Parameters of Traffic Anomaly Detection

What's the impact?

And how do we measure this?

Haakon Larsen, Augustin Soule,
Jennifer Rexford, Christophe Diot

Outline

- Background: traffic anomaly detectors
  - Motivation and approach
- Motivation
  - Impact of parameters on detectability
- Large parameter space to analyze
- How to evaluate anomaly detectors
  - …and associated parameters
- (Very) preliminary results

Anomalies are a problem

- Networks are large
  - thousands of nodes
- Networks are distributed
  - physically and logically
- Networks are heterogeneous
  - myriad of legacy/current technologies
- Networks carry lots of traffic
- Contain lots of different anomalies
  - configuration error, port scan, DDoS, spam, intrusion, phishing, worms, transient faults

We need automation

- Algorithm requirements:
  - efficiently process loads of data
  - to find disparate traffic anomalies
  - in different network settings
- Statistical-analysis techniques
  - MA, PCA, Kalman, Wavelets

High-level approach

- Consider the traffic trace as a timeseries of traffic measurements
  - e.g. timeseries of packet counts
- “Type” of the timeseries has varied
  - packets, bytes, flows, entropy values
  - or combinations of the above
- Good to have a network-wide view
  - ingress routers, input links, OD flows

Promising early results, but..

- “We detected lots of anomalies!”
  - DOS, flash, port/net scan, outage…
- “…when we slapped algorithm X tuned Y on network Z”
  - PCA is most prominent algorithm
- We want to know what impact X, Y, and Z had on the results

Axes of parameter space

- Network's data collection
  - Address anonymization
  - Temporal aggregation
  - Sampling
- Data processing
  - Aggregation formalism
    - ingress routers, OD flows
  - Detection Algorithm
    - Threshold, other parameters

	Abilene	Géant
Anonymization	11 bits	0 bits
Temporal Aggregation	5 minutes	15 minutes
Sampling	1/100	1/1000

Our parameter space

	Abilene	Geant
Algorithm	PCA, Kalman	PCA, Kalman
Threshold	PCA: 90%, 95%, 97.5%, 99%, 99.5%, 99.9% Kalman: 6, 7, 8, 9, 10, 11	PCA: 90%, 95%, 97.5%, 99%, 99.5%, 99.9% Kalman: 6, 7, 8, 9, 10, 11
Principal components in PCA	2, 3, 4, 5, 6, 7, 8, 9, 10, 25, 50	2, 3, 4, 5, 6, 7, 8, 9, 10, 25, 50
Traffic aggregation formalism	Ingress routers, OD flows, input links	PCA: ingress routers, input links Kalman: ingress routers, OD flows, input links
IP address anonymization	11 bits through 31 bits	0 bits through 31 bits
Temporal aggregation	5min, 15min, 30min, 45min, 60min	15min, 30min, 45min, 60min
Sampling (in addition to network's)	None, 1/10, 1/100, 1/1000	None, 1/10, 1/100

Oh-oh! What's the damage?

- **120 thousand** points in parameter space
- For each time-step, DB contains:
  - (time, ingress, egress, #packets, #bytes, #flows, entropy values)
- DB contains over **228 million** data-points
- **15 million** anomaly-labels
- How can we evaluate the impact of sweeping the parameter space, with so much data?

Evaluating Traffic Anomaly Detectors

- Manually labeled traces
  - Human expertise is brought to bear
  - Scales poorly on its own
- Algorithmically injected anomalies
  - Gives labeled trace
  - Thus far very crude algorithms
- Unlabeled data: compare against others
  - e.g. efficient approximation algorithm versus brute-force

