Network Detection of Interactive SSH Imposters Using Deep Learning

Julien Piet

UC Berkeley & Corelight

Aashish Sharma

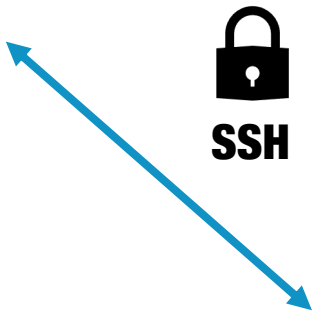
Lawrence Berkeley National Laboratory

Vern Paxson

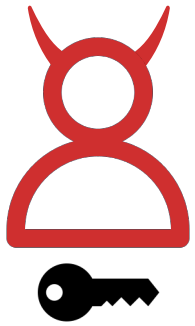
Corelight & UC Berkeley

David Wagner

UC Berkeley



SSH

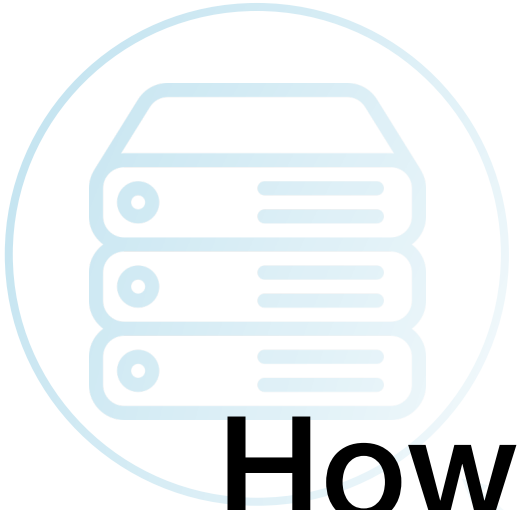


+

MFA

Execute any code with user privileges

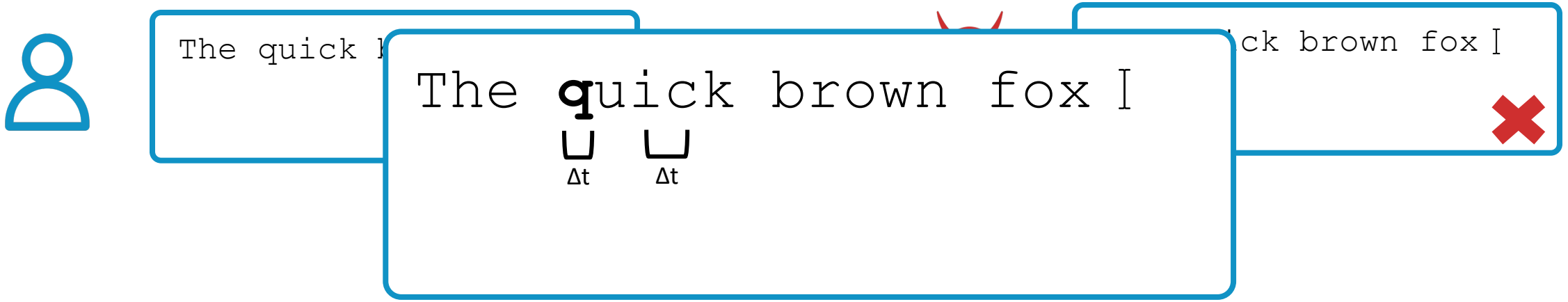
MFA devices can still be stolen



How to detect SSH impostors?

Recognize user behavior!

