# A Survey Of Data Center Network Virtualization

*สันติ มะลิซ้อน, สาธิต สีถาผล, วีรัท เกตุแก้ว, นิตญา โพธิ์ดา, ปาจรีย์ วงษ์กันยา, อัจฉรา ธานีกุล*

***บทคัดย่อ*** *- ด้วยข้อมูลที่มีการขยายตัวและเพิ่มความหลากหลายของแอพพลิเคชันบนอินเทอร์เน็ต ศูนย์ข้อมูล (Data centers :DCs) เป็นสิ่งที่มีประสิทธิภาพและสนับสนุนแนวโน้มการจัดเก็บข้อมูล และจัดให้มีแพล็ตฟอร์มเพื่อพัฒนาการใช้งานที่หลากหลายในการให้บริการเครือข่ายและแอพพลิเคชัน เช่น วีดีโอสตรีมมิ่ง, cloud computing บ่อยครั้งที่แอพพลิเคชันและบริการกำหนดความต้องการทรัพยากร multi-farious (หน่วยจัดเก็บข้อมูล, คำนวณการทำงาน, แบนด์วิดธ์, เวลาแฝง) บนโครงสร้างที่จำเป็นสถาปัตยกรรมศูนย์ข้อมูลที่มีอยู่ยังขาดความยืดหยุ่นที่มีประสิทธิภาพที่จะสนันสนุนแอพพลิเคชันเหล่านี้ กับผลลัพธ์ในการสนับสนุนระดับต่ำของ Qos, การพัฒนา, การจัดการที่ดีและการรักษาความปลอดภัยป้องกันการโจมตี ศูนย์ข้อมูลเครือข่ายเสมือนจริงเป็นแนวโน้มการแก้ปัญหาที่เป็นอยู่ ศูนย์ข้อมูลเสมือนคือสิ่งที่แสดงให้เห็นถึง การจัดการที่ดีกว่า มีความยืดหยุ่น ต้นทุนต่ำ ปรับขนาดได้ ใช้ทรัพยากรมีประโยชน์มากขึ้นและใช้พลังงานมีประสิทธิภาพ ในบทความนี้เรานำเสนอเครือข่ายศูนย์ข้อมูลเสมือนที่เป็นอยู่จริงในขณะนี้ ให้รายละเอียดเปรียบเทียบงานเซอร์เวย์ เราอภิปรายถึงหัวข้ออุปสรรค์งานวิจัยเพื่องานวิจัยในอนาคตและชี้ให้เห็นถึงศักยภาพโดยตรงเพื่อแก้ปัญหาเกี่ยวกับการออกแบบศูนย์ข้อมูล*

**คำนิยาม** : สถาปัตยกรรมศูนย์ข้อมูล, ระบบปฏิบัติการเสมือน, ศูนย์ข้อมูลเสมือน

## 1. บทนำ

เมื่อไม่นานมานี้ศูนย์ข้อมูลได้รับความสำคัญในเรื่องโครงสร้างพื้นฐานของการวัดประสิทธิภาพว่ามีความคุ้มค่ากับค่าใช้จ่ายหรือไม่ เพื่อการจัดเก็บข้อมูลขนาดใหญ่และโฮสติ้งสำหรับเซอร์วิสแอพพลิเคชันขนาดใหญ่ ในปัจจุบันบริษัทขนาดใหญ่เช่น Amazon, Google, Facebook and Yahoo! มักใช้ศูนย์ข้อมูลเพื่อจัดเก็บ, ค้นหาข้อมูล, การประมวลผลขนาดใหญ่ พร้อมกับการเกิดของ Cloud Computing การให้บริการของผู้ให้บริการในศูนย์ข้อมูลกลายเป็นธุรกิจที่มีมูลค่ามหาศาล ซึ่งเป็นบทบาทที่สำคัญมากในเทคโนโลยีสารสนเทศของอุตสาหกรรมในอนาคต

ถึงแม้ว่าในปัจจุบันศูนย์ข้อมูลมีความสำคัญ แต่สถาปัตยกรรมของศูนย์ข้อมูลยังไม่ได้รับการพัฒนาเท่าที่ควร โดยปกติแล้วศูนย์ข้อมูลใช้ Dedicated server ในการรันแอพพลิเคชัน มีผลทำให้การทำงานของเซิฟเวอร์มีประสิทธิภาพและต้นทุนในการดำเนินงานสูง ภายหลังจากมีเทคโนโลยีเซิฟเวอร์เสมือนช่วยแก้ปัญหาดังกล่าวได้ เช่น VMware[4], Xen[5] ซึ่งเครื่องคอมพิวเตอร์หนึ่งเครื่องสามารถมีเครื่องคอมพิวเตอร์เสมือนได้หลายเครื่อง เทคโนโลยีนี้สามารถแยกประสิทธิภาพของเครื่องที่ทำงานร่วมกันได้ เพื่อปรับปรุงประสิทธิภาพของแอพพลิเคชันและป้องกันการโจมตี อย่างไรก็ตามเซิฟเวอร์เสมือนเครื่องเดียวไม่เพียงพอต่อการจำกัดที่อยู่ทั้งหมดจากสถาปัตยกรรมศูนย์ข้อมูลในปัจจุบัน โดยเฉพาะอย่างยิ่งเครือข่ายศูนย์ข้อมูลยังคงอาศัย TCP/PC แบบเดิม ผลลัพธ์ที่เกิดขึ้นที่มีข้อจำกัดดังนี้

*1. ไม่มีการแยกประสิทธิภาพการทำงาน* : ทุกวันนี้ Cloud Application เช่น Search Engines และ Web services มีข้อจำกัดเรื่องการทำงานบนเครือข่าย ในรูปแบบ latency และ throughput อย่างไรก็ตามเทคโนโลยีเครือข่ายทุกวันนี้ สามารถให้บริการที่ดีที่สุดเท่าที่จะทำได้โดยไม่สามารถแยกประสิทธิภาพการทำงานได้ จึงเป็นเรื่องยากที่จะคาดการณ์ถึงคุณภาพของบริการสำหรับแอพพลิเคชันเหล่านี้ได้

*2. มีความเสี่ยงเพิ่มขึ้น* : ศูนย์ข้อมูลเครือข่ายแบบเดิมไม่สามารถจำกัดรูปแบบความต้องการและแบนด์วิดธ์ในแต่ละแอพพลิเคชันได้ ส่งผลให้เครือข่ายถูกโจมตีจากภายในได้ง่ายขึ้น เช่น ถูกรบกวนการทำงาน และ ถูกโจมตี Denial of Service (DoS)[6]

*3. การปรับปรุงแอพพลิเคชันไม่ค่อยดี* : ทุกวันนี้แอพพลิเคชันสำหรับองค์กรขนาใหญ่ใช้โปรโตคอลเฉพาะของแต่ละแอพพลิเคชันและ address spaces[7] การย้ายแอพพลิเคชันเหล่านี้ไปยังสภาพแวดล้อมแบบศูนย์ข้อมูล เป็นอุปสรรค์สำคัญเนื่องจากมักจะมีความยุ่งยากในการปรับเปลี่ยนโปรโตคอลและชุดคำสั่งของแอพพลิเคชัน

*4. จำกัดความยืดหยุ่นการจัดการ* : ในสภาพแวดล้อมของศูนย์ข้อมูลที่ทั้งเซิฟเวอร์และเครือข่ายมีการใช้งานหลายโปรแกรมร่วม ตัวแอพพลิเคชันเองจัดการองค์ประกอบเครือขายเพื่อความหลากหลาย

ของจุดมุ่งหมาย เช่น การทำงาน load balancing, การวิเคราะห์ข้อผิดพลาดและการป้องกันความปลอดภัย อย่างไรก็ตาม สถาปัตยกรรมเครือข่ายศูนย์ข้อมูลแบบเดิมไม่มีความยืดหยุ่นสำหรับผู้เช่าเพื่อจัดการองค์ประกอบเหล่านั้นในศูนย์ข้อมูล

5. *ไม่สนับสนุนนวัตกรรมเครือข่ายใหม่* : เนื่องจากศูนย์ข้อมูลแบบเดิมไม่มีความยืดหยุ่นทำให้ไม่สามารถเกิดนวัตกรรมใหม่ขึ้นมาได้ เป็นผลให้ยากที่จะเปลี่ยนแปลงศูนย์ข้อมูลแบบเดิมเช่น การอัพเกรด network protocol หรือไม่มีบริการเครือข่ายแบบใหม่ ในระยะยาวทำให้ประสิทธิภาพในการลงทุนลดลง

จากข้อจำกัดเรานี้ เราจะเสนองานวิจัยที่เกี่ยวข้องกับเครือข่ายศูนย์ข้อมูลเสมือนดังนี้ ประเด็นแรกคือสรุปงานวิจัยที่ผ่านมา สองคือเปรียบเทียบสถาปัตยกรรมที่สำคัญ สุดท้ายชี้ให้เห็นถึงงานวิจัยในอนาคต

หลังจากบทนำจะมีหัวข้อดังต่อไปนี้ คำศัพท์และคำจำกัดความที่เกี่ยวข้อง (ส่วนที่ 2) ถัดไปเป็นการสรุปข้อเสนอ(ส่วนที่ 3) ที่เกี่ยวข้องและเปรียบเทียบจากมุงมองต่างๆ (ส่วนที่ 4) และ (ส่วนที่ 5) จะกล่าวถึงงานวิจัยที่จะทำในอนาคตและสุดท้ายเป็นการสรุปบทความนี้(ส่วนที่ 6)

## 2. ความเป็นมา

ในส่วนนี้เราจะนำเสนอคำศัพท์ที่เกี่ยวข้องกับเครือข่ายศูนย์ข้อมูลเสมือนที่เราจะใช้ในบทความนี้ ตารางที่ I ใช้แสดงรายการอักษรย่อที่ใช้ตลอดบทความ

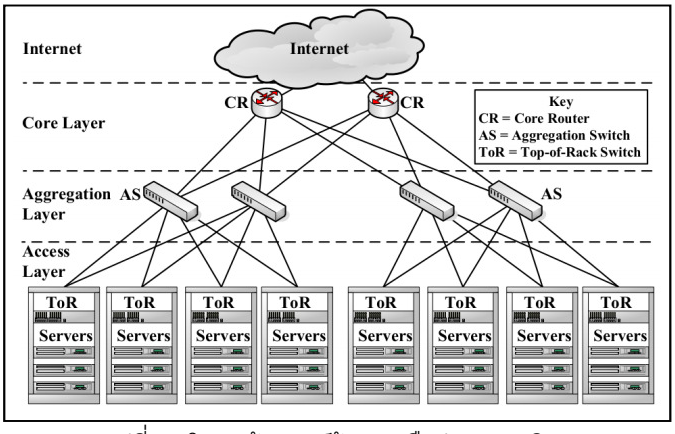
| Acronym | Description |
|---|---|
| DC | Data Center |
| VM | Virtual Machine |
| VN | Virtual Network |
| VDC | Virtual Data Center |
| VLAN | Virtual Local Area Network |
| ToR | Top-of-Rack |
| AS | Aggregation Switch |
| CR | Core Router |
| InP | Infrastructure Provider |
| SP | Service Provider |
| IaaS | Infrastructure-as-a-Service |

ตารางที่ 1 แสดงอักษรย่อที่ใช้ในบทความ

**ก. ศูนย์ข้อมูล**

ศูนย์ข้อมูล(data center: DC) ประกอบด้วย เครื่องเซิฟเวอร์ ตัวจัดเก็บและอุปกรณ์เครือข่าย เช่น สวิตท์, เร้าท์เตอร์ และสายเคเบิล ระบบจ่ายไฟฟ้า ระบบระบายความร้อน

ศูนย์ข้อมูลเครือข่าย คือ โครงสร้างพื้นฐานการสื่อสารที่ใช้ในศูนย์ข้อมูล และเป็นตัวกระจายโทโพโลยีเครือข่าย อุปกรณ์เราท์ติ้ง/สวิตท์ และใช้โปรโตคอล เช่น อีเทอร์เน็ตและไอพี ต่อไปนี้เราจะนำเสนโครงสร้างทั่วไปที่ใช้ในศูนย์ข้อมูลและโครงสร้างอื่นที่ผู้วิจัยอื่นเสนอไว้




รูปที่ 1 โครงสร้างศูนย์ข้อมูลเครือข่ายแบบเดิม

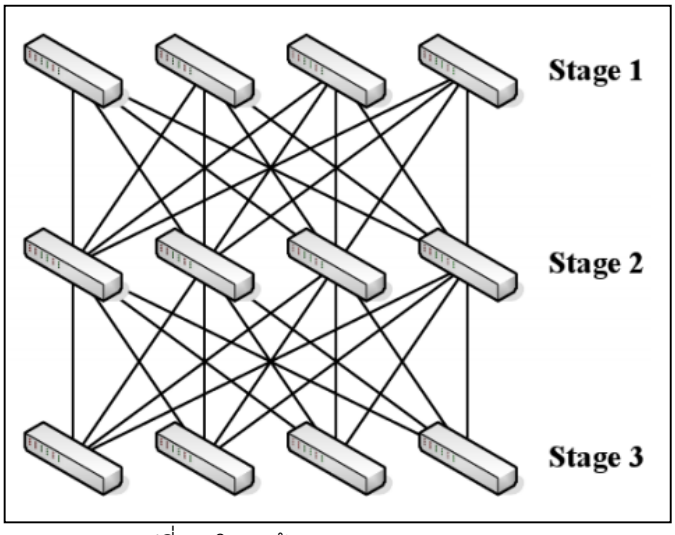
รูปที่ 1 แสดงโครงสร้างทั่วไปของเครือข่ายศูนย์ข้อมูล [9] ในโครงสร้างนี้ Top-of-Rack(ToR) จะสลับในการเข้าถึงแต่ละเลเยอร์ให้เชื่อมต่อไปยังเซิฟเวอร์ที่ติดอยู่กับแร็คทุกตัว แต่ละ aggregation switch(AS) ในชั้นของ aggregation (บางครั้งอ้างถึงการกระจายเลเยอร์) ส่งแต่จากการเข้าถึงหลายเลเยอร์(Tor) สลับไปถึง core layer สวิทต์ ToR ทุกตัวจะติดต่อไปยังสวิทต์ชั้น aggregation เพื่อให้ไม่เกิดความซ้ำซ้อน ส่วน core layer ให้ความปลอดภัยในการติดต่อระหว่าง aggregation switch และ core routers(CR) ติดต่อไปยังอินเทอร์เน็ต โดยเฉพาะกรณีของโครงสร้างแบบเดิมที่เป็นโครงสร้างในเลเยอร์ 2 ซึ่งใช้ในสวิทตย์เลเยอร์ 2 เท่านั้น

โครงสร้างคลอส(Clos topology)เป็นโครงสร้างที่สร้างขึ้นจากหลายขั้นตอนของสวิทต์[10] สวิทต์แต่ละตัวในขั้นตอนติดต่อกับสวิทต์ทุกตัวในขั้นตอนถัดไป ที่ให้เส้นทางที่กว้างและหลากหลายขึ้น รูปที่ 2 แสดงตัวอย่างของสามขั้นตอนของโครงสร้างคลอส

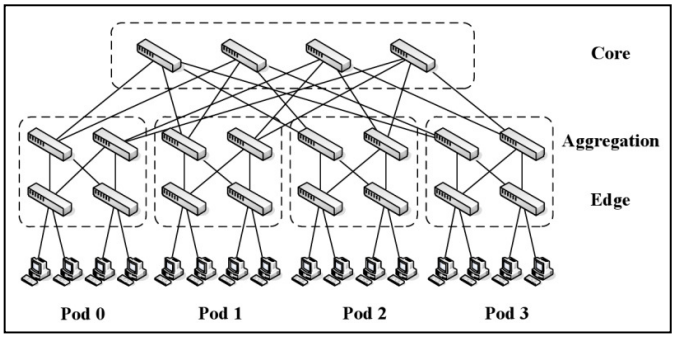
Fat-tree topology เป็น Clos topology แบบพิเศษที่จัดอยู่ในโครงสร้าง tree-like ซึ่งแสดงในรูปที่ 3 โดยโครงสร้างสร้างขึ้นจากเคพอร์ต(k-port) ซึ่งแต่ละตัวจะทำงานในเลเยอร์สอง(aggregation และ edge) ของสวิทต์ k/2

แต่ละสวิทต์ K/2 มีหนึ่งพอร์ตติดต่อกับเคพอด(k pods) โดยพอร์ตที่ i ของ core สวิทต์ใดๆ ที่เชื่อมต่อกับ pod I ดังนั้นพอร์ตถัดไปในชั้น aggregation ของแต่ละสวิทต์ pod ที่ติดต่ออยู่กับ k/2 แต่ละสวิทต์เอจ(edge switch) จะติดต่อโดยตรงกับ k/2 แต่ละพอร์ตที่ลดลงของ K/2 ของสวิทต์เอจเป็นการติดต่อกับ k/2 พอร์ตบนสวิทต์รวม[2]

โทโพโลยีด้านบนมีคุณลักษณะที่สร้างขึ้นอย่างเหมาะสมกับเครือข่ายศูนย์ข้อมูล อย่างไรก็ตามโทโพโลยีศูนย์ข้อมูลไม่จำกัดโครงสร้างในส่วนนี้ ตัวอย่างเช่น BCube[13] เป็นสถาปัตยกรรมเครือข่ายศูนย์ข้อมูลบนโครงสร้างไฮเปอร์คูป(hyper-cube) สิ่งที่น่าสนใจมากคือสามารถหาวิธีการเปรียบเทียบโครงสร้างเครือข่ายศูนย์ข้อมูลเครือข่ายที่เกี่ยวข้องใน [14]




รูปที่ 2 โครงสร้างคลอส(Clos topology)

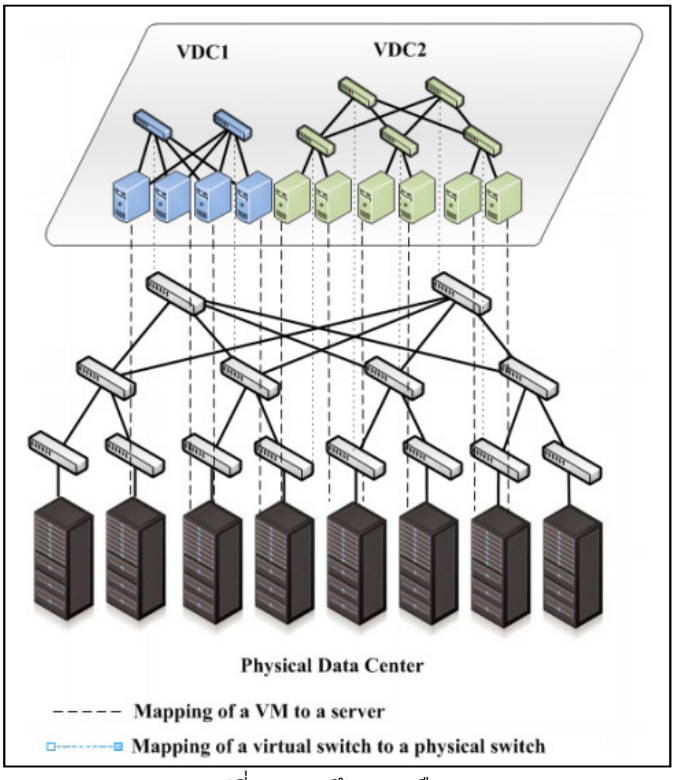



รูปที่ 3 แสดง Fat-tree topology

**ข. ศูนย์ข้อมูลเสมือน**

ศูนย์ข้อมูลเสมือนเป็นศูนย์ข้อมูลที่เป็นบางส่วนหรือทั้งหมดของฮาร์ดแวร์ เช่น เซิฟเวอร์ เราท์เตอร์ สวิตท์ และ สายเชื่อมต่อ อยู่ในรูปแบบเสมือนจริง โดยปกติฮาร์ดแวร์ทางกายภาพเป็นการจำลองการใช้ซอฟต์แวร์หรือเฟิร์มแวร์เรียกไฮเปอร์ไวเซอร์ที่แบ่งแยกอุปกรณ์ได้หลากหลายและเป็นอิสระต่อกัน เช่น ตัวอย่างเครื่องทางกายภาพ (เซิฟเวอร์) ผ่านแบบจำลองโดยใช้ไฮเปอร์ไวเซอร์เพื่อสร้างเครื่องเสมือน(VMs) ที่มีการสร้างตัวเก็บความจุต่างๆ เช่น หน่วยประมวลผลกลาง หน่วยความจำ พื้นที่จัดเก็บ และทำงานในระบบปฏิบัติการและแอพพลิเคชันที่แตกต่างกันได้

