บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ระบบบริหารจัดการการทำงานเครื่องเสมือนจริงด้วยภาษา PHP ซึ่งมีทฤษฎีและแนวคิดที่เกี่ยวข้อง ได้แก่ ตัวอย่างผลิตภัณฑ์ phpVirtualBox โครงการ OpenNebula และความหมายของ Web-Based Application ซึ่งผู้จัดทำระบบบริหารจัดการการทำงานเครื่องเสมือนจริง จะกล่าวถึงทฤษฎีดังกล่าวเพียงสังเขป

phpVirtualBox

เป็น Open Source ที่ถูกพัฒนาด้วย AJAX เพื่อให้ได้ส่วนติดต่อกับผู้ใช้ เช่นเดียวกับโปรแกรม VirtualBox ซึ่งถูกพัฒนาด้วยภาษา PHP สำหรับรูปแบบการเชื่อมต่อใหม่ที่ถูกพัฒนาขึ้นนี้ สามารถใช้ควบคุมการทำงานของ VirtualBox โดยโปรแกรมที่ถูกสร้างขึ้น พัฒนามาจากโครงการ Vboxweb phpVirtualBox ถูกสร้างขึ้นให้ผู้ใช้ทั่วๆไปที่สามารถเข้าถึง VirtualBox ผ่านทางเว็บได้

1. คุณสมบัติ

1.1 การเปิด/ปิด (Virtual Machine : VM)หรือการจัดการ VM

1.1.1 สั่งหยุดการทำงานจอง VM ไว้ชั่วคราว

1.1.2 บันทึกสถานะ การทำงานหรือยกเลิกสถานะ การทำงานของ VM ได้

1.1.3 ปิดเครื่อง VM ต้องการได้

1.2 การจับภาพ(Snapshots)

1.2.1 ถ่ายภาพ ลบ และคืนค่า ภาพที่เราถ่ายด้วยวิธี Snapshots ได้

1.3 ดู VM Log files ได้

1.4 กำหนดค่าต่างๆของ VM ได้(ค่าเริ่มต้นหรือค่าอื่นๆของ VM)

1.4.1 สามารถแสดงค่าที่เลือกได้

1.5 สื่อเป็นส่วนที่ช่วยจัดการเครื่องเสมือนที่เพิ่มเข้ามาใน VM

1.5.1 สามารถบันทึกข้อมูลต่างๆใน VM ได้

1.5.2 สามารถจัดการ คุณลักษณะ(featured) ใน VM ได้

1.5.3 สามารถกำหนดหรือสร้าง Harddisk ได้

1.5.4 สามารถ อ่าน (DVD/CD and Floppy images)ในขณะที่กำลังทำงานอยู่ได้

1.6 มีตัวช่วยสร้างเครื่องเสมือน

1.7 กำหนดค่าเครือข่ายทั่งโลก

1.7.1 เพิ่ม/ลบ/ตั้งค่าการเชื่อมต่อโฮสต์เท่านั้น

1.7.2 สามารถ ตั้งค่า DHCP สำหรับการเชื่อมต่อโฮสต์เท่านั้น

1.8 นำเข้า/ส่งออกเครื่องเสมือนจริง

1.9 เข้าถึงคอลโซลเครื่องเสมือนจริง

1.9.1 เพิ่มคุณสมบัติ username / password และรายละเอียดของหน้าจอ

1.9.2 สามารถเข้าไปดู VM เพื่อเปิดคอลโซล ที่น้อยที่สุด ในหน้าต่างเบราเซอร์ใหม่

1.10 สนับสนุนได้หลายภาษา (ต้องเป็น VirtualBox เวอร์ชั่น 3.2.8 ขึ้นไป)

