Detecting of Hidden Anomalies in DNS Communication

Ondrej Filip / ondrej.filip@nic.cz - presenter
Ondrej Mikle-Barat / ondrej.mikle@nic.cz
Karel Slaný / karel.slany@nic.cz

CZ DOMAIN REGISTRY

Outline

- Motivation
- Method description
  - original work
  - algorithm
  - DNS specifics
- Experiments
  - set-up
  - results
- Conclusion

Motivation

- Most of the internet communication starts with a DNS query.
  - There is a possibility to track communication at a certain level of DNS hierarchy.
    - e.g. for intrusion detection, botnet discovery
- We want a tool that is able to:
  - detect suspicious behaviour
  - scan high volume traffic
  - detect low volume anomalies
  - works in real-time = low computation cost
  - does not need any initial knowledge about the analysed traffic
- Will the tool be able to detect something at a ccTLD?

Original work

Extracting Hidden Anomalies using Sketch and Non Gaussian Multiresolution Statistical Detection Procedures by G. Dewaele, K. Fukuda, P. Borgnat, P. Abry, K. Cho

- Blindly analyses large-scale packet trace databases.
- Able to detect short-lived anomalies as well as longer ones.
- Detection method is sensitive to statistical characteristics.
- Promises a very low computation cost.

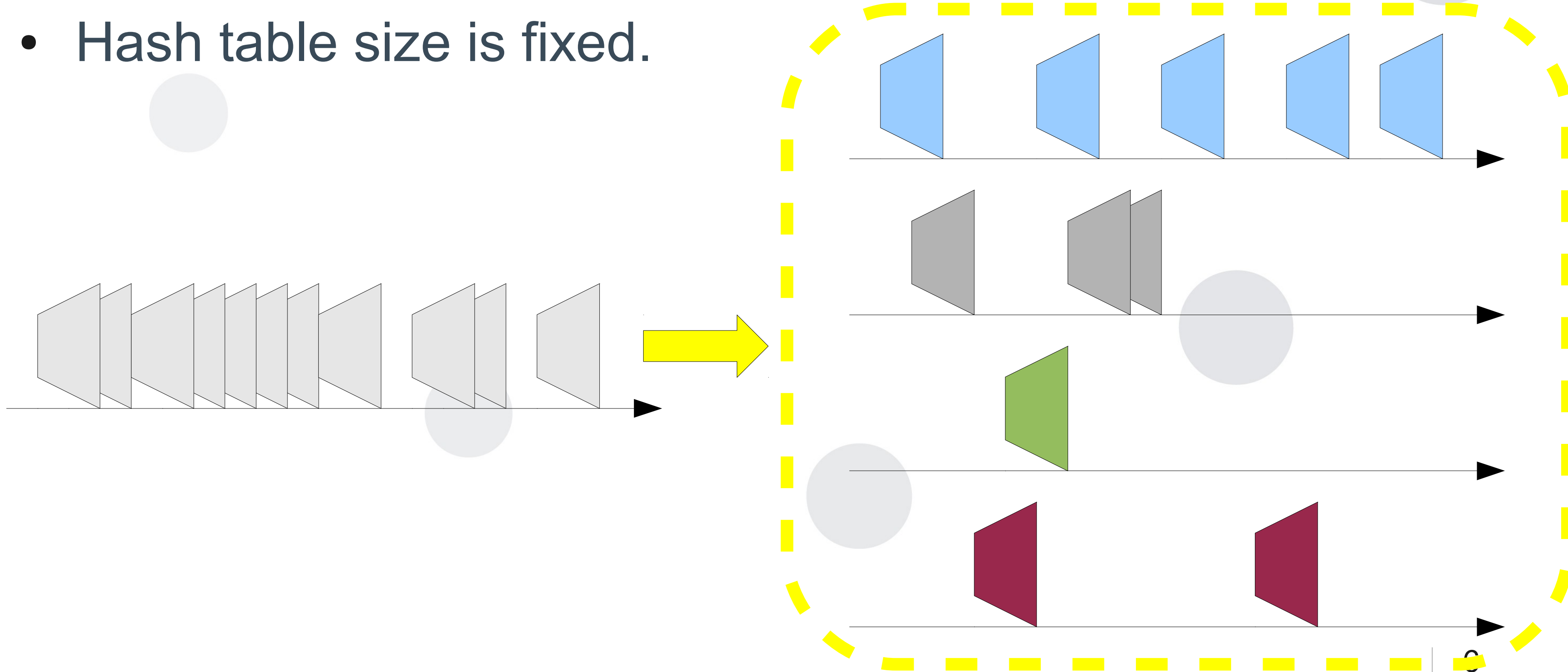
Method description

- The algorithm analyses the traffic using a sliding time-window within which the analysis is performed.
- The analysis iterates over following steps:

  1) random projection - sketches
  2) data aggregation
  3) Gamma distribution estimation
  4) reference values computation
  5) distance from reference evaluation
  6) sketch combination and anomaly identification

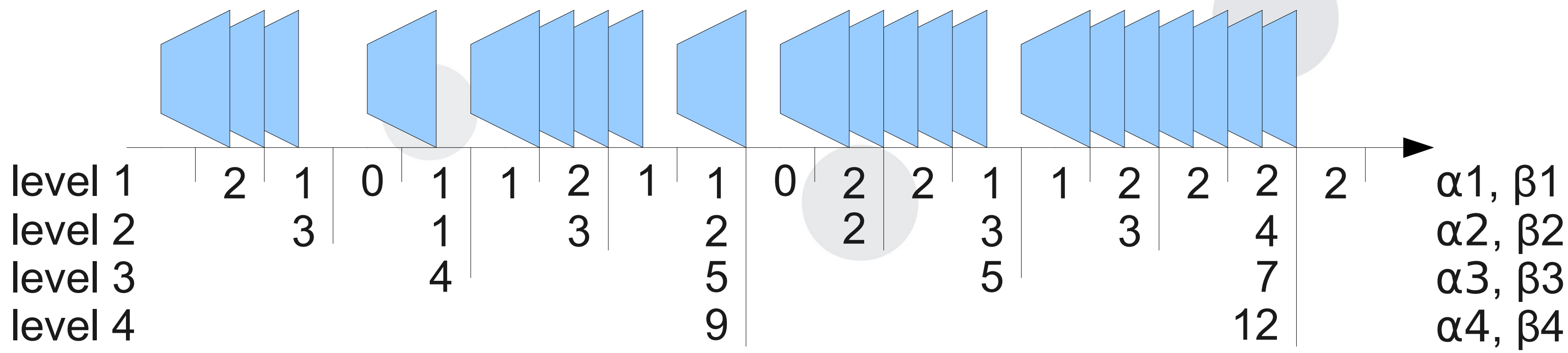
Random projections

- A fixed size time-window of captured traffic is split into sketches using a hash function.
- Selected packet attribute (policy) serves as hash key.
- Hash table size is fixed.



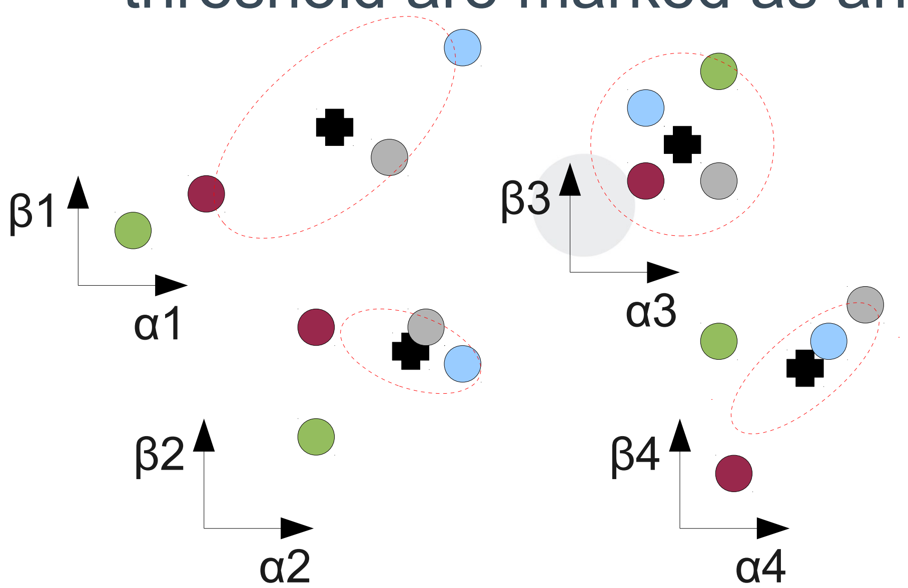
Aggregation, gamma distribution parameters

