Storytelling Security: Scalable Causal Analysis for Host-Wide Anomaly Detection

Danfeng (Daphne) Yao

Assistant Professor

Department of Computer Science

Virginia Tech

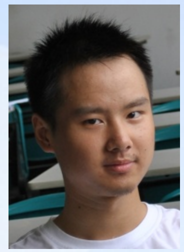
ARO STIR Report August 2012

Personnel and Collaborators in Yao group

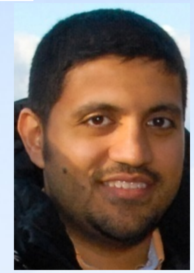
Current Ph.D. students


Kui Xu


Huijun Xiong


Johnny Shu


Tony Zhang


Hussain Almohri


Karim Elish

Incoming Ph.D. student

Fang Liu

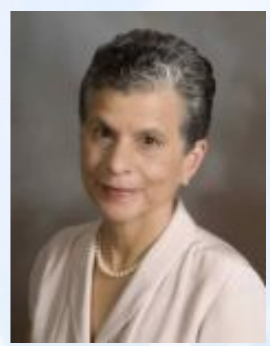
Graduated Ph.D. student

Saman Zarandioon


at Amazon.com

Collaborators in CS Dept


Naren Ramakrishnan


Barbara Ryder


Layne Watson

Virginia Tech.

Anomaly Detection For System Assurance

Problem: how to ensure system assurance?

- Signature based scanning, firewalls, IDS/IPS
- To detect malware behaviors at run time
  - E.g., system call execution, memory/stack access
- But what about zero-day malware/exploit?
  - To avoid infection

    E.g., to prevent remote code execution, MTD
  - To detect changes in code base

    E.g., TPM attestation
  - Anomaly detection

    E.g., [Denning '87], [Forrest et al. '96], [Sekar '01], [Giffin '04]
- But how to define the normalcy of a host?

Virginia Tech.

Requirements and Challenges of Anomaly Detection

Anomaly detection requires:

- Definitions for the <u>norm</u> or normalcy, or
- Mechanisms to <u>learn</u> normal patterns, and
- Mechanisms to observe and collect <u>authentic</u> data

Why simple statistical methods are inadequate in computer anomaly detection?

→ Data diversity

→ Data semantics

State-of-the-art anomaly detection solutions are limited to system calls

Our goal: *<u>host-wide</u> monitoring and anomaly detection*

Our <u>storytelling security</u> approach: to perform scalable structured causal analysis of events on a computer

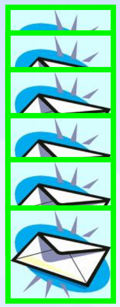
Virginia Tech

Our Existing Work on User-Intention Based Traffic Dependence Analysis

H. Zhang, D. Yao, N. Ramakrishnan, and M. Banick.
User Intention-Based Traffic Dependence Analysis for Anomaly Detection. ***Workshop on Semantics and Security (WSCS)***, in conjunction with *IEEE S&P*. 2012.

Virginia Tech.