ศูนย์ข้อมูลเสมือน เป็นการรวบรวมทรัพยากรเสมือนต่างๆ เช่น VMs, virtual switch, และ virtual router ซึ่งถูกเชื่อมต่อผ่านทาง virtual link ในขณะที่ VDC คือ ศูนย์ข้อมูลทางกายภาพซึ่งเป็นเทคนิคการปรับปรุงทรัพยากรเสมือน, VDC เป็นค่าคงที่ของศูนย์ข้อมูลเสมือนซึ่งประกอบจากทรัพยากรศูนย์ข้อมูลทางกายภาพ เครือข่ายเสมือน(VN) เป็นการรวมทรัพยากรเครือข่ายเสมือนได้แก่ Virtual nodes (end-hosts, switches, routers) และ virtual links ดังนั้น VN เป็นส่วนหนึ่งของ VDC และระดับของเครือข่ายเสมือนเป็นหนึ่งในระดับชั้นของเครือข่าย (ชั้นแอพพลิเคชันถึงชั้น physical) จากรูปที่ 4 แสดงถึง VDCs จำนวนมากสามารถใช้งานผ่านศูนย์ข้อมูลเสมือน




รูปที่ 4 ศูนย์ข้อมูลเสมือน

**ค**. Business Model

ในส่วนนี้ เราจะกล่าวถึงส่วนประกอบหลักของสภาพแวดล้อมของศูนย์ข้อมูลเสมือน

โดยเฉพาะอย่างแรกของความแตกต่างระหว่างรูปแบบเครือข่ายแบบเก่ากับรูปแบบเครือข่ายเสมือนเป็นการมีส่วนร่วมหลายคน โดยเฉพาะอย่างยิ่งในอดีตถือว่ามีแค่สองคน คือ ไอเอสพีเอส (ISPs) และผู้ใช้(end-users) ซึ่งภายหลังได้เสนอให้แยกกฎไอเอสพีแบบเดิมออกเป็นสองอย่าง คือ ผู้ให้บริการโครงสร้างพื้นฐาน(InP) และ ผู้ให้บริการ(SP) SPsจาก InPs เพิ่มโอกาสสำหรับนวัตกรรมเครือข่ายซึ่งแบ่งกฎในการปรับปรุงวิธีการเครือข่าย เช่น โพโตคอลบริการ(SP) จากกฎของเจ้าของและยังเก็บโครงสร้างทางกายภาพของ InP

ในบริบทของศูนย์ข้อมูลเสมือนนั้น InP เป็นบริษัทเจ้าของและจัดการโครงสร้างทางกายภาพของศูนย์ข้อมูล InP เช่าทรัพยากรเสมือนเพื่อให้บริการแบบมัลติเพล ผู้เช่าแต่ละรายจะสร้างศูนย์ข้อมูลเสมือนเกินโครงสร้างทางกายภาพของเจ้าของโดยใช้ InP เพื่อการปรับปรุงของบริการและแอพพลิเคชันให้ผู้ใช้ในอนาคต ดังนั้นหลาย SPs สามารถปรับปรุงแอพพลิเคชันได้มากกว่าโครงสร้างทางกายภาพของศูนย์ข้อมูล

## 3. งานวิจัยที่เกี่ยวข้อง

เครือข่ายศูนย์ข้อมูลเสมือนเป็นสิ่งที่อยู่ในช่วงเริ่มต้น และงานวิจัยเมื่อเร็วๆ นี้ให้ความสำคัญว่าจะใช้งานฟังก์ชันพื้นฐานได้อย่างไรและสามารถเพิ่มเติมส่วนไหนได้ในอนาคตของทรัพยากรเครือข่ายศูนย์ข้อมูลแต่ละส่วน รูปแบบการส่งต่อข้อมูลและการแยกประสิทธิภาพเครือข่าย ดังนั้นเราจึงมุ่งประเด็นที่สนใจในบทความนี้ดังนี้

รูปแบบการส่งต่อข้อมูลกับการกำหนดกฎที่ใช้ในการส่งต่อข้อมูลระหว่างโหนดเสมือน การรับประกันแบนด์วิดธ์และวิธีการใช้แบนด์วิดธ์ร่วมกันโดยแยกประสิทธิภาพเครือข่ายและการใช้ทรัพยากร์ร่วมกันในเครือข่ายมากขึ้น ตามลำดับ

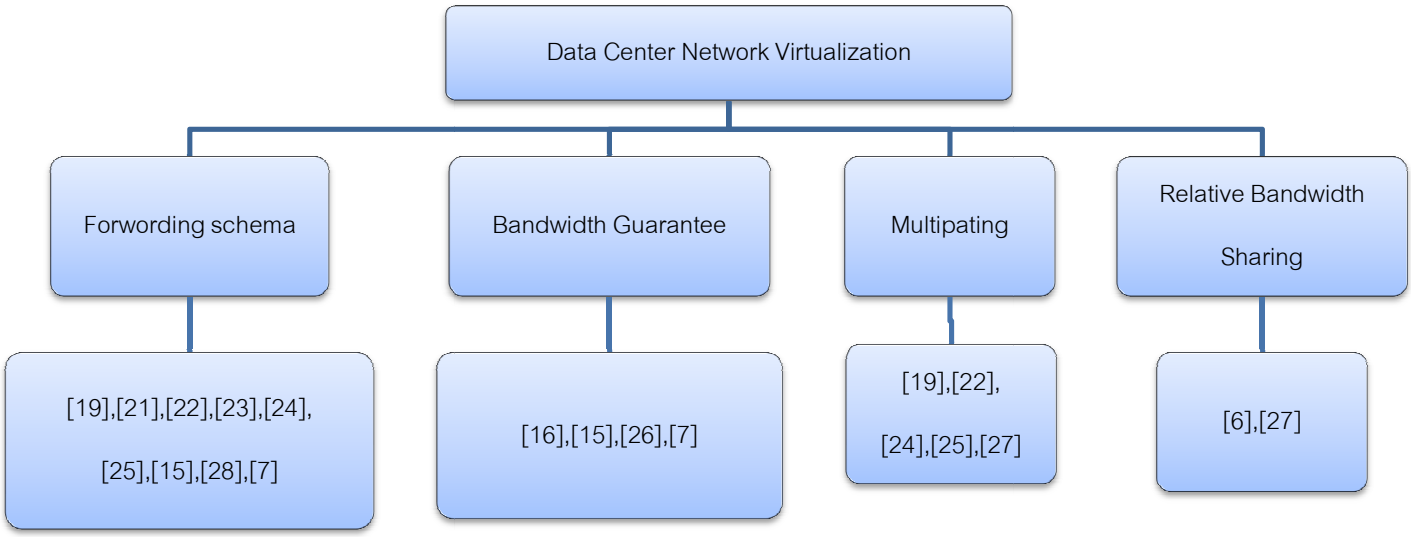
เทคนิค Multipath ใช้สำหรับกระจายการจราจรระหว่างเส้นทางที่ต่างกันในลำดับการทำงาน load-balancing และ fault-tolerance

ถึงอย่างนั้นมีคุณลักษณะที่ให้ผลตอบแทนที่คุ้มค่าเมื่อมีการสร้างศูนย์ข้อมูลเสมือน เช่น ความปลอดภัย โปรแกรมใช้งานง่าย จัดการง่าย อนุรักษ์พลังงาน และ fault-tolerance ดังนั้นเรากล่าวรายละเอียดเพิ่มเติมเกี่ยวกับคุณลักษณะที่ท้าทายเกี่ยวกับงานวิจัยในอนาคตด้านนี้ ต่อไปเราจะอธิบายโดยสังเขปถึงลักษณะที่เราให้ความสนใจในบทความนี้

กฎการกำหนดรูปแบบการส่งต่อเพื่อส่งต่อแพ็คเก็ตโดยใช้การสลับองค์ประกอบ จากพอร์ตขาเข้าและพอร์ตขาออก FIB รองรับการจับคู่ MAC Address เพื่อสลับพอร์ต เมื่อเกิดการตัดสินใจเกี่ยวกับการส่งต่อแพ็คเก็ต สนับสนุนการใช้แบนด์วิดธ์ร่วมกัน ซึ่งอาจจะใช้อุโมงค์ข้อมูลควบคุมความแออัด[6] มีการใช้ใน shim layer สกัดแพ็คเก็ตทั้งหมดที่เข้ามาและออกจากเซิฟเวอร์ แต่ละอุโมงค์ข้อมูลที่รองรับอัตราการส่งและมีการใช้ rate-limiter ทางเลือกอื่น [27] คือกลุ่มการแบ่งสำหรับดูแลการจราจร TCP, จำกัดอัตราเพื่อควบคุม UDP , และแบ่งศูนย์กลางสำหรับสนับสนุนนโยบายเพิ่มเติม เช่น การจัดการการไหลแบบเจาะจง, ทางเลือกที่ใช้คือการจัดคิว อีกทางคืออาศัย shim layer ด้านล่าง UDP

เทคนิคหนึ่งที่จะรับประกันผลแบนด์วิดธ์คือใช้ rate-limiters [7],[15],[16],[26] โดยเฉพาะอย่างยิ่ง rate-limiters รวมอยู่ในไฮเปอร์ไวเซอร์ของแต่ละเครื่อง มีบทบาทในการรับรองว่าทุกเครื่องเสมือนจะไม่มีการแบ่งแบนด์วิดธ์ที่มากเกินไป การทำงานของกระบวนการระดับผู้ใช้ในลีนุกซ์ไฮเปอร์ไวเซอร์(Linux hypervisor: dom 0) ซึ่งจะอาศัย Open vSwitch (ทำงานบนไฮเปอร์ไวเซอร์) เพื่อติดตามอัตราแต่ละลำดับ การรับประกันแบนด์วิดธ์ใน GateKeeper เป็นการจำกัดอัตราการใช้ในลีนุกซ์ hierchical token bucket(HTB) ตารางการทำงานในเซนไฮเปอร์ไวเซอร์(โดเมน 0) ใน end-host CloudNaaS อาศัย Open vSwitch ซึ่งแม้ว่าไม่มีสถานะที่ชัดเจน สามารถใช้จำกัดอัตราได้ การปรับปรุงการจำกัดอัตราโดยแบ่งให้โฮสต์สุดท้ายสร้างความเป็นไปได้ โดยหลีกเลี่ยงการสำรองแบนด์วิดธ์ในสวิทต์เท่าที่ศูนย์ข้อมูลเสมือนจัดการเฟรมเวิร์คเพื่อความแน่ใจในการข้ามแต่ละสวิทต์ไม่มากไปกว่าความสามารถในการเชื่อมโยง

ในตารางที่ 2 เราจะจำแนก surveyed projects จะกล่าวถึงคุณลักษณะเหล่านั้นและเน้นถึงคุณลักษณะที่อาจมีมากกว่าหนึ่งคุณลักษณะ ที่ทำเครื่องหมายแสดงถึงคุณลักษณะที่มีในแต่ Proposal




รูปที่ 10 การแบ่งกลุ่มศูนย์ข้อมูลตามคุณลักษณะ

| Proposal | References | Feature | | | |
|---|---|---|---|---|---|
| | | Forwarding Scheme | Bandwidth Guarantee | Multipathing | Relative bandwidth sharing |
| Traditional DC | [4], [5], [19] | ✓ | | ✓ | |
| SPAIN | [20] | | | ✓ | |
| Diverter | [21] | ✓ | | | |
| NetLord | [22] | ✓ | | ✓ | |
| VICTOR | [23] | ✓ | | | |
| VL2 | [24] | ✓ | | ✓ | |
| PortLand | [25] | ✓ | | ✓ | |
| Oktopus | [16] | | ✓ | | |
| SecondNet | [15] | ✓ | ✓ | | |
| Seawall | [6] | | | | ✓ |
| Gatekeeper | [26] | | ✓ | | |
| NetShare | [27] | | | ✓ | ✓ |
| SEC2 | [28] | ✓ | | | |
| CloundNaaS | [7] | ✓ | ✓ | | |

ตารางที่ 2 แสดงการจำแนก surveyed projects

*1. Traditional data center (DC)*

โดยทั่วไปการจำลองของสถาปัตยกรรมศูนย์ข้อมูลในปัจจุบันมักประสบผลสำเร็จบนการจำลอง server โดยแต่ละผู้เช่าจะครอบครองกลุ่มของ virtual server และการจำแนกระหว่างผู้เช่าผ่าน VLANs ซึ่งศูนย์ข้อมูลได้อาศัยการออกแบบที่เรียบง่ายที่สามารถดำเนินการโดยใช้ commodity switch และเทคโนโลยี hypervisor ที่มีความนิยม เช่น VMware [4], Xen [5] นอกจากนี้ผู้เช่ายังสามารถกำหนด address space ของ layer 2 และ layer 3 ได้ด้วยตนเอง

ข้อจำกัดหลักที่สำคัญของสถาปัตยกรรมศูนย์ข้อมูลในทั่วไปคือความสามารถในการปรับขยายตั้งแต่การที่ commodity switch ไม่สามารถออกแบบเพื่อการจัดการ VMs จำนวนมาก และจำนวนที่เกิดจาก traffic โดยเฉพาะอย่างยิ่ง switch ต้องมีการดูแลในด้านของ FIBs (Forwarding Information Base)สำหรับทุกๆ VM ที่สามารถเพิ่มขนาดของตารางการส่งต่ออย่างมาก นอกจากนี้ ตั้งแต่ VLANs ถูกใช้เพื่อการแยกแยะระหว่างผู้เช่า จำนวนของผู้เช่าได้ถูกจำกัดไว้ที่ 4096 ซึ่งเป็นจำนวนของ VLANs ที่อนุญาตโดยมาตรฐาน 802.1q [19]

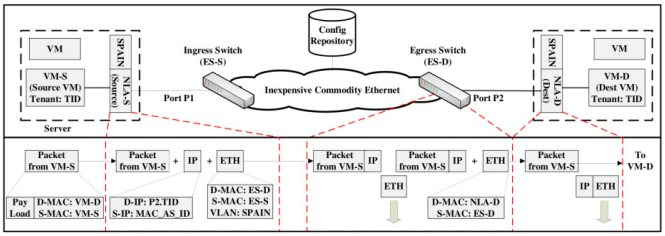
*2. Diverter*

สนับสนุนการแบ่งเชิงตรรกะของเครือข่าย IP เป็นสิ่งที่สำคัญสำหรับการอำนวยความสะดวกที่ดีกว่าของแอพพลิเคชันและความต้องการของบริการในสิ่งแวดล้อม multi-tenant ขนาดใหญ่ เช่น ศูนย์ข้อมูล Diverter[21] เป็นวิธีการทางซอฟต์แวร์ที่เข้าถึงเครือข่ายเสมือนเพื่อเครือข่ายข้อความที่สมมติว่าไม่ต้องกำหนดค่า switches

หรือ routers Diverter ถูกใช้ในซอฟต์แวร์โมดูล (เรียกว่า VNET) ที่ถูกติดตั้งทุกเครื่อง เมื่อ VM ส่ง Ethernet frame VNET จะแทนที่ MAC Address ต้นทางและปลายทาง โดย Address ที่เครื่องที่โฮสต์ต้นทางและปลาย VMs ตามลำดับ หลังจากนั้น Switches จะทำการส่งต่อแพ็คเก็ตที่ใช้ MAC Address ของเครื่อง VNET ใช้ปรับปรุงรุ่นของ ARP protocol เพื่อค้นหาเครื่องโฮสต์โดยเฉพาะ VM Diverter ต้องการให้ทุก VM มีรูปแบบการเข้ารหัสที่ระบุ IP address ของผู้เช่ารวมทั้ง subnet และที่อยู่เครื่องเสมือน(ปัจจุบันใช้ *10 .tenant.su bnet.v m*) ดังนั้นจึงไม่มีการชนกันของที่อยู่ของผู้เช่า VNET ทำการหาเส้นทางระหว่าง subnets โดยเขียน MAC Address ใหม่อีกครั้ง ซึ่งให้ภาพเสมือนเพื่อข้ามเกตเวย์ โดยสรุปแล้ว Diverter เครือข่ายเสมือนเลเยอร์ 3 ที่อนุญาตให้ผู้เช่าทุกรายสามารถควบคุม IP subnet และ VMs Address ของตนเองได้ ข้อจำกัดหลักใน proposal นี้คือไม่สามารถรับประกัน Qos ได้

*3. NetLord*

เพื่อเป็นการเพิ่มรายได้ ผู้ให้บริการของการให้บริการโครงสร้างพื้นฐาน (IaaS) มีความสนใจในการใช้ประโยชน์ในทรัพยากรอย่างเต็มรูปแบบ หนึ่งในทางเลือกที่มีประสิทธิภาพมากที่สุดที่จะบรรลุเป้าหมายคือการเพิ่มจำนวนของผู้เช่าในการแชร์โครงสร้าง NetLord เป็นสถาปัตยกรรมที่มุ่งมั่นที่จะพยายามปรับการขยายระบบของประชากรผู้เช่าในศูนย์ข้อมูล ซึ่งสถาปัตยกรรมเสมือน ของที่อยู่พื้นที่ของผู้เช่า L2 และ L3 ที่ซึ่งอนุญาตให้ผู้เช่าออกแบบและปรับปรุงพื้นที่ผู้ใช้และแอพพลิเคชันได้ตามความต้องการ




รูปที่ 5 แสดงสถาปัตยกรรม NetLord

แนวความคิดหลักๆของ NetLord แสดงในรูปภาพที่ 5 คือการห่อหุ้มแพ็คเก็ต ผู้เช่า L2 และส่งไปยัง fabric employing L2+L3 encapsulation โดยheader packet ของ L2 จะเจาะจง MAC address ของต้นทางและปลายทางของ VM ต้นทาง NetLord Agent ให้นำไปใช้กับแต่ละserverจริงที่ควบคุมVMทั้งหมดบนserver โดยเฉพาะส่วนที่ทำการ encapsulate packet L2 โดยการเพิ่ม header L2 และ L3 ดังรูป ตัวต้นทางและปลายทางที่อยู่ของ L2 ทำการกำหนด MAC address ของการสลับเข้าออกของ server hosting และต้นทาง VM ตามลำดับ ต้นทางพิเศษ IP address แสดง ID MAC address ของพื้นที่ผู้เช่า ซึ่งช่วยให้ผู้เช่าใช้พื้นที่ที่หลากหลายของ L2 ด้านปลายทาง IP address จะเจาะจง port ของการสลับออกไปยังภายนอกเพื่อส่งต่อpacketไปยัง server ปลายทางและ ID ของ hosting ผู้เช่า และ VM ปลายทาง packet จะส่งต่อผ่านเครือข่ายศูนย์กลางข้อมูลเพื่อการswitch ออกถึงพื้นฐานของ L2 fabric ผ่านเส้นทางที่เลือกโดย VLAN algorithmของ SPAIN [20] (โครงสร้างที่อาศัยการสนับสนุน VLAN ในสินค้าEthernetที่มีการ switch ไปยัง multipathing ที่เผยแพร่)

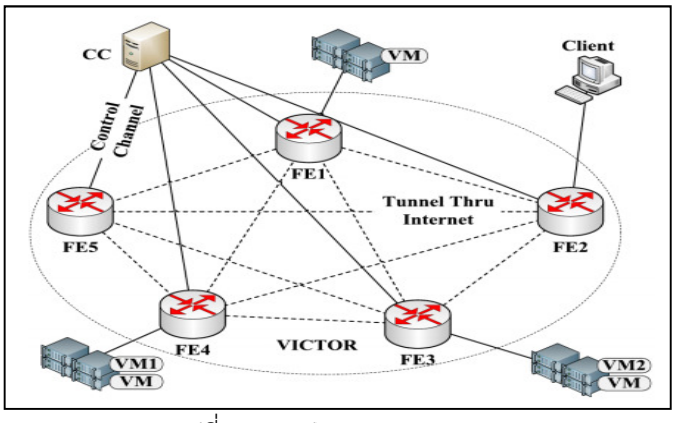
การส่งต่อ packet ออกจาก switch ไปยังserver ปลายทางจะขึ้นกับการค้นหา L3 ของportที่ออกไปข้างนอก และ NLA ทำการส่งต่อ packet บน server ปลายทางไปยัง VM ปลายทางโดยใช้ ID ของผู้เช่า พื้นที่ MAC address และที่อยู่ปลายทางของ L2 ของVMใน packet ที่ถูก encapsulate เพื่อสนับสนุน virtual routing NetLord ใช้เครื่องมือในการ routing เดิมที่ชื่อ Diverter [21] ในการสนับสนุน SPAIN multipathing และเก็บข้อมูลการตั้งค่าของผู้เช่า โดย NetLord ใช้หลายๆฐานข้อมูล

