CIDS:
Adapting Legacy Intrusion Detection Systems to the Cloud with Hybrid Sampling

Qingtang Xia*, Tianjia Chen†, Wei Xu‡

Institute for Interdisciplinary Information Sciences
Tsinghua University
*xqt13@mails.Tsinghua.edu.cn, †ctj2015@mail.Tsinghua.edu.cn, ‡ weixu@Tsinghua.edu.cn

2016-11-26

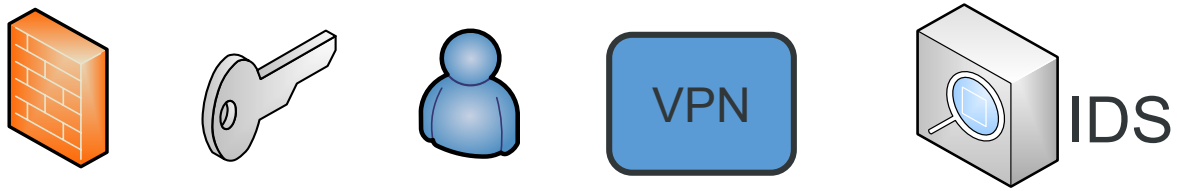
Outline

- Introduction
- CIDS overview
- Hybrid sampling strategy
- CIDS design and implementation
- Evaluation
- Conclusions

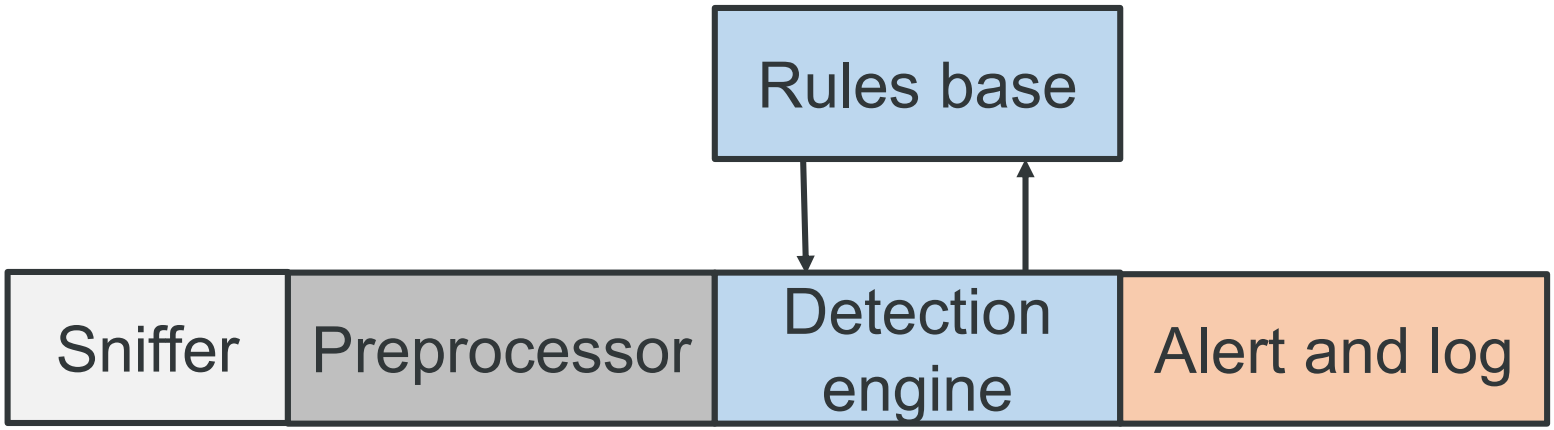
Security Challenges

Security requirement:
1. Confidentiality
2. Integrality
3. Availability

Threats:
1. Insider attacks
2. Outsider intrusions



IDS Mechanism



Problem:

1. Need too many computing, storage, and networking resources
2. Need a single point for deployment

Motivation

- **To decrease the IDS resources consumption**

Motivation

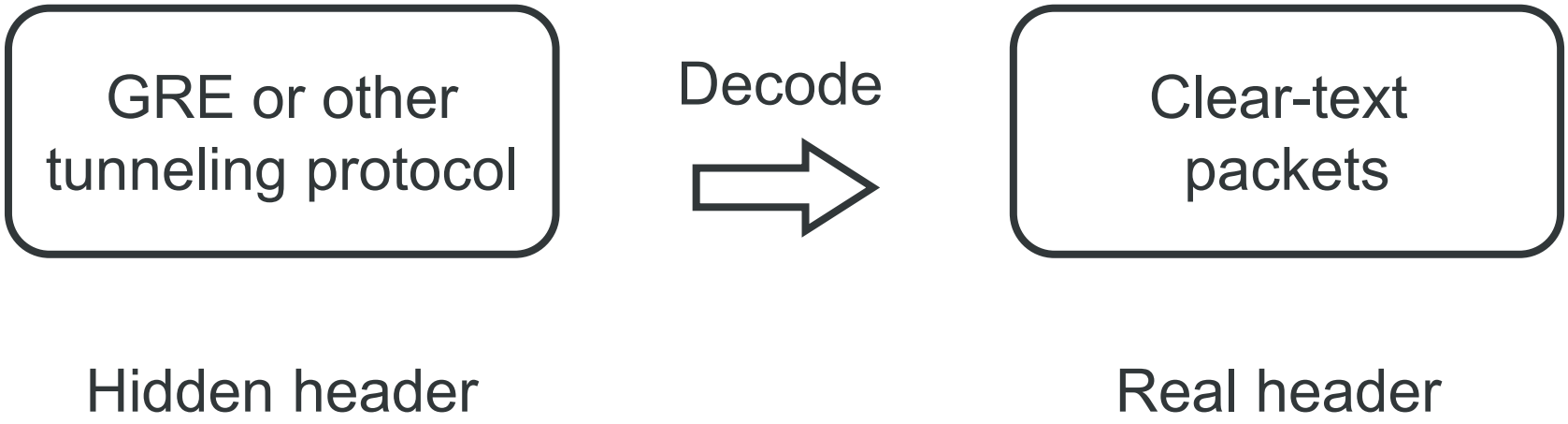
- **To detect inside and outside intrusions**

