設計與實作一基於虛擬化技術之網路及網路安全雲端測試平台

蔡邦維[1] 賴瑀庭[1] 羅孟彥[2] 楊竹星[1,*]
成功大學電機系電腦與通信工程研究所[1]、高雄應用科技大學資訊工程系[2]
csyang@ee.ncku.edu.tw[*]

摘要

由於近年來雲端技術與服務的發展，許多研究團隊都投入了虛擬化技術及雲端服務的開發。在網際網路蓬勃發展的現在，日新月異的新協定和應用亦使得資訊傳遞的安全性議題日趨複雜。因此具備仿真環境的測試平台就扮演了關鍵的重要角色。成功大學資通安全研究與教學中心建置的Testbed@TWISC 測試平台自 2006 年起即提供學術單位運算節點的租賃服務，提供設備測試、情境模擬和網路及網路安全議題的研究及教學使用。然而在虛擬化技術蓬勃發展的現在，許多原本依靠傳統實體主機處理的工作已逐漸被雲端運算單元所取代。況且以實體主機作為節點，其運算能力受限於單一主機，配置方式亦不一定能滿足虛擬化環境中的情境需求。因此本論文將提出一基於虛擬化技術之雲端測試平台系統設計，此系統以現有的Testbed@TWISC 平台為基礎，針對虛擬化資源新增控制模組以支援虛擬機器節點及軟體定義網路，並實作虛擬機器群組租賃及虛擬化網路拓樸的服務模式。並藉由虛擬化技術的優點使得測試平台硬體資源能夠有更加充分利用，滿足測試平台應具備的隔離性、封閉性、可紀錄性、可控制性與可儲存性，以實現一個基於虛擬化技術之網路及網路安全雲端測試平台。

關鍵詞：虛擬化、虛擬機器、群組租賃、拓樸、軟體定義網路、雲端測試平台

1. 前言

在網際網路蓬勃發展的現在，衍生出了許多新型態的協定和應用服務，相對也使得協定的驗證、服務的運作和資訊傳遞的安全性等議題日趨複雜。在這種情況下，具備仿真環境的測試平台即扮演著測試與驗證的關鍵角色。在 2003 年由行政院及國科會支持成立的台灣資通安全研究與教學中心(Taiwan Information Security Center, TWISC)，在台灣科技大學、交通大學及成功大學這三間學校設立了北中南三個分支。其中位於成功大學的資通安全研究與教學中心(TWSIC@NCKU)研究重點之一即為大型測試平台的設計，目標為建置一個能提供網路及網路安全測試的平台，讓開發人員進行軟硬體的測試與驗證。第一代的測試平台具備了 80 多台的實體主機、網路設備及資安分析設備，提供實體主機作為測試平台的租賃服務。但為了滿足封閉且隔離的測試環境，使用者需要親臨現場操作，而且也缺乏高效率的資源管理及紀錄保存。於是在 2007 開始建置的第二代測試平台採用了 Emulab[1]系統做為控制架構，並修改原始碼以支援不同規格的實體主機以符合開發和使用需求。此一測試平台(Testbed@TWISC[2])具備了多種層級的管理者權限控管等功能，其網頁介面能提供實體主機租賃服務和自訂網路拓樸，讓使用者擁有一個仿真的私有基礎設施。目前該平台擁有 220 台實體節點以及 10G 核心網路。為國內第一、世界第三大[3]的 Emulab 實驗平台。其設備放置於國家高速網路與計算中心南群機房，如圖 1。累計至今已有一千三百多名的使用者註冊、創建了八千個以上的情境腳本以及進行超過兩萬次的實驗。

然而在虛擬化技術蓬勃發展的現在，許多依靠傳統實體主機處理的工作逐漸被雲端運算單元所取代，使用傳統的實體主機所打造的測試平台，不一定能仿造虛擬化環境中的實驗情境。故自 2009 年起，成功大學資通安全研究與教學中心即開始進行虛擬化節點之可行性研究。但由於種種技術上的因素，雖然使用者可以自行安裝虛擬化軟體於實體主機上，但每台實體主機本身硬體資源有限，建置較大規模的架構不易。除此之外，對於系統而言每台實體節點難以共享資源結合成為一個龐大的資源池(Resource Pool)，無法達到更有效的資源使用率。綜合以上原因，本論文將提出一基於虛擬化技術之雲端測試平台系統設計，此系統以現有的Testbed@TWISC 平台為基礎，新增多個控制模組以支援虛擬機器節點及軟體定義網路。論文中亦將介紹虛擬機器群組租賃及虛擬化網路拓樸的服務模式、真實使用者的運用案例、目前所遭遇的問題、預期改善方式和未來發展方向。

本論文的第二章將介紹目前國內類似性質的雲端服務及測試平台。第三章及第四章將說明系統架構設計及實作細節。第五章是功能驗證與效能測試，第六章會介紹使用案例和討論，最後則是結論。

