Cloud Data Analytics for Security: Applications, Challenges, and Opportunities

Daphne Yao
Associate Professor of Computer Science
Turner Fellow and L-3 Fellow
Virginia Tech

Motivation: Security/Privacy as Enablers

My past work: Security Methodology Development

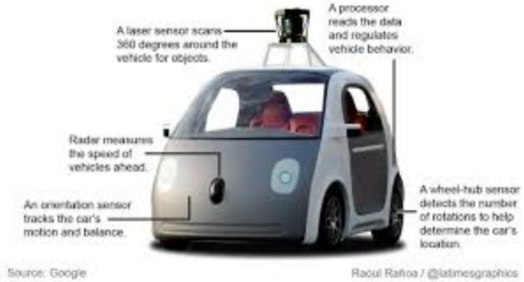
Near-0 false alarm enables analysts to focus on real attacks

Ongoing & future work: Intelligent secure systems and platforms that benefit large populations

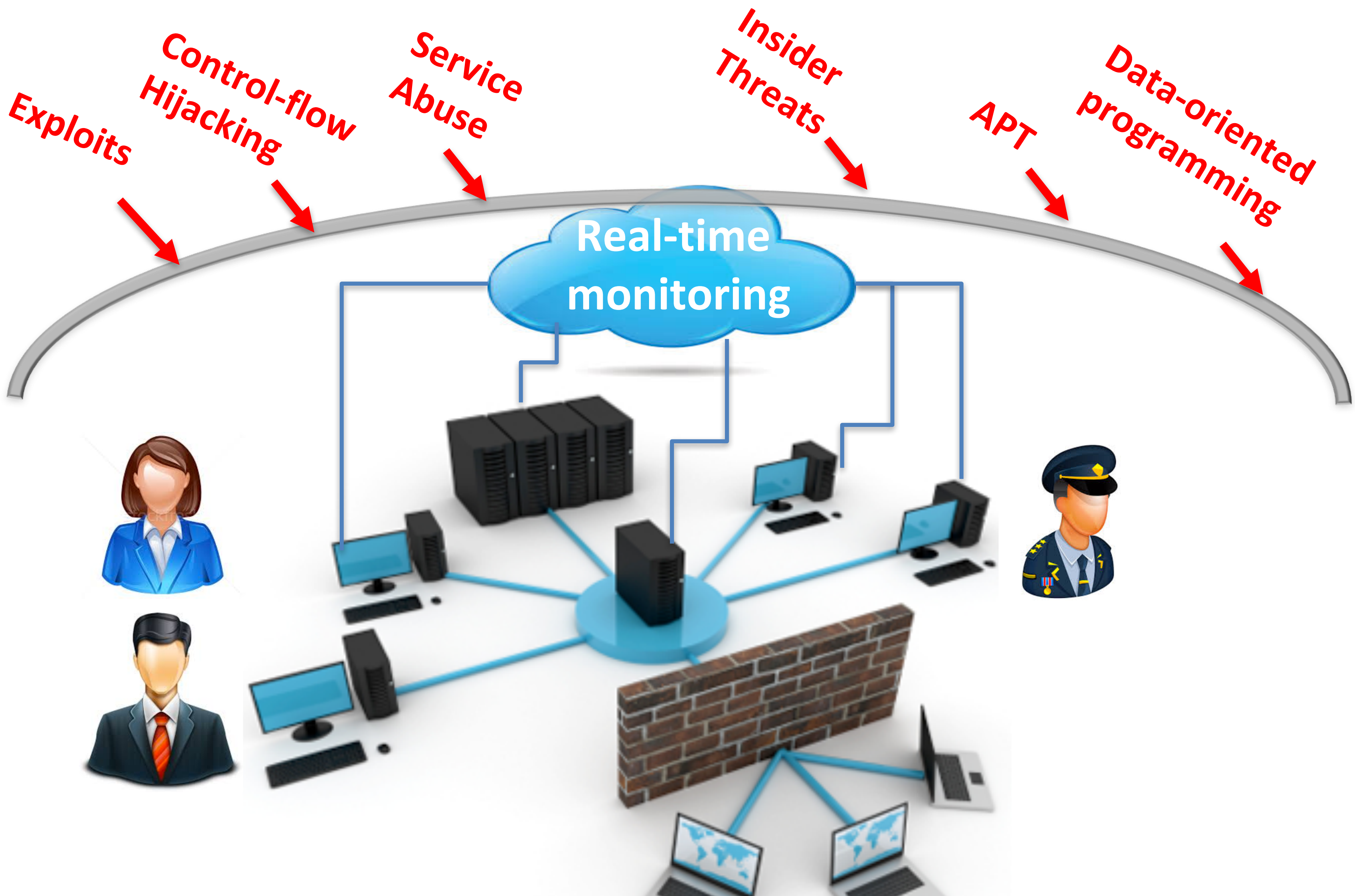
Enable new infrastructures

Improve quality of life

Enable new discoveries



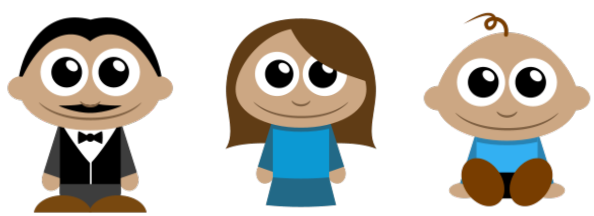
A Scenario:
Cloud Data Analytics for Organizational Security



Another Scenario:
Cloud Data Analytics for Smart Home Security

Origins of spam in a 2014 botnet study

- Embedded Linux servers
- mini-httpd, apache
- ARM devices, MIPS, Realtek chipset
- Open telnet, an SMTP server



https://www.proofpoint.com/us/threat-insight/post/Your-Fridge-is-Full-of-SPAM

A vision: To lift host protection to the cloud

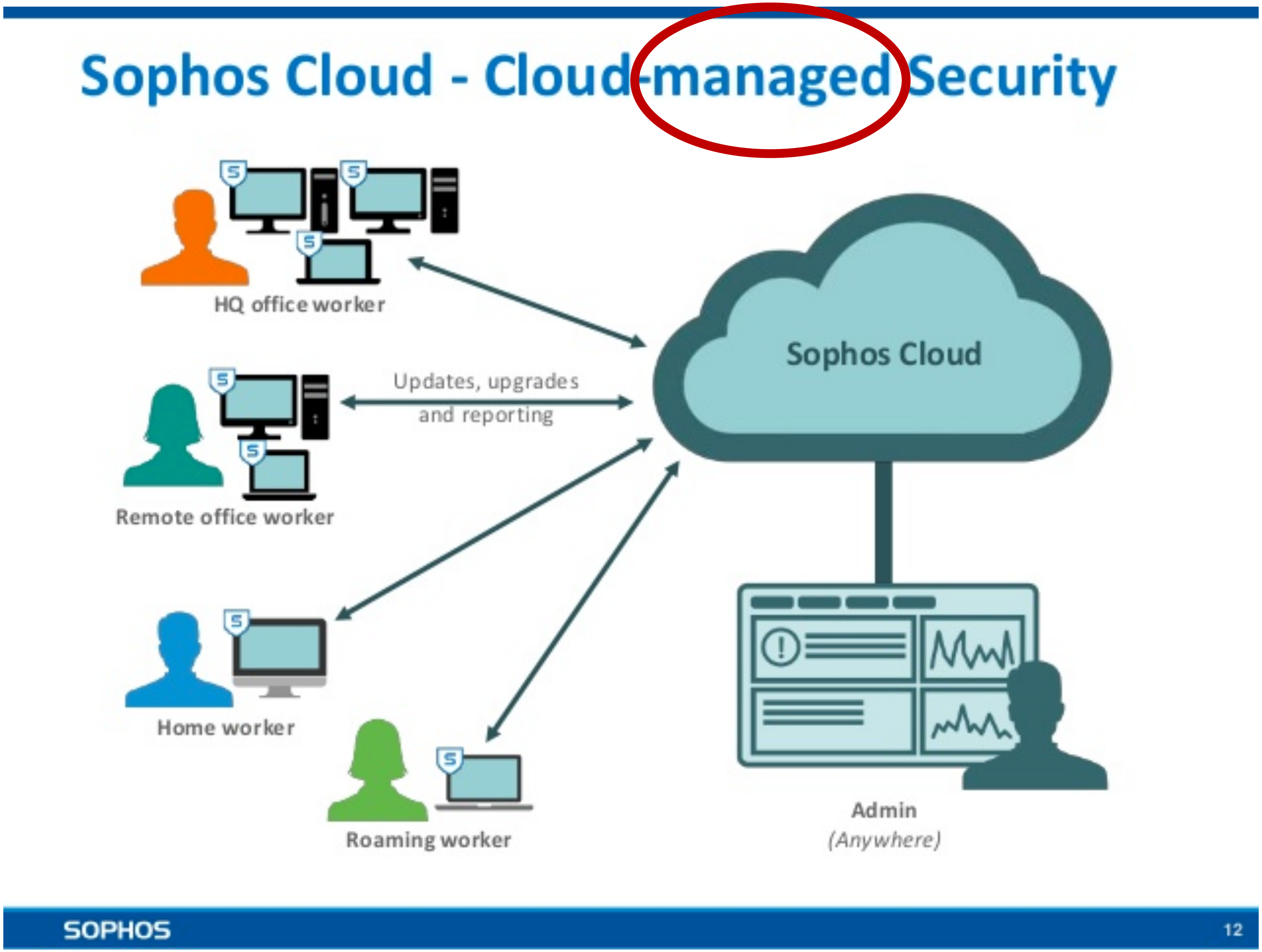
What have been done in cloud?

- Cloud anti-virus, e.g., Sophos and Symantec
- Protection of the cloud, e.g., VM sandboxing, [CloudDiag 2013]
- Software-as-a-sevice [Cloud Terminal 2012]

What have been done on host?

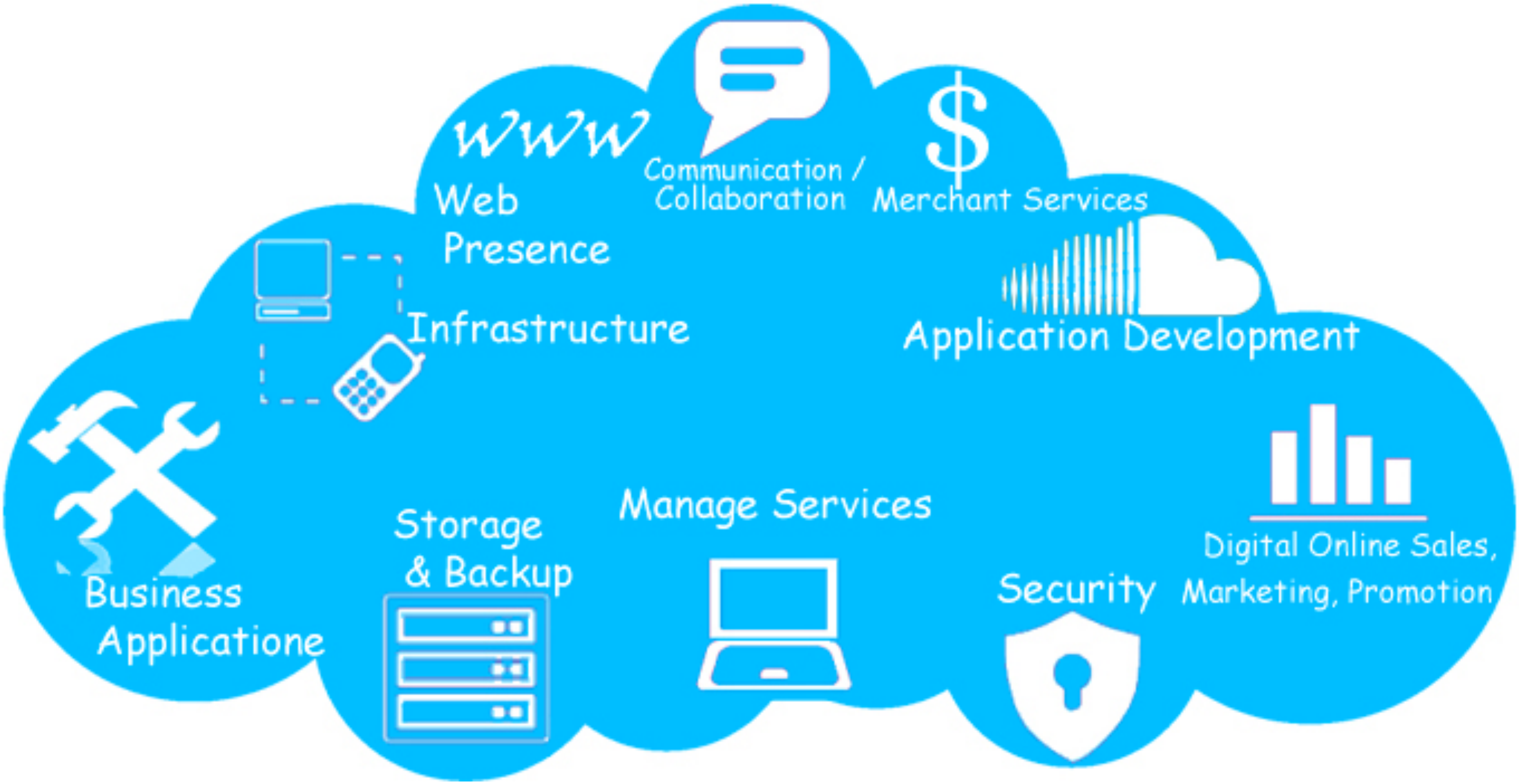
- Firewalls, host-based anti-virus
- Isolation, e.g., VMM
- Reference monitor, e.g., SELinux
- Trusted computing, e.g., TPM attestation
- **Data-driven anomaly detection**

Setup Type 1: the Cloud AV model



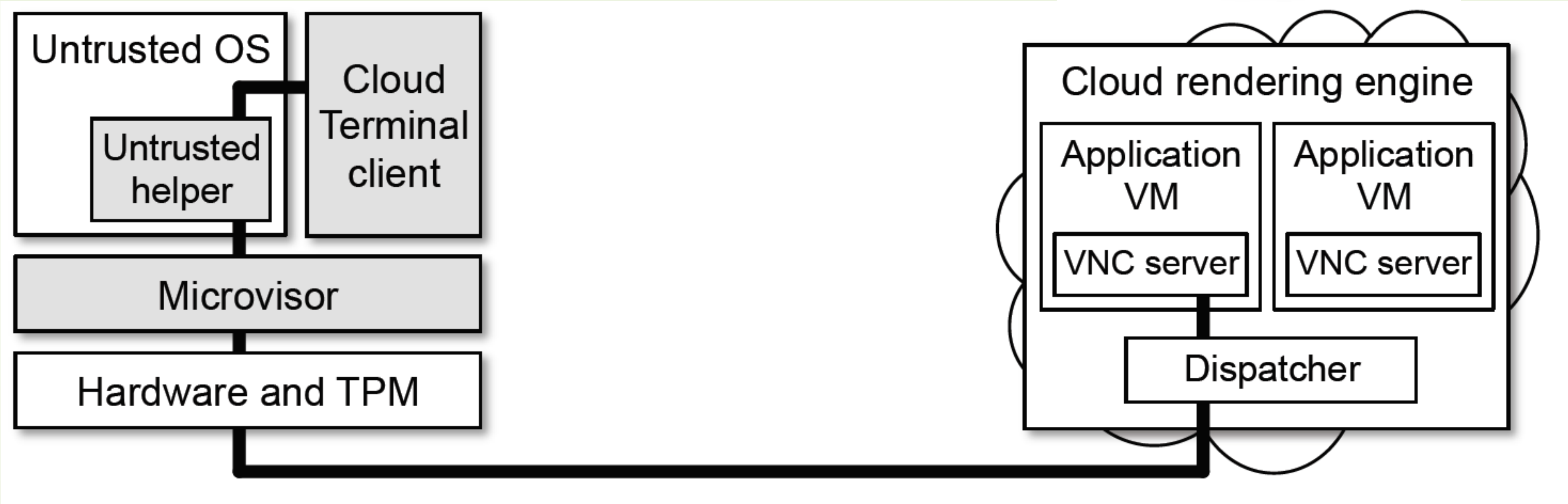
Setup Type 2: Everything in the cloud

[Gagzo.com]



https://www.comprompt.co.in/services/cloud-services/

Client



Cloud terminal [Martignoni 2012]

Setup Type 3: Your refrigerator cannot be in the cloud

WIRED Exclusive: Computer Virus Hits U.S. Drone Fleet

USINESS | CULTURE | DESIGN | GEAR | SCIENCE

NOAH SHACHTMAN SECURITY 10.07.11 1:11 PM

SHARE

SHARE

EXCLUSIVE: COMPUTER VIRUS HITS U.S. DRONE FLEET

Drone Control Station Operating System

http://theweek.com/article/index/241237/ (2011)

From NBC news (2013)

http://nbcnews.tumblr.com/post/47882129464#.UzGICChfd38

What does it take to lift program anomaly detection to the cloud?

