Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej

XVII Seminarium

ZASTOSOWANIE KOMPUTERÓW W NAUCE I TECHNICE' 2007
Oddział Gdański PTETiS
Referat nr 12

PRZEGLĄD MECHANIZMÓW ZABEZPIECZANIA SYSTEMU OPERACYJNEGO

Jerzy Kaczmarek[1], Michał Wróbel[2]

1. Wydział Elektroniki, Telekomunikacji i Informatyki, Politechnika Gdańska
tel: (58) 347 26 82 fax: (58) 347 27 27 e-mail: jkacz@eti.pg.gda.pl
2. Wydział Elektroniki, Telekomunikacji i Informatyki, Politechnika Gdańska
tel: (58) 347 10 37 fax: (58) 347 27 27 e-mail: wrobel@eti.pg.gda.pl

Streszczenie: Zapewnienie bezpieczeństwa systemu komputerowego jest jednym z najważniejszych zadań systemu operacyjnego. W artykule zostaną przedstawione mechanizmy i technologie zabezpieczania systemu operacyjnego Linux. Zostaną opisane metody aktywnej ochrony systemów operacyjnych poprzez blokowanie połączeń sieciowych i ograniczanie praw dostępu aplikacji do zasobów. Przedstawione zostaną również rozwiązania ograniczające szkody dokonywane przez intruzów. Różnorodność stosowanych mechanizmów bezpieczeństwa jest warunkiem skutecznej ochrony systemów komputerowych.

Słowa kluczowe: bezpieczeństwo, systemy operacyjne, Linux

1. WSTĘP

W dobie powszechnego dostępu do sieci Internet włamania do systemów komputerowych są jednym z najpoważniejszych zagrożeń dla użytkowników. Obecnie komputery służą nie tylko do tworzenia, przetwarzania i przechowywania danych, ale także do prowadzenia, często poufnej, korespondencji, dokonywania zakupów czy korzystania z usług bankowych. Również większość systemów wdrożonych w przedsiębiorstwach jest podłączona do sieci globalnej. Z tych powodów zapewnienie bezpieczeństwa systemów komputerowych jest jednym z ważniejszych zagadnień w dziedzinie informatyki.

W artykule zostaną przedstawione mechanizmy i technologie pozwalające na zwiększenie bezpieczeństwa systemu operacyjnego Linux. Mechanizmy te zostały podzielone na trzy grupy. Do pierwszej grupy można zaliczyć mechanizmy filtrowania ruchu sieciowego pozwalające na blokowanie dostępu do sieci lokalnych. Do drugiej grupy należą mechanizmy ograniczające szkody jakie może dokonać intruz po udanym włamaniu do systemu. W trzeciej grupie znajdują się metody wykrywania włamań oraz modyfikacji dokonywanych przez intruza w systemie plików.

2. FILTROWANIE DOSTĘPU DO KOMPUTERA

Najskuteczniejszym sposobem zapewnienia bezpieczeństwa systemu operacyjnego jest całkowite odizolowanie go od sieci Internet. Jednak obecnie tylko niewielka część komputerów działa w ten sposób. Zdecydowana większość maszyn jest podłączona do sieci, dotyczy do zarówno komputerów domowych, firmowych czy np. serwerów obliczeniowych.

Zamiast całkowitego odłączenia komputera od sieci Internet powszechnie stosuje się filtrowanie ruchu sieciowego wychodzącego i przychodzącego do systemu. Narzędzia do tego służące noszą nazwę zapór sieciowych (ang. firewall). Zapora sieciowa jest to system lub grupa systemów kontrolujących ruch pomiędzy dwoma sieciami na podstawie zdefiniowanych reguł chroniących je w ten sposób przed niepowołanym dostępem [1].

2.1. Mechanizm działania zapory sieciowej

Firewall pozwala na filtrowanie ruchu sieciowego pomiędzy siecią globalną i lokalną. W ten sposób można ograniczyć dostęp do zasobów lub usług tylko dla wybranych komputerów zewnętrznych, jak również kontrolować dostęp do komputerów zewnętrznych przez użytkowników sieci lokalnej. Zapory sieciowe pozwalają także na autoryzowanie użytkowników, śledzenie ruchu sieciowego czy ukrywanie architektury sieci lokalnych.

Funkcje zapory sieciowej mogą pełnić zarówno dedykowane urządzenia sprzętowe, jak i specjalne programy. Poszczególne rozwiązania mogą się znacząco różnić w sposobie implementacji poszczególnych funkcji filtrowania ruchu sieciowego.