Motivation

- **To adapt the dynamicity of the cloud**

Motivation

- **To get high detection efficiency**



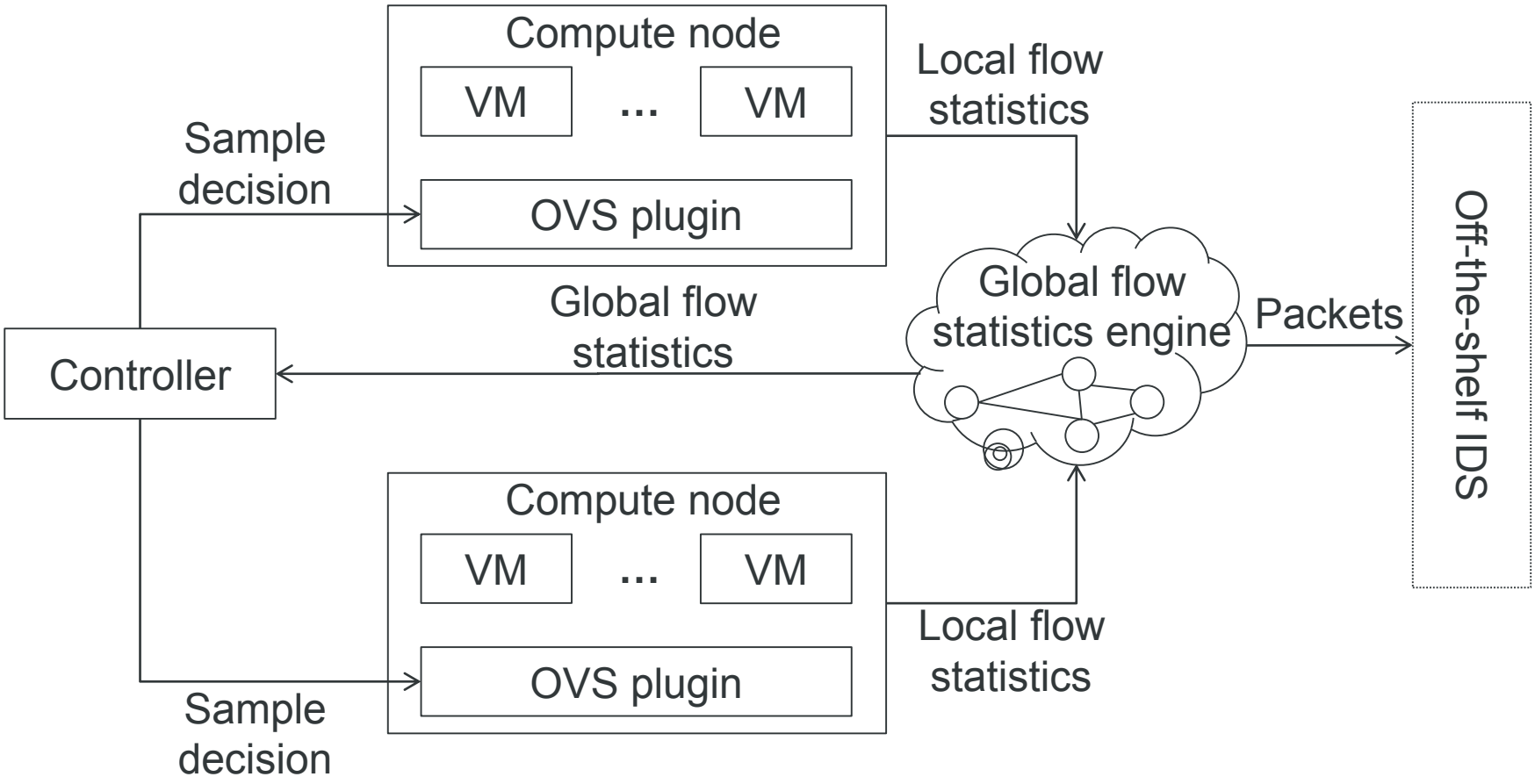
Main contribution

- Hybrid sampling algorithm based on local and global flow statistics
- Provide an SDN-based packet collection and monitoring mechanism
- Evaluate CIDS using real world attack traces in a production cloud

Outline

- Introduction
- CIDS overview
- Hybrid sampling strategy
- CIDS design and implementation
- Evaluation
- Conclusions

CIDS Architecture



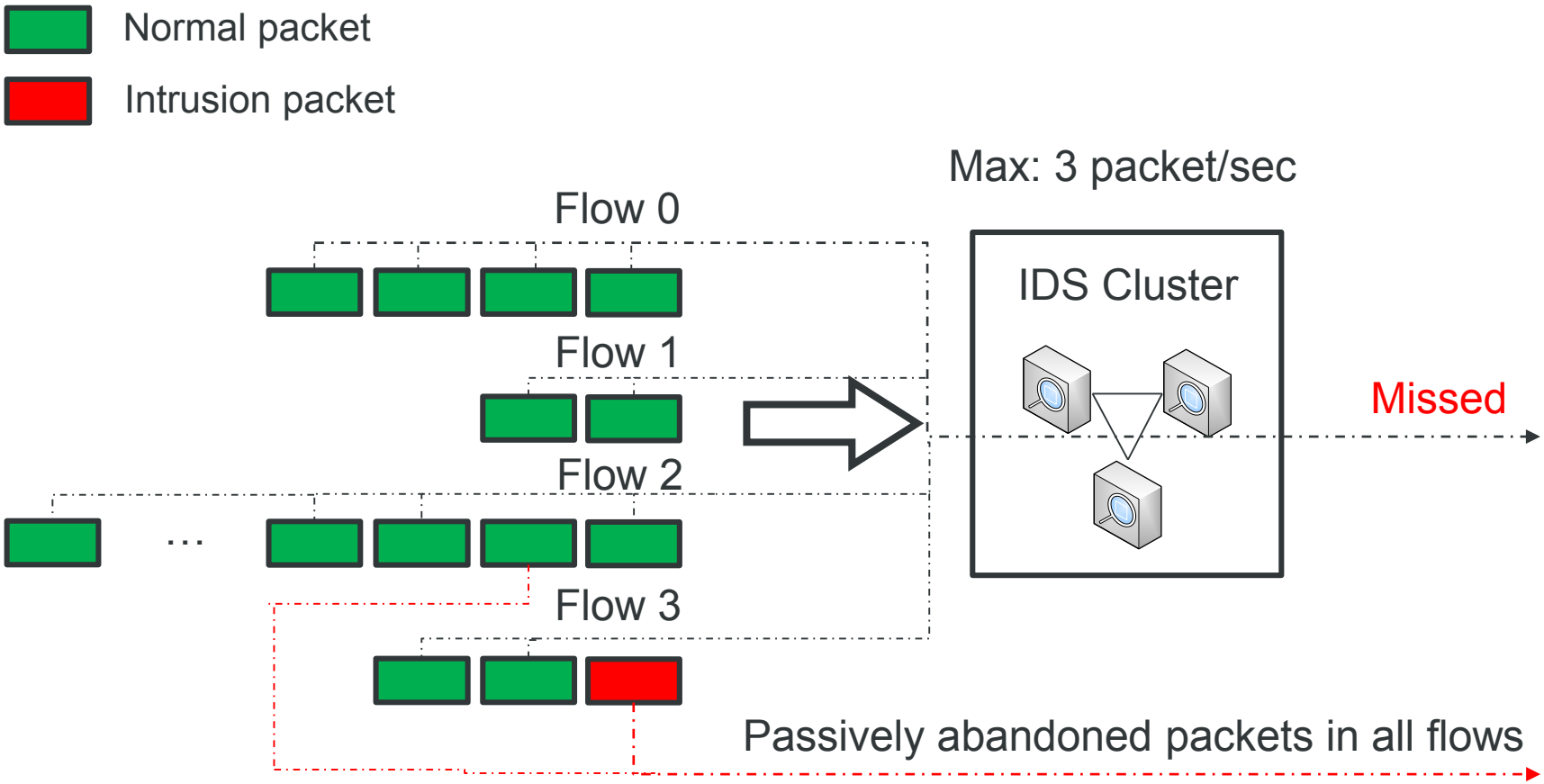
Main Ideas

- Eliminate unnecessary traffic as earlier as possible
- Local and global flow statistics
- IDS-aware sampling mechanism

