VIRGINIA TECH™

Accidental Data Leak Detection, Secure Coding, and Payment Card Ecosystem (Part 2)

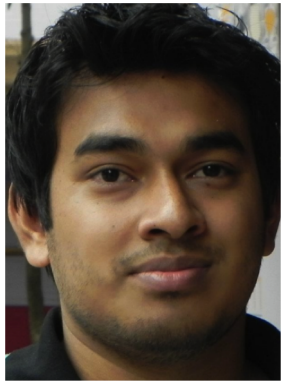
Danfeng (Daphne) Yao'
Professor of Computer Science
Virginia Tech

Brown EMCS Oct 2019

Acknowledgments

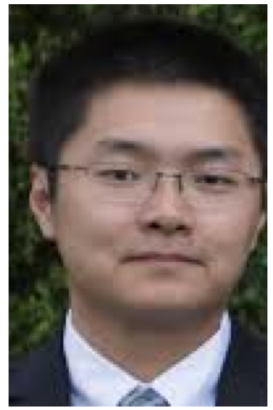
Stefan Nagy
(VT)

Xiaokui Shu
(IBM Research)

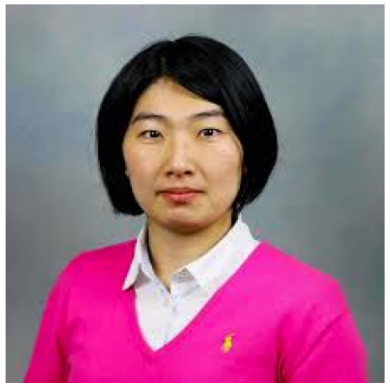


Sazzadur Rahaman
(VT)

Fang Liu
(Palo Alto Networks)



Jing Zhang
(AMD)

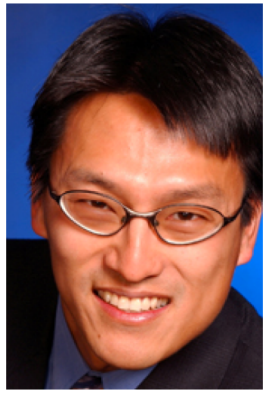


Na Meng
(VT)

Elisa Bertino
(Purdue)

Gang Wang
(UIUC)

Ali Butt
(VT)

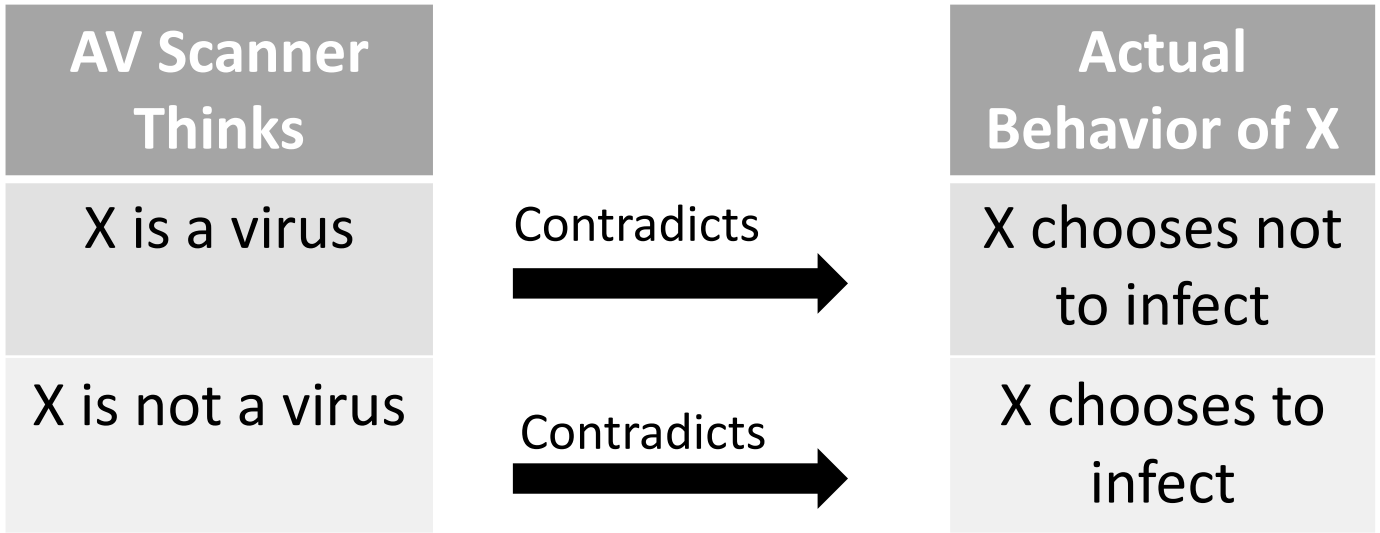


Wu Feng
(VT)

Impossibility of achieving absolute security

Smart virus X:

1. **if Scanner says no**
2. **then infect;**
3. **else do nothing;**



[Fred Cohen, J. of Virology 1987]

Absolute security is impossible. But people make money on security all the time.

Why?

Real quotes from the StackOverflow forum

*"Adding**csrf**().disable() solved the issue!!! I have no idea why it was enabled by default"*

"adding -Dtrust_all_cert=true to VM arguments"

"I want my client to accept any certificate (because I'm only ever pointing to one server)"

[Meng, Yao, et al. ICSE 2018] We examined 497 Java and security related StackOverflow Posts

Writing secure code is tough

```
1  // Create a trust manager that does not validate certificate chains
2  TrustManager[] trustAllCerts = new TrustManager[]{
3    new X509TrustManager() {
4      public java.security.cert.X509Certificate[]
           getAcceptedIssuers() {return null;}
5      public void checkClientTrusted(...) {}
6      public void checkServerTrusted(...) {} }};
7  // Install the all-trusting trust manager
8  try {
9    SSLContext sc = SSLContext.getInstance("SSL");
10   sc.init(null, trustAllCerts, new java.security.
         SecureRandom());
11   HttpsURLConnection.setDefaultSSLSocketFactory(sc
         .getSocketFactory());
12 } catch (Exception e) {}
```

[Meng, Yao, et al. ICSE 2018]

How Much Influence Does StackOverflow Have?

