CAB: A Reactive Wildcard Rule Caching System for Software-Defined Networks

Bo Yan, Yang Xu, Hongya Xing
Kang Xi, H. Jonathan Chao

August 22, 2014
Reactively Caching Rules on Demand

Controller

Switch

Rule Set
Reactively Caching Rules on Demand

Controller

Rule Set

... Install at a time ...

Switch
Reactively Caching Rules on Demand

Controller

Install at a time

Switch

Rule Set

TCAM
Reactively Caching Rules on Demand

Controller

Switch

Rule Set

TCAM
Reactively Caching Rules on Demand

Controller

Install on demand

Switch

Rule Set

TCAM
Reactively Caching Rules on Demand

Controller

Switch

Install on demand

Rule Set

TCAM
Wildcard rules enables:
- Natural intention of managing flows aggregately
- Higher reusability for each rule
- Easy update of policies
Wildcard rules enables:
- Natural intention of managing flows aggregately
- Higher reusability for each rule
- Easy update of policies

Locality of Traffic

NYC Dept.Edu (DoE) Data Center Traces
Wildcard rules enables:
- Natural intention of managing flows aggregately
- Higher reusability for each rule
- Easy update of policies
Challenge: Wildcard Rule Dependency

Switch Mem

F₁ (Src IP) → Rule set → F₂ (Dst IP)

1
2
3
Challenge: Wildcard Rule Dependency

Rule set

Switch Mem
Challenge: Wildcard Rule Dependency

Rule set

Wrong matching!

Switch Mem
Challenge: Wildcard Rule Dependency

Rule set

Switch Mem

Dependency has chain reaction

Wrong matching!
Challenge: Wildcard Rule Dependency

Dependency has chain reaction

Wrong matching!
Challenge: Wildcard Rule Dependency

Dependency has chain reaction

Rule set

Switch Mem

Hypothetical
Challenge: Wildcard Rule Dependency

Dependency has chain reaction

Rule set

Switch Mem

Caching 100s dependent rules
Cache all dependent rules
- Memory explosion
Methods to Accommodate Rule Dependency

Cache all dependent rules
- Memory explosion

Switch Mem

$F_1$ (Src IP) $F_2$ (Dst IP)
Methods to Accommodate Rule Dependency

Cache all dependent rules
- Memory explosion
Methods to Accommodate Rule Dependency

Cache all dependent rules
- Memory explosion

Cache exact match rules ([DevoFlow][Ethane])
- Frequent rule installations

Switch Mem

F₂ (Dst IP)
F₁ (Src IP)
Cache all dependent rules
- Memory explosion

Cache exact match rules [DevoFlow][Ethane]
- Frequent rule installations
Methods to Accommodate Rule Dependency

Cache all dependent rules
- Memory explosion

Cache exact match rules [DevoFlow][Ethane]
- Frequent rule installations

Split rule set and cache micro rules [Smart Rule Cache][DIFANE]
- Significantly larger rule set
Methods to Accommodate Rule Dependency

Cache all dependent rules
  - Memory explosion

Cache exact match rules [DevoFlow][Ethane]
  - Frequent rule installations

Split rule set and cache micro rules [Smart Rule Cache][DIFANE]
  - Significantly larger rule set
Methods to Accommodate Rule Dependency

Cache all dependent rules
- Memory explosion

Cache exact match rules [DevoFlow][Ethane]
- Frequent rule installations

Split rule set and cache micro rules [Smart Rule Cache][DIFANE]
- Significantly larger rule set
Methods to Accommodate Rule Dependency

Cache all dependent rules
- Memory explosion

Cache exact match rules [DevoFlow][Ethane]
- Frequent rule installations

Split rule set and cache micro rules [Smart Rule Cache][DIFANE]
- Significantly larger rule set

**Lack of efficiency in switch memory use**
- more cache miss at switch
- higher controller load, control bandwidth
- longer flow setup delay
Cache all dependent rules
- Memory explosion

