

基於內容核心網路實現資料命名網路 – 以即時通訊為例 Implement Named Data Network based on Content-Centric Networking – An Instant Message Case Study

廖上全¹, 曾繁勛², 趙涵捷^{1,3}, 周立德²

¹國立宜蘭大學資訊工程學系

²國立中央大學資訊工程學系

³國立東華大學電機工程學系

R0043010@ms.niu.edu.tw; fanhsuntseng@ieee.org;
hcc@niu.edu.tw; cld@csie.ncu.edu.tw

摘要

本研究使用內容核心網路(Content-Centric Networking, CCNx)平台作為實驗對象,該平台之實作概念主要依據資料命名網路(Named Data Networking, NDN),在內容核心網路中,IP(Internet Protocol)位址被轉換成以命名為基礎的連線方式,並透過雲端運算的虛擬化技術建立完整的虛擬化測試平台,為未來的研究提供完整且可規模的測試環境。在本研究中,首先建置內容核心網路平台,並在虛擬機器底下的Linux作業系統以及Android作業系統中安裝內容核心網路提供之聊天(Chat)程式,測試基於資料命名網路之即時通訊(Instant Message, IM),最後,我們實現基於內容核心網路之資料命名網路,並詳細記錄實驗建置與驗證過程。

關鍵詞:內容核心網路;資料命名網路;雲端運算;虛擬化技術。

Abstract

In this research, the Content-Centric Networking (CCNx) platform is implemented and experimented, which is essential for Named Data Networking (NDN). In CCNx platform, the IP address is translated into NDN-based communication for a novel Internet architecture. Moreover, we utilize the cloud-based virtualization technology to virtualize our test bed, which can be scaled up. Firstly, we implement the CCNx platform and install the Chat source code in Linux and Android OS on some virtual machines. Then, the Instant Message (IM) is tested based on the NDN and be verified. Finally, the NDN based CCNx platform is realized, and the implement instructions and verification processes are recorded clearly.

Keywords: Content-Centric Networking; Named Data Networking; Cloud Computing; Virtualization Technology.

1. 前言

現今網際網路中提供許多應用服務,而各式各樣的傳輸訊息也透過網際網路分享到世界各地,這些訊息的傳輸技術或服務地分享方式需要仰賴不斷地研究與創新,無論是透過HTML(HyperText Markup Language)、VoIP(Voice over IP)、Web 2.0以及林林總總的即時通訊服務,都需要乘載於網際網路之上,才能讓這些技術廣泛地被應用並改善人類的生活。而當這樣的技術不斷地演進,網際網路本身的問題層出不窮,如使用者在查詢或分享資料時,習慣以查詢的資料名稱做為描述資料的資料(Metadata),而不是一連串沒有意義的數字,例如IPv4或IPv6位址。為解決上述之問題,有人提出網域名稱系統(Domain Name System, DNS)服務,但傳統集中式架構易造成單點錯誤與系統過載問題,因此,當熱門資訊需要重覆且快速地傳遞給每位使用者時,內容核心網路[1]的技術因應而生,充分地運用上述之技術才得以呈現出現今的網際網路發展。

雖然目前已有許多改良技術使網際網路更為良善,但有許多組織提出預測,未來的網路流量成長速度遠高於網路建設速度[2],因此許多學者認為需要提出一個更有效率的網路架構,才能夠因應未來遽增的網路流量,其中一種方法就是加入快取(Cache)機制以達到更有效的頻寬使用率,此外,為了確保資料內容的安全性與可信度,也加入密碼學中的數位簽章及加密機制等。綜合上述幾點特性,包括定址方式、路由機制、安全模型等,並搭配快取功能的概念,產生了新的網路架構稱之為資料命名網路[3]。

2. 背景介紹

近年來,由於網際網路的應用蓬勃發展,導致人們生活更依賴各種網路服務,不僅可以線上收聽音樂還可以觀看最新的電視節目以及直撥串流等,越來越豐富的多媒體服務如雨後春筍般不斷地