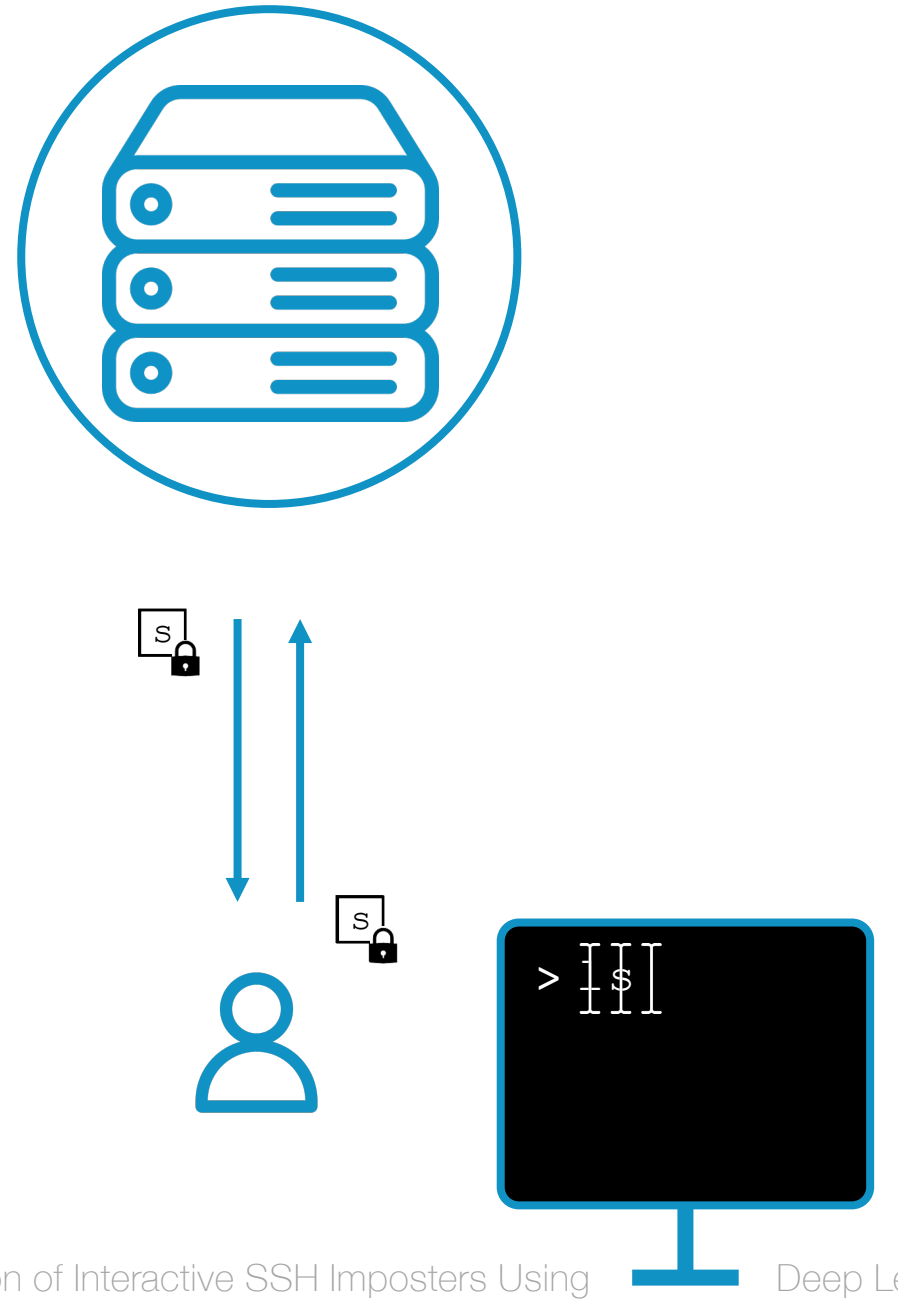
Keystroke Authentication



Existing techniques use keys, keypress and inter-keypress durations.

Would require using keyloggers → **deployment hurdle**

Would require using keyloggers → **privacy risk**



Keystrokes in SSH

Each keystroke is its own packet and is echoed by the server.

- Easy to identify keystrokes
- Can recover timing

Is it enough for authentication?

Contributions

Keystroke timings are enough for **scalable** and **accurate** authentication!

