ProvDP: Differential Privacy for System Provenance Datasets

Kunal Mukherjee[1], Jonathan Yu, Partha De[1], and Dinil Mon Divakaran[2]

1 The University of Texas at Dallas, Richardson, TX, USA

2 Institute for Infocomm Research, A*STAR, Singapore

ACNS 2025 Munich, Germany, 26 June 2025

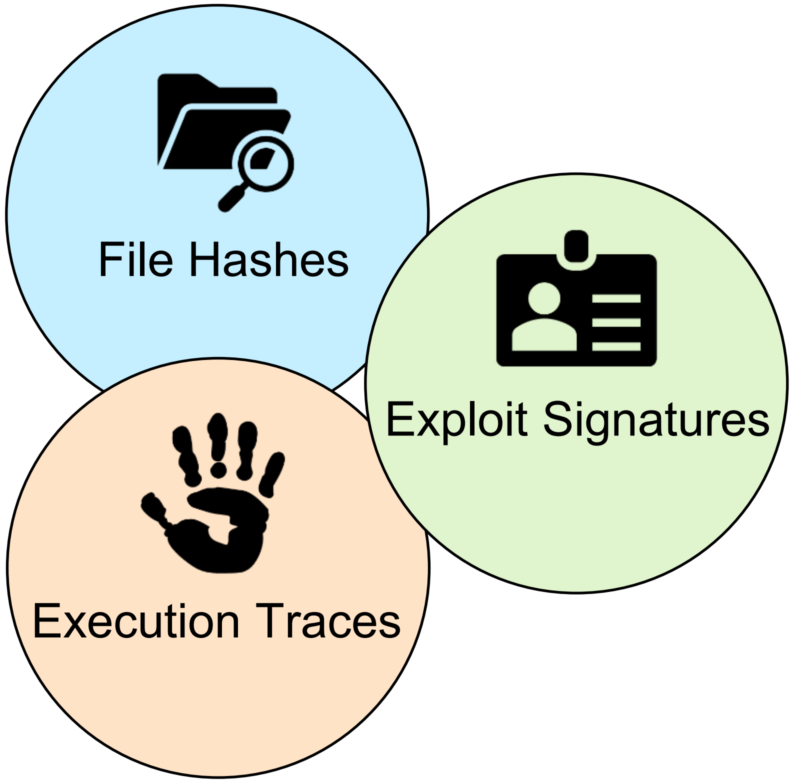
Agenda

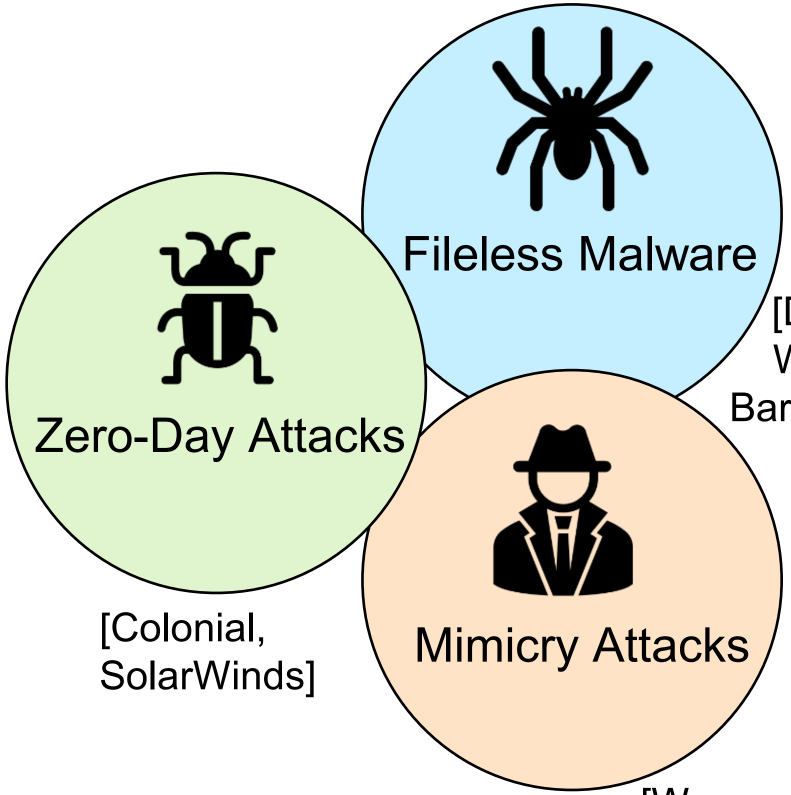
- **Background**
- Motivation
- ProvDP: Differential Privacy Framework
- Evaluation
- Discussion

Intrusion Detection Systems

Traditional Host *Intrusion Detection System (IDS)* detects **static artifacts**

Adversaries evade detection with **stealthy techniques**





[Dang '19, Wang '20, Barr-Smith '21]

[Colonial, SolarWinds]

