1. Anas ZAIN DIN, 2, Krzysztof JANUSZEWSKI

Uniwersytet Morski w Gdyni, Zakład Telekomunikacji Morskiej
ORCID. 1. 0000-0003-3094-6908, 2. 0000-0003-4230-5235

doi:10.15199/48.2021.12.45

Wirtualna platforma do realizacji zdalnych zajęć dydaktycznych

Streszczenie. *W artykule opisano projekt i realizację wirtualnej platformy, zbudowanej w oparciu o sprzętowe rozwiązanie chmurowe oraz oprogramowanie klasy Open Source, przeznaczonej do realizacji zajęć prowadzonych w trybie zdalnym. Omówiono przykładowe scenariusze ćwiczeń z zagadnień dotyczących bezpieczeństwa sieciowego oraz przedstawiono doświadczenia z eksploatacji wdrożonego systemu.*

Abstract*. The article describes the design and implementation of a virtual platform, built on the basis of a hardware cloud solution and Open Source class software, intended for the implementation of classes conducted remotely. Sample scenarios of exercises on network security issues is discussed, as well as experience from the operation of the implemented system. (**The implementation of a virtual platform, built on the basis of a hardware cloud solution)*

Słowa kluczowe: chmura obliczeniowa oprogramowanie Open Source, bezpieczeństwo systemów informatycznych, zdalny dostęp
Keywords: .cloud computing, Open Source class software, security of information systems, remote access

Wstęp

Wybuch pandemii COVID-19 na początku roku 2020 postawił uczelnie przed koniecznością realizacji procesu dydaktycznego w trybie zdalnym. Jest to sytuacja szczególnie uciążliwa, zwłaszcza podczas zajęć praktycznych. Zakład Telekomunikacji Morskiej UMG posiadał pewne doświadczenia w prowadzeniu praktycznych zajęć zdalnych, z zakresu sieci komputerowych, oraz bezpieczeństwa systemów informatycznych, nabyte podczas realizacji szkoleń w ramach programu Sieciowej Akademii Cisco (*Cisco Networking Academy*). Ze względu na zainteresowanie osób spoza Trójmiasta w roku 2010 zbudowano infrastrukturę umożliwiającą osobom zainteresowanym zdalne korzystanie ze sprzętu niezbędnego do poznawania zagadnień z zakresu sieci komputerowych [1].

Pandemia COVID-19 stworzyła jednakże nową sytuację – należało rozszerzyć możliwości dostępu do infrastruktury laboratoryjnej dla wszystkich studentów studiów stacjonarnych i niestacjonarnych, uczestniczących w zajęciach prowadzonych w Laboratorium Sieci Komputerowych (LSK). Przedmioty te obejmują wolumen kilkuset godzin rocznie zajęć dydaktycznych i ponad 200 kont studenckich. W związku z tym podjęto decyzję o rozbudowie infrastruktury laboratoryjnej, zakładając możliwie najniższe nakłady finansowe, prostotę konfiguracji i zarządzania systemem, oraz bezpieczeństwo. Naturalnym było więc zastosowanie dostępnego na rynku sprzętu, stanowiącego jedyną pozycję w budżecie przeznaczonym na omawianą rozbudowę, oraz oprogramowania klasy *Open Source*.

Architektura sprzętowa

Architektura sprzętowa LSK została rozbudowana o trzy serwery R610 firmy Dell wyposażone w dwa ośmiordzeniowe procesory Intel® Xeon® Silver 4110 taktowane zegarem 2.10 GHz. Są to procesory wielowątkowe, zatem każdy serwer dysponuje 32. procesorami logicznymi. Serwery te wyposażono ponadto w 128 GB pamięci fizycznej, oraz 8 dysków twardych o pojemności 1 TB każdy. Serwery posiadają ponadto dwie gigabitowe karty sieciowe. Sprzęt ten uzupełnia dotychczasową infrastrukturę LSK opisaną w pracy [1]. Serwery, o wysokości 1U każdy, zostały zamontowane w szafie RACK 19". Pamięć masowa serwerów została skonfigurowana w trybie RAID-60, udostępniając pojemność 3.5 TB na każdym węźle. Infrastrukturę sprzętową uzupełnia 52-portowy przełącznik ethernet SRW248G4 firmy Linksys. Jest to najsłabsza część infrastruktury, ze względu na udostępnianie tylko czterech portów o szybkości 1 Gb/s. Wszystkie urządzenia zasilane są z buforowego zasilacza UPS.