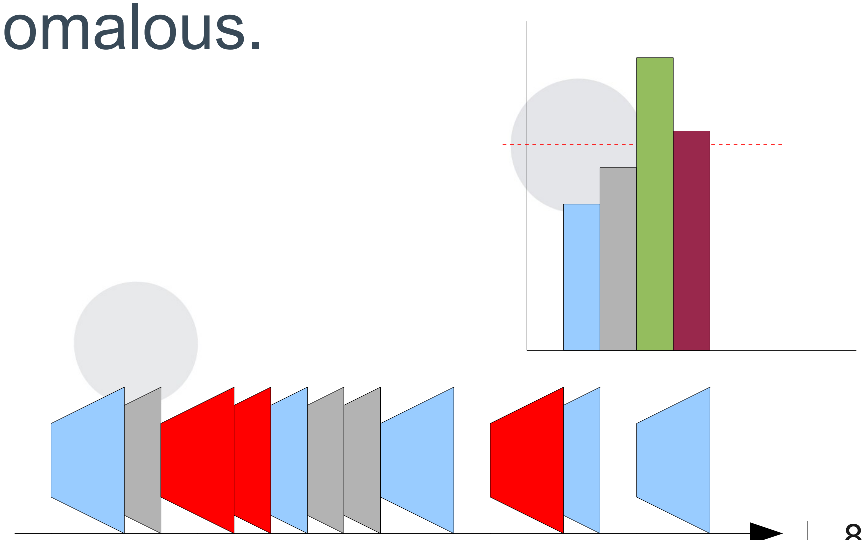
- The sketches are aggregated jointly over a collection of aggregation levels to form a series of packet counts which arrived during an aggregation period.
  - Aggregation levels transform the time-scale granularity.
- Data from the aggregated time series are modelled using Gamma distribution.
  - Shape (α) and scale (β) Gamma distribution parameters are computed for each aggregation level.



Reference values, identification of anomalous sketches

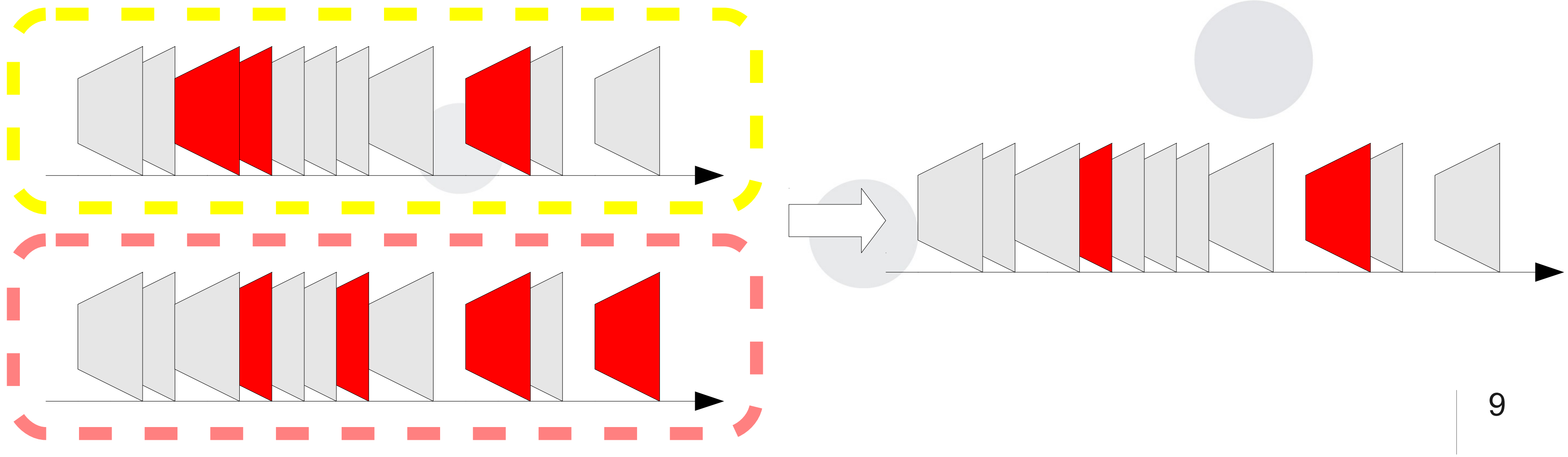
- For each aggregation level across all sketches standard sample mean and variance of the computed Gamma parameters are computed.
- For each sketch the average Mahalanobis distance to the 'centre of gravity' is computed.
- Sketches with their average distance exceeding a given threshold are marked as anomalous.





