STUDIA INFORMATICA 2011
Volume 32 Number 3A (98)

Sławomir PRZYŁUCKI, Daniel SAWICKI
Politechnika Lubelska, Katedra Elektroniki

ŚRODOWISKO TESTOWE DO BADANIA SIECI CHMUROWYCH

Streszczenie. Artykuł prezentuje środowisko testowe do szybkiego uruchamiania systemów chmur sieciowych. Główny nacisk położony jest na minimalizację niezbędnych zasobów sprzętowych, przy jednoczesnym zachowaniu łatwości rozbudowy systemu. Środowisko testowe zapewnia zachowanie wszystkich cech systemu Eucalyptus, który stanowi podstawę proponowanego rozwiązania. Wykorzystanie technik wirtualizacji umożliwiło zbudowanie całej testowej chmury na pojedynczym systemie komputerowym.

Słowa kluczowe: sieci chmurowe, Eucalyptus, wirtualizacja

CLOUD COMPUTING TESTBED

Summary. The article presents the testbed for fast prototyping of the cloud computing structures. The main aim of our proposition is the minimalization of the required hardware resources as well as an ease of system expansion. The testbed based on the Eucalyptus packet and it preserves all its specific features.

Keywords: cloud computing, Eucalyptus, virtualization

1. Klasyfikacja sieci chmurowych

Zgodnie z definicją, opracowaną przez NIST (ang. *The National Institute of Standards and Technology*) przetwarzanie danych w sieciach chmurowych oznacza model usługi, umożliwiającej użytkownikom wygodny dostęp na żądanie do wspólnej puli zasobów obliczeniowych (np. serwerów, pamięci masowych, aplikacji i usług), przy czym obsługa tych żądań jest realizowana przy możliwie minimalnej ingerencji dostawcy usługi lub operatora sieciowego. Klasyfikacje usług w chmurze opiera się na specyfice zasobów lub oprogramowania dostar-

czanego końcowym użytkownikom i w najczęściej spotykanym podejściu obejmuje trzy podstawowe klasy [1]:

- infrastruktura jako usługa (ang. *Infrastructure as Service* – IaaS) – oznacza środowisko sieciowe, optymalizowane pod kątem wirtualizacji zasobów,
- środowisko programowe jako usługa (ang. *Platform as Service* – PaaS) – oznacza usługę polegającą na udostępnieniu wirtualnego środowiska pracy programistom,
- oprogramowanie jako usługa (ang. *Software as Service* – SaaS) – oznacza dostęp do dedykowanych aplikacji, uruchamianych pod nadzorem dostawcy usługi.

Konfiguracja i wdrożenie sieci chmurowych mogą natomiast być realizowane w trzech podstawowych konfiguracjach. Konfiguracje te definiują zasady dostępu i kontroli korzystania z usług w odniesieniu do zasobów, z których one korzystają (zarówno wirtualnych, jak i fizycznych). W tym kontekście można wyróżnić trzy podstawowe kategorie sieci chmurowych [4].