Oprogramowanie systemowe

Główną funkcjonalnością oprogramowania systemowego, zainstalowanego na serwerach, niezbędną w projektowanym środowisku, jest możliwość uruchamiania różnych systemów operacyjnych. Jest to możliwe w środowisku umożliwiającym wirtualizację. W chwili obecnej istnieje wiele systemów operacyjnych udostępniających mechanizmy wirtualizacji. Najważniejsze z nich to ESXi firmy vmWare i Hyper-V firmy MIcrosoft. Są to środowiska o bardzo dużych możliwościach i szerokim spektrum zastosowań, oraz bardzo dobrym poziomie wsparcia. Ich pozyskanie i użytkowanie jest jednak związane ze znacznym poziomem kosztów niezbędnych dla nabycia stosownych licencji. Stąd zdecydowano się na zastosowanie środowiska wirtualizacyjnego Proxmox VE [2], które jest dostępne na zasadzie licencji otwartej i można je pozyskać bez ponoszenia kosztów. Podkreślić jednakże należy, że istnieje możliwość wykupienia kilku wersji licencji na to oprogramowanie, dającej użytkownikowi wsparcie techniczne i pomoc w rozwiązywaniu problemów.

Proxmox Virtualisation Environment (PVE) jest otwartoźródłową platformą przeznaczoną do wirtualizacji infrastruktury informatycznej. Technika wirtualizacji zastosowana w PVE wykorzystuje wirtualizację opartą na jądrze systemu hosta (KVM), oraz konteneryzację (LXC). Ponadto w PVE zaimplementowano mechanizmy zarządzania pamięcią masową zdefiniowaną programowo (*Software-Defined Storage*), oraz funkcjami sieciowymi. Zarządzanie środowiskiem PVE odbywa się przez interfejs WWW, co ułatwia tworzenie, konfigurację i nadzór nad maszynami wirtualnymi i kontenerami. Ponadto PVE charakteryzuje się wysoką dostępnością dla klastrów oraz zintegrowanymi narzędziami do odzyskiwania po awarii.

Oprogramowanie systemowe Proxmox VE zainstalowano na wszystkich dostępnych w LSK serwerach (4 szt.), nadając im nazwy pve1,...pve4. Następnie serwery te zintegrowano tworząc infrastrukturę jednolitej chmury obliczeniowej. Podejście takie umożliwia łatwą budowę obrazów wzorców maszyn wirtualnych, a następnie tworzenie na ich podstawie, na dowolnym węźle, niezbędnych przy realizacji konkretnego scenariusza ćwiczenia maszyn, ich migrację, uruchamianie, zatrzymywanie czy wykonywanie tzw. migawek. Wykorzystywane systemy operacyjne gościa mogą być w praktyce dowolne. W chwili obecnej uruchamiane są tam systemy z rodziny MS-Windows, różne dystrybucje systemu operacyjnego Linux, system operacyjny urządzeń sieciowych Cisco IOS i inne. O wydajności opracowanego systemu świadczy fakt,

że podczas realizacji zajęć w chmurze pracowało jednocześnie ponad dwieście maszyn wirtualnych.

W systemie założono konta dla użytkowników z uprawnieniami zależnymi od pełnionych przez nich ról. Zdefiniowano odrębne role dla administratorów, – umożliwiające pełną kontrolę nad działającymi maszynami, prowadzących zajęcia – uprawniające ich do zatrzymywania i uruchamiania maszyn oraz odtwarzanie ich stanów z migawek, a także dla studentów – z uprawnieniami pozwalającymi na wykorzystanie maszyn w celu przeprowadzenia ćwiczenia.