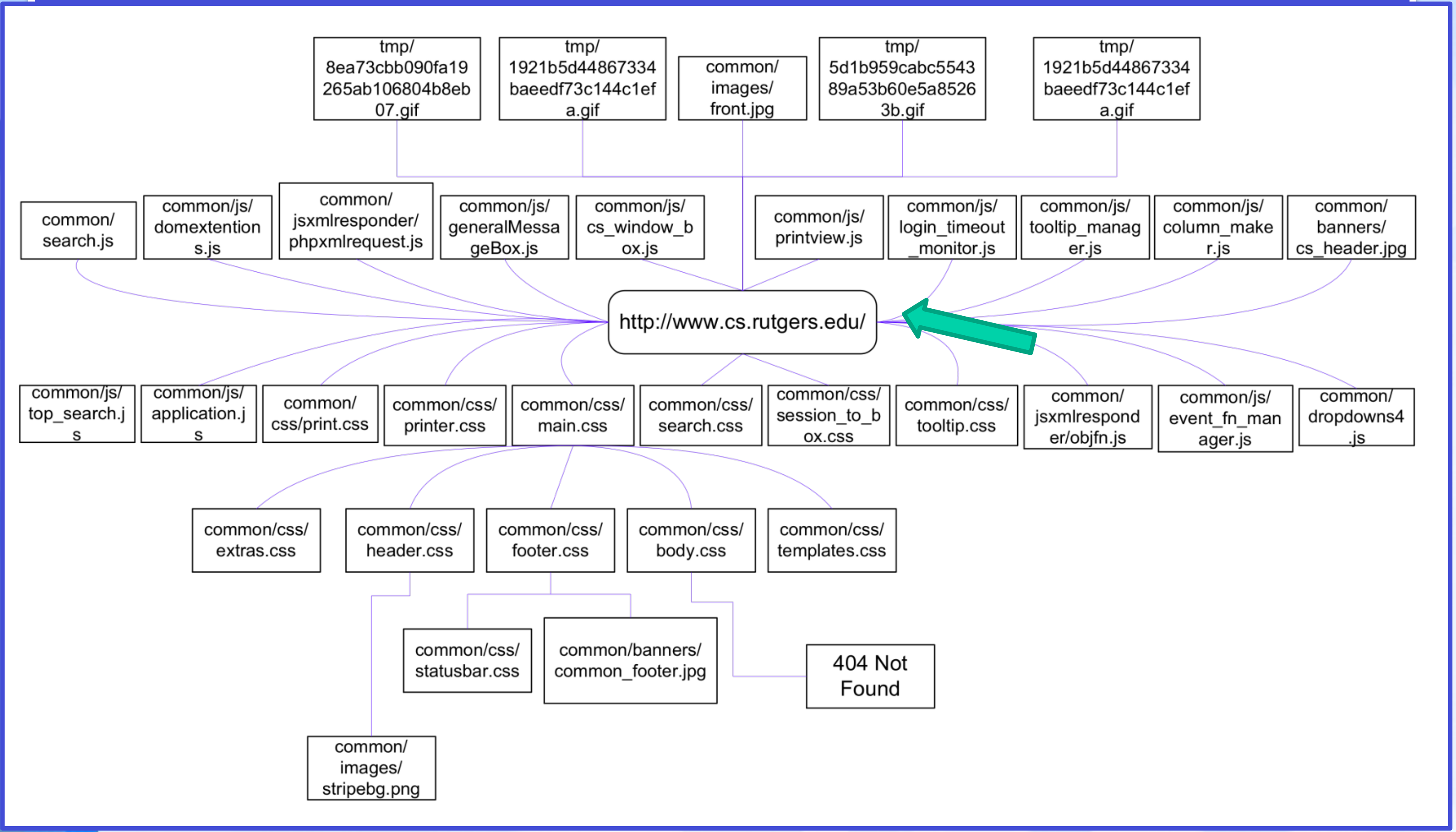
Cause and Effect in Traffic Anomaly Detection



How to distinguish the malicious outbound packets from the legitimate ones on a host?

