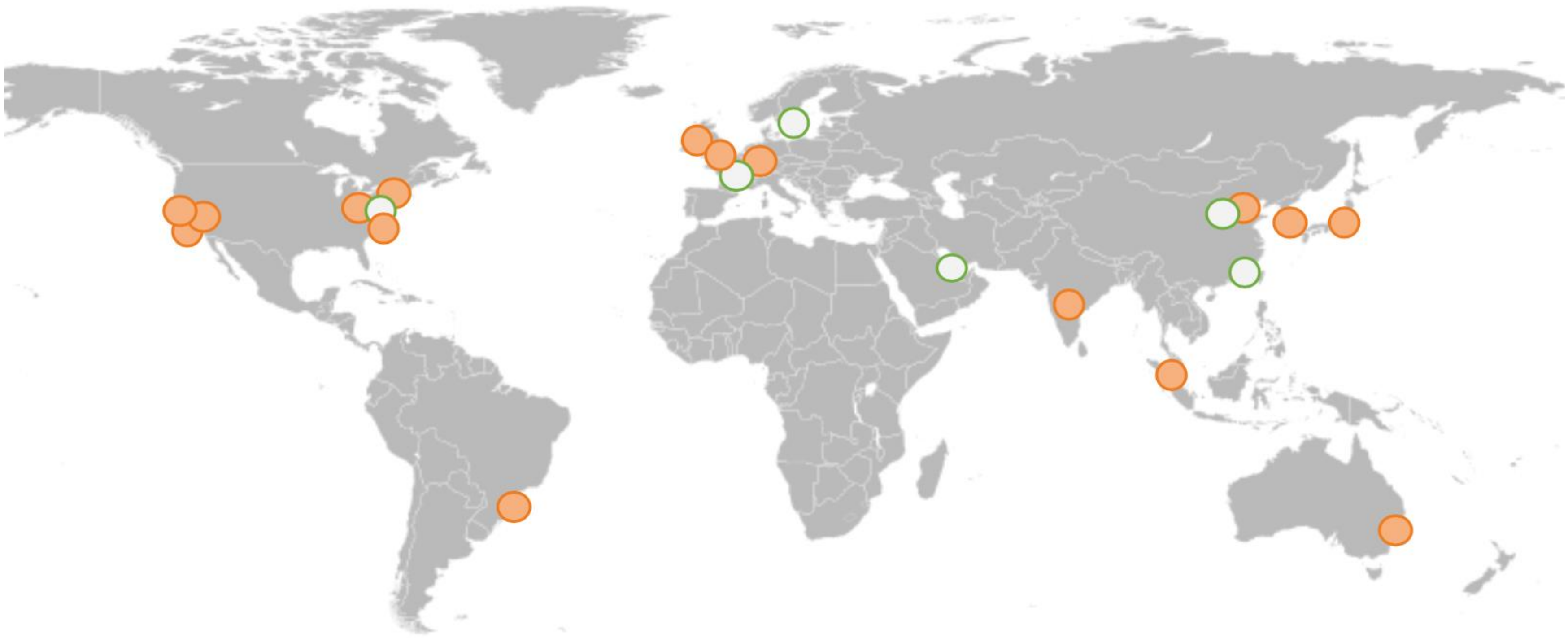
- After detection, we update the training model using the data fed into the classifier
- Before adding, we remove potential outliers using appropriate algorithms, i.e. one-class SVM
- Stop condition: We define a maximum training error *after* updating δ_ε, if the training error ε exceeds this, the process is stopped
- The new training error automatically configures the invalidation window size w_l^- (the higher the error, the larger the window)

Evaluation

Trying it out…

Setup in AWS

At the time of the experiment, 16 geographic *regions* in AWS
1 *region* = multiple *availability zones* (usually 2-3)



Setup in AWS

Fraunhofer AISEC

- 14 EC2 instances in 14 regions (excluding Beijing and AWS Gov Cloud)
- Instances with public IPv4 address with security groups that enable ICMP and SSH
- Origin of measurement was also in AWS, Frankfurt