NetLord สันนิษฐานว่า ขอบสวิตช์สนับสนุนการส่งต่อIP พื้นฐาน แต่ไม่ใช่กับทุกๆ Commercial Off-the-Shelf (COTS) switch [33] การนำเสนอเกี่ยวกับ encapsulation หมายถึง packet size ที่มากขึ้น ซึ่งเป็นการเพิ่มการดรอปและการกระจายตัว นอกจากนี้ NetLord ใช้ SPAIN ในการส่งต่อ multipath operating บนพื้นฐานที่ไม่ได้เป็น scalable ในที่สุดแม้ว่าสถาปัตยกรรมการเผยแพร่การแยกแยะในระหว่างผู้เช่าจะไม่สนับสนุนการการันตี bandwidth

*4. VICTOR*

ผู้เช่า Cloud มีความต้องการที่จะย้าย service ข้ามศูนย์ข้อมูลเพื่อรักษาสมดุลการโหลดระหว่างหรือข้ามศูนย์ข้อมูล หรือเพื่อประสิทธิภาพให้กับเซิฟเวอร์ อย่างไรก็ตามผู้ใช้ Cloud ต้องการความรวดเร็วและมีประสิทธิภาพของบริการและข้อมูล วิธีการหนึ่งที่ให้ผลแก่ผู้เช่าและผู้ใช้ Cloud คือการโยกย้ายของ VMs เพื่อหลีกเลี่ยงการหยุดชะงักของบริการ VMs ควรเก็บค่า IP address เดียวกันไว้ระหว่างการโยกย้าย แม้ว่าจะไม่ได้เป็นสิ่งที่ท้าทายสำหรับการโยกย้ายระหว่างเครือข่าย IP เดียวกัน ให้โยกย้ายเครือข่ายที่แตกต่างกันไม่ใช่ง่าย VICTOR (Virtually Clustered Open Router) [23] เป็นสถาปัตยกรรมที่สนับสนุนการโยกย้ายข้ามเครือข่ายของ VMs ช่วยให้การโยกย้ายสามารถเก็บ IP address เดิมได้

แนวคิดหลักสำคัญของ VICTOR แสดงในรูปที่ 6 คือการสร้าง Forwarding Elements (FE) (L3 devices) ที่ให้บริการเป็น virtual line card กับ Multiple virtual ports ของ single virtual router FEs จะกระจายผ่านเครือข่ายหลายๆเครือข่ายซึ่งจะช่วยสนับสนุนการโยกย้ายของ VMs แบบข้ามหลายๆเครือข่าย ส่วน control plane จะได้รับการสนับสนุน โดยหนึ่งหรือหลายๆตัวควบคุมส่วนกลาง (CC) และ VM จะเป็น ใช้งานบนเซิร์ฟเวอร์ที่เชื่อมต่อกับเพียงหนึ่งขอบของ FE โดย CC จะดูแลตารางโครงสร้างที่ระบุการเชื่อมต่อระหว่าง FEs และตารางที่อยู่การเชื่อมต่อที่กำหนดในระหว่าง VM และ FE ที่เป็น server hosting ที่ VM ใช้เชื่อมต่อ และ CC จะคำนวณเส้นทาง routing จากแต่ละ FE ไปยัง VMs และทำการกระจายข้อมูลระหว่าง FEs ซึ่งพึ่งพาตาราง routing เหล่านี้ในการใช้ส่งต่อ packet ข้อจำกัดหลักๆของ VICTOR คือจำเป็นต้องมีการสนับสนุน FIBs เป็นขนาดใหญ่ที่นำไปสู่ปัญหาในเรื่องของ scalability ที่เกี่ยวข้องกับ Fes




รูปที่ 6 สถาปัตยกรรม VIRTOR

*5. VL2*

VL2 [24] เป็นสถาปัตยกรรมเครือข่ายศูนย์ข้อมูลที่มีจุดมุ่งหมายที่จะมีความยืดหยุ่นในด้านการจัดสรรทรัพยากร server ทั้งหมดใน VL2 เป็นผู้เช่า(ซึ่งในเปเปอร์นี้เราจะเรียกว่าเป็น "service") ได้ทำการแชร์โดยไม่คำนึงถึง single addressing space ของตำแหน่งจริงซึ่งหมายถึงserverใดๆสามารถกำหนดแก่ผู้เช่าใดก็ได้

VL2 จะขึ้นกับการที่ไม่มีสมาชิกมากเกินไปใน Clos topology (ดูจากรูปที่ 2) ที่ให้ความสะดวกของการ routing และความยืดหยุ่น packet จะถูกส่งต่อโดยใช้ IP address สองประเภท คือ location-specific addresses (LAs) และ application-specificad-dresses (AAs) โดยใช้สวิทช์และเซิร์ฟเวอร์ตามลำดับ VL2 อาศัยระบบไดเรกทอรีสำหรับการทำ AA-to-LA mapping ก่อนส่ง packet server VL2 จะทำการ encapsulate packet ด้วย LA address ของ ToR switch ปลายทาง โดย switch จะไม่ได้รู้ถึง AA addressing ตั้งแต่ส่งต่อ packet โดยใช้ LAs เท่านั้น ที่ ToR switch ปลายทาง packet จะถูก decapsulated และส่งไปยัง AA server ปลายทาง การออกแบบ VL2 อาศัย VLB and ECMP ที่จะขยายtraffic เพื่อใช้ประโยชน์จากท่ามกลางเส้นทางที่หลากหลาย

การแยกระหว่าง addressing spaces ของ switch และ server ช่วยเพิ่มความยืดหยุ่นของ VL2 ตั้งแต่ ToR switch ไม่สามารถส่งต่อข้อมูลจำนวนมากของ server นอกจากนี้ ระบบไดเรคทอรีของ VL2 ยังจำกัดความต้องการสำหรับการร้องขอของ ARP และ DHCP ที่เป็น source ทั่วไปของ broadcast traffic ใน data centers และ VLB กับ ECMP ยังอนุญาตให้มีการสลายไปที่ลงตัวหลังจากที่เครือข่ายล้มเหลว

ข้อจำกัดหนึ่งของ VL2 คือขาดการการันตี bandwidth ที่แน่นอนระหว่างserver ซึ่งเป็นสิ่งจำเป็นสำหรับหลายๆ application การนำเสนอนี้ยังคู่อย่างที่โทโพโลยี (Clos) พื้นฐานและ ต้องการให้สวิทช์ใช้ OSPF, ECMP และ IP-in-IP encapsulation ซึ่งสามารถจำกัดการใช้งานได้

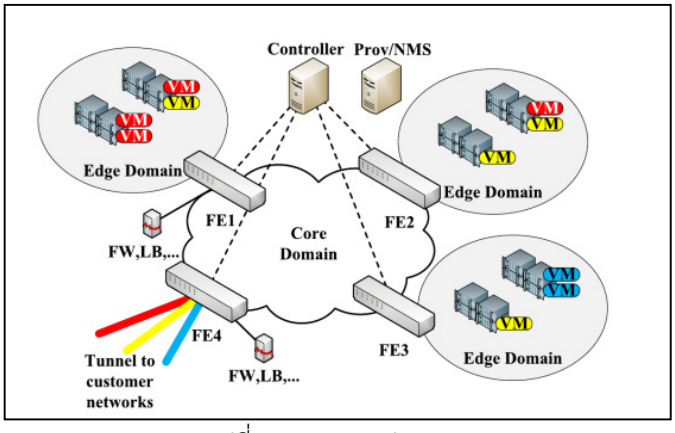
*6. PortLand*

การมีความสามารถในการขยายประชากร VM มีประสิทธิภาพในการโยกย้าย VM และการจัดการที่ง่าย เป็นลักษณะที่สำคัญของ data center ในปัจจุบันและต่อๆไป ที่อยู่ PortLand [25] ปัญหาทั้งหมดที่มีสำหรับ multi-rooted fat-tree topology (รูปที่ 3) โดยเฉพาะอย่างยิ่ง สถาปัตยกรรมนำเสนอกลไกของ L2 routingที่ใช้คุณสมบัติของโครงสร้าง PortLand สนับสนุนฟังก์ชัน plug-and-pay สำหรับ L2 ที่ช่วยลดความยุ่งยากอย่างมีนัยยะสำคัญของเครือข่าย data center

แนวคิดหลักของ PortLand คือการใช้ที่อยู่ลำดับชั้น Pseudo MAC (PMAC) ของ VMs สำหรับ L2 routing โดยเฉพาะอย่างยิ่ง aPMAC มีรูปแบบของ pod.position.port.vmid ที่ pod เป็นหมายเลข pod ของขอบswitch, port คือหมายเลข port ของสวิตช์ hostปลายทางที่มีการเชื่อมต่อ และ vmid เป็น ID ของ VM ใช้งานบนhostปลายทาง fabric manager (process รันบนเครื่องเฉพาะ) เป็นผู้รับผิดชอบในการช่วยด้วย ARP resolution, multicast, and fault tolerance ตารางการส่งต่อในการใช้ k-port switch ที่แต่ละ switch ถูกจำกัดยังO(k) records เนื่องจากคุณสมบัติของtopology แบบ multi-rooted fat-tree ขอบของswitch เพื่อสลับ server hosting ที่มีการเชื่อมต่อของ VM การmap actual MAC (AMAC) ของ VM ไปยัง PMAC ตำแหน่งของ switch ใน topology อาจมีการ

ตั้งค่าด้วยตนเองของผู้ดูแลระบบหรือเป็นแบบอัตโนมัติผ่าน Location Discovery Protocol (LDP) ที่อาศัยคุณสมบัติของโครงสร้างพื้นฐาน

แม้ว่า PortLand จะมีประโยชน์ แต่ก็มีข้อจำกัดบางประการในด้านสถาปัตยกรรม เช่น มีความต้องการ multi-rooted fat-tree topology ทำให้ PortLand ใช้ไม่ได้กับเครือข่าย topology data center อื่นๆ การแก้ไข ARP ที่ร้องขอโดย single server ที่ทำให้สร้างสถาปัตยกรรมที่มีความเสี่ยงต่อการโจมตีบน fabric manager และอีกประการคือ แต่ละ edge switch ควรมีอย่างน้อย ครึ่งหนึ่งของ port ที่เชื่อมต่อกับ server




รูปที่ 7 แสดงสถาปัตยกรรม SEC2

*7. SEC2*

เพื่อให้แน่ใจว่าการนำมาใช้อย่างกว้างขวางของcloud computing บน data center ซึ่งเป็นสิ่งที่สำคัญที่จะให้บริการแก่ผู้เช่าโดยมีการรับรองความปลอดภัย โดยเฉพาะอย่างยิ่งหนึ่งในปัญหาด้านความปลอดภัยที่สำคัญ แยกเป็นเครือข่ายเสมือนทุ่มเทให้กับผู้เช่าที่แตกต่างกัน แม้ว่าการใช้ VLANs อาจจะเป็นทางออกที่มีศักยภาพสำหรับ สนับสนุนเครือข่ายที่แยกกันในdata center ก็ยังมีอีกหลายข้อจำกัดของ VLANs ประการที่หนึ่ง จำนวนสูงสุดของ VLANs คือ 4K เพราะขนาดพื้นที่ของ VLAN ID ประการที่สอง การควบคุมต่อผู้ใช้ของนโยบายความปลอดภัยเป็นสิ่งที่ท้าทาย ประการที่สาม การมีหมายเลข VLANs เป็นจำนวนมากในเครือข่าย data center อาจก่อให้เกิดความซับซ้อนในการจัดการเครือข่ายและค่าใช้จ่ายที่เพิ่มขึ้นในการควบคุม Secure Elastic Cloud Computing (SEC2) [28] มีเป้าหมายที่จะแก้ปัญหาข้อบกพร่องเหล่านี้โดยการแยกการส่งต่อpacketและการควบคุมการเข้าถึง

SEC2 เป็นสถาปัตยกรรมเครือข่าย data center ที่ใช้เทคนิค network virtualization เพื่อให้บริการรักษาความปลอดภัยที่ยืดหยุ่นบน cloud computing ดังที่แสดงในรูปที่ 7 Network virtualization ได้รับการสนับสนุนผ่าน Forwarding Elements (FEs) and a Central Controller (CC) FEs คือ Ethernet switches ที่สำคัญที่มีความสามารถที่จะควบคุมระยะไกลจาก CC ที่เก็บที่อยู่การ mapping และ policy databases FEs แสดงการทำ address mapping, การตรวจสอบ policy and การบังคับใช้, และการส่งต่อ packet สถาปัตยกรรมเครือข่ายมีสองระดับ คือ one core domain และ multiple edge domains incorporating physical hosts โดย edge domain จะได้รับมอบหมาย eid ที่ไม่ซ้ำค่ากัน และเชื่อมต่อไปยัง core domain โดยหนึ่งหรือ FEs จำนวนมาก ลูกค้าแต่ละ subnet จะมี cnet ID ที่มีค่าไม่ซ้ำ ที่ VM สามารถระบุได้โดย (cnet id, IP) เพื่อแยกลูกค้าที่แตกต่างภายในขอบแต่ละโดเมน SEC2 ใช้ VLAN ที่มีขอบเขต จำกัดใน edge domain เดียวกัน จึงช่วยลดข้อจำกัดของจำนวนลูกค้าที่สามารถได้รับการสนับสนุนเนื่องจากขนาดของ VLAN ID หากลูกค้ามีการเข้าถึงแบบสาธารณะเพื่อ VM จะทำให้ FE บังคับทุกๆ packet ภายนอกผ่าน firewall and NAT middleboxes ก่อนที่จะถึงเครือข่ายส่วนตัว ข้อดีของ SEC2 คือไม่จำเป็นต้องเฉพาะเราเตอร์หรือสวิทช์ที่ข้ามทั้งเครือข่าย data center นอกจากนี้ SEC2 ยังสนับสนุนการโยกย้าย VM [23] และ VPC (Virtual Private Cloud) service ซึ่งในแต่ละเครือข่ายส่วนตัวของผู้ใช้ใน cloud มีการเชื่อมต่อกับ on-site network ผ่าน Virtual Private เครือข่าย (VPN) [34]

หนึ่งในข้อจำกัดของ SEC2 คือหนึ่ง edge domain ไม่สามารถสนับสนุน VLANs ผู้เช่าที่แตกต่างกันที่มากกว่า 4K นอกจากนี้ตั้งแต่ FEs เพิ่ม MAC header เมื่อ VM ปลายทางไม่ได้อยู่ในedge domain ทำให้ SEC2 ต้องการสวิทช์ที่สนับสนุนเฟรมขนาดใหญ่

*8. SPAIN*

ในปัจจุบัน spanning tree protocol (STP) ใช้สำหรับ Ethernet LANs ขนาดใหญ่ที่จะไม่มีประสิทธิภาพในการสนับสนุนเครือข่าย data center ที่ทันสมัยเพราะไม่ได้ใช้ประโยชน์จากความหลากหลายเส้นทางที่นำเสนอโดยเครือข่าย data center ส่งผลในการจำกัด bandwidth เป็นสองส่วนและความน่าเชื่อถือที่ต่ำ Smart Path Assignment In Networks (SPAIN) [20] ใช้ VLAN สนับสนุนสินค้าที่มีอยู่ของ Ethernet switch ที่จะให้เผยแพร่ multipathing โดยพลการ

SPAIN คำนวณเส้นทางเคลื่อนระหว่างคู่ของ edge switch และ pre-configures VLANs เพื่อระบุเส้นทางเหล่านี้ end-host agent ที่ถูกติดตั้งบนทุกๆ host จะกระจายกระแสข้ามเส้นทางและ

VLANs เพื่อปรับปรุง load balancing และหลีกเลี่ยงความล้มเหลว โดย egent สามารถ เปลี่ยนการใช้ VLANs บาง flow นอกจากนี้ agent ยังตรวจพบเส้นทางที่ล้มเหลวและ re-routes packet รอบๆ ความล้มเหลวโดยใช้เส้นทางที่แตกต่าง

ในขณะที่ SPAIN ให้ multipathing และปรับปรุง load balancing และ fault-tolerance การนำเสนอนี้มีประเด็นเกี่ยวกับ scalability โดยเฉพาะแม้ว่าขั้นตอนalgorithm ในการคำนวณเส้นทางที่เสนอโดย SPAIN จะดำเนินการเฉพาะเมื่อ topology เครือข่าย ถูกออกแบบ หรือ มีการเปลี่ยนแปลงอย่างมีนัยยะสำคัญ โครงสร้างมีการคำนวณอย่างสูงสำหรับ topology ที่ความสลับซับซ้อน นอกจานี้ SPAIN ต้องเก็บ switch หลายรายการเพื่อทุกๆปลายทางและ VLAN คือสร้างความกดดันมากขึ้นบนตารางการส่งต่อ switch เมื่อเทียบกับที่ Ethernet พื้นฐานทำ ในอนาคต จำนวน path จะถูกจำกัด จำนวนของ VLANs ที่อนุญาตโดย 802.1q standard (4096) [19] สุดท้าย การดูแลรักษาตาราง mapping ระหว่าง flow และ VLANs หมายถึงค่าใช้จ่ายที่เพิ่มขึ้นในแต่ละ end-host

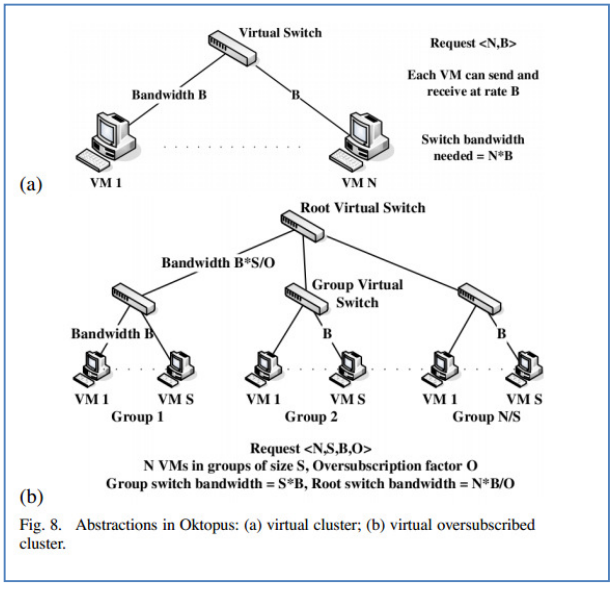


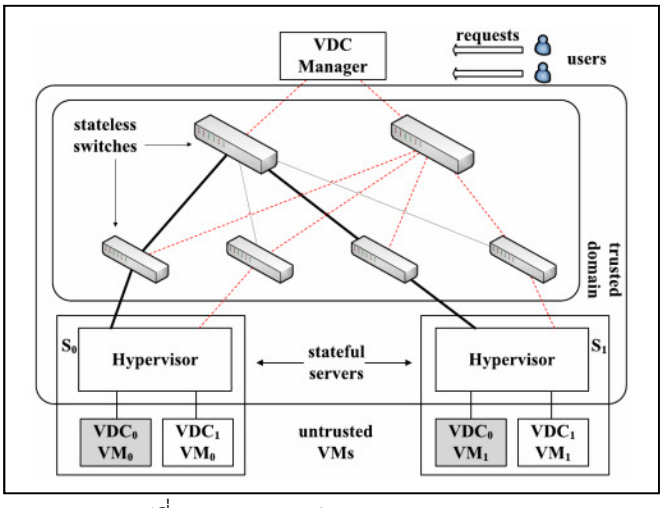
Fig. 8. Abstractions in Oktopus: (a) virtual cluster; (b) virtual oversubscribed cluster.