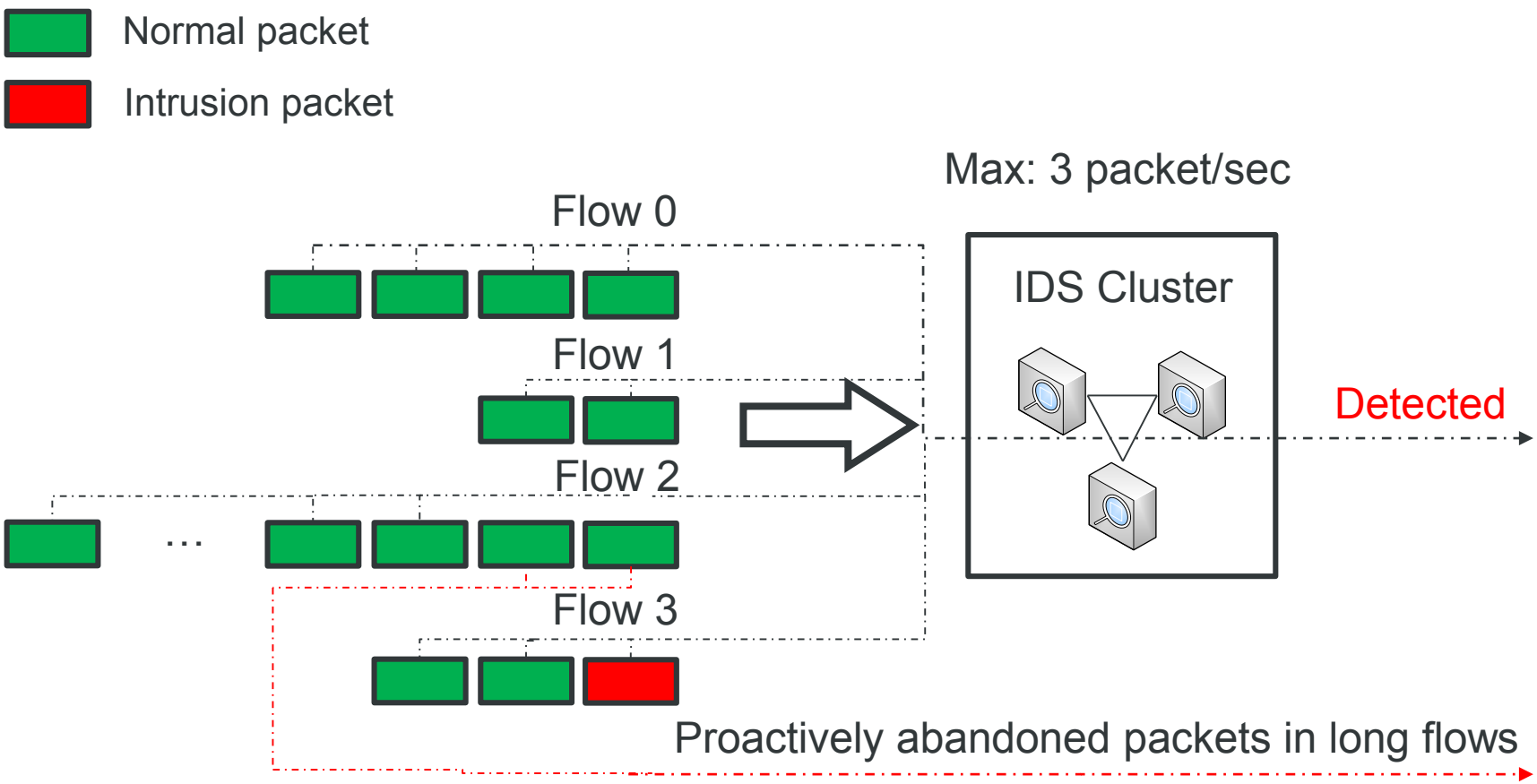
Outline

- Introduction
- CIDS overview
- Hybrid sampling strategy
- CIDS design and implementation
- Evaluation
- Conclusions

Overloaded IDS



Overloaded IDS (cont.)



