Generative Intrusion Detection and Prevention on Data Stream

HyungBin Seo and MyungKeun Yoon
Kookmin University

USENIX Security 2023

KMU KOOKMIN UNIVERSITY

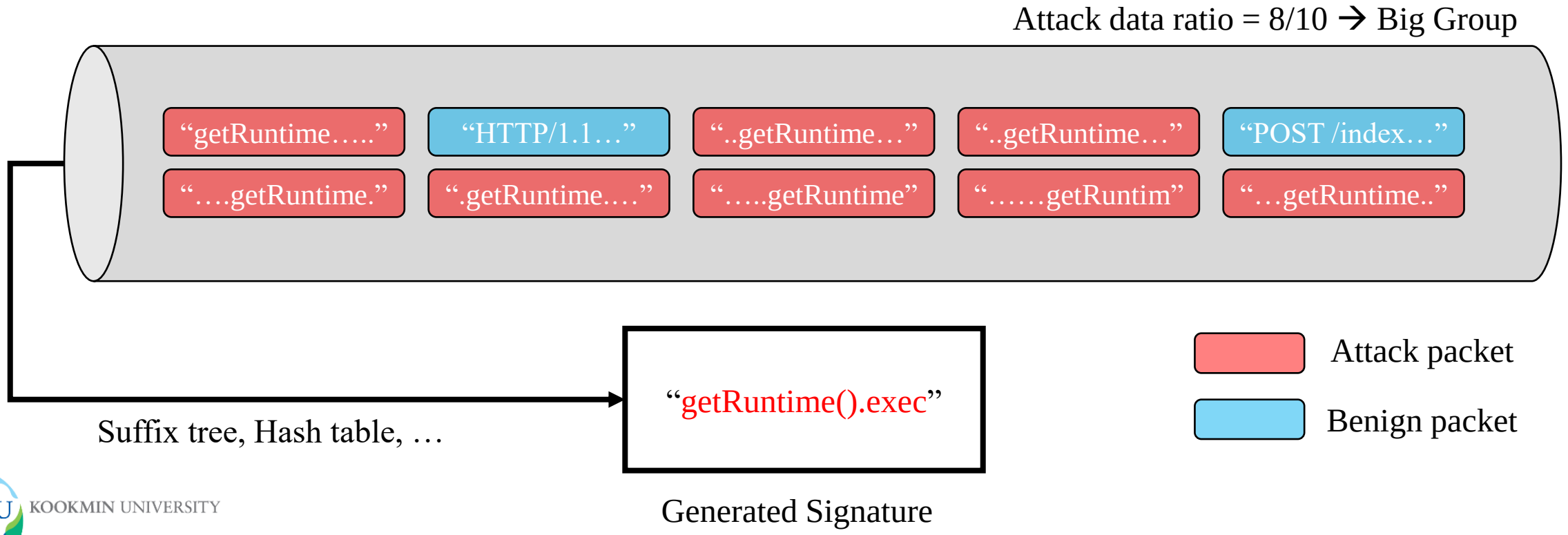
Motivation

- Different Types of Intrusion Detection
  - Persistent zero-day attack, signature-group generation, evidence support

Type		Accuracy	Automation	Zero-day
Misuse-detection (signature)	Manual rule composition [49, 53]	High	Low	Low
	Signature generation [9, 48]	**Medium**	**Medium**	**High**
	Signature-group generation (GIPS)	**High**	**High**	**High**
Anomaly-detection (ML)	Supervised learning [19, 52]	Medium	Medium	Medium
	Unsupervised learning [54, 62]	Low	Medium	High

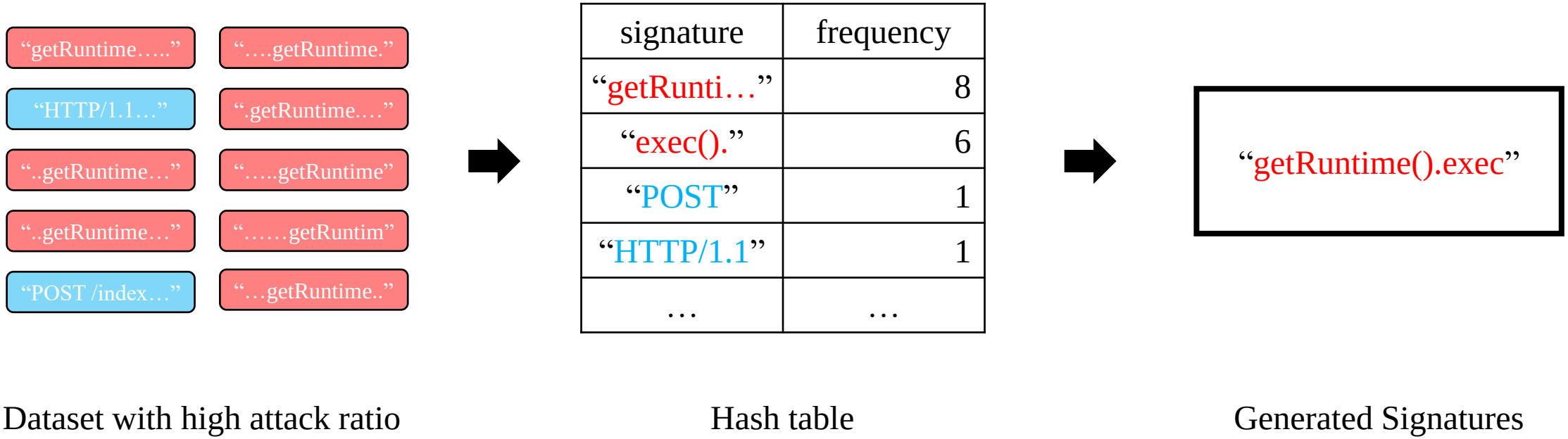
Motivation

- Previous signature-generation schemes [9], [24], [35], [39], [48]
  - high-volume attacks such as DDoS, worms



Motivation

- Previous signature-generation schemes [9], [24], [35], [39], [48]
  - high-volume attacks such as DDoS, worms