圖 1　Testbed@TWISC 平台

2. 相關研究

由於近年來雲端技術與服務的蓬勃發展，許多團隊都在投入運用虛擬化技術提供資源租賃服務的系統開發。有些研究團隊選擇了 Nimbus[4]、OpenStack[5]、OpenNebula[6]或 Eucalyptus[7]等專案進行修改，有些團隊則自行設計。此章節將會介紹目前國內幾個自行開發的雲端服務及平台。

2.1 簡單龍雲端服務

簡單龍雲端服務(EasyCloud[8])是國家高速網路與計算中心提供的虛擬機器租賃服務。其核心技術為雲端龍 Ezilla[9]專案，前端架構具備了簡潔的網頁介面，使用者可透過網頁建立與存取虛擬主機，這點與目前 Testbed@TWISC 平台的操作介面類似。其後端則是使用 KVM[10]技術的虛擬化環境。Ezilla 的特點為佈署與使用容易，目前交通大學的雲端管理系統[11]，即是基於此專案技術所建置的。以終端使用者和系統的觀點來看，Ezilla 系統中同一個終端使用者創建的虛擬機器相互為獨立個體。這部分與本論文中預期使用者會需要建立多個具關聯性的虛擬機器的服務取向稍有不同。

2.2 SAMEVED

SAMEVED[12, 13, 14]是中央大學平行與分散計算實驗室所開發的機房服務平台建置與管理軟體，使用者可使用虛擬私有網路(VPN)存取虛擬機器。其網頁介面包含了帳號管理，虛擬機器管理與監控，映像檔製作和虛擬區域網路標籤設定等功能，並具備群組租賃的概念(其稱之為虛擬資料中心)，能夠藉由文字指令設定虛擬機器網卡之連接。但在本論文撰寫時其公開發表之資料有限，且並未如同 EasyCloud 或者 Testbed@TWISC 一樣有具規模性質的公開服務平台，故僅能參考已發表之論文去了解，對於其實作部分並無實際的使用者經驗。

2.3 其他具備虛擬主機租賃服務之平台

除此之外，國內尚有其他具虛擬化技術之實驗平台，像是淡江大學團隊提出利用 OpenVZ[15]專案在 Xen[16]虛擬環境下再造出供學生實習使用的作業平台[17]，當使用者要求虛擬機器資源時其系統會配置一台內建 OpenVZ 環境的虛擬機器給使用者，使用者再於 OpenVZ 的環境下創建自己的實驗節點。高雄軟體科技園區的雲端運算實驗中心亦有於影音平台上公開其研發的 TRCK 雲端測試平台之資料[18]。

綜合以上幾個類似性質的雲端服務及虛擬化相關的平台技術，對於虛擬化資源的存取部分，主要著重在方便的介面和橋接虛擬機器管理者。而在虛擬化網路的部分，似乎無適合直接套用於現有 Emulab 系統的設計而須自行開發。除此之外，我們認為在測試平台上對於節點的租賃的需求都朝向由多個節點組成的架構。系統會切割出資源，提供使用者仿真(Emulated)的基礎設施(Infrastructure)去發展。因此提供虛擬機器群組租賃以及使用者自行定義的虛擬網路拓樸會是這類型雲端服務的取向，即是本論文中設計之核心目標。

3. 系統架構

由於傳統提供雲端運算節點租賃的服務平台以單一虛擬主機租賃服務為主，對於虛擬化網路拓樸的部分支援度較少，亦難以滿足作為網路及網路安全測試平台須具備的隔離性、封閉性、可紀錄性、可控制性與可儲存性之環境。因此本論文設計的新測試平台架構將著重於實現兩項重點功能：虛擬機器群組租賃服務和虛擬化網路拓樸，以實現一個具備虛擬化技術之網路及網路安全測試平台。

本論文中設計的測試平台主要目的是作為 Testbed@TWISC 平台的下一代候選版本，並提供網路及網路安全測試與研究教學使用。設計時，對於使用者向平台進行資源租賃的動作，視為在測試平台上建立一個實驗。此建立實驗的動作包含了向系統提交需求、獲得分配的資源以及存取資源。測試平台中共分四種資源租賃模式，包含使用實體機器作為節點的 Real Node Mode;以虛擬機器作為節點的 Virtual Node Mode; 以虛擬機器作為節點且使用 XVP[19]介面的 XVP Mode 和啟用具軟體定義網路功能之 OpenFlow[20] Switch 和 POX[21] Controller 的 OpenFlow Network Mode。使用者可自行選擇合適的實驗模式。當使用者要終止資源租賃服務時，視為在平台上結束一個實驗。結束實驗時系統會保存相關的設定資訊，並釋放占用的資源，流程圖如

