Provenance Histogram Explorer

Junfeng & Michael CPSC 547 Information Visualization

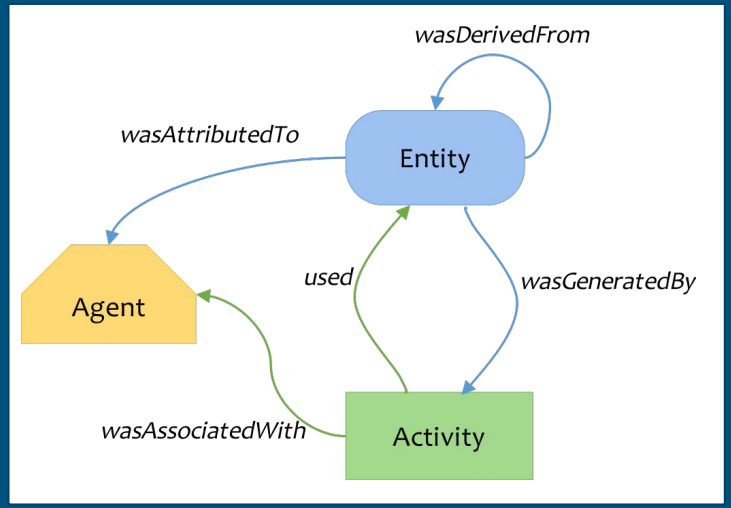
Provenance

- To understand the origin of data and its current state

Who drew that picture?
Who owns that picture?

Provenance

- Provenance in Computer Science



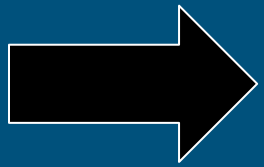
Who made that file?
Who owns that file?

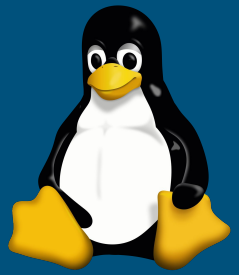
* World Wide Web Consortium PROV Model for web provenance @ 2013

Provenance

Intrusion Detection, Digital Forensic

What Agent (Browser, Web Site) modified my data?

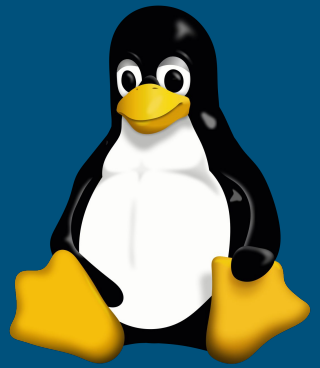




Operating System Provenance

An open research area

Why System Provenance?



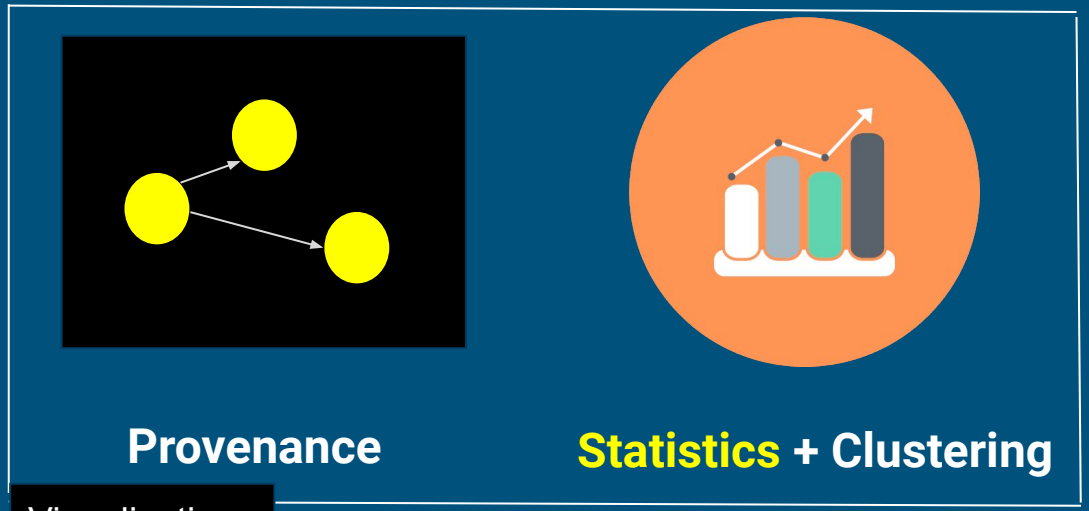
Challenge

‘Visualization’