In Setup Type 3:
autonomous host with detection in the cloud

Acknowledgments

Drs. Kui Xu (Google) · Xiaokui Shu (IBM Research) · Hao Zhang (Oracle)

Collaborators

- US Patent
- ACM CCS Tutorial 2016 on Program Anomaly Detection
- Work featured in Comm. of ACM

COMMUNICATIONS OF THE ACM

Network causal analysis

- Zhang, Yao, Ramakrishnan. ***AISec '16***, ***ASIACCS '14***, ***Computers & Security '16***

Global trace analysis

- Shu, Yao, Ramakrishnan. ***ACM CCS '15***
- Shu, Yao, Ramakrishnan, Jaeger (journal version under review)

Program analysis in HMM

- Xu, Yao, Ryder, Tian. ***IEEE CSF '15***

HMM with context

- Xu, Tian, Yao, Ryder. ***IEEE DSN '16***

Unified framework for program AD

- Shu, Yao, Ryder. ***RAID 2015***

Anti-virus Scanning is the First Line of Defense

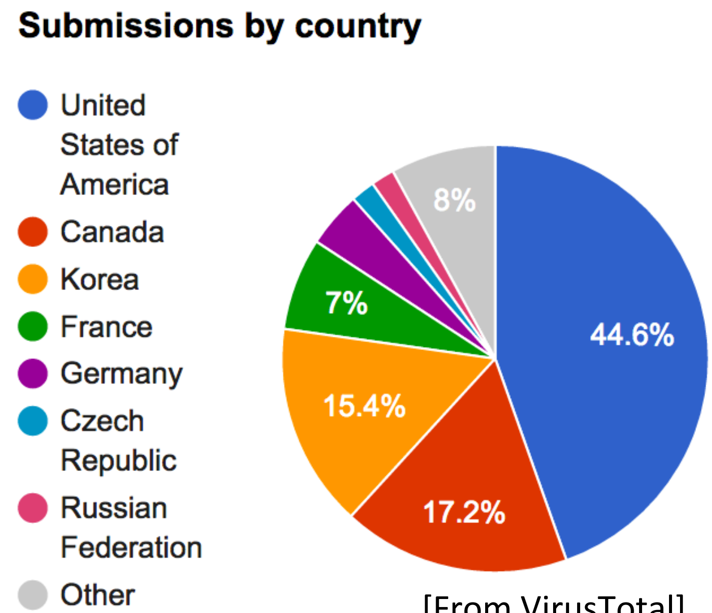
Vtzilla plugin

virustotal

For files (apps and PDFs), URLs

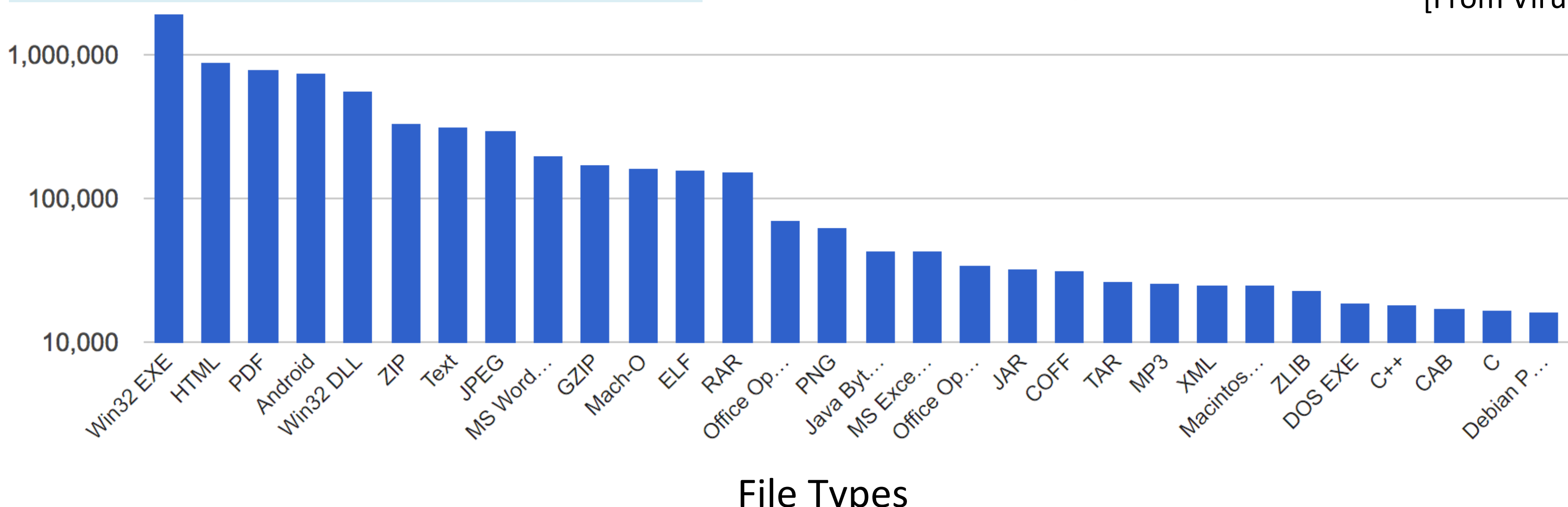
cuckoo

Cuckoo Sandbox for dynamic analysis



[From VirusTotal]

Number of submissions in a week
(March 19, 2017 – March 25, 2017)



[From VirusTotal]

Code or Behavior Classification is Undecidable

1. Program X
2. main()
3. { …
4. **if ! isVirus(X)**
5. **then infect;**
7. **else goto next;**
8. … }
9. }

Scanner Thinks		Actual Behavior of X
IsVirus returns True	Contradicts ➡	X chooses not to infect
IsVirus returns False	Contradicts ➡	X chooses to infect

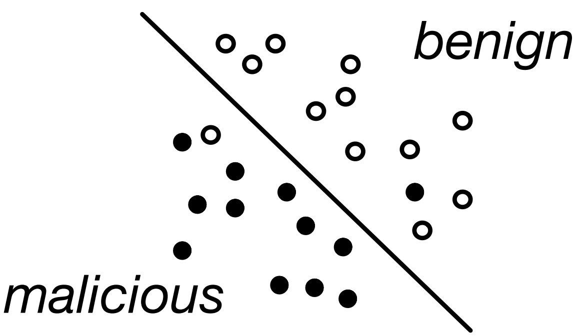
From [Fred Cohen, J. of Virology 1987]

How to detect/prevent zero-day malware/exploits?

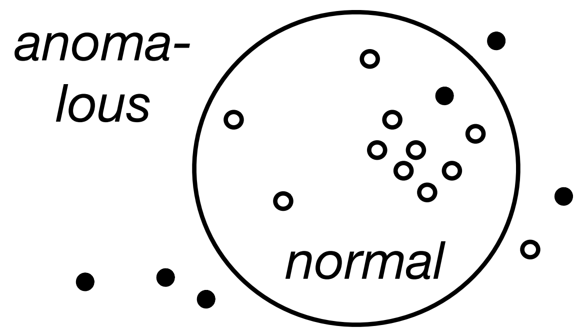
Formal verification, Control flow integrity

N-variant, Moving target defense

Anomaly-based detection [D. Denning ’87, Forrest et al. ’96]



(a) Classification



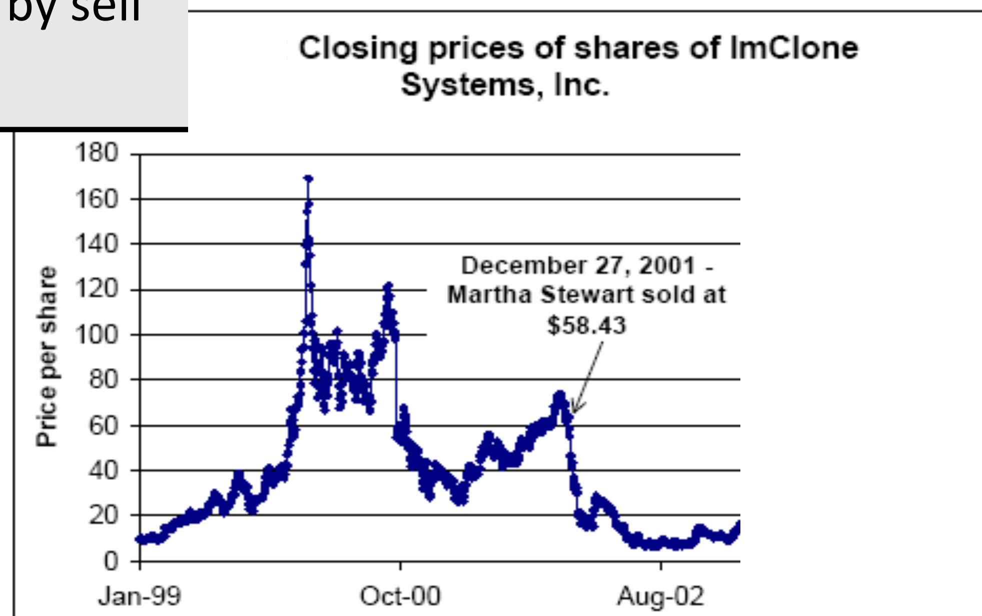
(b) Anomaly detection

[Wressnegger 2013]

Is Typical Insider Trading Detection Anomaly Detection?

Purchase Patterns	Sell Patterns
Buy low performing stocks	Sell high performing stocks
Buy before stock prices go up	Sell before stock prices drop
Purchase followed by purchase	Sell followed by sell

[Lorie 1968, Lakonishok 2001, Tamersoy 2014]

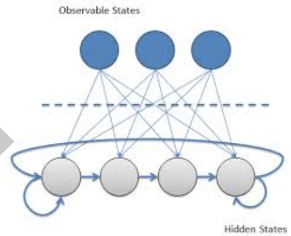


wigblog.blogspot.com/2004/07

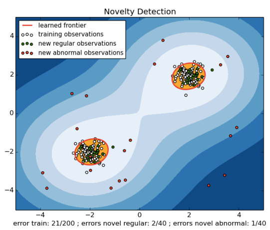
My Work on Anomaly Detection Methodology Development

Security logs,
Network headers,
Traffic payloads,
System traces,
Transaction logs
...

Prog analysis



Binary classification



Novelty detection

Program Tracing
(Library call,
System call,
Instruction sequences)

Program Analysis
(static)

ML/DM
(train and test)

Post Classification
Analysis

Simplest Program Anomaly Detection: n-gram

A 2-gram example:

ioctl()	open()
open()	read()
read()	setpgid()
setpgid()	setsid()
setsid()	fork()

1. From syscall traces of normal program executions (training data)

Runtime program trace

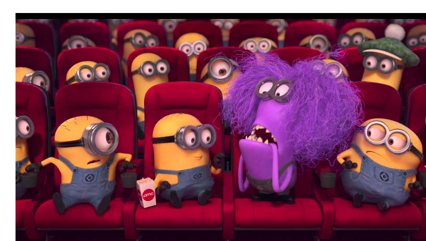
ioctl()
open()
write()
read()
setpgid()
setsid()
fork()

2. Test data

Runtime program trace	Found in DB?
Ioctl(), open()	✓
open(), **write()**	✗
write(), read()	✗
read(), setpgid()	✓
……	

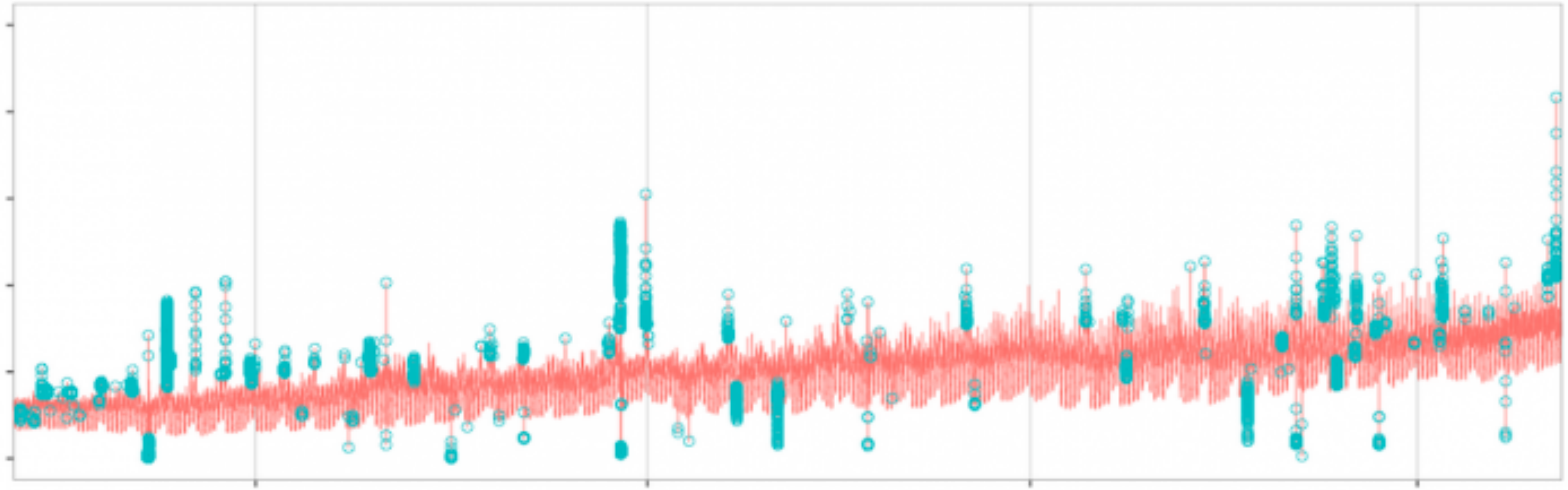
3. Classification

[Forrest 1996, Wressnegger 2013]



Who Uses Anomaly Detection on Programs/Systems?

- Average **$1.27million/year** on false alerts by an enterprise.
- **4%** of alerts are investigated, due to high false positives.
- An organization receives an average of **17,000 alerts/week**.