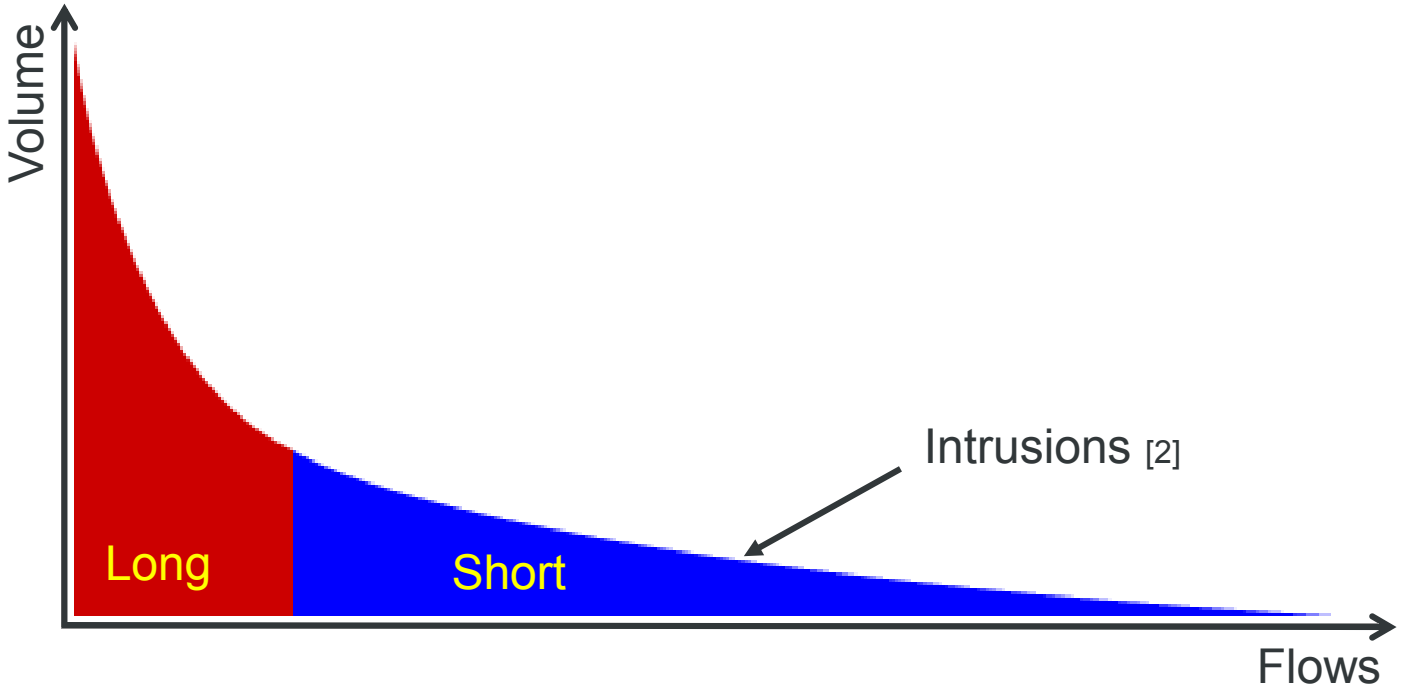
Maximum sample rate

$$p_M = U_{IDS} \times U_{network}$$

U_{IDS} : Utilization of IDS cluster

$U_{network}$: Network utilization

Flows distribution



[1]

[1] Jaeyeon Jung, Vern Paxson, Arthur W Berger, and Hari Balakrishnan. Fast portscan detection using sequential hypothesis testing. In *Security and Privacy, 2004. Proceedings. 2004 IEEE Symposium on*, pages 211– 225. IEEE, 2004.
[2] Arno Wagner and Bernhard Plattner. Entropy based worm and anomaly detection in fast IP networks. In 14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise (WETICE'05), pages 172–177. IEEE, 2005.

Local sampling algorithm

1. If no large-scale anomalies happen, continue;

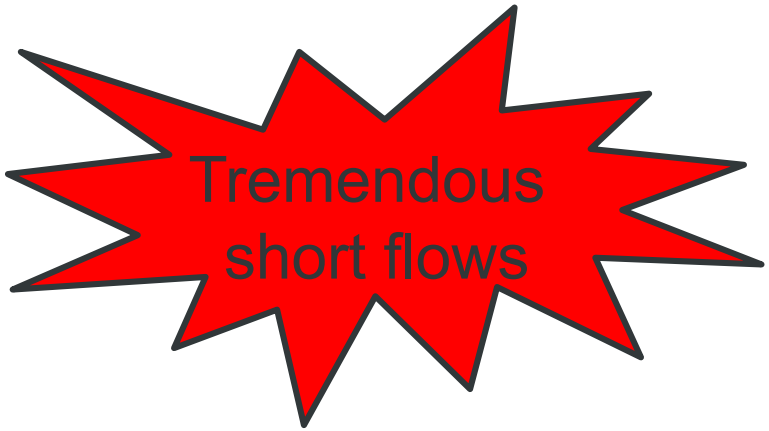
2. For each arriving flow f, check whether f is marked as intrusions. If yes, then set p_f=100%. If not, then continue;

3. For no marked f, calculate p_f

$$p_f = \frac{1}{L_{cur}}$$

Large-scale anomalies

- DDoS
- BotNet
- Worm
- Distributed Scan
- …



Feature entropies of large-scale anomalies

type	H(SRCIP)	H(SRCP)	H(DESTIP)	H(DESTP)
DDoS	↑	–	↓	–
Port scan	–	–	↓	↑
Network scan	–	–	↑	↓
Worm	–	–	↑	↓

Global sampling algorithm

1. Get the division of feature f_x;

$$f_x=\{(x_i, n_i), i=1, 2, \dots, N\}$$

2. Calculate entropies for each feature;

$$H(f_x)=\sum_{i=1}^{N}\frac{n_i}{|S|}log_2\frac{n_i}{|S|}\ ,\ where\ |S|=\sum_{i=1}^{N}\frac{n_i}{|S|}$$

3. Calculate expectation deviation $\xi(X)$;

$$\xi(X)=\frac{X-E(X)}{\delta(X)}$$

Global sampling algorithm (cont.)

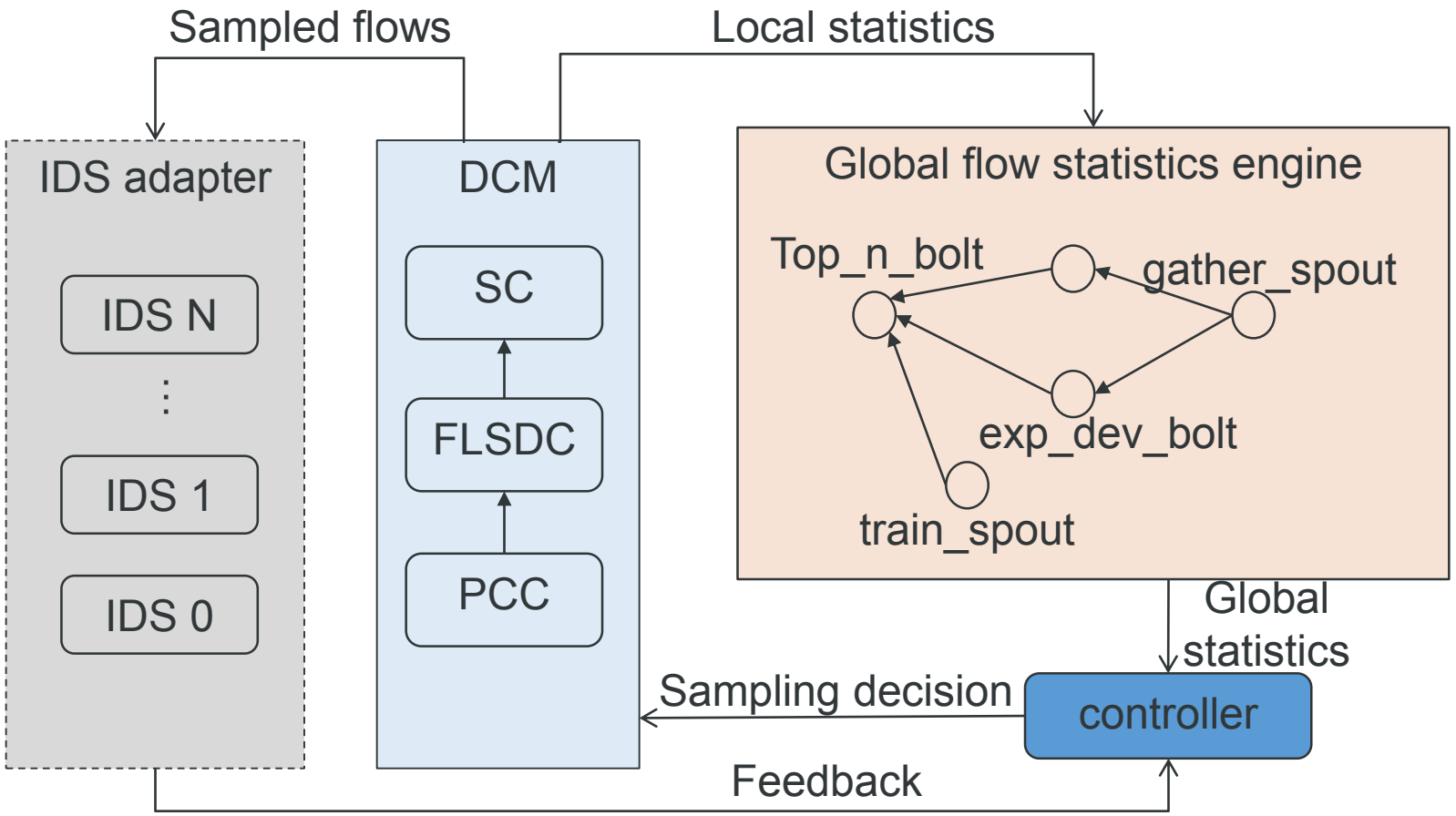
4. if $\xi(X) > \xi_{threshold}(X)$, then sort f_x with n_i, and get top *n* flows