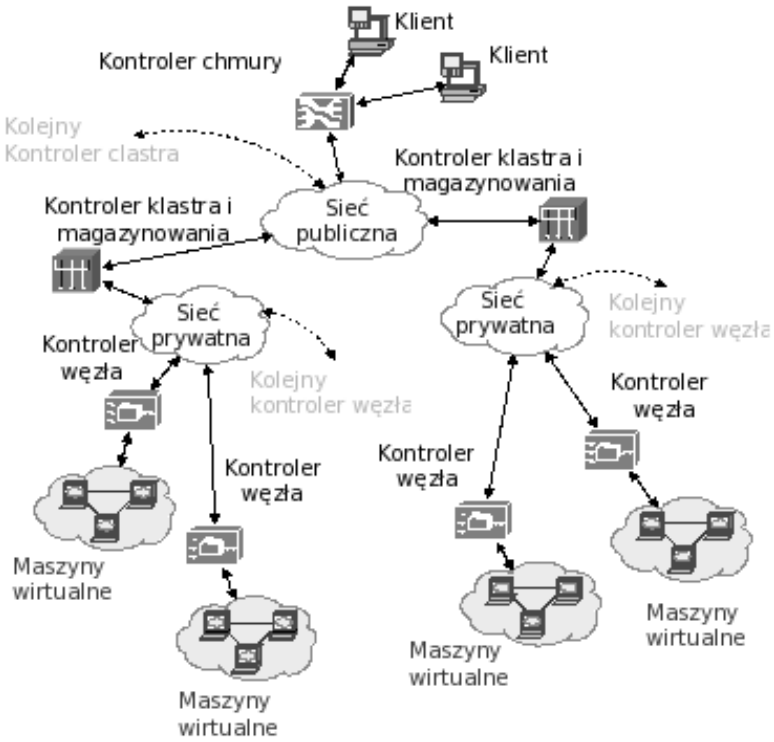
1. Chmury publiczne (ang. *public clouds*) – zapewniają dostęp do usług i powiązanych z nimi zasobów dla wszystkich użytkowników końcowych, a korzystanie tych usług odbywa się za pośrednictwem sieci rozległych, takich jak Internet. Dostawcy usług w chmurach publicznych udostępniają użytkownikom narzędzia konfiguracji tych usług, najczęściej opierając się na serwisach web. Z kolei użytkownicy, za pomocą tych narzędzi, wykupują usługi na zasadzie wynajęcia zasobów.
2. Chmury prywatne (ang. *private clouds*) – zapewniają użytkownikom dostęp do zasobów zgromadzonych wewnątrz infrastruktury sieciowej i programowej danej organizacji. Sposób zarządzania dostępem i korzystania z zasobów jest analogiczny do przypadku chmur publicznych. Jednak ze względu na to, że zasoby są skupione wewnątrz infrastruktury jednej organizacji, to możliwe jest zapewnienie wyższych lub niestandardowych wymogów odnośnie przetwarzania i przechowywania danych. W przypadku integracji kilku chmur prywatnych, określa się je mianem chmur społecznościowych (ang. *community cloud*).
3. Chmury hybrydowe (ang. *hybrid cloud*) – są połączeniem obu, wymienionych wyżej, typów w jeden system przetwarzania danych. Obecnie zdobywają one dużą popularność, ze względu możliwości przeniesienia do chmur publicznych tylko wybranych zadań. Rosnąca liczba rozwiązań chmur publicznych prowadzi do coraz częściej spotykanej struktury chmur połączonych (ang. *combined cloud*).

Biorąc pod uwagę wymienione wyżej kategorie sieci chmurowych, łatwo zauważyć, iż istotną i występującą we wszystkich przypadkach kategorią są chmury prywatne. W związku z tym obecnie rośnie zainteresowanie metodami szybkiego i odpowiednio elastycznego tworzenia środowisk testowych dla chmur prywatnych. Jednocześnie wymieniony we wstępie

podział ze względu na typ usług, jasno wskazuje, iż fundamentem wszystkich prac w obszarze przetwarzania chmurowego są usługi IaaS. Z tego też powodu rozwiązanie, opisane w dalszej części artykułu, poświęcone jest środowisku testowemu IaaS w strukturach chmur prywatnych.

2. System Eucalyptus

Architektura systemu Eucalyptus pozwala na budowę struktur chmur prywatnych, zgodnych z szeroko akceptowanym standardem Amazon EC2. Oznacza to, że użytkownicy tego systemu wykorzystują zasoby chmury w ten sam sposób, jak zasoby chmury publicznej, oferowanej przez Amazon. Pozwala to na prostą integrację tego rozwiązania w ramach projektów chmur hybrydowych [3]. Struktura systemu Eucalyptus przedstawiona jest na rys. 1 i składa się z czterech elementów: kontrolera węzła NC (ang. *Node Controller*), kontrolera clustra CC (ang. *Cluster Controller*), kontrolera magazynowania Walrus (ang. *Storage Controller*) oraz kontrolera chmury CLC (ang. *Cloud Controller*).



Rys. 1. Struktura sieci chmurowej opartej na systemie Eucalyptus
Fig. 1. Hierarchical structure of Eucalyptus

Kontroler węzła jest zasobem fizycznym (zazwyczaj pojedynczym hostem), na którym uruchamiane są poszczególne instancje maszyn wirtualnych. W najprostszym wypadku wystarczy jeden taki kontroler, ponieważ, w zależności od posiadanych zasobów sprzętowych, może on być podstawą do uruchomienia jednej lub więcej instancji maszyn wirtualnych.