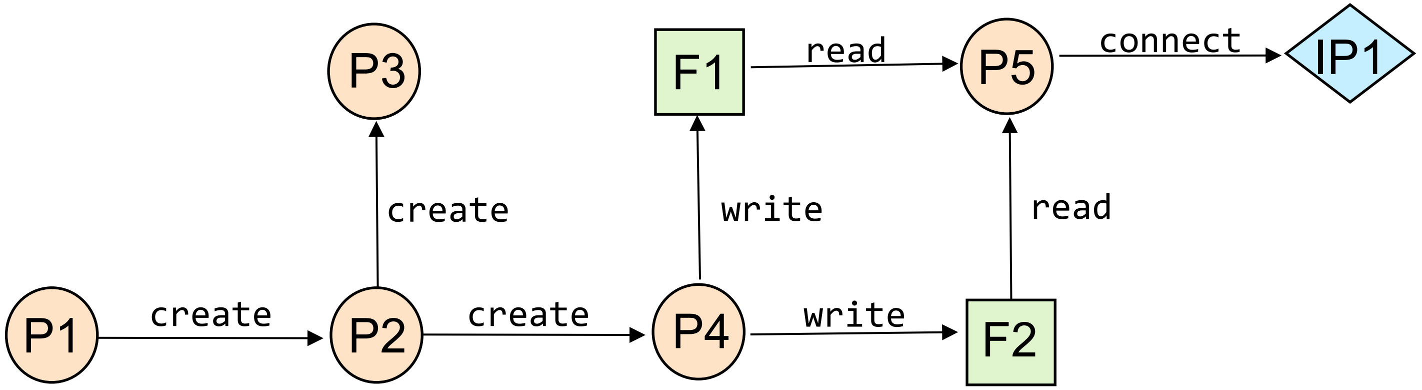
[Wagner & Soto '02, Tan & Maxion '03]

Traditional static IDS are insufficient

System Provenance

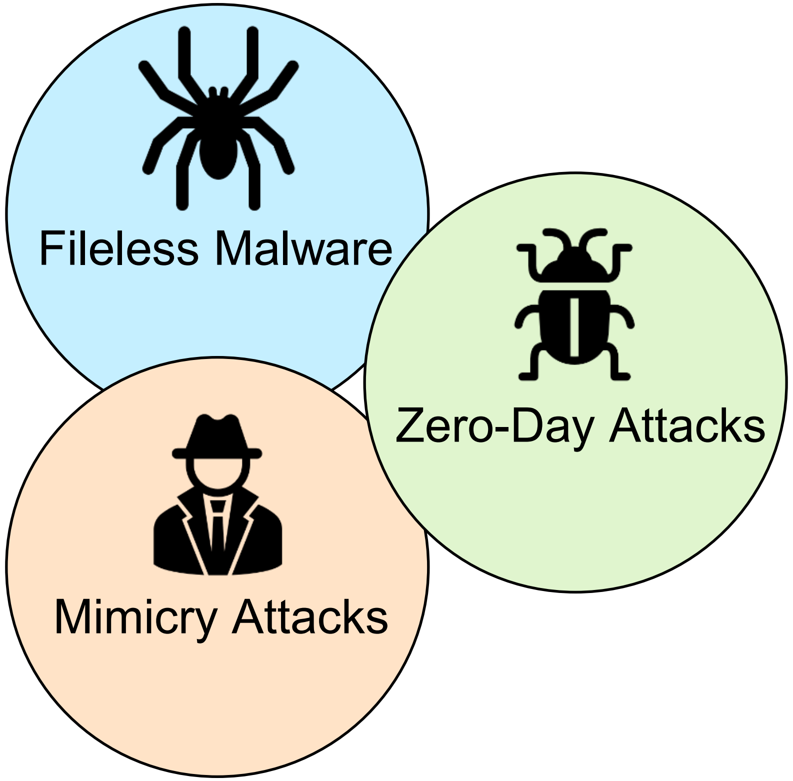
- **System Provenance** championed as a *host-based* dynamic defense
  - Influential works [Hassan '19, Wang '20, Han '21]

- System Provenance *causally* connects system resources
  - Captures *dynamic* control and data dependencies