5. Sample top n flows with p_M

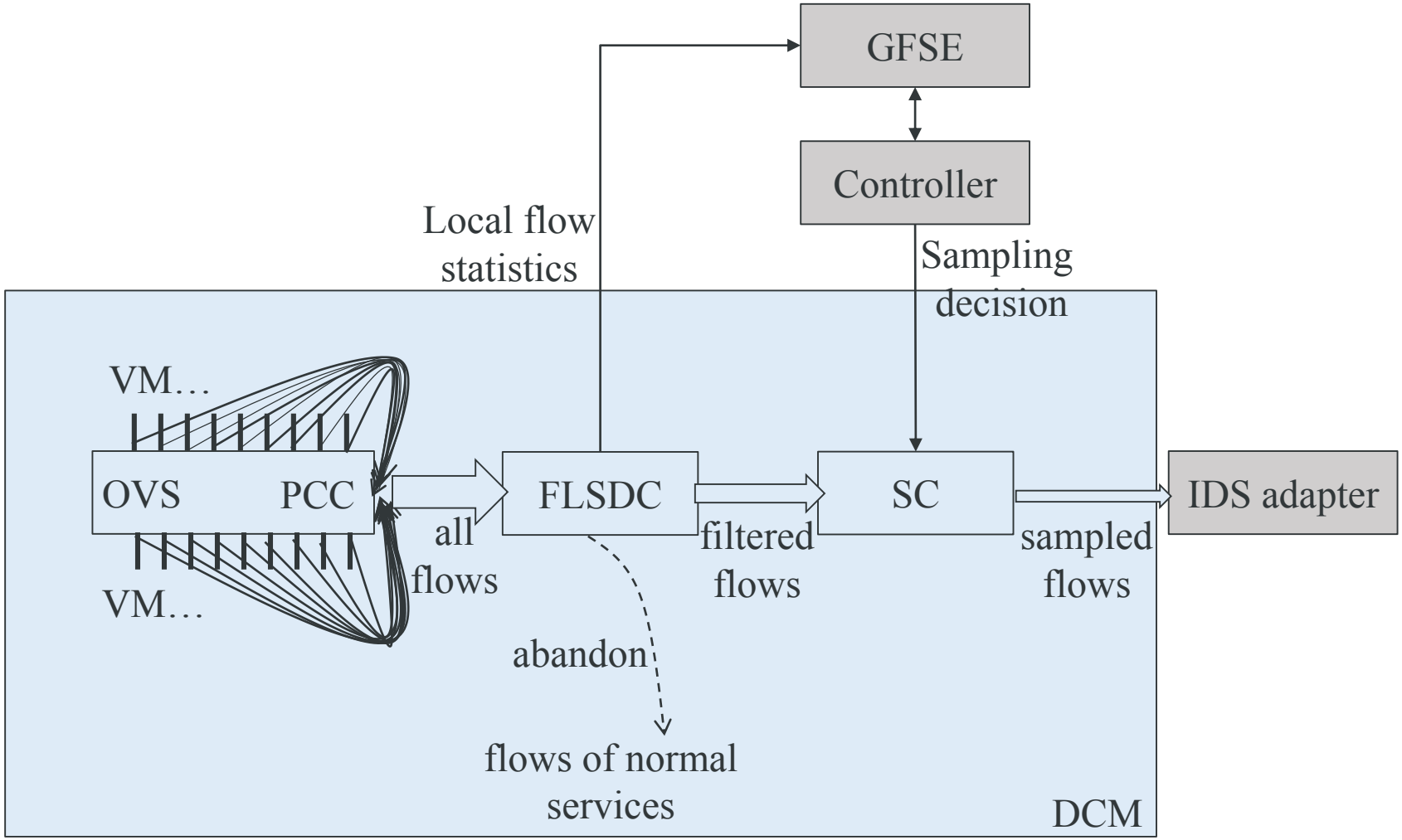
Outline

- Introduction
- CIDS overview
- Hybrid sampling strategy
- CIDS design and implementation
- Evaluation
- Conclusions