Zbiór NC połączony jest w ramach sieci prywatnej, zarządzanej przez kontroler klastra CC. Kontroler CC odpowiedzialny jest za trzy podstawowe funkcje: planowanie przydziału zasobów NC dla poszczególnych zadań, nadzór nad instancjami maszyn wirtualnych oraz zbieranie i raportowanie informacji o stanie realizacji zadań i stopniu wykorzystania zasobów. Ostatnie z tych zadań pozwala na ocenę możliwości realizacji poszczególnych zadań i informowanie o tym kontrolera chmury CLC. Razem z CLC działa kontroler magazynowania Walrus. Dostarcza on usługę przechowywania danych (obrazów maszyn wirtualnych, a także danych użytkowników), zgodną z interfejsem Amazon's S3. W każdym systemie Eucalyptus musi być uruchomiony jeden kontroler CLC i jeden Walrus. Kontroler CLC jest odpowiedzialny za cały proces obsługi żądania dostępu do zasobów zgłaszanego przez użytkownika (autentykację, kontrolę przestrzegania reguł SLA (ang. *Service Level Agreement*), monitoring sesji), jak również za zadania planowania na poziomie klastra przydziału zasobów do zgłoszonych żądań. Innymi słowy, CLC jest pomostem pomiędzy narzędziami udostępnionymi użytkownikowi a wewnętrzną strukturą planowania, przydziału i realizacji zadań, zgłoszonych przez użytkowników [3, 5].

2.1. Cechy systemu Eucalyptus

Eucalyptus zapewnia infrastrukturę sieciową, na której możliwe jest tymczasowe uruchamianie zasobów wirtualnych w postaci grupy maszyn wirtualnych VM (ang. *Virtual Machines*). Grupy VM lokowane są w wydzielonych podsieciach prywatnych, odseparowanych od ruchu w sieci prywatnej, łączącej poszczególne NC klastra [5].

Eucalyptus może zostać skonfigurowany w jednym z czterech trybów sieciowych (ang. *networking modes*): *managed*, *managed-noVLAN*, *static* oraz *system*. Poszczególne tryby określają sposób organizacji infrastruktury IaaS i definiują m.in. wykorzystanie przedstawionych poniżej, narzędzi wewnętrznych systemu Eucalyptus [5].

1. Sterowanie IP (ang. *IP Control*) – rozwiązanie dostępne we wszystkich trybach, z wyjątkiem trybu *system*. Dzięki temu mechanizmowi Eucalyptus przypisuje automatycznie adresy IP do poszczególnych VM. Natomiast w trybie *system* należy zapewnić zewnętrzny serwer DHCP (ang. *Dynamic Host Configuration Protocol*).
2. Grupy bezpieczeństwa (ang. *Security Groups*) – są to zestawy zasad, sieciowych, jakie stosowane są do wszystkich VM w ramach danej grupy. W podstawowej konfiguracji określają one zasady dostępu do poszczególnych VM, a tym samym do określonych zasobów. Grupy bezpieczeństwa mogą być stosowane w trybach *managed* oraz *managed-noVLAN*.
3. Elastyczne IP (ang. *Elastic IP*) – ta cecha pozwala na powiązania publicznego adresu użytkownika z adresami VM, w ramach danej usługi. Elastyczne IP są dostępne wyłącznie w trybach *managed* oraz *managed-noVLAN*.

4. Izolacja maszym virtualnych (ang. *VM Isolation*) – narzędzie to pozwala na wymuszenie izolacji ruchu pomiędzu grupami bezpieczeństwa, bez konieczności umieszczania ich w różnych podsieciach. Narzędzie to jest dostępne wyłącznie w trybie managed.

Środowisko testowe IaaS dla chmury prywatnej powinno dawać możliwość wykorzystania wszystkich cech i zalet pakietu Eucalyptus. Na podstawie przytoczonych wyżej informacji powinno ono zawierać wszystkie komponenty systemu, przy jednoczesnej możliwości szybkiej i prostej rozbudowy. Powinno też umożliwiać wykorzystanie wszystkich narzędzi i opcji konfiguracyjnych, dostępnych w najbardziej rozbudowanym trybie *managed*. Takie założenia spełnia proponowane środowisko testowe.