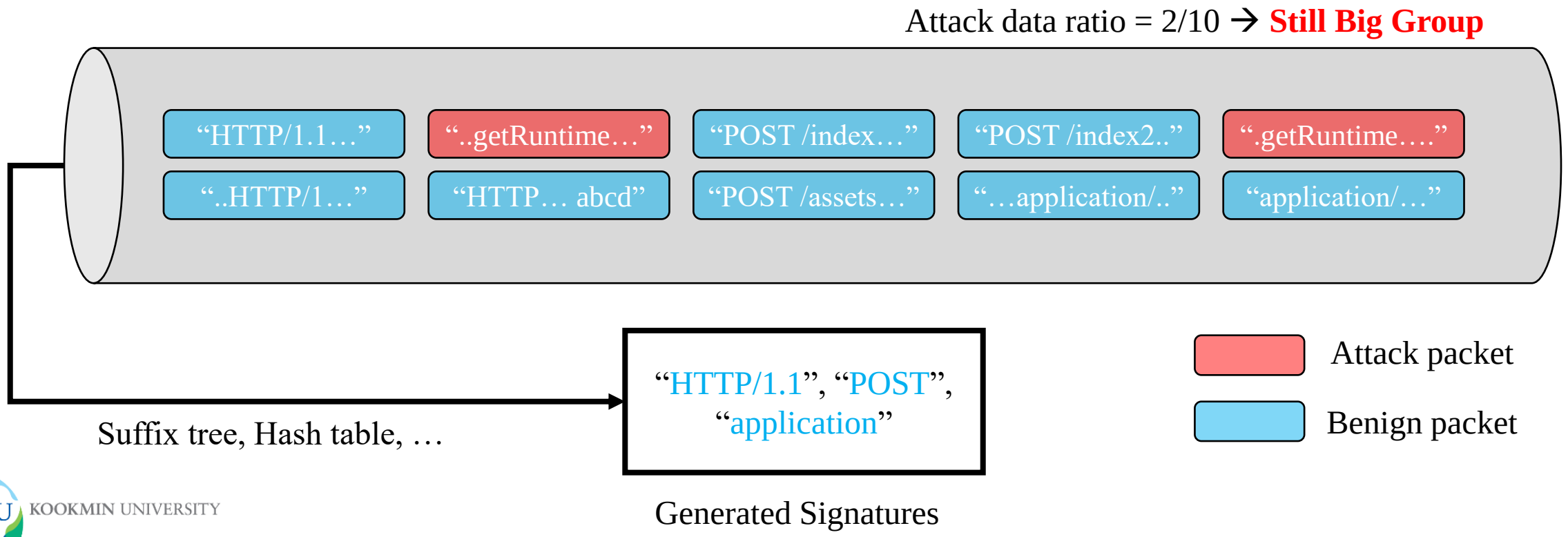
Dataset with high attack ratio

Hash table

Generated Signatures

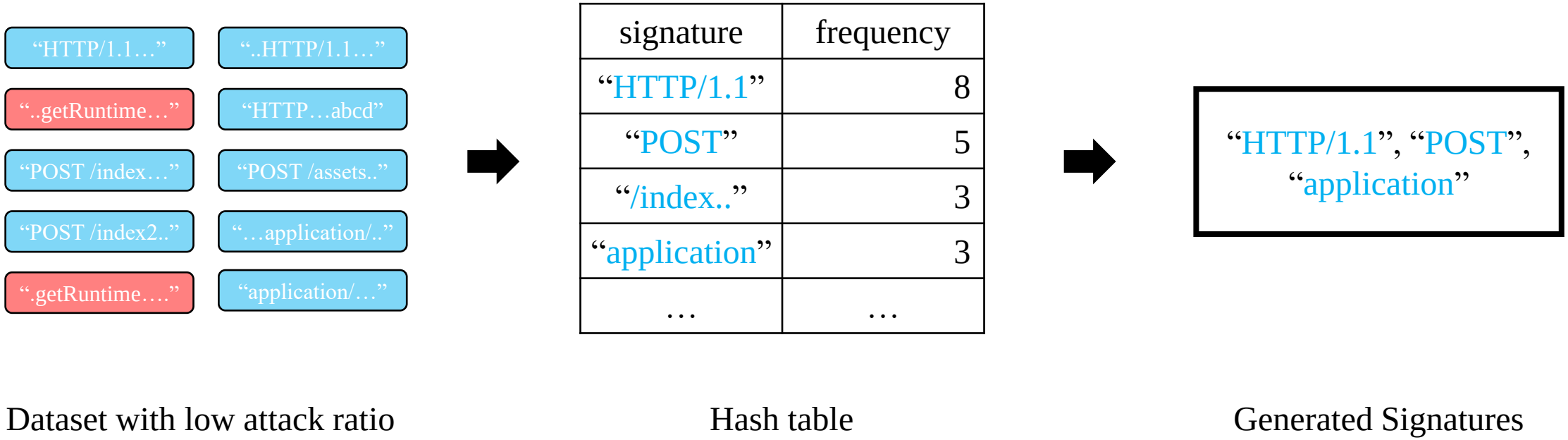
Motivation

- Previous schemes would cause false-positives for **persistent zero-day attacks.**



Motivation

- Previous schemes would cause false-positives for **persistent zero-day attacks.**



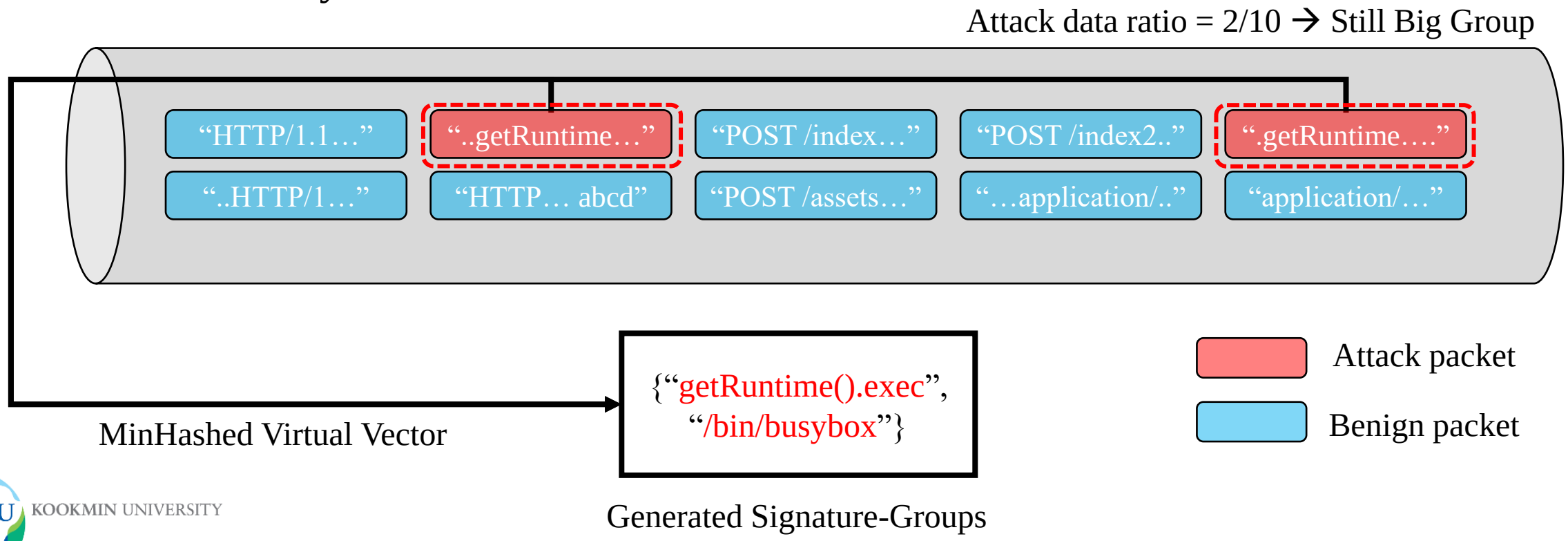
Dataset with low attack ratio

Hash table

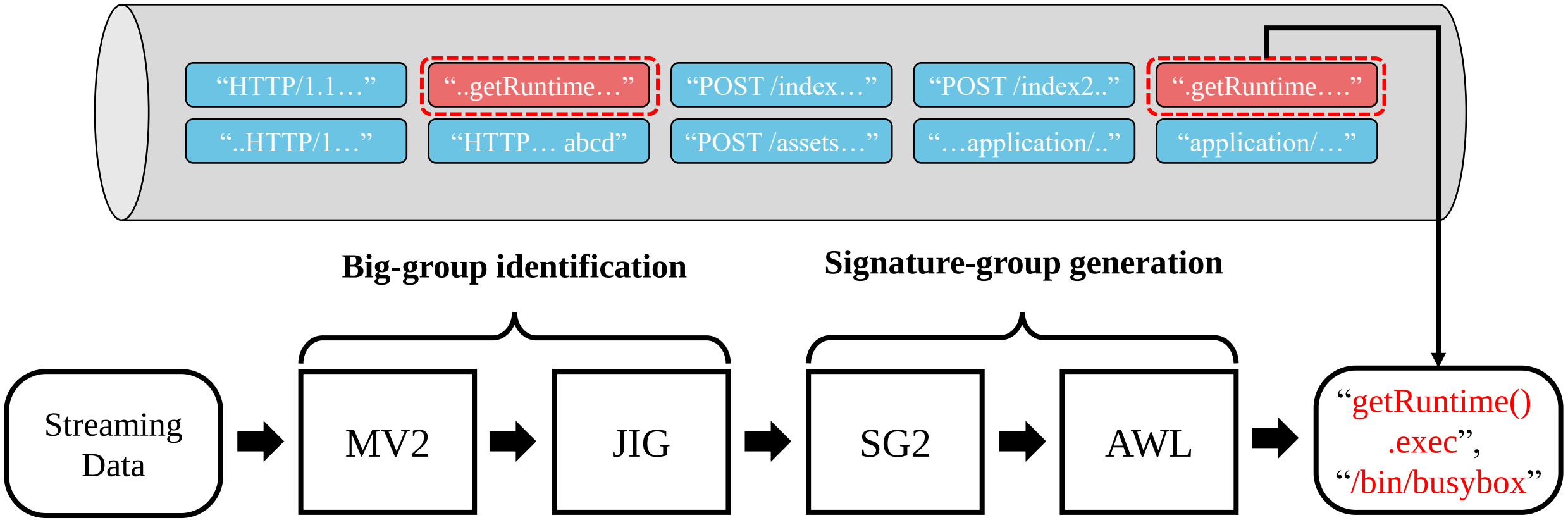
Generated Signatures

Motivation

- GIPS
  - Generative Intrusion detection and Prevention on data Stream
  - Data analysis tool