*9. Oktopus*

แม้ว่าผู้ให้บริการโครงสร้างให้ผู้เช่าได้ตามความต้องการทรัพยากรต่างๆผ่านการจัดสรร VM ใน data center ก็ยังไม่สนับสนุนการการันตีความสามารถบนทรัพยากรเครือข่ายแก่ผู้เช่า ความไม่ตรงกันระหว่างการตัดสินใจและการบรรลุเป้าหมายโดยนำไปสู่ปัญหาต่างๆ ขั้นแรก คือ ความแปรปรวนของประสิทธิภาพของเครือข่ายก่อให้เกิดประสิทธิภาพของโปรแกรมไม่แน่นอนในศูนย์ข้อมูลทำให้การจัดการประสิทธิภาพของโปรแกรม ต่อมา คือ ไม่สามารถคาดเดาประสิทธิภาพของเครือข่ายทำให้การผลิตและความพึงพอใจของลูกค้าลดลง นำไปสู่การสูญเสียรายได้ [16] Oktopus คือการดำเนินการ ของทั้งสองเครือข่ายเสมือน แบบนามธรรม (virtual cluster and virtual oversubscribed cluster) เพื่อควบคุมการออกระหว่างการการันตีประสิทธิภาพของสินค้าแก่ผู้เช่า ควบคุมราคา และรายได้ Oktopus ไม่เพียงเพิ่ม ประสิทธิภาพ application แต่ยังให้ความยืดหยุ่นที่มากกว่าแก่ผู้ให้บริการโครงสร้าง และช่วยให้ผู้เช่าในการหาสมดุลระหว่าง application ความสามารถสูงและราคาที่ถูก

virtual cluster แสดงในรูปภาพที่ 8 ผู้ให้บริการมี VM ที่เชื่อมต่อกับ single non-oversubscribed virtual switch ต่อไปจะมุ่งสู่application สำหรับข้อมูลจำนวนมาก เช่น MapReduce ที่มีลักษณะทั่วทั้งหมดโดยรูปแบบของtraffic ในรูปที่ 8 virtual oversubscribed cluster จำลอง cluster ที่เชื่อมต่อกันผ่านทาง virtual root switch ที่การใช้งานที่เหมาะสมกับรูปแบบการสื่อสารที่มีผู้เช่าสามารถเลือก สิ่งที่เป็นนามธรรมและระดับจาก oversubscription ของเครือข่ายเสมือนอยู่บนพื้นฐานของรูปแบบการสื่อสารของโปรแกรมผู้เช่าที่มีแผนจะปรับใช้ใน VDC (เช่น user-facing web-applications, data intensive applications) Oktopus ใช้ greedy algorithm เพื่อการจัดสรรทรัพยากรของ VDC

ข้อจำกัดหลักของ Oktopus คือจะทำงานได้เฉพาะกับเครือข่าย topologyแบบ tree-like physical ดังนั้น คำถามคือจะทำอย่างไรเพื่อการใช้ abstractions of Oktopus สำหรับ topologies อื่นๆ



รูปที่ 9 แสดงสถาปัตยกรรม SecondNet

10. SecondNet

SecondNet [15] มุ่งเน้นไปยังการการันตี bandwitdth ระหว่าง VMs ในหลายๆผู้เช่าของ virtualized data center นอกจากนี้การ

คำนวณและการจัดเก็บ ยังมีบัญชีสำหรับความต้องการbandwidth เมื่อใช้ VDC

องค์ประกอบหลักของสถาปัตยกรรม SecondNet แสดง ในรูปที่ 9 ที่ตัวจัดการ SecondNet ที่สร้าง VDCs บน เมทริกซ์ที่กำหนดการร้องขอของ bandwidth ระหว่างคู่ของ VMs โดย SecondNet กำหนดservice พื้นฐานไว้ 3 แบบ คือ ลำดับความสำคัญแบบสูง (type 0) แบบที่ดีกว่า best-effort service ที่ให้การการันตีสำหรับ bandwidth เพื่อ hopsแรกหรือสุดท้ายของเส้นทาง(type1) และ best-effort service (type2) SecondNet ใช้การปรับแต่งโครงสร้างการส่งต่อที่เรียกว่า port-switching source routing (PSSR) ที่ส่งต่อ packet โดยใช้ หมายเลขพอร์ตที่กำหนดไว้ล่วงหน้าแทน MAC address โดย PSSR จะปรับปรุงscalability ของ data plane เป็น path ที่มีการคำนวณที่โหนดต้นทาง ในทางนี้ switch กลาง ไม่ได้มีการตัดสินใจส่งต่อใดๆเกิดขึ้น

SecondNet ประสบความสำเร็จด้าน high scalability โดยการขเคลื่อนย้ายข้อมูลเกี่ยวกับ การจอง bandwidth จาก switch ไปยัง server hypervisor นอกจากนี้ SecondNet ยังให้ทรัพยากรอย่าง VM และ bandwidth มีการเพิ่มขึ้นแบบไดนามิค หรือ ลบออกจาก VDCs ในด้านการเคลื่อนย้าย SecondNet ยังสามารถจัดการความผิดพลาดและลดการกระจายตัวของทรัพยากร และ PSSR สามารถประยุกต์ใช้กับ Multiprotocol Label Switching (MPLS) [36] ที่ทำให้นำไปปรับใช้ได้ง่ายขึ้น ในด้านข้อจำกัดที่สำคัญของ SecondNet คือประสิทธิภาพการทำงาน อาจขึ้นอยู่กับโครงสร้างทางกายภาพของเครือข่าย และ SecondNet ไม่พิจารณาลักษณะการปฏิบัติงานอื่น ๆ ที่สามารถมีความสำคัญต่อผู้เช่าเช่น latency

*11. Gatekeeper*

Rodrigueset al. [26] ได้มองไปที่ปัญหาของเครือข่ายการแยกประสิทธิภาพการกำหนดความต้องการที่เกี่ยวข้อง และตอบโต้โครงการใหม่สนองความต้องการเหล่านั้นชื่อ Gatekeeper โดยในเปเปอร์นี้ยืนยันว่าวิธีแก้ปัญหาสำหรับการแยกประสิทธิภาพของเครือข่ายควรจะสามารถปรับขนาดในแง่ของจำนวน VMs ด้านการคาดการณ์ในแง่ของประสิทธิภาพเครือข่าย ด้านพฤติกรรมที่เป็นอันตรายต่อผู้เช่า และความยืดหยุ่นของการการันตีประสิทธิภาพ

Gatekeeper มุ่งเน้นไปยังด้านการรับประกัน bandwidth ระหว่าง banwidth ใน multiple-tenant data center และการบรรลุการใช้ banwidth ระดับสูงมักแสดงถึงการใช้ที่ไม่ค่อยมีประสิทธิภาพของ link bandwidth เมื่อความจุเพียงพอและกลับมาใช้งานได้ Gatekeeper address จะกำหนดอัตรามากน้อยสำหรับแต่ละคู่ VM และพารามิเตอร์ที่สามารถตั้งแต่ในขณะที่มั่นใจว่าlink capacitiesจะถูกใช้อย่างมีประสิทธิภาพโดยผู้เช่า

Gatekeeper จะสร้างlogical switchesที่มีการเชื่อมต่อระหว่างกันกับ VMs ไปยังผู้ใช้เดียวกัน โดย virtualNIC (vNIC) ของแต่ละ VM จะตรวจสอบ trafficที่เข้ามาโดยใช้ชุดของตัวนับ และรายงานความแออัดของ vNIC ของผู้ส่งที่เกินจากอัตราที่กำหนดไว้ และอัตราที่กำหนดโดยผู้ส่งจะใช้ข้อมูลควบคุม traffic rate เพื่อลดระดับความแออัด

Gatekeeper จะไม่ได้พิจารณาตัวชี้วัดประสิทธิภาพอื่น ๆ เช่น latency นอกจากนี้ Gatekeeper ยังอยู่ในการพัฒนาคุณสมบัติอื่นๆ เช่น การสร้างแบบไดนามิกและการลบข้อจำกัดต่างๆ

*12. CloudNaaS*

CloudNaaS [7] เป็นสถาปัตยกรรม virtual network ที่มีการสนับสนุนประสิทธิภาพสำหรับการปรับใช้และการจัดการองค์กรและ application ใน cloud โดยเฉพาะสถาปัตยกรรมที่เผยแพร่ชุดของ primitives ที่เหมาะสมกับความต้องการของการใช้งานทั่วไปรวมถึง application ที่เฉพาะเจาะจง, middlebox traversal, network broadcasting,VM grouping, and การจอง bandwidth

CloudNaaS ขึ้นอยู่กับการส่งต่อ OpenFlow เพื่อให้บรรลุวัตถุประสงค์ (เช่น middlebox traversal) ดังกล่าวข้างต้น การปรับปรุงแอพพลิเคชันใน CloudNaaS รวมถึงหลายๆ ขั้นตอน ขั้นแรกผู้ใช้ปลายทางระบุความต้องการเครือข่ายที่ควบคุม Cloud ใช้ primitives กำหนดโดยนโยบายภาษาเครือข่าย CloudNaaSขึ้นอยู่กับการส่งต่อ OpenFlow เพื่อให้บรรลุวัตถุประสงค์ (เช่น middlebox traversal) ดังกล่าวข้างต้น. การใช้งานโปรแกรมใน CloudNaaS รวมถึงหลายขั้นตอน. ครั้งแรกผู้ใช้ปลายทางระบุความต้องการเครือข่ายที่ควบคุมCloudใช้ primitives กำหนดโดยนโยบายภาษาเครือข่าย. หลังจากความต้องการเครือข่ายมีการแปลเป็นเมทริกซ์การสื่อสาร, ควบคุมCloudกำหนดตำแหน่งของ VMS และสร้างกฎเครือข่ายระดับที่สามารถติดตั้งสวิทช์ ปัจจุบัน CloudNaaS ใช้แก้ปัญหาการแก้ไขbin-packingสำหรับการวาง VMS ที่จะเข้าสู่การพิจารณาในการสื่อสารท้องถิ่น: นอกจากนี้ CloudNaaS ให้หลายเทคนิคที่จะลดจำนวนของรายการที่จำเป็นในการสลับกันซึ่งรวมถึง

(1) ความพยายามใช้เส้นทางเดียวสำหรับการจราจรที่ดีที่สุด

(2) การใช้เส้นทางการจราจรที่ถูก จำกัด สำหรับ QoS ขึ้นอยู่กับจำนวนของคลาสการจราจรที่กำหนดตามประเภทของ-of-Service บิต (ToS)

(3) กำหนดที่อยู่ติดกันเพื่อวาง VMs อยู่ด้านหลังedge swith เดียวกันและใช้บิตตัวแทนสำหรับรายการรวมส่ง IP

นอกจากนี้ยังสนับสนุนการ CloudNaaS กลไกออนไลน์สำหรับความล้มเหลวในการจัดการและการเปลี่ยนแปลงในข้อกำหนดนโยบายของเครือข่ายโดย reprovisioning VDCs. ปัจจุบัน CloudNaaS จะดำเนินการใช้สวิทช์ OpenFlow ที่ใช้สำหรับการส่งต่อ;โฮสต์สิ้นสุดเปิดใช้เครือข่าย stack vSwitch สำหรับการส่งต่อและสอดคล้องกับ OpenFlow ปัจจุบัน CloudNaaS จะดำเนินการใช้สวิทช์ OpenFlow ที่ใช้สำหรับการส่งต่อ;โฮสต์สิ้นสุดเปิดใช้เครือข่าย stack vSwitch สำหรับการส่งต่อและสอดคล้องกับ OpenFlow

ข้อ จำกัด หนึ่งของ CloudNaaS คือการ จำกัด การจราจรไปยังเส้นทางน้อยอาจนำไปสู่ความแออัดและ / หรือการใช้งานเครือข่ายขัดสน. หา trade-off ที่ดีระหว่างความยืดหยุ่นและเครือข่ายการใช้ยังคงเป็นปัญหาที่ท้าทายสำหรับ CloudNaaS

*13. Seawall*

[6] Seawall เป็นโครงการจัดสรรแบนด์วิดธ์ที่ช่วยให้. ผู้ให้บริการโครงสร้างพื้นฐานเพื่อกำหนดวิธีแบนด์วิดธ์จะที่ใช้ร่วมกันในเครือข่ายศูนย์ข้อมูลที่มีหลายผู้เช่า. ความคิดของ Seawall คือการกำหนดน้ำหนักไปยังหน่วยงานเครือข่ายการสร้างการจราจร (เช่น VMS, กระบวนการ),และจัดสรรแบนด์วิดธ์ตามน้ำหนักเหล่านี้ในทางสัดส่วน. Seawall ใช้อุโมงค์ที่ควบคุมความแออัดระหว่างคู่ของเครือข่ายหน่วยงานเพื่อบังคับใช้นโยบายแบนด์วิดธ์ร่วมกัน. shim layer ดำเนินการเป็น NDIS (Network Interface Specification Driver)ตัวกรองแพ็คเก็ตเป็นผู้รับผิดชอบ intercepting แพ็กเก็ตและปรับอัตราผู้ส่งที่ส่งแพ็คเก็ต

Seawall บังคับแยกแบนด์วิดธ์ระหว่างผู้ที่แตกต่างกันและป้องกันไม่ให้ผู้เช่าประสงค์ร้ายจากการบริโภคทรัพยากรเครือข่ายทั้งหมด นอกจากนี้ต้องการให้ Seawall ทางเครื่องกายภาพจะเก็บข้อมูลสถานะเท่านั้นสำหรับองค์กรของตัวเองซึ่งช่วยเพิ่มความสามารถในการรองรับ นอกจากนี้,ต้องการให้ Seawallเป็นเครื่องกายภาพรักษาข้อมูลสถานะเท่านั้นสำหรับเอนทิตี้ของตนเอง, ซึ่งช่วยเพิ่มปรับขยาย นอกจาก Seawall is agnosticสำหรับโปรโตคอลการขนส่งที่ใช้โดยผู้เช่า, จำนวนflowที่ใช้โดยองค์กรและและจำนวนของปลายเอนทิตี้ส่งเข้า ในทุกกรณี Seawall แบ่งแบนด์วิดธ์และการแยกตามสัดส่วนบังคับ.ยิ่งไปกว่านั้น, Seawall ช่วยให้น้ำหนักที่จะแก้ไขแบบไดนามิกเพื่อรองรับการเปลี่ยนแปลงในความต้องการของผู้เช่า.แม้ว่า Seawall ไม่ได้อยู่ที่ความล้มเหลวอย่างชัดเจนก็คือการปรับตัวเข้ากับเครือข่ายแบบไดนามิกเงื่อนไขทำให้เกิดความผิดพลาด.

ต้นแบบ Seawall เป็นครั้งแรกที่ดำเนินการเฉพาะใน Windows7 และ Hyper-V. ยิ่งไปกว่านั้นโดยปราศจากการควบคุมรับเข้าเรียน.มันน่าจะเป็นไปได้ที่ Seawall จะสามารถรับประกันแบนวิดธ์สำเร็จแน่นอนสำหรับจำนวนหน่วยงานที่เพิ่มขึ้น

*14. NetShare*

NetShare [27]รับมือปัญหาของจัดสรรแบนด์วิดในเครือข่ายข้อมูลเสมือนจริงศูนย์เสนอสถิติกลไกมัลติเพล็กที่ไม่ต้องมีการเปลี่ยนแปลงใด ๆ ในสวิทช์หรือเราเตอร์ NetShare ใช้การเชื่อมโยงการจัดสรรแบนด์วิดธ์สำหรับผู้เช่าในทางสัดส่วนและประสบความสำเร็จสูงสำหรับการให้บริการโครงสร้างพื้นฐาน ใน NetShare ศูนย์ข้อมูลเครือข่ายเชื่อมโยงร่วมกันระหว่างบริการการใช้งานหรือกลุ่มองค์กรมากกว่าการเชื่อมต่อแต่ละหมู่ ด้วยวิธีนี้, one service/application/กลุ่มไม่สามารถhogแบนด์วิธที่มีการเปิดการเชื่อมต่อมากขึ้น.

NetShare สามารถดำเนินการได้ในสามวิธีที่เป็นไปได้:การจัดสรรกลุ่ม, การควบคุมปริมาณอัตรา, และการจัดสรรจากส่วนกลาง NetShare ใช้จัดสรรกลุ่มในการจัดการการไหล TCP จัดสรรกลุ่มผู้ใช้ตามคิวเพื่อให้การจัดสรรแบนด์วิดธ์เป็นไปอย่างยุติธรรมในการให้บริการที่แตกต่างกันและจะดำเนินผ่าน Deficit Round Robin (DDR) [37]. การควบคุมปริมาณ Rate จะถูกใช้ในการควบคุมการจราจรที่สร้างโดยแหล่ง UDP และหลีกเลี่ยงการใช้แบนด์วิธมากเกินไปและมีการใช้งานผ่าน shim layer ด้านล่าง UDP ที่แต่ละโฮสต์. shim layerควบคุมอัตราการส่งโดยการวิเคราะห์การจราจรวัดที่ด้านรับและปรับอัตราการส่งตาม ที่จะใช้มากขึ้น นโยบายทั่วไปเช่นการจัดสรรแบนด์วิดธ์ที่ไม่ได้ใช้เฉพาะกระแสโครงการจัดสรรแบนด์วิดใช้ส่วนกลาง. NetShare อาศัยโปรโตคอลเส้นทางที่จะจัดการกับความล้มเหลวและmultipath เป็นไปได้ด้วยการใช้ ECMP.

scalability ของ NetShare scalability ของ NetShareอาจเป็นปัญหาเพราะคิวจะต้องมีการกำหนดค่าที่พอร์ตสวิตช์แต่ละสำหรับแต่ละบริการ /application นอกจากนี้ NetShare อาศัยคุณลักษณะเฉพาะของสวิตช์ศูนย์กลางการใช้กลไกการซึ่งจะช่วยลด deployability ของมัน

นอกจากนี้ NetShare มีวัตถุประสงค์เพื่อบรรลุความเป็นธรรมในการจัดสรรแบนด์วิดธ์และจึงไม่ได้ให้รับประกันใด ๆ กับแบนด์วิดธ์ที่แน่นอนที่จะให้บริการ.

## 4. การเปรียบเทียบ

ในขณะที่การสำรวจก่อนหน้านี้ส่วนข้อเสนอการวิจัยที่โดดเด่นและคุณสมบัติเด่นของพวกเขาในส่วนนี้เปรียบเทียบข้อเสนอเหล่านี้ใช้ชุดของตัวชี้วัดเชิงคุณภาพ โดยเฉพาะอย่างยิ่งเราประเมินข้อเสนอโดยใช้เกณฑ์ 5 อย่างต่อไปนี้: scalability, fault-tolerance, deployability, QoS support, และ load-balancing. scalability และ fault-tolerance มีความสำคัญเรื่องออกแบบสำหรับศูนย์ข้อมูลที่ประกอบไปด้วยเซิร์ฟเวอร์และทรัพยากรเครือข่ายจำนวนมากและคาดว่าจะสนับสนุนจำนวนของโปรแกรมประยุกต์ขนาดใหญ่ที่ผู้เช่า.