3. Struktura środowiska testowego

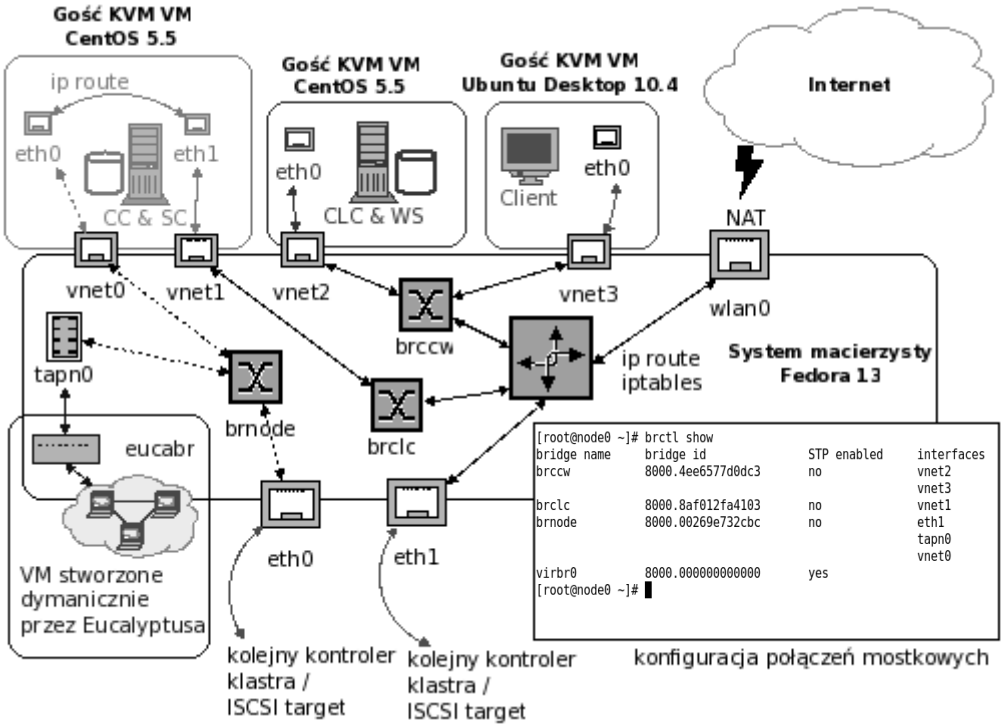
Na podstawie dokonanego przeglądu literatury, a także opisów autorów systemu Eucalyptus, realizacja pełnej struktury chmury prywatnej, pracującej w trybie *managed*, wymaga zastosowania przynajmniej dwóch maszyn fizycznych, jednej dla potrzeb NC oraz drugiej, na której uruchomione zostałyby pozostałe komponenty, tj. kontrolery CC, CLC oraz Walrus. Proponowane środowisko testowe omija konieczność stosowania wielu maszyn i dzięki intensywnemu wykorzystaniu dostępnych mechanizmów wirtualizacji zasobów pozwala uruchamiać proste struktury IaaS na pojedynczej maszynie, a jednocześnie rozbudowywać je w razie potrzeby do wymaganej wielkości, bez konieczności konfigurowania wszystkich elementów od podstaw. Wykorzystana platforma sprzętowa jest powszechnie dostępna na komputerach klasy desktop, a nawet laptop, więc dodatkową cechą proponowanego rozwiązania jest jego ekonomiczność.

Podstawą do budowy środowiska testowego był komputer PC, wyposażony w czterordzeniowy procesor firmy Intel ze wsparciem dla sprzętowej wirtualizacji, 8 GB pamięci RAM, pamięć dyskową o pojemności 500GB oraz 3 interfejsy sieciowe. Zasoby te zostały, z wykorzystaniem wirtualizacji opartej na hypervisorze KVM, przypisane do elementów struktury systemy Eucalyptus w następujący sposób:

- kontroler węzła (NC) – gospodarz KVM (4 rdzenie, 8 procesorów logicznych, współdzielonych), 4 GB RAM (współdzielona z gośćmi), 60 GB HDD SATA, system Fedora 14 (x86_64),
- kontroler klastra oraz kontroler magazynu (CC & SC) – gość KVM (4 rdzenie, 8 procesorów logicznych, współdzielonych), 1 GB RAM, 40 GB HDD SATA (sterownik virtio), system CentOS5.5 (x86_64),

- kontroler chmury oraz Walrus (CLC & WS) – gość KVM (4 rdzenie, 8 procesorów logicznych, współdzielonych), 1 GB RAM, 40 GB HDD SATA (sterownik virtio), system CentOS5.5 (x86_64),
- host użytkownika – gość KVM (1 rdzeń, 2 procesory logiczne, współdzielone), 512 MB RAM, 10 GB HDD SATA (sterownik virtio), system Ubuntu 10.4 (x86_32).