Anomaly identification

- All packet attributes (hash keys) contained in an anomalous sketch are considered suspicious.
- Using a different hash function provides a different mapping into sketches resulting in various anomalous sketches.
- A list of attributes corresponding to detected anomalies is obtained by combining the results for several hash functions and computing the intersection of anomalous sketches.



Modification for DNS

- The method was designed to analyse the whole TCP/IP traffic.
    - Works with TCP/IP connection identifiers (src/dst port/address).
- We extended it to meet DNS traffic specifics.
- Policies:
    - IP address policy
        - Based on original paper, uses the TCP/IP connection identifiers.
        - Supports IPv4 and IPv6.
        - Helps finding suspicious traffic sources.
    - Query name policy
        - First domain name of the query is extracted and used as hash key.
        - Helps finding suspicious traffic from legitimate sources.

The tool

- Standalone application is freely available at git://git.nic.cz/dns-anomaly/
- Command line parameters:
  - window size + detection interval
  - count of aggregation levels
    - Aggregation steps are power of 2 in seconds (i.e. 1,2,4,8,...).
  - analyse shape, scale or both
  - detection threshold
  - policy
  - hash function count
  - sketch count (hash table size)

Experiments

Tested on DITL 2011 data collected in April 2011 on .cz authoritative DNS servers.

parameter	value
time-window size	10 minutes
detection interval	10 minutes
hash function count	25
hash table size	32
aggregation levels	8
distance threshold	0.8

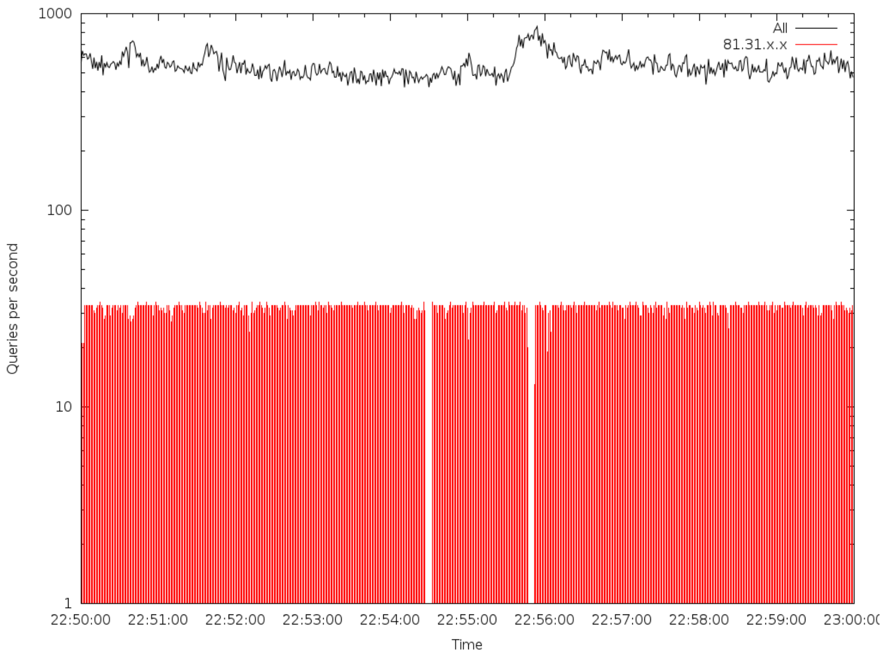
Using these settings the analyser is able to process 10 minutes of traffic (126MB) in 1.8 second on a E5400@2.70GH.