Insecure Posts	Total Views	No. of Posts	Min Views	Max Views	Average
Disabling CSRF Protection*	39,863	5	261	28,183	7,258
Trust All Certs	491,567	9	95	391,464	58,594
Obsolete Hash	91,492	3	1,897	86,070	30,497
Total Views	**622,922**	**17**	-	-	-

StackOverflow posts that make insecure suggestions have a large influence on developers.

* In Java Spring Security for web applications

Social Dynamics on Stackoverflow

User: skanga [0]

“Do NOT EVER trust all certificates. That is very dangerous.”

“the "accepted answer" is wrong and INDEED it is DANGEROUS. Others who blindly copy that code should know this.”

User: MarsAtomic [6,287]

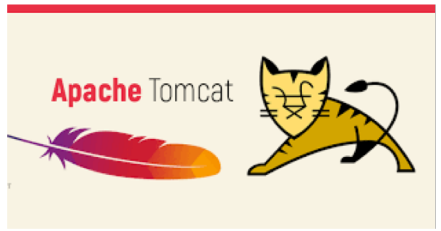
“once you have sufficient reputation you will be able to comment”

“If you don't have enough rep to comment, … then participate … until you have enough rep.”

https://stackoverflow.com/questions/10594000/when-i-try-to-convert-a-string-with-certificate-exception-is-raised

CryptoGuard

Deployment-quality Accuracy and Scalability



Maximum & minimum LoC: 2,571K (Hadoop), 1.1K (Commons Crypto); and average LoC: 402K

Security Issues we found in professionally developed Apache software projects

How to measure the quality of PCI scanners?

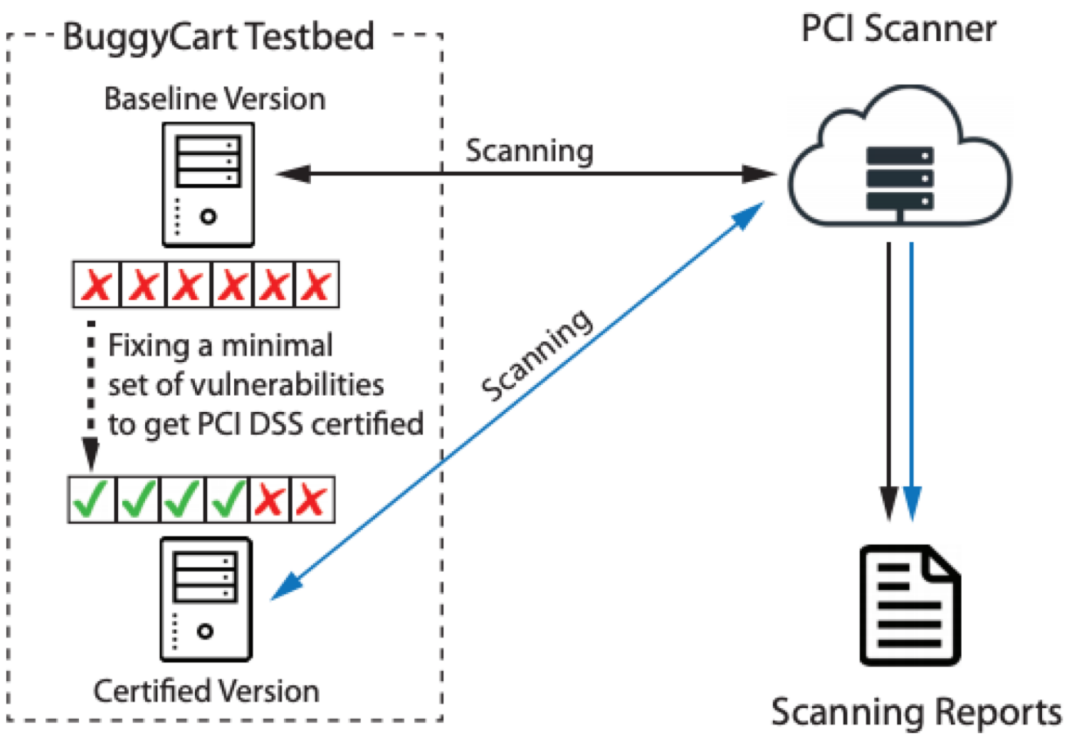
Security Certification in Payment Card Industry: Testbeds, Measurements, and Recommendations. S. Rahaman, G. Wang, and D. Yao. *ACM Conference on Computer and Communications Security (CCS) 2019*. London, UK.

Can We Measure the Strength of PCI Enforcement?

Our BuggyCart Testbed embeds 35 vulnerabilities (to open source)

- Network security (14 test cases)
- System security (7 test cases)
- Web Application security (8 test cases)
- Secure storage (6 test cases) – cannot be detected by external scans

Our BuggyCart Testbed and Commercial PCI Scanners Selected



PCI Scanners	Price	Spent Amount
Scanner 1	$2,995/Year	$0 (Trial)
Scanner 2	$2,190/Year	$0 (Trial)
Scanner 3	$67/Month	$335
Scanner 4	$495/Year	$495
Scanner 5	$250/Year	$250
Scanner 6	$59/Quarter	$118
Scanner 7	Unknown	N/A
Scanner 8	$350/Year	N/A
Total	-	$1198

Web Security Test Cases Are Particularly Weak

Req.	Test Cases	Vul. Location	Is Within ASV Scope?	Scanner 1		Scanner 2		Scanner 3		Scanner 4		Scanner 5	
				Baseline	Certified	Baseline	Certified	Baseline	Certified	Baseline	Certified	Baseline	Certified
6.5	26. Sql Inject in Admin Login	Webapp	Yes	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	27. Sql Inject in Customer Login	Webapp	Yes	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	28. Disable password retry restriction	Webapp	Yes	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	29. Allow passwords with len <8	Webapp	Yes	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	30. Javascript source integrity check	Webapp	Yes	●	✓	✗	✗	✗	✗	✗	✗	✗	✗
	31. Don't hide program crashes	Webapp	Yes	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	32. Implant XSS	Webapp	Yes	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	33. Implant CSRF	Webapp	Yes	◐	✓	✗	✗	✗	✗	✗	✗	✗	✗