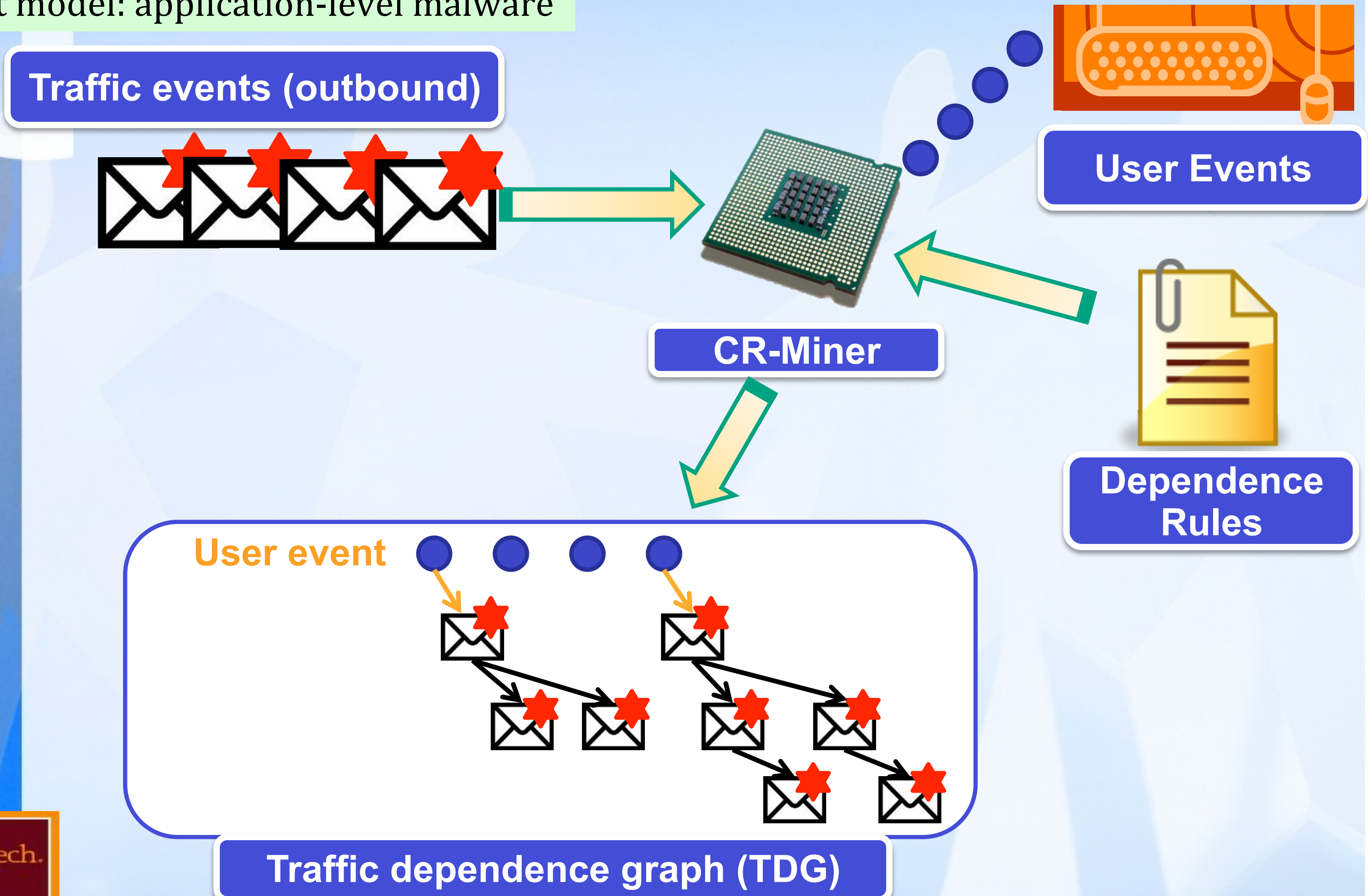
Our approach: To identify ***dependence*** among outbound traffic