Overall Process of GIPS



- MV2: Minhashed Virtual-Vector
- JIG: Jaccard-Index Grouping

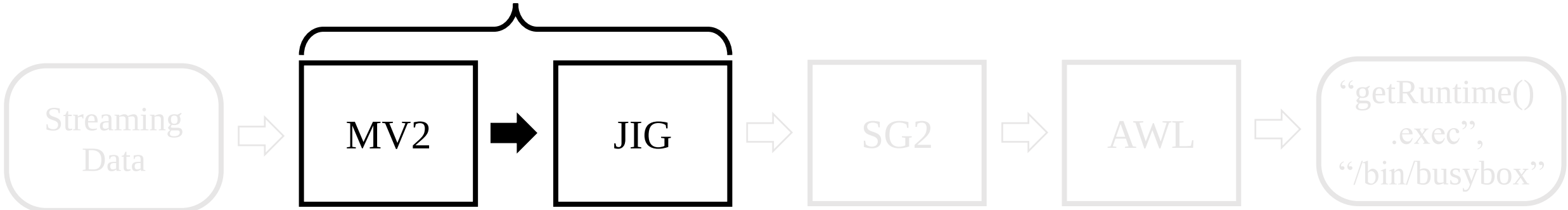
- SG2: Signature-Group Generation
- AWL: Automatic WhiteListing

Overall Process of GIPS

Streaming Data ➡ MV2 ⇨ JIG ⇨ SG2 ⇨ AWL ⇨ “getRuntime() .exec”, “/bin/busybox”

Overall Process of GIPS

Big-group identification

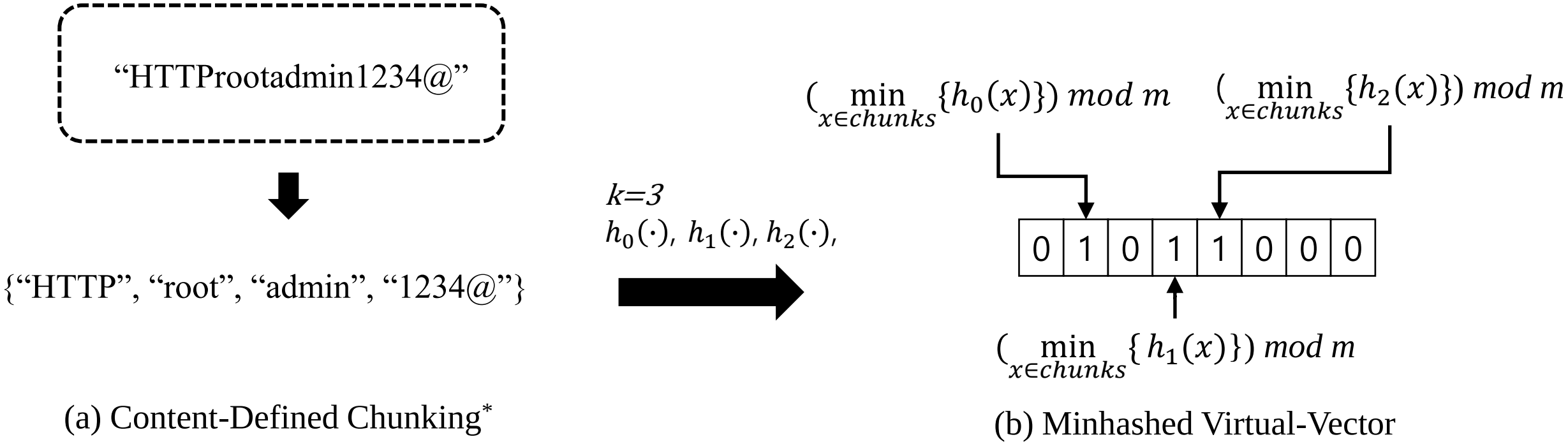


Overall Process of GIPS

Signature-group generation

Streaming Data ⇨ MV2 ⇨ JIG ⇨ SG2 ➡ AWL ➡ “getRuntime() .exec”, “/bin/busybox”

MV2: Minhashed Virtual-Vector



(a) Content-Defined Chunking*

(b) Minhashed Virtual-Vector

KOOKMIN UNIVERSITY

* "AE: An Asymmetric Extremum content defined chunking algorithm for fast and bandwidth-efficient data deduplication", in IEEE Conference on Computer Communications, 2015.

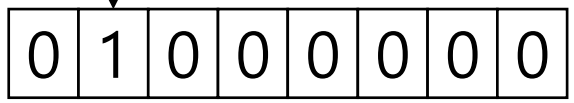
MV2: Minhashed Virtual-Vector

{“HTTP”, “root”, “admin”, “1234@”}

$h_0("HTTP") = 71$
$h_0("root") = 48$
$h_0("admin") = 9$
$h_0("1234@") = 57$

➡ $\min_{x \in chunks} h_0(x) = 9$

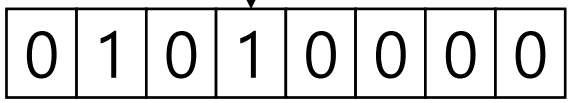
$9 \; mod \; 8$



$h_1("HTTP") = 87$
$h_1("root") = 11$
$h_1("admin") = 95$
$h_1("1234@") = 24$

➡ $\min_{x \in chunks} h_1(x) = 11$

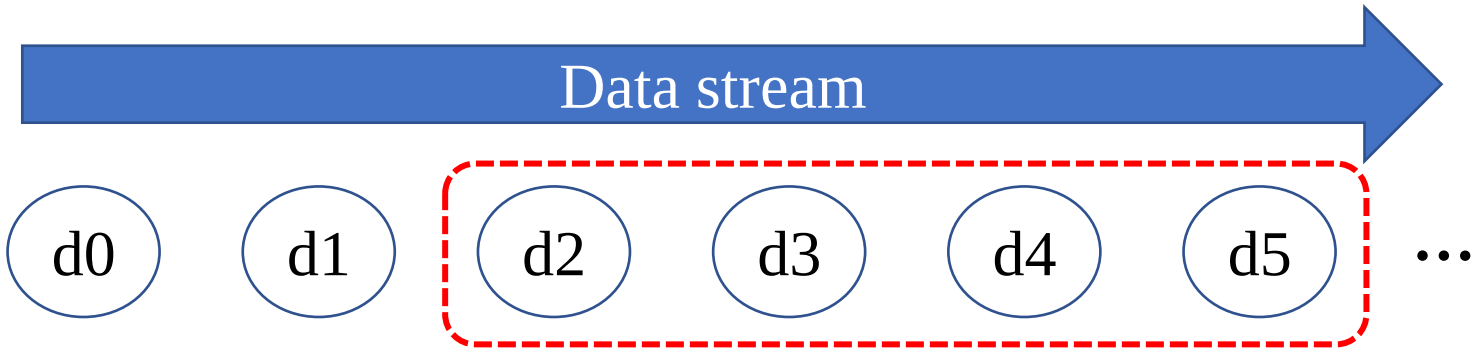
$11 \; mod \; 8$



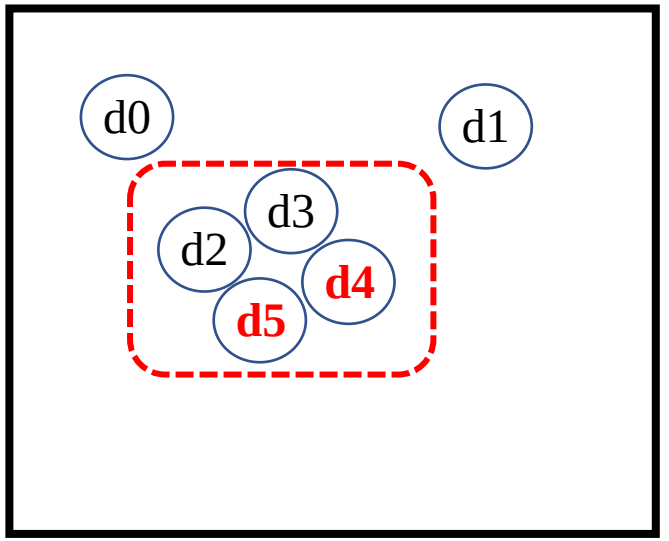
JIG: Jaccard-Index Grouping

- Big-group of d_i from data stream

$$sg(d_i) = \{d_j | j(d_i, d_j) > \theta_J\} \cup \{d_i\} \qquad b_r(d_i) = \frac{|sg(d_i)|}{i} > \theta_B$$



As their similarity to d5 is greater than θ_J,
d5 belongs to a Big-group



Naïve clustering on data stream requires a large time & space-complexity!