Cache exact match rules [DevoFlow][Ethane]
- Frequent rule installations

Split rule set and cache micro rules [Smart Rule Cache][DIFANE]
- Significantly larger rule set

Problem: how to accommodate rule dependency with efficient mem use?
Solution: CAChing rules in Buckets (CAB)

CAB Controller

Rules

Rule set (Controller)
Solution: CAChing rules in Buckets (CAB)

CAB Controller

Buckets

Rules

Rule set (Controller)
Solution: CAChing rules in Buckets (CAB)

CAB Controller

- Buckets
- Rules

Bucket Filter

Flow Table

OpenFlow Switch

Rule set (Controller)

- F1
- F2

- A
- B
- C
- D
- E
- F
- G
- H
- I

- 6
- 1
- 2
- 3
- 4
- 5
- 7
- 8
- 9

8/24/2014
Solution: CAChing rules in Buckets (CAB)
Solution: CAChing rules in Buckets (CAB)

Cache Miss at Bucket Filter
Solution: CAChing rules in Buckets (CAB)

Cache Miss at Bucket Filter
Install bucket F and rule 3 & 4
Solution: CAchering rules in Buckets (CAB)

Cache Miss at Bucket Filter
Install bucket F and rule 3 & 4
f₁ is set up
Solution: CAChing rules in Buckets (CAB)

Matched Bucket F, and Rule 3
Solution: CAChing rules in Buckets (CAB)

Matched Bucket F, and Rule 3

Switch Mem

8/24/2014
Solution: CAching rules in Buckets (CAB)
Cache Miss at Bucket Filter
Install bucket C and rule 2 (&3)
f₃ is set up
Solution: **CAching rules in Buckets (CAB)**

Now no more 100s dependent rules

**Only rules within requested bucket**
Bucket size affects memory efficiency
Bucket size affects memory efficiency

Larger buckets (2x2)
Bucket size affects memory efficiency

Larger buckets (2x2)
Bucket size affects memory efficiency

Larger buckets (2x2)
- More rules cached each time
- Unmatched rules cached
Bucket size affects memory efficiency

Larger buckets (2x2)
- More rules cached each time
- Unmatched rules cached

Smaller buckets (4x4)
Bucket size affects memory efficiency

Larger buckets (2x2)
- More rules cached each time
- Unmatched rules cached

Smaller buckets (4x4)
Bucket size affects memory efficiency

Larger buckets (2x2)
• More rules cached each time
• Unmatched rules cached

Smaller buckets (4x4)
• More buckets cached
Bucket size affects memory efficiency

Larger buckets (2x2)
- More rules cached each time
- Unmatched rules cached

Smaller buckets (4x4)
- More buckets cached

Choosing bucket size affects switch memory efficiency
Decision tree based generation algorithm [HyperCut]

**No. of associate rules** in each bucket is bounded

Technical problems:
How to select the fields to partition?
[see paper]
Performance Evaluation
- Cache miss rate
- Bandwidth Consumption
- Flow setup latency (see paper)

Parameter Setting
- Effects of Tuning bucket size

Comparison
- Caching rules in Buckets (CAB)
- Caching exact match rules (CEM)
- Caching micro rules (CMR)
- Caching dependent rules (CDR)
Cache Miss and Control Bandwidth Performance

- CDR: Dependent Rules
- CMR: Micro Rules
- CAB: Bucket + Rules
- CEM: Exact Match

> 10x less cache miss
Cache Miss and Control Bandwidth Performance

> half less control bandwidth use
Effect of tuning bucket size

Memory overflow

Cache miss / sec

Rules installed / sec

Small bucket

Large bucket

Bucket Size (N)
CAB is a novel wildcard rule caching system which

Resolves **rule dependency** in wildcard rule caching

Achieves **efficient switch memory use** helps reducing control network bandwidth
flow setup latency
controller load

Is **fully compatible** with the latest OpenFlow standards
Thank you!
Q&A