2. ความต้องการของระบบ

2.1 PHP เว็บเซิร์ฟเวอร์ เวอร์ชั่น 5.1.0 ขึ้นไป

2.2 VirtualBox เวอร์ชั่น 3.2.x ขึ้นไป

2.3 เรียกใช้ผ่านเว็บเบราเซอร์ ต่อไปนี้

2.3.1 Firefox เวอร์ชั่น 3.6.0 ขึ้นไป

2.3.2 Internet Explorer เวอร์ชั่น 8 ขึ้นไป

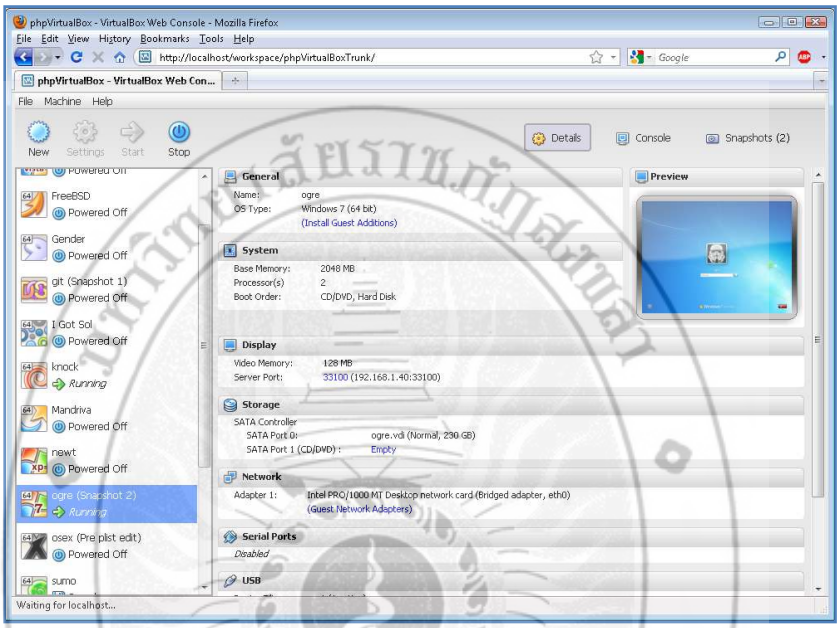
2.3.3 Opera เวอร์ชั่น 10.0 ขึ้นไป

2.3.4 Safari เวอร์ชั่น 4.0 ขึ้นไป

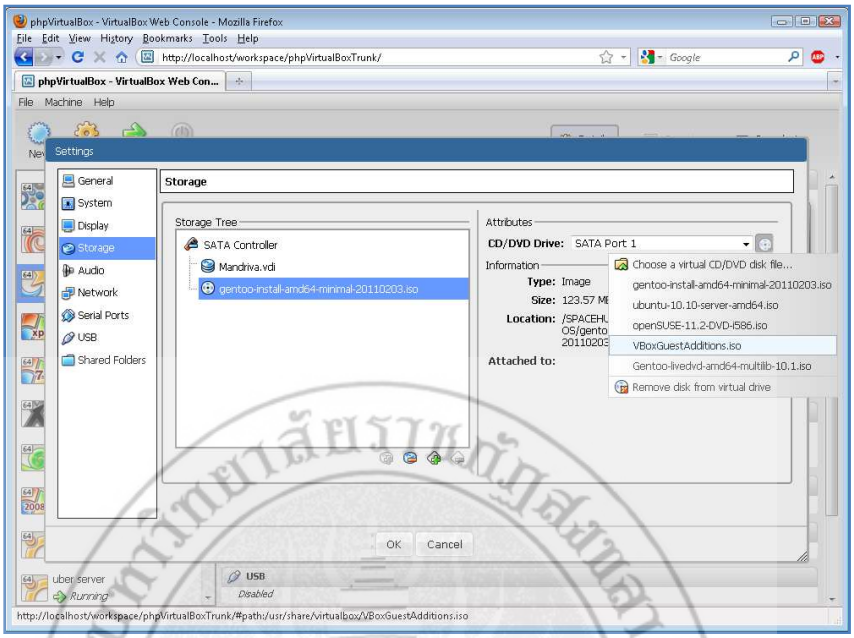
2.3.5 Chrome เวอร์ชั่น 5.0 ขึ้นไป

2.3.6 เว็บเบราเซอร์อื่นๆ

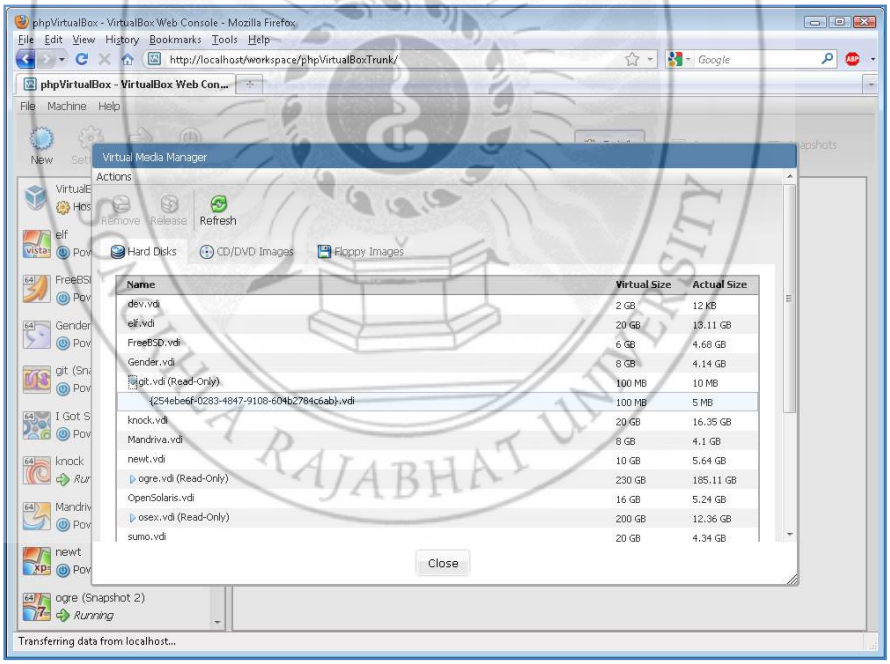