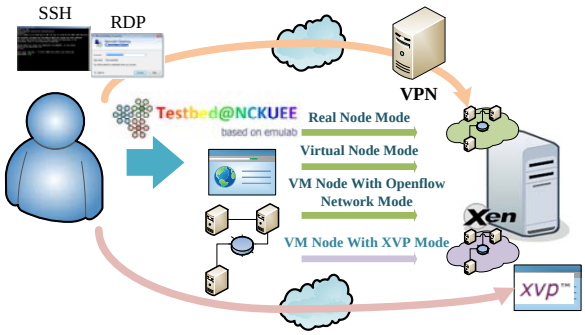

圖 2 實驗創建流程圖

圖 2 所示。如果使用者在實驗結束後又想要重新建取得同樣配置的資源，可以藉由已儲存的資訊快速建立。每個實驗所配置的資源可能由一個節點或多個節點組成，而節點與節點之間亦可能有網路連接。除此之外，每一個實驗都是獨立的，分屬不同實驗的節點彼此之間無法互通。

為了方便系統的開發，新設計的測試平台共分成五個主題：介面、權限、控制結構、資源管理及操作監控。介面部分因為原本的 Testbed@TWISC 平台已具備完整的網頁介面可供操作，僅需小部分的修改即可實現多層次的權限管理。至於控制結構和資源管理則須重新設計，並整合虛擬化的資源。特別是資源管理部分，由於原本的 Testbed@TWISC 平台對於資源的控管方式為熄燈式管理(Lights-out Management, LOM)，例如使用遠端電源控制器切斷和供給運算節點主機電源，此一方式在虛擬化的環境中並不適用，故需規劃一個新的資源管理機制去整合實體與虛擬兩種類型的資源。最後在監控部分，新設計的測試平台須能監控實體節點與虛擬節點的運作狀況，並具備容錯功能，以及將資訊呈現在介面提供使用者和管理者知悉。

在接下來的章節，會把新設計的測試平台系統依上述的介面、權限、控制結構、資源管理及操作監控這五個主題分別進行介紹。

3.1 介面(Interface)

本論文中所設計的測試平台在前端介面部分共分成兩層，第一層位於前端伺服器，授權的使用者能經由測試平台的網頁介面建立、結束、重啟實驗，知悉實驗的細節，包含實驗拓樸，實驗節點資訊。這部份主要繼承於 Emulab 的原生設計，其網頁介面如圖 3 所示。具備了能讓使用者設定實驗拓樸的 NetBuild[22]模組，因此可確保使用者轉換到升級後的新平台不會有操作上的困難。

在第二層部分，則是後端的資源存取。依照使用者選擇的實驗創建模式共分為兩種，第一種是使用 IP 網路作為控制實驗節點的存取。適用於 Real Node Mode、Virtual Node Mode 和 OpenFlow Network Mode 模式。利用以上三種創建模式所建立的實驗，可透過 VPN 連線，再藉由安全殼協議(SSH)或遠端桌面協定(RDP)進行遠端存取。第二種則是使用 VNC 代理(VNC Proxy)的方式。讓使用者連至代理伺服器，經由其網頁介面存取實驗節點。

3.2 權限(Authorization)

新設計的測試平台中，對於權限管理也同樣分為兩層。使用者所擁有的權限會決定介面上可供使用的功能，此為第一層。管理者可在平台上為特定目的用途設立專案(Project)，並指定專案的上位使用者(Project Leader)，並由上位使用者審查是否讓一般使用者作為新成員(New User)加入。此部分和現有的 Testbed@TWISC 平台不同之處在於新設計的系統加強了與操作介面有關的安全控管，並限制了專案使用者從介面上能取得的資訊。在第二層的權限部分，亦是屬後端資源存取的管理。為了確保節點只能由創建者進行存取，節點中會植入使用者於前端平台註冊時相同的帳號密碼，或產生的亂數密碼加強節點存取的安全性。

3.3 控制結構(Control Framework)

控制結構為系統後端的主體，承接前端的介面以及資源控管。其中在管理模組部分是沿用現存的 Testbed@TWISC 平台中的設計。由於加入了虛擬化資源的存取，故新增了一個模組作為中介層來承接異質性資源的控制命令，如圖 4 所示灰色區塊 Testbed@TWISC Control Framework 所包含的模組。此一中介模組能根據不同的創建模式，將需求遞送到對應的資源管理層中的模組。其中 Real Node 模式如同實體節點，Virtual Node 模式則是將使用者透過介面提交的需求解析並轉譯成虛擬機器管理者界面能接受的指令。而 OpenFlow Network Mode 是基於 Virtual Node Mode 進一步將虛擬實驗網路用 Software-based 的 OpenFlow Switch 和 POX Controller 來構築連接虛擬節點間的虛擬網路。在此我們將著重於與虛擬資源相關的控制結構作介紹。

對虛擬化的運算節點部分，是由預先安裝不同作業系統的虛擬機器範本依照使用者的需求產生虛擬機器映像檔。當使用者利用 NetBuild 網頁設計實驗拓樸並由系統產生相對應的 NSfile[23]後，節點的作業系統、硬體規格和實驗網路的型態都會由