JIG: Jaccard-Index Grouping

d0 = “GET /index.html HTTP/1.1”
d1 = “HTTProotadmin1234@”
d2 = “GET / HTTP/1.1 admin”
d3 = “HTTP root directory”
d4 = “GET root12, admin123@”
d5 = “GET root12, admin123@”

(a) Data stream

MV	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

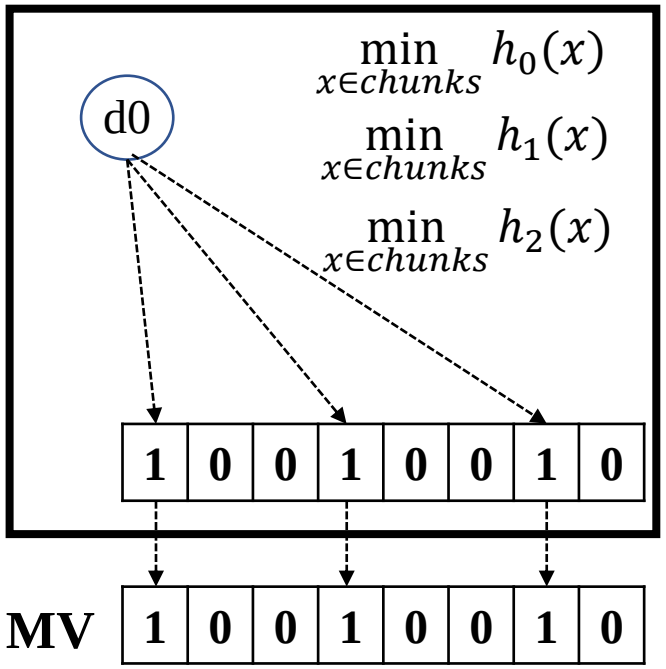
θ_J=0.6

(b) Counter array

JIG: Jaccard-Index Grouping

d0 = "GET /index.html HTTP/1.1"
d1 = "HTTProotadmin1234@"
d2 = "GET / HTTP/1.1 admin"
d3 = "HTTP root directory"
d4 = "GET root12, admin123@"
d5 = "GET root12, admin123@"

(a) Data stream



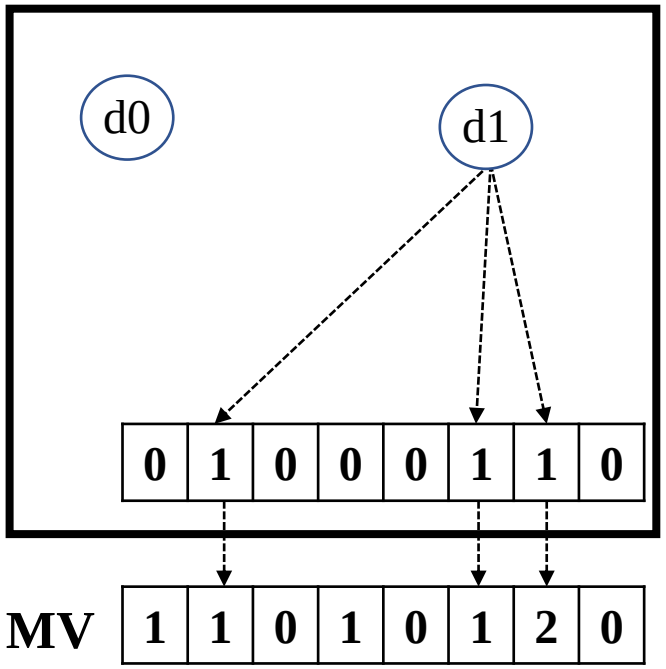
θ_J=0.6

(b) Counter array

JIG: Jaccard-Index Grouping

d0 = “GET /index.html HTTP/1.1”
d1 = “HTTProotadmin1234@”
d2 = “GET / HTTP/1.1 admin”
d3 = “HTTP root directory”
d4 = “GET root12, admin123@”
d5 = “GET root12, admin123@”

(a) Data stream



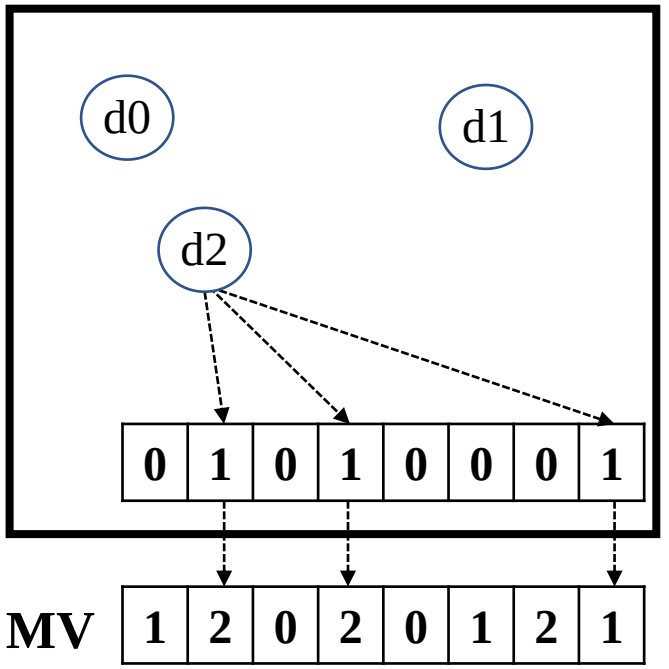
θ_J=0.6

(b) Counter array

JIG: Jaccard-Index Grouping

d0 = “GET /index.html HTTP/1.1”
d1 = “HTTProotadmin1234@”
d2 = “GET / HTTP/1.1 admin”
d3 = “HTTP root directory”
d4 = “GET root12, admin123@”
d5 = “GET root12, admin123@”

(a) Data stream



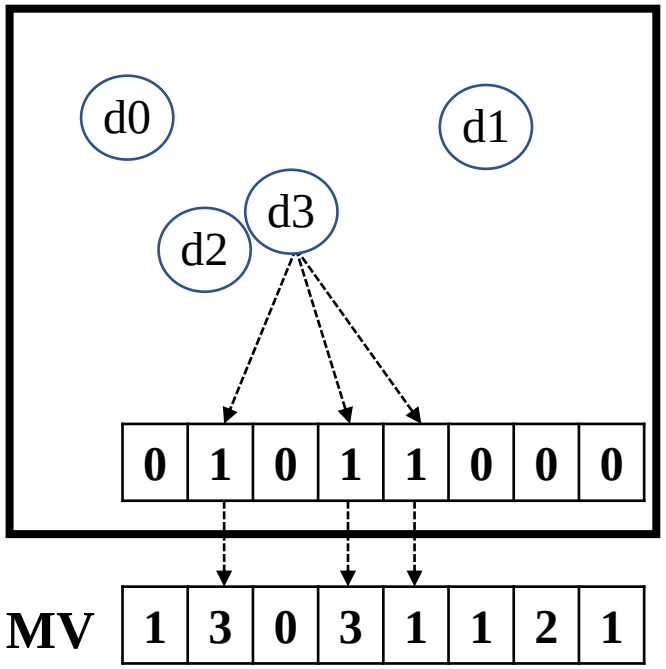
θ_J=0.6

(b) Counter array

JIG: Jaccard-Index Grouping

d0 = “GET /index.html HTTP/1.1”
d1 = “HTTProotadmin1234@”
d2 = “GET / HTTP/1.1 admin”
d3 = “HTTP root directory”
d4 = “GET root12, admin123@”
d5 = “GET root12, admin123@”