A Technical Challenge



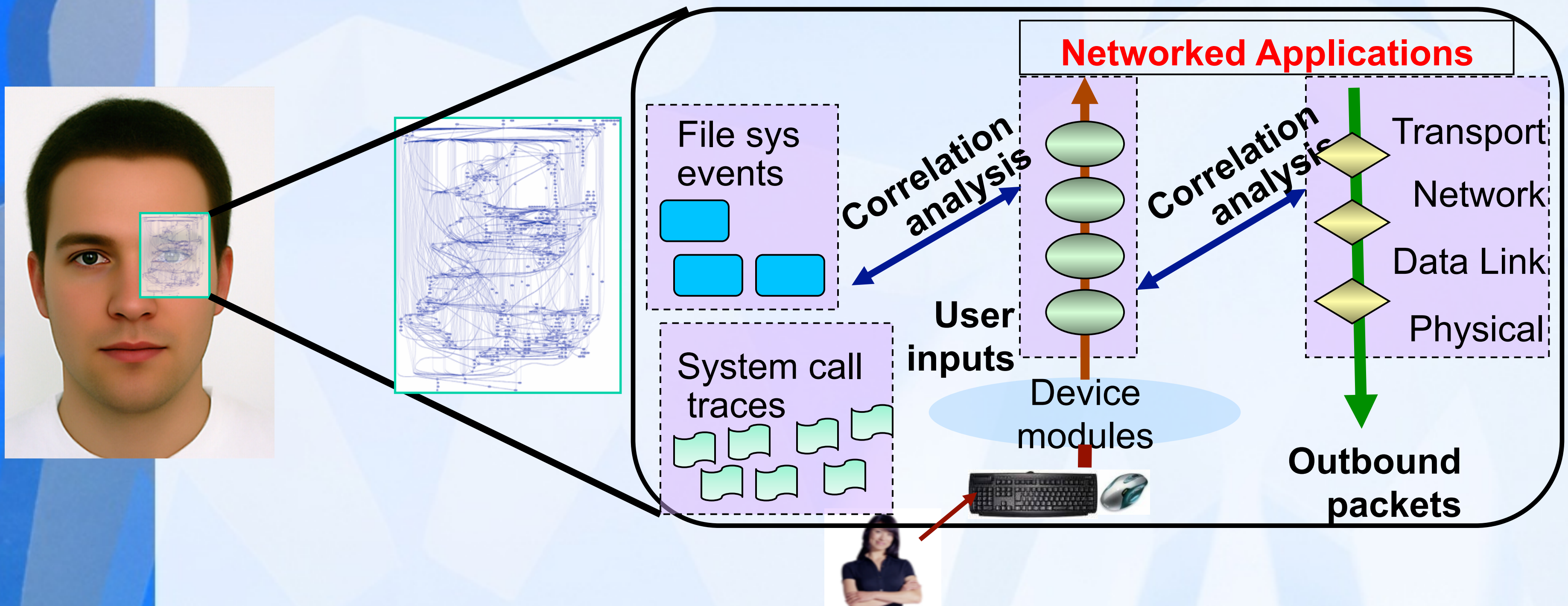
Work Flow of CR-Miner

Threat model: application-level malware



Virginia Tech

Our Storytelling Security Vision: Scalable Structured Causal Analysis for Host-Wide Monitoring



Virginia Tech.