圖 3　測試平台首頁

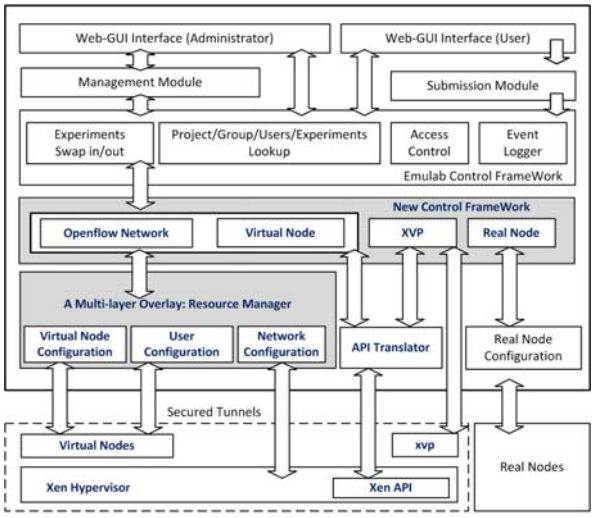

圖 4　系統架構圖

控制結構搜尋可用資源，並送交至下一小節將會介紹的資源管理覆蓋層(Overlay)執行配置命令，來建立使用者要求的虛擬節點群組和虛擬網路。虛擬節點的映像檔都盡可能節省磁碟空間，並維持作業系統的原生套件版本以確保創建的單純環境。而使用者每次配置的資源需求都會由控制結構紀錄配置資訊，當資源釋放後若使用者想要再一次租賃相同配置的資源時即可快速提交需求，或修改和刪除此一配置資訊。實現可紀錄性、可控制性與可儲存性。

在連接虛擬節點的虛擬網路部分，常見的測試平台通常會區分成控制網路(Control Network)和資料網路(Data Network)。控制網路是傳輸系統控制命令、監控資訊和提供使用者遠端連線存取節點等，資料網路則為節點間資料傳遞之用。在本論文第二章中提及的幾個雲端服務與測試平台，即有提供使用者以 VPN 方式連入其控制網路以存取租賃的節點。這對於一般的虛擬機器租賃服務是合理而且適用的，然而若作為一個具網路安全測試功能的平台，這樣去劃分看似為獨立的兩種網路，但實質上卻不能確保其安全性。原因為使用者是不可以信任的，即使分配給使用者的節點有對兩種網路配置了不同的網路卡介面。由於使用者具備了一定程度的作業系統權限可修改節點內的設定，當設定不慎時即有可能將資料網路的封包誤傳控制網路進而發生問題。例如像是進行分散式阻斷攻擊實驗時，若使用者在節點上設定了錯誤的路由表，即有可能影響控制網路甚至造成系統癱瘓。為了改善此一問題，本論文中實作了一個搭配 XVP 控制模式去簡化控制網路的佈署和操作。使用此模式時，控制結構可在劃分資源時進行限制，讓使用者租賃的虛擬機器群組及網路都位於同一台實體伺服器上，縮減因節點發生問題導致實體機器受影響的範圍。在 XVP 模式下，使用者透過網頁介面聯繫 VNC 代理伺服器，再透過其網頁介面存取和控制虛擬節點。

3.4　資源管理(Resource Management)

為了能讓使用者存取虛擬環境中的資源，在資源管理的部分新增了一個資源管理模組扮演著覆蓋層的角色，如圖 4 灰色區塊 Muti-layer Overlay: Resource Manager，其運作方式如同虛擬機器管理者(Virtual Machine Manager)的代理，能夠配合控制結構的命令進行資源池中的資源分配(Resource Allocation)。此部分是實作在提供虛擬資源的實體主機上。受分配的資源可區分為虛擬機器、虛擬機器網路卡、虛擬網路三種。此一模組會根據控制結構傳來的要求，檢視目前剩餘資源足夠的實體機器，並選擇一個適合的實體主機傳送命令給虛擬機器管理者生成虛擬機器、虛擬網卡和符合拓樸設計的虛擬網路。

本論文中設計的資源管理層，能支援群組租賃的模式概念，可接受多台不同作業系統的虛擬機器和虛擬網路拓樸的要求，以及在完成虛擬資源的分配後自動檢查是否創建了正確數量的虛擬機器和虛擬網路。每張虛擬網卡的實體位址碼(Media Access Control Address, MAC Address)具備亂數和預設兩種產生方式。如該虛擬網卡介面須設定 IPv4 位址，則其位置碼會由一個唯一辨識碼(Universal Unique IDentifier, UUID)再加上轉換為 16 進位的 IPv4 位址組成。虛擬機器啟動後預設的系統服務會自動解析虛擬網卡的實體位置碼並設定 IPv4 位址於該網路介面。

由於測試平台須具備的隔離性和封閉性是提供使用者存取資源時的兩個重要指標，而連接節點間的網路即是此部分的關鍵，控制結構須能確保分配給使用者的網路資源是個人私有的。若網路為實