Zdalny dostęp do zasobów LSK

Bezpieczny dostęp do zasobów chmury LSK zrealizowano z zastosowaniem wirtualnej sieci prywatnej (VPN). W tym celu na jednym z węzłów środowiska PVE uruchomiono serwer z systemem operacyjnym Linux na którym skonfigurowano serwer OpenVPN [3], Serwer ten wyposażono ponadto w interfejs zarządzania dostępny przez przeglądarkę. Interfejs ten umożliwia tworzenie kont dla użytkowników, generację certyfikatów dostępowych, oraz zarządzanie serwerem.

W celu nawiązania połączenia z zasobami LSK użytkownik wykorzystuje oprogramowanie klienckie OpenVPN, oraz wygenerowany certyfikat. Uwierzytelniony użytkownik ma dostęp do zasobów pracujących w środowisku PVE, zgodnie ze scenariuszem prowadzonych zajęć.

Model sieci komputerowej

Głównym celem budowy opisywanego środowiska było umożliwienie prowadzenia szkoleń z zakresu zaawansowanych zagadnień dotyczących sieci komputerowych, realizowanych w trybie zdalnym [1]. Rozbudowa systemu umożliwia realizację zajęć z wielu zagadnień dotyczących informatyki praktycznej. Tytułem ilustracji możliwości opracowanego systemu, przedstawiono wybrane scenariusze ćwiczeń praktycznych, dotyczących zagadnień bezpieczeństwa sieci i systemów komputerowych. Opracowane ćwiczenia są realizowane w topologii sieciowej, w której skład wchodzą: serwer i stacja robocza z systemem operacyjnym MS-Windows, serwer usług sieciowych z systemem operacyjnym linux, dwa routery z systemem operacyjnym Cisco IOS, oraz dwa komputery z systemem Kali Linux, wyposażone w oprogramowanie przeznaczone do przeprowadzania standardowych testów penetracyjnych.

Celem zapewnienia realizmu prowadzonych ćwiczeń topologia ma dostęp do sieci Internet. Przedstawioną topologię uruchomiono w 10 egzemplarzach, udostępniając studentom dostęp do interfejsu komputerów z systemami Windows oraz Kali Linux.

Odtworzenie sieci

Celem ćwiczenia jest praktyczne wykorzystanie narzędzi i programów do odtworzenia schematu sieci i znalezienie tras między urządzeniami. Student, mając dostęp wyłącznie do wybranych komputerów powinien odtworzyć schemat całej sieci wraz z adresami IP wszystkich interfejsów sieciowych.

Podczas tego ćwiczenia student zapoznaje się z praktycznym wykorzystaniem następujących narzędzi i oprogramowania: *ipconfig, ifconfig, ping, tracert, traceroute, netstat, Zenmap*.

Skanowanie sieci

Ćwiczenie to polega na skanowaniu sieci lokalnych i zdalnych oraz zbieraniu informacji na temat istniejących urządzeń. Informacje jakie można uzyskać zależą od techniki skanowania. Mogą one dotyczyć tylko adresów IP i masek. Można również uzyskać bardziej szczegółowe informacje, np. o działających usługach sieciowych, systemach operacyjnych oraz podatnościach i lukach w zabezpieczeniach tych systemów [4]. Celem ćwiczenia jest zapoznanie się z metodami i narzędziami skanowania sieci, oraz interpretacja i analiza uzyskanych informacji dotyczących każdego urządzenia w sieci. Proces skanowania wykonany jest z maszyny Kali1 oraz Kali2 w zależności od rodzaju skanowania.

Skanowanie warstwy drugiej, polega na szybkim skanowaniu sieci LAN i zbieraniu informacji o istniejących komputerach w sieci lokalnej. Technika ta polega na wysyłaniu zapytań ARP i kontroli czy host docelowy jest aktywny i czy odpowiada na przesłanie żądanie. Skanowanie warstwy trzeciej polega na wysyłaniu żądań ICMP echo i oczekiwaniu na odpowiedzi ze zdalnych hostów.

Skanowanie warstwy czwartej wykorzystuje protokoły TCP i UDP. Aby wykonać skanowanie na warstwie czwartej, musi działać system, który będzie odpowiadał na żądanie TCP/UDP. W badanej sieci maszyną, pełniącą tę rolę jest serwer linux1, na którym działają różne usługi TCP/UDP, oraz routery na których działa system CiscoIOS i uruchomione są usługi WWW, SSH oraz telnet.