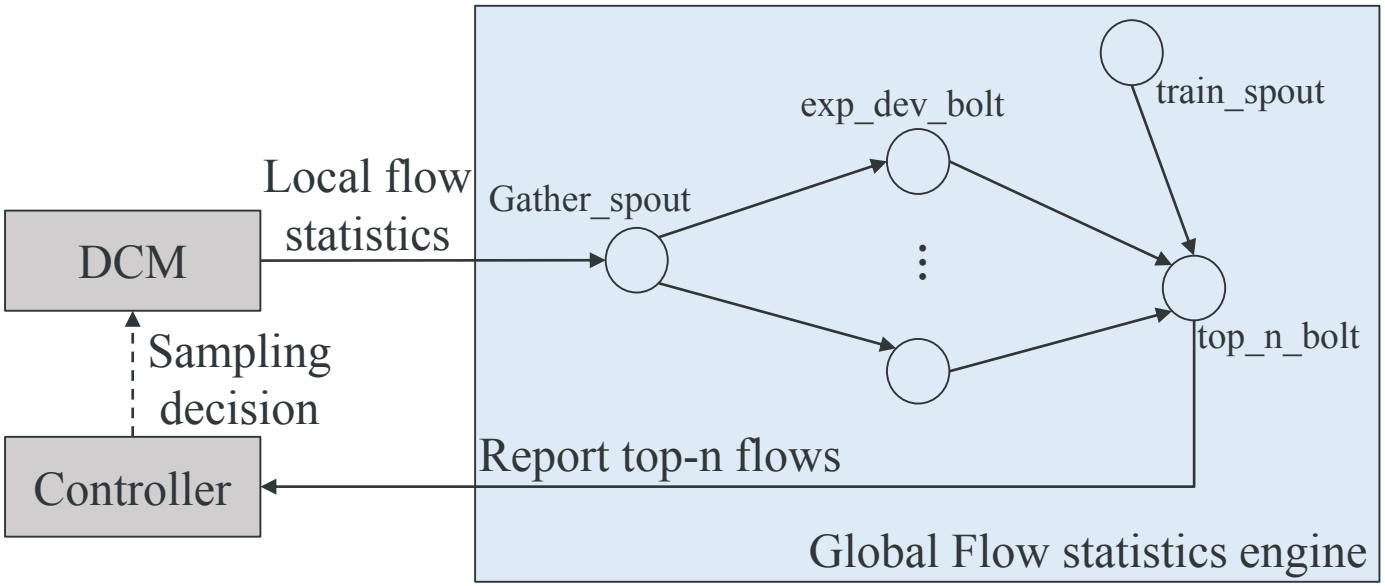
Implementation of CIDS



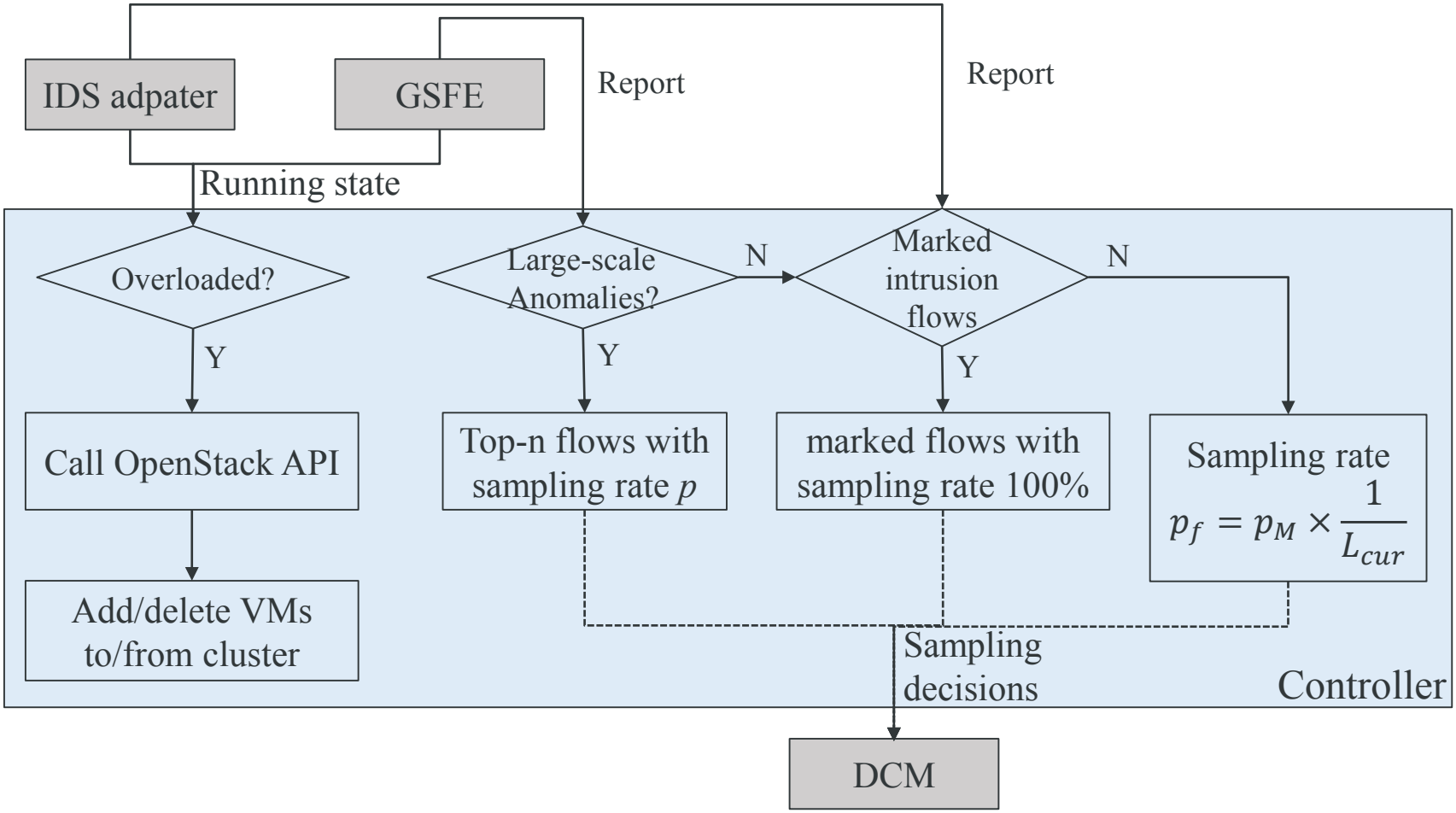
Data Collection Module



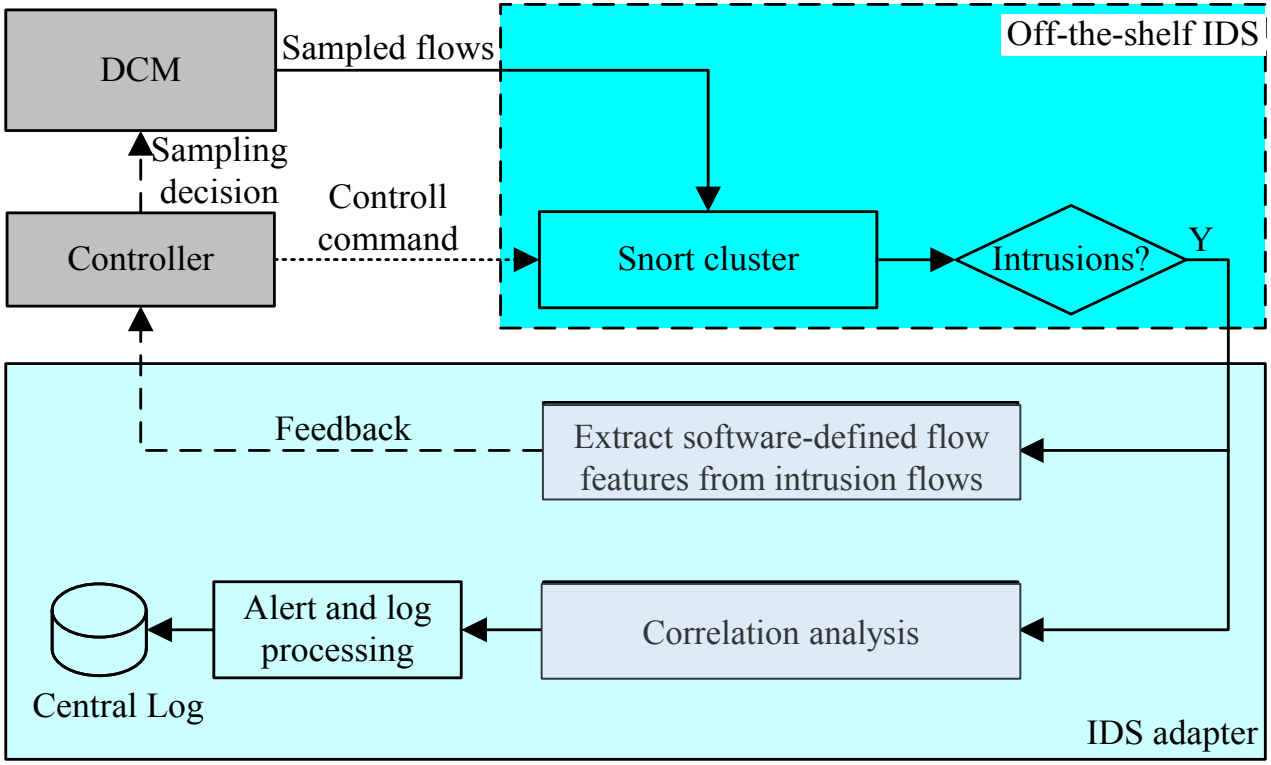
Global Flow Statistics Engine



Controller


IDS adpater
GSFE
Report
Report
Running state
Overloaded?
Large-scale Anomalies?
N
Marked intrusion flows
N
Y
Y
Y
Call OpenStack API
Top-n flows with sampling rate p
marked flows with sampling rate 100%
Sampling rate
$p_f = p_M \times \frac{1}{L_{cur}}$
Add/delete VMs to/from cluster
Sampling decisions
Controller
DCM

IDS adapter


DCM
Sampled flows
Off-the-shelf IDS
Sampling decision
Controll command
Controller
Snort cluster
Intrusions?
Y
Feedback
Extract software-defined flow features from intrusion flows
Alert and log processing
Correlation analysis
Central Log
IDS adapter

Outline

- Introduction
- CIDS overview
- Hybrid sampling strategy
- CIDS design and implementation
- Evaluation
- Conclusions

Experiment Setup

Configuration of our OpenStack cloud platform

Item	Configure
Nodes	125 2U servers
CPU	12 cores