體網路因其網路交換器多半是共用的，可使用虛擬區域網路標籤(Virtual Local Area Network Tag, VLAN Tag)來區隔。此一方式實作容易。然而虛擬區域網路標籤有 4K 的數量限制，當預期的使用者較多就必須搭配其他設計達到高效率的配置。有些系統亦會搭配通道技術(Tunnel)跨越實體網路解決此一問題。但本系統中設計強制讓每個使用者租賃的資源位於同一台實體伺服器內。雖然這樣會造成租賃規模的限制，但由於虛擬網路皆屬內部橋接，故可避免掉跨越實體網路介面埠口會遇到的安全問題。除此之外，本論文設計的系統並未規劃將常見於網路模擬軟體中的傳輸延遲(Transmission Latency)和封包遺失率(Packet Loss Rate)之功能提供使用者，原因是本系統設計的目標是朝向仿真(Emulation)而非動態調整之模擬(Simulation)環境，故虛擬化環境中的網路傳輸應是 Best-effort 的運作模式，我們認為這樣較能反映出實際的狀況。

3.5 操作監控(Operation Monitor)

監控與容錯一直都是系統設計中重要的一環。在 Testbed@TWISC 系統上已經設計了一個基本的事件紀錄模組，負責記錄系統所有行為，包含工作提交、資源分配及回收等。經過移植(Porting)後即可使用在新系統上。另外透過資源管理覆蓋層從虛擬機器管理者取得的反饋資訊，也可以得知每次配置資源和回收資源的狀況。並藉由自我測試的模組可確認是否已正確執行了控制結構的命令。因此當系統出現異常時，平台維運人員可藉由這些資訊檢視系統過去發生的行為或判斷問題點。

4. 測試平台軟硬體配置及佈署

為了不影響目前上線中的 Testbed@TWISC 之運作，本論文實作的部分是建立於一套備用的系統上(Testbed@NCKUEE[24])進行開發。系統運作環境如圖 5 所示，以千兆乙太網路(Gigabit Ethernet)建構核心網路並連接前後端伺服器、儲存設備、資源池伺服器群、實體節點等設備。平台上有關虛擬化資源的軟硬體資料如表 1。目前測試平台上提供了多種不同的虛擬機器範本，使用者可指定或混用多種不同作業系統的虛擬機器作為實驗節點。但目前實體節點與虛擬節點的混用尚在測試中故不開放使用。除此之外，考量到測試平台須能滿足資安實驗所需的隔離與封閉環境，租賃的虛擬機器群組都會由資源管理模組配置在同一台實體機器上，虛擬機器間的虛擬網路不會跨越實體網路，以確保實驗網路的單純性。實現一個具備虛擬化技術之網路及網路安全測試平台的雲端服務。

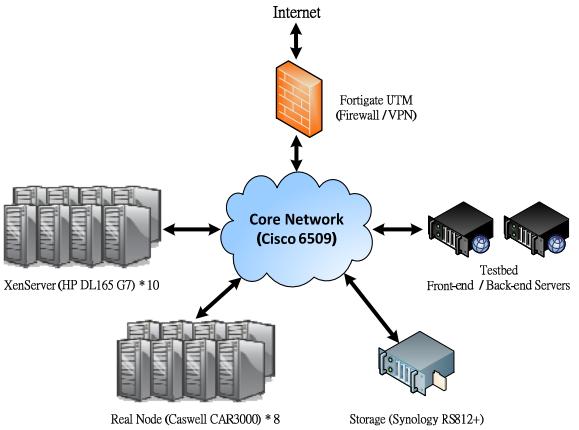

圖 5 系統運作環境

表 1 虛擬化設備相關硬體規格

項目	規格型號
網路交換器	Cisco WS-C6509
前後端伺服器	HP DL120 G7 (with 4 cores Intel CPU and 4G RAM)
儲存設備	Synology RS812
資源池伺服器	HP DL165 G7 (with 16 cores AMD CPU and 64G RAM)
Emulab 版本	4.336
XenServer 版本	6.1

5. 功能驗證與效能測試

本論文中實作的測試平台為基於現行運作的 Testbed@TWISC 平台發展而來，對於網頁介面、權限管理及前後端伺服器等部分皆遵循既有的設計理念，故限於篇幅故在此並未列出檢核項目的細節。本章節將著重於虛擬化節點部分的功能驗證與效能測試。共分為三小節，包含虛擬機器節點的創建效能測試結果，虛擬網路拓撲驗證以及虛擬機器進行分散式阻斷服務攻擊實驗情境的測試的介紹。

5.1 虛擬節點群組租賃測試

本論文中所設計的系統須能提供使用者虛擬機器群組租賃的服務，故須考量使用者提交的資源需求會同時有多個虛擬機器。而在 Xen 對於藉由虛擬機器範本產生虛擬機器磁碟映像檔時，有 Fast Clone 和 Full Copy 兩種方式[25]。使用鏈結方式複製映像檔範本的速度快，空間也較節省，較能符合需求。然而鏈結方式越長會使存取效能越來越差，導致虛擬機器啟動後的磁碟讀寫效能下降。故本實