(a) Data stream



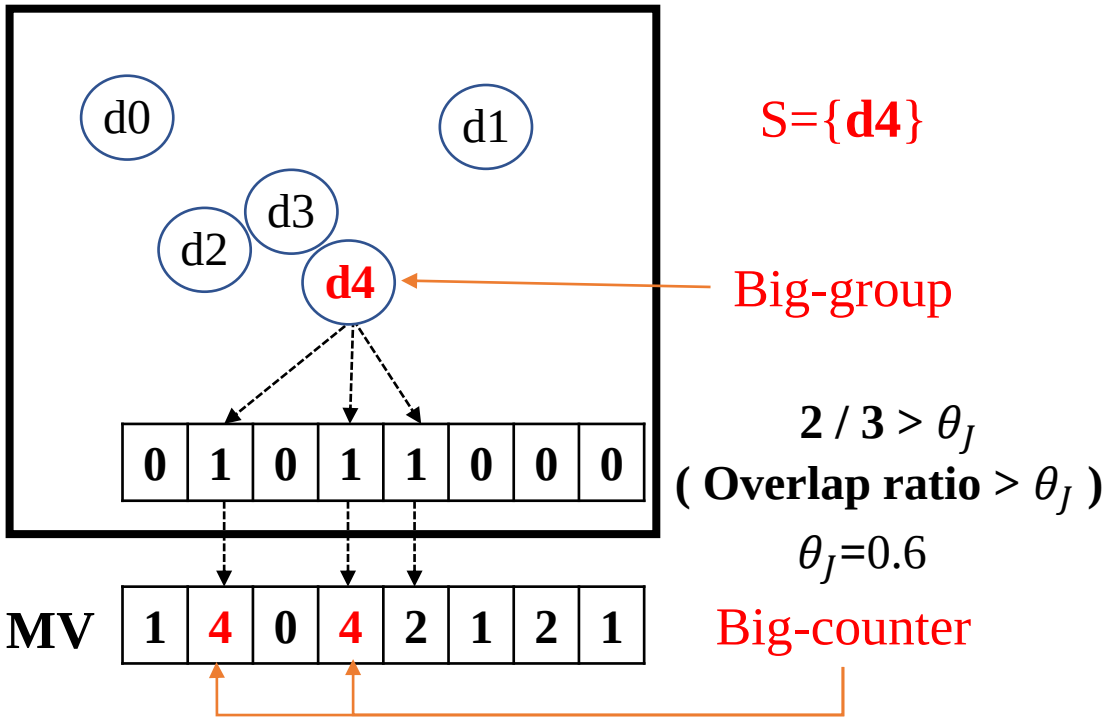
(b) Counter array

θ_J=0.6

JIG: Jaccard-Index Grouping

d0 = “GET /index.html HTTP/1.1”
d1 = “HTTProotadmin1234@”
d2 = “GET / HTTP/1.1 admin”
d3 = “HTTP root directory”
d4 = “GET root12, admin123@”
d5 = “GET root12, admin123@”

(a) Data stream



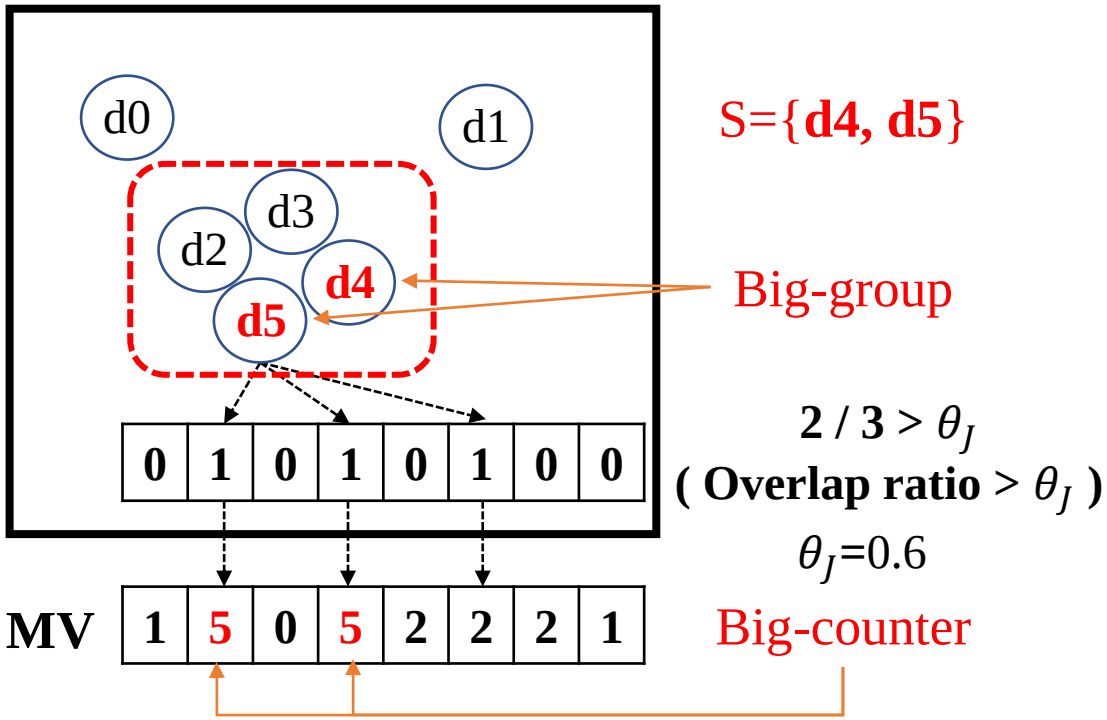
(b) Counter array

Mean + 6 * Sigma = 2.8
(Iterative Outlier Removal Algorithm)

JIG: Jaccard-Index Grouping

d0 = “GET /index.html HTTP/1.1”
d1 = “HTTProotadmin1234@”
d2 = “GET / HTTP/1.1 admin”
d3 = “HTTP root directory”
d4 = “GET root12, admin123@”
d5 = “GET root12, admin123@”