We leverage real network data with over **600,000** unique SSH sessions over **5 years**

Using deep learning, we authenticate users:

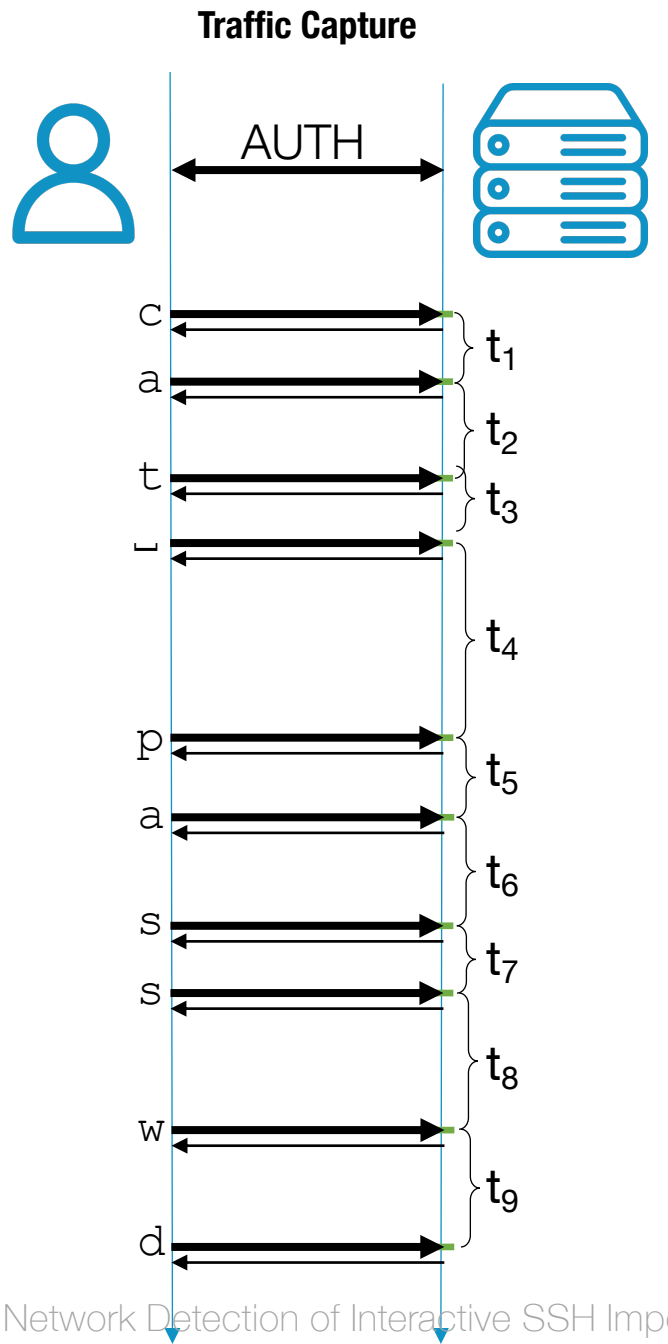
In as little as 10 seconds.