Wewnętrzne połączenia zostały zrealizowane poprzez trzy mosty o nazwach, odpowiednio: brnode, brclc oraz brccw. Architektura środowiska testowego została przedstawiona na rys. 2, natomiast konfiguracja interfejsów i elementów sieciowych środowiska testowego przedstawiona jest w tabeli 1.



Rys. 2. Struktura chmury IaaS na bazie systemu Eucalyptus (w prawym dolnym rogu przedstawiona jest konfiguracja mostów łączących wszystkie elementy chmury)
Fig. 2. The structure of proposed IaaS testbed based on Eucalyptus system

Tabela 1

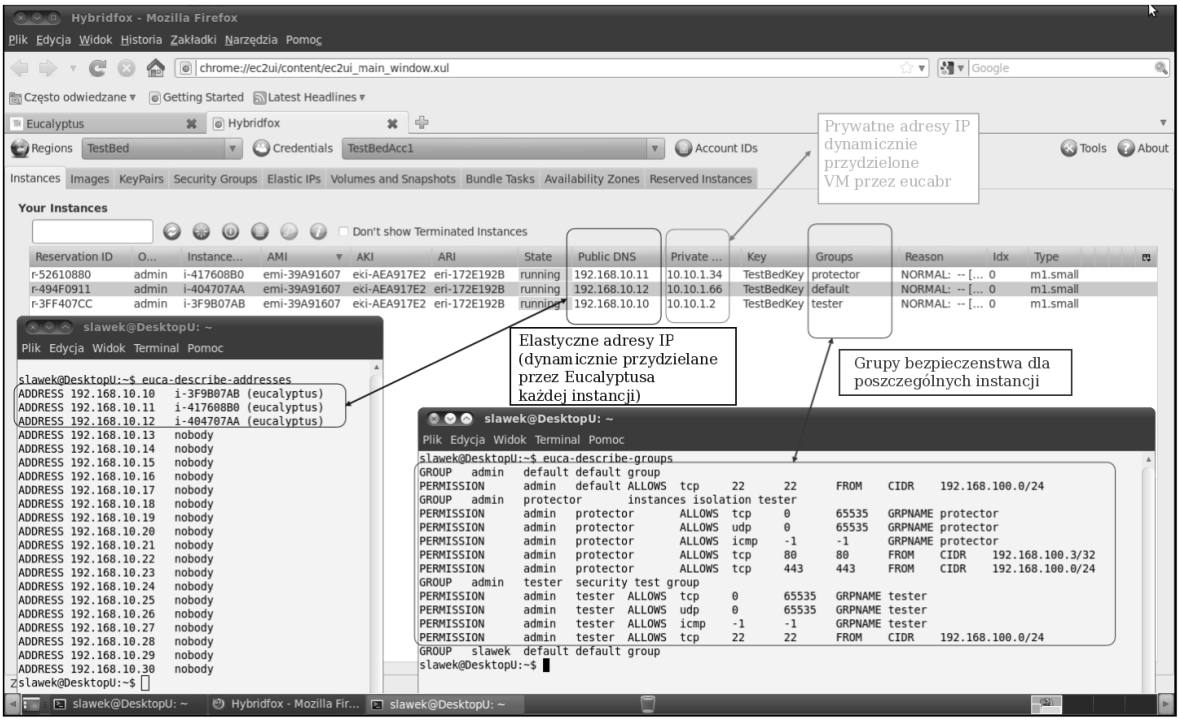
Ustawienia sieciowe w elementów składowych środowiska testowego IaaS

Parametr	Kontroler klastra i chmury (CC oraz CLC)	Kontroler węzła (NC)
VNET_PUBINTERFACES	eth1 (vnet1)	tapn0
VNET_PRIVINTERFACES	eth0 (vnet0)	tapn0
VNET_SUBNET	10.0.10.0	
VNET_NETMASK	255.255.255.0	
VNET_ADDRSPERNET	32	
VNET_PUBLICIPS	od 192.168.10.10 do 192.168.10.100	
VNET_DNS	192.168.100.3	
VNET_DHCPDEAMON	/usr/bin/dhcpd	
VNET_DHCPUSER	dhcpd	

