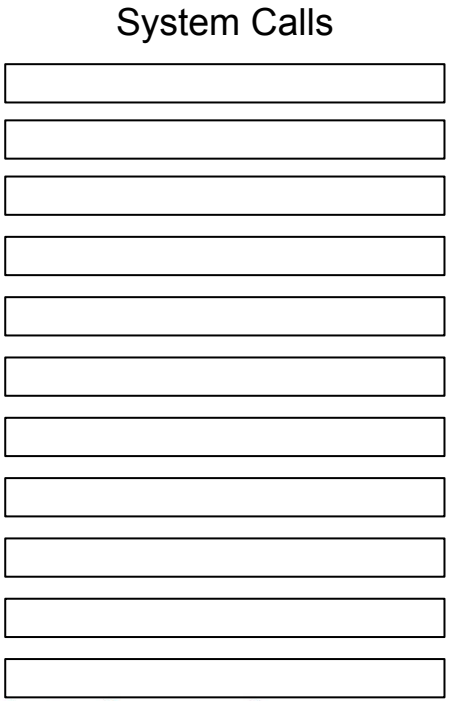
University of BRISTOL

Building a provenance-based IDS

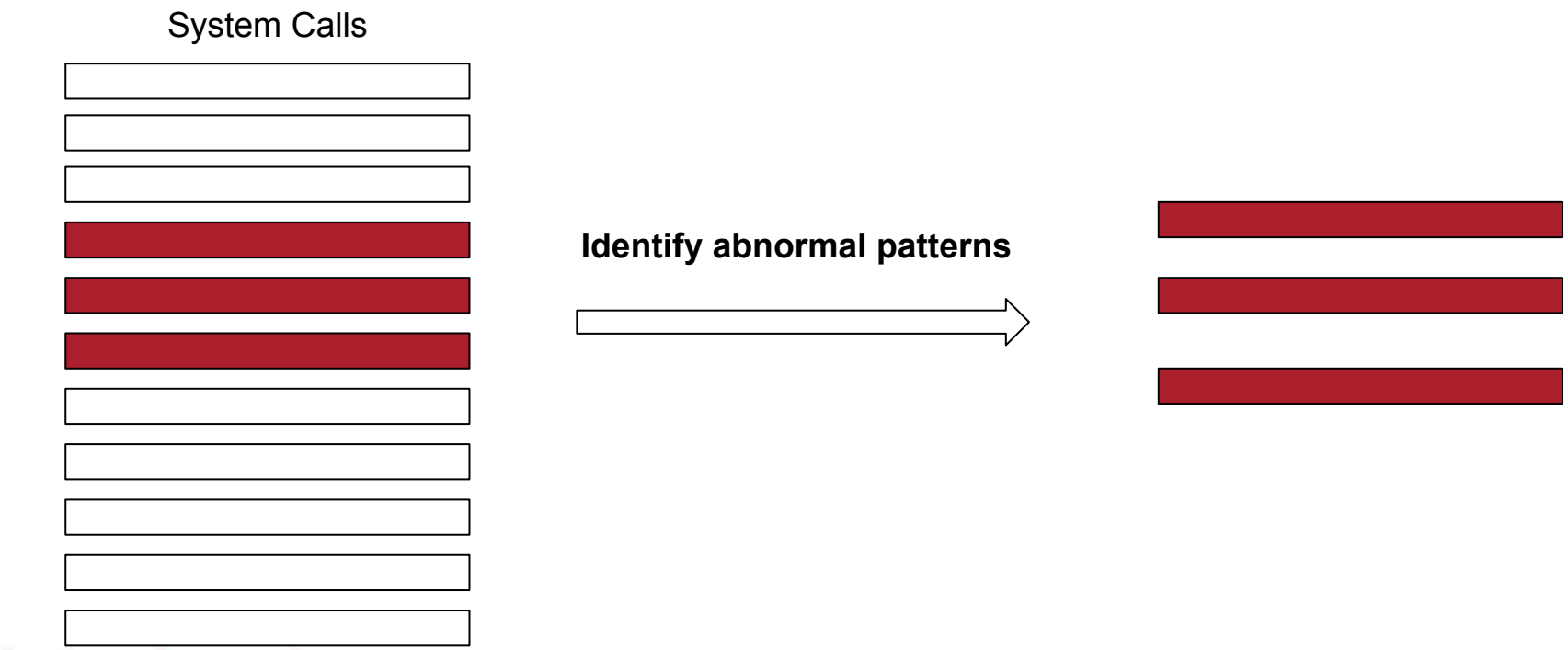
Thomas Pasquier
11/11/2019, Alan Turing Institute