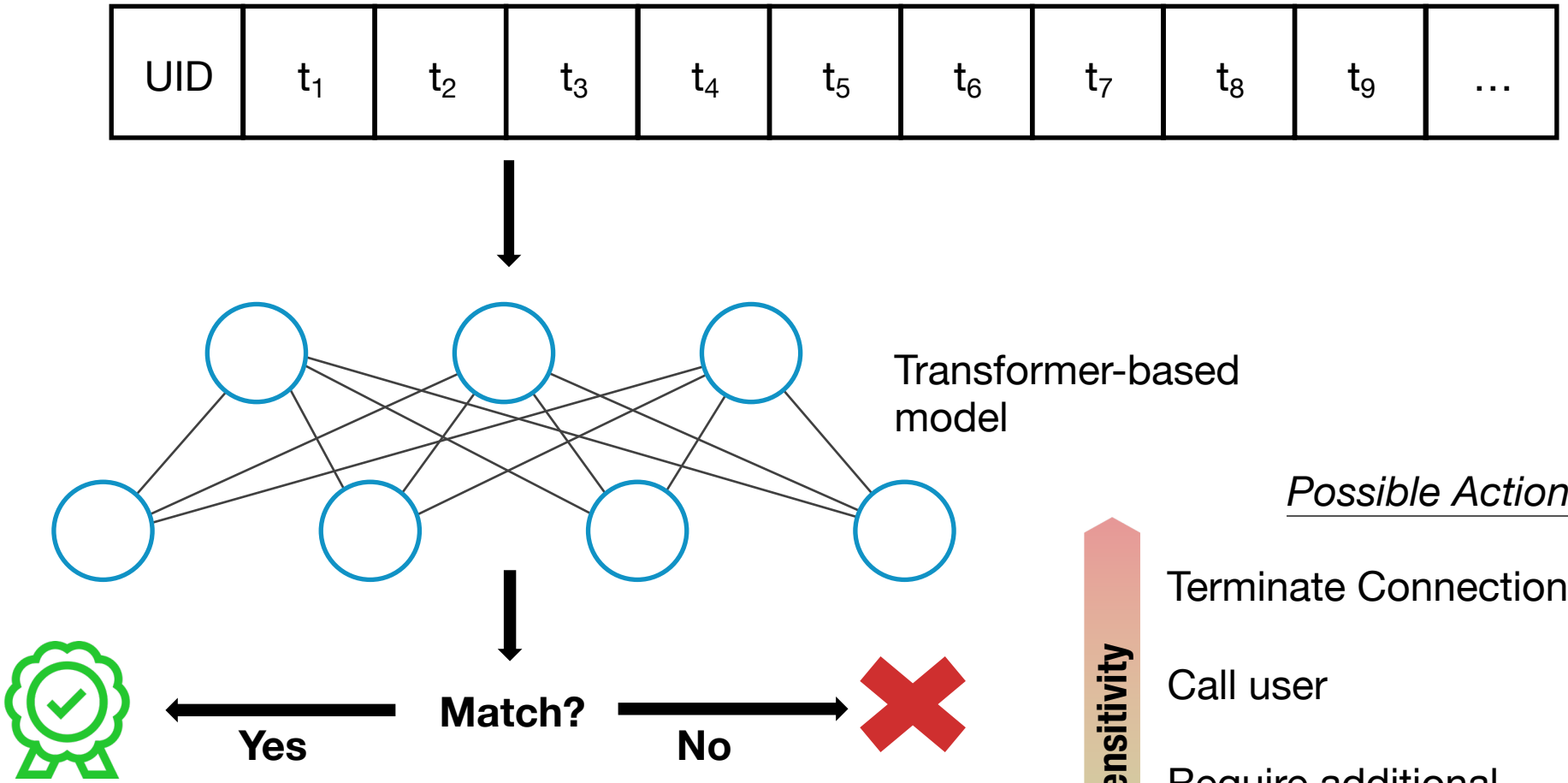
Among hundreds of unique users.

With under four minutes of training data per user.

In real network environments with congestion.