Events and their attributes

Dependence rules specify relations of attributes of dependent events

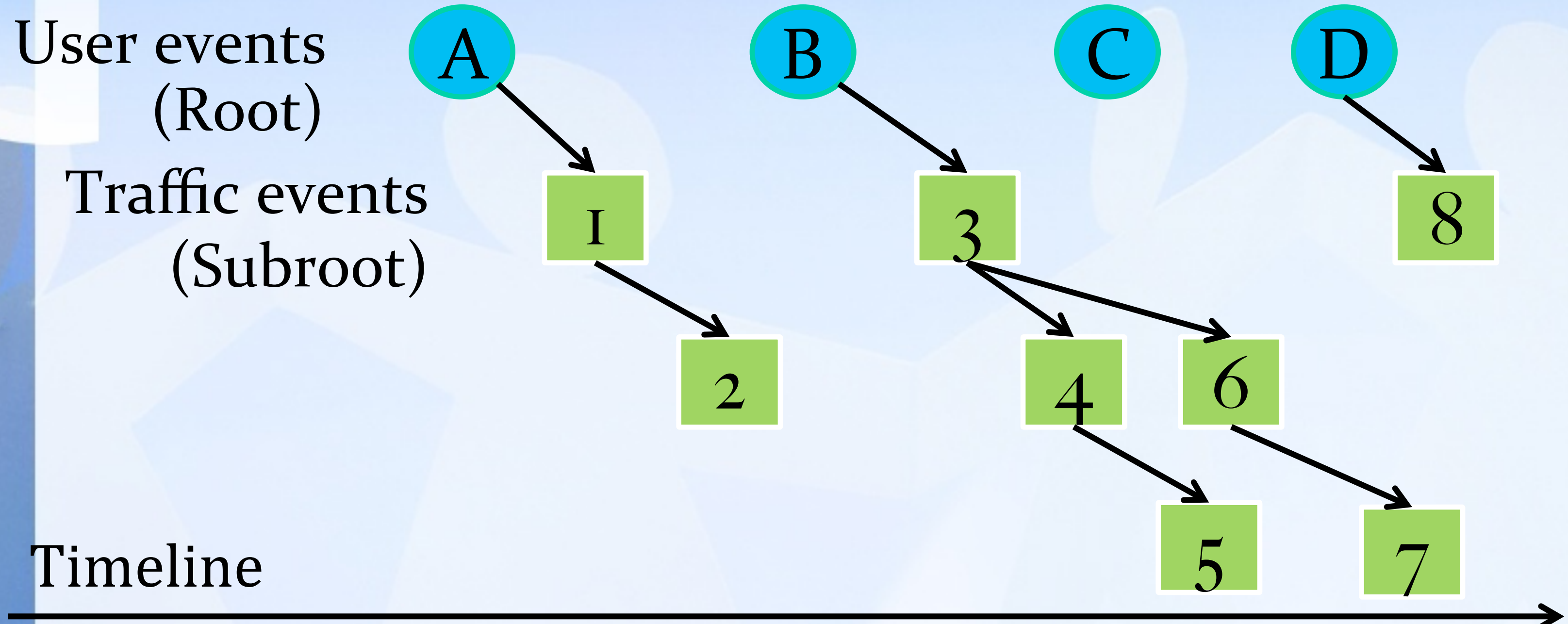
User events

	Timestamp	Event Name	Value	URL
A	0:0:01.077	KeyDown	Return	http://www.engadget.com/
B	0:0:02.910	MouseClick - Left	X=1069 Y=474	http://www.cnet.com/
C	0:0:03.000	Wheel	-120	N/A