Zbieranie informacji o systemach operacyjnych działających na poszczególnych hostach, oraz usługach sieciowych uruchomionych na poszczególnych portach znane jest pod nzwą *Fingerprinting*. Na podstawie uzyskanych w tej fazie ataku sieciowego wyników, można określić jakie usługi sieciowe działają na danej maszynie i analizować wersje powiązanego z nimi oprogramowania. Skanowanie tego typu realizowane jest przy użyciu silnika Nmap NSE (Nmap Scripting Engine). Jest to rozbudowane narzędzie wyspecjalizowanych skryptów, których można używać do automatyzacji skanowania systemów zdalnych.

Podczas tego ćwiczenia studenci zapoznają się z praktycznym wykorzystaniem narzędzi i programów: *nmap, netdiscover, ping, fping, netcat, NSE, Zenmap, Wireshark, host, nslookup*.

Atak Man-In-The-Middle (MiTM) z użyciem techniki ARP spoofing

Atak ARP spoofing pozwala na przechwycenie danych w obrębie sieci lokalnej. Atak ten polega na rozsyłaniu w sieci LAN pakietów ARP zawierających fałszywe adresy MAC. Wskutek tego działania, stacja robocza przesyłająca zapytanie ARP, otrzymuje w odpowiedzi od atakującego adres MAC jego komputera. Poprawnie wykonany atak powoduje, że ofiara nie ma nawet świadomości, że jest podsłuchiwana[5].

Celem ćwiczenia jest zapoznanie się z atakiem MiTM i jego skutkami. Ćwiczenie praktycznie dowodzi jak bardzo ważne jest zabezpieczanie i szyfrowanie protokołów sieciowych. Atak ten jest przeprowadzony w sieci LAN. Maszyną atakującą jest Kali1 a pozostałe urządzenia w sieci są ofiarami.

Atak rozpoczynany jest ze stacji Kali1 i polega na wykonaniu skanowania hostów w sieci lokalnej celem uzyskania ich adresów MAC oraz adresów IP. Następnie, atakująca maszyna podszywa się jako brama domyślna w sieci LAN, celem przejęcia całego ruchu kierowanego do routera. Atak wykonywany jest w kilku etapach:

- wykonanie ataku "arpspoofing",
- za pomocą programu Wireshark dokonuje się przechwycenia wybranych protokołów sieciowych takich jak ICMP, telnet, HTTP, SMTP, POP i innych
- przechwycenie adresów URL odwiedzanych przez ofiarę za pomocą narzędzia "urlsnarf",
- przechwycenie obrazów i zdjęć przesyłanych w sieci przez niezabezpieczone protokoły za pomocą narzędzia "driftnet",

- przechwycenie loginów i haseł przesyłanych przez niezabezpieczone protokoły takich jak ftp, smtp, pop, telnet za pomocą narzędzia "dsniff",

Atatk DNS spoofing - jest to technika phishingu polegająca na wykonaniu przez atakującego przesłania do serwera DNS fałszywej informacji kojarzącej nazwę domeny z adresem IP. Serwer DNS zapamiętuje ją na pewien czas i zwraca klientom zapamiętany adres IP, czego skutkiem jest przeniesienie na fałszywą stronę.

Praktycznie wykorzystywane podczas tego ćwiczenia narzędzia i programy to: *arpspoofing, Wireshark, urlsnarf, driftnet, dsniff, ettercap*.

Ataki typu DoS (Denial of Service)

Jest to rodzaj ataku mający na celu uniemożliwienie działania systemu komputerowego lub usługi sieciowej. Polega on na zalewaniu maszyny atakowanej pakietami sieciowymi w celu przeciążenia i spowolnienia jego odpowiedzi na zapytania klientów.

Celem ćwiczenia jest zapoznanie się z różnymi technikami ataku DoS/DDoS oraz zaobserwowanie efektów i skutków takiego ataku w sieciach. Ataki tego typu mogą być przeprowadzanie w różnych warstwach stosu protokołów komunikacyjnych.