bristol.ac.uk

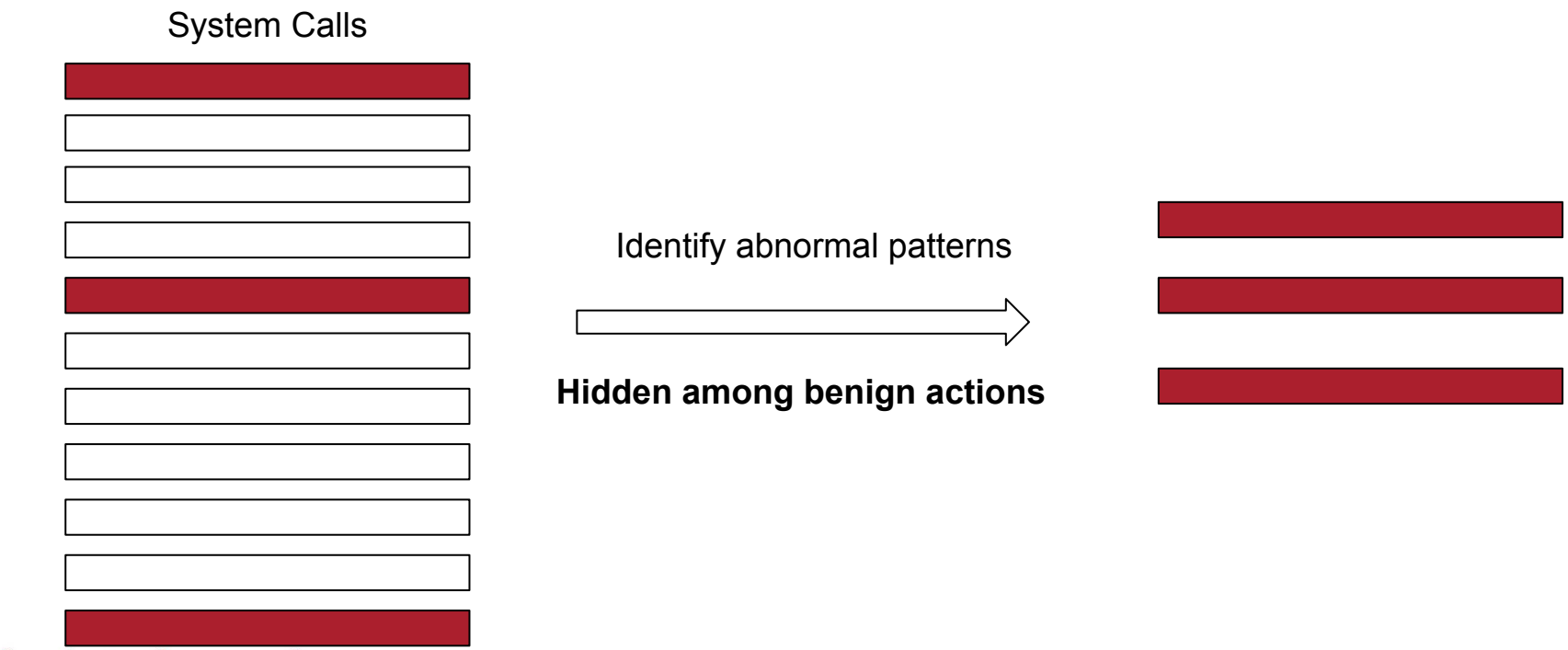
System call based intrusion detection



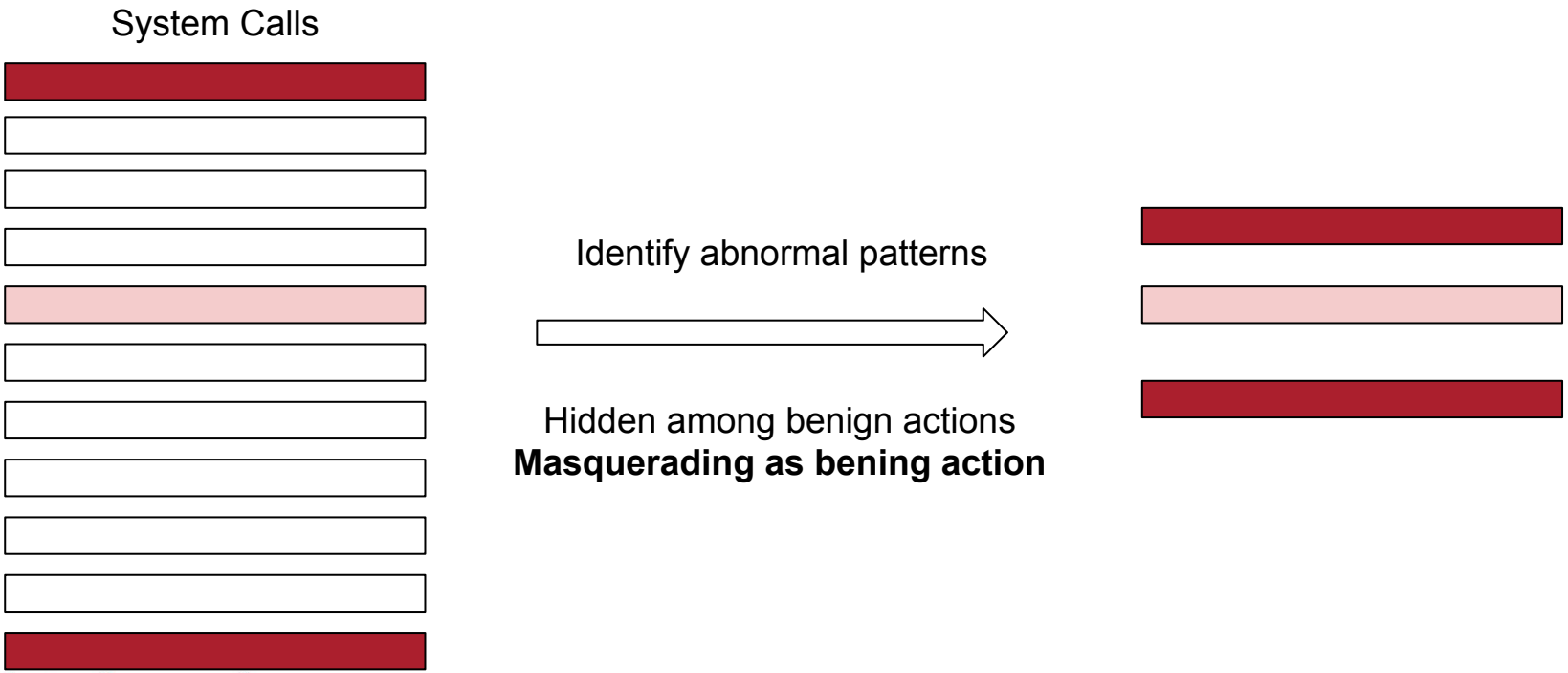
System call based intrusion detection



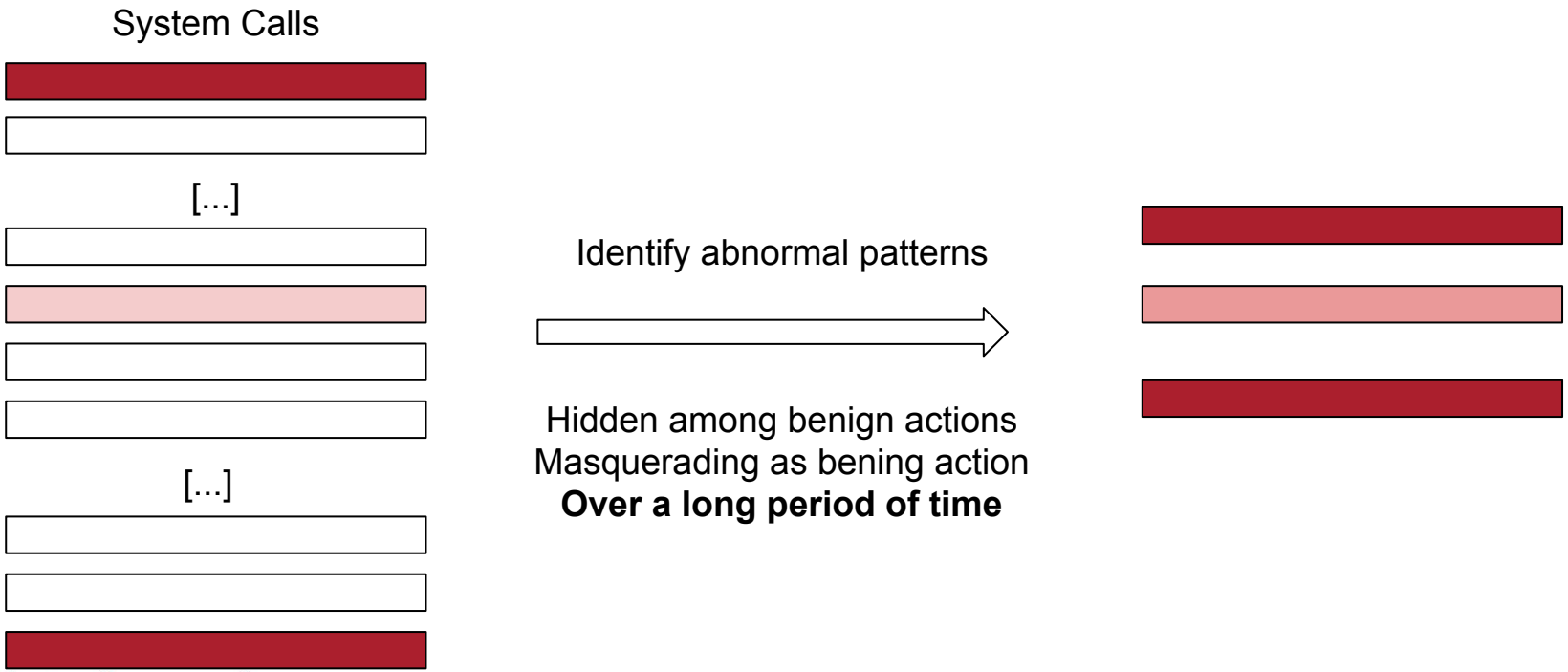
System call based intrusion detection



System call based intrusion detection



System call based intrusion detection

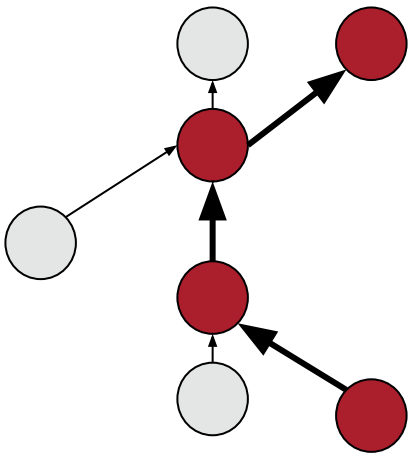


University of BRISTOL

What to do?