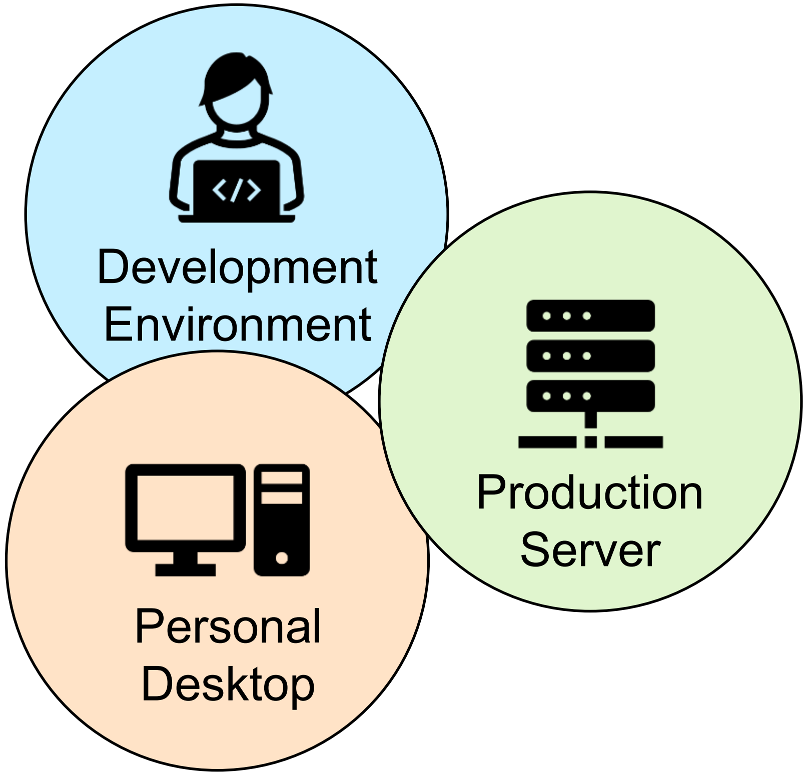


Provenance-based IDS

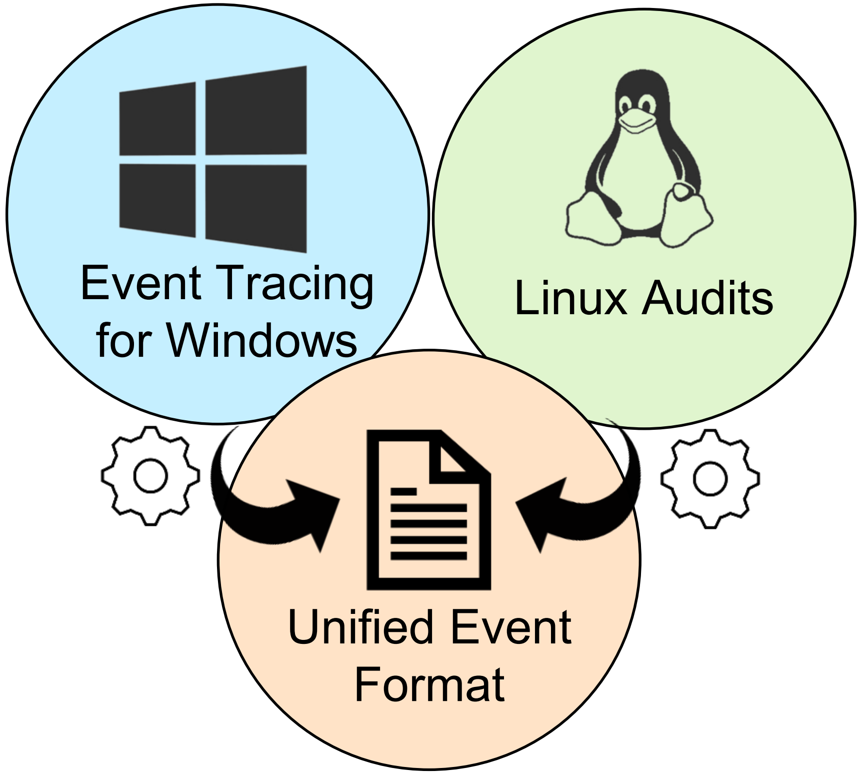
Provenance captures **runtime behaviors**



ML models are **fine-tuned** for different environments



Event collection frameworks provide **platform independence**



Provenance Graph Definition

Nodes

Process

File

Network

Example metadata:
- process: pid, cmd
- file: path, permissions
- network: ip/port

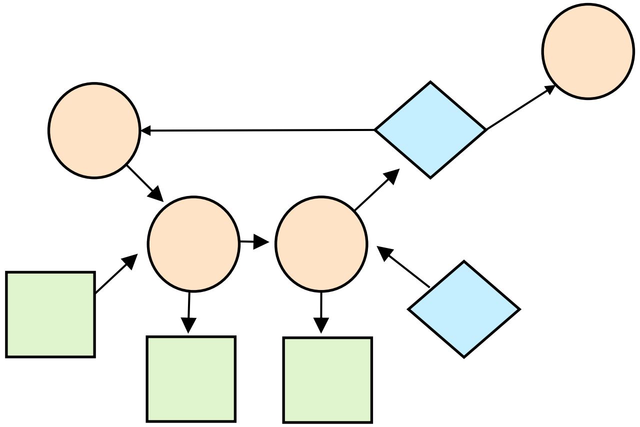
Edges

To

From

	Process	File	Network
Process	Create Kill	Write	Write
File	Read	Illegal	Illegal
Network	Read	Illegal	Illegal