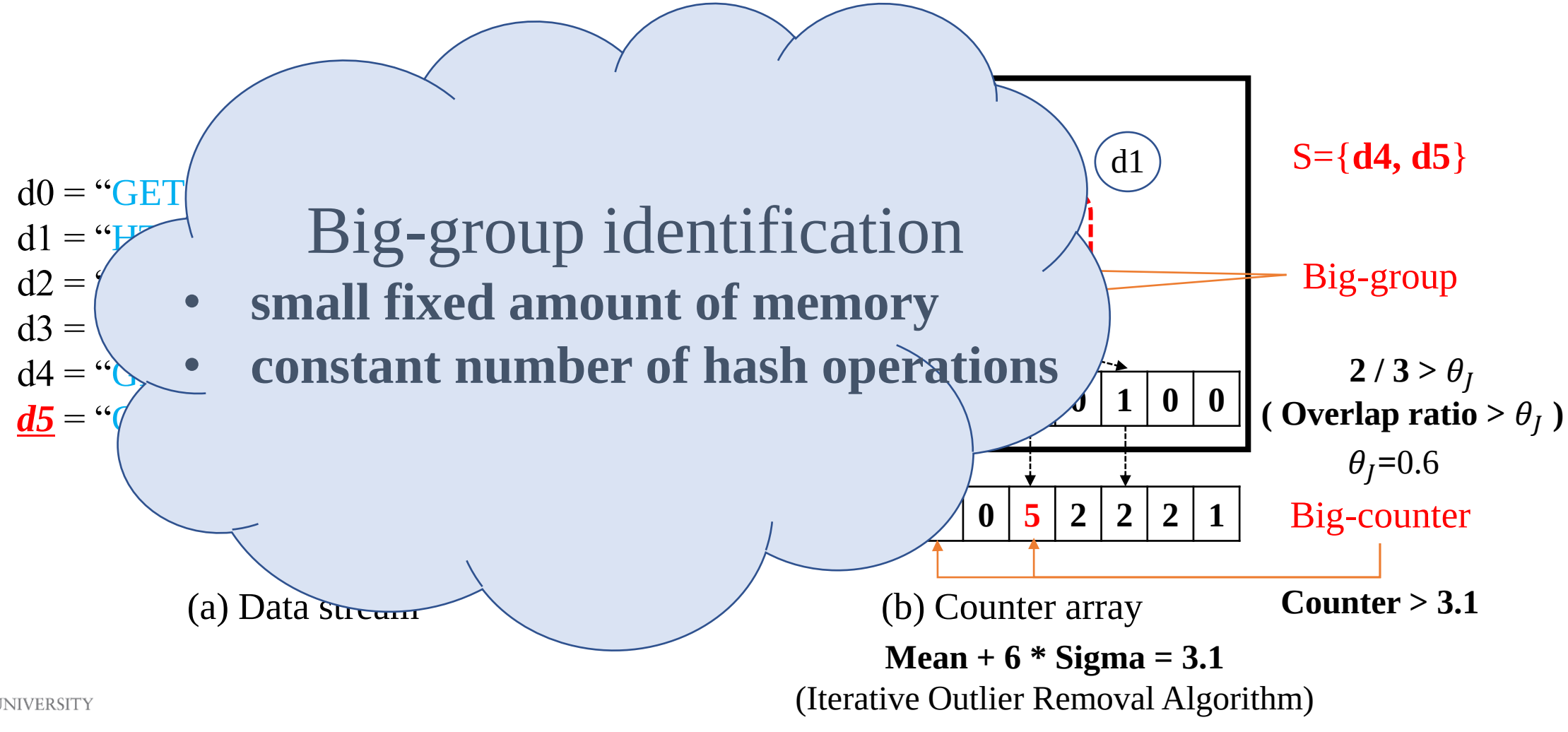
(a) Data stream



(b) Counter array

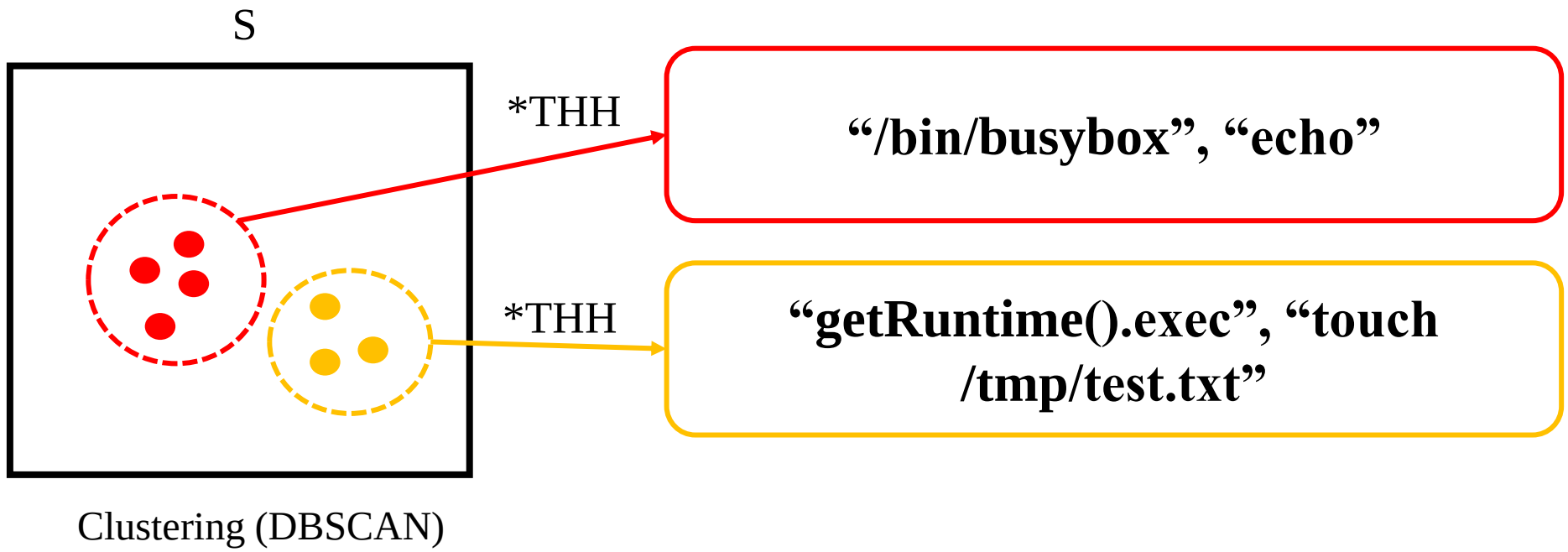
Mean + 6 * Sigma = 3.1
(Iterative Outlier Removal Algorithm)

JIG: Jaccard-Index Grouping



SG2: Signature-Group Generation

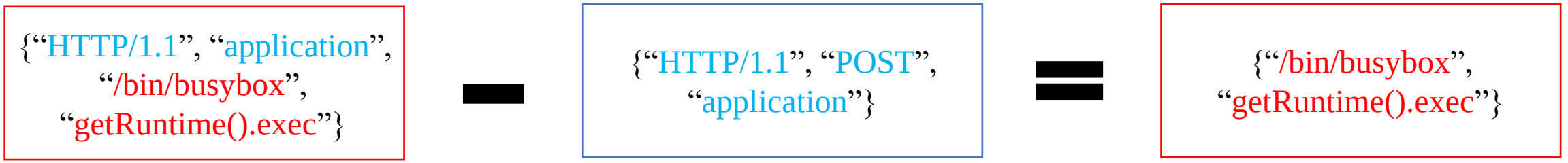
- Signature-group generation from S
  - Clustering & Signature extraction (Triple-Heavy-Hitter*)
  - Other schemes can be used instead



* “Zero-day signature extraction for high-volume attacks”, in IEEE/ACM Transactions on Networking, 2019.

AWL: Automatic WhiteListing

- AWL
  - (SG2 THH) – (Global THH)



(a) SG2 (b) AWL (c) GIPS

Experiments

- Dataset
  - SIM1-1 ~ SIM3-3
    - Simulated dataset for big-group identification
    - https://tinyurl.com/2gfz4f7v
  - IoT1~8
    - A.K.A. IoT23, https://www.stratosphereips.org/datasets-iot23
  - IDS1~8
    - A.K.A. CICIDS2017, https://www.unb.ca/cic/datasets/ids-2017.html
  - ISP1~3
    - Private dataset
    - Suspicious packets from IPS

Experiments

- SIM1-1 ~ SIM3-3
  - https://tinyurl.com/2gfz4f7v

SIM3-1

k \ m	2^{11}	2^{12}	2^{13}	2^{14}
32	P:0.83 R:1.00	0.83 1.00	0.83 1.00	0.83 1.00
64	P:0.99 R:1.00	0.99 1.00	0.99 1.00	0.99 1.00
128	P:1.00 R:1.00	1.00 1.00	1.00 1.00	1.00 1.00

Group rate: 0.1

SIM3-2

k \ m	2^{11}	2^{12}	2^{13}	2^{14}
32	0.15 0.96	0.15 0.98	0.15 0.99	0.15 0.99
64	0.94 0.98	0.94 0.99	0.94 0.99	0.94 1.00
128	1.00 0.99	1.00 0.99	1.00 0.99	1.00 1.00

Group rate: 0.01

SIM3-3

k \ m	2^{11}	2^{12}	2^{13}	2^{14}
32	0.01 0.14	0.01 0.14	0.01 0.16	0.03 0.52
64	0.00 0.00	0.95 0.15	0.98 0.46	0.99 0.76
128	0.99 0.05	1.00 0.49	1.00 0.75	0.99 0.87

Group rate: 0.001

Figure 6: Precision (P) and Recall (R) of JIG for SIM datasets with different big-group ratios, k, and m.