Memory	128 GB DRAM
Storage	10 TB disks
Ethernet Interface	Two of 10GE ports, four of 1GE ports

Experiment Setup (cont.)

Configuration of Software Environment

Item	Configure
Intrusion detection cluster	31 VMs
Storm cluster	16 VMs
Controller	1 VM
Data Collection Module	Every compute node of OpenStack
Flavor of Virtual Machine	2 vcpu, 4096 MB vmemory, 100 GB vdisk, 100 Mbps vinterface

Experiment Setup (cont.)

Category	NO.	Attack name	Dataset	attack packets	Enlarge factor
User to Root	1	loadmodule	KDD'99	10×23	10
Remote to Local	2	imap		10×84	10
	3	named		10× 47	10
	4	ssttrojan		10×60	10
DoS	5	teardrop		512× 7 ×254×1	512
	6	mailbomb		512×1×1×667	512
	7	land		512×9×254×2	512
	8	synflood	CAIDA 2014	1,440,562	1
Mix	9	nmap	KDD'99	256×38×254×3	256
	10	ipsweep		256×9×254×1	256
Worm	11	Code red II	CAIDA 2014	5,609,294	1

Evaluation traces from KDD'99 and CAIDA 2014:

(1) Training dataset and background traffic

First and third weeks of KDD

(2) Produce attack traffic

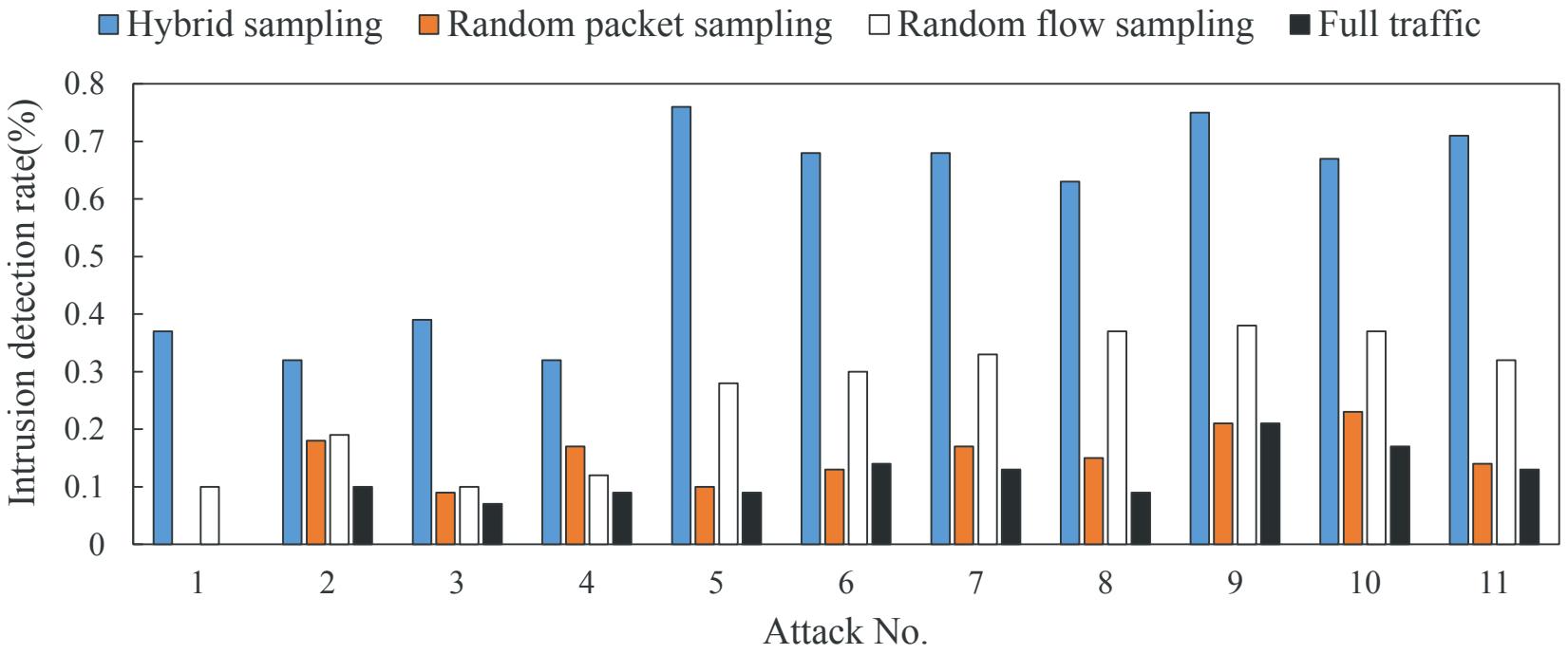
Forth and fifth weeks of KDD and CAIDA 2014