- Can you picture Operating System operations?
- What operations matter? (e.g. Recurrent)

UNICORN

- An unpublished research on system provenance
- Our project visualizes components of UNICORN



Why UNICORN?

- It is easy for us to look at nice pictures and figures than to go through 0s and 1s or string of characters.

- Since automated intrusion detection system can make mistakes and it requires human labor to verify their decisions, it will take much less time for the human validator to visualize problematic part of the provenance graph.

Professor Margo Seltzer & Michael Han

Why UNICORN?

Log Examples

```
0 1 56888932933 56888932933 56888932933 2
```

```
20172839394839 -> 1
29399404094949 -> 3
...
```

Output

0 1

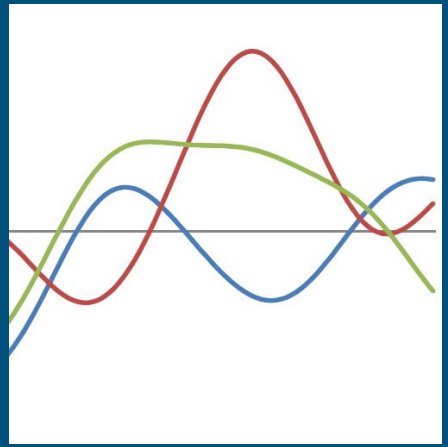
“Not Intuitive”

Task Abstraction

Finding and analysing anomalies that triggered UNICORN

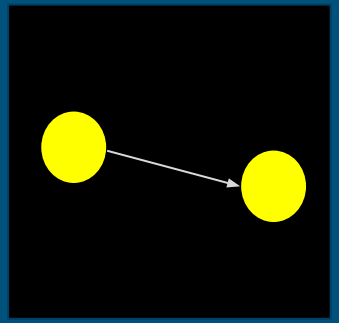
Ransomware vs. Compression

Exploring the `shape' of the histograms of different programs

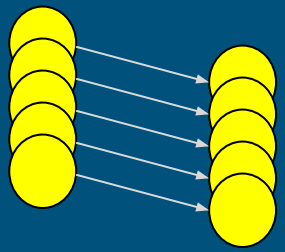


Identify the overall trends in time-series histograms

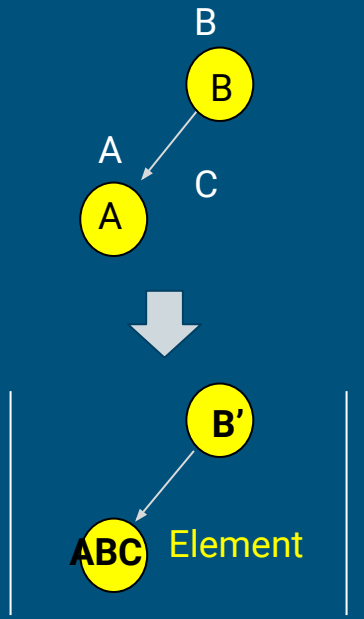
Data Abstraction (Simplified)