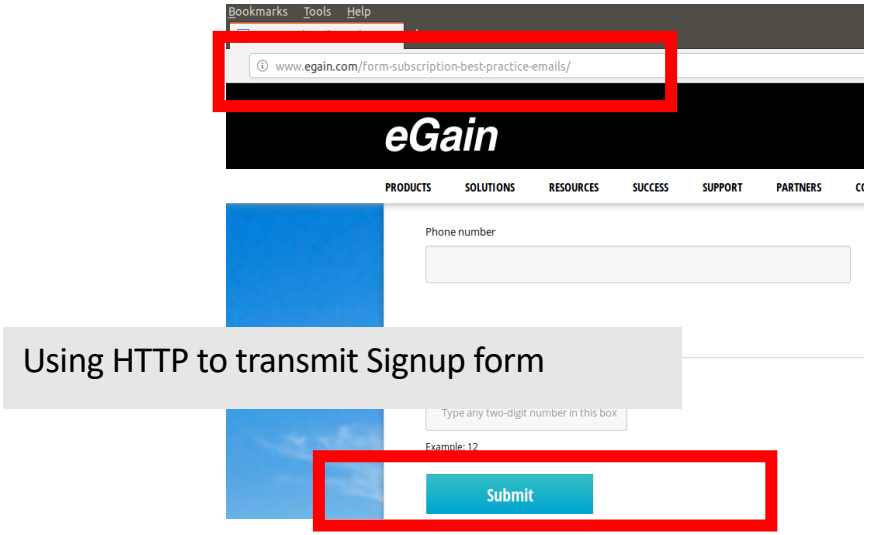
Worrisome PCI scanners security – Summary of Testbed Results

	Scanner 1	Scanner 2	Scanner 3	Scanner 4	Scanner 5
#Vul. Detected (29 Total*)	**21**	**16**	**17**	**16**	**7**
#Vul. Remaining in Certified Ver.	**7**	**15**	**18**	**20**	**25**
#Vul. detected, but did not fix	**0**	**3**	**7**	**7**	**4**

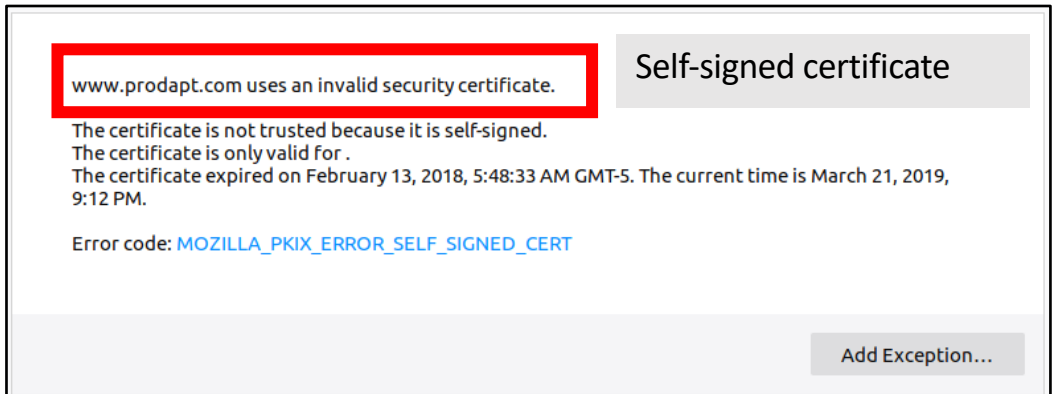
*All 29 vulnerabilities violate the PCI's data security specifications and are required by the specifications to be removed.

Assessed 1203 e-commerce sites with our PCICheckerLite tool

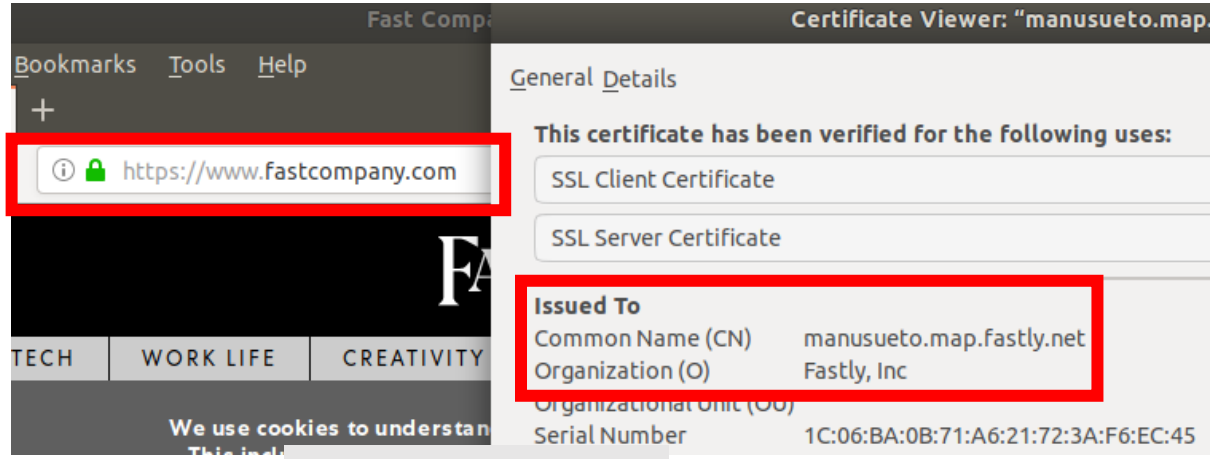
E-commerce Websites		#Vul. Websites	
		At least 1	At least 2
Category (810)	Business (122)	113	81
	Shopping (163)	143	99
	Arts (78)	76	54
	Adults (65)	65	43
	Recreation (84)	75	58
	Computer (57)	56	44
	Games (42)	42	31
	Health (60)	55	41
	Home (102)	93	65
	Kids & Teens (37)	36	21
Ranking (393)	Top (288)	277	203
	Bottom (105)	104	87
Total (1,203)		1,135 (94%)	827 (69%)