被提出，而智慧型手機的普及率更加速網路流量的劇增。

根據研究預測，未來的網路流量複合成長速度將遠大於網路基礎建設所能提供的頻寬，這將造成頻寬不足無法順利使用多媒體服務之困境。為解決上述之問題，必須從更有效率的傳輸架構著手，許多研究學者紛紛提出自己的想法與嶄新的概念，有學者提出了資訊核心網路 (Information-Centric Network, ICN) 或資料命名網路等想法，其目標是將目前以主機為基礎 (Host-based) 的網路轉換成以資訊為基礎 (Information-based) 的網路；此外，再搭配快取機制，所謂的快取機制是在網路結點中加入可以暫時儲存資料的緩衝記憶體 (Buffer)，如此一來，將無須針對應用程式配置額外的設定，且與傳輸的內容無關，再加上良好的替換演算法，就可以節省並降低網路頻寬的使用量，這些特點在目前的網際網路中是前所未見的。

因為新的網路架構已從主機為基礎的網路轉換至資訊為基礎的網路，這代表驗證資料本身的完整性變的格外重要，因為相同的一份資料可能由多個來源共同提供，使用者需要能夠自己驗證資料的正確性及可靠性，接下來我們將介紹目前資料命名網路的主流計劃及其特色。

2.1 ANR Connect

ANR Connect 研究計畫在 2011 年於法國巴黎發起[4]，其中包含六個主要的任務，依據不同的研究目標和工作內容，分別從任務 0 到任務 6，由產業界和學術界的研究人員共同組成的團隊執行。該計劃基於 CCNx 提供之開放原始碼版本，並額外提出三項主要的功能，分別為流量控制、命名和路由轉送機制與流量管理，以及快取記憶體機制等研究，強調其最終目標是基於經濟與佈署策略考量後，獲得更好的實驗結果與效能數據，並將其結果回饋給 CCNx 計劃。

2.2 CCNx

由美國所發起的開放原始碼計畫[5]，因早年 IP 網路的資料交換介於主機之間，但這樣的方式與現今大眾網路使用風格迥異，使用者查詢火車時刻表資訊時，並不會特別記憶或追溯資料是儲存在哪台電腦主機，雖然目前已有許多技術可協助和支援轉換，但對網路系統的整體效能都是額外的開銷。CCNx 計畫強調將目前網路的使用方式從資料定址及主機位置的「Where」，轉變成使用者所需資料的「What」，這樣的轉變將更貼近實際的使用需求，然而，要達成上述之轉變，需要從網路架構上著手，如圖 1 所示，我們將說明目前的 IP 網路和內容核心網路以 Content Chunks 為資料定義上的差異，以及在內容核心網路中，另外新增的安全 (Security) 層及策略 (Strategy) 層。

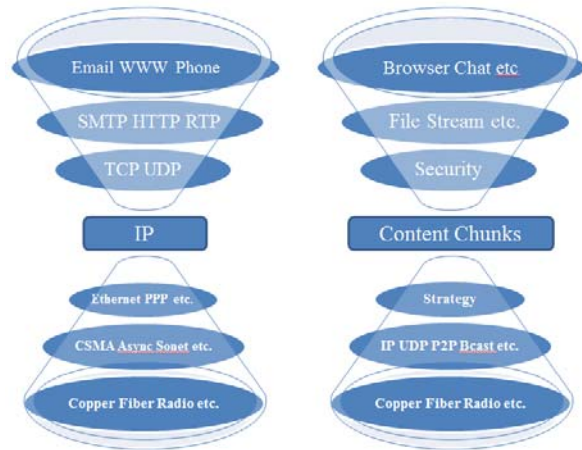


圖 1 IP 網路對應內容核心網路之各層協定

在內容核心網路中，資料的取得已經不需要與第三層協議識別如 IP 和第二層協議識別做綁定，取而代之的是策略層做為選擇資料來源最有效的選取方式，又因為現今的每個裝置都可能有一個以上的網路連接方式，如 Wi-Fi、藍芽 (Bluetooth)、乙太網路 (Ethernet) 等，抑或是同類型的鏈結來源但是有多重連線等，透過策略層的協助即可避免掉常見的迴路問題。

當資料有多個來源時，必需要透過安全層所提供之安全驗證機制來確保資料本身的完整性及可靠度，以確保資料未遭受竄改，目前在內容核心網路中尚未明確定義數位簽章機制及加密機制，但是已有提出參考模型。