System Design



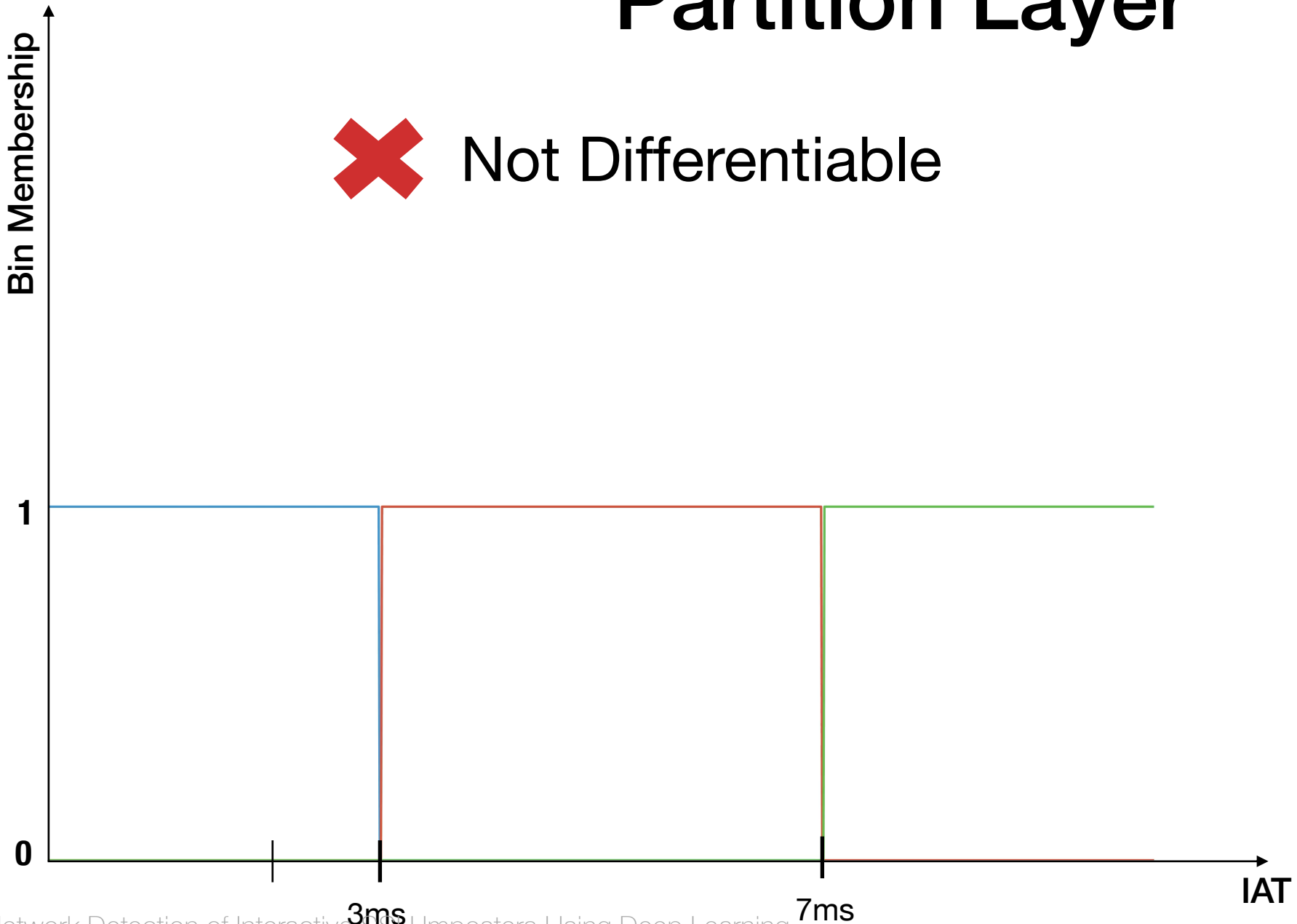


Possible Actions

Security Sensitivity

- Terminate Connection
- Call user
- Require additional factor
- Log anomaly

Partition Layer



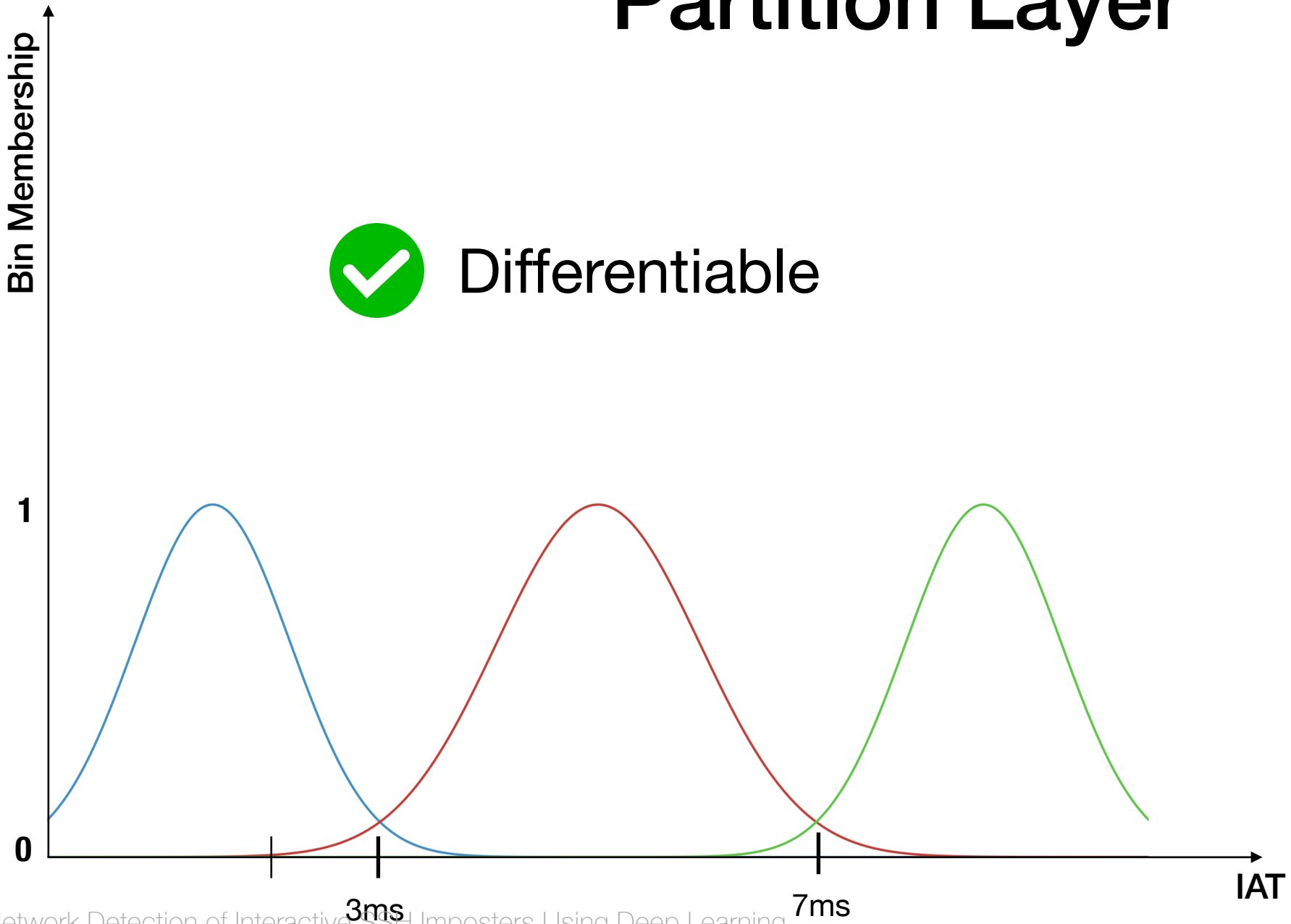
Not Differentiable

Inter-Arrival Time

2ms

Bin A	0.77
Bin B	≈0
Bin C	≈0

Partition Layer



Differentiable

Inter-Arrival Time

2ms

Bin A	0.77
Bin B	≈0
Bin C	≈0

Data Processing

BERKELEY LAB

5 years of border traffic.
600,000 interactive SSH session
4,000 users, **2,750** servers

Network Tap + Centralized System Logs

Process

Training: 3 months of data | **Testing: 1 month of data**

Select users with enough traffic

Use negative sampling for simulating impostors

Retrain once a month

Results

Training Threshold		
>15,360 keystrokes *1 hour of typing* **66 Users**	> 5,120 keystrokes *19 min of typing* **183 Users**	> 1,024 keystrokes *4 min of typing* **444 Users**
Evaluation Results		
8 FPs/day 1% FNR	17 FPs/day 2% FNR	29 FPs/day 6% FNR

Discussion

Scalable and non-intrusive impostor detection

Accurate for **months** & low FNR for **years** after training

Robust to congestion and multi-device users

Operational impact of **false positives**

User **coverage**

We leverage keystroke dynamics to authenticate users over interactive SSH channels

We identify 98% of imposters, incurring a manageable load of false positives

We evaluated on 5 years of real-world data with hundreds of users

Link to code

Thank you for your attention!

If you have any questions, feel free to reach out at **piet@berkeley.edu**