Traffic events

	Timestamp	Object Requested	Remote Domain Name	Referrer
1	0:0:02.863	/	www.engadget.com	http://www.engadget.com/
2	0:0:02.873	/media/main.css	www.engadget.com	http://www.engadget.com/...
3	0:0:03.113	/	www.cnet.com	null

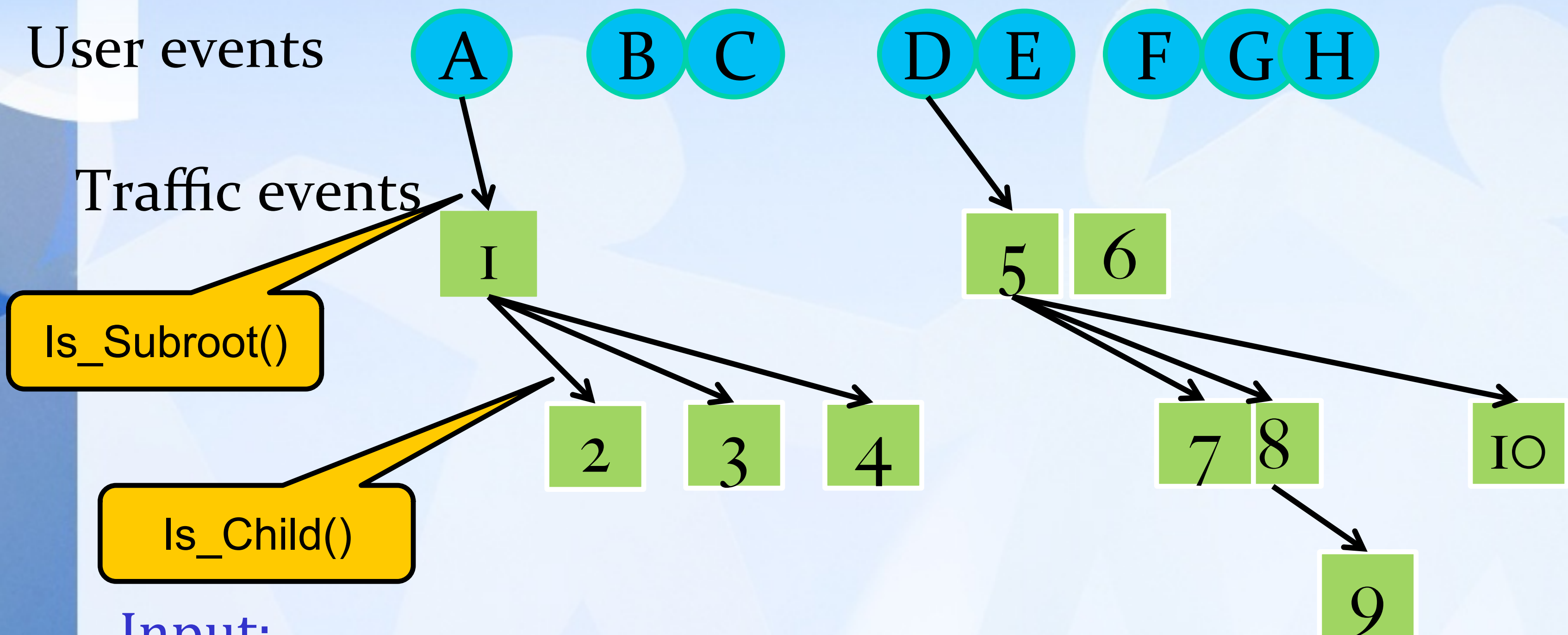
Definitions in Our Traffic Dependency Graph (TDG)