Source Node Id	0
Dest Node Id	1
Source Node type	A
Dest Node type	B
Edge type	C

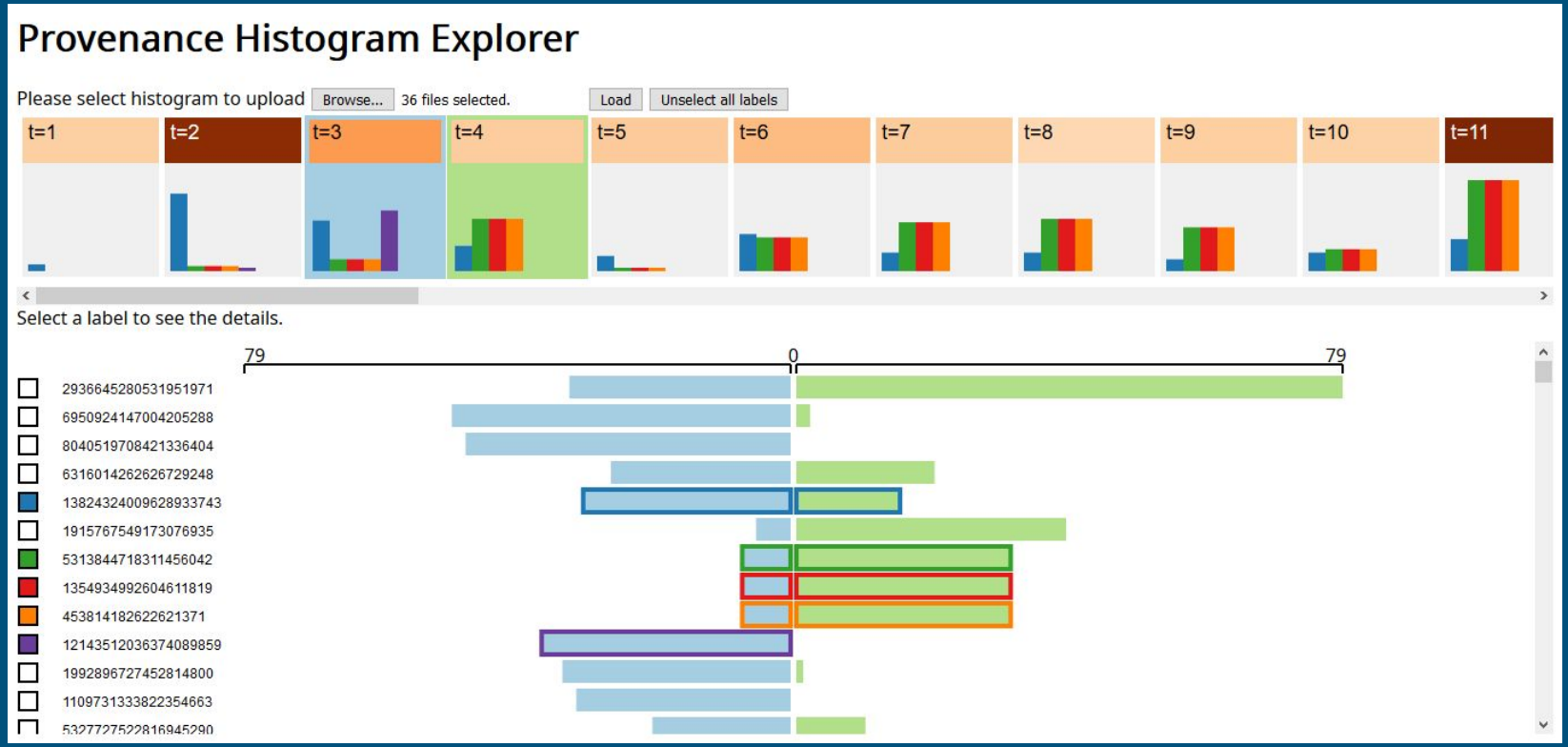


From a few hundreds of edges to billions of edges



Iterates every vertex for K times to find the sub graph structure

The Application



Implementation

- We built a data processing pipeline to capture and process Camflow data.
- The web application is built using **d3.js**.

Feedback

We interviewed Professor Margo Seltzer, demonstrated our application, and asked for her opinions

- She became interested in the data, and wished to see more.
- The visualization is ‘super interesting’ as the user can see the trajectory of the bins over time.
- The example (Firefox) dataset is not ideal - comparison of benign and malicious activities might be more interesting.

Future Works

- Change time frame definitions - put the histogram generated for 1,000 edges in each frame.
- Consider testing the application using different data sets - for example, using curl to visit 100 different websites.
- Produce visualization for the cumulative histogram.

- **Map the histogram back to the original provenance graph**.

Thank you!