驗分為兩部分：第一個實驗為連續進行 25 次虛擬機器的產生於儲存設備(Storage)中，並計算每次複製所需的時間，以及使用本機磁碟(Local Disk)作為對照比較其效能差異。使用的虛擬機器範本配置 2 個 vCPU、512MB 的記憶體以及 8GB 的虛擬機器磁碟空間。實驗的結果如圖 6 所示。從圖中可發現，儲存設備的速度比起本機磁碟還快，但單次產生虛擬機器映像檔所花費的時間都差不多。在第二個實驗部分，在產生映像檔後隨即啟動虛擬機器，並計算複製開始到虛擬機器啟動的時間。實驗二的結果如圖 7 所示，儲存設備和本機磁碟的時間都會隨著啟動並運作的虛擬機器數量變多而漸漸增加，但仍在可接受的範圍內。

綜合以上兩個實驗的結果，我們評估現有的單台伺服器硬體應付 25 台虛擬機器以內的小規模群組租賃是足夠的。在比較儲存設備和本機磁碟的實驗結果後，亦可發現關鍵點為儲存設備的效能。若能將使用者需求平均分散在全部資源池的 10 台實體伺服器上，不考慮 CPU 及網路使用率，最佳情況下估計應可乘載 250 個的虛擬機器運作無虞。但隨著未來虛擬節點乘載量的成長，共用的儲存設備應改用較高等級的規格以避免效能上的瓶頸。

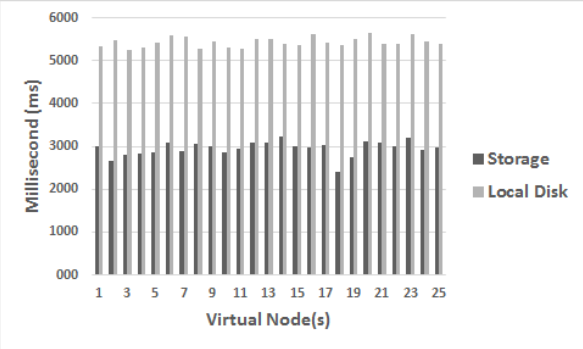

圖 6 虛擬機器新增時間(毫秒/個數)

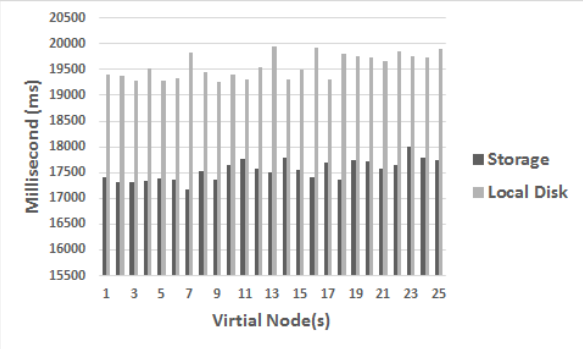

圖 7 虛擬機器新增及啟動時間(毫秒/個數)

5.2 虛擬網路拓樸驗證

本論文中所設計的測試平台第二個主要重點，是提供自定義的虛擬化網路於使用者創建的實驗中。控制結構要能正確接收使用者要求的拓樸資訊，並由資源管理層切割出虛擬網路資源。在驗證數個由 Testbed@TWISC 轉移過來的實驗時，最後皆得到符合預期的拓樸和實驗結果。本章節將列舉其中的一個測試案例—由預設開啟封包轉送功能之 Linux 作業系統的節點自動將網卡介面的封包根據作業系統的路由表進行埠口轉送之實驗情境。此實驗情境裡的節點其拓樸設定如圖 8 所示。拓樸由 16 個節點所組成，節點與節點間相連在一起。相連的節點其網路介面會設定同個子網域的 IPv4 位址。實驗測試方式為使用 Ping[26]工具進行測試，並計算 Ping 封包的來回時間(Round Trip Time)。實驗的結果如圖 9 所示。自其中一個位於拓樸末端的節點使用 Ping 工具量測與其他節點的互通狀況時，中間所經過的節點越多其封包來回時間所花費的時間就越長，此一數據符合預期的結果。

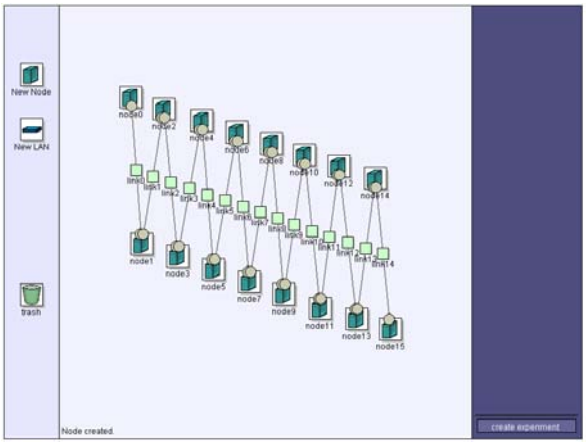

圖 8 於測試平台網頁上設定的實驗拓樸

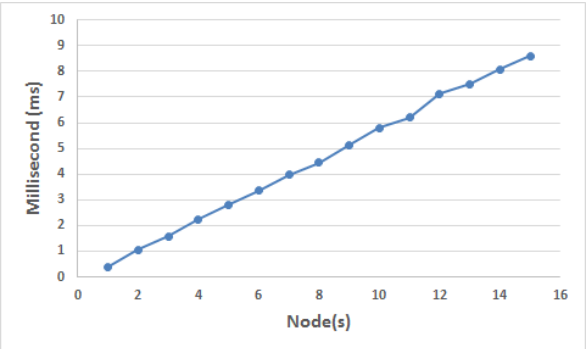

圖 9 RTT 數據(毫秒/經過的節點數)