4. Weryfikacja działania

Przedstawione poniżej wyniki weryfikacji działania zaproponowanego środowiska testowego mają na celu potwierdzenie pełnej funkcjonalności rozwiązania w kontekście przedstawionych cech chmur prywatnych i pakietu Eucalyptus, tj. grup bezpieczeństwa, elastycznego IP oraz izolacji VM. Do celów zarządzania systemem testowym wykorzystano dwa narzędzia programowe: graficzne w postaci Hybridfox [7] oraz program z interfejsem tekstowym euca_tools. Oba były zainstalowane na systemie gościnnym, którym był Ubuntu Desktop 10. Inni klienci mogą być podłączeni do systemu poprzez interfejs wlan0. Zewnętrzny klient jest niezbędny w przypadku konieczności przygotowania własnego obrazu z udostępnianym oprogramowaniem.

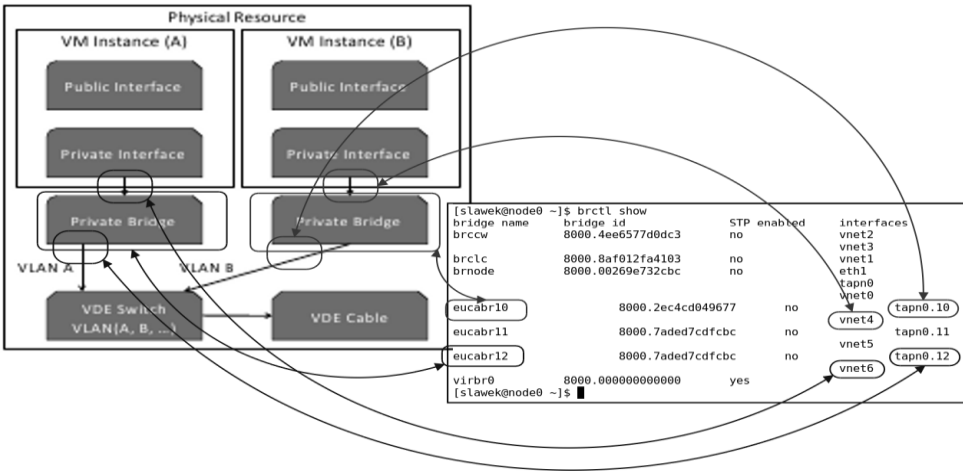
Rysunek 3 przedstawia trzy instancje systemu Ubuntu, uruchomione w testowym środowisku IaaS. Każda z nich przypisana jest do różnych grup bezpieczeństwa. W prawym dolnym rogu na rys. 3 znajduje się opis stworzonych grup (default, protector oraz tester). W lewym dolnym rogu znajduje się raport programu euca_tool, który potwierdza wykorzystanie mechanizmu elastycznych IP. Oprócz adresów prywatnych, Eucalyptus dynamicznie, bez udziału administratora, przypisał poszczególnym instancjom VM adresy publiczne. Dzięki temu użytkownicy mogą z łatwością łączyć się i korzystać z przyznanych zasobów.



Rys. 3. Implementacja grup bezpieczeństwa i elastycznych IP w prezentowanym środowisku testowym IaaS

Fig. 3. Security groups and elastic IP implementation inside the IaaS testbed

Interfejsy prywatne wszystkich instancji maszyn VM są połączone za pomocą narzędzia VDE (ang. *Virtual Distributed Ethernet*) [8]. Utworzony most VDE łączy interfejsy TAP, przypisane poszczególnym VM z wykorzystaniem techniki VLAN. Określony numer VLAN-u łączy wszystkie VM, przypisane do zadania zgłoszonego przez danego użytkownika. Zaimplementowany sposób wykorzystania dynamicznej infrastruktury VLAN w środowisku testowym przedstawiony jest na rysunku 4.



Rys. 4. Izolacja instancji VM w środowisku testowym IaaS
Fig. 4. The instances isolation inside the IaaS testbed