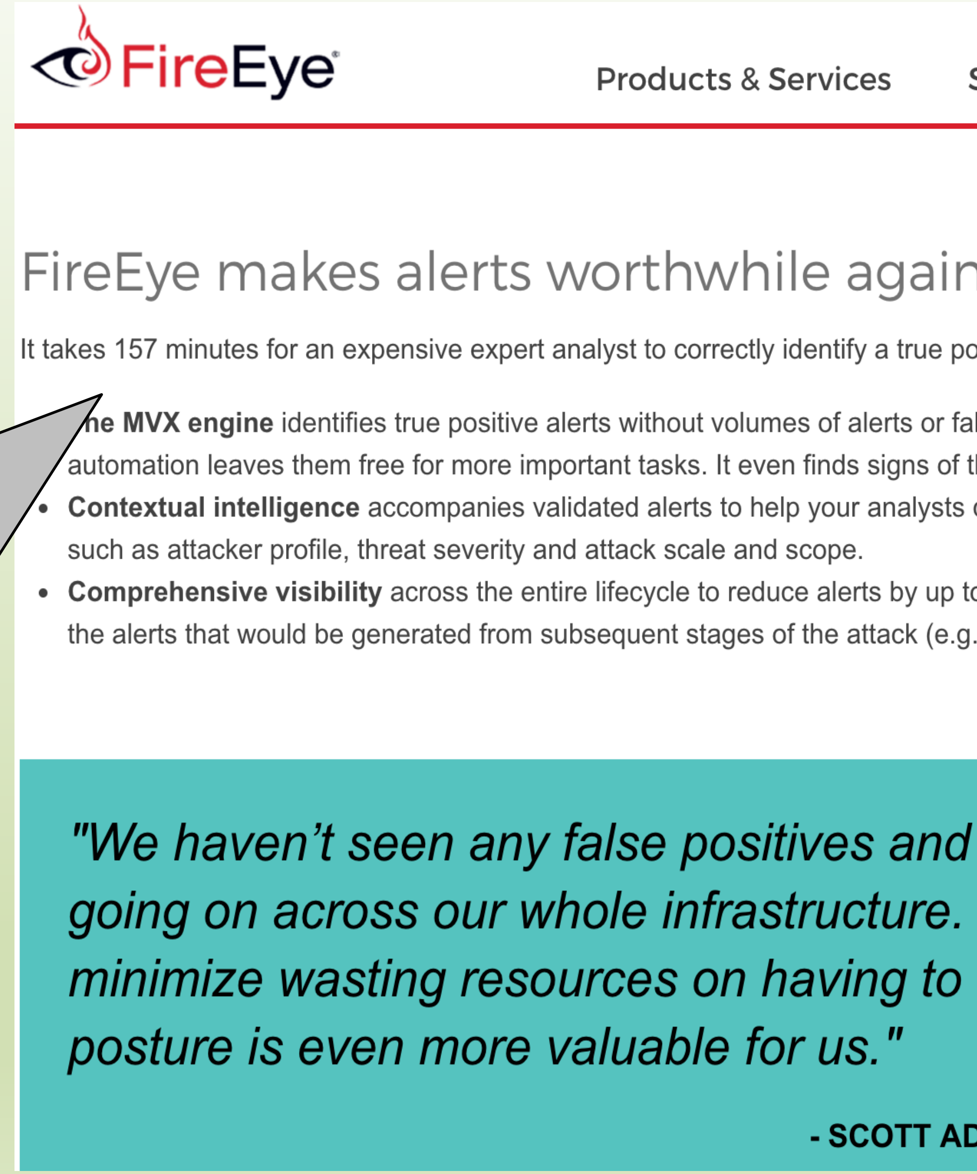


Twitter Anomaly Detection.
https://blog.twitter.com/2015/

From [Ponemon Institute]

Manual alert confirmation is costly

157 minutes



FireEye.com

Big Data, Big Bucks

twitter

NETFLIX

splunk>

LOGGLY

sumologic

NEXDEFENSE

ThetaRay

SCALYR

loglogic

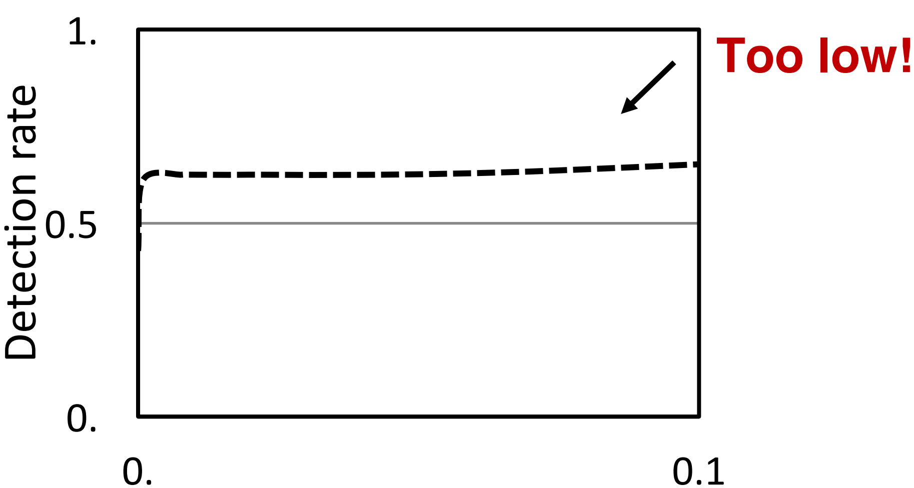
ALERT LOGIC®
Security. Compliance. Cloud.

graylog

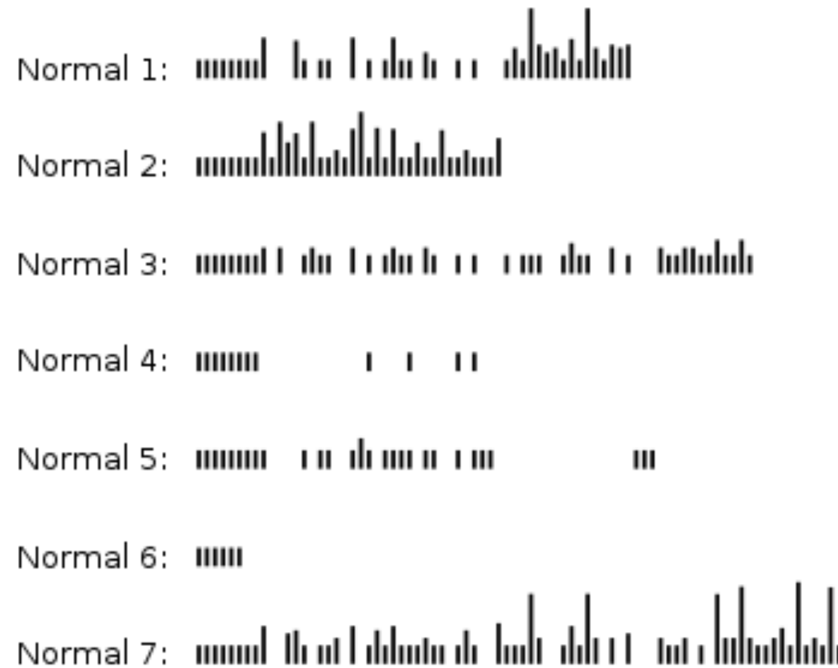
LogRhythm™
The Security Intelligence Company

elastic

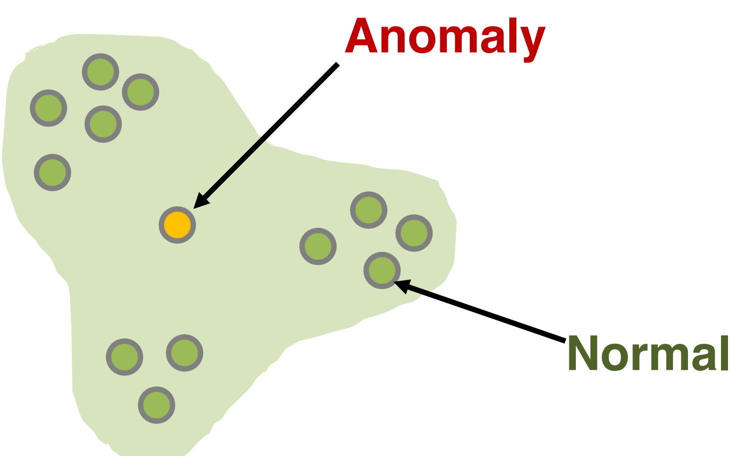
Challenges: Diverse Normal Behaviors, High FP



False positive rate
(1-class SVM on libpcre)



Distribution of function calls in libpcre



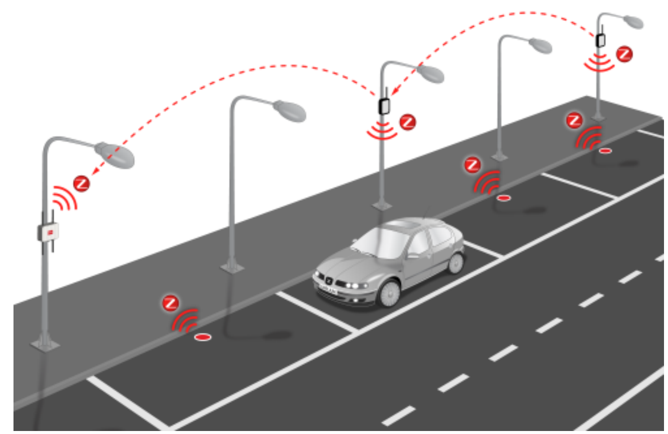
False alarms & missed detection can be harmful

Voice-recognition based authentication [CITI Taiwan]

NATIONAL CENTER FOR MISSING & EXPLOITED CHILDREN®
www.missingkids.com

Child pornography detection (FP 1 out of 2 billions)

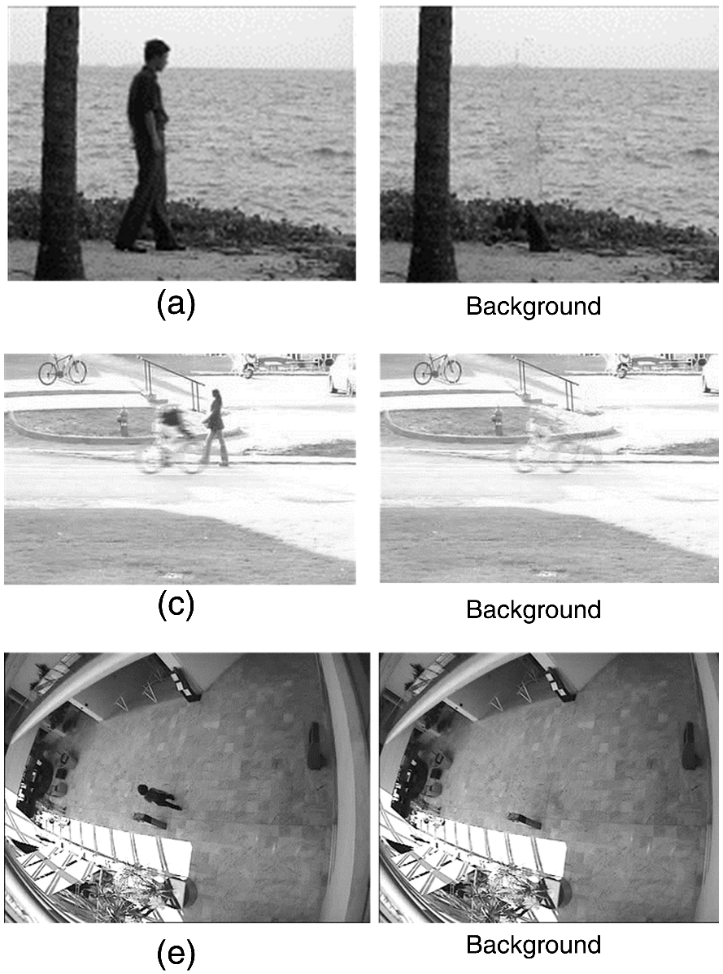
Spam detection



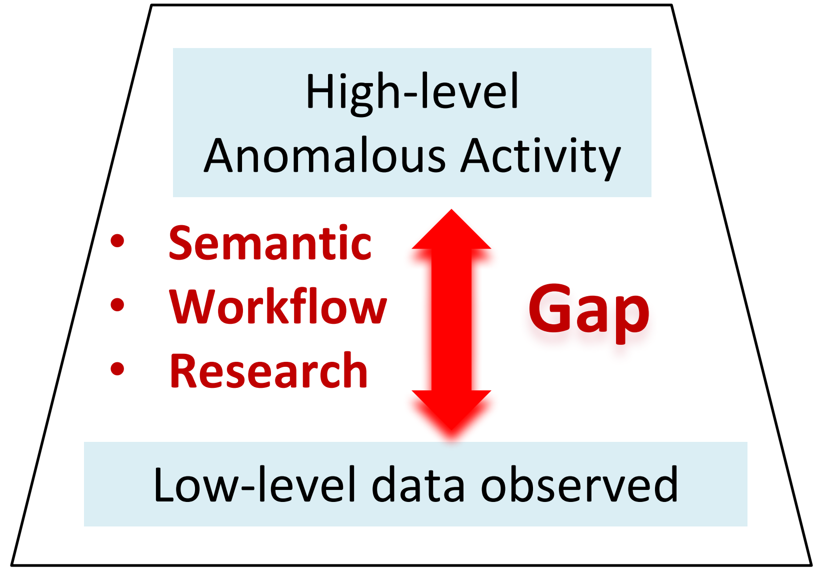
Pavement distress detection w/ sensors

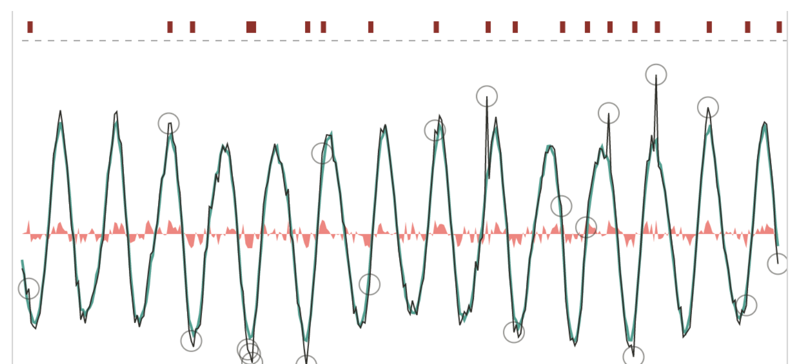
You found some weird data. Are they meaningful?

rPCA [Candès 2009] works well for motion detection in videos



Images from [Wang 2016]





[Netflix Robust Anomaly Detection]