Results

Types of traffic labelled as anomalies:

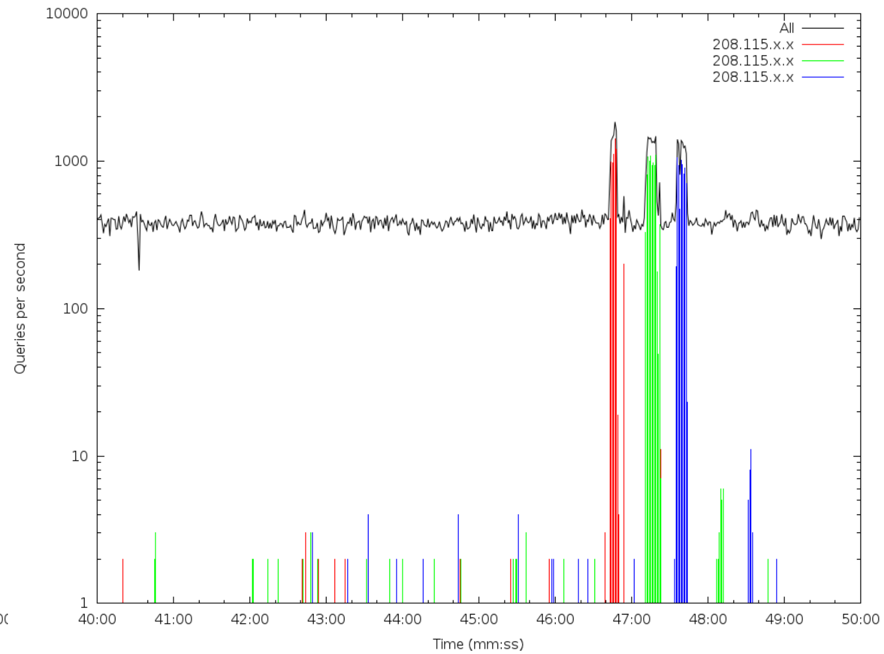
- Traffic form legitimate sources (exhibiting specific patterns)
  - large recursive resolvers, web crawlers
- Domain enumeration
  - Blind or dictionary based (gTLD domain, prefix and postfix alteration for given words – e.g. bank or various trademarks)
  - With the knowledge of the content (little or no NXDOMAIN replies)
- Suspicious
  - Traffic generated by broken resolvers or testing scripts.
    - e.g. bursts of queries for the same name from single host
  - Repeated queries due to short TTL