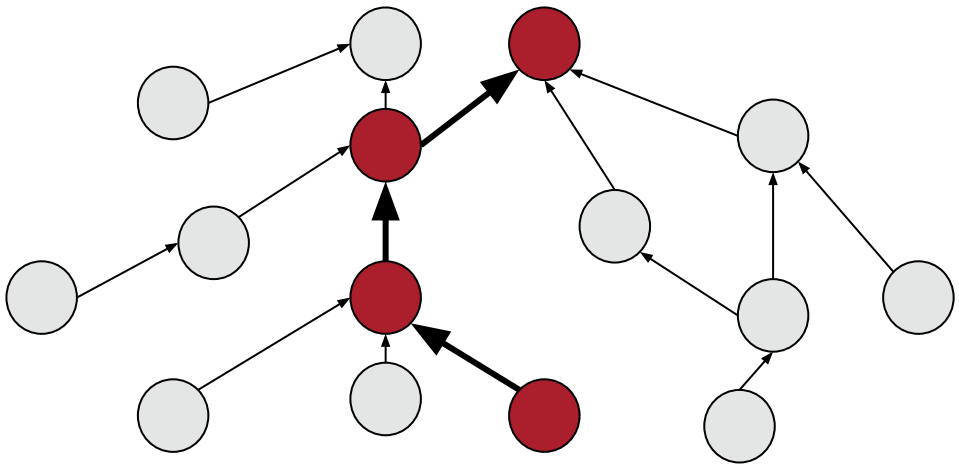
Provenance-based intrusion detection

- **Intuition**: provenance graph **exposes causality relationships** between events



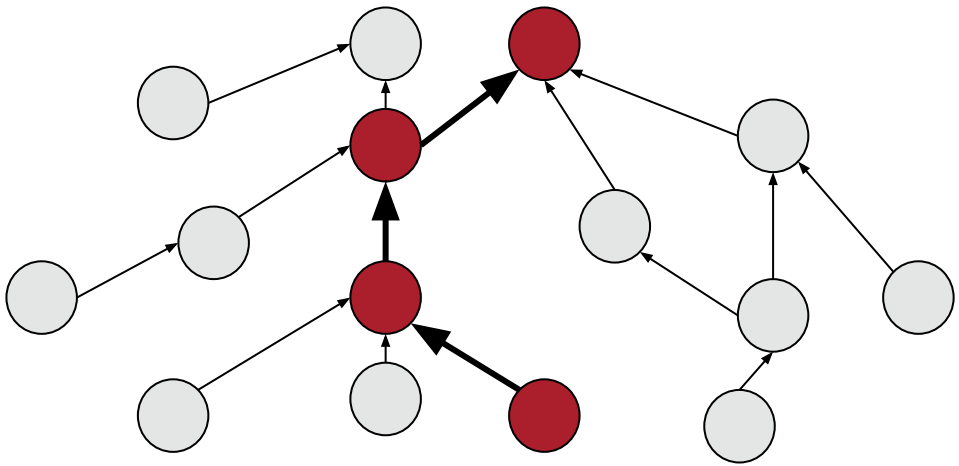
Provenance-based intrusion detection

- Intuition: provenance graph **exposes causality relationships** between events



Provenance-based intrusion detection

- Related system states are connected even across long period of time



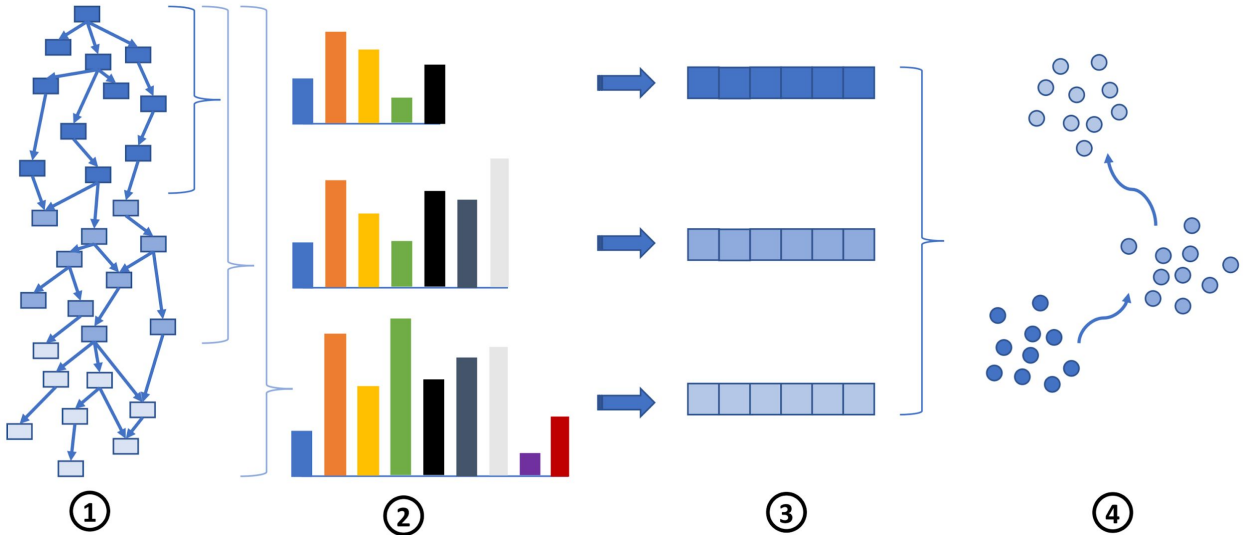
What is provenance in an operating system?