5. Podsumowanie

Struktury i usługi w sieciach chmurowych łącza w sobie wiele mechanizmów z obszaru inżynierii sieciowej, a także oprogramowania. Zaproponowane rozwiązania środowiska testowego IaaS mogą stać się przydatne w wielu badaniach i wdrożeniach systemów chmurowych. Ponadto, należy podkreślić, iż nie proponowano dotąd rozwiązania obejmującego wyłącznie pojedynczy system komputerowy, który jednocześnie, z łatwością pozwalałby na rozbudowę do pełnowymiarowych struktur IaaS. Możliwość zaimplementowania wszystkich podstawowych cech chmur prywatnych czyni opisane środowisko testowe przydatnym w pracach nad rozwiązaniami z obszarów PaaS i SaaS. Przykładem, w którym opisana architektura IaaS została wykorzystywana, jest analiza usługi strumieniowej PaaS, polegającej na dostępie do danych multimedialnych zgromadzonych w chmurze, a udostępnianych poprzez stos protokołów UPnP (ang. *Universal Plug and Play*). Dzięki implementacji środowiska testowego z wykorzystaniem pojedynczego systemu komputerowego (laptop) możliwe było skonfigurowanie usługi i przygotowanie scenariusza testów oraz ich uruchomienie, bez konieczności budowy rozbudowanego środowisko sieciowego. Następnie, wykorzystując możliwość szybkiego podłączenia udziałów iSCSI (testy różnych typów treści multimedialnych) oraz

zewnętrznych kontrolerów węzłów (ocena wydajności rozwiązania) możliwa była weryfikacja rozwiązania w środowisku docelowym.

BIBLIOGRAFIA

1. Vaquero L. M., Rodero-Merino L., Caceres J., Linder M.: A Break in the Clouds: Towards a Cloud Definition, ACM SIGCOMM Computer Communication Review, Vol.39, No.1, 2009, s. 50÷55.
2. Foster I., Zhao Y., Raicu I., Lu S.: Cloud Computing and Grid Computing 360-Degree Compared, Grid Computing Environments Workshop, 2008, s. 1÷10.
3. Nurmi D., Wolski R., Grzegorczyk Ch., Obertelli G., Soman S., Youseff L., Zagorodnov D.: The Eucalyptus Open-source Cloud-computing System, 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID), Vol. 0, 2009, s. 124÷131.
4. Armbrust M., Fox A., Griffith R., Joseph A. D., Katz R., Konwinski A., Lee G., Patterson D., Rabkin A., Stoica I., Zaharia M.: Above the Clouds: A Berkeley View of Cloud Computing, UC Berkeley Reliable Adaptive Distributed Systems Laboratory, September 2010.
5. Johnson D., Murari K., Raju M., Suseendran R. B., Girikumar Y.: Eucalyptus Beginner's Guide – UEC Edition, CSS Corp. 2010, http://www.csscorp.com/eucauecbook, June 2010.
6. Eucalyptus Administrator's Guide (2.0), http://open.eucalyptus.com/wiki/, September 2010.
7. Hybridfox, http://code.google.com/p/hybridfox/, September 2010.

Recenzenci: Prof. dr hab. inż. Stanisław Kozielski
Prof. dr hab. inż. Tadeusz Wieczorek

Wpłynęło do Redakcji 15 marca 2011 r.

Abstract

The concept of the cloud computing covers wide range of software and hardware solutions. For this reason, development of an universal testbed for this technology is unlikely possible. However, the growing interest in the public as well as the private clouds, demands fast and flexible testing environment. In so far as we know from review of available re-

sources, the proposed testbed, based on single hardware system with all security and management features, has not been reported yet.

Presented revision of the single machine cloud system has proved, that all the most important feature (security groups, elastic IP and VM appliances isolation) are supported. The test cloud was managed by the two software tools, graphical Hybridfox and text one, euca-tools. Both were installed on the VM guest Ubuntu Desktop 10. Thanks to that, all configurations and diagnostics tasks can be carried out on one machine. The other client can be easily add via wlan0 (interface connected to external network) with standard iptables modifications. External client is necessary in the case when custom image is to be created.

Additionally to the fact that the proposed testbed can be implemented on typical single computer hardware, it offers the possibility of fast expansion to more complex enterprise-like environment by connecting to next node or cluster controllers. The range of that expansion is not limited by resources of the presented testbed. This means that any idea can be configured and pre-tested on the testbed and final assessment of private cloud, in the target system, can be realized without the time-consuming reconfiguration. The IaaS structure, implemented in described testbed can be supplemented by the PaaS and the SaaS software frameworks so it may be a good starting point for research and development even the most complex cloud systems.

Adresy

Sławomir PRZYŁUCKI: Politechnika Lubelska, Katedra Elektroniki, ul. Nadbystrzycka 38A, 20-618 Lublin, admin@spg51.net
Daniel SAWICKI: Politechnika Lubelska, Katedra Elektroniki, ul. Nadbystrzycka 38A, 20-618 Lublin, sawi@politechnika.lublin.pl