Generic traffic



Recursive resolver

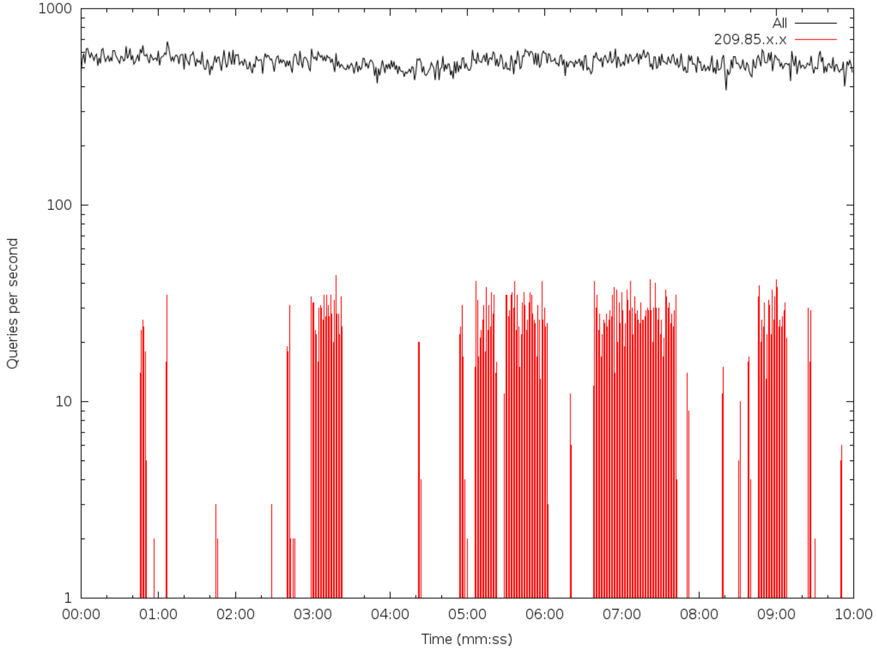
srcIP policy
Originates at webhosting/ISP. The pattern is very regular with a period of approximately 12 seconds.

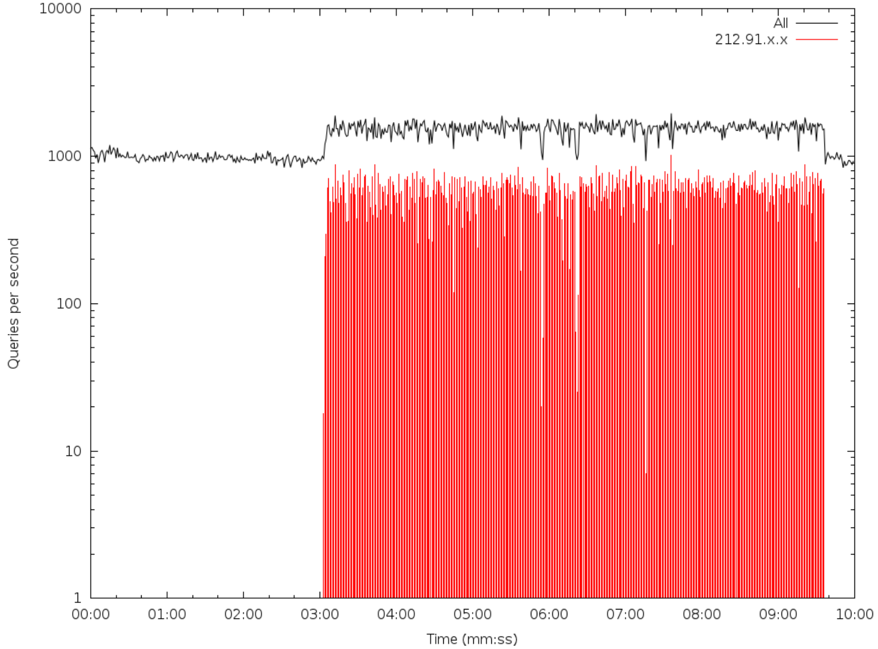


Web crawler farm

srcIP policy
Possibly web crawlers. They generate lots of queries whenever they encounter sites with many references.

Domain enumeration





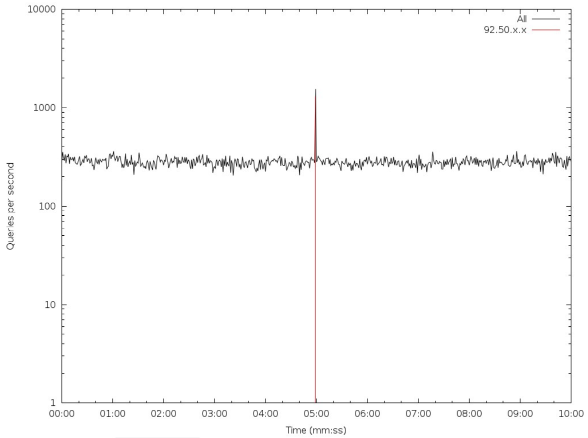
Blind domain enumeration

srcIP policy
When analysing the DNS queries a pattern emerged – prefixes and postfixes variation using well-known trademarks.