3. ภาพหน้าจอ



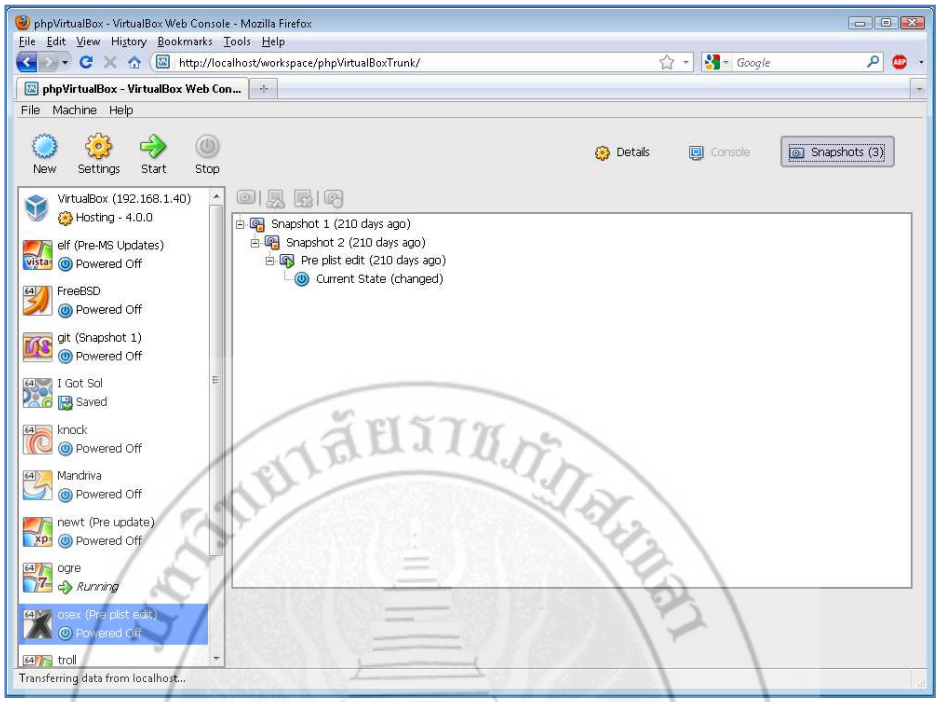
ภาพที่ 2.1 หน้าอินเตอร์เฟซหลัก



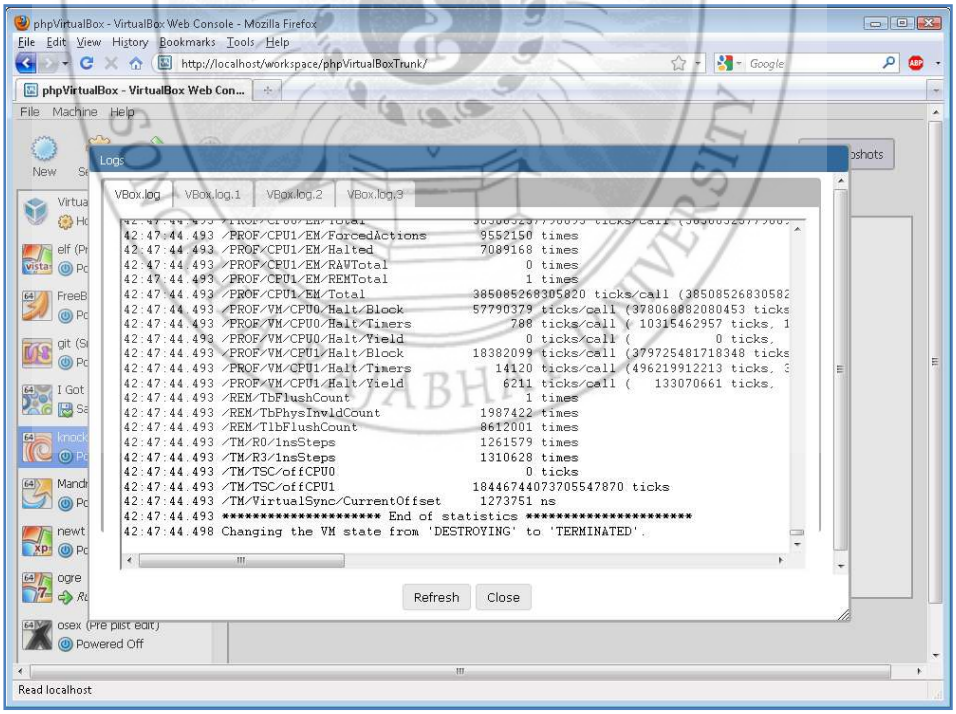
ภาพที่ 2.2 การตั้งค่า VM



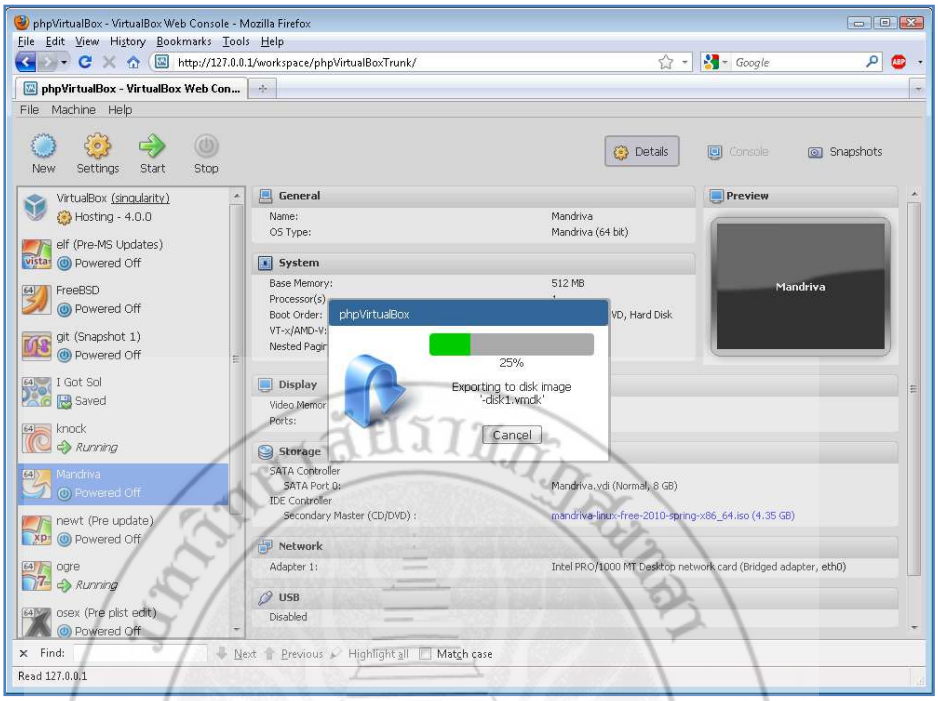
ภาพที่ 2.3 Virtual Media Manager



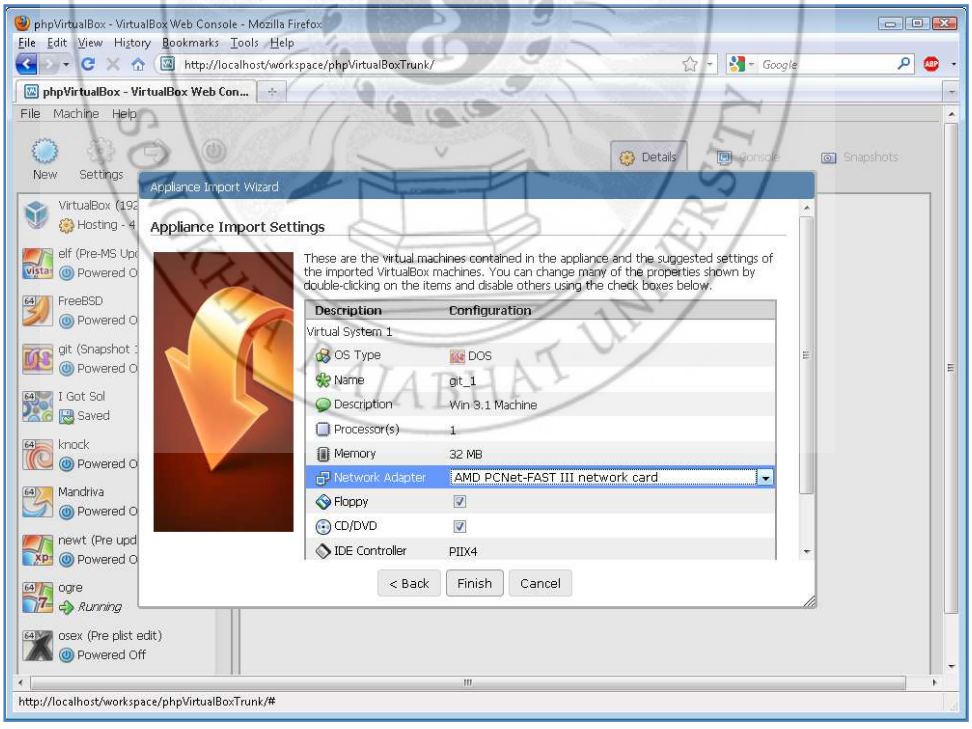
ภาพที่ 2.4 การรูป Snapshots



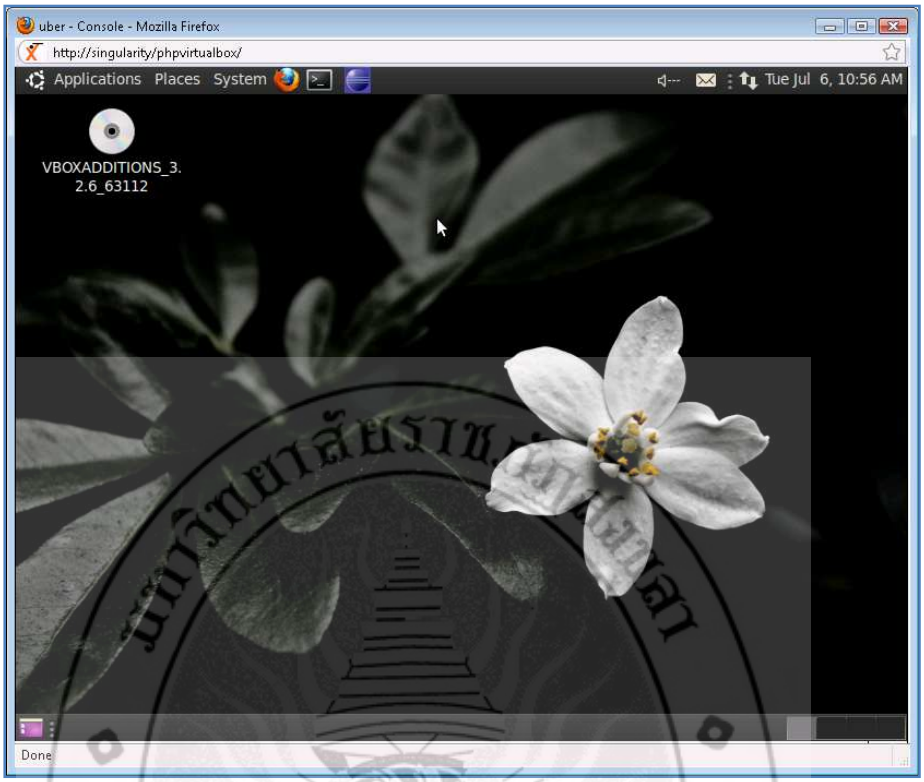
ภาพที่ 2.5 VM Logs



ภาพที่ 2.6 ความคืบหน้าในการดำเนินงาน



ภาพที่ 2.7 การนำเข้า VM



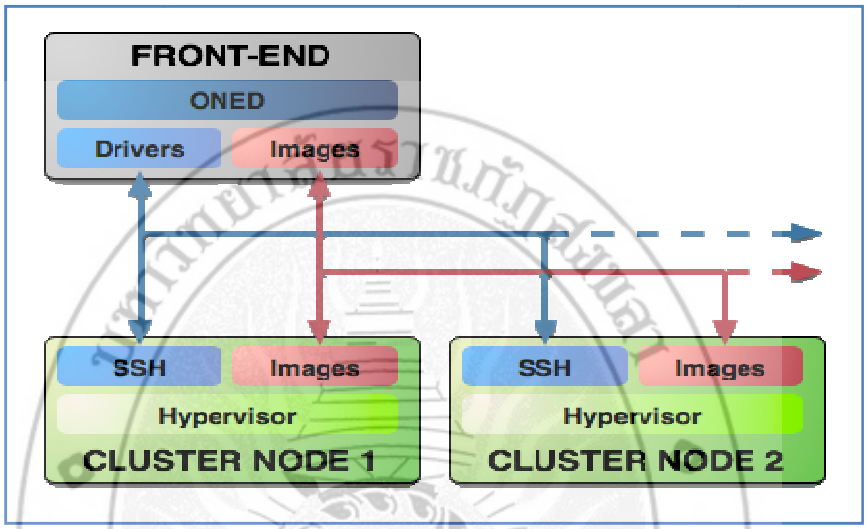