Example metadata:
- timestamp
- file/network: bytes written/read

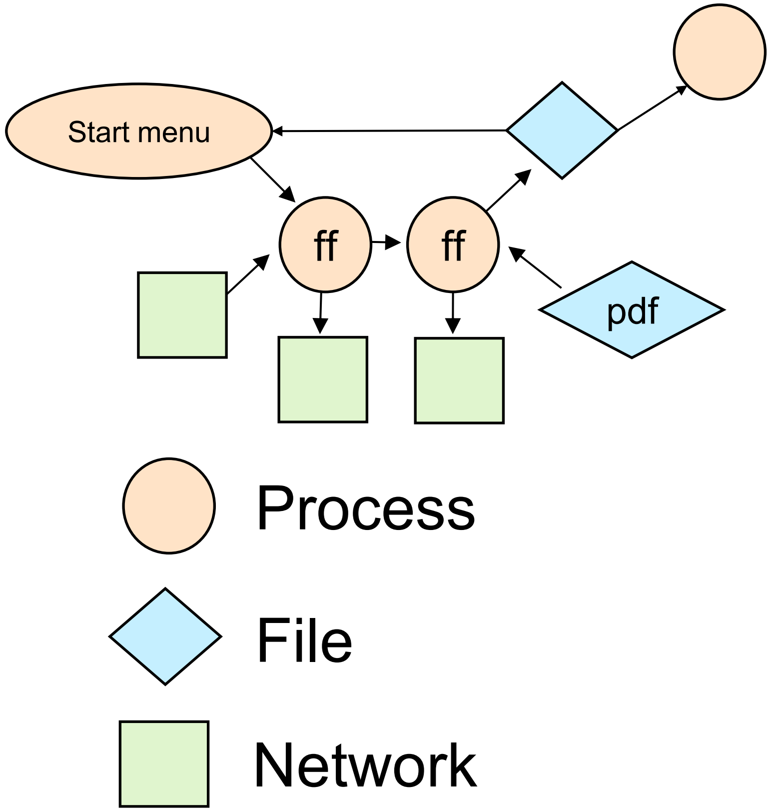


Agenda

- Background
- **Motivation**
- ProvDP: Differential Privacy Framework
- Evaluation
- Discussion

Provenance-Based IDS Data Sensitivity

- Edge/Node attributes can be used maliciously (e.g. business client names)
  - Trivially solved by masking file names and IPs
- Structure of the graph reveals user behavior
  - Can be used for spearphishing or targeted malware



Motivation

- Provenance-based IDS require a lot of data.
- Sharing data will improve datasets, and allow better defenses
- But system provenance data is inherently private!

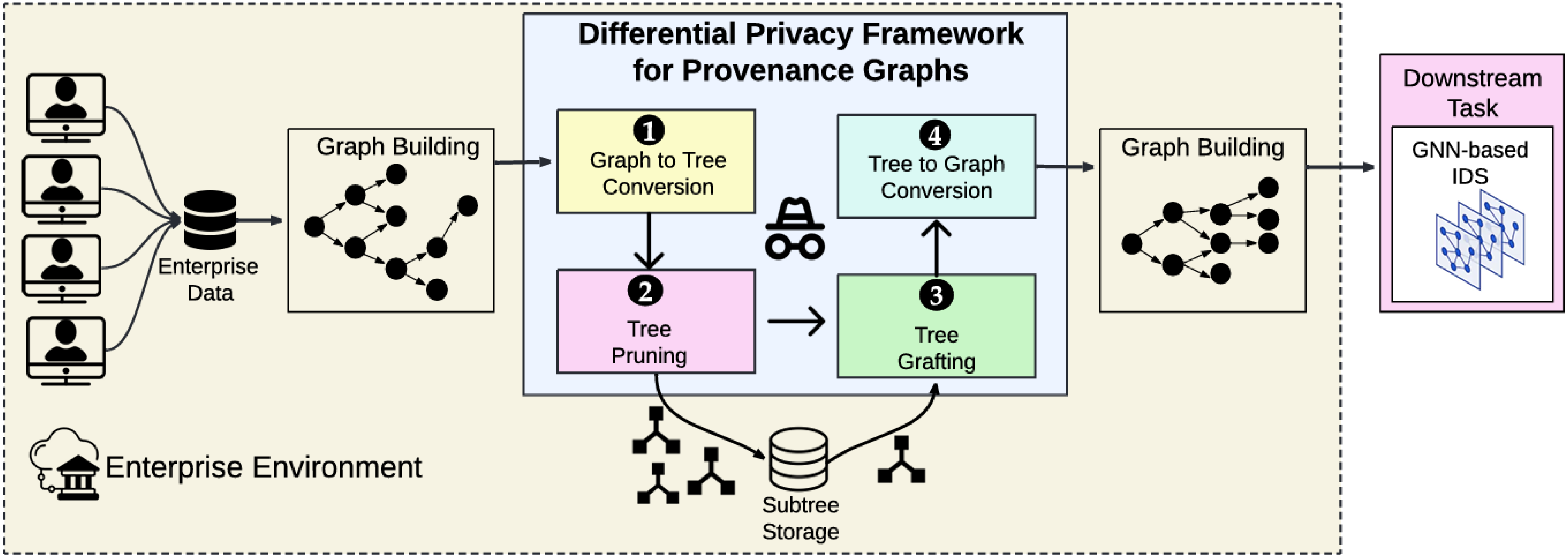
- Solution: Create a framework to allow sharing provenance data privately

- There is **no** existing work applying differential privacy to system provenance graphs.