- Represent interactions between system objects
- Represented as a **directed acyclic graph**
- Information Flows
- **Relationship** between **kernel object states**
- History of a system execution

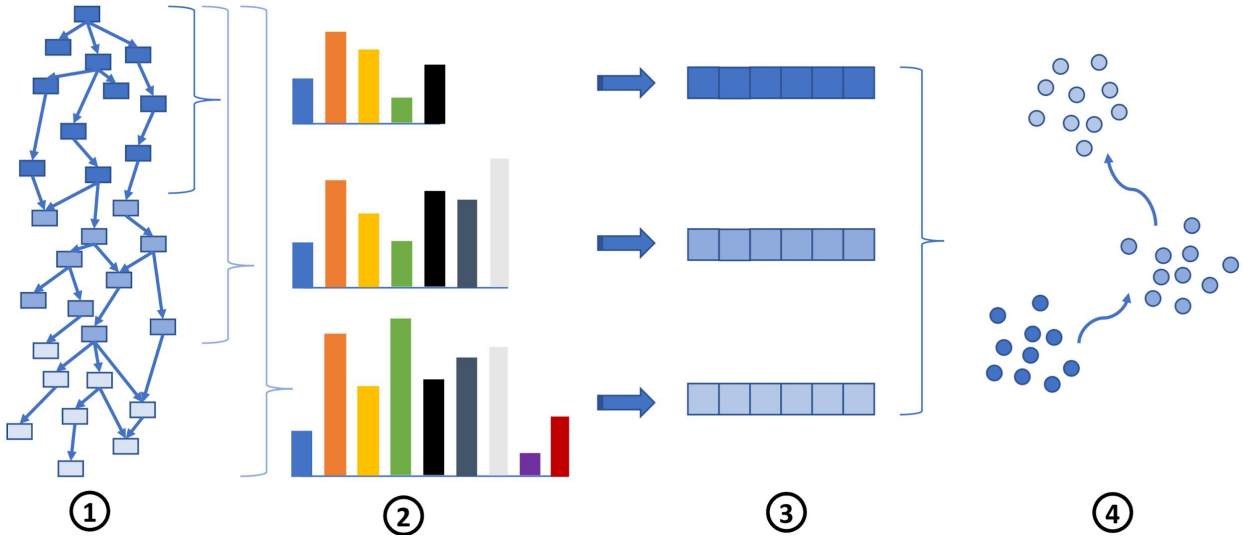
Provenance-based Intrusion Detection

- We target environment with minimal human intervention
  - Relatively well defined behaviour
  - In particular CI/CD pipeline
- Build a model of system behaviour (unsupervised, batch training)
  - in a controlled environment
  - from a representative workload
- Detect deviation from the model
- Several approaches being explored…

Detecting intrusion (an example)

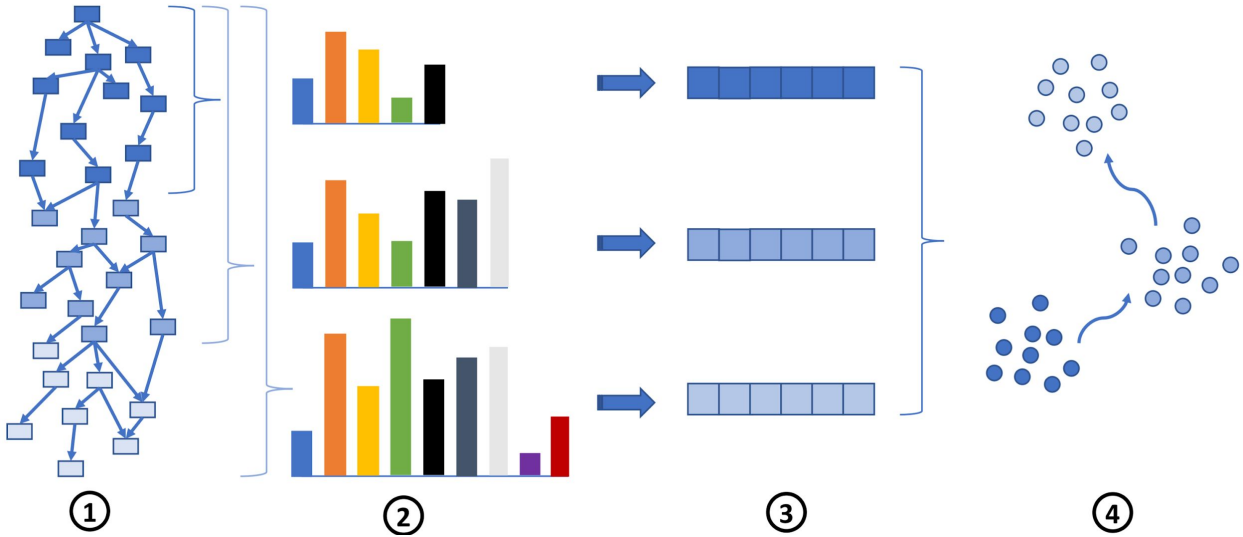


Detecting intrusion (an example)