Known domain enumeration

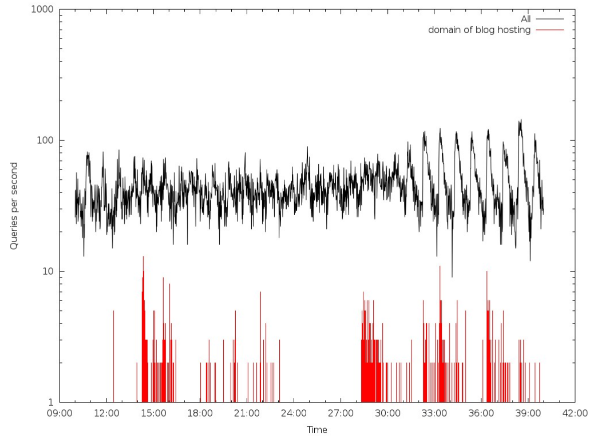
srcIP policy
The source must have a very good knowledge about the content of the domain. Very few NXDOMAIN replies are generated.

Other suspicious



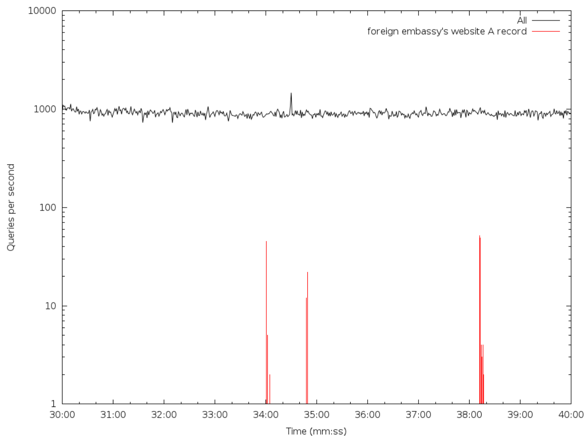
Broken resolver

srcIP policy
Hundreds of queries for a single record are generated in less than two seconds.



Possible spam attack

qname policy
Multiple hosts are querying same MX record.