ในวันนี้ศูนย์ข้อมูลมักจะใช้ชุดสินค้าเซิร์ฟเวอร์และฮาร์ดแวร์เครือข่าย , deployability เป็นปัญหาสำคัญที่เกี่ยวกับวิธีการที่เปลี่ยนโครงสร้างพื้นฐานเป็นสิ่งที่จำเป็นสำหรับการดำเนินการโดยเฉพาะอย่างยิ่งสถาปัตยกรรม. QoS เป็นกังวลเกี่ยวกับผู้เช่าที่เพิ่มขึ้นและเป็นสิ่งสำคัญที่เป็นความสำเร็จของสถาปัตยกรรมศูนย์ข้อมูลเสมือนจริง. ในที่สุด loadbalancing เป็นวัตถุประสงค์ที่สำคัญของผู้ประกอบการเครือข่ายสำหรับวิศวกรรมจราจรและความแออัดในการลดศูนย์ข้อมูลเครือข่าย เราสรุปผลของการเปรียบเทียบของเราในตาราง III-VI. แต่ละตารางเปรียบเทียบข้อเสนอที่เฉพาะเจาะจงใช้เกณฑ์ของคุณลักษณะเฉพาะ ในหัวข้อย่อยต่อไปนี้เราจะให้อภิปรายรายละเอียดของการประเมินผลของแต่ละของเราตัวชี้วัดประสิทธิภาพของการทำงาน

| Proposal | References | ข้อดี | ข้อเสีย |
|---|---|---|---|
| Traditional DC | [4], [5], [19] | - มีการออกแบบที่เรียบง่าย<br>- สามารถกำหนด address space ของ layer 2 และ layer 3 ได้ด้วยตนเอง | - ไม่สามารถออกแบบเพื่อจัดการ VMs จำนวนมากได้<br>- มีการจำกัดจำนวนผู้เช่าไว้ที่ 4096 |
| Diverter | [21] | - สนับสนุนการแบ่งเชิงตรรกะของเครือข่าย IP | - ไม่สามารถรับประกัน Qos |
| NetLord | [22] | - อนุญาตให้ผู้เช่าออกแบบและปรับปรุงพื้นที่ผู้ใช้และแอพพลิเคชันได้ตามความต้องการ | - ไม่สนับสนุนการการันตี bandwidth |
| VICTOR | [23] | - ช่วยให้การโยกย้ายสามารถเก็บ IP address เดิมได้ | - จำเป็นต้องมีการสนับสนุน FIBs เป็นขนาดใหญ่ที่นำไปสู่ปัญหาในเรื่องของ scalability ที่เกี่ยวข้องกับ Fes |
| VL2 | [24] | - มีความยืดหยุ่นในด้านการจัดสรรทรัพยากร | - ขาดการการันตี bandwidth ที่แน่นอนระหว่างserver |
| PortLand | [25] | - มีความสามารถในการขยายประชากร VM มีประสิทธิภาพในการโยกย้าย VM และการจัดการที่ง่าย<br>- ช่วยลดความยุ่งยากของเครือข่าย data center | - มีความต้องการ multi-rooted fat-tree topology |
| Oktopus | [16] | - การันตีประสิทธิภาพของสินค้าแก่ผู้เช่า ควบคุมราคาและรายได้<br>- ยังให้ความยืดหยุ่นที่มากกว่าแก่ผู้ให้บริการโครงสร้างและช่วยให้ผู้เช่าในการหาสมดุลระหว่าง application | - ไม่สนับสนุนการการันตีความสามารถบนทรัพยากรเครือข่ายแก่ผู้เช่า<br>- ความแปรปรวนของประสิทธิภาพของเครือข่ายก่อให้เกิดประสิทธิภาพของโปรแกรมไม่แน่นอนในศูนย์ข้อมูล<br>- ทำงานได้เฉพาะกับเครือข่ายtopologyแบบ tree-like physical |
| SecondNet | [15] | - การันตี bandwitdth ระหว่าง VMs ในหลายๆผู้เช่าของ virtualized data center | - ประสิทธิภาพการทำงาน อาจขึ้นอยู่กับโครงสร้างทางกายภาพของเครือข่าย |

| | | | |
|---|---|---|---|
| | | - การคำนวณและการจัดเก็บ ยังมีบัญชีสำหรับความต้องการbandwidth เมื่อใช้ VDC | - ไม่พิจารณาลักษณะการปฏิบัติงานอื่น ๆที่สามารถมีความสำคัญต่อผู้เช่าเช่น latency |
| Seawall | [6] | - ใช้อุโมงค์ที่ควบคุมความแออัดระหว่างคู่ของเครือข่ายหน่วยงานเพื่อบังคับใช้นโยบายแบนด์วิดธ์ร่วมกัน<br>- ป้องกันไม่ให้ผู้เช่าประสงค์ร้ายจากการบริโภคทรัพยากรเครือข่าย ให้น้ำหนักที่จะแก้ไขแบบไดนามิกเพื่อรองรับการเปลี่ยนแปลงในความต้องการของผู้เช่า | - ปรับตัวเข้ากับเครือข่ายแบบไดนามิกเงื่อนไขทำให้เกิดความผิดพลาด |
| Gatekeeper | [26] | - รับประกัน bandwidth ระหว่าง banwidth ในmultiple-tenant data center | - ไม่ได้พิจารณาตัวชี้วัดประสิทธิภาพอื่น ๆ เช่น latency |
| NetShare | [27] | - เพื่อบรรลุความเป็นธรรมในการจัดสรรแบนด์วิดธ์<br>- อาศัยคุณลักษณะเฉพาะของสวิตช์ศูนย์กลางการใช้กลไกการซึ่งจะช่วยลด deployability | - ไม่ได้ให้รับประกันใด ๆ กับแบนด์วิดธ์ที่แน่นอนที่จะให้บริการ. |
| SEC2 | [28] | - ให้บริการรักษาความปลอดภัยที่ยืดหยุ่นบน cloud computing | - edge domain ไม่สามารถสนับสนุน VLANs ผู้เช่าที่แตกต่างกันที่มากกว่า 4K |
| CloundNaaS | [7] | - สนับสนุนประสิทธิภาพสำหรับการปรับใช้และการจัดการองค์กรและapplication ใน cloud | - จำกัด การจราจรไปยังเส้นทางน้อยอาจนำไปสู่ความแออัดและ / หรือการใช้งานเครือข่ายขัดสน |

ตารางที่ 3 แสดงถึงข้อดีข้อเสียของศูนย์มูลแต่ละประเภท

*A. Scalability*

ประสบความสำเร็จอย่างสูงกับความยืดหยุ่นในศูนย์ข้อมูลเสมือนต้องใช้พื้นที่ที่อยู่ที่สนับสนุนผู้เช่าจำนวนมากและ VMS ของพวกเขานอกจากนี้ ตั้งแต่ปัจจุบัน commodity สวิทช์มักมีขนาดหน่วยความจำที่จำกัด มันเป็นสิ่งจำเป็นที่จะทำให้จำนวนของ forwarding states ในสวิตช์แต่ละขั้นเพื่อให้เกิดความยืดหยุ่น

| Proposal | Scalability | Fault-tolerance | Deployability | QoS | Load balancing |
|---|---|---|---|---|---|
| Traditional DC | Low | No | High | No | No |
| Diverter | High | Yes | High | No | No |
| NetLand | High | No | Low | No | Yes |
| VICTOR | Low | Yes | Low | No | No |
| VL2 | High | Yes | Low | No | Yes |
| PortLand | High | Yes | Low | No | Yes |
| SecondNet | High | Yes | High | Yes | No |
| SEC2 | Low | No | Low | No | No |
| CloudNaaS | Low | Yes | Low | Yes | No |

ตารางที่ 4 แสดงเปรียบเทียบคุณภาพของโครงสร้างการส่งต่อ

| Proposal | Scalability | Fault-tolerance | Deployability | QoS | Load balancing |
|---|---|---|---|---|---|
| Traditional DC | Low | No | High | No | No |
| SPAIN | Low | Yes | High | No | Yes |
| NetLord | High | No | Low | No | Yes |
| VL2 | High | Yes | Low | No | Yes |
| PortLand | High | Yes | Low | No | Yes |
| Traditional DC | Low | No | High | No | No |
| SPAIN | Low | Yes | High | No | Yes |
| NetLord | High | No | Low | No | Yes |
| VL2 | High | Yes | Low | No | Yes |

ตารางที่ 5 แสดงการเปรียบเทียบคุณภาพเกี่ยวกับ multipathing

| Proposal | Scalability | Fault-tolerance | Deployability | QoS | Load balancing |
|---|---|---|---|---|---|
| Oktopus | High | Yes | High | Yes | No |
| SecondNet | High | Yes | High | Yes | No |
| Gatekeeper | High | Yes | High | Yes | No |
| CloudNaaS | Low | Yes | Low | Yes | No |

ตารางที่ 6 แสดงการเปรียบเทียบคุณภาพการรับประกันBandwidth

| Proposal | Scalability | Fault-tolerance | Deployability | QoS | Load balancing |
|---|---|---|---|---|---|
| Seawall | High | Yes | High | No | No |
| NetShare | Low | Yes | Low | No | Yes |

ตารางที่ 6 แสดงการเปรียบเทียบคุณภาพ Bandwidth ที่ใช้ร่วมกัน

ตาราง VIIแสดงจำนวนตัวเลขสูงสุดของผู้เช่า, VMs ต่อผู้เช่า และขนาดของตารางการส่งต่อตัวเลขสูงสุดของผู้เช่าและ VMS ส่วนใหญ่ขึ้นอยู่กับจำนวนบิตที่ใช้ในการระบุผู้เช่าและ VMSจำนวนของ VMS ต่อผู้เช่าขึ้นอยู่กับพื้นที่ที่อยู่ที่สนับสนุนโดย IPv4 ซึ่งสามารถยืดออกเมื่อใช้ IPv6ทั้งนี้ขึ้นอยู่กับการส่งต่อโครงร่างขนาดของตารางการส่งต่อขึ้นอยู่กับจำนวนของ VMS, physical machines, switches or pods.ในทางปฏิบัติจำนวนของ VMS สูงกว่าจำนวนเครื่องทางกายภาพซึ่งในทางกลับกัน.... สูงกว่าจำนวนสวิทช์นอกจากนี้เรายังสังเกตเห็นว่าVICTOR and Portland ไม่สามารถสนับสนุนหลายผู้เช่า

ท่ามกลางสถาปัตยกรรมที่สำรวจในเอกสารนี้ ,Second-Net,Seawall,Gatekeeperบรรลุความยืดหยุ่นสูงโดยการรักษาสถานะที่ end-hosts (เช่น hypervisors) มากกว่าในสวิทช์. NetLord และ VL2 บรรลุ scalability สูงผ่านencapsulation packet รักษาสถานะการส่งต่อเฉพาะสำหรับสวิทช์ในเครือข่าย. diverter ยังสามารถปรับขนาดได้เพราะตารางการส่งต่อของสวิทช์ที่มีอยู่ เฉพาะ MAC addresses ของโหนดทางกายภาพ (ไม่ใช่ของ VMS). ตรงกันข้าม , SPAIN, VICTOR, and CloudNaaS จะปรับขนาดได้น้อยลงเพราะพวกเขาต้องการการบำรุงรักษาของสถานะต่อ VM ในการ switching/forwarding element. แม้ว่าการเพิ่มประสิทธิภาพให้ CloudNaaS สำหรับการปรับปรุง scalability บางอย่างเช่นการเพิ่มประสิทธิ ข้อจำกัดเส้นทางความหลากหลายที่ให้บริการในเครือข่ายและการเสื่อมประสิทธิภาพโดยรวมของวิธีการ

ในอนาคต,CloudNaaS ถูกนำมาใช้อยู่ในปัจจุบันใช้ OpenFlow, และ OpenFlow เป็นที่รู้จักกันมีปัญหาเรื่อง scalability ในศูนย์ข้อมูลขนาดใหญ่เนื่องจากการใช้ตัวควบคุมส่วนกลาง. SEC2 ไม่ได้ปรับขนาดเพราะaddressing schema จำกัดจำนวนผู้เช่าและซับเน็ต ที่สนับสนุนในเครือข่าย. NetShare อาศัยส่วนกลางจัดสรรแบนด์วิดธ์ซึ่งทำให้มันยากที่จะวัดสำหรับศูนย์ข้อมูลขนาดใหญ่.

*B. Fault-tolerance*

ในบริบทของศูนย์ข้อมูลเสมือนจริง, fault-tolerance ครอบคลุมการจัดการความล้มเหลวขององค์ประกอบในระนาบข้อมูล (เช่น สวิทช์และการเชื่อมโยง) และระนาบควบคุม (เช่นระบบค้นหา) เราพบว่าส่วนใหญ่ของสถาปัตยกรรมที่แข็งแกร่งกับความล้มเหลวในส่วนประกอบระนาบข้อมูล. ยกตัวอย่างเช่น SecondNet ใช้ spanning treeสืบทอดสัญญาณช่องที่จะตรวจสอบความล้มเหลวและขั้นตอนวิธีการจัดสรรในการจัดการกับพวกเขา. ตัวแทนSPAIN สามารถสลับระหว่าง VLANs ในการเกิดขึ้นของความล้มเหลว, NetLord อาศัยSPAIN สำหรับ fault-tolerance และ VL2 และ NetShare พึ่งพาเส้นทางโปรโตคอล (OSPF). Diverter, VICTOR, and SEC2 ใช้โครงสร้างพื้นฐานการส่งต่อพื้นฐานสำหรับการกู้คืนความล้มเหลว. แผนดังกล่าวเป็น Oktopus และCloudNaaSที่จับความล้มเหลว โดยคำนวณการจัดสรรแบนด์วิดใหม่สำหรับเครือข่ายที่ได้รับผลกระทบ. แผนการรวมทั้ง Seawall และGatekeeper สามารถปรับให้เข้ากับความล้มเหลวโดยการคำนวณการจัดสรรอัตราของแต่ละ flow.

การควบคุมส่วนประกอบระนาบในศูนย์ข้อมูลสถาปัตยกรรมเครือข่ายรวมถึงระบบการค้นหาส่วนกลางสำหรับการแก้ปัญหาคำสั่งที่อยู่ (NetLord, VICTOR, VL2, Portland, SEC2), เทคโนโลยีการจัดการการไหลของส่วนกลาง (ใช้ CloudNaaS OpenFlow), tree spanning สัญญาณ (SecondNet) และโปรโตคอลการกำหนดเส้นทาง (NetShare และ VL2).ความล้มเหลวของการควบคุมระนาบเหล่านี้ชิ้นส่วนที่สามารถนำไปสู่การทำงานของบางส่วนหรือทั้งหมดศูนย์ข้อมูลและผลในการไร้ความสามารถในการตรวจสอบความล้มเหลวในข้อมูลระนาบ.

ความล้มเหลวของการควบคุมระนาบเหล่านี้ส่งผลกระทบต่อความล้มเหลวในสถาปัตยกรรมที่มีระนาบควบคุมตามspanning

treeโปรโตคอล ที่ขึ้นอยู่กับเวลาที่โปรโตคอลใช้ในการมาบรรจบหลังจากมีการเปลี่ยนแปลงโครงสร้าง.

การปรับตัวในโปรโตคอลพื้นฐาน spanning tree protocolเช่น Rapid Spanning Tree Protocol (RSTP)สามารถลดเวลาการบรรจบ ความล้มเหลวที่คล้ายกับ STP ในกรณีของการกำหนดเส้นทางโปรโตคอลเช่น OSPF ต้องคำนวณเส้นทางใหม่ซึ่งอาจใช้เวลาเวลาตัวแปรขึ้นอยู่กับขนาดของเครือข่ายและการกำหนดค่าโปรโตคอลปัจจุบัน.อย่างไรก็ตามดังที่แสดง [24], เวลาการบรรจบกันของ OSPF (น้อยกว่าหนึ่งวินาที) ไม่ได้เป็นปัจจัยที่ต้องห้ามในแบบ real เครือข่ายข้อมูลศูนย์

OpenFlow ใช้โดย CloudNaas มีพื้นฐานอยู่บนส่วนกลางตัวควบคุมที่กำหนดพฤติกรรมของ OpenFlow ใช้โดย CloudNaas มีพื้นฐานอยู่บนส่วนกลางตัวควบคุมที่กำหนดพฤติกรรมของ OpenFlow ตามสวิทช์ผ่านชุดของกฎและการกระทำที่เกี่ยวข้อง. การออกแบบส่วนกลางควบคุมของ OpenFlow มันสามารถส่งความล้มเหลวและเพิ่มประสิทธิภาพ bottlenectks. [39] HyperFlow เป็นข้อเสนอเป้าหมายในการให้ส่วนกลาง logically แต่กายภาพกระจายตัวควบคุม OpenFlow. ใน HyperFlow เมื่อความล้มเหลวที่เกิดขึ้นในหนึ่งตัวควบคุม, สวิทช์ที่เกี่ยวข้องกับตัวควบคุมความล้มเหลวจะ reconfigured เพื่อสื่อสารกับตัวควบคุมอื่นให้สามารถใช้ได้

ระบบกระจายการค้นหาสามารถใช้ในการลดผลกระทบเชิงลบของความล้มเหลวในระบบการค้นหาที่อยู่ สำหรับตัวอย่างเช่น VL2 สถาปัตยกรรมแนะนำใช้การจำลองแบบของสถานะเซิร์ฟเวอร์ (RSM) เครื่องที่จะใช้การจำลองแบบไดเรกทอรีระบบซึ่งช่วยให้ความน่าเชื่อถือที่มีผลต่อประสิทธิภาพ

*C. Deployability*

จากดังกล่าวข้างต้นเป็น, deployability เป็นลักษณะสำคัญของศูนย์เครือข่ายใด ๆ ของสถาปัตยกรรมข้อมูล virtualization. ในการเปรียบเทียบของเราสรุปไว้ในตาราง III-VI, เราประเมิน deployability ของสถาปัตยกรรมที่สูงถ้าสถาปัตยกรรมสามารถใช้งานผ่านสวิทช์สินค้าที่มีการปรับเปลี่ยนซอฟแวร์. บนมืออื่น ๆ , deployability ต่ำหมายถึงสถาปัตยกรรมที่ต้องการอุปกรณ์ที่มีคุณสมบัติที่ไม่สามารถใช้ได้ในสวิทช์ทุกครั้ง (เช่นการส่งต่อการสนับสนุน L3 เฉพาะโปรโตคอล)

เราสรุปรายละเอียดการเปรียบเทียบของเราเกี่ยวกับ deployability ในตารางที่ VIII ซึ่งอธิบายคุณลักษณะที่จำเป็นที่จะต้องดำเนินการใน hypervisors (ในเครื่องทางกายภาพ) edge สวิทช์และสวิทช์หลัก. Commodity switchesส่วนใหญ่ส่งต่อสนับสนุน L2 และเทคโนโลยี VLAN ในขณะที่Commodity hypervisors สร้างเพียง isolated VA ส่งต่อสคีมาระบุกฎสำหรับการส่งแพ็กเก็ตโดยองค์ประกอบเปลี่ยนจากพอร์ตขาเข้ากับพอร์ตขาออก. FIB ช่วยให้การ map ที่อยู่ MAC กับพอร์ตสวิตช์เมื่อการตัดสินใจเกี่ยวกับการแพ็คเก็ต forwarding.Ms ตารางยังแสดงให้เห็นซึ่งโครงการต้องมีเซิร์ฟเวอร์เป็นศูนย์กลางการจัดการ.ทั้งนี้ขึ้นอยู่กับรูปแบบการเซิร์ฟเวอร์นี้สามารถมีฟังก์ชันที่แตกต่างกันเช่นการจัดการที่อยู่ Portland,VL2), การจัดการผู้เช่า(NetLord และ VL2), เส้นทางการคำนวณ (VICTORและ SEC2), และการจัดสรรทรัพยากร (SecondNet)

เราสังเกตุว่าในขณะที่บางการสำรวจสถาปัตยกรรม(SPAIN, Diverter, and Seawall) ต้องเปลี่ยนเฉพาะในhypervisor ส่วนใหญ่ของการสำรวจสถาปัตยกรรมต้องใช้คุณสมบัติของฮาร์ดแวร์โดยเฉพาะอย่างยิ่งคุณสมบัติเหล่านี้รวม MACin-MAC (SEC2) encapsulation, L3 forwarding (VL2, Net-Lord), DRR (NetShare), เครือข่ายบริการไดเรกทอรี (NetLord, VL2, Portland, VICTOR, SEC2) และฮาร์ดแวร์โปรแกรม(CloudNaaS) ที่ไม่ง่ายในการสนับสนุนcommodityสวิทช์ ดังนั้นการดำเนินการสถาปัตยกรรมเหล่านั้นสามารถเพิ่มค่าใช้จ่ายรวมของเครือข่าย อย่างไรก็ตามกับฮาร์ดแวร์วิวัฒนาการและการยอมรับอย่างกว้างของฮาร์ดแวร์โปรแกรมมันไม่ได้ยกเว้นว่าเทคโนโลยีเหล่านี้จะกลายเป็นตำแหน่งพื้นฐานในอนาคตอันใกล้

สุดท้ายเราต้องการจะพูดถึงการจัดการข้อมูลศูนย์กลางมีแนวโน้มที่จะใช้อุปกรณ์สินค้าที่มีราคาถูกและเปลี่ยนได้อย่างง่ายดาย อุปกรณ์นี้ไม่สามารถใช้ได้เสมอมันมีความยืดหยุ่นไม่เพียงพอ ยกตัวอย่างเช่นในกรณีของศูนย์ข้อมูลแบบดั้งเดิมสวิทช์ commodity ต้องเก็บMAC addresses ของ VMsโฮสต์. มันเจือจางปัญหา scalability เพราะสวิทช์commodity มักจะมีจำนวน จำกัดของทรัพยากร (เช่นขนาดของตาราง FIB) [22]. อย่างไรก็ตามโครงร่างที่เสนอในการส่งต่อ NetLord เรียกcommodity สวิทช์ในการจัดเก็บที่อยู่ MAC เฉพาะedge switches. จำนวนของสวิตช์ถูกมากน้อยกว่าจำนวนของ VMS ในศูนย์ข้อมูลอย่างมากช่วยเพิ่มความยืดหยุ่น. The number of switches being much smaller than the number of VMs in a data center drastically improves scalability. ในสถาปัตยกรรมทั้งแบบธรรมดาและ NetLord สินค้า

สวิทช์ที่ใช้ แต่โครงร่างส่งต่อทำให้ความแตกต่างจึงมีปัญหา scalability no ใน NetLord.