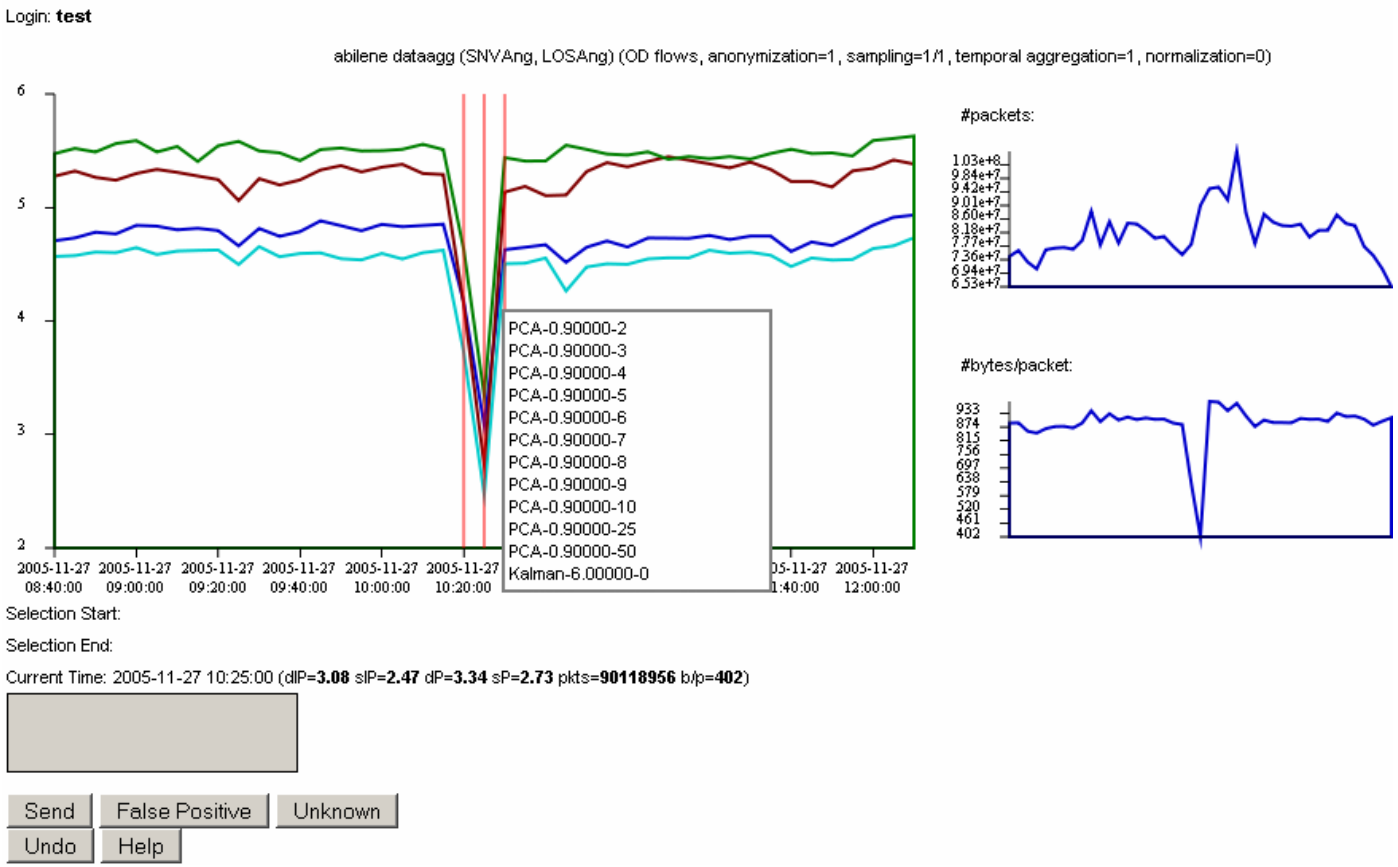
Our approach: manual labeling++

- Manually classify temporal *regions*
  - $[t_0, t_1]$ may contain many anomalies
- Manually label subset of parameter space
  - e.g. 0 bits, 8 bits, 16 bits anonymization
- Infer between points in parameter space
- Akin to Pr(X | Y & Z)
  - X = “true anomaly at time t with 8 bits anon”
  - Y = “true anomaly at time t with 0 bits anon”
  - Z = “true anomaly at time t with 16 bits anon”

Our tool: WebClass

- Easy to use, web-based tool to classify regions of a timeseries
  - Anomalous, false positive
  - Additional optional text field
- Support for concurrent usage
- Support for large parameter space
  - Ability to limit it for returned anomalies