ภาพที่ 2.8 เปิดคอลโซลในหน้าต่างเบราเซอร์ได้

โครงการ OpenNebula

OpenNebula เป็นโครงการโอเพนซอร์สมีจุดประสงค์เพื่อสร้างเครื่องมือบริหารจัดการ Cloud Computing ที่มีความยืดหยุ่นในการขยายขนาด Data Center infrastructure ได้ง่าย ซึ่งมีการพัฒนามาจนถึงรุ่นที่ 2.0.1 แล้ว หลังจากโครงการนี้เดินมาในรูปแบบโอเพนซอร์สได้ระยะหนึ่งซึ่งผลงานของชุมชน OpenNebula.org และ C12G Lab ที่ช่วยกันพัฒนา เครื่องมือในการบริหารจัดการ Cloud Infra แบบ Open Source ก็ได้รับการตอบรับอย่างดี ไม่ว่าจะเป็น CERN, Chaina Mobile, ESAC, Nikhef, SARA และยังมีผู้พัฒนาต่อยอดจากโครงการนี้อีกมากเช่น StratusLab, Bon Fire, Reservoir, OCCI, D-Grid, 4 CaaSt เป็นต้น

OpenNebula ออกแบบมาเพื่อรองรับการใช้งานบน Infrastructure เดิมให้มากที่สุดซึ่งการออกแบบเน้นการเพิ่มหรือขยาย Infra ได้ยืดหยุ่น ซึ่งคล้ายคลึงกับการวางโครงสร้างในระบบคลัสเตอร์ทั่วไป จากความยืดหยุ่นคุณสมารถประยุกต์ OpenNebula เพื่อใช้จัดการ Cloud Infrastructure ในรูปแบบต่างๆ ได้ ไม่ว่าจะเป็น Private Cloud, Hybridge Cloud และ Public Cloud ในบทความนี้จะครอบคลุมเพียง Private Cloud เท่านั้น