Using HTTP to transmit Signup form



Self-signed certificate



Wrong hostname

Key PCI Takeaways

5 out of 6 PCI scanners

certify vulnerable merchant sites

94% websites (out of 1,203)

Not PCI compliant

[Rahaman, Wang, Yao. ACM CCS 2019]

Data Leak Detection as an Add-on Service by Cloud Providers to Prevent Data Exposure

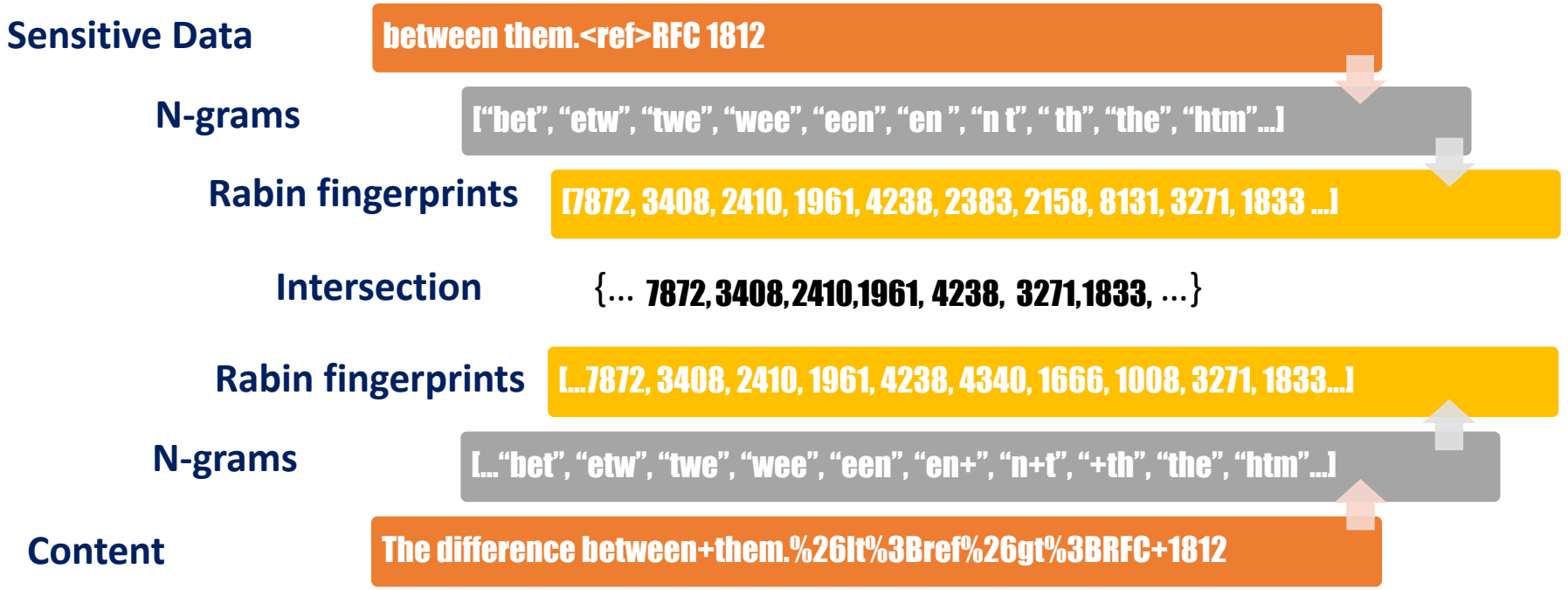
Clients do not want the cloud/detection providers to learn about the sensitive information.

How?