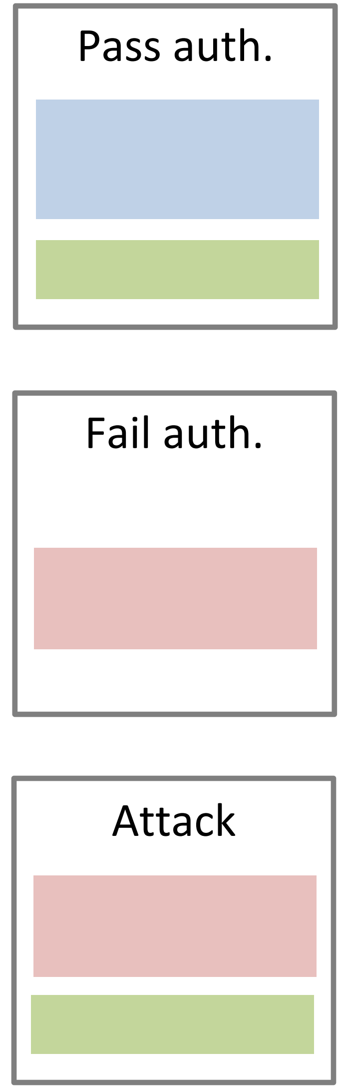
Semantics of Anomalies in Security

Actions of Attacks and Attack Preparations

- **Control-flow hijacking**
  - Return-oriented programming (ROP)
  - Backdoors
- **Control-flag hijacking**
  - Data-oriented programming (DOP) (not be detected by CFI)
- **Service abuse attacks**
  - Denial of Service (DoS)
  - Memory overread
- **Workflow/state violation**
  - E.g., bypass authentication
- **Exploit preparation**
  - Heap manipulation
  - Address space layout randomization (ASLR) probing

SSHD flag variable overwritten attack

```
void do_authentication(...) {
  int authenticated = 0;
  while (!authenticated) {
  […buffer overflow vulnerability…]
    if (auth_password(...)) {
      memset(...);
      xfree(...);
      log_msg(...);
      authenticated = 1;
      break;
    }
    memset(...);
    xfree(...);
    debug(...);
    break;
  }
  if (authenticated) {
    ...
```

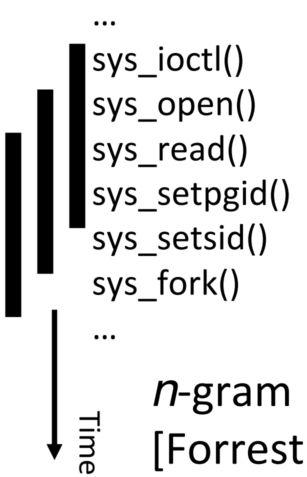


Expected

Expected

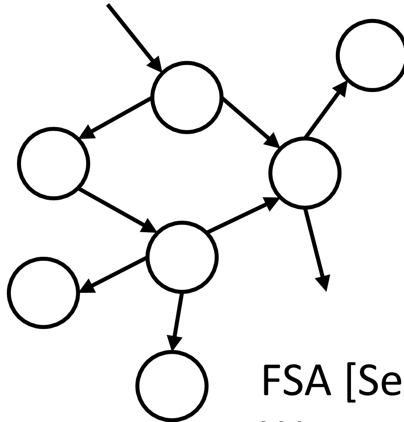
Local analysis cannot detect the anomaly

From [Chen '05]

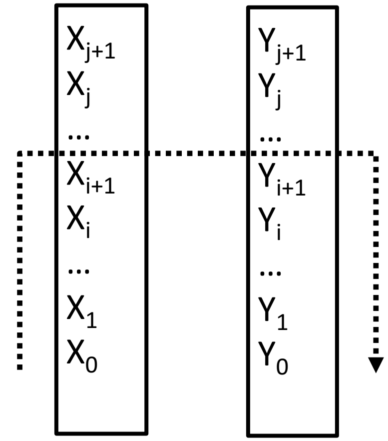


n-gram
[Forrest 1996]

[Forrest 2008]

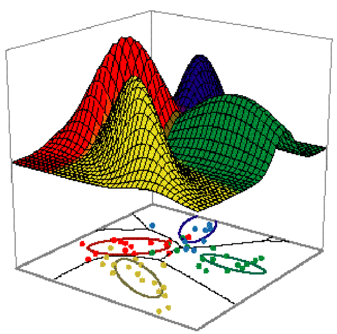


FSA [Sekar 2001, Wagner 2001]



PDA [Feng 2003, Feng 2004, Giffin 2004]

[Chandola 2009]



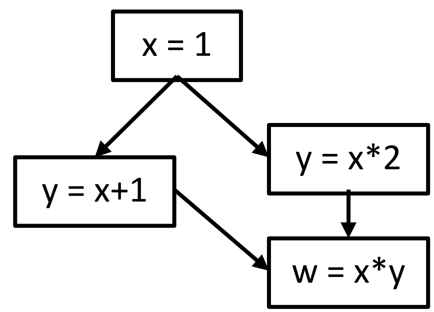
Machine learning [Lee 1998, Mutz 2006, Xu 2015, Xu 2016, Shu 2015]

[Wagner 2002]

Static Program Analysis
+
Dynamic Program Analysis

Hybrid detection [Gao 2004, Liu 2005]

[Feng 2004]



Data-flow analysis [Giffin 2006, Bhatkar 2006]

[Shu, Yao, Ryder. RAID 2015]

Old and New Challenges of Data-driven Anomaly Detection

Subtlety
- Stealthy attacks, e.g., ROP, DOP

Scale of Data
- Cloud support
- HPC
- Transparency

Experimental Reproducibility
- Security guarantees
- Benchmarks, baselines, open source

Definition of Anomalies
- Domain knowledge
- Inter-discipline
- Usability

Interpretation of Anomalies
- Semantic gap
- Meanings of anomalies
- Usability

Accuracy of Detection

Use 3 Host Protection Solutions as Examples

1: HMM-based local anomaly detection

2: Global trace analysis for frequency anomalies

3: Triggering relation discovery of system and network events

How to Lift Host Protection to the Cloud?

Issue 1: Incomplete Traces

Incomplete Traces

Incomplete Behavior Model

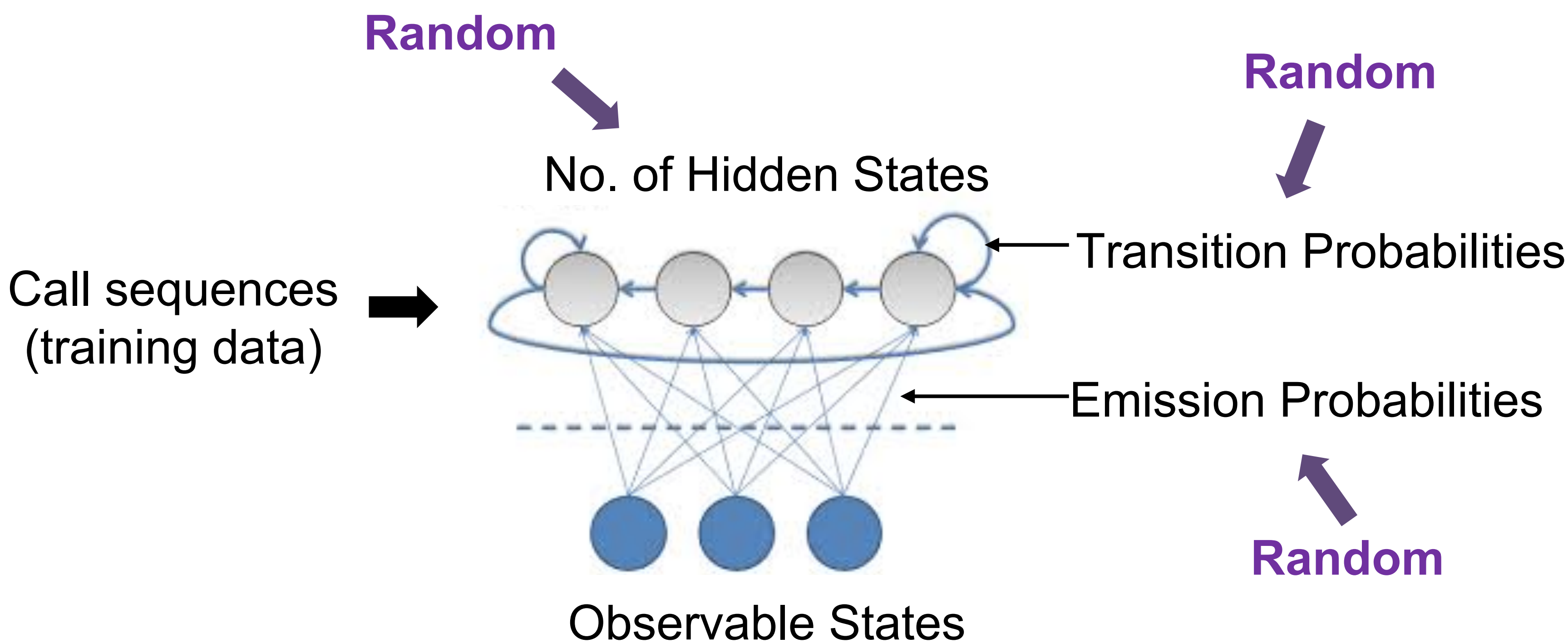
False Alarms
Missed Detection

Program	# of test cases	branch coverage	line cov.
flex	525	81.34%	76.04%
grep	809	58.68%	63.34%
gzip	214	68.49%	66.85%
sed	370	72.31%	65.63%
bash	1061	66.26%	59.39%
vim	976	54.99%	51.93%

From SIR

By Shel Silverstein

How to do make HMM smarter in anomaly detection?

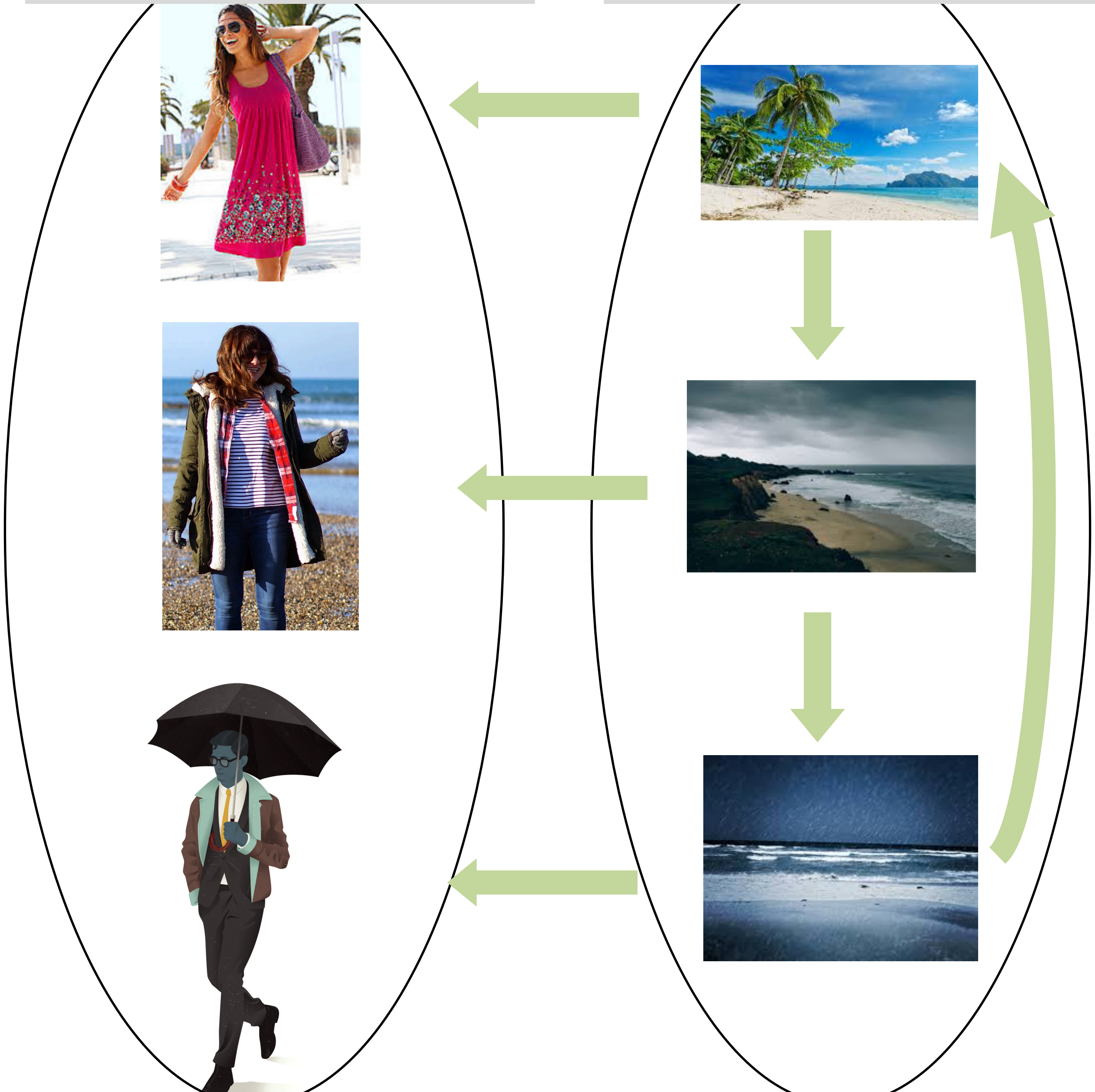


Better HMM initialization based on programs

Program analysis for HMM
- Xu, Yao, Ryder, Tian. ***IEEE CSF ’15***

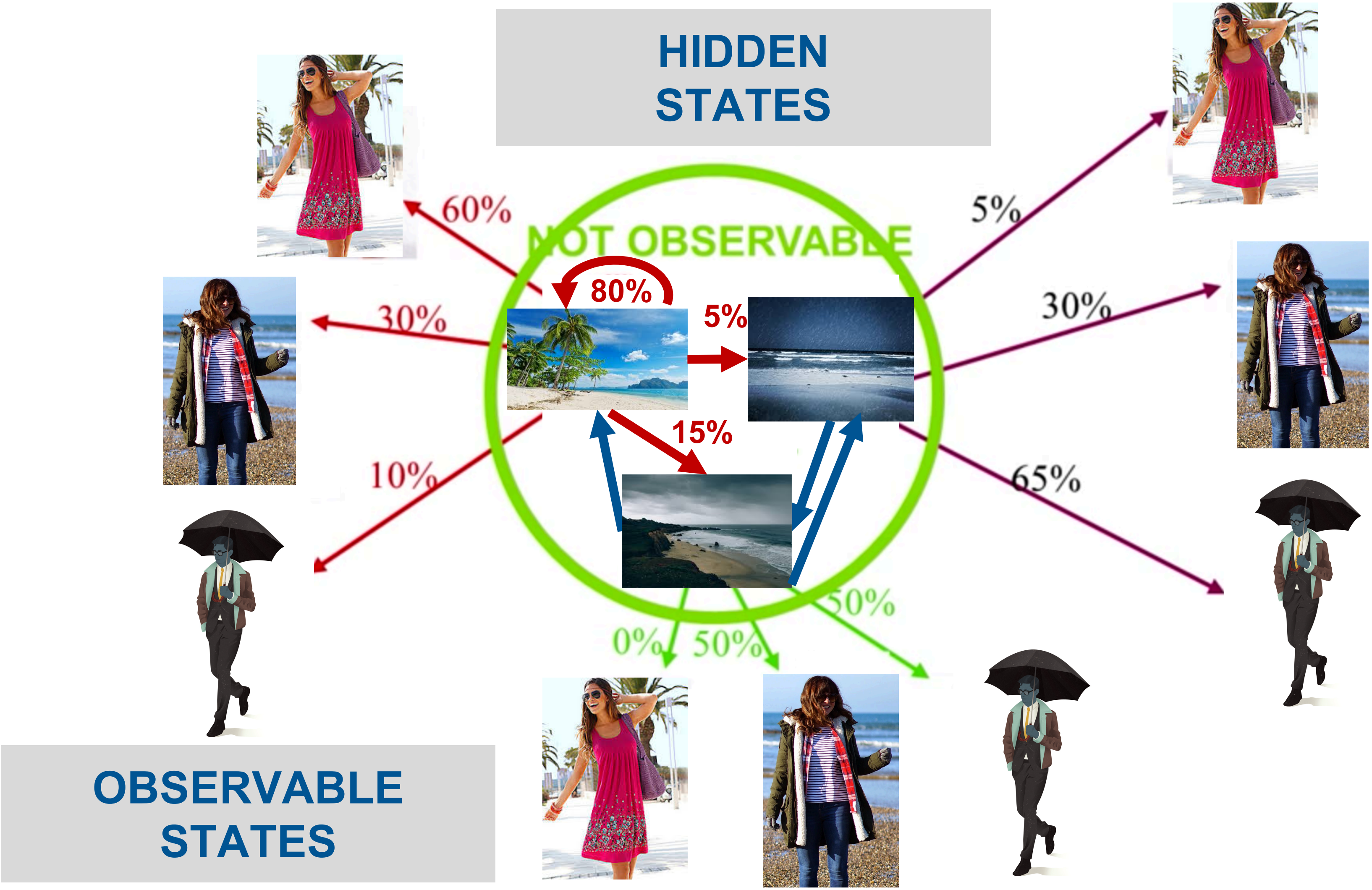
HMM with context
- Xu, Tian, Yao, Ryder. ***IEEE DSN ’16***