2.3 DONA

此計劃的全名為 Data-Oriented (and Beyond) Network Architecture，在 2007 年由柏克萊 (Berkeley) 大學的研究人員提出[6]。在 DONA 計畫中，架構層疊於 OSI 七層的第三層網路層之上，架構中提出解析處理者 (Resolution Handlers, RH)，其扮演角色如同現今網際網路中的 DNS 伺服器，若需要發布資料，使用者需要向解析處理者進行註冊動作，當使用者要求資料時，需先執行 Find(P:L) 向解析處理者請求服務，如圖 2 所示，每個註冊者 (Register) 都有一個 TTL (Time To Live) 值，確保資源的時效性，並運用 Anycast 傳送模式做為選擇解析處理者的主要機制。DONA 具備與目前網路裝置相容性，例如防火牆 (Firewall) 的過濾機制，其原理是透過政策 (Policy) 設定來決定 Register 和 Find 命令的行為，過濾不正確或是惡意的訊息資訊；此外，為了增加且善用頻寬的使用率，DONA 亦導入快取機制、訂閱 (Subscribe) 機制以及避免設備發生異常或是錯誤設定等，最後在個人電腦或伺服器主機的 Linux 作業系統上實作，以展現系統運作成果。

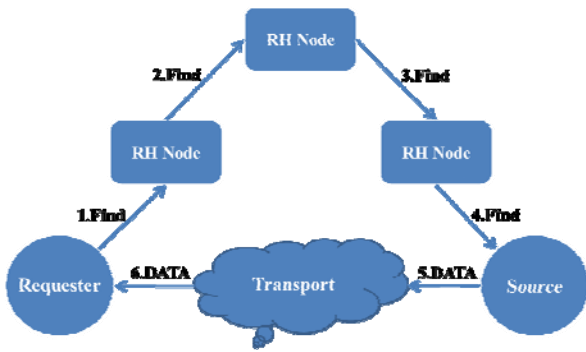


圖 2 DONA 網路傳輸流程

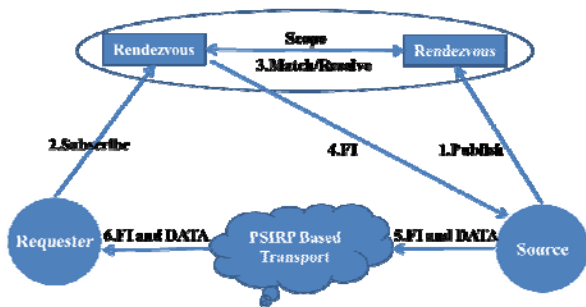


圖 3 PSIRP 網路傳輸流程

2.4 PSIRP

PSIRP 是由歐洲學術界與產業界會員所組成的計畫，其全名為 Publish-Subscribe Internet Routing Paradigm [7]。在 PSIRP 中，將最小的資料單位稱為具名資料物件(Named Data Objects, NDO)，而資料的發布及訂閱透過特定的媒合系統(Rendezvous System)，當訂閱者提出具名資料物件要求時，其中包括範圍識別碼(Scope Identifier, SI)和媒合識別碼(Rendezvous Identifier, RI)，當媒合機制找出符合的具名資料物件後，將傳送一個轉送識別碼(Forwarding Identifier, FI)給資料來源端，透過轉送識別碼的內容可以識別出訂閱者的請求，最後，將資料傳送給訂閱者，其 PSIRP 的整體傳輸流程如圖 3 所示。

PSIRP 計畫分別在不同作業系統底下皆有釋出開放原始碼，其名稱稍微有點不同，例如在 Linux 作業系統的名稱為 Blackadder，而在 FreeBSD 作業系統則稱之為 Blackhawk。目前都處理前期的實驗階段，並且已利用雲端運算中的虛擬化技術，在虛擬主機上建置小規模的測試平台，是目前各計畫中進度較快的。

2.5 SAIL

SAIL(Scalable and Adaptive Internet Solutions)是由歐洲發起大型研究計畫，其目標為尋求未來的創新網路架構，其中不僅僅包含技術層面的考量，

管理層面的考量也在研究範圍中，並將計畫再細分三個子計畫 [8]，分別為 Netinf (Network of Information)、Open Connectivity Services 以及 Cloud Networking，本研究專注於 Netinf 所研究的命名、定址、路由、名稱解析、資源管理及安全隱私等網路議題，其特點在於具名資料物件可以透過名稱解析(Name Resolution)或是基於命名路由(Name-based Routing)兩種方式取得，以適應各種使用環境需求，而具名資料物件的註冊及位置綁定則是透過名稱解析服務(Name Resolution Service, NRS)或是透過路由協定發布路由訊息，Netinf 節點本身可以暫存具名資料物件資料，一旦使用者提出請求時，可以透過最佳來源取得具名資料物件的副本，再者，如果在較小型的網路環境中可以免去建立名稱解析服務，直接透過路由協定交換具名資料物件路由訊息即可。