The Basic Set-Intersection Approach

Adding some twists to the set intersection based detection approach

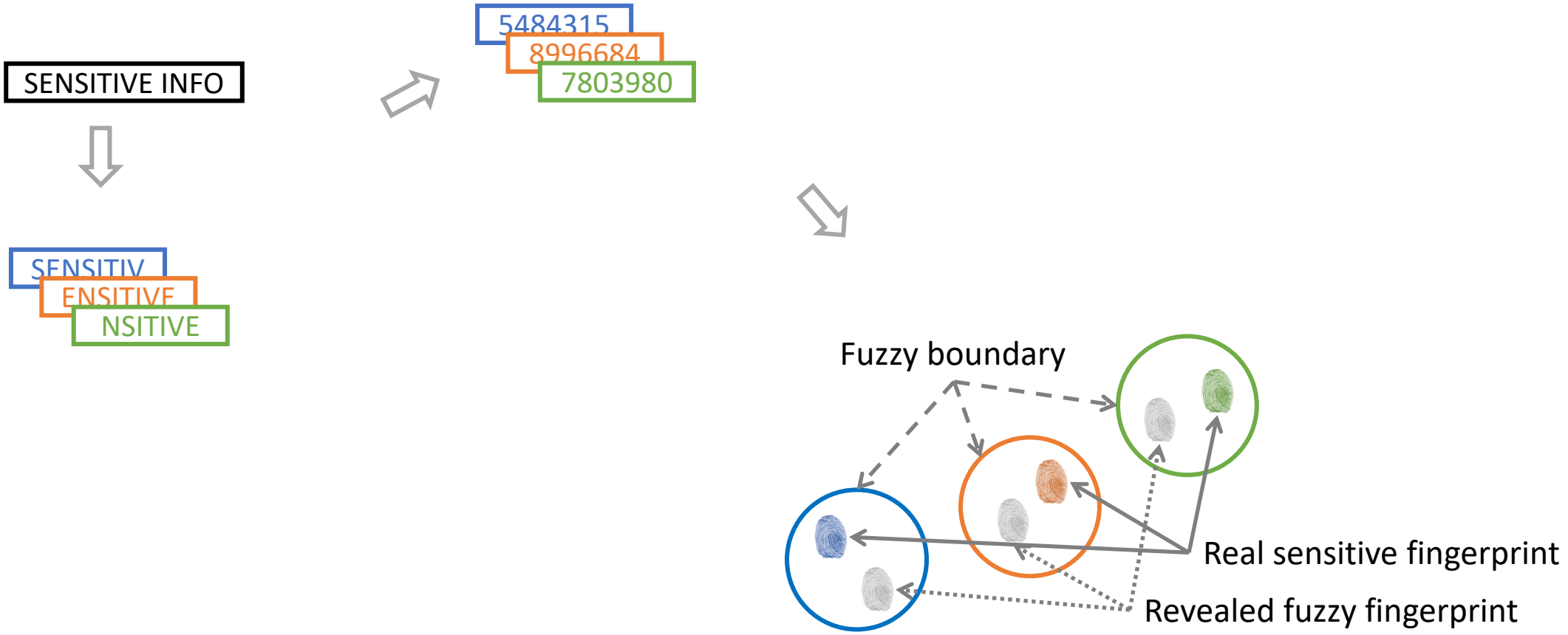
N-gram generation & Rabin fingerprints



[Shu and Yao. SecureComm '12] [Shu, Yao, and Bertino. *IEEE TIFS* '15]

21

A Twist -- Fuzzy Fingerprints



[Shu, Yao, and Bertino. *IEEE TIFS* '15]

Top 25 most downloaded article of IEEE Signal Processing Society in 2018

Another work: Detection of transformed accidental data leak?

Auto-formatting (WordPress)

The application layer contains the higher-level protocols used by most applications for network communication. Examples of application layer protocols include the File Transfer Protocol (FTP) and the Simple Mail Transfer Protocol (SMTP).[19] Data coded according to application layer protocols are then encapsulated into one or (occasionally) more transport layer protocols (such as TCP or UDP), which in turn use lower layer protocols to effect actual data transfer.

The+application+layer+contains+the+higher-level+protocols+used+by+most+applications+for+network+communication.+Examples+of+application+layer+protocols+include+the+File+Transfer+Protocol+(FTP)+and+the+Simple+Mail+Transfer+Protocol+(SMTP).[19]+Data+coded+according+to+application+layer+protocols+are+then+encapsulated+into+one+or+(occasionally)+more+transport+layer+protocols+(such+as+TCP+or+UDP),+which+in+turn+use+lower+layer+protocols+to+effect+actual+data+transfer.

Partial source code leak

```
def encode(msg, pubkey, verbose=False):
    chunksize = int(log(pubkey.modulus, 256))
    outchunk = chunksize + 1
    outfmt = '%%0%dx' % (outchunk * 2,)
    bmsg = msg if isinstance(msg, binary_type) else msg
    result = []
    for start in range_func(0, len(bmsg), chunksize):
        chunk = bmsg[start:start + chunksize]
        chunk += b'\x00' * (chunksize - len(chunk))
        plain = int(hexlify(chunk), 16)
        coded = pow(plain, *pubkey)
        bcoded = unhexlify((outfmt % coded).encode())
        if verbose:
            print('Encode:', chunksize, chunk, plain, c
        result.append(bcoded)
```

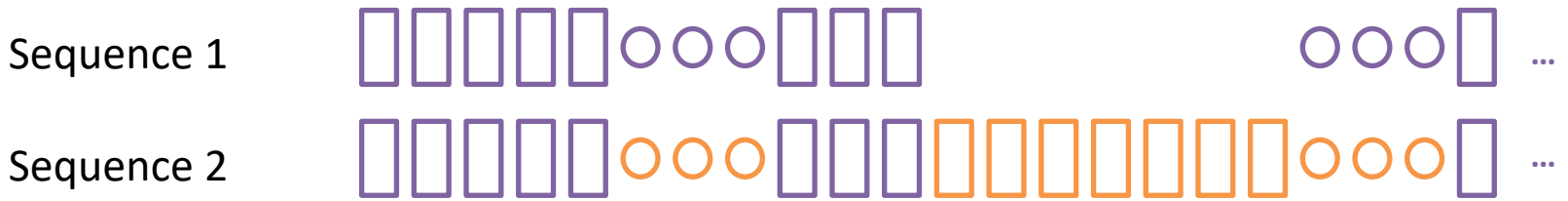
```
return b''.join(result).rstrip(b'\x00').decode('utf-8')

def __delitem__(self, item):
    self._remove_from_dict(item)
    self.heap = [(v,k) for v,k in self.heap if k != ite
    chunk += b'\x00' * (chunksize - len(chunk))
    heapq.heapify(self.heap)

def pop(self):
    _, smallest = heapq.heappop(self.heap)
    self._remove_from_dict(smallest)
    return smallest
```

[Shu, et al. *IEEE TIFS 2016*]

Transformed data leak –
Our sequence-alignment based detection



[Shu, et al. *IEEE TIFS 2016*]

Also invented a smart sampling algorithm

2 identical input streams:

```
1, 9, 4, 5, 3, 5, 9, 7, 6, 6, 3, 3, 7, 1
1, 9, 4, 5, 3, 5, 9, 7, 6, 6, 3, 3, 7, 1
```

Output of random sampling:

```
1, -, 4, -, 3, 5, -, 7, -, 6, -, -, 7, 1
-, 9, -, 5, -, 5, -, 7, -, 6, 3, -, -, 1
```