การติดตั้ง OpenNebula เพื่อสร้าง Private Cloud โครงสร้างการเชื่อมต่อจะใช้โครงสร้างเดิมเหมือนกับการเชื่อมต่อแบบคลัสเตอร์ทั่วไป ซึ่งมีเครื่อง front-end และเครื่อง node ที่จะให้ VM ไปทำงานบนนั้น ซึ่งเครื่อง node มีการเชื่อมต่อกับ front-end อย่างน้อย 1 เส้นดังภาพ



ภาพที่ 2.9 เครื่อง node มีการเชื่อมต่อกับ front-end

1. **โครงสร้างอย่างง่ายสำหรับ OpenNebula มีดังนี้**

1.1 Front-end สำหรับ run OpenNebula และ Cluster Services

1.2 Nodes เอาไว้ povision virtual machine ต้องสนับสนุน Hypervisor

1.3 Image repository เป็น storage แบบใดก็ได้เอาไว้เก็บ image ของ VM

1.4 OpenNebula daemon เป็น service หลักของระบบซึ่งจะจัดการ life-cycle ของ VMs และจัดการระบบอื่นๆ ใน Cluster เช่น network, stroage และ hypervisor เป็นต้น

1.5 Driver เป็นโปรแกรมที่ใช้เชื่อมต่อกับ cluster subsystem ในรูปแบบพิเศษเช่น hypervisor หรือ storage file system เป็นต้น

1.6 oneadmin ผู้ดูแลระบบ private cloud ซึ่งจะเป็นผู้จัดการ VM, Virtual Network, node และ users ในระบบ

1.7 Users ผู้ใช้ในระบบ OpenNebula สร้างและจัดการ virtual machine, virtual network ของตนเอง

2. ความต้องการของระบบ

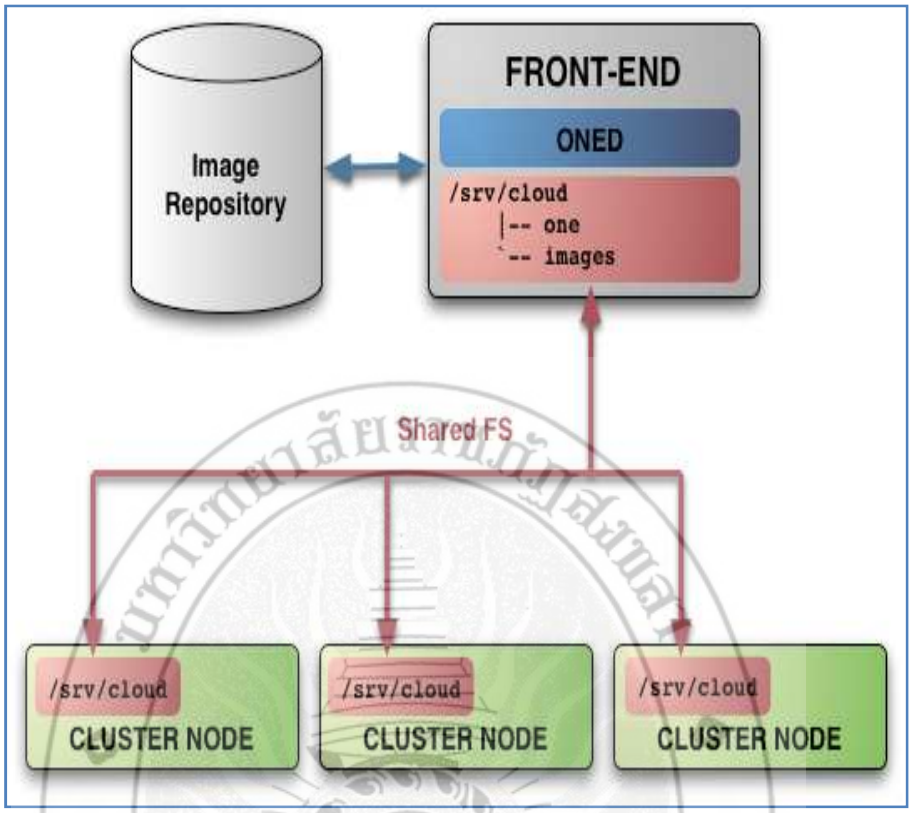
2.1 Cluster Front-End จะเป็นเครื่องที่เข้าถึง image repository ซึ่งควรจะมี storage ขนาดใหญ่เพื่อรองรับ VM images สำหรับ Private Cloud โดยทั่วไปจะเก็บ master image ซึ่งอาจจะ clone เมื่อสั่ง start VM เครื่อง front-end ควรวางแผนในเรื่องการจัดการ storge ด้วยซึ่งจะมากน้อยขึ้นอยู่กับจำนวน VM ที่ run อยู่บน cluster node สำหรับการติดตั้งเฉพาะ OpenNebula ใช้พื้นที่เพียง 10MB เท่านั้น OpenNebula สามารถติดตั้วได้ 2 โหมด ดังนี้

2.1.1 system-wide : binary, log files และ configuration ทั้งหมดจะถูกติดตั้งและจัดการโดย root ซึ่งต้องใช้สิทธิ์ root ในการบริการจัดการ