3. 內容核心網路架構

為了能夠以嶄新的方式達到資料命名網路的精神，在內容核心網路中，提出了兩種全新的封包(Packet)格式，分別為興趣封包(Interest Packet)與資料封包(Data Packet)，其各自的封包欄位與組成如圖 4 所示。在興趣封包中，包含了內容名稱(Content Name)、資料範圍(Selector)以及臨時(Nonce)等三個主要欄位，主要用途是將需要的資料範圍及其定義(Name)傳送出去，透過標準的內容核心網路結點協助轉發；而另一種則是資料封包，封包欄位包含內容名稱、特徵(Signature)、簽署資訊(Signed Info)、資料(Data)，其用途是將符合興趣封包所要求的資料回傳給要求者(Requester)，並依據其特徵、簽署資訊等欄位內容，加以驗證資料本身的正確性及可靠性。

內容暫存(Content Store)的用途如同現今網路設備的緩衝記憶體(Buffer Memory)，主要用於存放曾經轉送過的內容，相較於目前的 IP 網路，設備轉送封包後不會保存任何資料，在內容核心網路中透過演算法的計算，盡可能留下可重覆使用的資料，如當日的熱門新聞和資訊，當不同的使用者要求相同的內容時，即可節省內容提供者(Content Provider)下載相同資料的頻寬，使用者可以透過特徵及簽署資訊確保資料無被竄改，另外，提高擊中率(Hit Ratio)的替換演算法及內容暫存的配置大小及方法在內容核心網路中亦是個重要的研究議題。



圖 4 內容核心網路封包格式

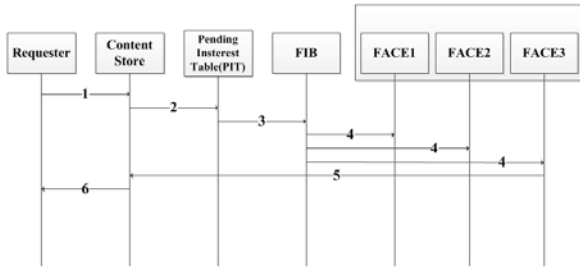


圖 5 內容核心網路封包轉送流程圖

轉送資訊資料庫(Forwarding Information Base, FIB)的用途與 IP 路由器內的轉送資訊資料庫相同，用於查詢潛在的資料來源，因為內容核心網路本身架構可以避免迴圈問題，故不受限於生成樹(Spanning Tree)演算法之限制，允許同時向多個來源請求資料。

PIT(Pending Interest Table)用於追蹤尚未被滿足的興趣封包資料及其來源，就像是童話故事糖果屋裡的麵包屑，當收到符合的資料封包時，內容核心網路就會依據 PIT 的記錄內容，將資料封包轉送至興趣封包的來源，並將 PIT 中此筆對應的資料刪去，如果在定義的時間內都沒有接收到符合的資料封包時，PIT 也會將此筆資料刪除避免浪費表格空間，此時若使用者依舊需要這份資料，則需要額外再發出一個興趣封包的請求。

在介紹完三個主要元件後，我們以圖 5 的信令圖為範例，說明興趣封包和資料封包在內容核心網路中的轉送流程：

- 1) 請求者發出興趣封包至下一個內容核心網路的節點。
- 2) 收到興趣封包後，檢查目前是否有符合的資料封包，如果有，則立即回覆給請求者，如果沒有，則再往下一步處理。
- 3) 在 PIT 中是否有相同的興趣封包記錄，如果有，即無須再次記錄，如果沒有，則記錄下來並往下一步處理。
- 4) 尋找潛在的 Face，並透過這些 Face 將興趣封包送出，並等待回覆。
- 5) 從 Face3 收到符合的資料封包後，透過替換演算法確認該筆資料，並放入內容暫存，再將過時的資料丟棄。
- 6) 根據 PIT 的記錄內容，確認需要此資料封包的請求者，將資料正確地傳送給請求者，刪除在 PIT 中的記錄。