Agenda

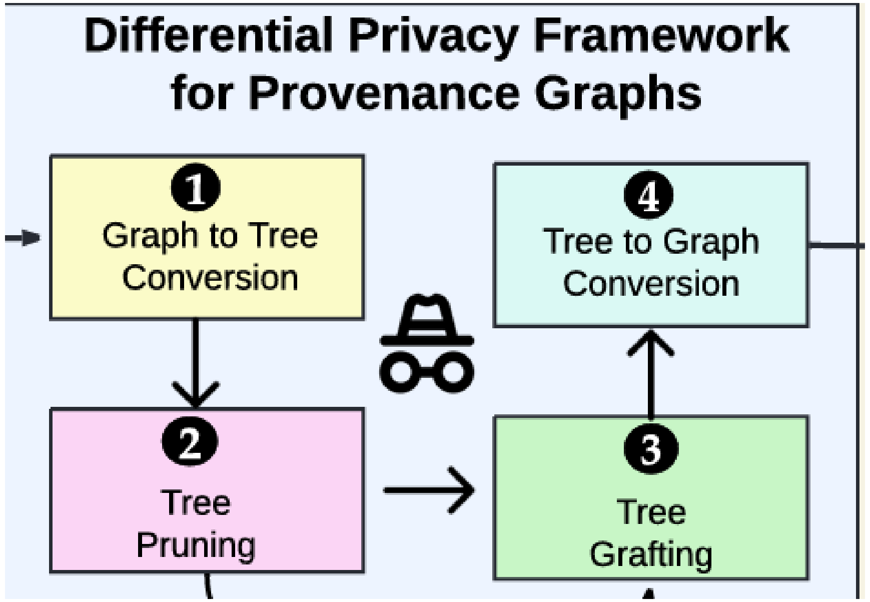
- Background
- Motivation
- **ProvDP: Differential Privacy Framework**
- Evaluation
- Discussion

ProvDP: Differential Privacy Framework

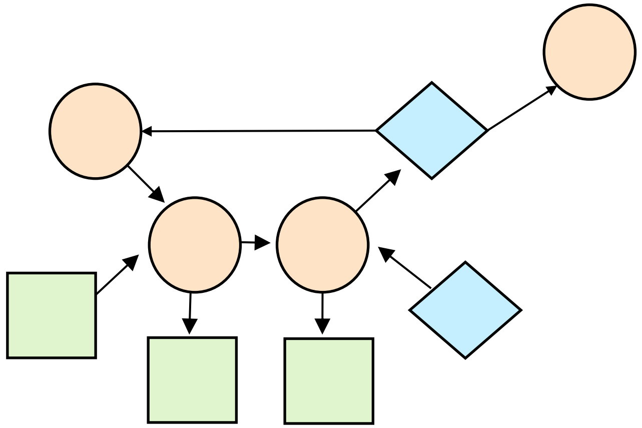


Privacy Budget Allocation

- Pruning and Grafting are both differentially private mechanisms
- $\epsilon = \epsilon_1 + \epsilon_2 =$ total privacy budget
- $\delta \in [0, 1]$ controls allocation of budget
- Pruning $\epsilon_1 = \delta\epsilon$
- Grafting $\epsilon_2 = (1 - \delta)\epsilon$

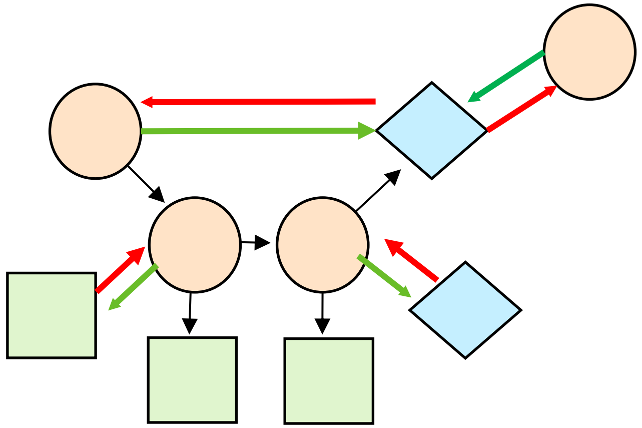


Graph to tree conversion



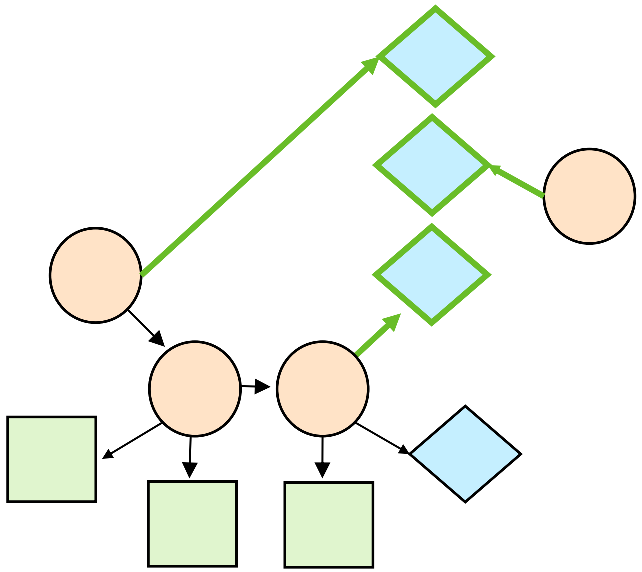
Graph to tree conversion

1. Break cycles: Invert outgoing edges from file/network nodes
    - Edge direction can be restored from metadata
    - Graph is now acyclic