Data Collection

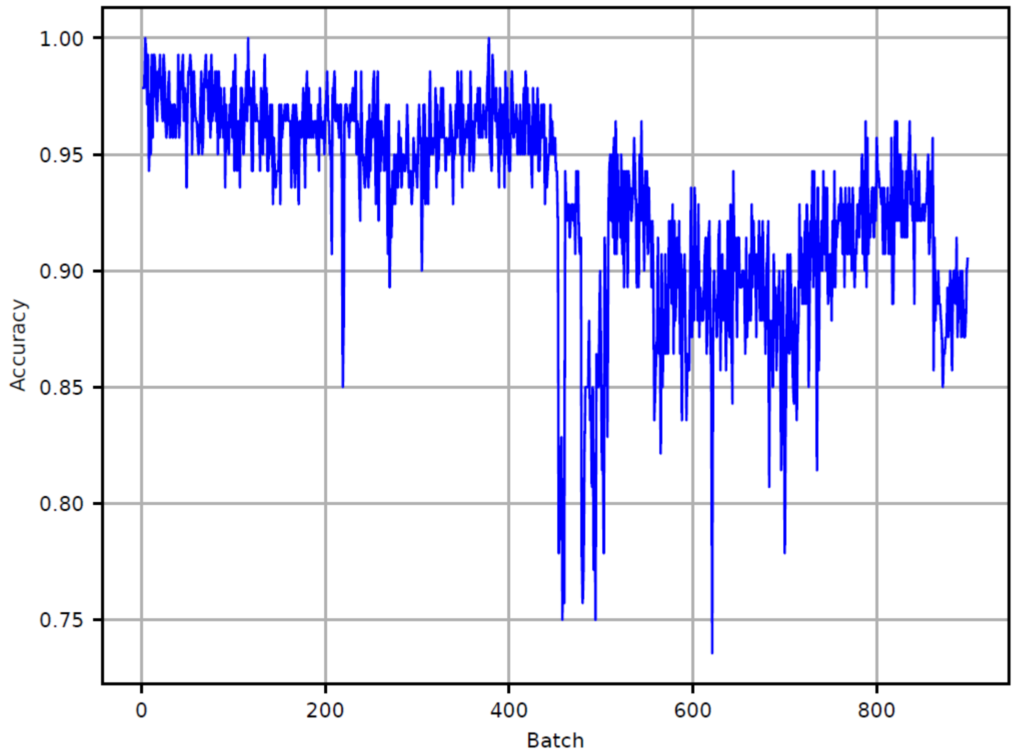
- *mtr* to gather traceroute and *nping* to collect TCP delay (port 22)
- Experiment duration
  - 17th December 2016 – 23rd December 2016
  - 15th December 2016 – 3rd January 2017
- In total 139699 delay measurements

Training

- Implemented using *scikit-learn* using the *LinearSVC* classifier
- 10% of the data used as the training set
  - Upper bound on the training error of $\hat{\varepsilon} = 0.0327$
  - We tolerate training error after updating $\delta_\varepsilon < 0.35$