Istnieją dwie powszechnie stosowane polityki filtrowania ruchu sieciowego, które są definiowane w następujący sposób:

- wszystko, co nie jest dozwolone, jest zabronione,
- wszystko, co nie jest zabronione, jest dozwolone.

Pierwsze podejście zakłada, że domyślnie blokowany jest cały ruch sieciowy, a dopiero poszczególne reguły pozwalają na odblokowanie pożądanych usług [2]. Taka

Recenzent: dr inż. Miron Kłosowski – Wydział Elektroniki, Telekomunikacji i Informatyki Politechnika Gdańska

polityka zapewnia większe bezpieczeństwo, jednak kosztem pewnych ograniczeń dla użytkowników.

W drugim podejściu domyślnie wszystko jest odblokowane, a wdrożone reguły blokują dostęp do określonych zasobów. Zapewnia to większą elastyczność, jednak kosztem znacznego nakładu pracy koniecznej do zapewnienia pełnego bezpieczeństwa chronionej sieci lub systemu [3].

2.2. Rodzaje zapór sieciowych

W zależności od danych używanych do filtrowania ruchu sieciowego, wyróżnia się kilka rodzajów zapór sieciowych. Każdy rodzaj wykorzystuje informacje zapisane w innej warstwie modelu ISO-OSI [4].

Zapora sieciowa filtrująca pakiety (ang. *packet filtering firewall*) kontroluje ruch sieciowy na podstawie danych zawartych w opisie pakietów sieciowych. Pozwala ona na filtrowanie pakietów pochodzących z określonych adresów IP, sieci, podsieci, czy portów TCP lub UDP.

Filtr kontekstowy, zwany również zaporą sieciową z analizą stanów (ang. *stateful packet inspection firewall*), pozwala na analizowanie i filtrowanie ruchu sieciowego w kontekście całych sesji komunikacyjnych. W tym celu mogą być badane wszystkie warstwy modelu OSI począwszy od warstwy sieciowej, a skończywszy na warstwie aplikacji.

Serwer pośredniczący (ang. *proxy server*) jest zaporą sieciową działającą na najwyższej warstwie modelu OSI, warstwie aplikacji. Jego działanie polega na całkowitym odseparowaniu chronionej sieci od Internetu. Komputer w sieci wewnętrznej nie może nawiązać bezpośrednio połączenia z żadnym zewnętrznym serwerem, lecz łączy się z serwerem proxy i dopiero on nawiązuje połączenie z komputerem docelowym. W ten sposób połączenie klient-serwer jest całkowicie kontrolowane poprzez zaporę pośredniczącą co daje administratorom większą kontrolę nad ruchem sieciowym, można np. kontrolować połączenia wykonywane przez aplikacje.

Inne rodzaje zapór sieciowych, takie jak NAT (ang. *Network Address Translation*) czy VPN (ang. *Virtual Private Network*) nie są stosowane bezpośrednio do filtrowania ruchu sieciowego, ale są wykorzystywane np. do ukrywania istniejących sieci lokalnych [5].

Obecnie większość zapór sieciowych łączy różne metody przetwarzania pakietów. Istniejące rozwiązania są rozbudowywane, jak również powstają nowe rozwiązania.

Jednak nawet najbardziej złożone zapory sieciowe nie są w stanie całkowicie ochronić systemu. Pozwalają jedynie na częściowe lub całkowite odseparowanie sieci lokalnej od środowiska zewnętrznego. Zapory sieciowe nie chronią przed następującymi zagrożeniami:

- wirusy komputerowe,
- konie trojańskie,
- ataki dokonywane wewnątrz sieci,
- błędy (np. przepełnienia stosu) w programach.

Z tych powodów, aby zapewnić pełne bezpieczeństwo systemów, oprócz zastosowania zapór sieciowych, konieczne jest wykorzystywanie innych metod zabezpieczania systemów komputerowych.

3. OGRANICZENIE MOŻLIWOŚĆI INTRUZA

Wśród wielu projektów mających na celu poprawienie bezpieczeństwa systemu operacyjnego, można wyróżnić takie, które zakładają, że nie można stworzyć systemu, który całkowicie wyeliminuje groźbę włamania. Takie rozwiązania koncentrują się na ograniczaniu możliwości działania intruza, który włamał się do systemu.

Do tej grupy systemów zabezpieczających należą m.in. SELinux, który ogranicza przywileje procesów, AppArmor ograniczający skutki błędnego wykonywania programów oraz mechanizm wirtualizacji, który umożliwia całkowite odseparowanie od siebie usługi oferowanych przez system operacyjnego.