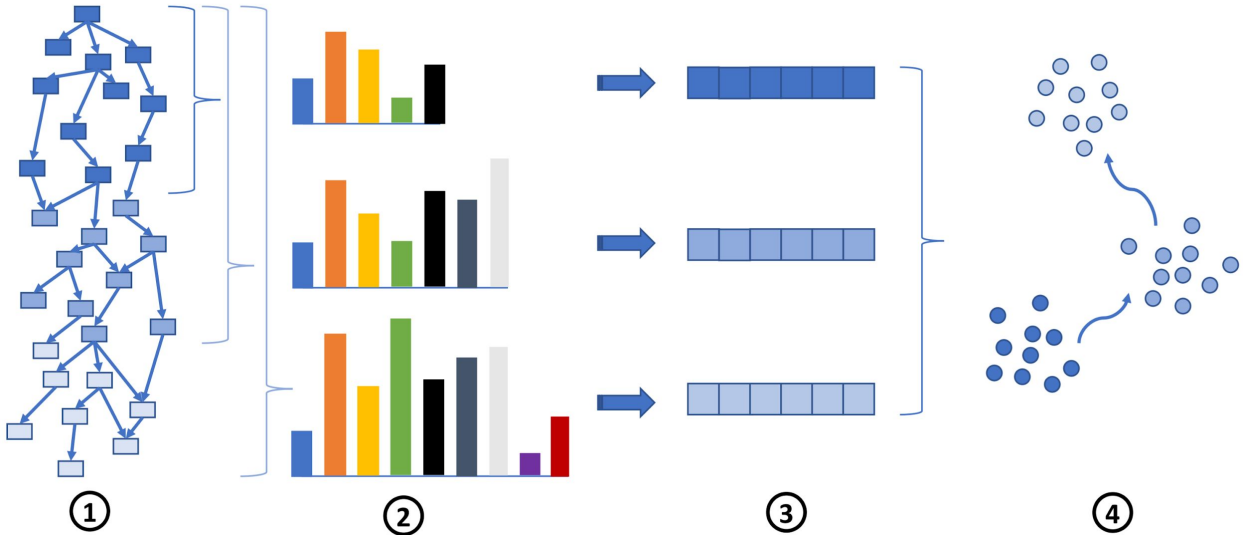
1) Graph streamed in, converted to histogram, labelled using struct2vec

Detecting intrusion (an example)



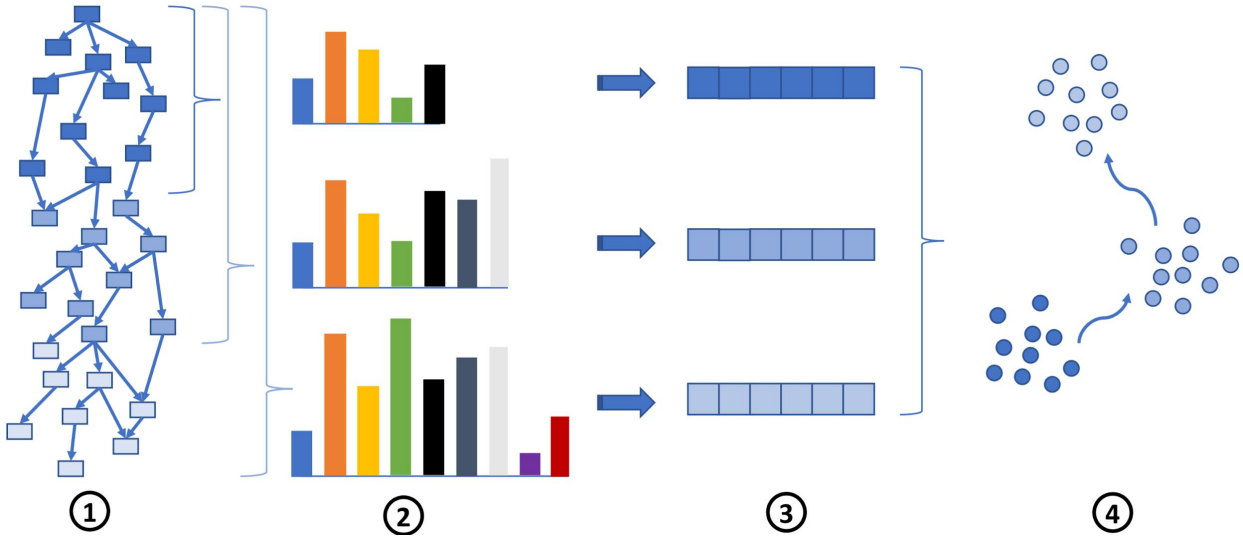
2) At regular interval, histogram converted to a fixed size vector using locality-sensitive hashing

Detecting intrusion (an example)



3) Feature vectors are clustered

Detecting intrusion (an example)



4) Cluster forms “meta-state”, transitions are modelled

In deployment, anomaly detected via clustering and “meta-state” model

Do provenance-based IDS work?

Experiment	Precision	Recall	Accuracy	F-Score
StreamSpot (baseline)	0.74	N/A	0.66	N/A
$R = 1$	0.51	1.0	0.60	0.68
$R = 3$	0.98	0.93	0.96	0.94

TABLE II: Comparison to StreamSpot on the StreamSpot dataset. We estimate StreamSpot's average accuracy and precision from the figure included in the paper [85], which does not report exact values. They did not report recall or F-score.

Experiment	Precision	Recall	Accuracy	F-Score
DARPA CADETS	0.98	1.0	0.99	0.99
DARPA ClearScope	0.98	1.0	0.98	0.99
DARPA THEIA	1.0	1.0	1.0	1.0

TABLE V: Experimental results of the DARPA datasets.

How do we build datasets?