*D. QoS Support*

QoS ในเครือข่ายเสมือนจะประสบความสำเร็จโดยการจัดสรรรับประกันแบนด์วิดธ์สำหรับแต่ละการเชื่อมโยงเสมือน Oktopus, SecondNet, Gatekeeper และ CloudNaaS ให้การรับประกันแบนด์วิธจัดสรรสำหรับแต่ละเครือข่ายเสมือน. ในขณะ, Seawall และ NetShare ให้บริการ weighted fair-sharing ที่ยุติธรรมของแบนด์วิดธ์ระหว่างผู้เช่าอย่างไรก็ตามพวกเขาไม่ให้การรับประกันเกี่ยวกับการจัดสรรแบนด์วิดว่าไม่เหตุการ์ที่ไม่ประสิทธิภาพ. ในขณะที่ส่วนที่เหลือสถาปัตยกรรมไม่ได้หารือเกี่ยวกับประเด็น QoS เราเชื่อว่ามันเป็นไปได้ที่จะสนับสนุน QoS ในสถาปัตยกรรมเหล่านี้โดยถูกต้องรวมพวกเขากับคนที่รับประกันแบนด์วิธสนับสนุน (เช่นการใช้มาตรการOktopus เป็น NetLord)

*E. Load-balancing*

โหลดสมดุลเป็นคุณลักษณะที่พึงประสงค์สำหรับการลดเครือข่ายขณะที่การปรับปรุงความแออัดทรัพยากรเครือข่ายและประสิทธิภาพของโปรแกรม.ระหว่างสถาปัตยกรรมที่สำรวจในเอกสารนี้, SPAIN and NetLord(ซึ่งอาศัยSPAIN ประสบความสำเร็จเรื่อง load-balancingโดยการกระจายการจราจรหลายspanning trees. เพื่อให้บรรลุ load balancing และตระหนักถึง multipathing, Portland and VL2 พึ่งพา ECMP และ VLB.สุดท้าย Diverter, VICTOR, and SEC2 เป็นหลักที่อยู่รูปแบบที่ไม่ชัดเจนอยู่load-balancing.

*F. Summary*

การเปรียบเทียบของเราที่แตกต่างกันที่นำเสนอสถาปัตยกรรมเปิดเผยการสำรวจหลายหัวข้อ. หนึ่ง ไม่มีทางออกที่ดีสำหรับทุกปัญหา ที่ควรจะเลือกที่อยู่ในคอนเทน ศูนย์กลางข้อมูลเครือข่าย. นี้เป็นหลักเนื่องจากสถาปัตยกรรมของแต่ละสถาปัตยกรรมพยายามที่จะมุ่งเน้นไปที่ลักษณะเฉพาะของ virtualization ศูนย์ข้อมูล. ในขณะ เราเชื่อว่ามันเป็นไปได้ที่จะรวมคุณสมบัติที่สำคัญบางส่วนของสถาปัตยกรรมที่ ใช้ประโยชน์จากผลประโยชน์ของตน. ยกตัวอย่างเช่นมันเป็นไปได้รวม VICTOR และ Oktopus ในการปรับใช้เวอร์ช่วลศูนย์ข้อมูลที่มีแบนด์วิดธ์ขณะที่การให้รับประกันสนับสนุนที่มีประสิทธิภาพสำหรับ VM โยกย้าย. ประการที่สอง, การหาสถาปัตยกรรมที่ดีที่สุด (หรือการรวมกัน) ต้องใช้ความเข้าใจอย่างระมัดระวังจากความต้องการด้านประสิทธิภาพของการใช้งานที่อาศัยอยู่ในศูนย์ข้อมูล. ดังนั้นประเด็นที่กล่าวถึงในส่วนนี้ ต้องใช้ความพยายามส่งเสริมการวิจัยในบริบทที่แตกต่างกันของcloudสภาพแวดล้อม.

## 5. งานวิจัยในอนาคต

ในส่วนนี้เราจะหารือบางส่วนของทิศทางที่สำคัญสำหรับสำรวจในอนาคตเกี่ยวกับข้อมูลเครือข่ายเสมือนจริงศูนย์

*A. Virtualized Edge Data Centers*

ที่สุดของการศึกษาที่มีอยู่จนถึงขณะนี้กับข้อมูลศูนย์เครือข่าย virtualization ได้มุ่งเน้นในศูนย์ข้อมูลขนาดใหญ่ที่มีหลายพันเครื่อง แม้ว่าข้อมูลที่มีขนาดใหญ่ economy-of-scaleและมีการจัดการสูงเนื่องจากธรรมชาติส่วนกลางของพวกเขา, พวกเขามีข้อจำกัด โดยธรรมชาติของพวกเขาเมื่อมันมาถึงบริการโฮสติ้ง. โดยเฉพาะอย่างยิ่งปัจจัยทางเศรษฐศาสตร์บอกว่าจะมีเพียงไม่กี่คนของข้อมูลขนาดใหญ่สร้างศูนย์ขึ้นในสถานที่ที่การก่อสร้างและการดำเนินงานค่าใช้จ่าย (เช่นพลังงาน) ต่ำ [40]เป็นผลให้ศูนย์ข้อมูลเหล่านี้อาจจะอยู่ห่างไกลจากผู้ใช้ส่งผลให้ต้นทุนการสื่อสารที่สูงขึ้นและคุณภาพการให้บริการอาจมีsub-optimalในแง่ของ delay, jitter and throughput

แรงบันดาลใจจากการสังเกตนี้ข้อเสนอล่าสุดเช่น Mist [41] EdgeCloud [42] ศูนย์ไมโครข้อมูล [43], ศูนย์ nanodata [44] ได้รับการหยิบยกขึ้นเพื่อสนับสนุนการสร้างศูนย์ขนาดเล็กของข้อมูลสำหรับการให้บริการโฮสติ้งที่เครือข่ายedge (เช่นการเข้าถึงเครือข่าย)ที่สามารถจัดการบริการใกล้กับผู้ใช้.

ในบททความนี้เราใช้ edge data centers เป็นคำศัพท์เฉพาะสำหรับอ้างถึงศูนย์ข้อมูลเครือข่ายขนาดเล็กที่ตั้งบนเครือข่าย edge.ขณะที่การวัดค่าใช้จ่ายกับผลการดำเนินการไม่เหมือนศูนย์ข้อมูลขนาดใหญ่ edge data center เสนอผลประโยชน์ที่สำคัญเปรียบเทียบกับศูนย์ข้อมูลระยะไกล [43]:

(1) พวกเขาสามารถนำเสนอ Qos ที่ดีกว่าสำหรับความล่าช้าของข้อมูล(delay-sensitive) มีผลต่อประสิทธิภาพการทำงานของแอพพลิชั่น เช่น วิดีโอสตรีมมิ่ง,เกมออนไลน์,โทรศัพท์และการประชุมทางเว็บ

(2) พวกเขาสามารถลดค่าใช้จ่ายเครือข่ายการสื่อสารโดยการลดการจราจรข้ามเส้นทางในการให้บริการเครือข่าย

(3) ต้นทุนในการสร้างของ edge data centers จะต่ำกว่าเมื่อเทียบกับข้อมูลจากศูนย์ข้อมูลระยะไกลขนาดใหญ่ ในความเป็นจริงการสื่อสารโทรคมนาคมที่มีอยู่จำนวนมากและอินเทอร์เน็ตผู้ให้บริการ (ISP) ยินดีที่จะยกระดับของพวกเขาที่มีอยู่โครงสร้างพื้นฐานเพื่อให้บริการเพิ่มมูลค่าโดยใช้edge datacenters [45] ดังนั้นจึงคาดว่าในอนาคต cloud จะเป็นโครงสร้างพื้นฐานแบบหลายชั้นที่ edge data centers จะเสริมศูนย์ข้อมูลระยะไกลในการให้บริการที่มีคุณภาพสูงและให้บริการออนไลน์ที่มีต้นทุนต่ำ

คล้ายกับศูนย์ข้อมูลขนาดใหญ่, virtualization เป็นสิ่งจำเป็นในศูนย์ข้อมูลที่ทันสมัยสำหรับ VDCs สนับสนุนจากหลายผู้เช่าโดยมีวัตถุประสงค์ที่หลากหลายและประสิทธิภาพการทำงานเป้าหมายการจัดการอย่างไรก็ตามvirtualizing edge data centers ยังมีประเด็นหลาย ที่มีความท้าทายในงานวิจัยใหม่:

-สำหรับผู้ให้บริการอย่างใดอย่างหนึ่งปัญหาพื้นฐานคือวิธีการที่ดีที่สุดที่แบ่งโครงสร้างพื้นฐานบริการระยะไกลระหว่างศูนย์และ EDGEเพื่อให้บรรลุ tradeoff ดีที่สุดระหว่างประสิทธิภาพและต้นทุนการดำเนินงาน? ปัญหานี้เป็นที่รู้จักกันทั่วไปว่าเป็นปัญหาที่เกิดขึ้นกับตำแหน่งบริการ[46] หาวิธีการแก้ปัญหานี้เป็นสิ่งจำเป็นสำหรับผู้ให้บริการที่จะใช้ข้อมูลedgeสถาปัตยกรรมศูนย์บริการตามปัญหานี้การแชร์หลายๆอย่างคล้ายกับปัญหาที่เกิดขึ้นแบบจำลองแบบดั้งเดิมของตำแหน่งอย่างไรก็ตามวิธีการที่มีอยู่ยังไม่ได้ศึกษากรณีแบบไดนามิกที่เงื่อนไขความต้องการและระบบ (ราคาทรัพยากรเช่นและเครือข่ายเงื่อนไข) สามารถเปลี่ยนแปลงได้ตลอดเวลาในกรณีนี้หากการตั้งค่าตำแหน่งจะต้องเปลี่ยนแปลง,มันก็ยังเป็นสิ่งที่จำเป็นที่จะต้องพิจารณาค่าใช้จ่ายของการปรับโครงสร้าง(เช่น โยกย้าย VM) ในรูปแบบการเพิ่มประสิทธิภาพ

-วิธีการจัดการอย่างมีประสิทธิภาพบริการพื้นที่ในหลายศูนย์ข้อมูล? ที่จะมีจำนวนมากของขอบศูนย์ข้อมูลการตรวจสอบและทรัพยากรในการควบคุมดังกล่าวโครงสร้างพื้นฐานขนาดใหญ่มีความท้าทายและอาจมีค่าใช้จ่ายอย่างมีนัยสำคัญ การลดoverhead การจัดการเป็นประเด็นหลักของการตรวจสอบ

เราเชื่อว่าที่อยู่กับความท้าทายในการวิจัยข้างต้นจะสิ่งสำคัญต่อความสำเร็จของแบบหลายขั้นโครงสร้างพื้นฐานคลาวด์.

*B. Virtual data center embedding*

สามารถรองรับ VDCs จำนวนมาก ขึ้นอยู่กับการทำแมปปิ้ง virtual resourcesอย่างมีประสิทธิภาพทางกายภาพหนึ่ง.นี้ปัญหาคือปกติจะเรียกว่าฝังแบบและได้รับเรื่องของการวิจัยในบริบทของเครือข่าย[48] virtualization - [51]. สถาปัตยกรรมศูนย์ข้อมูลเช่น Second-Net และ Oktopus [16] heuristics ตรงกันข้าม ได้เสนอที่จะรับมือด้วย NP-hardness ของปัญหาการฝัง.อย่างไรก็ตามมีปัญหาอื่น ๆ หลายประการเกี่ยวกับการออกแบบศูนย์ข้อมูลเสมือนอัลกอริทึม embedding.

- ในศูนย์ที่ข้อมูล virtualized มีทรัพยากรอื่นๆ นอกเหนือจากเซิร์ฟเวอร์ทางกายภาพที่สามารถ virtualized ซึ่งรวมถึงเราเตอร์สวิตช์ อุปกรณ์เก็บข้อมูล และระบบรักษาความปลอดภัย ที่มีอยู่ไว้ใช้แก้ปัญหาการฝังสำหรับศูนย์ข้อมูลได้มุ่งเน้นเฉพาะในการบรรลุการฝัง VM ที่ดีที่สุดเพื่อตอบสนอง ความต้องการแบนด์วิดธ์และการประมวลผล เราเชื่อว่า อัลกอริทึมการฝังสำหรับ VDCs ควรพิจารณาความต้องการสำหรับทรัพยากรอื่น ๆ ด้วย
- ความต้องการของทรัพยากรสำหรับการใช้งานศูนย์ข้อมูลสามารถเปลี่ยนช่วงเวลาซึ่งหมายความว่าการฝัง VDCs ที่อาจมีการเปลี่ยนแปลง ในการออกแบบอัลกอริทึมใหม่ VDC ที่คำนึงถึงค่าใช้จ่ายการปรับโครงสร้างของ VMS (เช่น ต้นทุนการโยกย้าย) และ virtua topologies ยังคงเป็นปัญหาในการวิจัย
- ปริมาณการใช้พลังงานเป็นปัญหาใหญ่ในศูนย์ข้อมูลตั้งแต่นั้นบัญชีสำหรับจำนวนข้อมูลที่สำคัญต้นทุนการดำเนินงานศูนย์ โดยว่าตามรายงาน Carey [52] ของที่สำนักงานคุ้มครองสิ่งแวดล้อม (EPA) ศูนย์ข้อมูลการบริโภคประมาณ 3% ของการใช้ไฟฟ้าทั้งหมดของสหรัฐในปี 2001 นอกจากนี้การใช้พลังงานโดยประมาณของเซิร์ฟเวอร์ศูนย์ข้อมูลเป็นประมาณ 2% ของกระแสไฟฟ้าของโลก ที่สอดคล้องกับ [53] อุปกรณ์คอมพิวเตอร์ของทั่วไป 5000 ตารางฟุต ส่วนศูนย์ข้อมูลซึ่งรวมถึงโปรเซสเซอร์อุปกรณ์ไฟฟ้าเซิร์ฟเวอร์ส่วนประกอบเซิร์ฟเวอร์อื่น ๆ ที่เก็บและสื่อสารอุปกรณ์สิ้นเปลือง 52% ของการใช้พลังงานรวมDC; ระบบการจ่ายประกอบด้วยการกระจายอำนาจของ UPS (แหล่งจ่ายไฟสำรอง), ระบายความร้อน, แสง และสวิตช์อาคารบริโภค 48% Greenberg et al. [54] รายงานศูนย์เครือข่ายของข้อมูลใช้ 10-20% ของพลังงานรวม การออกแบบศูนย์ข้อมูลเสมือน "Green" ที่ฝังอัลกอริทึมที่ใช้เป็นบัญชีปริมาณการใช้พลังงานจะช่วยผู้ดูแลระบบในการลดต้นทุน และสอดคล้องกับความกังวลด้านสิ่งแวดล้อมใหม่ โดยเฉพาะ เครือข่ายเสมือนช่วยใน

การลดใช้พลังงานผ่านการลดจำนวนทางกายภาพเราเตอร์/สวิตช์ที่จำเป็นต้องใช้งาน โดยรวมเสมือนทรัพยากรในจำนวนน้อยลงของกายภาพจำนวนมาก อย่างไรก็ตาม แม้ความพยายามล่าสุดในการออกแบบเครือข่ายศูนย์ข้อมูลพลังงานทราบว่า [55], อัลกอริธึมการฝังอยู่ไม่มีได้พิจารณาต้นทุนของพลังงาน ความท้าทายหลักในการลดการใช้พลังงานเป็นวิธีการที่จะร่วมกันเพิ่มประสิทธิภาพตำแหน่งของ VMS และ VNS เพื่อการประหยัดพลังงาน

- fault-tolerance เป็นอีกหนึ่งปัญหาสำคัญในศูนย์ข้อมูลเสมือนจริง ความล้มเหลวของการเชื่อมโยงทางกายภาพสามารถก่อให้เกิดการหยุดชะงัก VDCs หลายตัวที่ใช้ link เพื่อที่จะลดประสิทธิภาพของโปรแกรมเนื่องจากความล้มเหลว เป็นสิ่งจำเป็นสำหรับผู้เช่าที่จะหา embeddings ที่ทนความผิดพลาดของงานที่มีอยู่บนเครือข่ายฝังบอลลูนเสมือนจริง [56] หมายถึงขั้นตอนที่เริ่มต้นไปในทิศทางนี้

- ในที่สุดปัญหาของการฝัง VDC ยังทำให้เกิดคำถามในการหาข้อมูลที่เหมาะศูนย์กายภาพสำหรับการฝัง topologies VDC

- ในที่สุดผู้เช่าบางคนอาจต้องการที่จะปรับใช้ VDCs ข้ามศูนย์ข้อมูลจากภูมิภาคหลายนี้ ทำให้เกิดปัญหาของการฝัง VDCs ข้ามโดเมนบริหารในหลายกรอบที่พบฝังที่มีประสิทธิภาพโดยไม่ต้องเสียสละความเป็นเอกเทศของแต่ละผู้ให้บริการโครงสร้างพื้นฐานจะกลายเป็นปัญหาที่ท้าทาย มีผลงานล่าสุดเช่น Polyvine [57] หมายถึงความพยายามที่เริ่มต้นสำหรับการแก้ปัญหานี้

- แม้ว่าสถาปัตยกรรมศูนย์ข้อมูลที่นำเสนอได้อาศัยอยู่บนเครือข่ายที่แตกต่างกันเช่น topologies, Fat-Tree และ Clos, มันจะไม่ชัดเจนสำหรับโทโพโลยีที่เหมาะที่สุดสำหรับการฝัง VDC ยกตัวอย่างเช่นมันได้รับรายงานว่าโครงการประสบความสำเร็จในการฝัง SecondNet เซิร์ฟเวอร์สูงและการใช้ประโยชน์เครือข่ายสำหรับ BCube [15] มากกว่าการใช้ประโยชน์สำหรับโทโพโลยีแบบ fat-tree ดังนั้นเราเชื่อว่ามันเป็นสิ่งที่สำคัญในการวิเคราะห์ผลกระทบของการฝัง VDC เกี่ยวกับการออกแบบของเครือข่ายข้อมูลทางกายภาพที่ศูนย์ topologies

*C. Programmability*

โปรแกรมเครือข่ายจะมีแรงจูงใจมาจากความปรารถนาที่จะเพิ่มความยืดหยุ่นและนวัตกรรมจากการจำแนกการทำงานของเครือข่ายเพื่ออำนวยความสะดวกการ โดยการแนะนำของปรโตคอลใหม่และสถาปัตยกรรม โดยระบุเพียงโปรแกรมเครือข่ายสามารถกำหนดเป็นความสามารถในการเรียกใช้รหัสของบุคคลที่สามบนอุปกรณ์เครือข่าย (เช่น เราเตอร์) ทั้งในการควบคุมระนาบ และระบายข้อมูล โปรแกรมเครือข่ายที่ได้รับเมื่อเร็วๆนี้ ให้ความสนใจในการต่ออายุในการวิจัยร่วมกัน โดยเฉพาะอย่างยิ่งแนวคิดของเครือข่ายซอฟแวร์ที่กำหนดไว้ (SDN) มีวัตถุประสงค์เพื่อให้ง่ายใน API สำหรับการเขียนโปรแกรมเครือข่ายการควบคุมระนาบ ประโยชน์ของโปรแกรมเครือข่ายจากเทคนิค virtualization ไม่คำนึงถึงบริบทการพิจารณา (เช่น เครือข่าย ISP หรือ DC) ตัวอย่างเช่น การเรียกรหัสที่กำหนดเองเมื่อโหนดเสมือนไม่เพียงแต่มีผลกระทบต่อโหนดเสมือนอื่นๆของเครือข่าย (Isolation) แต่ยังไม่ก่อให้เกิดการหยุดชะงักในพื้นผิวทางกายภาพซึ่งเป็นข้อกังวลที่สำคัญ สำหรับการนำโปรแกรมเครือข่าย ในบริบทของศูนย์ข้อมูลเสมือนจริง, โปรแกรมเครือข่ายให้เป็นอินเตอร์เฟซ แบบแยกส่วนสำหรับการแยกทางกายภาพของ topologies จาก virtual topologies ช่วยให้จะได้รับการจัดการและพัฒนา ในขณะที่เราได้เห็นแล้วข้อเสนอของสถาปัตยกรรมจำนวนมากที่สำรวจใน Paperนี้จะอาศัยเทคโนโลยีการเขียนโปรแกรมเครือข่ายเช่น Openflow อย่างไรก็ตามในบริบทของศูนย์ข้อมูลเสมือนจริง multitenant, โปรแกรมเครือข่ายจะขึ้นอยู่กับจำนวนของความท้าทายการวิจัย:

- ปัจจุบันศูนย์ข้อมูลข้อเสนอสถาปัตยกรรมเครือข่ายเพียงแต่ช่วยให้การควบคุม 2 ชั้น และ 3 ชั้นโปรโตคอลนี้ มีเอกสารที่ออกแบบการใช้งานของชั้นเดียวกัน 2 ชั้นและ 3 ชั้นโปรโตคอล (เช่น IPv4 และ Ethernet) โดยผู้เช่าทั้งหมดให้ APIs ของโปรแกรมสำหรับ virtualization ในที่ชั้นที่แตกต่างกันของเครือข่าย stack ซึ่งจะเพิ่มความยืดหยุ่นอย่างมีนัยสำคัญกับข้อมูลเครือข่ายศูนย์