(3) Mix traffic

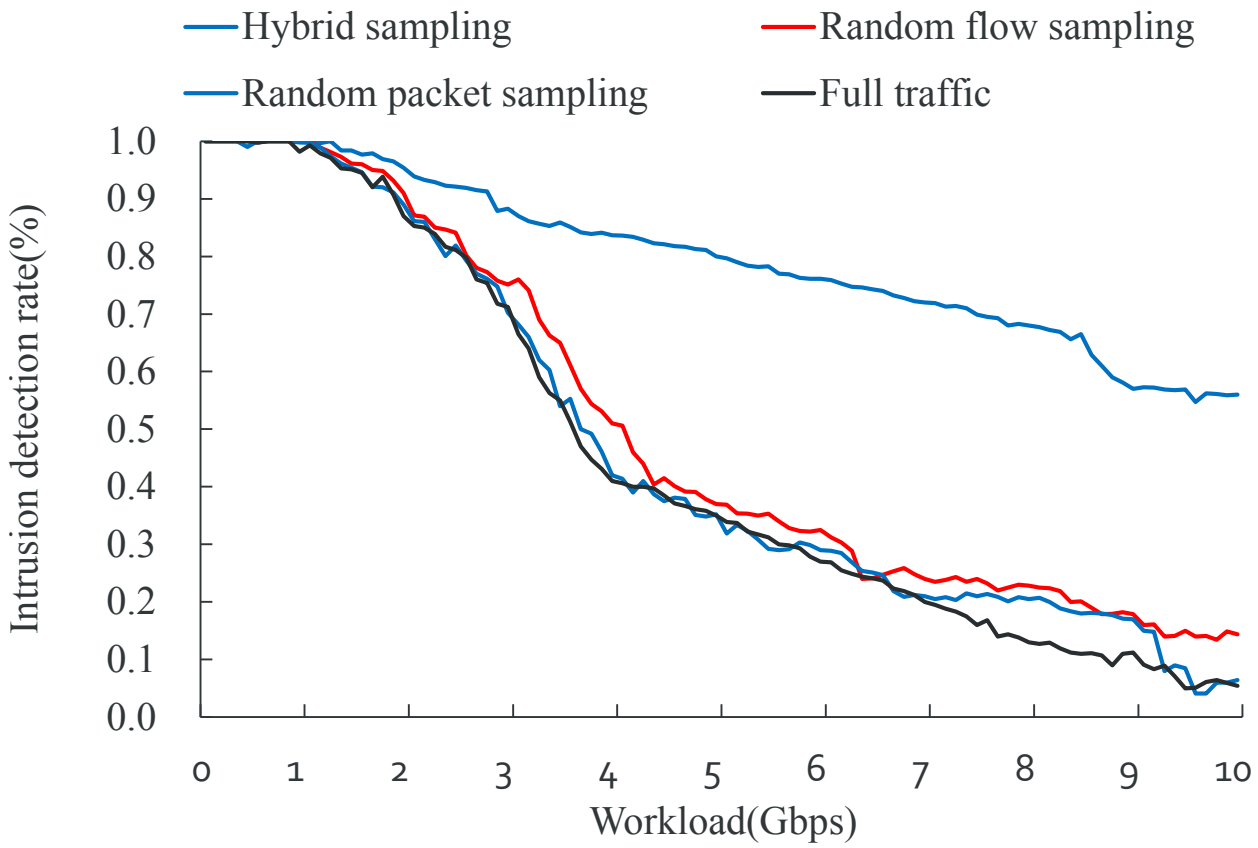
Sampled valid attack packets

	1	2	3	4	5	6	7	8	9	10	11
Total attack packets	231	840	470	600	910,336	341,504	2,340,864	1,440,562	1,853,184	146,304	5,609,294
CIDS Hybrid Sampling	70	320	193	200	491,581	167,337	1,193,841	619,442	1,037,783	86,319	3,477,762
Random Packet Sampling	0	130	32	102	118,344	40,980	234,086	129,651	222,382	20,483	617,022
Random Flow Sampling	32	220	85	71	236,687	64,886	421,356	273,707	333,573	43,891	1,290,138

Intrusion detection rate



Stability of IDR under different loads



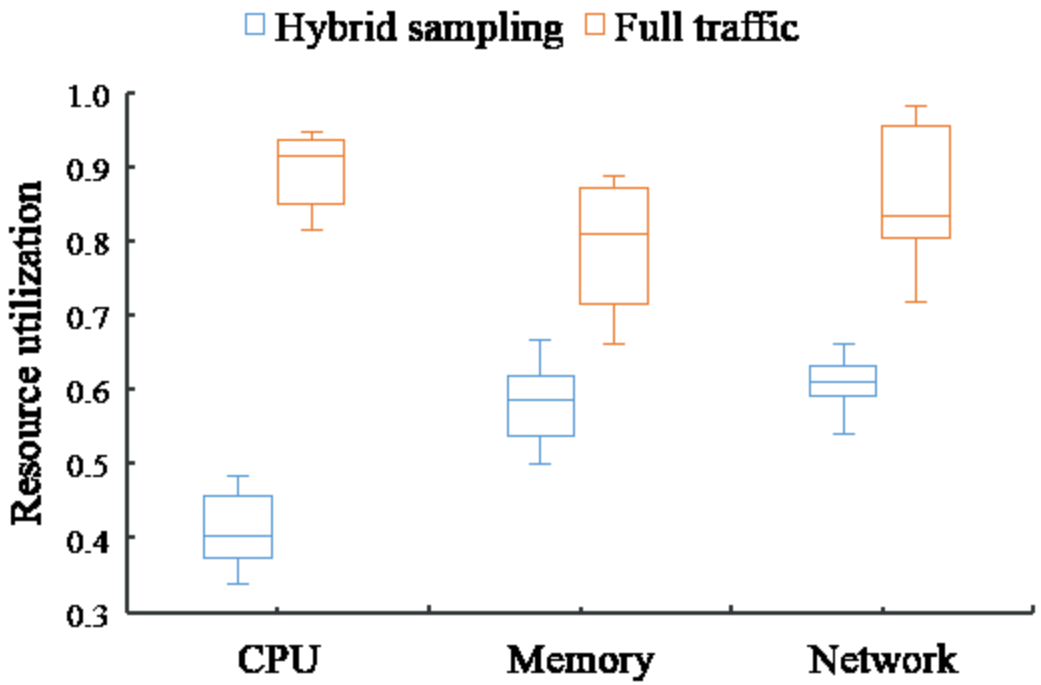
Detection Latency

	DoS	U2R	R2L	Probe	Worm
Hybrid sampling	3.45	0.23	0.89	3.03	3.76
Random flow sampling	9.10	1.12	2.90	12.94	10.38
Random packet sampling	13.31	1.47	3.16	13.22	16.97
Full traffic	15.95	1.52	3.48	13.91	17.0

Small scale attacks: CIDS get short detection latency

Large scale attacks: CIDS need long detection latency

Performance of IDS Cluster



Outline

- Introduction
- CIDS overview
- Hybrid sampling strategy
- CIDS design and implementation
- Evaluation
- Conclusions

Conclusions

- Traditional IDS hard to detect intrusions in the cloud
- Combine SDN based data collection with IDS-aware sampling mechanisms
- Demonstrate the effectiveness on a production cloud

Thanks!
Q&A