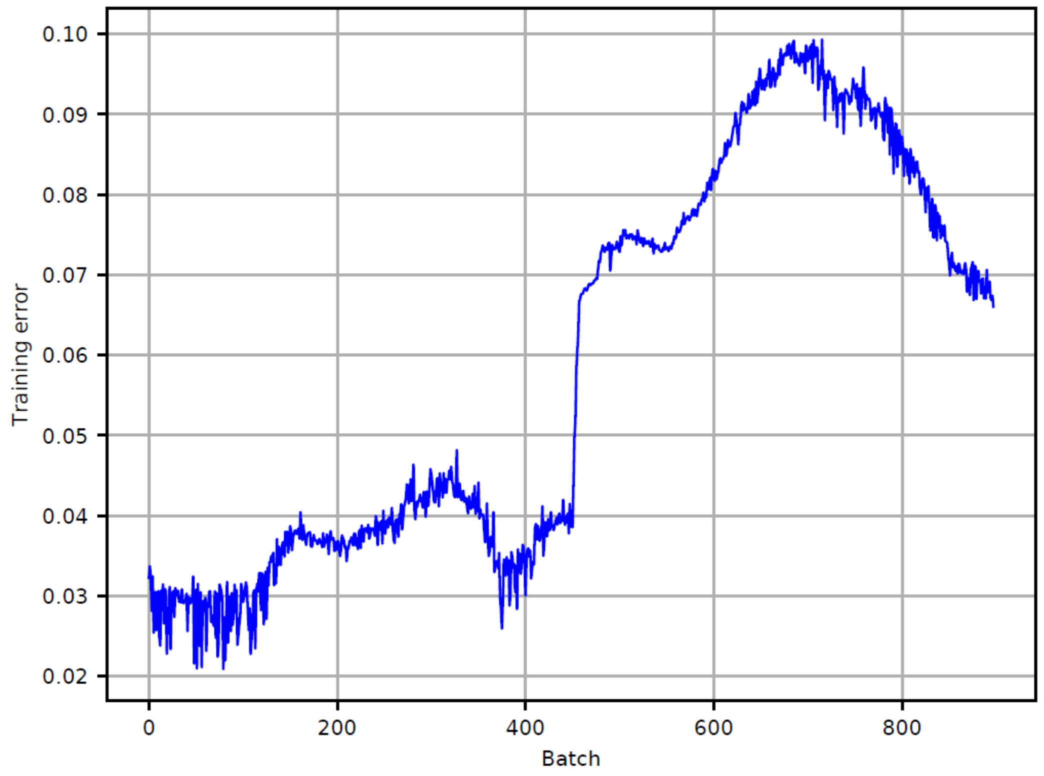
Detection

- Remaining 90 % of the dataset are used as the test set
- Split up in 898 successive batches
- Each batch simulates the Collect new samples step of the process
- Location is predicted and compared to the expected value

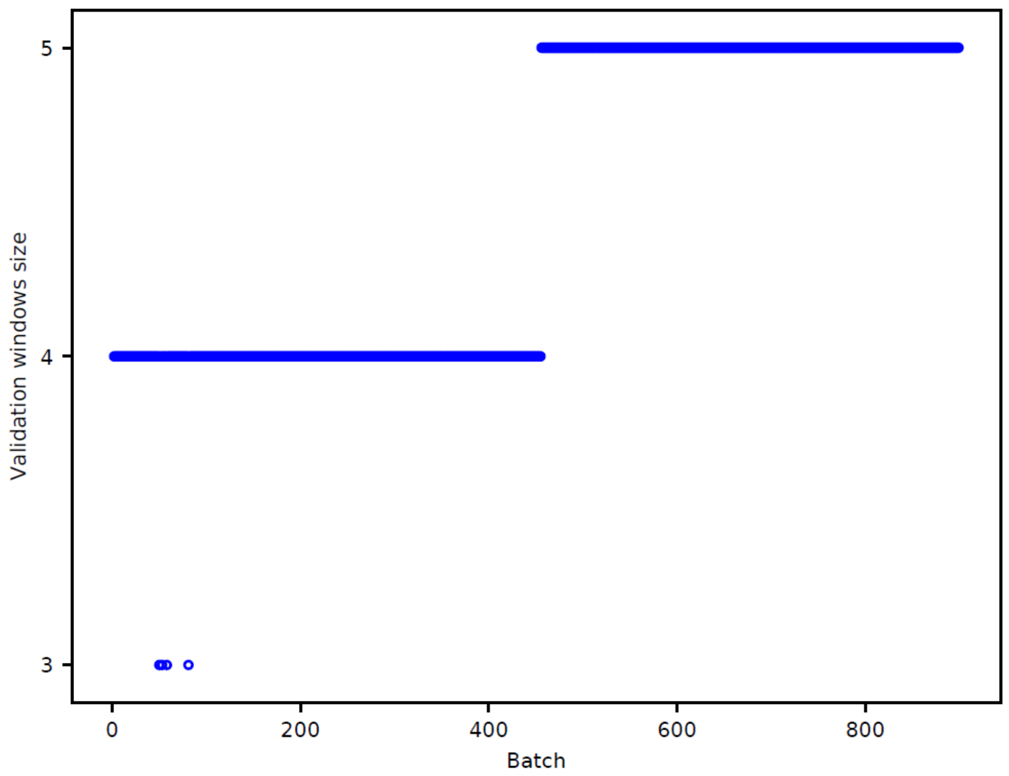


Training error vs. window size

Observed training error



Invalidation window size



Result

Fraunhofer AISEC

- Test accuracy varies between 73.57 % and 100 %
- However, during the experiment, the invalidation window size was never exceeded
- As expected, no location change was observed during the experiment

Parameter	$\bar{x}$ (%)	$\hat{x}$ (%)	sd (%)	max (%)	min (%)
Test accuracy per batch	92.96	94.28	4.35	100	73.57

Conclusions

… and Future Work

Conclusions

Fraunhofer AISEC

- Introduction of an adaptive process to detect changes in the location of virtual resources
- Demonstration of feasibility by evaluating 14 AWS regions
- SVM classifier performed very well during evaluation (avg 92.96 %)

Limitations and Future Work

Fraunhofer AISEC

- We need to further study the affect of L2/L3 load balancers on the measurements
- Extend research from *service* location to *data* location
- Investigate performance of other classifiers, such as Random Forest
- Apply more sophisticated methods to detect concept drifts

Questions?