- ในขณะที่เทคโนโลยีเครือข่ายโปรแกรมมีการจัดการที่มีความยืดหยุ่นของการให้เช่า และผู้ให้บริการโครงสร้างพื้นฐานที่พวกเขายังเปิดโอกาส สำหรับผู้เช่าที่เป็นอันตรายต่อโครงสร้างพื้นฐานทางที่ผิด การให้บริการโครงสร้างพื้นฐานจำเป็นต้องกำหนดวิธีเพื่อให้เข้าถึงและวิธีการควบคุมมากขึ้นที่จะมอบหมายให้ผู้เช่าเพื่อให้ผู้เช่าได้รับระดับที่น่าพอใจของความยืดหยุ่นในแง่ของการเขียนโปรแกรมของอุปกรณ์เครือข่ายที่มั่นใจในความปลอดภัยและการรักษาความปลอดภัยในการอยู่ร่วมกันของผู้เช่าหลาย

ผู้ขายอาจมีเครือข่ายที่ไม่ได้มาตรฐาน API สำหรับการเขียนโปรแกรมที่เป็นกรรมสิทธิ์ ท้าทายการวิจัยที่น่าสนใจคือการ ทำความเข้าใจผลกระทบของการสร้างระบบเครือข่ายศูนย์ข้อมูลจาก

อุปกรณ์ที่ต่างชนิดกันกับฮาร์ดแวร์ที่แตกต่างกันระดับ APIs แนะนำ heterogeneity มีทั้งข้อดีและข้อเสีย ข้อเสียที่สำคัญคือ ค่าใช้จ่ายการบริหารนำโดยอินเตอร์เฟซที่แตกต่างกัน ในขณะที่ข้อดีก็คือ มีบางคุณสมบัติที่เป็น vendor-specific อาจเป็นที่ต้องการในบางสถานการณ์

ปัจจุบัน OpenFlow [58] และชั้น virtualization ของ FlowVisor [59] เป็นเทคโนโลยีที่ดีที่สุดที่นำเสนออย่างชัดเจนเพื่อให้บรรลุการเขียนโปรแกรมในเครือข่ายข้อมูลศูนย์ OpenFlow เป็น abstraction layer ที่ช่วยให้ผู้ใช้สามารถกำหนดพฤติกรรมของสวิทช์เครือข่ายโดยใช้คอมโพเนนต์พิเศษที่เรียกว่าตัวควบคุม FlowVisor [59] คือ เลเยอร์การจำลองเสมือนเครือข่ายที่ช่วยให้หลายตัวควบคุม (controller หนึ่งเช่าต่อ) ควบคุมสวิตช์ OpenFlow ร่วมกันได้ล่าสุดความพยายามของการวิจัยมีการดำเนินการปรับใช้ OpenFlow ในข้อมูลศูนย์เครือข่าย [39], [60], [61] ข้อจำกัดของ OpenFlow อย่างใดอย่างหนึ่งเป็นการขยายขีดความสามารถ ในปัจจุบัน OpenFlow adopts มีสถาปัตยกรรมแบบศูนย์รวมที่ตัวควบคุมเดียวรับผิดชอบในการจัดการทั้งหมดสวิตช์ที่ได้เปิดใช้งาน OpenFlow ในเครือข่ายตั้งแต่เครือข่ายศูนย์ข้อมูลขนาดใหญ่ โดยทั่วไปทำหน้าที่ล้านกรแสพร้อมสลับ OpenFlow ซึ่งอาจเป็นปัญหาคอขวดของประสิทธิภาพการทำงานได้ โดยเฉพาะอย่างยิ่ง DevoFlow [61] ที่ควบคุมเฉพาะมากกว่าส่วนย่อยของกระแส และ [39] HyperFlow ใช้กระจายตัวควบคุมที่มีมุมมองเชิงตรรกะแบบครบวงจร ในทำนองเดียวกัน scalability ของ FlowVisor ก็ยังเป็นเรื่องที่ต้องตรวจสอบต่อไปให้จำนวนมากของผู้เช่าที่เกี่ยวข้องกับการอยู่ในศูนย์ข้อมูลเสมือนจริง ทางที่เป็นไปได้สำหรับการปรับปรุง scalability FlowVisor คือ การกำหนดจำนวนที่เหมาะสมและตำแหน่งของอินสแตนซ์ FlowVisor ในโปรแกรมเครือข่ายศูนย์ข้อมูล โดยในที่สุดแพลตฟอร์มโปรแกรมอื่น ๆ (เช่น เครือข่ายที่ใช้งานตัวแทนมือถือ และสคริป MIB) ก็อาจได้รับการประเมินในบริบทของเครือข่ายข้อมูลเสมือนจริง

*D. Network performance guarantees*

ข้อมูลการค้าในทุกวันนี้จะกลับไปเป็นจำนวนมากมายของโปรแกรมประยุกต์ที่มีความต้องการด้านประสิทธิภาพที่หลากหลาย ตัวอย่างเช่น ผู้ใช้หันมาใช้งานเช่นเว็บเซิร์ฟเวอร์ และเวลาจริงโปรแกรม (เช่น เล่นเกม), มักจะต้องแฝงการสื่อสารต่ำในขณะใช้งานข้อมูลจำนวนมาก เช่นงาน MapReduce ปกติปรารถนา throughput เครือข่ายสูง ในบริบทนี้ เป็นปัญหาที่ท้าทายการออกแบบเครือข่ายศูนย์ข้อมูลที่ยืดหยุ่น ยังยืดหยุ่นสำหรับการสนับสนุนวัตถุประสงค์ประสิทธิภาพเครือข่ายที่หลากหลาย ข้อมูล virtualization ศูนย์เครือข่ายที่มีความสามารถในการเอาชนะความท้าทายเหล่านี้โดยการหารเครือข่ายศูนย์ข้อมูลเข้าสู่เครือข่ายตรรกะหลายตัวที่สามารถจัดเตรียมได้อย่างอิสระเพื่อให้บรรลุวัตถุประสงค์ที่ต้องการประสิทธิภาพการทำงาน ยกตัวอย่างเช่น สถาปัตยกรรมที่นำเสนอเป็นจำนวนมาก เช่นSecondNet และ Oktopus กลไกที่ได้นำเสนอในการจัดสรรแบนด์วิดธ์รับประกันให้กับแต่ละศูนย์ข้อมูลเสมือน

แต่การให้บริการรับประกันแบนด์วิธที่เข้มงวดจะนำไปสู่การใช้ประโยชน์ที่ต่ำถ้าผู้เช่าไม่ใช้ประโยชน์อย่างเต็มที่แบนด์วิดธ์การจัดสรรอื่นๆ, จะมีน้ำหนักที่ยุติธรรมซึ่งสามารถร่วมกันแก้ปัญหา เช่น Seawall และมีความสามารถในการใช้ทรัพยากรสูง แต่พวกเขาไม่ได้ให้การรับประกันทรัพยากรแต่ละศูนย์ข้อมูลเสมือน มีความขัดแย้งระหว่างการเพิ่มการใช้เครือข่าย และการให้การรับประกันประสิทธิภาพของเครือข่ายเป็น การออกแบบโครงการจัดสรรแบนด์วิดธ์ที่ดีต้องออกแบบระหว่างทั้งสองวัตถุประสงค์ที่สำคัญในปัญหาการวิจัย คือ แบบข้อมูลสภาพเสมือนจริง

อย่างไร ทำงานที่มีอยู่บนเครือข่ายแบบเสมือนการศูนย์ข้อมูลได้ถูกเป็นหลักเน้นจัดสรรแบนด์วิดท์สำหรับขบวนการคาดการณ์ throughputs ปัญหาความล่าช้าในการให้การรับรองยังคงเป็นปัญหาเปิดเป็นมันไม่เพียง แต่ต้องจัดสรรแบนด์วิดแยก แต่ยังมีกลไกการควบคุมอัตราการที่มีประสิทธิภาพ ตัวอย่างเช่น S3 [62] เป็นกลไกควบคุมกระแสที่มุ่งหวังที่จะตามกระแสกำหนดเวลาหนึ่งในความท้าทายโดยเฉพาะอย่างยิ่งในสภาพแวดล้อมของศูนย์ข้อมูลคือ TCP incast ยุบปัญหา [63], ที่มาพร้อมกันของแพคเก็ตจากทุนหมุนเวียนระยะสั้นจำนวนมากสามารถเกินขนาดบัฟเฟอร์ในเครือข่ายสวิตช์ เป็นผลในการหน่วงเวลาของเครือข่ายสำคัญเพิ่มขึ้น เราเชื่อว่า การแก้ปัญหาใด ๆ ที่มีการหน่วงเวลา guarantees ในเครือข่ายศูนย์ข้อมูลต้องมีความสามารถของการจัดการ TCP incast ยุบ เราเชื่อว่าปัญหาให้การหน่วงเวลารับประกันในสภาพแวดล้อม multi-tenant ได้ยังต้องสอบสวนเพิ่มเติม

*E. Data center management*

ในระบบศูนย์ข้อมูล virtualized ผู้ให้บริการโครงสร้างพื้นฐานที่มีหน้าที่ในการจัดการทรัพยากรทางกายภาพของศูนย์ข้อมูลในขณะที่ผู้ให้บริการจัดการเสมือนทรัพยากร (เช่นคอมพิวเตอร์ เก็บข้อมูล เครือ ข่าย I/O) ปันส่วนไปยังศูนย์ข้อมูลเสมือนของตน ข้อดีที่สำคัญของศูนย์ข้อมูลที่ virtualized คือ ทรัพยากรทางกายภาพจะถูกจัดการ โดยผู้ให้บริการโครงสร้างพื้นฐานเดียว นี้จะช่วยให้ผู้ให้บริการ

โครงสร้างพื้นฐานที่จะมีมุมมองแบบเต็มของระบบจึงอำนวยความสะดวกในการจัดสรรทรัพยากรที่มีประสิทธิภาพและการจัดการของความล้มเหลว อย่างไรก็ตามยังคงมีความท้าทายหลายอย่างที่ต้อง addressed ในศูนย์ข้อมูลเสมือนจริง ได้แก่ :

- การตรวจสอบเป็นงานที่ท้าทายเนื่องจากจำนวนขนาดใหญ่ของทรัพยากรในศูนย์ข้อมูลการผลิต วิธีตรวจสอบจากส่วนกลางได้รับจากการปรับขยายและความยืดหยุ่นที่ต่ำ การตรวจสอบสหกรณ์ [64] และ gossipping [65] มีจุดมุ่งหมายที่จะเอาชนะข้อจำกัด เหล่านี้โดยการเปิดใช้โซลูชั่นการตรวจสอบการกระจายและมีประสิทธิภาพสำหรับสภาพแวดล้อมขนาดใหญ่ ปัญหาสำคัญคือการลดค่าลบผลกระทบของการจัดการจราจรบนประสิทธิภาพของเครือข่าย ในเวลาเดียวกัน, การหาทางออกที่ปรับขนาดได้สำหรับการรวมข้อมูลการตรวจสอบที่เกี่ยวข้องโดยไม่ทำลายความแม่นยำเป็นความท้าทายที่จะต้องจัดการโดยการตรวจสอบเครื่องมือที่ออกแบบมาสำหรับศูนย์ข้อมูล สุดท้ายให้มุมมองที่กำหนดเองและแยกสำหรับผู้ให้บริการของแต่ละบุคคลและการกำหนดร่วมกันระหว่างระบบการตรวจสอบของผู้ให้บริการโครงสร้างพื้นฐานและผู้ให้บริการยังต้องสำรวจต่อไป
- การจัดการพลังงานอย่างมีประสิทธิภาพเป็นสิ่งสำคัญสำหรับการลดต้นทุนการดำเนินงานของศูนย์ข้อมูล หนึ่งในความท้าทายหลักที่มีต่อการใช้พลังงานที่ดีที่สุดคือการออกแบบสัดส่วนพลังงานสถาปัตยกรรมศูนย์ข้อมูลการใช้พลังงานที่จะถูกกำหนดโดยการใช้เซิร์ฟเวอร์และเครือข่าย [55], [66] ElasticTree [55], ตัวอย่างความพยายามเพื่อให้ได้พลังงานส่วน โดยการจ่ายไฟให้กับปิดสวิตช์และการเชื่อมโยงแบบไดนามิก ในแง่นี้ข้อมูล virtualization ศูนย์เครือข่ายยังสามารถนำไปสู่การลดการใช้พลังงานแม้ว่าการรวมเครือข่าย (เช่นผ่านการย้ายเครือข่ายเสมือนจริง [67]) อย่างไรก็ตามการลดปริมาณการใช้พลังงานสามารถมาในราคาลดประสิทธิภาพ VDC ดังนั้น จึง ออกแบบสัดส่วนพลังงานข้อมูลศูนย์สถาปัตยกรรม factoring ในเครือข่ายเสมือนและหาสมดุลที่ดีระหว่างการใช้พลังงานและประสิทธิภาพการทำงาน VDC คำถามการวิจัยที่น่าสนใจ
- การตรวจสอบและการจัดการของความล้มเหลวเป็นความต้องการขั้นพื้นฐานใดๆ ของสถาปัตยกรรมศูนย์ข้อมูลเพราะความล้มเหลวของทรัพยากรทางกายภาพที่อาจจะมีผลต่อหลายผู้ให้บริการสถาปัตยกรรมที่มีอยู่ส่วนใหญ่พึ่งล้มเหลว reactive handing วิธี ข้อเสียเปรียบหลักของพวกเขาคือการตอบสนองเวลานานอาจซึ่งสามารถส่งผลกระทบในเชิงลบประสิทธิภาพของโปรแกรม จะเป็นการดีที่การจัดการข้อบกพร่องควรใช้ในลักษณะเชิงรุก ซึ่งระบบการทำนายการเกิดของความล้มเหลว และกระทำก่อนที่จะเกิดขึ้น ในทางปฏิบัติการจัดการความผิดเชิงรุกจะมักจะมั่นใจโดยวิธีการของความซ้ำซ้อนเช่นการจัดเตรียมเส้นทางสำรอง และให้ความน่าเชื่อถือสูงโดยไม่ต้องเสียค่าใช้จ่ายมากเกินไปเป็นปัญหาที่น่าสนใจสำหรับการสำรวจในอนาคต

*F. Security*

การรักษาความปลอดภัยได้รับประเด็นที่สำคัญของสถาปัตยกรรมเครือข่ายใดๆ ปัญหาคือที่มาในบริบทของศูนย์ข้อมูลเสมือนจริงเนื่องจากปฏิสัมพันธ์ที่ซับซ้อนระหว่างผู้เช่าและผู้ให้บริการโครงสร้างพื้นฐานและในหมู่ผู้เช่าเอง แม้ว่า virtualization ของเซิร์ฟเวอร์และเครือข่ายศูนย์ข้อมูลสามารถรักษาความปลอดภัยที่มีความท้าทาย เช่น การจำกัดการรั่วไหลของข้อมูลการดำรงอยู่ของช่องด้านข้างและการโจมตีซึ่งรบกวนประสิทธิภาพของเทคโนโลยี virtualization ในทุกวันนี้ซึ่งยังคงห่างไกลที่จากการเติบโต โดยเฉพาะอย่างยิ่งช่องโหว่ต่างๆในเทคโนโลยี virtualization เซิร์ฟเวอร์เช่น VMware [68], Xen [69] และ PC Microsoft เสมือนและเซิร์ฟเวอร์เสมือน [70] ได้รับการเปิดเผย ซึ่ง ช่องโหว่ที่คล้ายกันมีแนวโน้มที่จะเกิดขึ้นในองค์ประกอบของเครือข่ายโปรแกรมเช่นกัน ดังนั้นไม่เพียงแต่ทำเทคนิค virtualization เครือข่ายการป้องกันไม่ให้การรับประกันจากการโจมตีและภัยคุกคามที่มีอยู่กับเครือข่ายทางกายภาพและเสมือน แต่ยังนำไปสู่การช่องโหว่ด้านความปลอดภัยใหม่ ตัวอย่างเช่น โจมตี VM อาจนำไปสู่การโจมตี hypervisor VM โฮสต์เซิร์ฟเวอร์กายภาพ Vm อื่น ๆ ตามมาโจมตีโฮสต์ บนเซิร์ฟเวอร์นั้นและในที่ สุด เครือข่ายเสมือนทั้งหมดที่ใช้ร่วมกันที่เซิร์ฟเวอร์ [71]

นอกจากช่องโหว่ความปลอดภัยที่เกี่ยวข้องกับการบรรเทาเทคโนโลยี virtualization มีความจำเป็นที่จะให้โครงสร้างพื้นฐานการตรวจสอบและการตรวจสอบคือเพื่อตรวจสอบกิจกรรมที่เป็นอันตรายจากผู้เช่าและผู้ให้บริการโครงสร้างพื้นฐานมันจะเรียกว่า ศูนย์ข้อมูลเครือข่ายและพบลักษณะที่แตกต่างกว่าการจราจรในเครือข่ายของข้อมูลดั้งเดิม [72] ดังนั้นกลไกที่เหมาะสมอาจจะต้องตรวจสอบความผิดปกติของเครือข่ายในเครือข่ายข้อมูลเสมือนจริง auditability ในศูนย์ข้อมูลเสมือนจริงควรจะร่วมกันระหว่างผู้เช่าและผู้ให้บริการโครงสร้างพื้นฐานในการป้องกันพฤติกรรมที่เป็นอันตรายจากทั้งสองฝ่าย อย่างไรก็ตามมักจะมีค่าใช้จ่ายที่เกี่ยวข้องกับการโครงสร้าง

พื้นฐานดังกล่าวโดยเฉพาะอย่างยิ่งในศูนย์ข้อมูลขนาดใหญ่ ใน [73] ผู้เขียนแสดงให้เห็นว่ามันเป็นความท้าทายในการตรวจสอบบริการเว็บในสภาพแวดล้อมคอมพิวเตอร์เมฆโดยไม่ต้องทวีความรุนแรงขึ้นประสิทธิภาพของโปรแกรม เราคาดว่าปัญหาที่จะเลวร้ายลงไปอีกเมื่อการขยายกิจกรรมของเครือข่ายใน VDC ยังคงมีงานที่จะต้องทำในการออกแบบกลไกที่ปรับขนาดได้และมีประสิทธิภาพสำหรับการตรวจสอบและการตรวจสอบศูนย์ข้อมูลเสมือนจริง

สุดท้าย ในระบบศูนย์ข้อมูล multi-tenant ผู้เช่าอื่นอาจประสงค์ความปลอดภัยในระดับต่างๆ แนะนำเพิ่มเติมความซับซ้อนของการจัดการนโยบายและกลไกการรักษาความปลอดภัยที่แตกต่างกัน นอกจากนี้การอยู่ร่วมกันและการมีปฏิสัมพันธ์ของระบบรักษาความปลอดภัยหลายคาดว่าในศูนย์ข้อมูล multitenant เป็นปัญหาที่ยังไม่ได้รับการแก้ไข ตัวอย่างเช่นความขัดแย้งระหว่างไฟร์วอลล์และการบุกรุกระบบตรวจจับการนโยบายของผู้ให้บริการโครงสร้างพื้นฐานและผู้ให้บริการจะต้องมีการตรวจพบและแก้ไขได้ [74]

*G. Pricing*

การกำหนดราคา คือปัญหาสำคัญในหลายสภาพแวดล้อมของศูนย์ข้อมูลไม่เพียงแต่เพราะมีผลโดยตรงต่อรายได้ของผู้ให้บริการโครงสร้างพื้นฐาน แต่ยังเพราะมันให้แรงจูงใจสำหรับผู้เช่าที่จะประพฤติในทางที่นำไปสู่ผลลัพธ์ที่ต้องการเช่นการใช้ทรัพยากรสูงสุดและ ประสิทธิภาพของโปรแกรม [1].โดยทั่วไปพูดโครงการราคาการออกแบบที่ดีควรจะเป็นทั้งสองอย่างเป็นธรรมและมีประสิทธิภาพ ความเป็นธรรมหมายความว่าดีเหมือนกันที่ควรจะขายในราคาที่เหมือนกัน ประสิทธิภาพราคาของดีจะนำไปสู่ผลลัพธ์ที่มีประสิทธิภาพ (เช่นการจับคู่อุปทานและอุปสงค์). ทุกวันนี้การให้บริการโครงสร้างพื้นฐานสัญญาว่าจะให้ทรัพยากรที่จะเช่าในลักษณะความต้องการและผู้เช่าเรียกเก็บอัตราแบนวิดธ์สำหรับ VM และใช้งานเครือข่าย. แม้จะมีการกำหนดแผนที่ง่าย แต่ยังคงทุกข์ทรมานจากข้อบกพร่อง จากค่าของ VMS และเครือข่ายสำหรับผู้เช่าเดียวสามารถพึ่งพาอาศัยซึ่งกันและกัน ตัวอย่างเช่นประสิทธิภาพของเครือข่ายยากจนสามารถยืดเวลาการทำงานของงานที่ผู้เช่าทำให้ต้นทุนในการใช้งานเพิ่มขึ้น VM [75]. virtualization ศูนย์ข้อมูลเครือข่ายสามารถแก้ไขปัญหานี้โดยจัดสรรแบนด์วิดธ์เพื่อรับประกันสำหรับแต่ละ VM [16] สำหรับศูนย์ข้อมูลเสมือนกับการเชื่อมต่อที่ดีที่สุดความสามารถ, ข้อเสนอล่าสุดของการกำหนดราคาทรัพยากรที่โดดเด่น (DRP) [75] ดูเหมือนจะเป็นทางออกที่มีแนวโน้มที่จะกำจัดการพึ่งพาระหว่าง VM และใช้งานเครือข่าย