OBSERVABLE STATES | **HIDDEN STATES**



Hidden Markov Model (HMM)

Markov process (memoryless) where some states are not observable



[Adapted from Udacity]

HMM-based Program Anomaly Detection

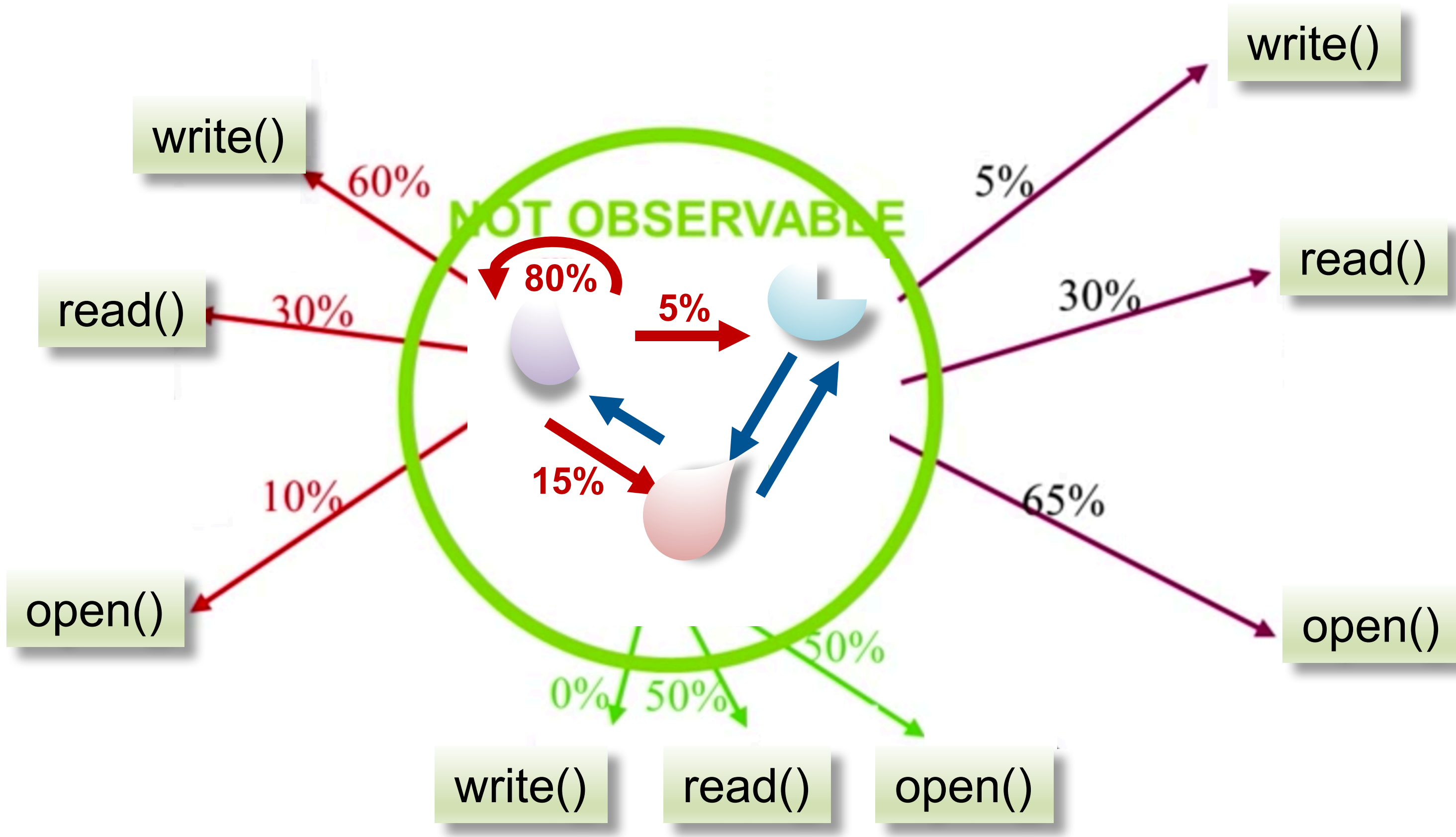
Probabilistic, Path sensitive, Local analysis, Semi-supervised training

[Forrest et al. 1999]

write()
read()
ioctl()
open()
write()
read()
setpgid()
setsid()
setsid()
fork()
setpgid()
setsid()
fork()

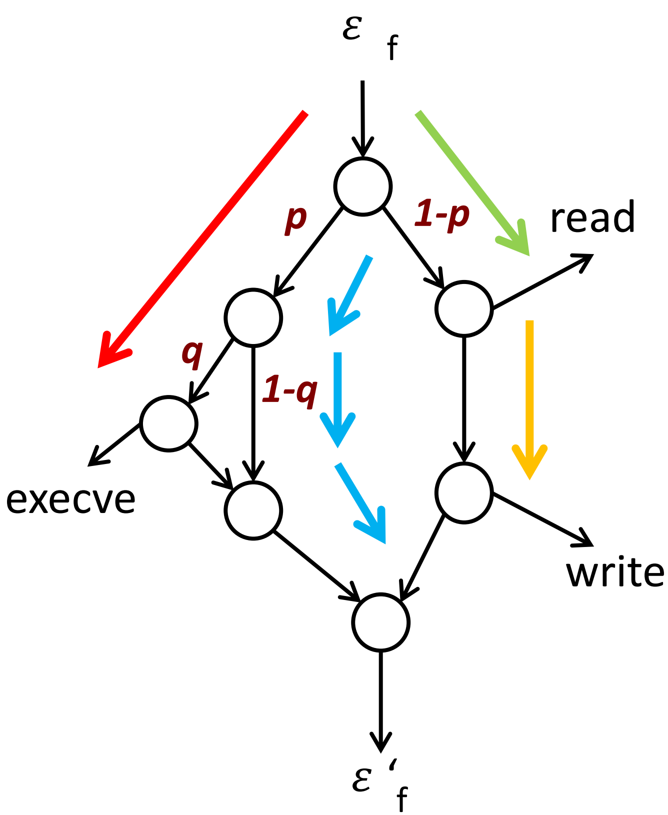
ioctl()
open()
write()
read()
ioctl()
open()
write()
read()
setpgid()
ioctl()
open()

TRAINING DATA



Can we do better than random initialization?

STILO: STatically InitiaLized markOv



Function: f

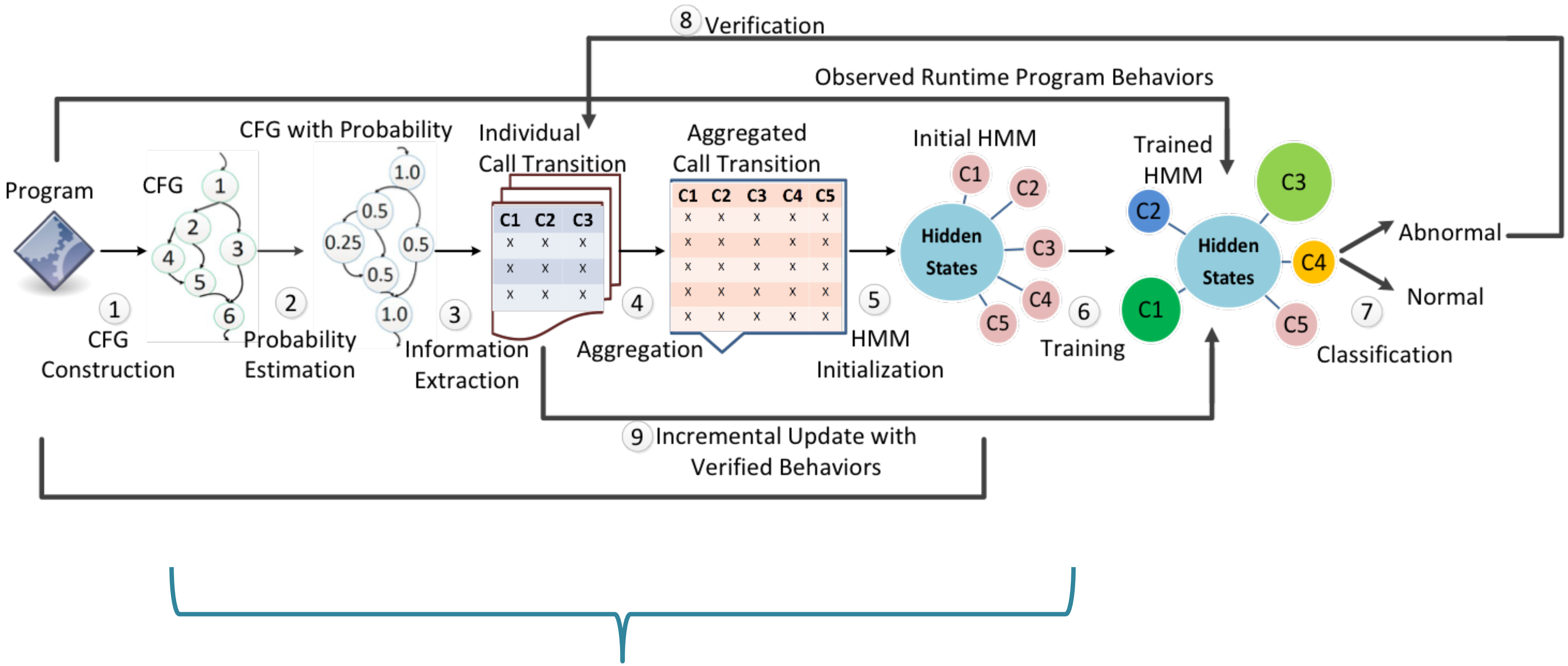
Transition probability of a call pair is its likelihood of occurrence during the execution of the function

Example of call pair	Transition probability
read $\longrightarrow$ write	1-p
read $\longrightarrow$ read	0
execve $\longrightarrow$ ε_f'	pq

	ε_f'(exit)	read	write	execve
ε_f (entry)	p(1-q)	1-p	0	pq
read	0	0	1-p	0
write	1-p	0	0	0
execve	pq	0	0	0

p, q are statically estimated.

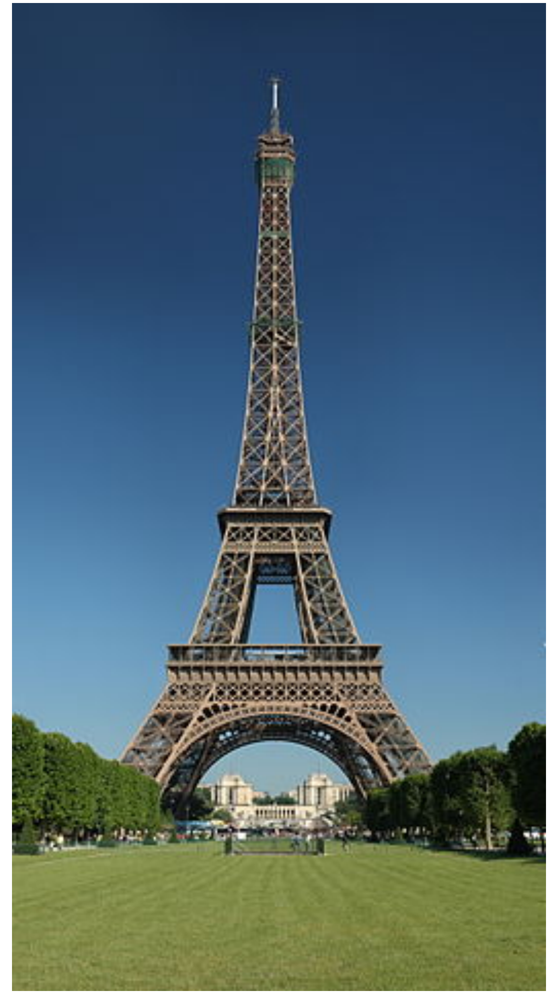
Host Security Solution 1:
Local Anomaly Detection with STILO



Static Program Analysis based HMM Initialization (New Contributions)

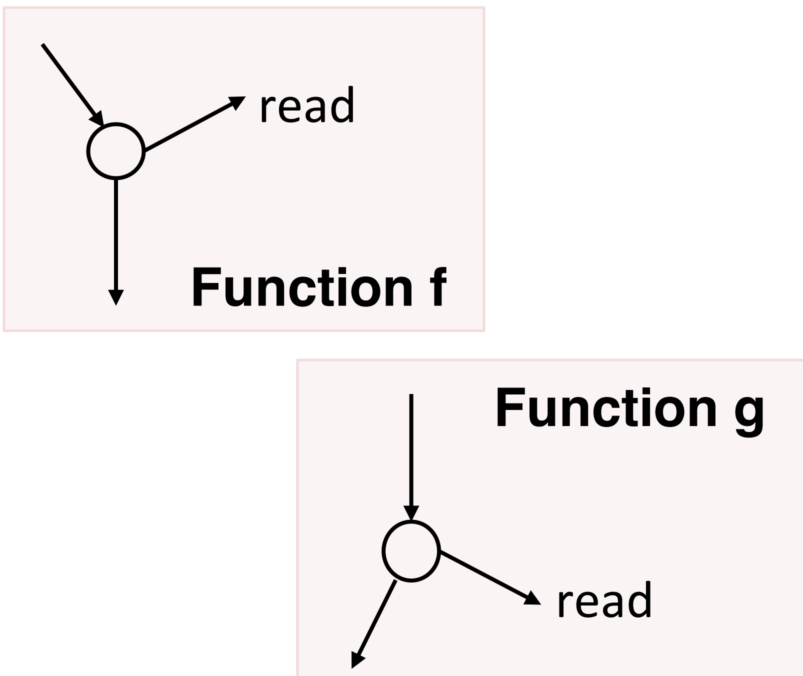
Improvement with Context Sensitivity

Why need context sensitive detection?