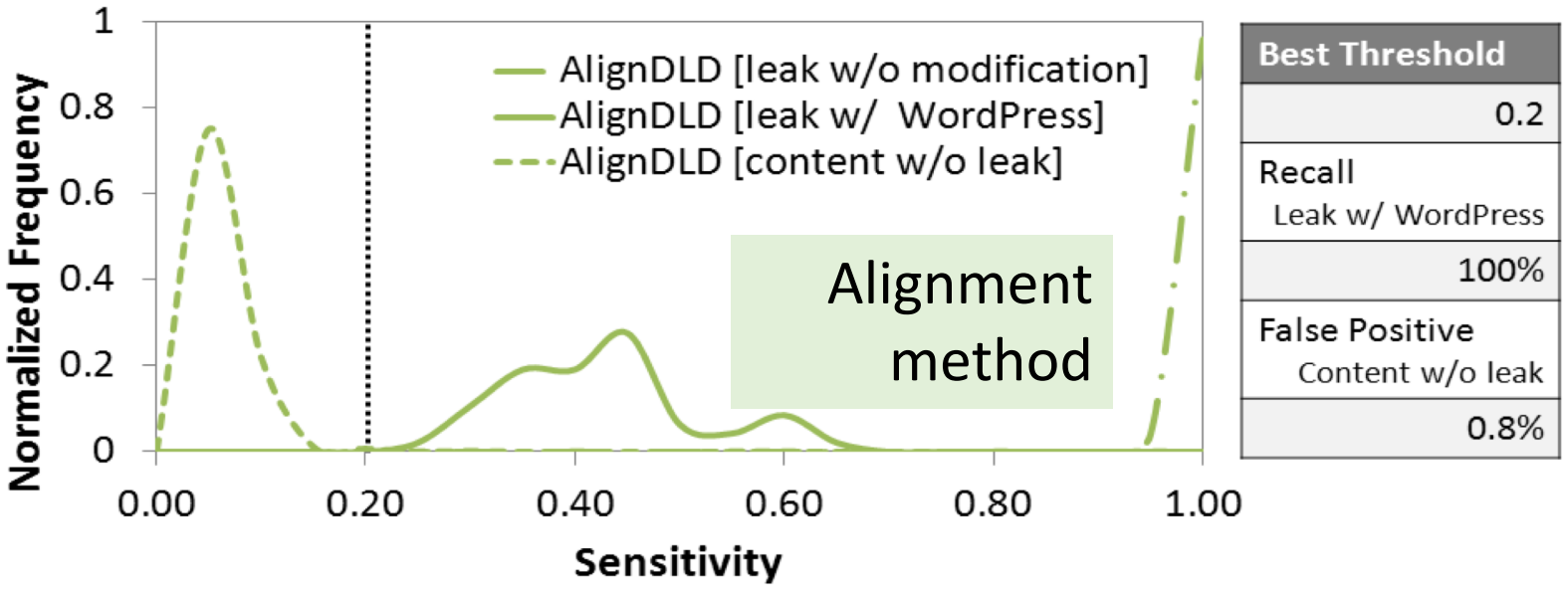
Output of our comparable sampling:

```
1, -, 4, -, 3, 5, -, -, -, -, 3, 3, -, 1
1, -, 4, -, 3, 5, -, -, -, -, 3, 3, -, 1
```

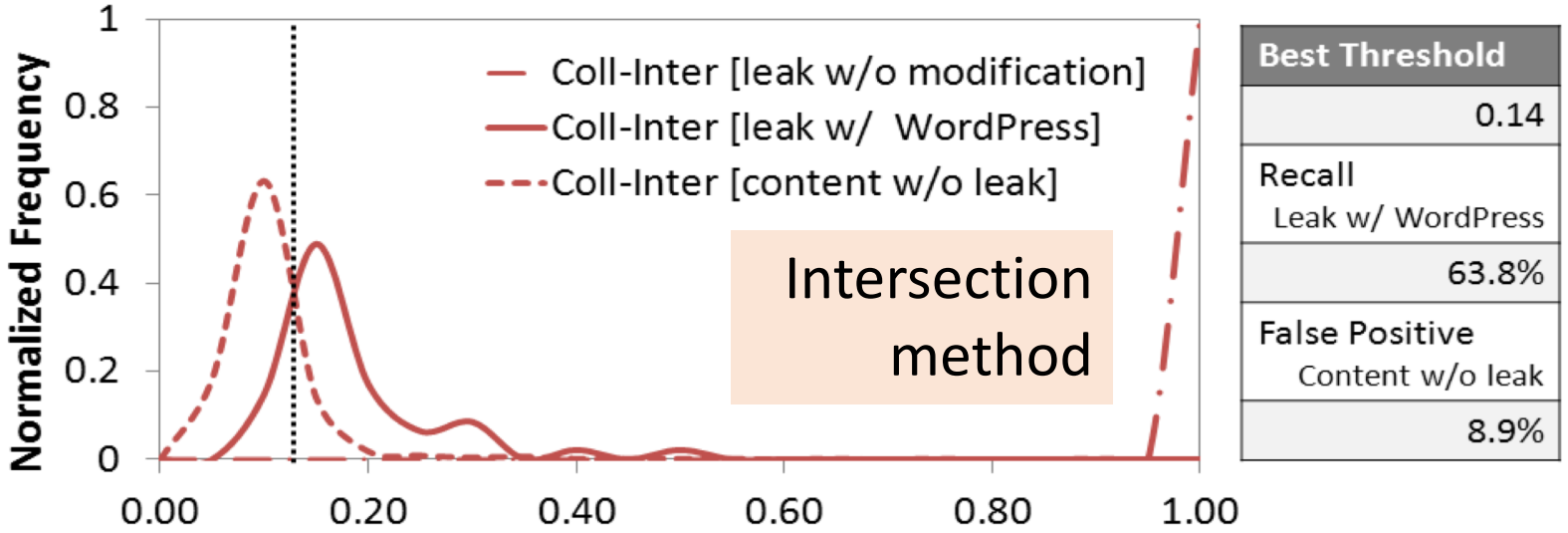
If *x is* a substring of *y*, then *x'* (the sample of *x*) is a substring of *y'* (the sample of *y*).