Definition of security: a legitimate traffic event should belong to a tree in a TDG that is rooted at a legitimate user event.

Otherwise, it is a vagabond traffic event

Our BFS-Based Algorithm to Construct Traffic Dependence Graph

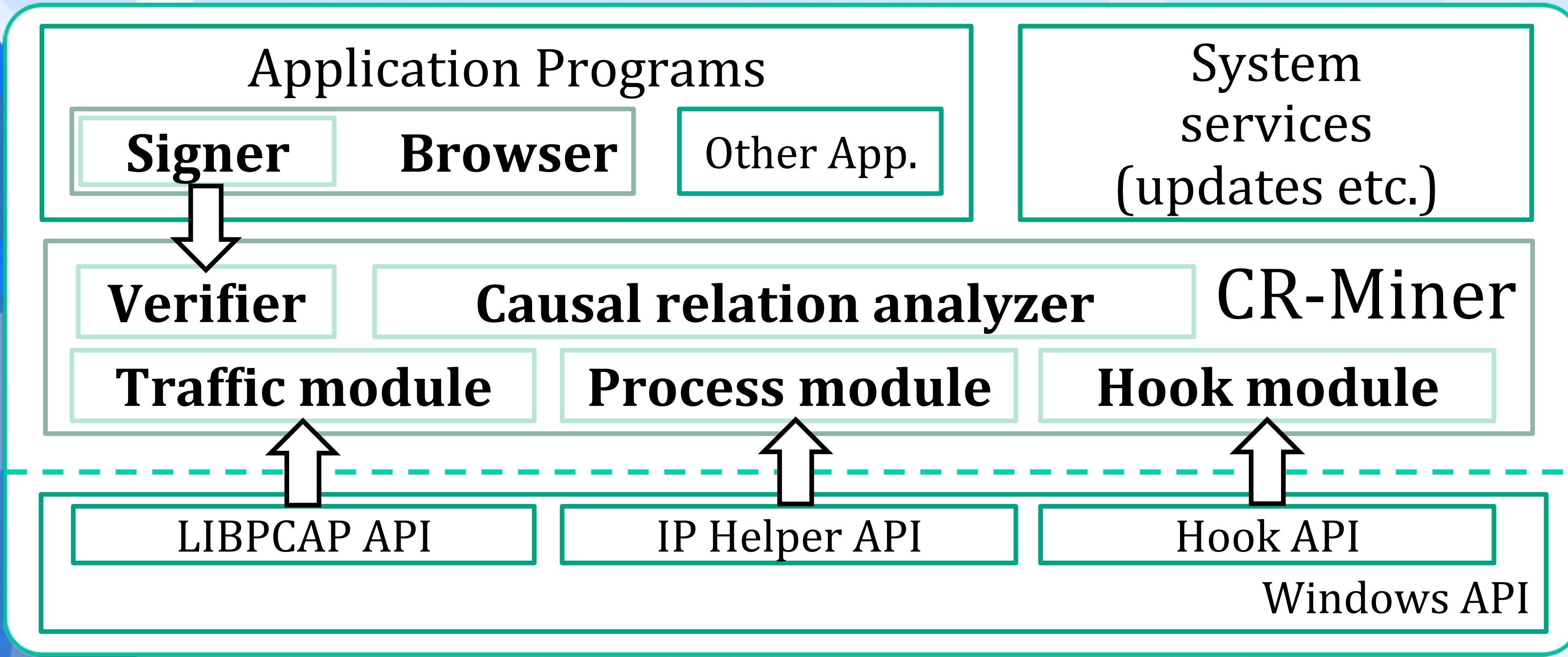


Input:
- an existing TDG (trees of events, which root at user events)
- a new outbound traffic event ***q***

Output:
- whether or not ***q*** is legitimate

Implementation Architecture

Our prototype in Windows is called CR-Miner.



Signer and verifier for the integrity of HTTP requests with MAC

Highlights on Experiments

User study with 20 participants; 30-minute surfing for each user

Hit rate: percentage of traffic events whose parents are identified by CR-Miner

1. How accurate is the dependency inference algorithm?
   - ≥ 98% hit rates for all users
   - Average 99.6% with white listing (0.4% contains true positives)
   - 99.72% for top 20 Alexa.com websites (i.e., 0.28% false positives)
2. Does the inference accuracy suffer in noisy traffic?
   - 99.2% accuracy in two-user merged data set
3. Can we detect real-world stealthy malware traffic?
   - Infostealer spyware
   - Proof-of-concept password sniffier (malicious Firefox extension similar to Firespyfox)

H. Zhang, D. Yao, N. Ramakrishnan, and M. Banick.
User Intention-Based Traffic Dependence Analysis for Anomaly Detection**.** ***Workshop on Semantics and Security (WSCS)***, in conjunction with *IEEE S&P*. 2012.

Virginia Tech.
VT

Related Work in Yao Group

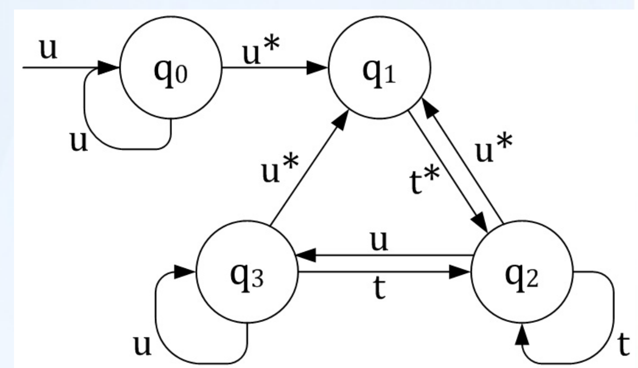
- What/who causes outbound traffic
  - [Hao et al. IEEE WSCS ’12]
- What/who downloads files on the computer
  - [Xu et al. NSS ‘11]
- Where the keystroke is from; where the packet is from (cryptographic provenance verification)
  - [Xu et al. IEEE TDSC ‘12]
- Whether or not the apps behave
  - [Elish et al. IEEE MoST ‘12]

→ **For preserving system integrity**

Virginia Tech.

Future Work on Storytelling Security

- **To automatically mine causal relations with machine learning techniques**
  - E.g., how to define features considering the data diversity and semantics?
  - Our preliminary work on naïve Bayesian classifier (for pair-wise dependencies) gave promising results
- **To model storytelling security, theoretical analysis**
  - E.g., general requirements, components, workflow, limitations
  - E.g., FSA representation
  - E.g., connection with Schneider's EM



- **Experimental demonstration**
  - E.g., including DNS traffic in traffic dependency analysis
  - E.g., analysis of server-side applications

Virginia Tech