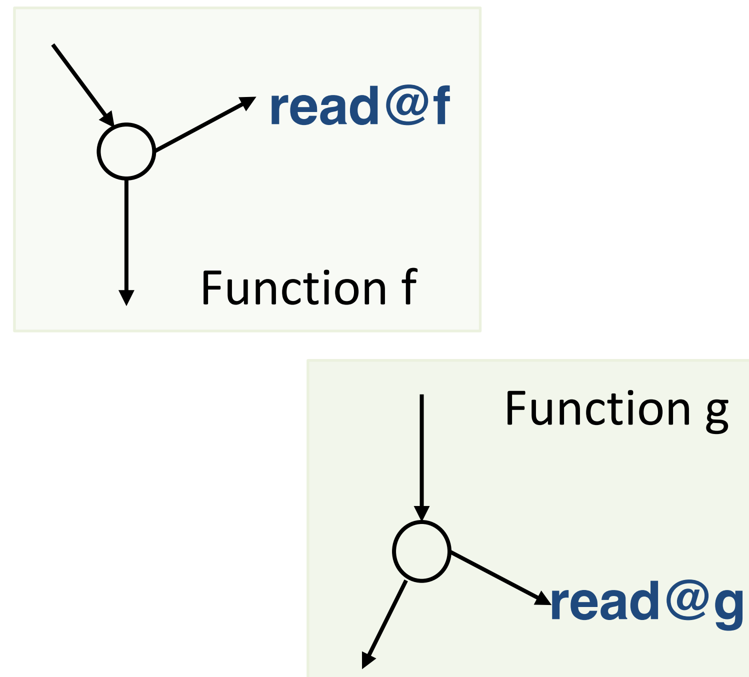


Improvement with Context Sensitivity

BEFORE: Context insensitive
(**STILO-basic**)



AFTER: 1-level calling context sensitive
(**STILO-context**)



… read …. read ….

… **read@f** …. **read@g** ….

Scalability:
K-mean clustering reduces the # of hidden states

[Xu, Tian, Yao, Ryder. *IEEE DSN* '16]

Reduction of Hidden States for Efficiency

Before clustering

One-to-one mapping -- a hidden state represents a single call

After clustering

Many-to-one mapping -- a hidden state may represent multiple similar calls

Program Model	# distinct calls	# states after clustering	Estimated training time reduction
bash	1366	455	88.91%
vim	829	415	74.94%
proftpd	1115	372	88.87%

- K-mean clustering, based on similarity between call-transition vectors
- Aim at 1/2 to 1/3 reduction of nodes

STILO Evaluation

Model	With Static Analysis	With Caller Context
Regular-basic	-	-
Regular-context	-	Yes
STILO-basic	Yes	-
STILO-context	Yes	Yes

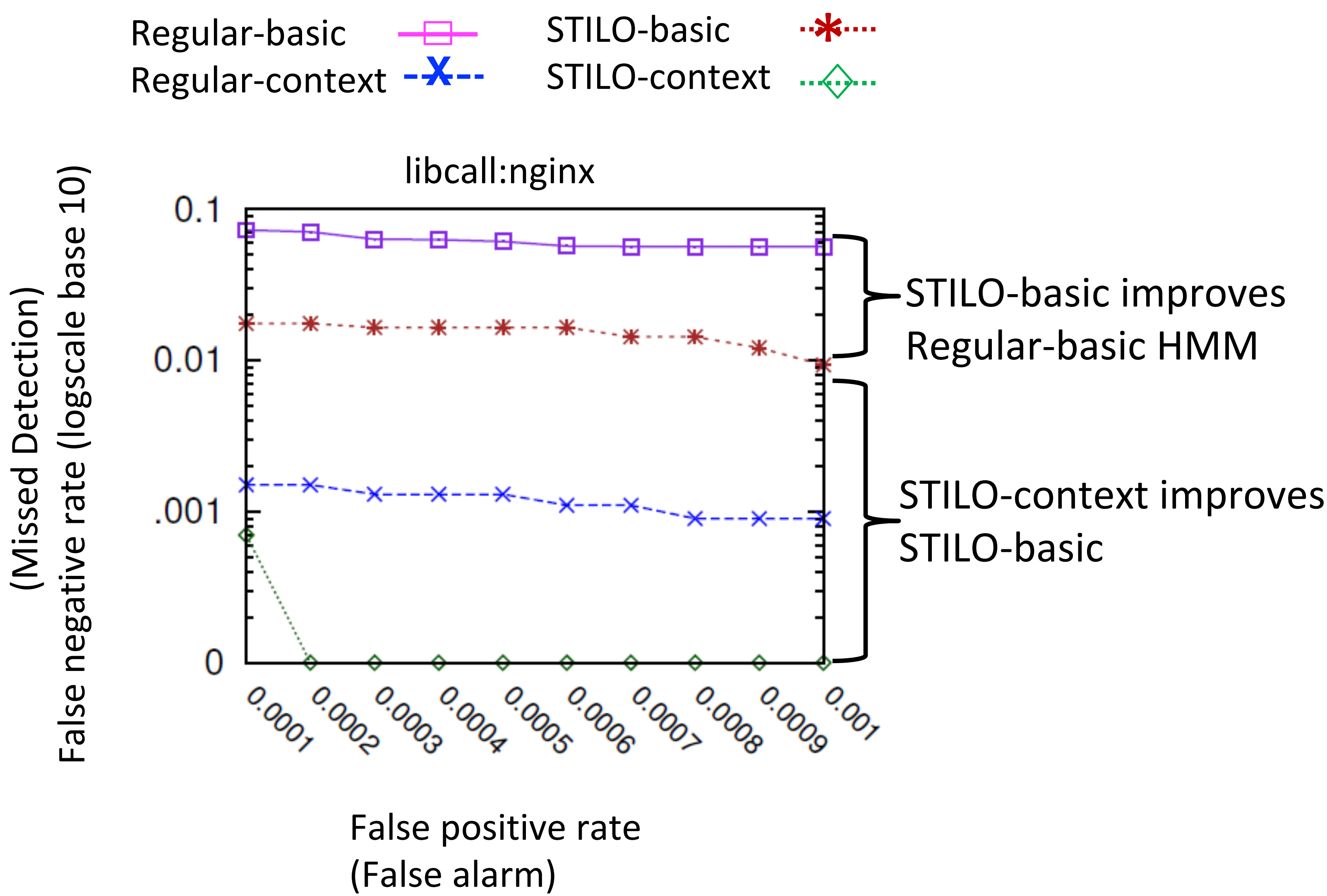
2 Linux server programs: nginx, proftpd

6 Linux utility programs: flex, grep, gzip, sed, bash, vim

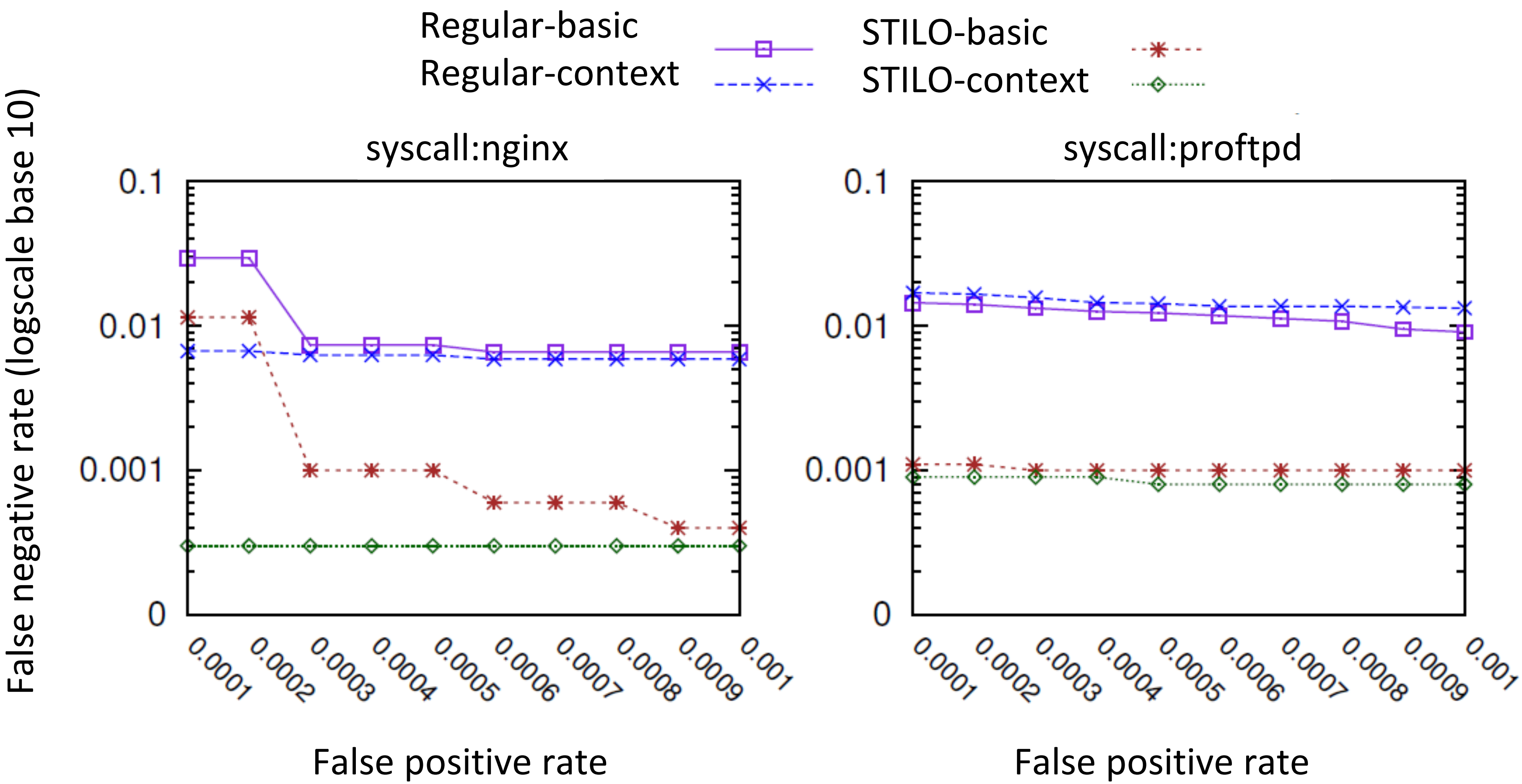
1. **Normal:** total 130,940,213 segments
2. **Abnormal-S:** 160,000 Abnormal-S segments (permute 1/3 calls)
3. **Abnormal-A:** attack call sequences obtained from exploits

Dynist for static program analysis, Jahmm library for HMM, 1st-order Markov, strace/ltrace for collection, SIR for test cases, 10-fold cross validation, 15-grams from traces

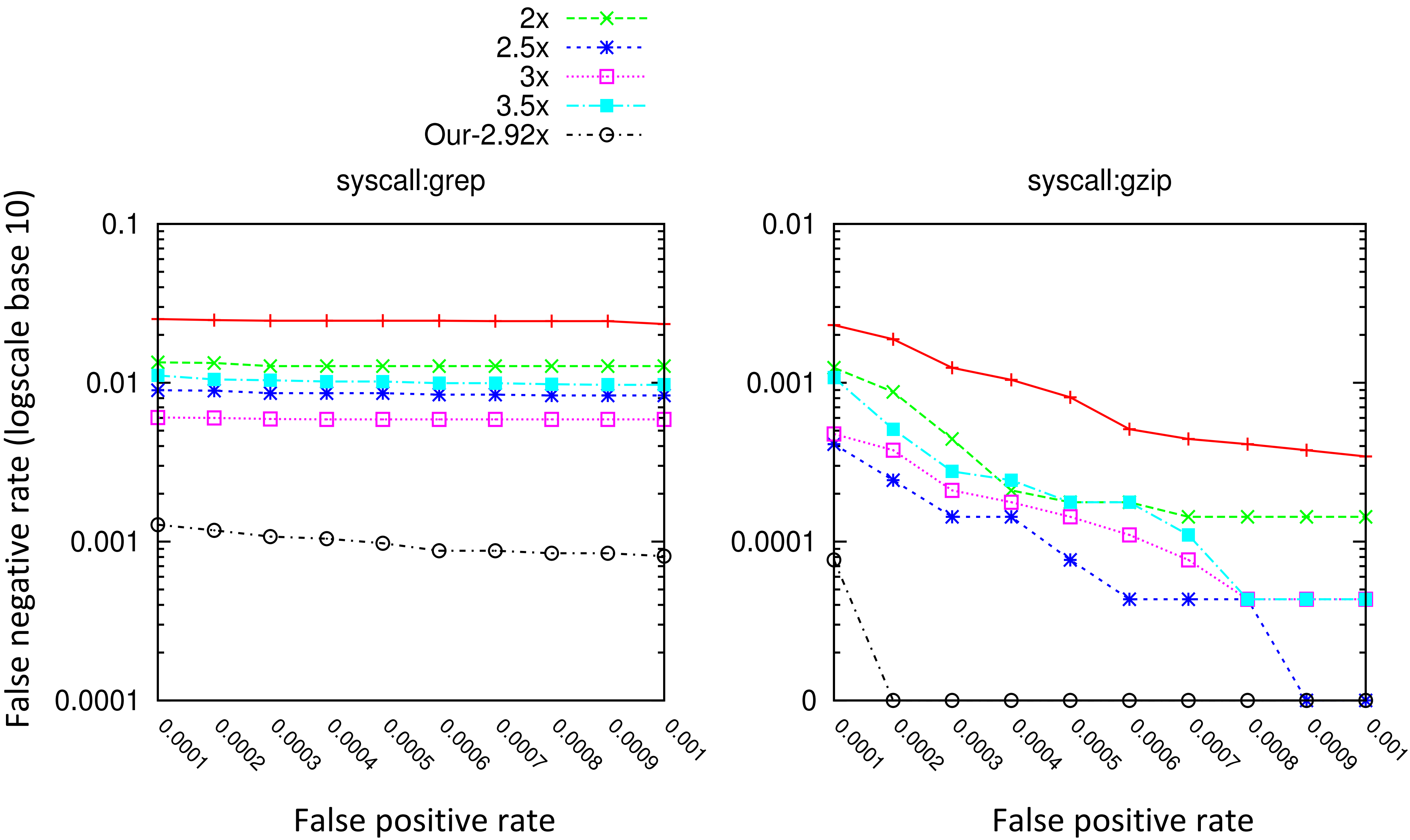
For libcalls, false negative (missed detection) of context-sensitive models drops by 2-3 orders



For syscalls, context improves false negative rate by 10 folds.
Less dramatic improvement than libcalls.