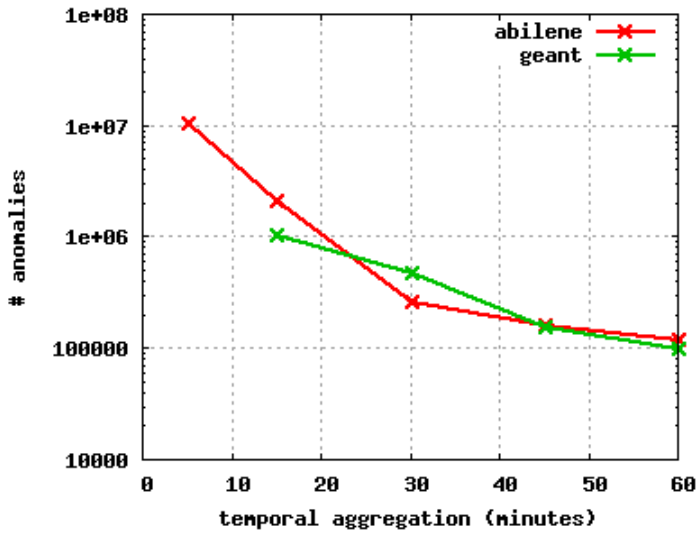
WebClass in action

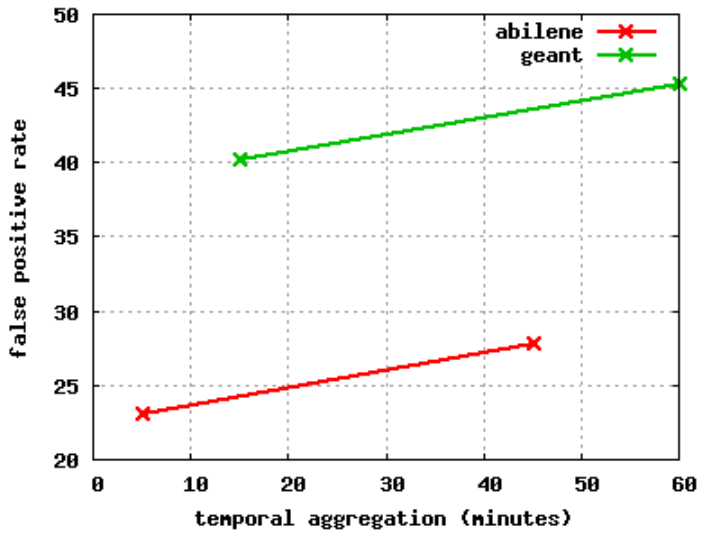


Longer term

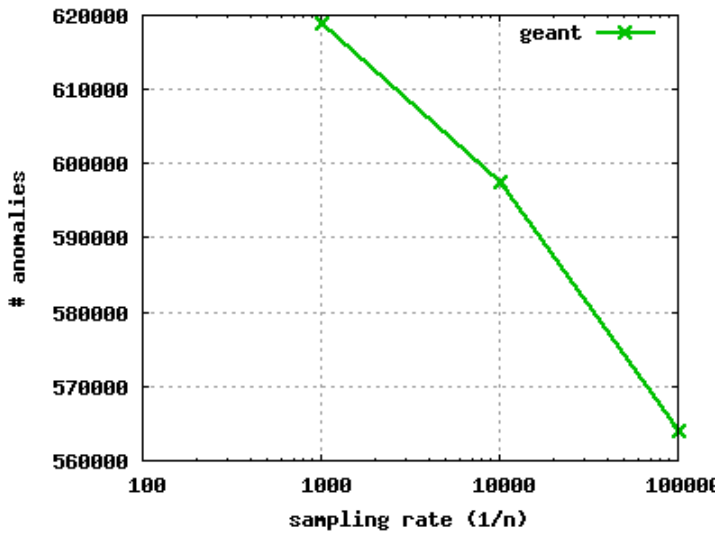
- Share WebClass with community
- Conditionally share resulting data with community
  - If they help us classify anomalies!
- Release test-suite of labeled traces
  - Sweeping the parameter space
  - Allows for comparison of detectors
- And come up with a better name…