Experiments

- IoT1~8
  - IoT23, https://www.stratosphereips.org/datasets-iot23
  - Table4 for Precision & Recall

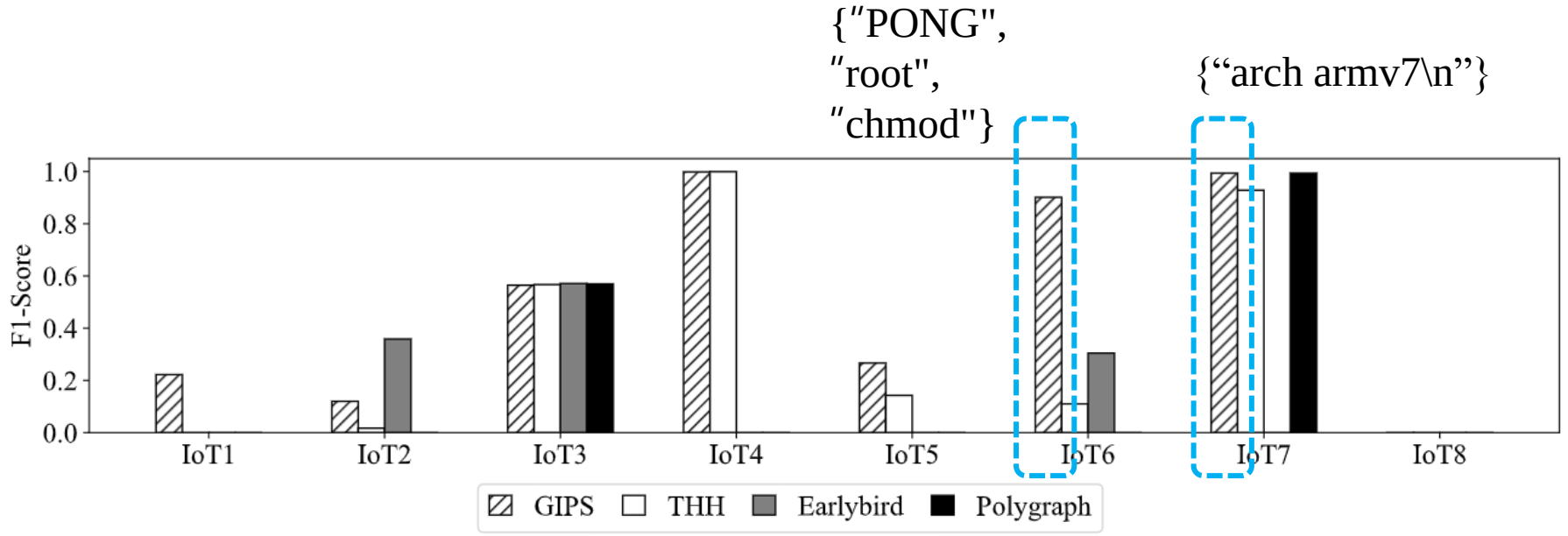


Figure 7: F1-Score of GIPS, THH [9], Earlybird [48], and Polygraph [39] for IoT datasets.

Experiments

- IoT1~8
  - IoT23, https://www.stratosphereips.org/datasets-iot23
  - Table4 for Precision & Recall

GIPS ≈ THH

Attack ratio = 0.99

No repetitive contents

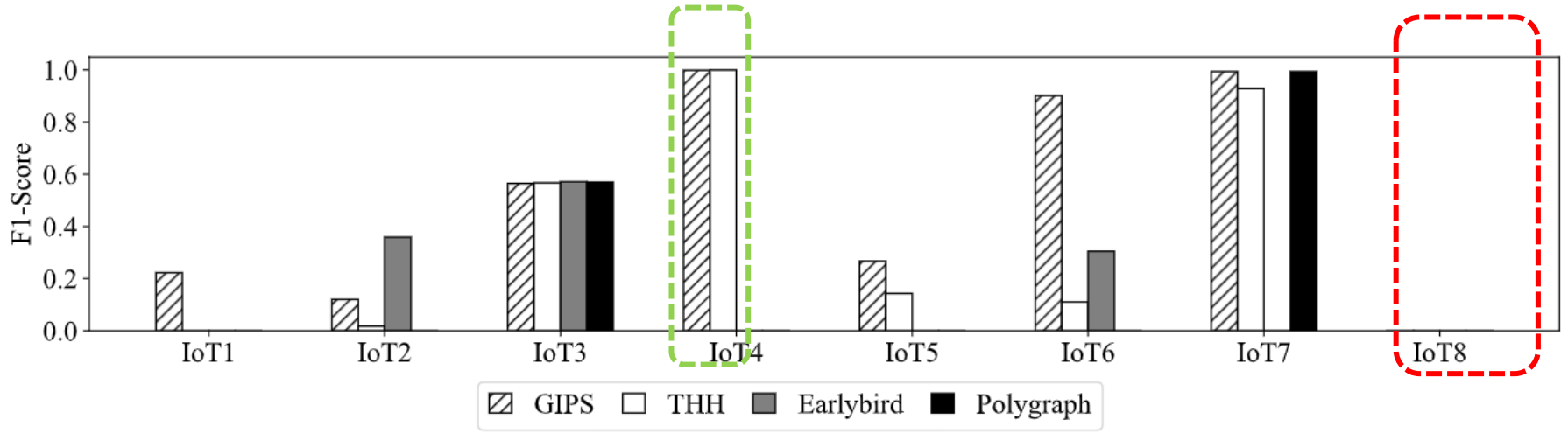


Figure 7: F1-Score of GIPS, THH [9], Earlybird [48], and Polygraph [39] for IoT datasets.

Experiments

- IDS1~8
  - CICIDS2017, https://www.unb.ca/cic/datasets/ids-2017.html
  - Table5 for Precision & Recall

{"/dv/vulnerabilities", "Cookie: security=low"}

{"GET / HTTP/1.0\r\n\r\n\r\n"}

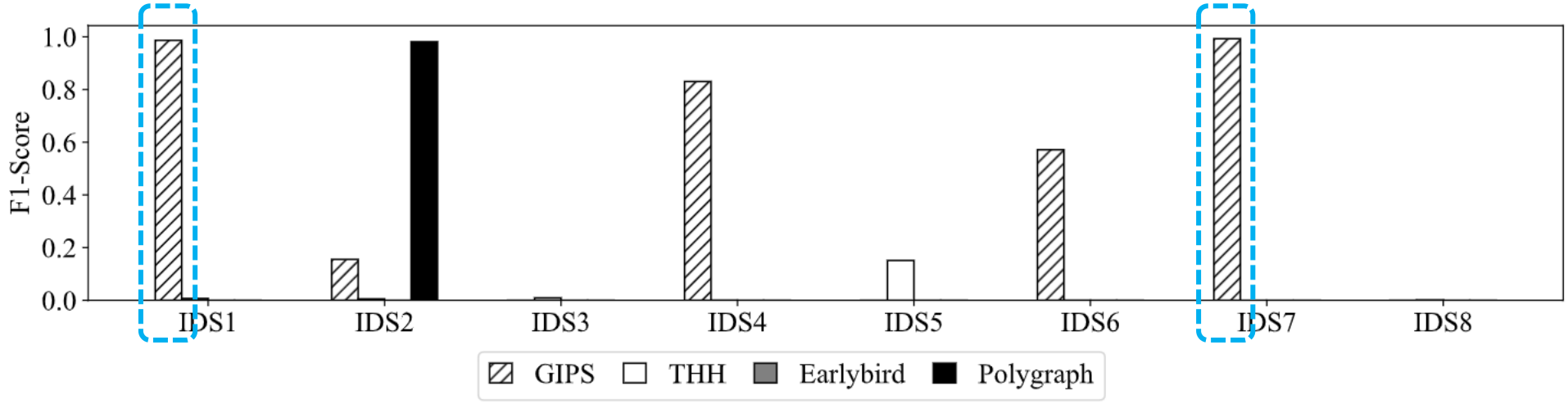


Figure 10: F1-Score of GIPS, THH [9], Earlybird [48], and Polygraph [39] for IDS datasets.