Increasing hidden states in regular HMM does not guarantee classification accuracy



Detection of Real-world Attacks

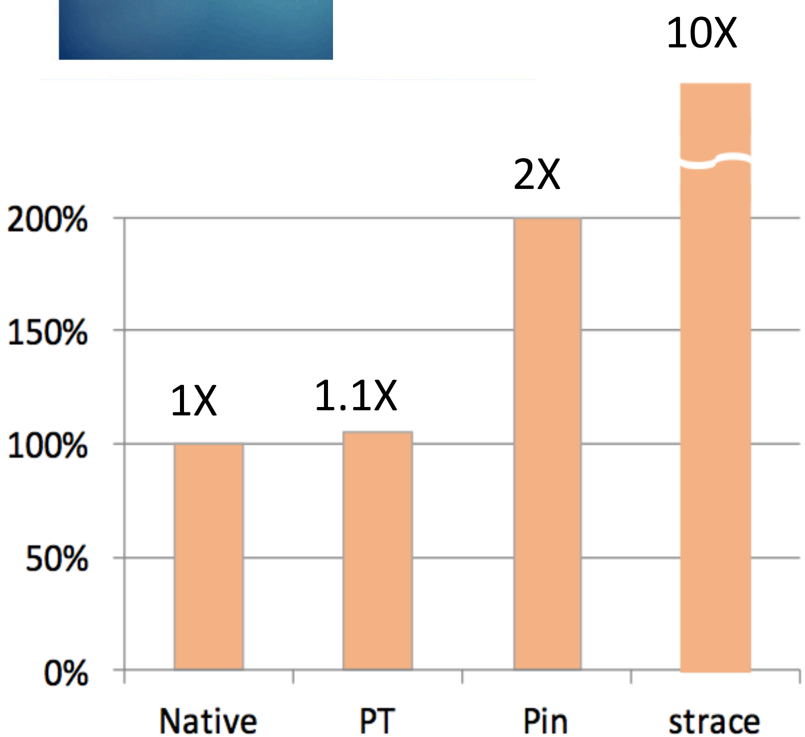
ROP attack segments against gzip (syscalls)

ID	Prob in STILO	Prob in Regular HMM
S_1	0	0.2
S_2	$2.20 \times e^{-15}$	0.29
S_3	$1.54 \times e^{-5}$	0.25
S_4	0	0.27
S_5	0.0005	0.33
S_6	0	0.23
S_7	0.0004	0.26

STILO gives much lower probabilities for attack sequences

Exploit	Payload
Buffer Overflow (gzip)	ROP
	ROP_syscall_chain
Backdoor (proftpd)	bind_perl
	bind perl ipv6
	generic cmd execution
	double reverse TCP
	reverse_perl
	reverse_perl_ssl
	reverse_ssl_double_telnet
Buffer Overflow (proftpd)	guess memory address

Ongoing Work: Hardware-assisted Program Tracing for Anomaly Detection



A control block of libc library
7ffff7a54b01 libc.so <__libc_start_main+177>

A control block for main function
400506 a.out <main+0>
4003e0 a.out <puts@plt+0>

A control block from loader to resolve call
7ffff7df02f0 ld.so <_dl_runtime_resolve+0>

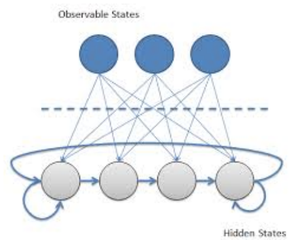
In collaboration with Trent Jaeger (PSU)

Performance and Ease of Deployment

Training Traces (host)

Probability forecast

HMM init & training



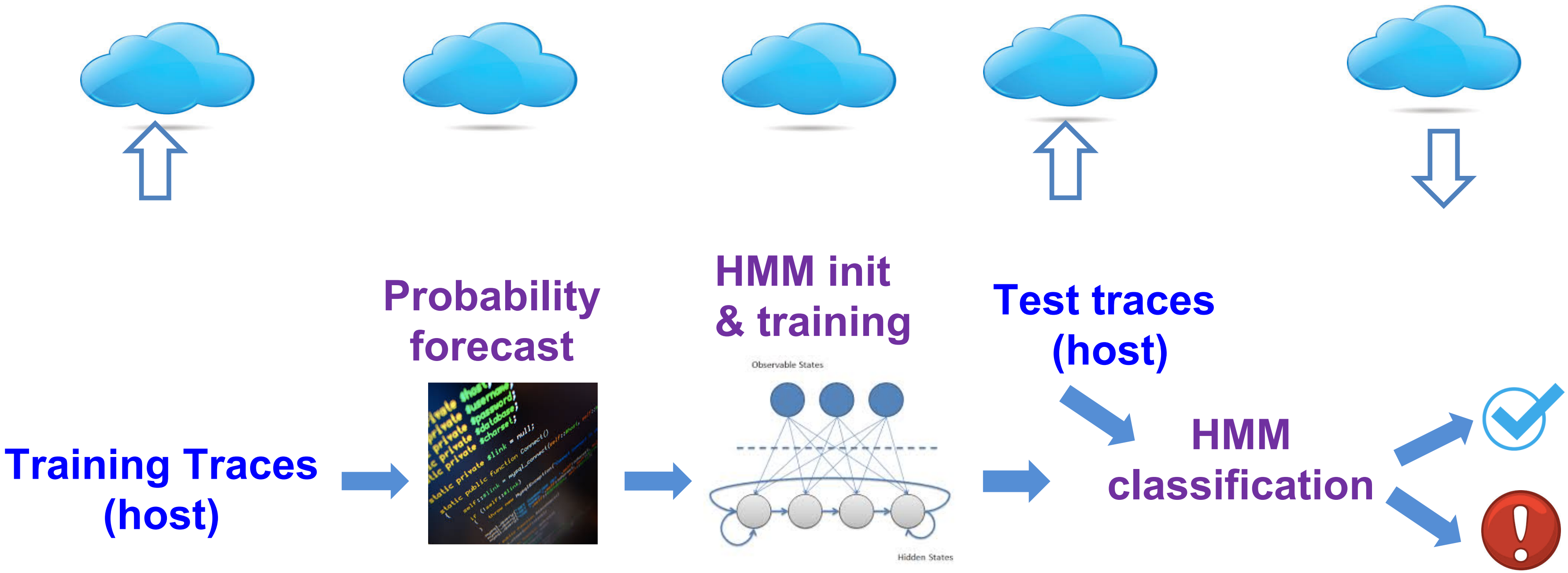
Test traces (host)

HMM classification

Could be Fast	Fast and slow	Painfully slow	Extremely fast

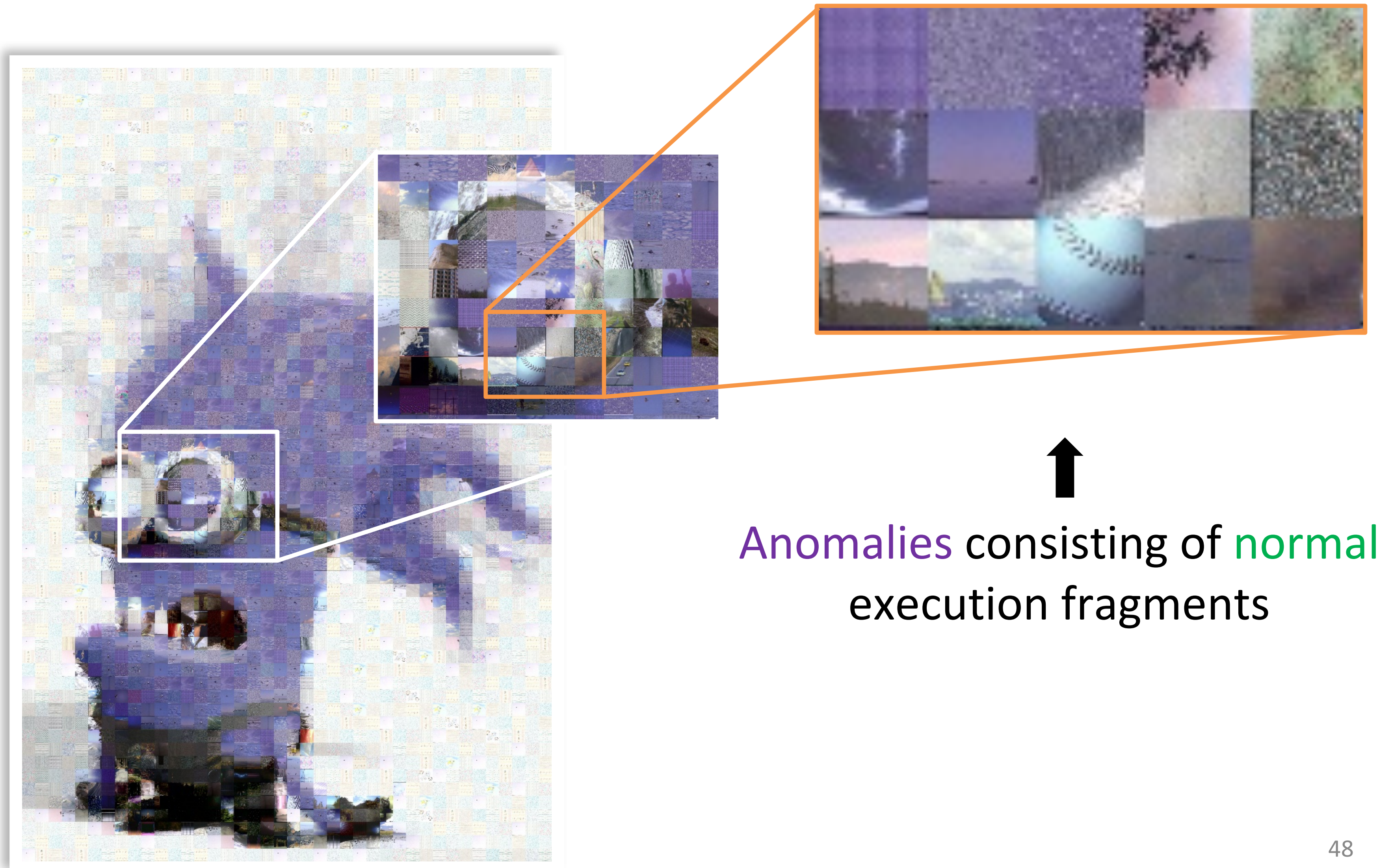
Moderate	Not easy	Not easy to set up	Moderate

What does it take to outsource STILO detection to the cloud?



Issue 2: Local Analysis

Local analysis is inadequate



Anomalies consisting of normal execution fragments

Attack Model, Problem Statement

Cooccurrence Anomaly

Normal 1: a b d a c e a

Normal 2: c b e a c c e c f

Normal 3: f d c e c c f e d

Anomaly: a b d a c c f e d

Attack examples:

- Non-control data attack
- Fragment-based mimicry attack
- Workflow violation attack

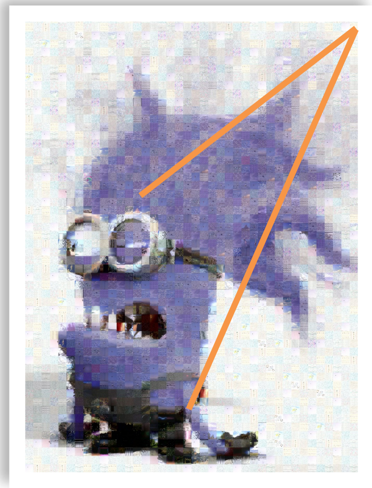
Frequency Anomaly

Attack examples:

- DoS attacks
- Directory harvest attacks

Problem Statement:

- Given an **extremely long trace**, should **any** set of events co-occur?
- With the expected **frequency**?



Can n-gram still work?

Host Security Solution 2: Global Anomaly Detection

An infinite long call trace:

… bar, main, foo, bar, bar, …

chop into

Behavior instance

Long trace segments

Behavior instance

convert into

1. Transition frequency matrix

	main	foo	bar	goo
main	0	24	0	0
foo	0	0	30	0
bar	2	6	89	1
goo	0	0	0	0

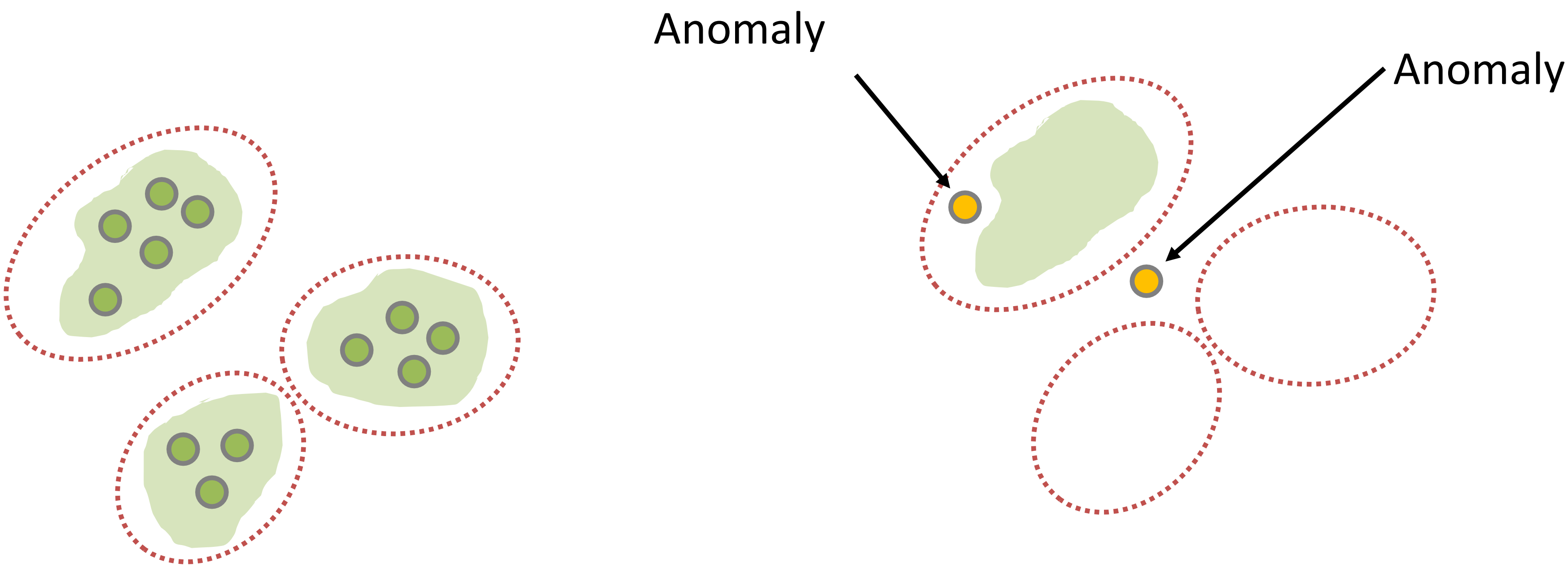
2. Event co-occurrence matrix

F	T	F	F
F	F	T	F
T	T	T	T
F	F	F	F

Matrix representation is path insensitive

[Shu, Yao, Ramakrishnan. *ACM CCS '15*]

Our Solution: Grouping Similar Normal Behaviors



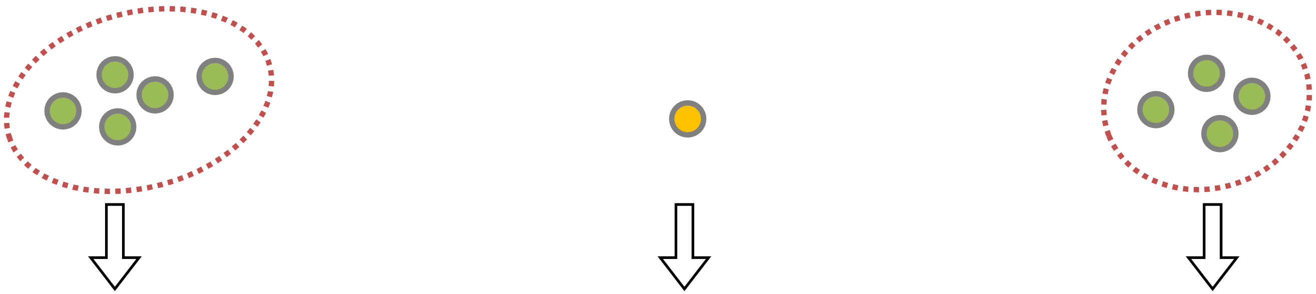
Training Phase

Detection Phase

A trace segment represented by matrices

Montage Anomalies Fall Between Clusters

sshd



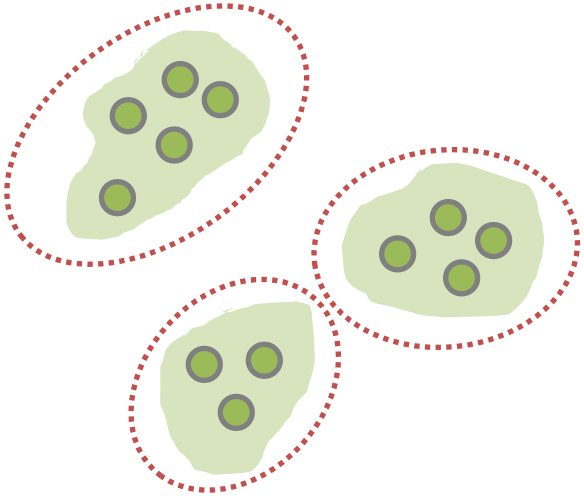
Pass Auth. (expected)	Anomalous: attack	Fail Auth. (expected)
…	…	…
do_auth > xfree	do_auth > debug	do_auth > debug
do_auth > log_msg	do_auth > xfree	do_auth > xfree
do_auth > packet_start	do_auth > packet_start	do_auth > packet_start
…	…	…
pwrite > buffer_len	pwrite > buffer_len	pwrite > buffer_len
do_auth > do_auth	do_auth > do_auth	do_auth > pread
…	…	…