4. CCNx 平台建置

目前在各相關計畫中以 CCNx 計畫的進度最快，CCNx 已提供下載開放原始碼，且在美國已建置完成測試平台，主要用途除了實現和驗證內容核

心網路的概念之外，更納入了基於 OSPFN(Open Shortest Path First for named data Networking)的路由研究。

4.1 安裝及編譯

為了能夠正確地完成編譯，本研究使用 CCNx 0.7.0(於 2012 年 12 月 8 號釋出，目前最新版本為 0.8.0rc1)並使用官方建議之作業系統 ubuntu-12.04.1-desktop[9]為測試環境，搭配的 JAVA 編譯環境為 JDK SE 6 Update 38，而在 Android 平台中，我們使用的版本號為 4.03[10]。本研究團隊曾因使用 OpenJDK 做為 JAVA 平台編譯環境，遭遇 Android 平台 APP 編譯正常，但在執行時立即出現錯誤後退出，經過漫長的除錯過程並發信詢問 CCNx 開發團隊，在數次信件往來後才確認為 OpenJDK 導致的異常問題，最後換回 Oracle JDK 重新編譯才解決這個棘手的難題。

4.2 Linux 平台及 Android 平台運作測試

CCNx 目前皆使用 JAVA 做為執行環境，在正確安裝及編譯後，還需要 JVM 才能夠順利執行，故在執行前需先設定環境變數，使程式順利運作。因為 CCNx 目前還在實驗階段，至今尚未有作業系統將其納入為網路堆疊的一部份，故在執行 CCNx 應用程式前，都需先啟用 CCNx Daemon 為系統提供 API 呼叫，而 CCNx Daemon 雖可以不需要依附在 TCP/IP 協定之上，但為了能更快速地驗證 CCNx 的架構平台，目前皆使用層疊(Overlay)的方式運作 CCNx，並以 UDP 做為其傳輸層協定。在開始說明實驗流程前，必須先說明，為保障實驗環境避免遭受惡意攻擊行為，我們移除實驗環境中有關設備的部分資訊，如系統網址(Uniform Resource Locator, URL)與 IP 等資訊。

為了能夠清楚地瞭解目前 CCNx Daemon 的運作情況，啟動 CCNx Daemon 服務後，系統會啟動一個後台的網頁伺服器(Web Server)並監聽 9695 埠號(Port)，我們使用任意一種網頁瀏覽器，輸入對應的網站 IP 位址並指明 9695 埠號，即可清楚地看到目前系統中所有 Face 的狀態，如圖 6 中所示。其中包括目前本地 Face 使用的是 IPv4 位址或是 IPv6 位址、該 Face 所接收和傳送的資料量、興趣封包的總數等相關資訊。

在 Android 平台中，同樣地需要先啟動 CCNx 的服務，CCNx 的服務控制介面如圖 7 中所示，其中可以看到目前網卡 eth0 使用的是 IPv6，以及相關路徑、除錯等設定。接著，下來同前面所述開啟 Android 平台上之瀏覽器，即可看到目前 CCNx 平台的運作情況，如圖 8 所示，與 Ubuntu 作業系統相同，包含 Face 0 到 Face 5 的狀態、傳送與接收的資料量、以及轉傳的興趣封包和資料封包等相關資訊。