[Shu, et al. *IEEE TIFS 2016*]

Transformed leak stands out in the alignment-based detection



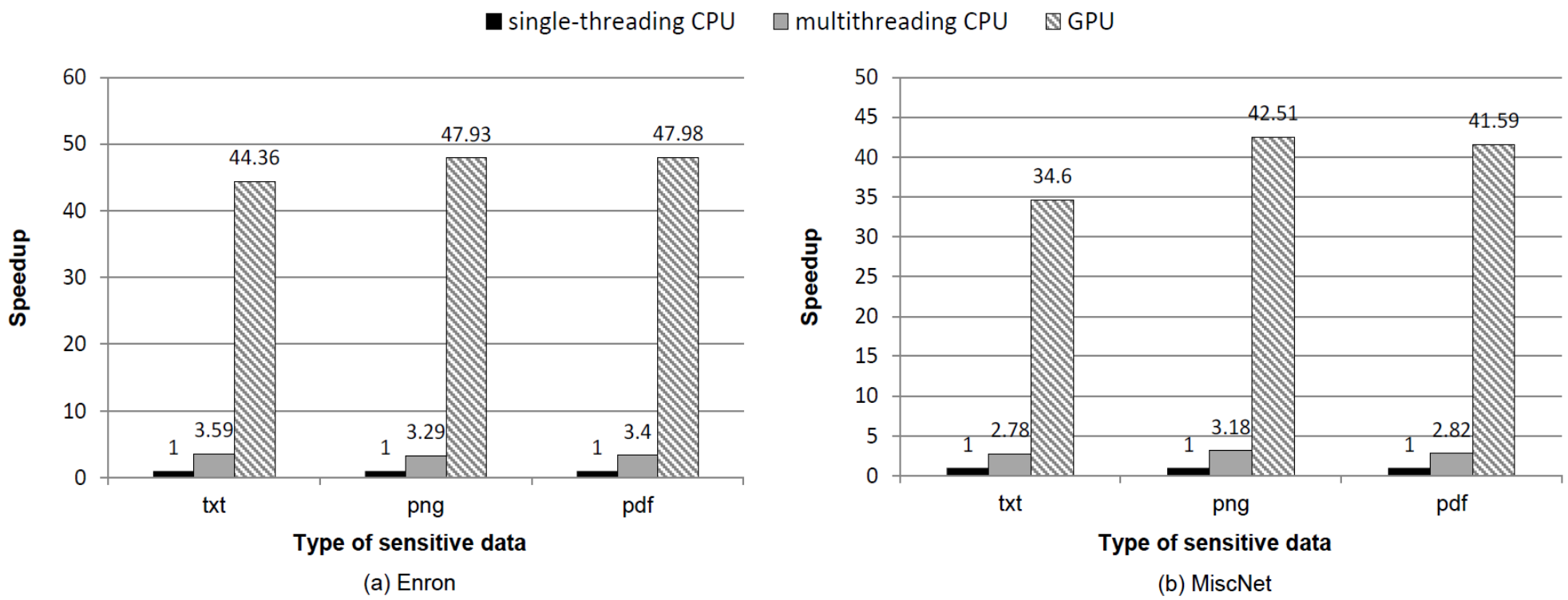
Best Threshold
0.2
Recall Leak w/ WordPress
100%
False Positive Content w/o leak
0.8%



Best Threshold
0.14
Recall Leak w/ WordPress
63.8%
False Positive Content w/o leak
8.9%

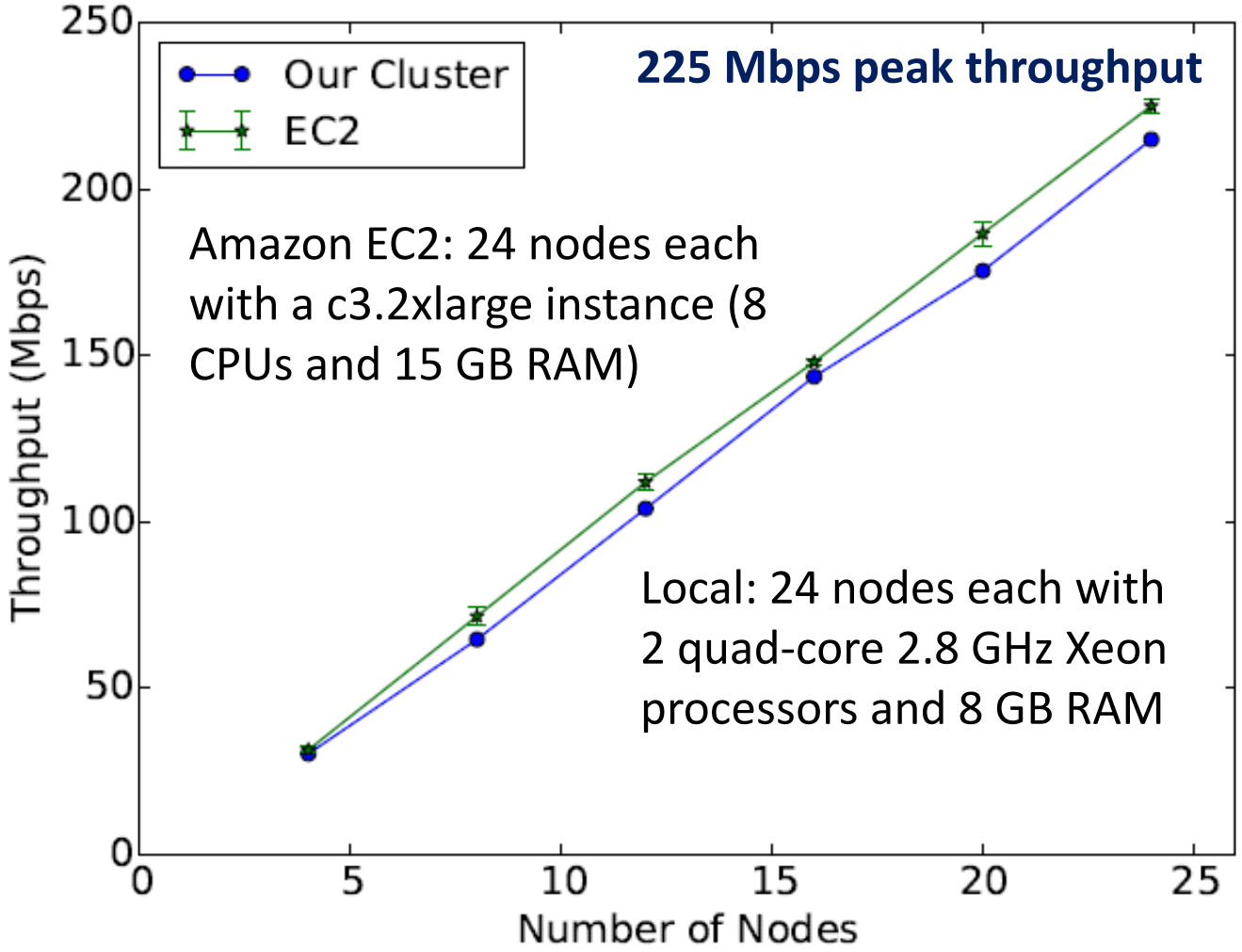
Enron dataset (2.6GB): 150 users, 517,424 emails. 3-grams.