5.3 資安實驗情境測試

為進一步了解虛擬節點的網路效能，我們選擇了常見的分散式阻斷攻擊主題進行測試。本實驗使用了教育部資通安全聯盟資安實習教材中的情境，其網路拓樸設計如圖 10 所示，包含了 1 台作為攻擊主機的節點、16 台跳板節點和 1 台受害主機節點，合計需要 18 台虛擬機器。分別測試 Ping of Death 以及 Syn Flooding 兩種攻擊行為。實驗結果顯示兩種攻擊行為下均能使受害主機產生預期的癱瘓結果。同時為了測試虛擬網路的效能，本實驗使用了兩種測試工具，分別是使用 Sar[27]工具程式在 5，10，15，20，25，30 分鐘的流量取樣，同時亦以安裝在受害主機上的 BitMeter OS[28]監控收到的封包量作為比對。

本實驗結果如表 2 所示。表 2 是虛擬網路橋接介面的封包轉送流量，在不計因緩衝佇列(Buffer Queue)溢出而被丟棄的封包情形下，16 台虛擬機器最高可同時產生約 8.53Gbps 的流量到橋接介面。BitMeter OS 在受害主機上所量測到的累計封包接收量接近 13GB，最終量測結果亦符合預期。此實驗結果與交通大學團隊設計的攻擊與防禦實驗平台[29]參考比較，可看出本論文實作的系統及搭配的硬體對於網路傳輸亦有不錯的封包承載能力。

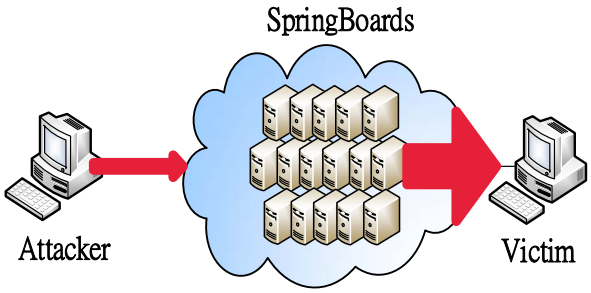

圖 10 分散式阻斷攻擊情境

表 2 虛擬網路橋接介面流量統計

取樣時間	量測結果
第 05 分鐘	6.57 Gbps
第 10 分鐘	6.06 Gbps
第 15 分鐘	6.46 Gbps
第 20 分鐘	8.01 Gbps
第 25 分鐘	7.88 Gbps
第 30 分鐘	8.53 Gbps

6. 討論

為驗證本論文中建置的測試平台在提供真實使用者使用時，系統是否能和內部壓力測試時有相同的乘載能力，我們選擇在 TWISC@NCKU 舉辦的第十屆全國資訊安全研習營請學員協助測試，測試過程中由課程講師協助建立註冊帳號，並由網頁介面設定實驗網路拓樸、送出虛擬化節點的實驗需求。目的是觀察控制結構的運作，以及系統乘載能力是否和壓力測試時結果一致。每位學員會送出由三台虛擬機器組成的虛擬機器群組需求，其中 2 台是 Ubuntu 的虛擬機器，另 1 台則是 Windows7 作業系統。在測試中，參加的使用者們在 19 分鐘內送出了 81 次的實驗建立要求(失敗後重複送出者皆計入)，其中大多數的實驗皆能正確建立，然而卻遇到兩個問題。首先是競態條件(Racing Condition)的部分設計不夠完善，在複製虛擬機器映像檔到啟動虛擬機器的時間比預期的長，由於控制結構對於資源使用率的取樣非即時同步，導致虛擬機器映像檔已複製完畢，但卻因資源管理層選定的實體伺服器記憶體不足而無法啟動虛擬機器。這在原本的測試中沒有遇到此一狀況。另外在 XVP 模式下所建立的虛擬節點群組，因全體使用者共用位於同一台虛擬機器上的入口網頁介面操作虛擬節點，導致該台虛擬機器負荷過重，刷新使用者可存取的虛擬節點過程會出現卡在等待(Pending)的狀態。關於上面的兩個問題，目前已進行資源管理排程器的修改，並提供分散式的入口網頁來分散負載。經過此一驗證可得知，本論文實做的系統在小規模(Branch-Scale)時可正常運作，但面對真實使用者使用時仍有許多地方需要處理，未來將會持續地進行測試與修正。