2.1.2 self-contained : OpenNebula จะติดตั้งในที่ๆ กำหนดโดยระบบเอง ไม่จำเป็นต้องใช้สิทธิ์ root ในการบริหารจัดการ

2.2 Cluster Node เครื่อง Node จะ run VMs ไม่ต้องการ storage เพิ่มเติมแต่อย่างใด ยกเว้นเครื่อง Node ไม่ได้ใช้ Share storage ร่วมกับ Node อื่นๆ ข้อมูล image จะเก็บไว้ที่เครื่อง Node แทน

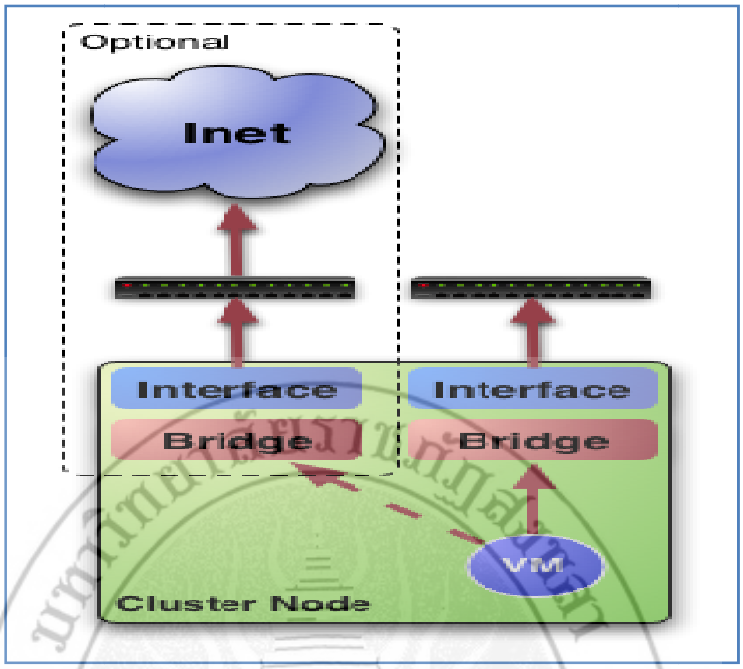
2.3 Storage เครื่อง front-end จะทำหน้าที่ export image repository ให้กับ cluster node ซึ่งขนาดของ storage ที่ใช้เก็บเป็นเรื่องสำคัญ ทั้งนี้ขึ้นอยู่กับจำนวน VM และขนาดของ image VM ด้วย ซึ่งการ start VM ตัว front-end จะ clone image ออกมาเพื่อใช้งานกับ VM ที่กำลังจะ start ดังนั้นต้องมีพื้นที่เพียงพอที่ใช้เก็บ VM image ทั้งหมด



ภาพที่ 2.10 VM image ทั้งหมด

2.4 User Account ผู้ดูแลระบบ OpenNebula จะใช้ชื่อบัญชีผู้ใช้ว่า oneadmin ซึ่งจะเป็นผู้จัดการ OpenNebula Services รวมถึงงานอื่นๆ ด้วย เช่น จัดการ VM, Virual Network, Image เป็นต้น สามารถประยุกต์ใช้ NIS เข้ามาช่วยจัดการบัญชีผู้ใช้ของ OpenNebula ได้เช่นกัน

2.5 Network สำหรับระบบเครือข่าย ไม่ได้มีอะไรเป็นพิเศษมากนัก หากต้องการได้ประสิทธิภาพให้กับ VM ควรจะมี NIC ที่มากกว่า 1 เชื่อมต่อไปที่ VM โดยการ bridge NIC เข้าไปที่ VM

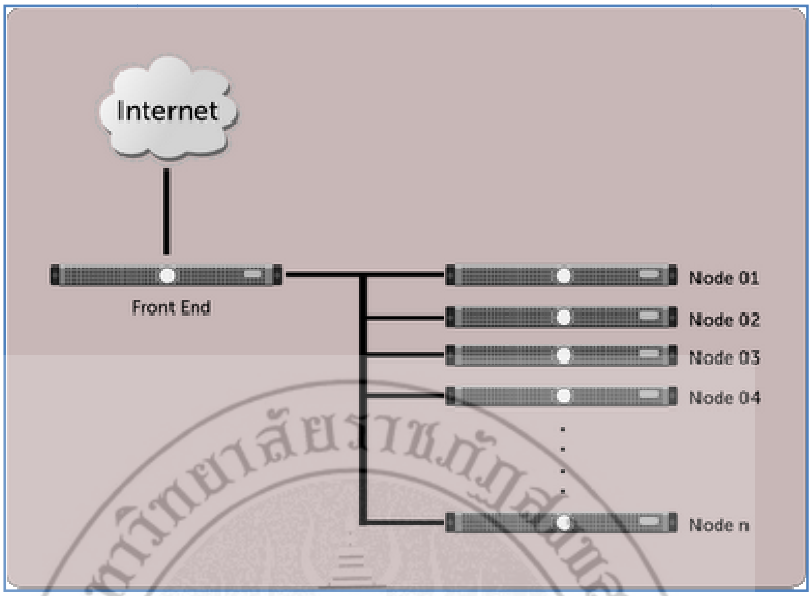


ภาพที่ 2.11 การ bridge NIC เข้าไปที่ VM

2.6 Secure Shell Access Front-End ทำงานกับ Node โดยผ่าน ssh โดยใช้ชื่อผู้ใช้ oneadmin ดังนั้นการเชื่อมต่อระหว่าง Front-End กับ Node ควรเชื่อมต่อแบบ ssh โดยใช้ ssh key ซึ่งการเชื่อมต่อแบบนี้ไม่จำเป็นต้องใช้ password อีกต่อไปแต่จะใช้ Key Authentication แทน