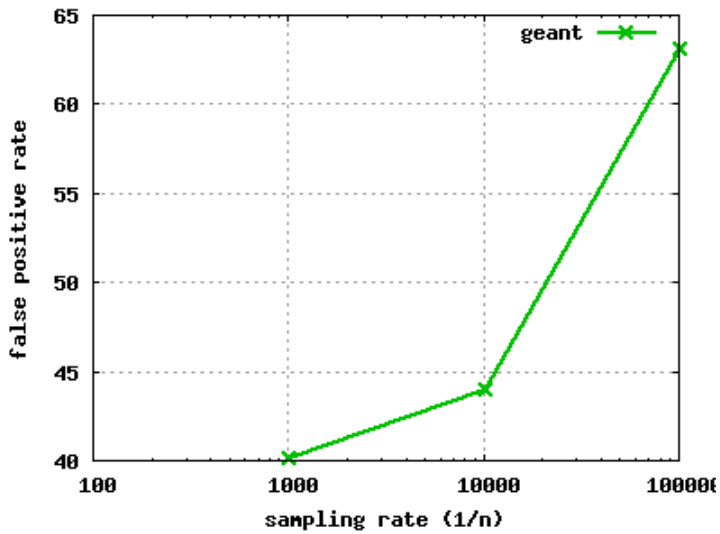
Impact of temporal aggregation



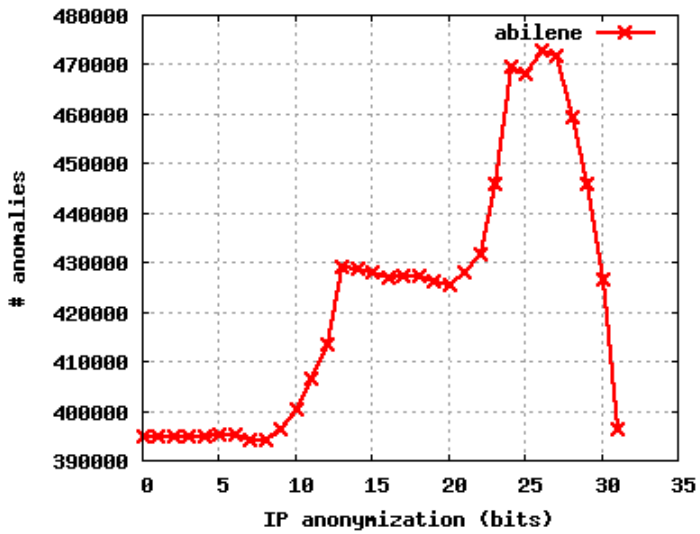


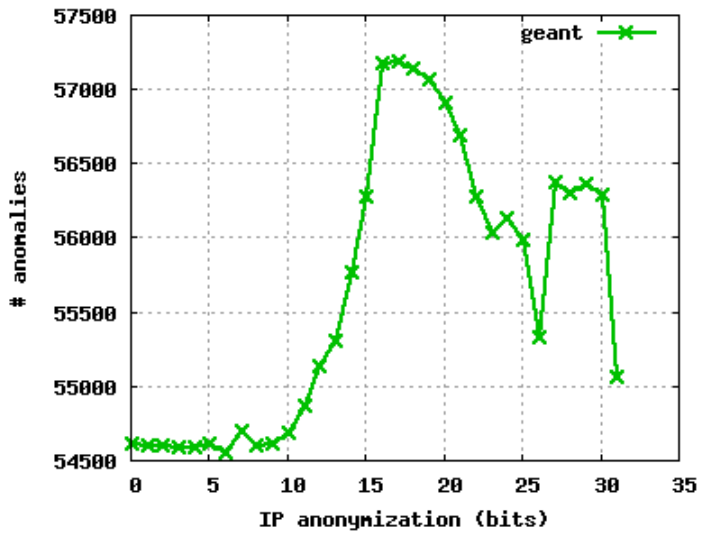
Impact of sampling





Impact of IP anonymization





Thank you!

Questions are welcome…

http://www.thlab.net/~hlarsen/webclass/
hlarsen@cs.princeton.edu
augustin.soule@thomson.net