7. 結論

本論文提出了一個基於虛擬化技術的測試平台系統設計，修改現有的 Testbed@TWISC 測試平台架構，結合 Xen 的虛擬機器管理層和 Emulab 系統，使其支援虛擬機器節點及軟體定義網路，並在建置後實際運用於教學與研究使用。此系統能藉由虛擬化技術的優點使得硬體資源能夠有更加充分的被使用，滿足測試平台應具備的隔離性、封閉性、可紀錄性、可控制性與可儲存性。相較於其他測試平台，本論文中實作的系統修改了 Emulab 控制層及結合 Xen 的虛擬機器管理者層，能由加密通道控制資源池中的伺服器，動態建立及刪除分配給每個使用者的虛體機器，並結合虛擬網路技術，提供具備網路拓樸的虛擬機器群組租賃服務，作為網路及網路安全測試之教學與研究使用。

誌謝

本研究承蒙行政院國家科學委員會計畫編號 NSC 100-2218-E-151-003-MY3 和 NSC 102-2219-E-006-001.經費補助，以及國立成功大學頂尖大學計畫部分經費補助，特此感謝。

參考文獻

[1] Brian White, Jay Lepreau, Leigh Stoller, Robert Ricci, Shashi Guruprasad, Mac Newbold, Mike Hibler, Chad Barb and Abhijeet Joglekar, An Integrated Experimental Environment for Distributed Systems and Networks, 5th Symposium on Operating Systems Design and Implementation, pp. 255-270, 2002.

[2] Chi-Sung Laih, Jung-Shian Li, Mao-Jie Lin, Shiau-Han Chang, Li-Da Chen, Shih-Hsien Tseng and Michael Chang, Development and Operation of Testbed@TWISC., the 3rd Joint Workshop on Information Security, 2008.

[3] Other Emulab Testbeds, https://wiki.emulab.net/Emulab/wiki/OtherEmulabs

[4] Dynamic Creation and Management of Runtime Environments in the Grid, Keahey, K., M. Ripeanu, and K. Doering. Workshop on Designing and Building Web Services, 2003.

[5] OpenStack, http://www.openstack.org/

[6] OpenNebula, http://opennebula.org/

[7] Eucalyptus, http://www.eucalyptus.com/

[8] 簡單龍雲端服務, http://easycloud.nchc.org.tw/

[9] Building a Dedicated Cloud (Ezilla) - Integrating Storage, Network and Computing, NCHC E-Paper, Issue 69.

[10] Kernel-based Virtual Machine , http://www.linux-kvm.org

[11] 交通大學雲端管理系統, http://carweb.cs.nctu.edu.tw/cloud

[12] 黃瀞瑩，一個提供彈性虛擬資料中心的雲端服務平台，http://ir.lib.ncu.edu.tw/handle/987654321/48430

[13] 張凱富，雲端彈性虛擬機房服務平台之資源控管中心，http://ir.lib.ncu.edu.tw/handle/987654321/54430

[14] Shao-Jui Chen, Jing-Ying Huang, Cheng-Ta Huang and Wei-Jen Wang, SAMEVED: A System Architecture for Managing and Establishing Virtual Elastic Datacenters, International Journal of Grid and High Performance Computing, Vol. 5, Issue 2, 2013.

[15] OpenVZ, http://openvz.org

[16] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris,A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, “Xen and the Art of Virtualization,” in SOSP ’03: Proceedings of the nineteenth ACM symposium on Operating systems principles. New York, NY, USA: ACM Press, pp. 164–177, 2003.

[17] 陳俊廷，虛擬化網路安全實習驗室之規劃與建置，http://etds.lib.tku.edu.tw/etdservice/view_metadata?etdun=U0002-2605200816143400

[18] 雲端測試平台之彈性 IP 功能，高軟雲端運算實驗中心，http://www.youtube.com/watch?v=awHT-OB6nbw

[19] Cross-platform VNC-based and Web-based Management for Citrix XenServer and Xen Cloud Platform, http://www.xvpsource.org

[20] Nick McKeown; Tom Anderson; Hari Balakrishnan; Guru Parulkar; Larry Peterson; Jennifer Rexford; Scott Shenker; Jonathan Turner. OpenFlow: Enabling Innovation in Campus Networks. ACM 08, New York, NY, USA, March 14, 2008.

[21] POX, http://www.noxrepo.org/pox/about-pox

[22] Moore, K., Dongarra, J. "NetBuild," University of Tennessee Computer Science Technical Report, UT-CS-O1-461, 2001.

[23] NSfile, https://wiki.emulab.net/wiki/Tutorial

[24] Testbed@NCKUEE, https://testbed.ee.ncku.edu.tw

[25] XenServer Storage Overview, http://support.citrix.com/servlet/KbServlet/download/18670-102-671563/XenServer_Storage_Overview.pdf

[26] Ping, http://linux.die.net/man/8/ping

[27] Sar, http://linux.die.net/man/1/sar

[28] BitMeter OS, http://codebox.org.uk/bitmeterOs

[29] 蔡孟儒，許晏峻，吳育松，An IaaS Cloud For Attack and Defense Experiments, 第二十一屆資訊安全會議(the 11th Cryptology and Information Security Conference), 2011