ข้อเสียเปรียบที่สอง ในปัจจุบันข้อมูลการกำหนดราคาโครงการศูนย์ที่พวกเขาไม่ให้สิ่งจูงใจสำหรับผู้เช่าเพื่อให้บรรลุผลตามที่ต้องการ.โดยเฉพาะอย่างยิ่งพวกเขาไม่ได้ (1) กระตุ้นการซื้อของทรัพยากรเมื่อมีความต้องการอยู่ในระดับต่ำและ (2) การปราบปรามความต้องการมากเกินไป (ในขณะที่ให้ความสำคัญกับการใช้งานที่สำคัญ) เมื่อมีความต้องการอยู่ในระดับสูง.วิธีการแก้ปัญหาที่มีแนวโน้มที่จะแก้ไขปัญหานี้คือการใช้รูปแบบการกำหนดราคาในตลาดที่ขับเคลื่อนด้วยทรัพยากรที่ราคาขึ้นและลดลงตามอุปสงค์และอุปทาน ในมุมมองนี้การให้บริการ Amazon EC2 เช่นจุดแสดงถึงความพยายามเชิงพาณิชย์ครั้งแรกอย่างเต็มที่ต่อตลาดที่ขับเคลื่อนด้วยการกำหนดราคาโครงการ. บริการ Amazon EC2 เช่นจุดแสดงถึงความพยายามเชิงพาณิชย์ครั้งแรกอย่างเต็มที่ต่อตลาดที่ขับเคลื่อนด้วยการกำหนดราคาโครงการ.เทคนิคที่คล้ายกันนอกจากนี้ยังสามารถนำมาใช้สำหรับเครือข่ายข้อมูลเสมือนจริงศูนย์ที่ราคาทรัพยากรสำหรับชั้นเรียนบริการที่แตกต่างกัน (รับประกันแบนด์วิธ เช่น besteffort) แตกต่างกันตามความต้องการทรัพยากรอย่างไรก็ ตามการออกแบบการตลาดที่ขับเคลื่อนด้วยโครงการจัดสรรทรัพยากรที่จัดสรรทรัพยากรหลายชนิด (เช่น VM และ bandwidth) ที่มีการค้ำประกันคุณภาพการให้บริการที่แตกต่างกันยังคงเป็นปัญหาที่ท้าทาย

ในขณะที่การอภิปรายเพื่อให้ห่างไกลได้รับการมุ่งเน้นไปที่การกำหนดราคาทรัพยากรภายในศูนย์ข้อมูลกรณีการกำหนดราคาทรัพยากรเมฆนอกศูนย์ข้อมูลยังเป็นความท้าทายอย่างหนึ่ง. โดยเฉพาะอย่างยิ่งเมื่อผู้เช่ามีความประสงค์จะปรับใช้ศูนย์ข้อมูลเสมือนข้ามศูนย์ข้อมูลหลาย ๆ มีความจำเป็นต้องพัฒนากลไกไม่เพียง แต่จะช่วยให้ผู้เช่าตัดสินใจที่เหมาะสมของการฝัง VDCs ผ่านเครือข่ายหลาย แต่ยังเพื่อให้ผู้ให้บริการทั้งผู้เช่าและโครงสร้างพื้นฐานเพื่อการเจรจาต่อรอง คุณภาพการให้บริการและรูปแบบการกำหนดราคา.ที่มีอยู่ทำงานเช่น V-Mart [76] หมายถึงความพยายามครั้งแรกในทิศทางนี้.

## 6.สรุป

ศูนย์ข้อมูลได้กลายเป็นโครงสร้างพื้นฐานที่มีประสิทธิภาพสำหรับการจัดเก็บข้อมูลและพื้นที่การใช้งานเครือข่ายขนาดใหญ่. อย่างไรก็ตามข้อมูลแบบดั้งเดิมของสถาปัตยกรรมเครือข่ายศูนย์ไม่เหมาะสำหรับอนาคตผู้เช่าหลายสภาพศูนย์ข้อมูล. virtualization เป็นเทคโนโลยีที่มีแนวโน้มในการออกแบบศูนย์ข้อมูลปรับขนาดได้

และ deployable ได้อย่างง่ายดายที่มีความยืดหยุ่นตอบสนองความต้องการของการใช้งานของผู้เช่าในขณะที่ลดค่าใช้จ่ายด้านโครงสร้างพื้นฐานการจัดการปรับปรุงความยืดหยุ่นและลดการใช้พลังงาน

ในบทความนี้เราสำรวจสถานะในศูนย์ข้อมูลวิจัยเครือข่ายเสมือนจริง.เราได้พูดถึงแผนการที่นำเสนอจากมุมมองที่แตกต่างกันนักวิจัยเน้นแนวโน้มได้รับดังต่อไปนี้เมื่อมีการออกแบบสถาปัตยกรรมเหล่านี้.นอกจากนี้เรายังมีการระบุบางส่วนของทิศทางการวิจัยที่สำคัญในข้อมูล virtualization ศูนย์เครือข่ายและวิธีการที่มีศักยภาพ. แม้ว่าข้อเสนอในปัจจุบันปรับปรุง scalability ให้กลไกสำหรับสมดุลภาระการค้ำประกันให้แบนด์วิดธ์มีปัญหาที่ท้าทายและที่ยังไม่ได้สำรวจความสำคัญ . การออกแบบเครือข่ายสมาร์ทที่ทันสมัยให้รับประกันประสิทธิภาพที่เข้มงวดการณ์แบบจำลองทางธุรกิจและการกำหนดราคาที่มีประสิทธิภาพการรักษาความปลอดภัยเพื่อให้มั่นใจและมีโปรแกรมที่สนับสนุนข้อมูลโครงสร้างพื้นฐานหลายฉัตรและหลายอภิมหาศูนย์การดำเนินการตั้งสำรองฯ ที่มีความยืดหยุ่นและการเชื่อมต่อการจัดการระหว่างผู้เช่าและผู้ให้บริการและการพัฒนาเครื่องมือที่มีประสิทธิภาพ การจัดการข้อมูลที่ virtualized ศูนย์จะแนะนำที่สำคัญสำหรับการวิจัยในอนาคต

**เอกสารอ้างอิง**


[1] Amazon Elastic Compute Cloud (Amazon EC2). http://aws.amazon.com/ec2/.

[2] D. Carr, “How Google Works,” July 2006.

[3] J. Dean and S. Ghemawat, “MapReduce: Simplified Data Processing on Large Clusters,” in Proc. USENIX OSDI, December 2004.

[4] WMware. http://www.vmware.com.

[5] Xen. http://xen.org.

[6] A. Shieh, S. Kandulaz, A. Greenberg, C. Kim, and B. Saha, “Sharing the Data Center Network,” in Proc. USENIX NSDI, March 2011.

[7] T. Benson, A. Akella, A. Shaikh, and S. Sahu, “CloudNaaS: A Cloud Networking Platform for Enterprise Applications,” in Proc. ACMSOCC, June 2011.

[8] M. Chowdhury and R. Boutaba, “A Survey of Network Virtualization,” Computer Networks, vol. 54, no. 5, pp. 862–876, 2010.

[9] “Data Center: Load Balancing Data Center Services SRND,” 2004.

[10] W. Dally and B. Towles, Principles and Practices of Interconnection Networks. Morgan Kaufmann Publishers Inc., 2004.

[11] C. Leiserson, “Fat-Trees: Universal Networks for Hardware-Efficient Supercomputing,” IEEE Trans. Comput., vol. 34, no. 10, pp. 892–901, 1985.

[12] M. Al-Fares, A. Loukissas, and A. Vahdat, “A Scalable, Commodity Data Center Network Architecture,” in Proc. ACM SIGCOMM, August 2008.

[13] C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, Y. Shi, C. Tian, Y. Zhang, and S. Lu, “BCube: A High Performance, Server-centric Network Architecture for Modular Data Centers,” in Proc. ACM SIGCOMM,August 2009.

[14] L. Popa, S. Ratnasamy, G. Iannaccone, A. Krishnamurthy, and I. Stoica, “A Cost Comparison of Datacenter Network Architectures,” in Proc. ACM CoNext, November 2010.

[15] C. Guo, G. Lu, H. Wang, S. Yang, C. Kong, P. Sun, W. Wu, and Y. Zhang, “SecondNet: A Data Center Network Virtualization Architec-ture with Bandwidth Guarantees,” in Proc. ACM CoNEXT, December 2010.

[16] H. Ballani, P. Costa, T. Karagiannis, and A. Rowstron, “Towards Predictable Datacenter Networks,” in Proc. ACM SIGCOMM, August 2011.

[17] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson, “Measuring ISP Topologies with Rocketfuel,” IEEE/ACM Trans. Netw., vol. 12, no. 1, pp. 2–16, 2004.

[18] Q. Zhang, M. F. Zhani, Q. Zhu, S. Zhang, R. Boutaba, and J. Hellerstein, “Dynamic Energy-Aware Capacity Provisioning for Cloud Computing Environments,” in Proc. IEEE/ACM International Conference on Auto-nomic Computing (ICAC), September 2012.

[19] “IEEE Standard for Local and Metropolitan Area Networks – Virtual Bridged Local Area Networks,” IEEE Std 802.1Q-2005, May 2006.

[20] J. Mudigonda, P. Yalagandula, M. Al-Fares, and J. Mogul, “SPAIN:COTS Data-Center Ethernet for Multipathing over Arbitrary Topologies,” in Proc. ACM USENIX NSDI, April 2010.

[21] A. Edwards, F. A, and A. Lain, “Diverter: A New Approach to Networking Within Virtualized Infrastructures,” in Proc. ACM WREN, August 2009.

[22] J. Mudigonda, P. Yalagandula, B. Stiekes, and Y. Pouffary, “NetLord: A Scalable Multi-Tenant Network Architecture for Virtualized Datacen-ters,” in Proc. ACM SIGCOMM, August 2011.

[23] F. Hao, T. Lakshman, S. Mukherjee, and H. Song, “Enhancing Dynamic Cloud-based Services using Network Virtualization,” in Proc. ACM VISA, August 2009.

[24] A. Greenberg, J. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. Maltz, P. Patel, and S. Sengupta, “VL2: A Scalable and Flexible Data Center Network,” in Proc. ACM SIGCOMM, August 2009.

[25] R. Mysore, A. Pamboris, N. Farrington, N. Huang, P. Miri, S. Rad-hakrishnan, V. Subramanya, and A. Vahdat, “PortLand: A Scalable Fault-Tolerant Layer 2 Data Center Network Fabric,” in Proc. ACM SIGCOMM, August 2009.

[26] H. Rodrigues, J. R. Santos, Y. Turner, P. Soares, and D. Guedes, “Gate-keeper: Supporting Bandwidth Guarantees for Multi-tenant Datacenter Networks,” in Proc. WIOV, June 2011.

[27] T. Lam, S. Radhakrishnan, A. Vahdat, and G. Varghese, “NetShare: Virtualizing Data Center Networks across Services,” Technical Report CS2010-0957, May 2010.

[28] F. Hao, T. Lakshman, S. Mukherjee, and H. Song, “Secure Cloud Computing with a Virtualized Network Infrastructure,” in Proc.USENIX HotCloud, June 2010.

[29] C. Hopps, “Analysis of an Equal-Cost Multi-Path Algorithm,” IETF RFC 2992, November 2000.

[30] R. Zhang-Shen and N. McKeown, “Designing a Predictable Internet Backbone Network,” in Proc. ACM HotNets, November 2004.

[31] , “Designing a Predictable Internet Backbone with Valiant Load-Balancing,” in Proc. IWQoS, June 2005.

[32] N. Leavitt, “Is Cloud Computing Really Ready for Prime Time?” Computer, vol. 42, no. 1, pp. 15–20, January 2009.

[34] E. Rosen and Y. Rekhter, “BGP/MPLS IP Virtual Private Networks (VPNs),” IETF RFC 4364, February 2006.

[35] R. Perlman, “An Algorithm for Distributed Computation of a Spanning Tree in an Extended LAN,” ACM Computer Communication Review, vol. 15, no. 4, pp. 44–53, September 1985.

[36] E. Rosen, A. Viswanathan, and R. Callon, “Multiprotocol Label Switch-ing Architecture,” IETF RFC 3031, January 2001.

[37] M. Shreedhar and G. Varghese, “Efficient Fair Queuing Using Deficit Round-Robin,” IEEE/ACM Trans. Netw., vol. 4, no. 3, pp. 375–385, 1996.

[38] “IEEE Std 802.1D-2004, IEEE Standard for Local and Metropolitan Area Networks, Media Access Control (MAC) Bridges.” 2004.

[39] A. Tootoonchian and Y. Ganjalir, “HyperFlow: a Distributed Control Plane for OpenFlow,” in Proc. NSDI INM/WREN, April 2010.

[40] ´I. Goiri, K. Le, J. Guitart, J. Torres, and R. Bianchini, “Intelligent Placement of Datacenters for Internet Services,” in Proc. IEEE ICDCS, June 2011.

[41] B. Ahlgren, P. Aranda, P. Chemouil, S. Oueslati, L. Correia, H. Karl, M. S¨ollner, and A. Welin, “Content, Connectivity, and Cloud: Ingre-dients for the Network of the Future,” IEEE Commun. Mag., vol. 49, no. 7, pp. 62–70, July 2011.

[42] S. Islam and J.-C. Gregoire, “Network Edge Intelligence for the Emerg-ing Next-Generation Internet,” Future Internet, vol. 2, no. 4, pp. 603– 623, December 2010.

[43] k. Church, A. Greenberg, and J. Hamilton, “On Delivering Embarrass-ingly Distributed Cloud Services,” in Proc.ACMHotNets, October 2008.

[44] V. Valancius, N. Laoutaris, C. Diot, P. Rodriguez, and L. Massouli´e, “Greening the Internet with Nano Data Centers,” in Proc. ACM CoNEXT, December 2009.

[45] M. B. Mobley, T. Stuart, and Y. Andrew, “Next-Generation Managed Services: A Window of Opportunity for Service Providers,” CISCO Technical Report, 2009.

[46] D. Oppenheimer, B. Chun, D. Patterson, A. Snoeren, and A. Vahdat, “Service Placement in a Shared Wide-Area Platform,” in Proc. USENIXn ATEC, June 2006.

[47] L. Qiu, V. Padmanabhan, and G. Voelker, “On the Placement of Web Server Replicas,” in Proc. IEEE INFOCOM, April 2001.

[48] M. Yu, Y. Yi, J. Rexford, and M. Chiang, “Rethinking Virtual Network Embedding: Substrate Support for Path Splitting and Migration,” ACM Computer Communication Review, vol. 38, no. 2, pp. 17–29, April 2008.

[49] M. Chowdhury, M. Rahman, and R. Boutaba, “Virtual Network Embed-ding with Coordinated Node and Link Mapping,” in Proc. INFOCOM, April 2009.

[50] M. R. Rahman, I. Aib, and R. Boutaba, “Survivable Virtual Network Embedding,” in Proc. IFIP Networking, May 2010.

[51] N. F. Butt, N. M. M. K. Chowdhury, and R. Boutaba, “Topology-Awareness and Reoptimization Mechanism for Virtual Network Em-bedding,” in Proc. IFIP Networking, May 2010.

[52] Energy Efficiency and Sustainability of Data Centers. http://www.sigmetrics.org/sigmetrics2011/greenmetrics/Carey GreenMetricsKeynote060711.pdf.

[53] Energy Logic: Reducing Data Center Energy Consumption by Creating Savings that Cascade Across Systems.

[54] A. Greenberg, J. Hamilton, D. Maltz, and P. Patel, “The Cost of a Cloud: Research Problems in Data Center Networks,” ACM Computer Communication Review, vol. 39, no. 1, pp. 68–73, 2009.

[55] B. Heller, S. Seetharaman, P. Mahadevan, Y. Yiakoumis, P. Sharma, S. Banerjee, and N. McKeown, “ElasticTree: Saving Energy in Data Center Networks,” in Proc. USENIX NSDI, April 2010.

[56] M. Rahman, I. Aib, and R. Boutaba, “Survivable Virtual Network Embedding,” NETWORKING 2010, pp. 40–52, 2010.

[57] M. Chowdhury and R. Boutaba, “PolyViNE,” in Proc. ACM VISA, August 2010.

[58] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: Enabling Innovation in Campus Networks,” ACM Computer Communication Review, vol. 38, no. 2, pp. 69–74, April 2008.

[59] R. Sherwood, G. Gibb, K.-K. Yap, G. Appenzeller, M. Casado, N. McK-eown, and G. Parulkar, “Can the Production Network Be the Test-bed?” in Proc. USENIX OSDI, October 2010.

[60] A. Tavakoli, M. Casado, T. Koponen, and S. Shenker, “Applying NOX to the Datacenter,” in Proc. ACM HotNets, August 2009.

[61] A. Curtis, J. Mogul, J. Tourrilhes, P. Yalagandula, P. Sharma, and S. Banerjeer, “DevoFlow: Scaling Flow Management for High-Performance Network,” in Proc. ACM SIGCOMM, August 2011.

[62] C. Wilson, H. Ballani, T. Karagiannis, and A. Rowstron, “Better never than Late: Meeting Deadlines in Datacenter Networks,” in Proc. ACM SIGCOMM, August 2011.

[63] H. Wu, Z. Feng, C. Guo, and Y. Zhang, “ICTCP: Incast Congestion Control for TCP,” in Proc. ACM CoNEXT, November 2010.

[64] K. Xu and F. Wang, “Cooperative Monitoring for Internet Data Centers,” in Proc. IEEE IPCCC, December 2008.

[65] F. Wuhib, M. Dam, R. Stadler, and A. Clemm, “Robust Monitoring of Network-wide Aggregates through Gossiping,” IEEE Trans. Network Service Management, vol. 6, no. 2, pp. 95–109, 2009.

[66] H.Yuan,C.C.J.Kuo,andI.Ahmad,"EnergyEfficiency in Data Centers and Cloud-based Multimedia Services: An Overview and Future Directions," in Proc. IGCC, August 2010.

[67] Y. Wang, E. Keller, B. Biskeborn, J. van der Merwe, and J. Rex-ford, "Virtual Routers on the Move: Live Router Migration as a Network-Management Primitive," ACM Computer Communication Re-view, vol. 38, pp. 231–242, August 2008.

[68] VMWare vulnerability. http://securitytracker.com/alerts/2008/Feb/1019493.html.

[69] Xen vulnerability. http://secunia.com/advisories/26986.

[70] Virtual PC vulnerability. http://technet.microsoft.com/en-us/security/bulletin/MS07-049.

[71] J. Szefer, E. Keller, R. Lee, and J. Rexford, "Eliminating the Hypervisor Attack Surface for a More Secure Cloud," in Proc. ACM CSS, October 2011.

[72] T. Benson, A. Anand, A. Akella, and M. Zhang, "Understanding Data Center Traffic Characteristics," ACM SIGCOMM Computer Communi-cation Review, vol. 40, no. 1, pp. 92–99, 2010.

[73] A. Chukavkin and G. Peterson, "Logging in the Age of Web Services," IEEE Security and Privacy, vol. 7, no. 3, pp. 82–85, June 2009.

[74] E. Al-Shaer, H. Hamed, R. Boutaba, and M. Hasan, "Conflict Classifi-cation and Analysis of Distributed Firewall Policies," IEEEJ. Sel. Areas Commun., vol. 23, no. 10, pp. 2069–2084, 2005.

[75] H. Ballani, P. Costa, T. Karagiannis, and A. Rowstron, "The Price Is Right: Towards Location-Independent Costs in Datacenters," 2011.

[76] F.-E. Zaheer, J. Xiao, and R. Boutaba, "Multi-Provider Service Nego-tiation and Contracting in Network Virtualization," in Proc. IEEE/IFIP NOMS, April 2010.