GPU acceleration of AlignDLD



Testing Platforms	# of Cores
CPU	
Intel Core i5 2400, Sandy-Bridge microarchitecture	4
GPU (single)	
Nvidia Tesla C2050, Fermi architecture	448

Hadoop (distributed hashtable) implementation of the set intersection based detection



37 GB Enron Email Corpus as content

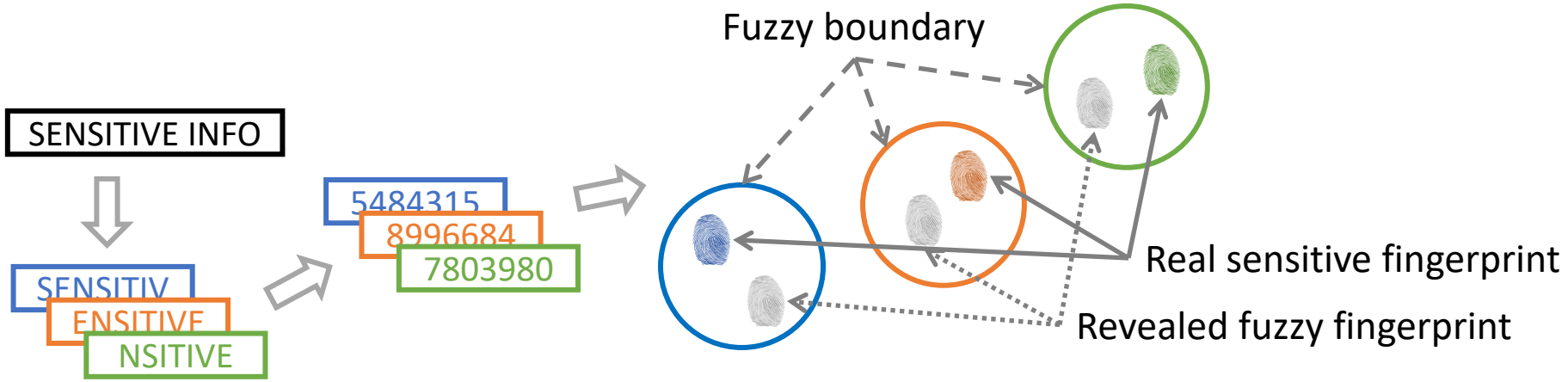
[Liu, et al. *ACM CODASPY 2015*]

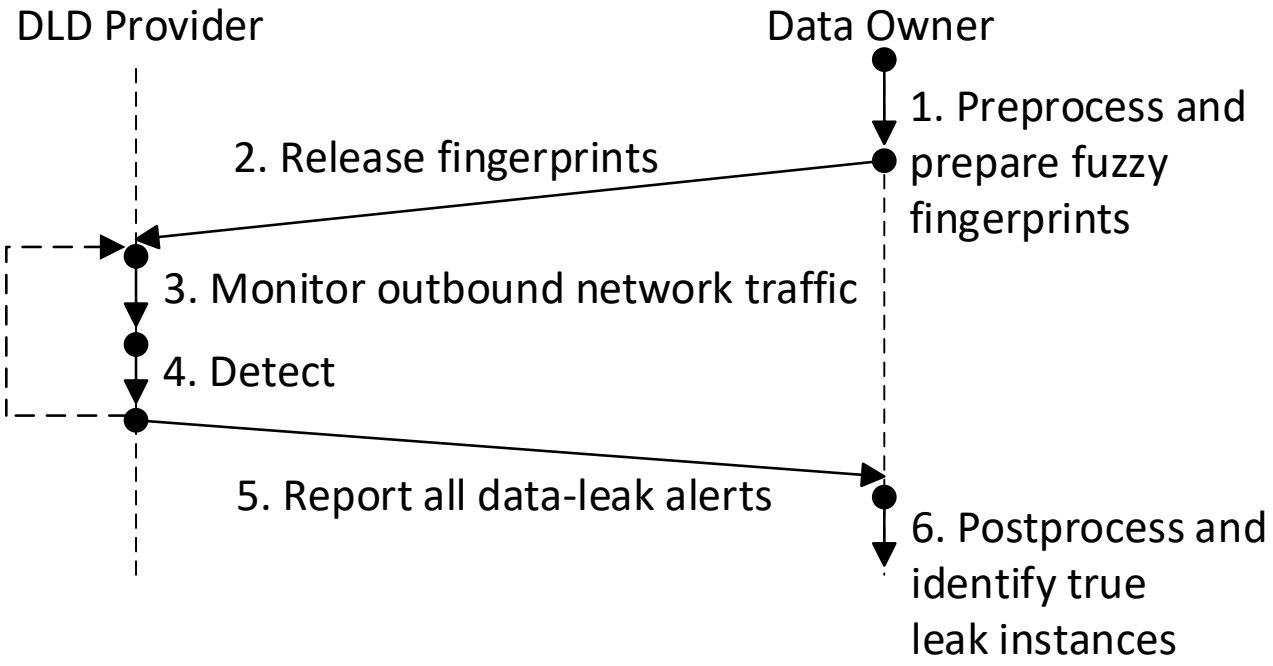
What executives should do?


OCTOBER IS
NATIONAL CYBERSECURITY AWARENESS MONTH
Learn how you can get involved at
STAYSAFEONLINE.ORG/NCSAM
OWN
SECURE
PROTECT
IT.
OCTOBER 2019
National Cybersecurity
Awareness Month
#BeCyberSmart

Questions?

Fuzzy fingerprints and the detection protocol





PPDLD

1) Pre-process
2) Release
3) Monitor
4) Detect
5) Report
6) Post-process

[Shu, Yao, and Bertino. *IEEE TIFS* '15]