Experiment	Precision	Recall	Accuracy	F-Score
StreamSpot (baseline)	0.74	N/A	0.66	N/A
$R = 1$	0.51	1.0	0.60	0.68
$R = 3$	0.98	0.93	0.96	0.94

TABLE II: Comparison to StreamSpot on the StreamSpot dataset. We estimate StreamSpot's average accuracy and precision from the figure included in the paper [85], which does not report exact values. They did not report recall or F-score.

Experiment	Precision	Recall	Accuracy	F-Score
DARPA CADETS	0.98	1.0	0.99	0.99
DARPA ClearScope	0.98	1.0	0.98	0.99
DARPA THEIA	1.0	1.0	1.0	1.0

TABLE V: Experimental results of the DARPA datasets.

Experiment	Precision	Recall	Accuracy	F-Score
SC-1	0.85	0.96	0.90	0.90
SC-2	0.75	0.80	0.77	0.78

TABLE VII: Experimental results of the supply-chain APT attack scenarios.

Attack looks like "normal workload"

How do we evaluate provenance-based IDS?

We never got the algorithm to work with SPADE (auditd) data

Experiment	Precision	Recall	Accuracy	F-Score
StreamSpot (baseline)	0.74	N/A	0.66	N/A
$R = 1$	0.51	1.0	0.60	0.68
$R = 3$	0.98	0.93	0.96	0.94

TABLE II: Comparison to StreamSpot on the StreamSpot dataset. We estimate StreamSpot's average accuracy and precision from the figure included in the paper [85], which does not report exact values. They did not report recall or F-score.

Experiment	Precision	Recall	Accuracy	F-Score
DARPA CADETS	0.98	1.0	0.99	0.99
DARPA ClearScope	0.98	1.0	0.98	0.99
DARPA THEIA	1.0	1.0	1.0	1.0

TABLE V: Experimental results of the DARPA datasets.

Experiment	Precision	Recall	Accuracy	F-Score
SC-1	0.85	0.96	0.90	0.90
SC-2	0.75	0.80	0.77	0.78

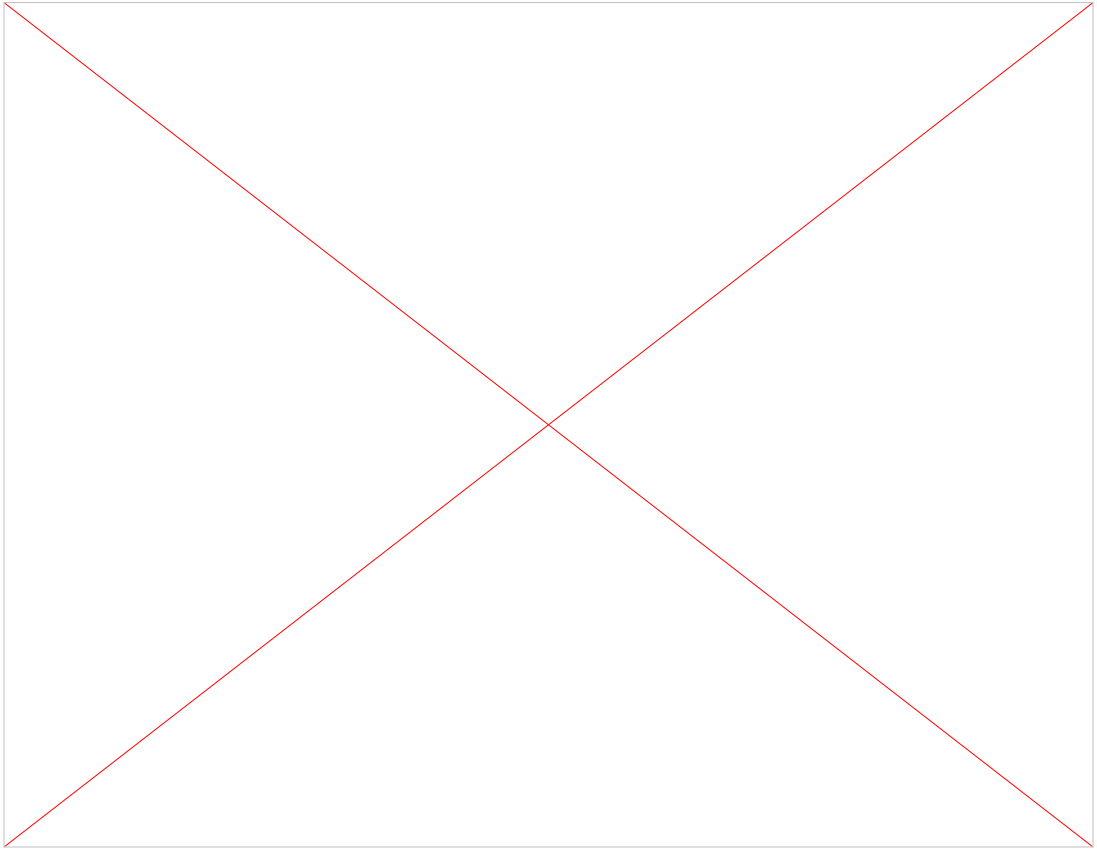
TABLE VII: Experimental results of the supply-chain APT attack scenarios.

What is provenance in an operating system?