3.1. Security-Enhanced Linux

Security-Enhanced Linux, w skrócie SELinux, jest rozszerzeniem jądra systemu operacyjnego Linux, stworzonym przez Narodową Agencję Bezpieczeństwa USA (ang. *National Security Agency, NSA*) i udostępnionym publicznie na zasadach Open Source w 2000 roku. Rozwiązanie to powstało, aby zapewnić bezpieczeństwo krytycznych zasobów informatycznych w organizacjach rządowych i militarnych.

W systemach typu Unix domyślną polityką kontroli dostępu do zasobów jest tzw. swobodna kontrola dostępu (ang. *Discretionary Access Control, DAC*). Oznacza to, że właściciel może innym użytkownikom dowolnie udostępniać własne zasoby.

SELinux jest praktyczną implementacją innego, uznawanego za znaczenie bezpieczniejszy, modelu kontroli zwanego modelem obowiązkowej kontroli dostępu (ang. *Mandatory Access Control, MAC*). Konfiguracja praw dostępu w modelu MAC przypomina reguły zapór sieciowych. Administrator definiuje filtry dostępu do zasobów, które nie mogą być przez nikogo innego zmienione. Podobnie jak w przypadku zapór sieciowych nadrzędna zasada głosi, że to co nie jest dozwolone jest zabronione [6].

W przypadku, gdy intruz włamie się wykorzystując błąd w aplikacji działającej z uprawnieniami użytkownika root, powstałe szkody będą się ograniczały tylko do zasobów zdefiniowanych w regule dla tej aplikacji, a nie jak w modelu DAC do całego systemu.

SELinux implementuje również wielopoziomowy model bezpieczeństwa (ang. *Multi-Level Security, MLS*), powszechnie stosowany w systemach wojskowych, który pozwala na przesyłanie informacji z dolnych poziomów bezpieczeństwa do górnych, czyli np. z poziomu poufnego do tajnego, czy z tajnego do ściśle tajnego.

Pomimo zastosowania kontroli dostępu opartej na rolach (ang. *Role-Based Access Control, RBAC*), system SELinux jest skomplikowany i trudny do wdrożenia oraz utrzymania. Pomimo znacznych zalet zwiększających bezpieczeństwo, nie jest powszechnie stosowany poza organizacjami rządowymi czy militarnymi.

3.2. System zabezpieczeń AppArmor

System AppArmor jest podobny do rozwiązania SELinux, lecz jego wdrożenie i konfiguracja jest zdecydowanie łatwiejsza. AppArmor implementuje częściowo model obowiązkowej kontroli dostępu do zasobów (MAC). Model ten nie jest stosowany dla całego

systemu, lecz jedynie dla określonych, krytycznych aplikacji.

AppArmor najlepiej nadaje się do chronienia serwerów sieciowych, które udostępniają usługi publicznie. Przykładowo w razie włamania przez, chroniony przez AppArmor, serwer HTTP intruz nie będzie w stanie odczytywać dowolnych plików z całego systemu czy uzyskać dostęp do powłoki, lecz jedynie uzyska dostęp do plików, które były używane przez proces serwera[7].

AppArmor z uwagi na większą szybkość działania i prostszą konfigurację jest dobrą alternatywą dla SELinux, zwłaszcza dla serwerów sieciowych. Jednak w przypadku systemów wieloużytkownikowych zabezpieczenia przez niego stosowane mogą nie być wystarczające.

3.3. Wirtualizacja

Obecnie wzrasta zainteresowaniem wykorzystaniem mechanizmu wirtualizacji do zapewnienia bezpieczeństwa systemów operacyjnych. Wirtualizacja jest techniką pozwalającą na uruchamianie wielu systemów operacyjnych tzw. systemów gościa (ang. *guest system*) na jednym systemie operacyjnym zwanym gospodarzem (ang. *host system*).

Wyróżniane są dwa główne typy wirtualizacji:

- wirtualizacja sprzętu,
- wirtualizacja programów.

Implementacja wirtualizacji sprzętu polega na stworzeniu programowego emulatora komputera, w którym instalowany jest system operacyjny. Przy wirtualizacji programów system gościa może się w pewnej przestrzeni odwoływać do warstwy sprzętowej komputera gospodarza [8]. Najnowsze rozwiązania wykorzystują specjalne procesory, które umożliwiają dokonywanie prawdziwej, sprzętowej wirtualizacji. Z punktu widzenia bezpieczeństwa nie ma znaczenia, który z typów wirtualizacji jest wykorzystywany, jest to raczej kwestia wydajności poszczególnych systemów wirtualnych.