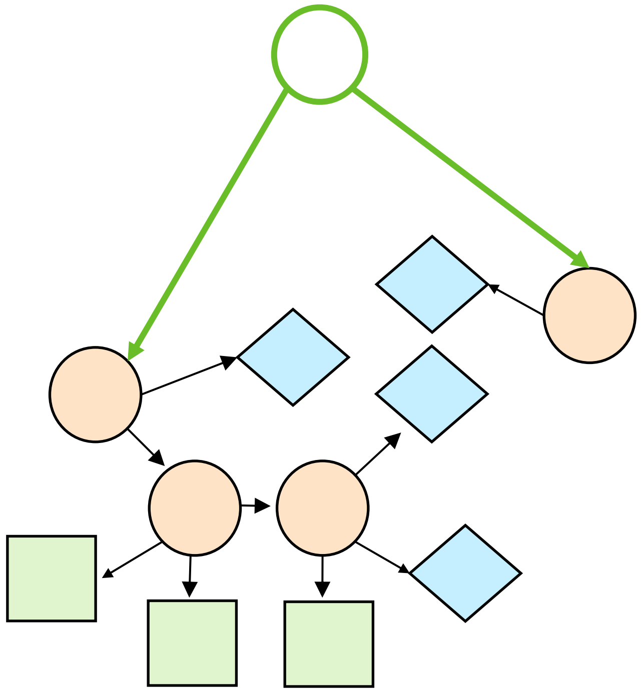


Graph to tree conversion

1. Break cycles:
   - Invert outgoing edges from file/network nodes
   - Edge can be restored from metadata
   - Graph is now acyclic
2. Remove lattice structure:
   - Duplicate file/network nodes for each in edge
   - Can be restored from file path / IP address / port
   - Removes lattice structure
   - Graph is now a forest



Graph to tree conversion



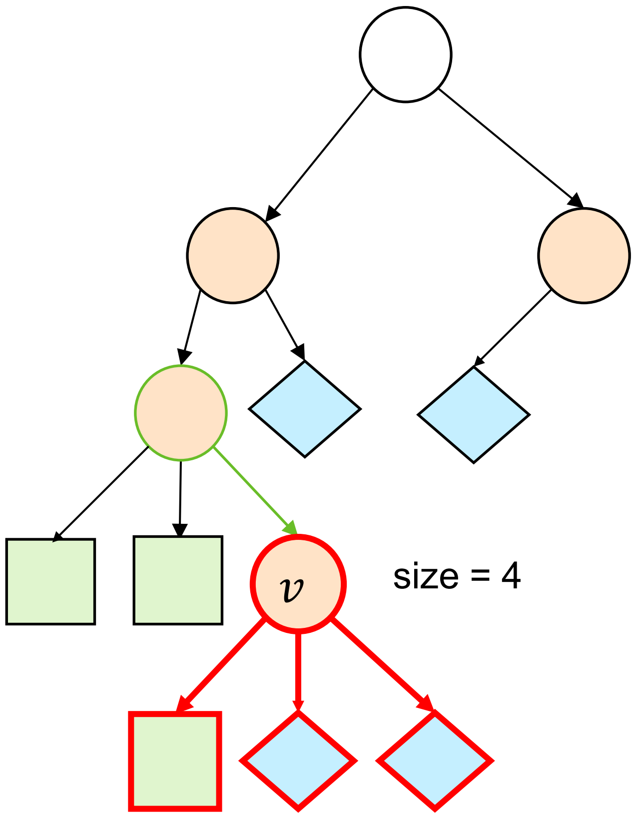
1. Break cycles:
   - Invert outgoing edges from file/network nodes
   - Edge can be restored from metadata
   - Graph is now acyclic
2. Remove lattice structure:
   - Duplicate file/network nodes for each edge
   - Can be restored from file path / IP address / port
   - Removes lattice structure
   - Graph is now a forest
3. Forest to tree:
   - Connect all process roots to a virtual root node
   - Graph is now a tree

Pruning Algorithm

- Run on each graph inside dataset
- Starting at the root node, traverse the graph.
- Randomly prune subtree rooted at node v
  - $S(v)$ is a function of subtree size, height, depth, outdegree
  - Each feature is weighted by $\alpha, \beta, \gamma, \eta$, respectively.

$$P(\text{prune } v) = \frac{1}{1 + e^{\epsilon_1 / 2S(v)}}$$

- For each pruned subtree:
  - Mark v for and store the subtree size s_v
  - Store pruned subtree, along with its parent node and edge