2.7 Hypervisor เทคโนโลยี Virtualization จะต้องติดตั้งที่ cluster node ซึ่งบัญชีผู้ใช้ oneadmin ต้องสามารถควบคุมจัดการและติดตาม VM ได้ OpenNebula สามารถทำงานร่วมกับ Virtualization Technology ได้หลากหลาย เช่น Xen, KVM, VMWare เป็นต้น

OpenNebula ออกแบบมาโดยใช้โครงสร้าง Cluster เดิมที่มีอยู่แล้วได้ ดังนั้นจาก Minimum Configuration เดิมซึ่งสามารถใช้งานได้เลย จากภาพเราจะมีเครื่อง Front End (Head Node) เพื่อทำหน้าที่จัดการ Image Repository, VMs, Scheduling ใน Cluster Node ที่เชื่อมต่ออยู่ จาก Minimun Configuration เครื่อง Front End จะมีการ์ดแลนอย่างน้อย 2 ใบ ใบหนึ่งเชื่อมต่อไปยังเครือข่ายภายนอก หรือ Internet ส่วนอีกใบหนึ่งจะเชื่อมต่อกับ Cluster Node ดังภาพ สำหรับเครื่อง Node มีการ์ดแลนอย่างน้อย 1 ใบ



ภาพที่ 2.12 ส่วน Internet ส่วนเชื่อมต่อกับ Cluster Node

เครื่อง Front End จะทำหน้าที่ควบคุม VM และ Image Repository ดังนั้นเครื่อง Front End จะเปิด Services NFS เพื่อให้เครื่อง Node ทั้งหมดเข้ามาเรียกใช้ VM Image จากเครื่อง Front End ได้ ดังนั้นในการขยายขนาด Stroage คุณสามารถเลือกใช้ Distributed Storage เพื่อเพิ่มจำนวน Node และ Storage ได้ในตัว (วิธีการนี้ไม่ขอพูดถึงในเนื้อหาครับ) สำหรับเครื่อง Cluster Node แนะนำให้เป็นเลขคู่ 2, 4, 6 และ 8 เป็นต้น สำหรับ Cluster Node ที่เราจะติดตั้งมี 2 เครื่อง

หลังจากเราสร้าง Cluster ของ Private Cloud กันไปในตอนที่แล้ว ตอนนี้จะกล่าวถึงการสร้าง Services Image ให้กับ OpenNebula กันครับ OpenNebula ใช้เทคโนโลยีที่มีอยู่เข้ามาปรับใช้ได้อย่างลงตัว ไม่ว่าจะเป็นการใช้งาน Virtualization อย่าง Xen, KVM หรือ แม้กระทั่ง VMWare ผ่าน OpenNebula Driver การจัดการ Disk Image หรือ Service Image เราสามารถใช้ Image ได้หลากหลาย ไม่ว่าจะเป็น raw, qcow, qcow2, VirtualBox, หรือแม้กระทั่ง VMWare ได้เช่นกัน

สรุป OpenNebula เป็นโครงการ Cloud Open Source ที่ได้รับความนิยมมากตัวหนึ่งเพราะความยืดหยุ่นและการใช้งานร่วมกับระบบ อื่นๆ ได้ (ecosystem) นั่นหมายความว่าคุณสามารถติดตั้งให้ OpenNebula เป็น Cloud แบบใดก็ได้ ไม่ว่าจะเป็น Private Cloud, Public Cloud หรือแม้กระทั่ง Hybridge Cloud การเชื่อมต่อ OpenNebula Cloud เราสามารถขยายขนาดออกไปได้เรื่อยๆ ไม่ได้ยึดติดว่าจะต้องเป็น Cloud Infra ที่ใช้ OpenNebula เหมือนกัน

Web-Based Application

Web-Based คือ การทำงานผ่านทางโปรแกรม Browser ซึ่งอาศัยการเชื่อมต่อทางอินเตอร์เน็ต หมายความว่าเพียงแค่มีโปรแกรม Browser ไม่ว่าจะเป็น Internet Explorer, FireFox, Safari, Opera หรือแม้กระทั่ง Google Chrome ก็สามารถใช้งานโปรแกรม หรือ Applications ใดๆ ก็ได้ โดยโปรแกรมหรือ Applications เหล่านั้น จะติดตั้งบน Server แห่งใดแห่งหนึ่ง หรืออาจติดตั้งในสำนักงานใหญ่ก็ได้