Wirtualizacja pozwala na odseparowanie środowisk roboczych. Atak, włamanie czy zainfekowanie wirusem systemu gościa nie jest groźne dla systemu gospodarza, jak i innych wirtualnych systemów. Przykładowo w serwerach internetowych można wdrożyć usługę np. www, ftp czy pocztę na osobnym serwerze wirtualnym. Można też tworzyć osobny system wirtualny dla różnych obsługiwanych domen. W takim przypadku udane włamanie na serwer konkretnej domeny nie pociągnie za sobą zagrożenia dla innych domen.

Coraz częściej wirtualizacja jest wykorzystywana w procedurach pozwalających na przywracanie systemu po awarii (ang. *disaster recovery*). W związku z tym, że wirtualne systemy nie są związane z konfiguracją sprzętową, ułatwione jest przeniesienie obrazu wirtualnego systemu na inny komputer z całkiem różną konfiguracją sprzętową [9].

Wirtualizacja jest również wykorzystywana do automatycznego tworzenia tzw. migawek (ang. *snapshot*) kopii zapasowych systemu, który działa jako system gościa na stacjach roboczych. W przypadku wykrycia włamania lub zawirusowania można przywrócić bezpieczną konfigurację z migawki.

Wadą systemów bezpieczeństwa opartych na wirtualizacji są stosunkowe duże wymagania sprzętowe. Komputer, na którym zainstalowane są wirtualne serwery jest tzw. punktem krytycznym i jego awaria może być poważnym problemem. Taka sytuacja nosi nazwę pojedynczego punktu awarii (ang. *single point of failure*) i aby jej zapobiec konieczne jest posiadanie serwerów rezerwowych.

Wirtualizacja jest stosunkowo nową metodą zabezpieczania systemów operacyjnych, o dużym potencjale. Pozwala na zarządzanie bezpieczeństwem systemów operacyjnych w sposób kompleksowy. W miarę pojawiania się nowych procesorów z wbudowaną technologią wspierającą wirtualizację sprzętową można spodziewać się znacznego wzrostu wykorzystania tej technologii do zapewniania bezpieczeństwa systemów operacyjnych.

4. DETEKCJA WŁAMAŃ

Ważną grupą produktów podnoszących bezpieczeństwo systemów operacyjnych stanowią programy pozwalające na wykrywanie włamań. Do tej grupy należą m.in. systemy detekcji włamań i systemy sprawdzania integralności systemów plików. Rozwiązania tego typu nie zabezpieczają bezpośrednio systemów komputerowych, ale pomagają wykrywać niebezpieczeństwa związane z podejrzaną aktywnością w sieci czy nieautoryzowane zmiany w plikach systemowych.

4.1. Systemy detekcji włamań

Działanie systemów detekcji włamań (ang. *Intrusion Detection System, IDS*) polega na monitorowaniu ruchu sieciowego i, w razie wykrycia określonych zdarzeń, powiadomieniu administratora systemu. Bardziej rozbudowane systemy IDS mogą automatycznie podjąć działania obronne, np. zablokować podejrzane adresy IP czy wyłączyć atakowane usługi.

Pod względem miejsca w którym działają systemy IDS można wyodrębnić systemy działające w sieci (ang. *Network Intrusion Detection System, NIDS*) lub na komputerze docelowym (ang. Host *Intrusion Detection System, HIDS*). W zależności od metod wykrywania podejrzanych zachowań można wyodrębnić systemy działające na bazie sygnatur znanych ataków lub takie, które wykrywają anomalie w zwykłym ruchu sieciowym. Systemy IDS, które tylko wykrywają zagrożenie i informują o tym administratorów zwane są systemami pasywnymi. Systemy aktywne mogą podejmować określone akcje mające na celu uniemożliwienie ataku [10].

W najnowszych rozwiązaniach granica pomiędzy systemami wykrywania włamań a zaporami sieciowymi zaczyna zanikać. Powstają systemy prewencji (ang. *Intrusion Prevention System, IPS*), które łączą filtrowanie ruchu sieciowego z aktywnym systemem wykrywania włamań [11].

Proces wdrożenia systemów IDS wymaga czasochłonnego dostosowania konfiguracji do środowiska sieciowego, aby z jednej strony nie generował zbyt wiele fałszywych alarmów (ang. *false positives*), a z drugiej wykrywał faktyczne próby włamania [12].