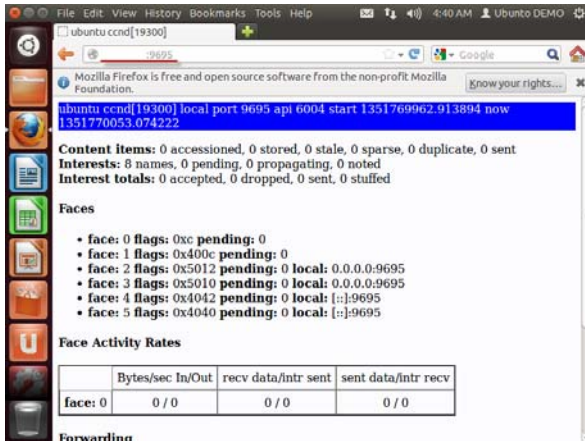


圖 6 CCNx 系統中各 Face 的狀態

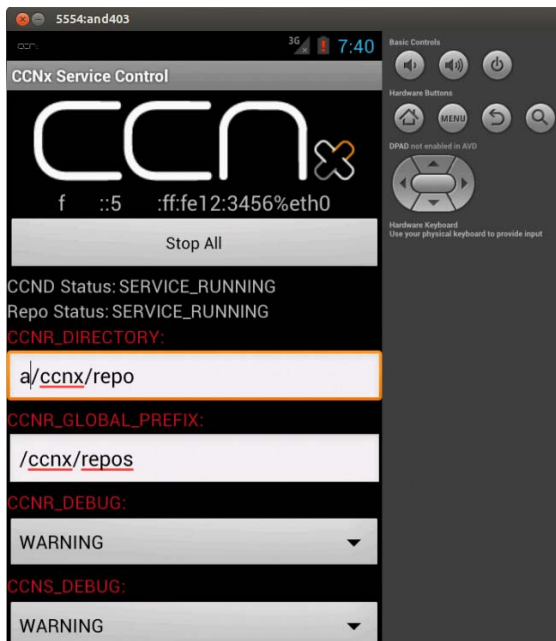


圖 7 Android 平台中 CCNx Service 控制程式



圖 8 CCNx Android 平台中各 Face 的狀態

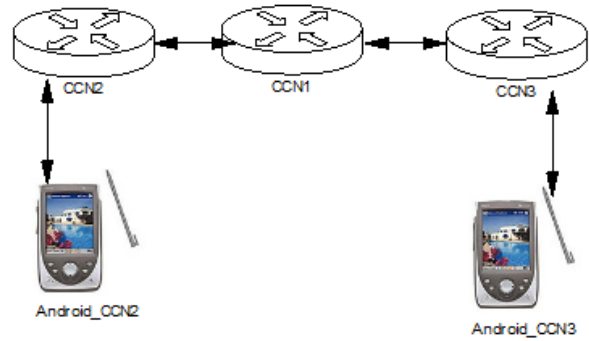


圖 9 多 CCN Node 互連測試拓模

4.3 多個 CCNx Node 互連驗證

在本節中，我們測試多個 CCNx 節點互連驗證，其實驗環境如圖 9 所示，為了建置下列的網路拓模(Network Topology)進行互連測試，需使用以下的硬體設備：

- 3 部電腦，且在 TCP/IP 架構下預先測試能否互相通訊，同時需先安裝 CCNx 服務
- 2 部 Android 手機，具備 CCNx 服務及 Chat 程式(本實驗未使用實體手機，而是分別在 CCN2 和 CCN3 使用模擬器執行)

由於目前內容核心網路尚未採納任何一種明確的路由協定，故每個 CCN 節點需要各別手動建立靜態路由資訊，以下為每個 CCN 節點需要輸入的指令，以建立 CCN 連線。

- Node CCN1 執行指令：

- 1) ccnd
- 2) ccndc add / udp [CCN2 的 IP]
- 3) ccndc add / udp [CCN3 的 IP]

- Node CCN2 執行指令：

- 1) ccnd
- 2) ccndc add / udp [CCN1 的 IP]
- 3) adb forward tcp:9699 tcp:9695
- 4) ccndc add ccnx:/ tcp 127.0.0.1 9699

- Node CCN3 執行指令：

- 1) ccnd
- 2) ccndc add / udp [CCN1 的 IP]
- 3) adb forward tcp:9699 tcp:9695
- 4) ccndc add ccnx:/ tcp 127.0.0.1 9699