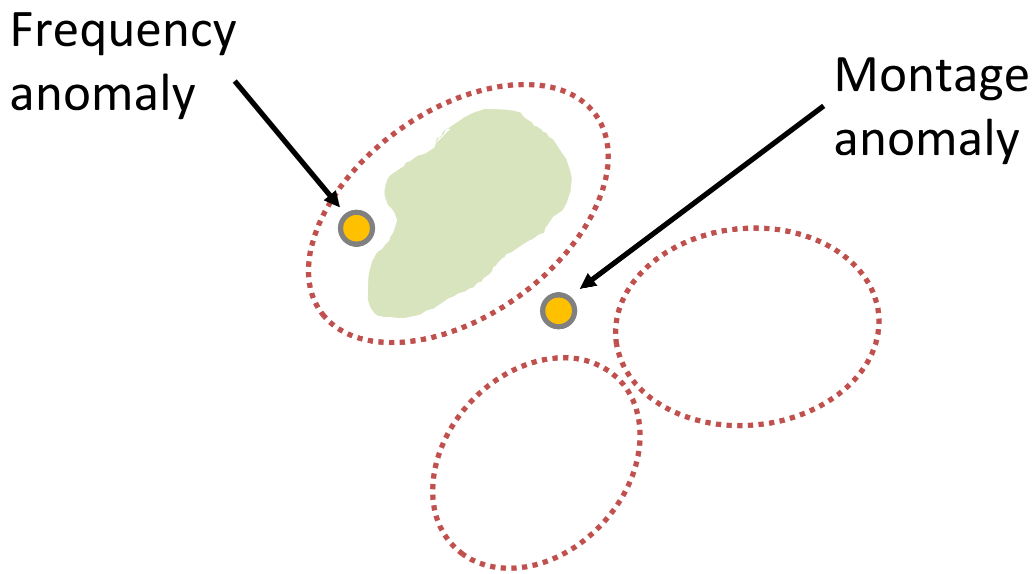
Function call trace
(collected through Pintool)

Our Operations

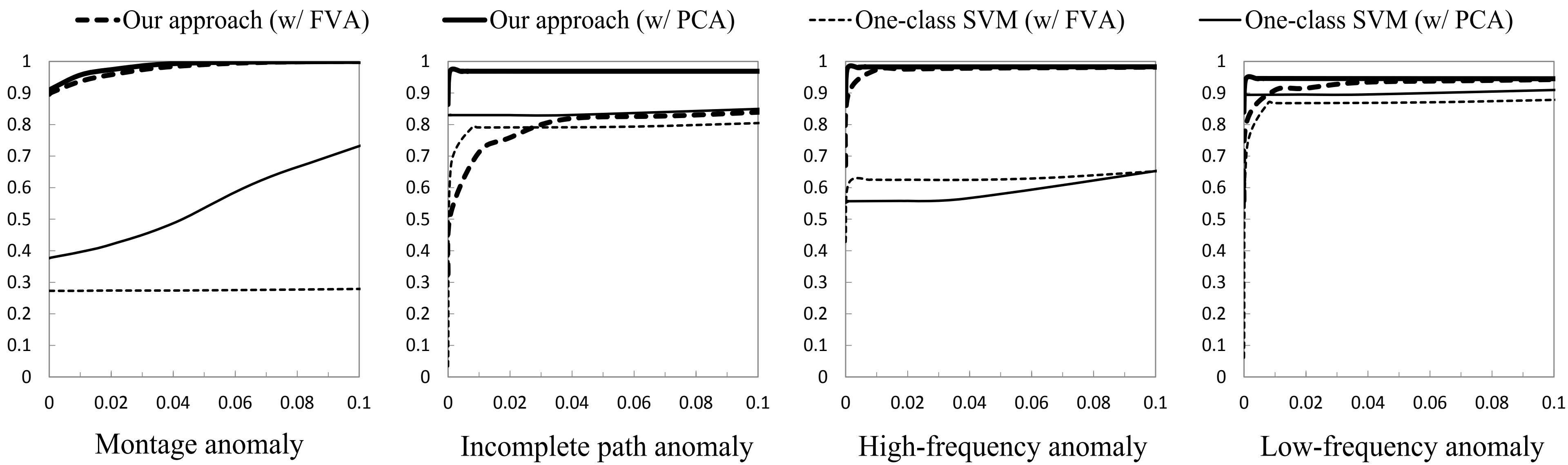
- Inter-cluster training
- Intra-cluster training

- Inter-cluster detection on co-occurrence matrices
- Intra-cluster detection on frequency matrices





Exp 1: Detection Accuracy vs. False Positive in Synthetic Anomalies



Frequency Anomalies

Under 10-fold cross-validation with 10,000 normal test cases, 1,000 synthetic anomalies.

Exp 2: Detection of Real-world Attacks in Complex Programs

sshd

Training w/
4,800 normal behavior instances (34K events each)

libpcre

Training w/
11,027 normal behavior instances (44K events each)

sendmail

Training w/
6,579 normal behavior instances (1K events each)

Flag variable overwritten attacks w/ various lengths

Regular Exp. DoS
3 malicious patterns
8-23 strings to match

Directory harvest attack w/ probing batch sizes:
8 to 400 emails

100% Detection accuracy
0.01% Average false alarm rate

How to lift this host security solution to the cloud?

Privacy

- Trust the provider or not?
- What is leaked, if detection is outsourced to the cloud?
- Is it possible to relax the privacy model?

Transparency

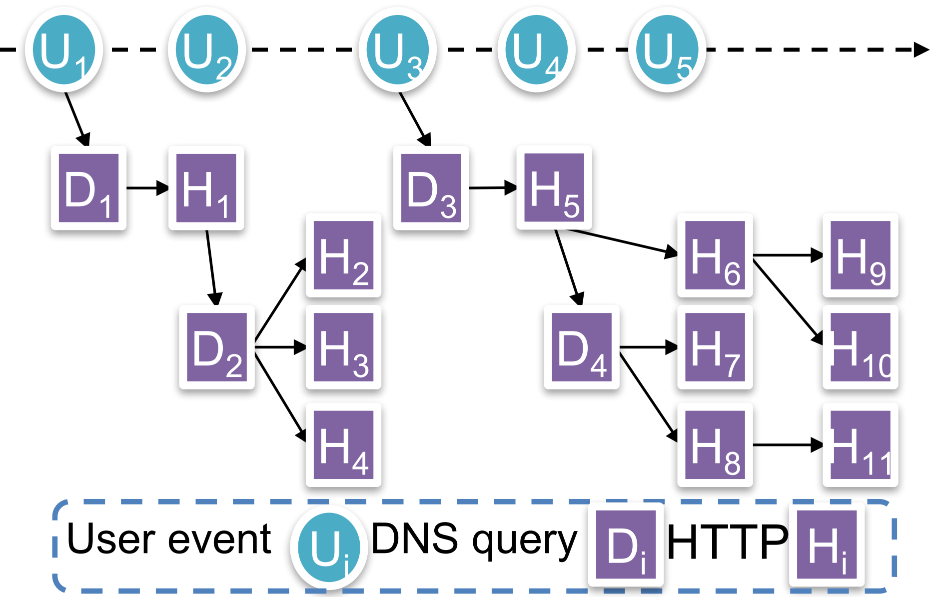
- Does the client need to be involved?
- Client gives feedback on detection results, like spam detection?

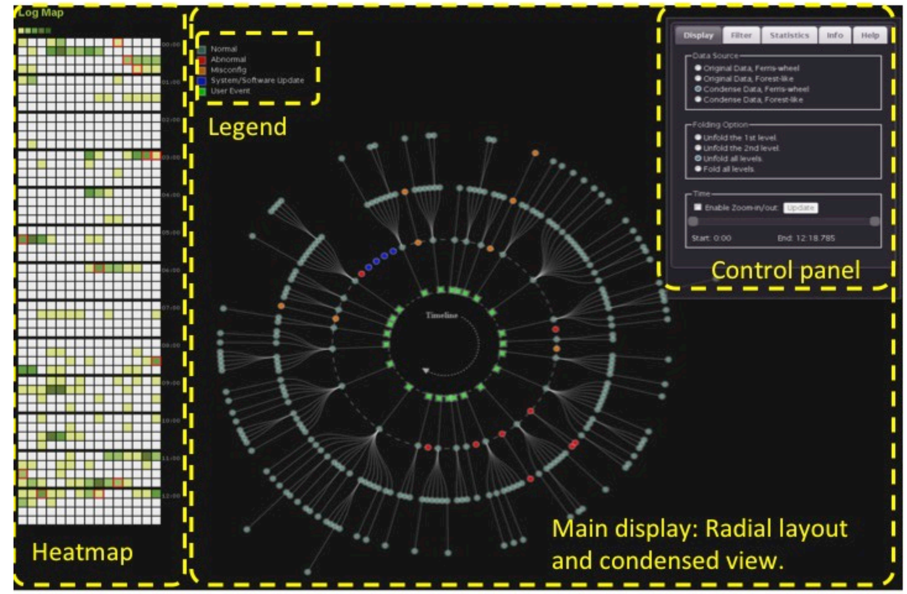
Correctness

- **How can client trust provider do a decent job?**

Host Security Solution 3: Triggering Relation Discovery

Triggering Relation Graph (TRG)





US Patent Granted.
NSF CAREER Award.

Prototypes for

- Android traffic, Linux traffic
- Filesys events

How to lift this analysis to the cloud?

[Zhang AISec '16] [Zhang *C&S* 2016]
[Zhang *ASIACCS* '14] [Xu *IEEE TDSC* '12]

Future Work: Anomaly Detection as a Cloud Service

Is it possible to
- be transparent to clients?
- for interdisciplinary data?
- with domain knowledge?
- in production systems?

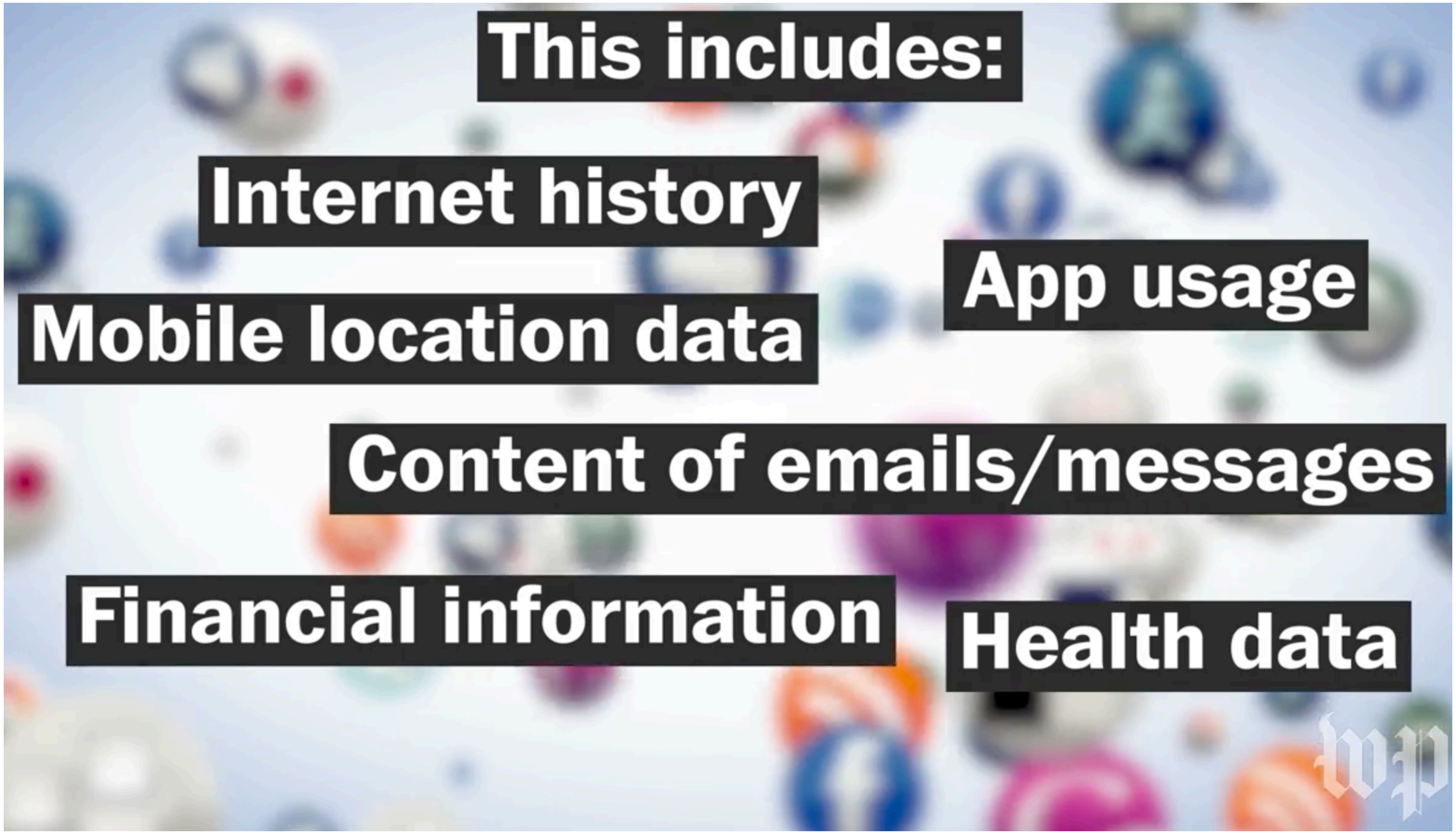
Can domain experts understand these suggestions?

- Some algorithms are not good for global anomalies;
- The safe bet is to try first global detection algorithms;
- If willing to wait (not real-time detection), use nearest neighbor
- If the dataset is small, definitely avoid clustering;
- Restart k-mean multiple times to obtain stable clusters;
- Avoid unsupervised anomaly detection for extremely high dimensions;

[Goldstein and Uchida 2016]

Privacy, is it a lost battle (at least in US)?

- US Internet service providers (ISP) to monitor customers' behavior online
- without users' permission,
- to use personal information to sell highly targeted ads



[Washington Post, March 28, 2017]

Lifting data-driven host protection to the cloud

Thank you for your attention!

Questions?

More information:

- http://people.cs.vt.edu/danfeng/
- CCS program anomaly detection tutorial video and slides
- System traces, hands-on exerises