4.2. Ochrona integralności plików

Mechanizm ochrony integralności systemu plików (ang. *file system integrity checker*) pozwala na wykrycie

zmian dokonanych przez intruza w plikach systemowych. Po udanym włamaniu do systemu intruz często pozostawia tzw. tylne drzwi (ang. *backdoor*) poprzez zamianę któregoś z plików systemowych na inny, specjalnie spreparowany. Takie działanie może umożliwić w przyszłości obejście wszystkich zabezpieczeń i uzyskanie dostępu do systemu

Działanie systemów ochrony integralności plików opiera się na bazie wzorcowych sygnatur, czyli kryptograficznych skrótów plików, które zostały obliczone w chwili instalacji systemu. Na podstawie tej bazy system jest w stanie wykryć, które pliki zostały zmienione przez intruza [13].

W zależności od rodzaju systemu ochrony integralności pliki są sprawdzane okresowo lub w momencie próby odczytu chronionego pliku. Najnowsze rozwiązania tego typu, działające na poziomie jądra systemu operacyjnego, pozwalają nie tylko na pasywne wykrywanie zmian, ale również mogą aktywnie blokować dostęp do plików.

Zastosowanie programów ochrony integralności systemów plików pozwala nie tylko na wykrycie włamania, ale również na znaczne skrócenie czasu potrzebnego na usunięcie wszystkich pozostawionych przez intruza plików. Zamiast kompletnej reinstalacji systemu wystarczy przywrócić oryginalne wersje zmienionych plików.

5. WNIOSKI KOŃCOWE

Obecnie wszystkie komputery podłączone do sieci Internet powinny być wyposażone w systemy bezpieczeństwa. W komputerach osobistych powszechnie stosowane są zapory sieciowe i programy antywirusowe. Systemy serwerowe, które udostępniają usługi poprzez sieć Internet wymagają zastosowania bardziej zaawansowanych zabezpieczeń, gdyż są narażone na większe niebezpieczeństwo. Zastosowanie mechanizmów bezpieczeństwa wymaga dużego nakładu pracy koniecznego ich do wdrożenia i utrzymania.

Przedmiotem badań autorów niniejszego artykułu są systemy ochrony integralności plików. Prowadzone prace mają na celu stworzenie systemu aktywnej ochrony integralności plików wraz z możliwością automatycznego przywracania zmienionych plików. Takie rozwiązanie nie tylko umożliwia wykrycie włamania czy zablokowanie dostępu do zaatakowanych plików, ale również pozwala na ciągłą pracę na niezmodyfikowanym systemie nawet w przypadku skutecznego ataku przez intruza.

6. BIBLIOGRAFIA

1. Chapman D. B., Zwicky E. D, *Building Internet Firewalls*, O'Reilly 2nd editon, 2002
2. Goncalves, M., *Firewalls Complete,* McGraw Hill, Nowy Jork, 1998.
3. Marcus J., Ranum A., *Network Frewall,* World Conference on System Administration and Security, Waszyngton 1992.
4. Ingham K., Forrest S., *A history and survey of network firewalls.*, Tech. Rep. 2002-37, University of New Mexico Computer Science Department, 2002
5. Zalenski R., *Firewall technologies*, IEEE Potentials, Vol. 21, No. 1, pp. 24–29, Feb. 2002.
6. Smalley S., *Configuring the SELinux Policy*, NAI Labs Report #02-007, 2002.
7. Bauer, M., *Paranoid penguin: an introduction to Novell AppArmor*, Linux Journal 148, 2006
8. Lawton K., *Running multiple operating systems concurrently on an ia32 pc using virtualization techniques*. http://www.floobydust.com/virtualization/lawton_1999.txt
9. Liska A., *The Practice Of Network Security. Deployment Strategies For Production Environments*, Prentice Hall PTR, 2002
10. Debar H., Dacier M., Wespi A., *Towards a taxonomy of intrusion detection systems,* Computer Networks, 1999
11. Zhang X., Li C., Zheng, W., *Intrusion Prevention System Design,* The Fourth International Conference on Computer and Information Technology, 2004.
12. Axelsson S., *The Base-Rate Fallacy and its Implications for the Difficulty of Intrusion Detection,* Proceedings of the 6th ACM Conference on Computer and Communications Security, 1999.
13. Kim G., Spafford E., *Experiences with Tripwire: Using Integrity Checkers for Intrusion Detection,* Purdue Technical Report CSD-TR-94-012, 1994.

REVIEW OF THE OPERATING SYSTEMS PROTECTION MECHANISMS

One of the most important tasks of the operating system is to provide computer system security. This paper describes protection mechanisms and technologies for the Linux operating system. We first present methods of active protection through blocking net connections and limiting application privileges. Then we present solutions for minimizing damages inflicted by the intruders. Diversity of security mechanisms is a required condition of effective computer systems protection.