Experiments

- IDS1~8
  - CICIDS2017, https://www.unb.ca/cic/datasets/ids-2017.html
  - Table5 for Precision & Recall

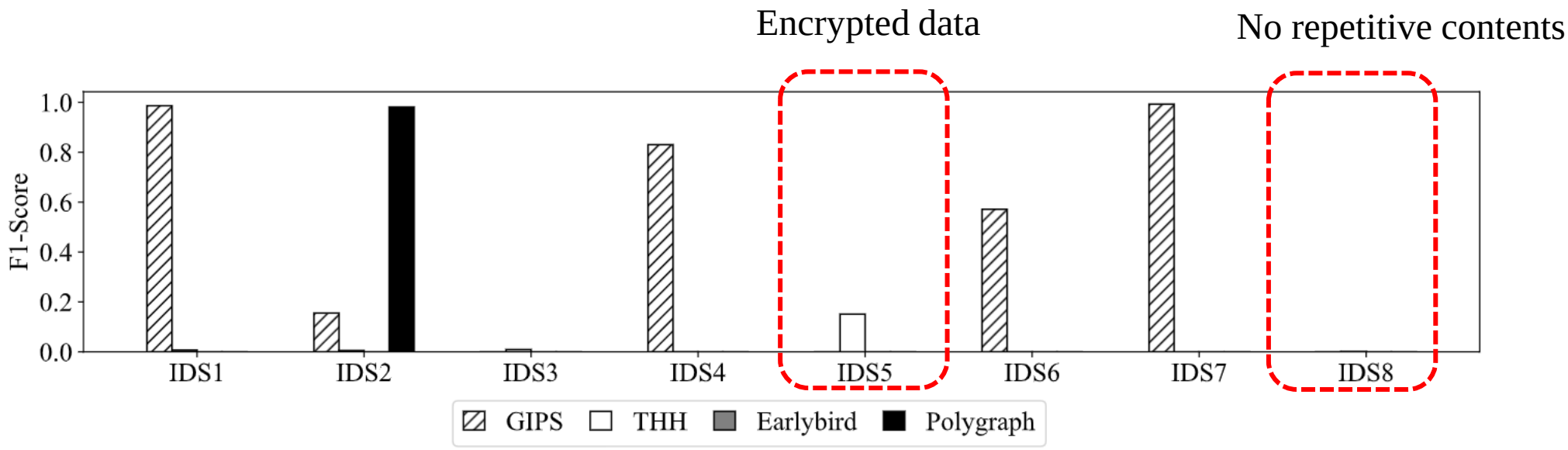


Figure 10: F1-Score of GIPS, THH [9], Earlybird [48], and Polygraph [39] for IDS datasets.

Experiments

- ISP1~3
  - Private dataset
  - Table6 for Precision & Recall

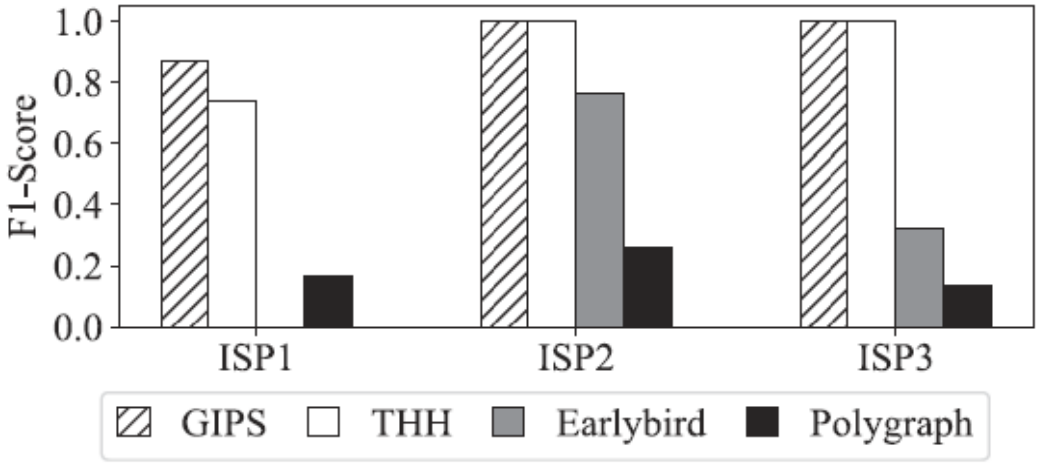
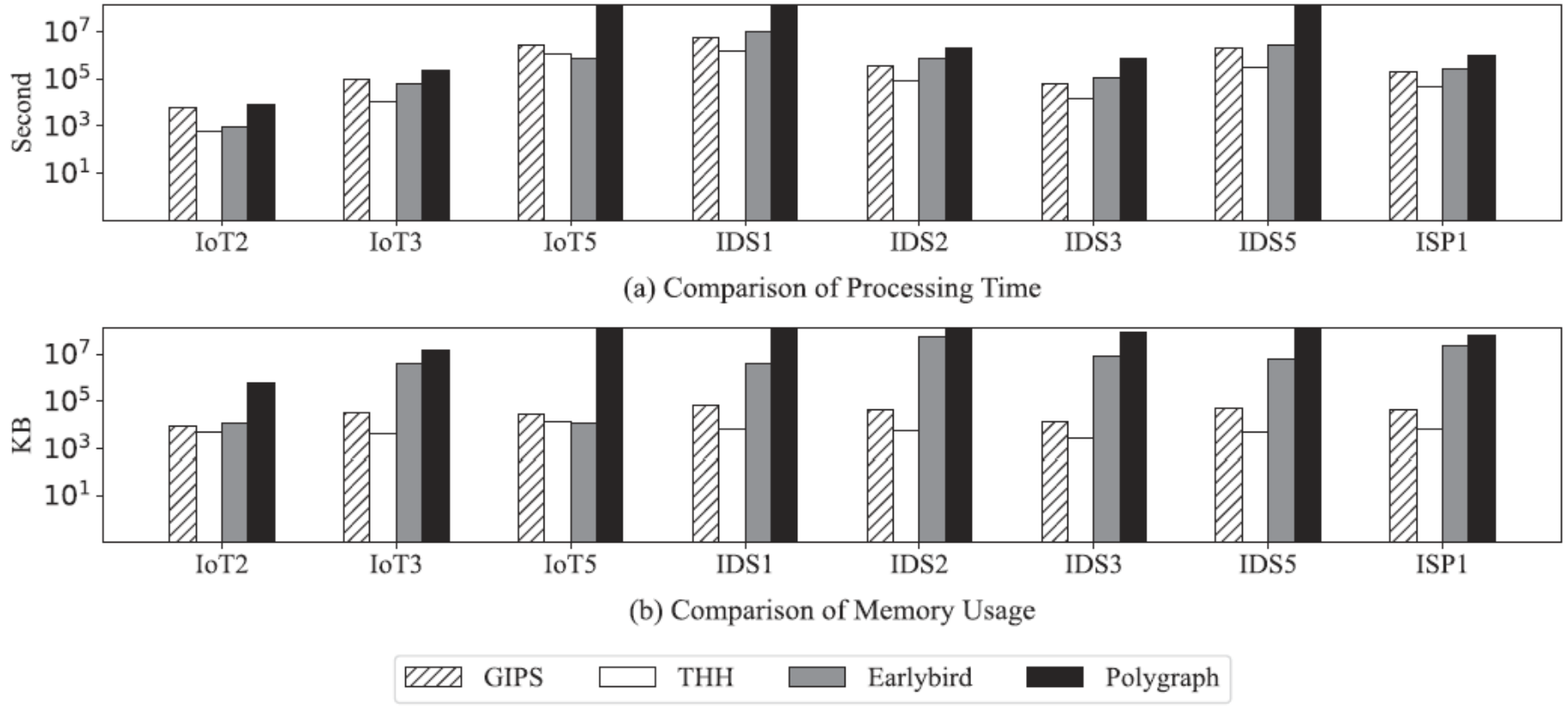


Table 7: GIPS Signatures and Attacks for ISP Datasets

Dataset	Signature	Related Attack
ISP1	*getRuntime().exec("touch /tmp/test.txt"), /bin/busybox*	CVE-2022-22963 [40] CVE-2017-17215 [37]
ISP2	0002736c0000ff	CVE-2010-1574 [36]
ISP3	*\x00\x00\x00DHIP\x00...* *"Port" : 37777,"RemoteVideo...*	Reflection attack [41]

Experiments

- Processing time and memory usage



Conclusion

- A new generative IDS/IPS to mitigate persistent zero-day attacks
- Fast and compact identification of big-groups from data streams
- Automatic generation of robust group-signatures
- Work from a high to a low attack ratio, for example from 0.99 to 0.001
- Limitations
  - Encrypted data, repetitive contents
  - Data analysis tool rather than real-time system

Thank You