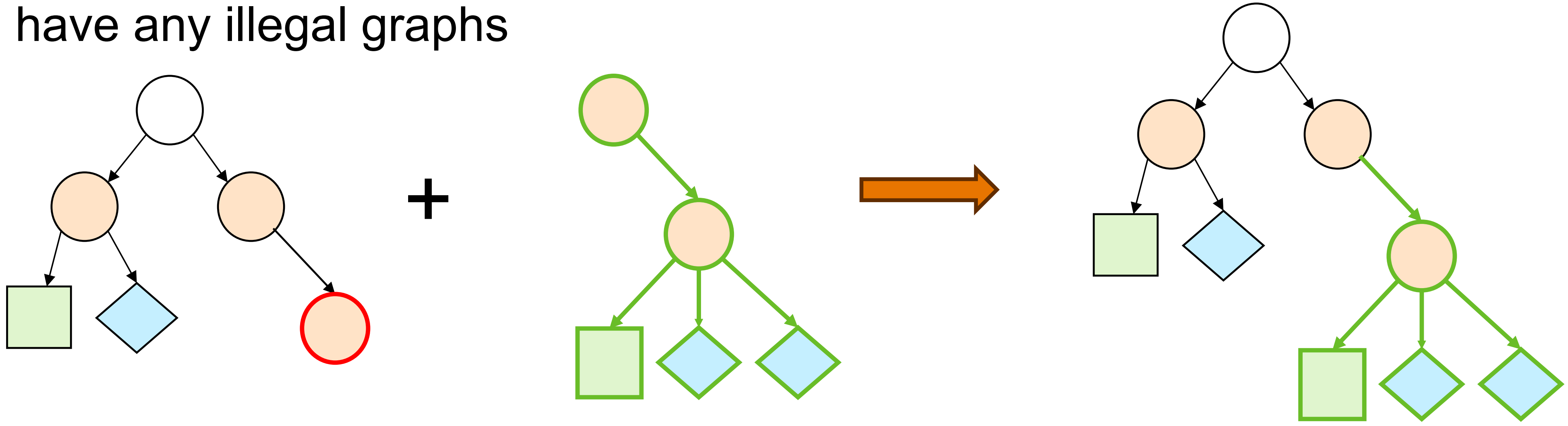


Grafting Algorithm

- **After pruning all graphs:**
  - Bucket of all pruned subtrees B
  - Graphs have marked nodes indicating where we pruned
  - Marked nodes have size of the original subtree s_v
- Randomly replace all marked nodes
  - Perturb original size s_v by adding noise: $\tilde{s}_v = s_v + Lap(\frac{1}{\epsilon_2})$
  - Randomly sample a subtree from B
  - Each subtree $t \in B$ has probability $\mathrm{p}_t = x/(1 + |\tilde{s}_v - s_t|)$ of being chosen
    - Normalization factor $x = 1/\sum_{t \in B} p_t$

Note on Grafting

- When pruning, store the subtree, and its parent relation
- When grafting, we replaced the marked node and its incoming edge
- Replacing the incoming edge guarantees we don't have any illegal graphs



Agenda

- Background
- Motivation
- ProvDP: Differential Privacy Framework
- **Evaluation**
- Discussion

Evaluation Baseline: Extended Top-m Filter

- Top-m Filter (TmF) [Nguyen '15]
  - Edge-differentially private
  - Efficiently flips bits in adjacency matrix
  - Designed for undirected graphs
  - randomly creating edges can lead to illegal provenance graphs

- Extended Top-m Filter (ETmF)
  - Input: Source and Destination vertices V_s, V_d
  - Perturb upper diagonal of adjacency matrix of subgraph only containing nodes in V_s, V_d, and edges $(u, v) \in V_s \times V_d$
  - Run ETmF for all possible legal source, destination pairs (e.g. process -> file, file -> process, …)

Evaluation: IDS Performance

- Trained GNN-based IDS on different datasets
- ProvDP adds noise more strategically under the same budget

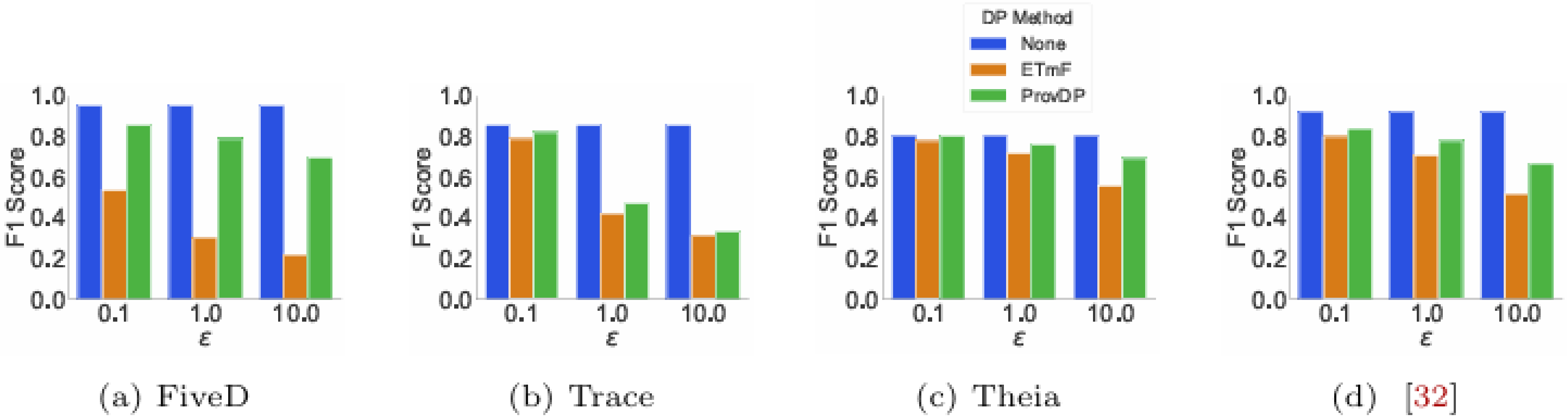
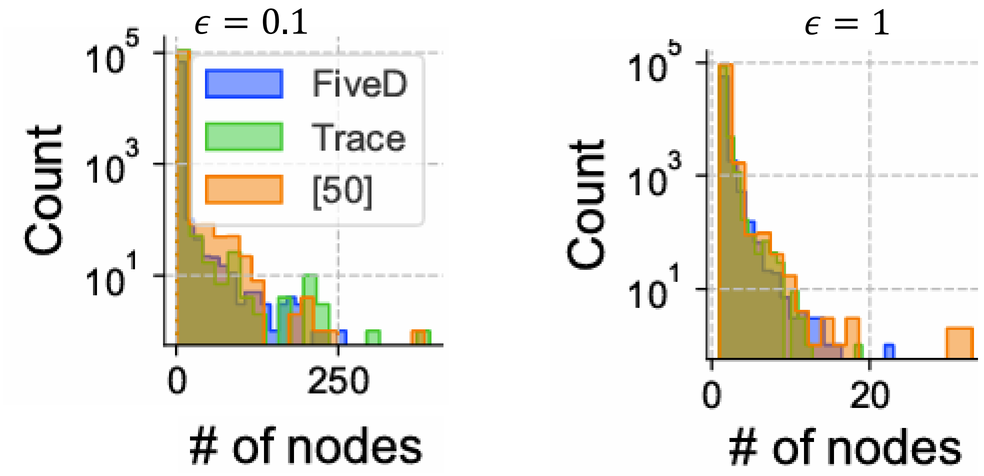