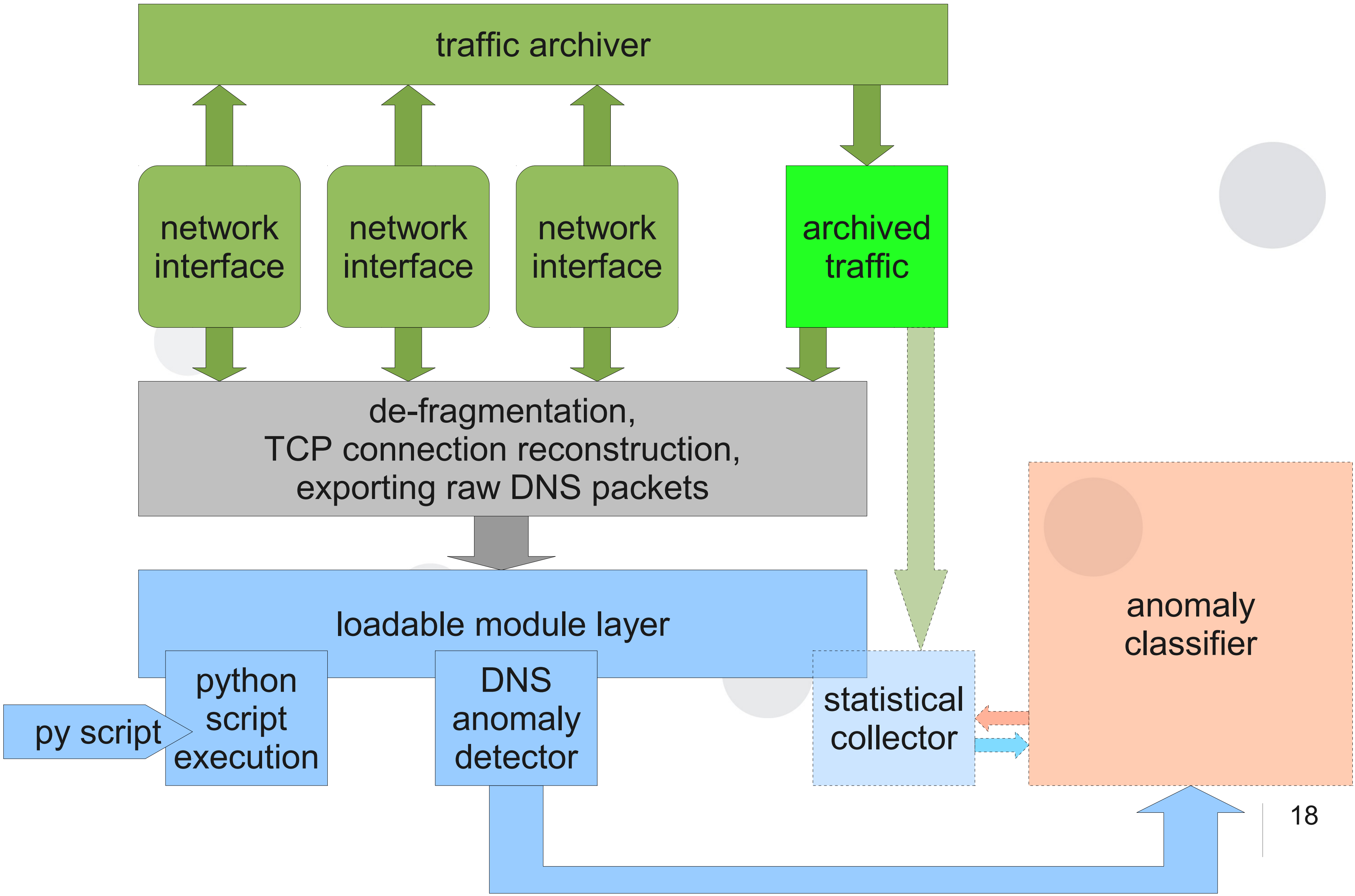
???

qname policy
Multiple hosts evenly distributed around the world are generating bursts of queries for the same record.
The pattern is visible throughout the entire tested period - always as characteristic spikes.

Anomaly detection conclusion

- The tool is able to pinpoint low- and high-volume anomalies.
- Two policies implemented with different effect:
  - IP policy serves best for domain enumeration detection.
  - Query name policy divulges domain-related events.
    - e.g. presence of short TTL domains (fast flux)
- The classification of the anomalies is currently left to be done manually.
  - Future work: automate this process.

A system under development



DNS anomaly classifier

- used to classify output from the DNS anomaly detector
- random forest classifier is being used because
  - highly accurate classifier
  - efficient run on large data sets
  - gives an estimate of what variables are important in the classification
  - soft decisions
- classifiers can be saved for future use
  - Classifiers are sensitive to the source of classified data. Different sources need to have separate classifiers.

Input variables – statistical data

- 62 variables serve for classification of anomalous data
- relative and absolute measures regarding the volume of
  - query types, return codes, ttl
- penetration of various selected identifiers
  - BGP prefixes, ASN, IP addresses, country of origin, query names
- also takes into account
  - query time, total traffic volume, server response time

Classifier performance and accuracy

- The training set contains approximately 1 million classified samples (six days of anomalous traffic).
- Training a classifier containing 200 trees each containing 15 nodes lasts about 2 hours.
- Classifying of such a large data set using the trained forest lasts about 10 minutes.

- Classifying out-of-bag data yields 80% accuracy.
  - The accuracy was determined by comparing classifier results with hand classified data.

Work in progress

- statistical collector module
  - replacement for DSC?
- performance improvements in the random forest classifier
  - increasing accuracy, increasing speed, reducing memory consumption
- communication protocol
  - change collector settings, module reloading, attaching/detaching network devices

The End

Thank you for your attention.

Questions?