- Represent interactions between system objects
- Represented as a **directed acyclic graph**
  - … or not?
- Information Flows
- **Relationship** between **kernel object states**
  - … or not?
- History of a system execution

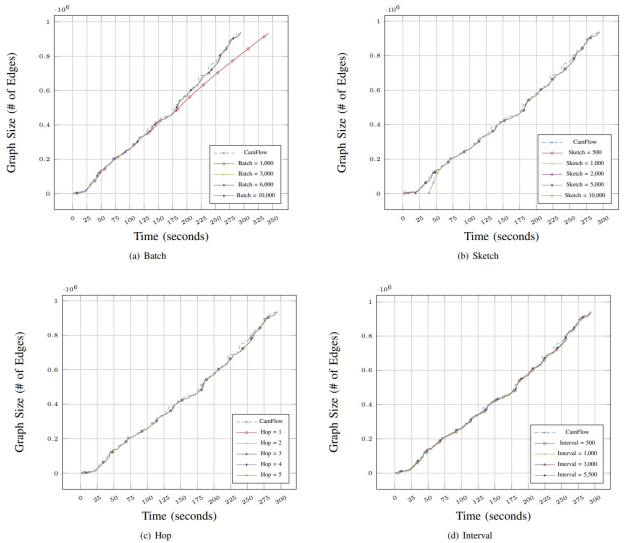
What is provenance in an operating system?

- Multiple capture levels
- Share similar syntax…
- … but different semantic



Can we analyse provenance at runtime?

- Performance?
  - Pasquier et al., **Runtime Analysis of Whole-System Provenance**, CCS 2018
  - Expect previous properties
  - Previous algorithm is practical

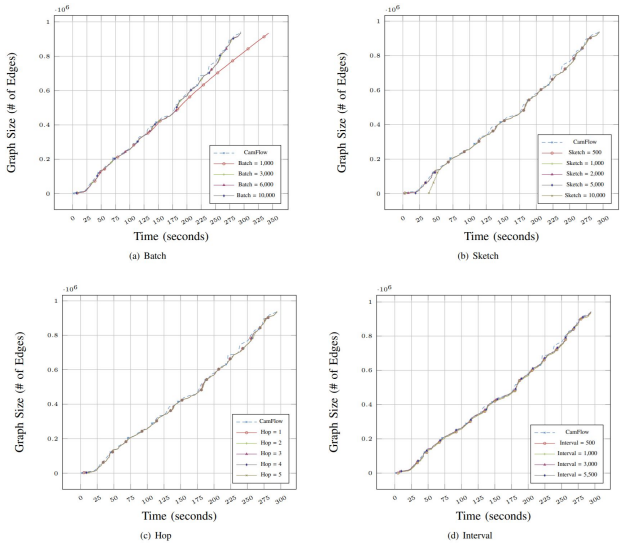


(a) Batch (b) Sketch (c) Hop (d) Interval

Fig. 4: Total number of processed edges over time (in seconds) in the SC-1 experimental workload with varying batch sizes (4(a)), sketch sizes (4(b)), hop counts (4(c)), and intervals of sketch generation (4(d)). Dashed blue line represents the speed of graph edges streamed into BABAR for analysis. Triangle maroon baseline has the same configurations as those used in our experiment and indicates the values of the controlled parameters (that remain constant) in each figure.

Can we analyse provenance at runtime?

- Performance?
  - Pasquier et al., **Runtime Analysis of Whole-System Provenance**, CCS 2018
  - Expect previous properties
  - Previous algorithm is practical
  - … only with CamFlow
  - … cycle are problematic
  - … ordering properties
  - … and more!



(a) Batch (b) Sketch (c) Hop (d) Interval

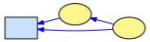
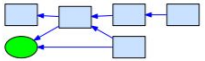
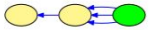
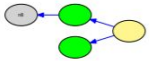
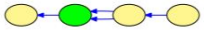
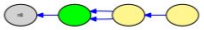
Fig. 4: Total number of processed edges over time (in seconds) in the SC-1 experimental workload with varying batch sizes (4(a)), sketch sizes (4(b)), hop counts (4(c)), and intervals of sketch generation (4(d)). Dashed blue line represents the speed of graph edges streamed into BABAR for analysis. Triangle maroon baseline has the same configurations as those used in our experiment and indicates the values of the controlled parameters (that remain constant) in each figure.

Can we be sure that capture is accurate and complete?

- Chan et al., **ProvMark: A Provenance Expressiveness Benchmarking System**, Middleware 2019
  - Dynamic provenance benchmark
  - Compared 3 systems (CamFlow, SPADE (auditd), OPUS)
- Pasquier et al., **Runtime Analysis of Whole-System Provenance**, CCS 2018
  - Static analysis of Linux Kernel
  - Generated model for CamFlow
  - Manual verification

Can we be sure that capture is accurate and complete?

- Chan et al., **ProvMark: A Provenance Expressiveness Benchmarking System**, Middleware 2019
  - Dynamic provenance benchmark
  - Compared 3 systems (CamFlow, SPADE (auditd), OPUS)

	open	read	write	dup	setuid	setresuid
SPADE				Empty		
OPUS		Empty	Empty			Empty
CamFlow				Empty		

The questions we ask ourselves

- What is provenance in an operating system?
- Do provenance-based IDS work?
- How do build datasets?
- How do evaluate provenance-based IDS?
- Can we evaluate provenance at runtime?
- Can we be sure that capture is accurate and complete?

University of BRISTOL

Thank you! Questions?

More info online: http://camflow.org