- Node Android_CCN2 執行指令：

- 1) 執行 CCNx Service Control
- 2) 執行 CCNxChat 程式 Remote Host 輸入 CCN2 的 IP

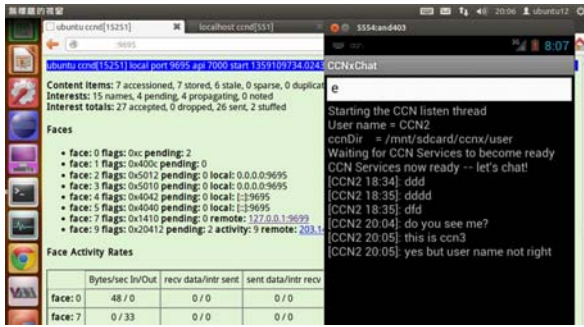


圖 10 Android 平台的 CCN2 運作畫面

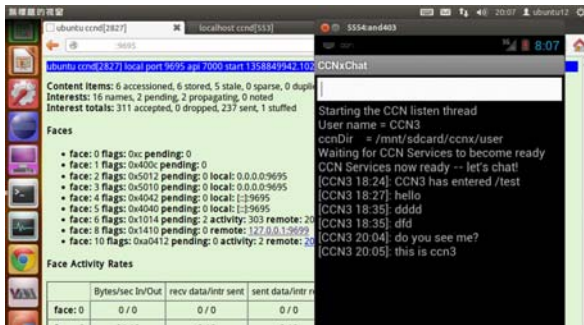


圖 11 Android 平台的 CCN3 運作畫面

- Node Android_CCN3 執行指令：
 - 1) 執行 CCNx Service Control
 - 2) 執行 CCNxChat 程式 Remote Host 輸入 CCN3 的 IP

執行完上述指令後，即可以透過 CCNx 架構達成即時通訊，如圖 10 和圖 11 所示，即為 Android 平台透過 CCNx 的 Chat 程式交談畫面，圖 10 和圖 11 的左半邊分別為節點 CCN2 和 CCN3 的執行即時狀態表，可以看到目前是否有送出訊息或著是接收到訊息，當有異常錯誤發生時，在各節點狀態表中可以查看異常資訊內容，藉此做為除錯依據；而在圖 10 和圖 11 的右半邊為兩個 CCN 節點的對話視窗，系統會自動通知使用者名稱為何，如圖 10 為節點 CCN2 以及圖 11 為節點 CCN3，接著會等待 CCN 服務是否備妥(Ready)，並在服務順利開啟後，由系統自行通知 CCN 節點，目前已經可以透過 CCNx 系統開始進行即時通訊。

5. 結論

本研究透過 CCNx 計畫提供之開放原始碼，建置小規模的測試環境，並且使用雲端運算的虛擬化技術，將測試平台虛擬化，達到可規模的虛擬化測試平台。再者，我們在 Linux 和 Android 平台中編譯 CCNx，使用 CCNx 提供之 Chat 程式進行即時通訊服務，並利用內建的驗證工具進行服務驗證，提供完整的建置過程與經驗分享，以提供國內相關研

究的知識基礎並加速內容核心網路以及資料命名網路之研究。

然而，目前的資料命名網路與內容核心網路仍是基於 IP 網路所建置，故許多 IP 網路之限制依然存在，尚未完全實現資料命名網路的精神。在未來，我們將針對資料命名網路之內容暫存特性和資料驗證特性進行探討，冀望建置效能更好且可靠的資料命名網路服務。

致謝

本研究感謝國科會 102 年度「自我最佳化平行流量實現高能源效率綠能雲端計算」(計畫編號 101-2221-E-197-008-MY3)與教育部 102 年度網通前瞻技術教學推動聯盟中心計畫「雲端服務平台開發」的資助。

參考文獻

- [1] B. Ahlgren, C. Dannowitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A Survey of Information-Centric Networking," IEEE Communications Magazine, vol. 50, no. 7, pp. 26-36, July 2012.
- [2] M. F. Bari, S. R. Chowdhury, R. Ahmed, R. Boutaba, and B. Mathieu, "A Survey of Naming and Routing in Information-Centric Networks," IEEE Communications Magazine, vol. 50, vol. 12, pp. 44-53, December 2012.
- [3] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking Named Content," in Proceeding of the International Conference on Emerging Networking Experiments and Technologies (CoNEXT '09), Rome, Italy, December 1-4, 2009, pp. 1-12.
- [4] ANR Connect. 2013, August 1. Available: <http://www.anr-connect.org/>
- [5] Project CCNx. 2013, August 1. Available: <http://www.ccnx.org>
- [6] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A Data-Oriented (and Beyond) Network Architecture," in Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '07), Kyoto, Japan, August 27-31, 2007, pp. 181-192.
- [7] Publish-Subscribe Internet Routing Paradigm. 2013, August 1. Available: <http://www.psrp.org/>
- [8] Scalable and Adaptive Internet Solutions. 2013, August 1. Available: <http://www.sail-project.eu/>
- [9] Android Developer. 2013, August 1. Available: <http://developer.android.com/>
- [10] Ubuntu Linux. 2013, August 1. Available: <http://www.ubuntu.com/>