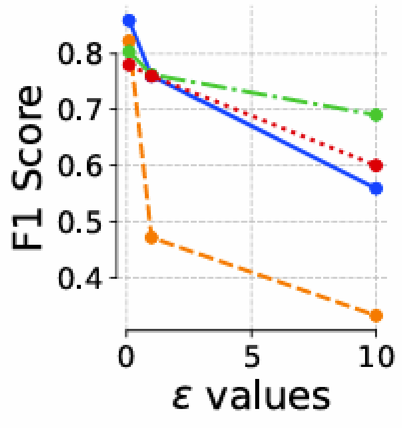
Fig. 3: Detection performance of GNN-based IDS using different privacy budgets.

Evaluation: Privacy Budget

Count of subtrees pruned, grouped by subtree size



IDS Performance



Agenda

- Background
- Motivation
- ProvDP: Differential Privacy Framework
- Evaluation
- **Discussion**

Discussion

- Real-world implementation
- Scalability (Grafting $O(n^2)$)
- Generalization to alternative IDS models (ex. Path or subgraph based)

Thank you for your time!

Jonathan Yu
me@jonathancyu.dev
https://www.linkedin.com/in/jonathancyuu/

Kunal Mukherjee *
kunmukh@gmail.com
www. KUNMUKH.com

* In Academic job market for Fall 2026

References

Wagner & Soto ’02 - Wagner, David, and Paolo Soto. "Mimicry attacks on host-based intrusion detection systems." *Proceedings of the 9th ACM Conference on Computer and Communications Security*. 2002.

Tan & Maxion ’03 - Tan, Kymie MC, and Roy A. Maxion. "Determining the operational limits of an anomaly-based intrusion detector." *IEEE Journal on selected areas in communications* 21.1 (2003): 96-110.

Velickovic ’17 - Veličković, Petar, et al. "Graph attention networks." *arXiv preprint arXiv:1710.10903* (2017).

Hassan ’19 - Hassan, Wajih Ul, et al. "Nodoze: Combatting threat alert fatigue with automated provenance triage." *network and distributed systems security symposium*. 2019.

Dang ’19 - Dang, Fan, et al. "Understanding fileless attacks on linux-based iot devices with honeycloud." *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*. 2019.

Ying ’19 - Ying, Zhitao, et al. "Gnnexplainer: Generating explanations for graph neural networks." *Advances in neural information processing systems* 32 (2019).

Wang ’20 - Wang, Qi, et al. "You Are What You Do: Hunting Stealthy Malware via Data Provenance Analysis." *NDSS*. 2020.

Han ‘21 - Han, Xueyuan, et al. "{SIGL}: Securing Software Installations Through Deep Graph Learning." *30th USENIX Security Symposium (USENIX Security 21)*. 2021.

Barr-Smith ’21 - Barr-Smith, Frederick, et al. "Survivalism: Systematic analysis of windows malware living-off-the-land." *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2021.

Zeng ‘22 - Zeng, Jun, et al. "Shadewatcher: Recommendation-guided cyber threat analysis using system audit records." *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022.

Colonial – Easterly, Jen “The Attack on Colonial Pipeline: What We’ve Learned & What We’ve Done over the Past Two Years: CISA.” Cybersecurity and Infrastructure Security Agency CISA, 8 Aug. 2023, www.cisa.gov/news-events/news/attack-colonial-pipeline-what-weve-learned-what-weve-done-over-past-two-years.

SolarWinds - “The Solarwinds Cyber-Attack: What You Need to Know.” *CIS*, 9 Nov. 2021, www.cisecurity.org/solarwinds.

Nguyen, H.H., Imine, A., Rusinowitch, M.: Differentially private publication of social graphs at linear cost. In: Proceedings of the 2015 IEEE/ACM Interna tional Conference on Advances in Social Networks Analysis and Mining 2015. p. 596–599. ASONAM ’15, Association for Computing Machinery, New York, NY, USA(2015).