Atak na usługę DHCP. Atak ten nazywany jest DHCP Denial of Service Attack. Polega on na wyczerpaniu całej puli DHCP. Proces ten skutkuje wyczerpaniem całej puli adresowej DHCP, co powoduje, że stacje robocze istniejące w sieci nie będą mogły uzyskać adresów IP z serwera DHCP.

Ważną klasą ataków typu *Denial of Service* są ataki wykorzystujące protokół ICMP (*Internert Control Message Protocol*). Można tu wymienić dwa ataki tego typu: *Smurf Attack* oraz *Ping of Death*.

Smurf Attack jest jednym z najstarszych ataków typu DDoS, wykorzystującym pakiety ICMP. Atak polega na wysyłaniu pakietu ICMP *Echo Request* z rozgłoszeniowym adresem docelowym. Powoduje to, że wszystkie urządzenia będą odpowiadały na ten pakiet pakietem ICMP *Echo Reply*.

Ping of Death jest to atak, wykorzystujący koncepcję wysyłania złośliwego polecenia ping do innego komputera, przekraczającego maksymalny rozmiar datagramu IPv4. Prawidłowo utworzony pakiet ping ma zazwyczaj rozmiar 56 lub 64 bajty, jeśli bierze uwzględnia się nagłówek ICMP, a 84 bajty wraz z nagłówkiem protokołu internetowego w wersji 4. Jednak każdy pakiet IPv4 (w tym ping) może mieć nawet 65 535 bajtów. Niektóre systemy komputerowe nigdy nie zostały zaprojektowane do prawidłowej obsługi pakietu ping większego niż maksymalny. Podobnie jak inne duże, ale dobrze uformowane pakiety, ping śmierci jest przed transmisją dzielony na grupy po 8 oktetów. Gdy komputer docelowy ponownie złoży zniekształcony pakiet, może wystąpić przepełnienie bufora, powodując awarię systemu i potencjalnie umożliwiając wstrzyknięcie złośliwego kodu.

SYN Flood Attack jest to atak typu DoS, mający na celu wyczerpywania zasobów atakowanego systemu. Działanie polega na wysyłaniu dużej liczby segmentów TCP SYN do zdalnego portu np. 80. Dla każdego otrzymanego pakietu SYN , atakowany system odpowiada pakietem SYN+ACK i utrzymuje otwarte połączenie w oczekiwaniu na nadejście finalnego pakietu ACK. Atakowany serwer jest przeładowany nadmierną liczbą półotwartych połączeń, co powoduje odmowę działania usługi.

Land Attack polega na wysyłaniu przez atakującego segmentów TCP SYN, gdzie adres źródłowy i docelowy oraz numery portów są ustawione jednakowo na adres ofiary. Powoduje to, że ofiara odpowiadając na te segmenty, wchodzi w pętlę, zwiększając obciążenie urządzenia.

Przykładem ataku w warstwie aplikacji może być Slowloris Attack wykorzystujący protokół HTTP i metodę GET. Atak ten polega na otwieraniu wielu połączeń i bardzo powolnym dosyłaniu częściowych żądań HTTP do serwera docelowego oraz pozostawianiu ustanowionego połączenia w stanie otwartym, tak długo jak to możliwe. Skutkuje to wyczerpaniem puli wolnych wątków obsługujących żądania HTTP.

Podczas realizacji tego ćwiczenia studenci zapoznają sie z praktycznym wykorzystaniem programów *scapy, hping3, metasploit, slowloris*.

Ataki z wykorzystaniem narzędzia Metasploit

Metasploit jest to otwarto-źródłowe narzędzie służące do testów penetracyjnych i łamania zabezpieczeń systemów teleinformatycznych [6]. Metasploit zawiera bazę gotowych programów pozwalających na włamanie się do maszyn zdalnych. Celem ćwiczenia jest pokazanie podatności w systemie Windows oraz Linux. Celem ataku są dwa różne systemy operacyjne: serwer pracujący pod kontrolą system Linux z różnymi usługami sieciowymi, oraz serwer pracujący pod kontrolą systemu MS-Windows Server 2000. Atak rozpoczynany jest od skanowania obu maszyn (SRV1 oraz linux1) celem uzyskania informacji o podatnościach i lukach w zabezpieczeniach systemów. Atak przeprowadzany jest z użyciem silnika Nmap NSE. Atakowane maszyny posiadają między innymi następujące podatności:

Linux1 posiada lukę w usłudze FTP. Jest to luka w programie, znana pod nazwą VSFTPD v2.3. Luka ta została odkryta w 2011 roku, kiedy doszło do włamania na repozytoria pakietu Vsftpd, w wyniku którego pliki wykonywalne pakietu zostały zamienione na pliki zawierające błędny kod. Luka ta pozwala nieautoryzowanym użytkownikom zalogowanie się do serwera. Zalogowanie się na takie konto powoduje automatyczne udostępnienie powłoki użytkownika root na porcie 6200.

SRV1 posiada lukę w serwerze SMB. Jest to popularna luka w systemach WinXP znana pod nazwą MS08-067. Docelowa usługa SMB odbiera połączenie i wywołuje funkcję przepełniającą bufor stosu maszyny ofiary. Następnie atakująca maszyna wysyła pakiet danych większy niż oczekuje ofiara. Dane te są zawarte w ładunku, co powoduje przepełnienie docelowego bufora. Exploit ten umożliwia atakującemu przyjęcie kontroli nad powłoką ofiary.

Ćwiczenie to ma na celu zapoznanie studentów z oprogramowaniem *metasploit, msfconsole, nmap*, oraz skryptami *Nmap Scripting Engine*.

Wnioski

Opracowane środowisko wirtualizacyjne pozwala na prowadzenie zajęć z wielu przedmiotów z zakresu technik komputerowych. Opisane w artykule scenariusze prowadzonych zajęć z zakresu bezpieczeństwa systemów informatycznych stanowią niewielki wycinek zastosowań systemu w procesie dydaktycznym. Doświadczenia zebrane podczas dotychczasowej eksploatacji pozwala stwierdzić, że tak prowadzone zajęcia są atrakcyjne dla studentów i stosunkowo mało obciążające dla prowadzących. W artykule nie opisano szeregu narzędzi, opracowanych w ramach rozbudowy systemu, realizujących zadania z obszaru administracji systemem, odtwarzania maszyn po zajęciach i awariach, zarządzania użytkownikami i ich uprawnieniami i wielu innych. W ramach rozbudowy systemu planowane jest wdrożenie obiektowego systemu plików Ceph, oraz narzędzia wirtualizacji sieci OpenStack.

Autorzy pragną podziękować Dziekanowi Wydziału Elektrycznego Uniwersytetu Morskiego w Gdyni prof. dr. hab. inż. Krzysztofowi Góreckiemu za sfinansowanie sprzętu niezbędnego do uruchomienia opisanego rozwiązania.

***Autorzy**: mgr inż. Anas Zain Din, Uniwersytet Morski w Gdyni, Zakład Telekomunikacji Morskiej, ul. Morska 81-87, 81-225 Gdynia E-mail: a.zaindin@we.umg.edu.pl,*
dr inż. Krzysztof Januszewski, Uniwersytet Morski w Gdyni, Zakład Telekomunikacji Morskiej, ul. Morska 81-87, 81-225 Gdynia E-mail: k.januszewski@we.umg.edu.pl

LITERATURA

[1] Januszewski K., Zdalny dostęp do zasobów Laboratorium Sieci Komputerowych, Zeszyty Naukowe AMG, 2015, nr 90, 139-148,
[2] Goldman R. Learning Proxmox VE, wyd. PACKT Publishing, Birmingham - Mumbai, 2016,
[3] Januszewski K., Szelągowski P., System rezerwacji czasu dostępu do zasobów laboratorium sieci komputerowych na podstawie pakietu LAMP, Zeszyty Naukowe AMG, nr 95, 2016, 148-156,
[4] Huthens J. Skanowanie sieci z Kali Linux. Receptury, wyd. Helion, Gliwice 2015,
[5] Józefiok A, CCNA Security, wyd. Helion, Gliwice 2016,
[6] Agarwal M, Singh A. Metasploit. Receptury pentestera, wyd. Helion Gliwice 2014.