STUDIA INFORMATICA 2003
Volume 24 Number 2A (53)

Tomasz Jordan KRUK[1], Robert MICHALSKI
Politechnika Warszawska, Instytut Automatyki i Informatyki Stosowanej

REAGOWANIE NA INCYDENTY W SYSTEMIE UNIX

Streszczenie. W artykule opisano praktyczne metody i techniki reagowania na incydenty w systemach komputerowych pracujących pod kontrolą systemu Unix. Przedstawiono metody reakcji wstępnej oraz metody analizy dochodzeniowej. Opisano zaimplementowane rozwiązanie własne.

Słowa kluczowe: bezpieczeństwo, analiza powłamaniowa, incydent naruszenia bezpieczeństwa systemu komputerowego.

UNIX FORENSICS TECHNIQUES FOR INCIDENT RESPONSE

Summary. Practical methods and incident response techniques under Unix operating system have been described. Initial response and investigation analysis methods have been presented. Proposed and implemented own solution has been described.

Keywords: security, forensic analysis, computer incident.

1. Wstęp

Czytając cykliczne raporty okresowe SANS/FBI, CERT oraz śledząc listy bugtraq nie sposób się oprzeć wrażeniu, że liczba wykrywanych luk w oprogramowaniu sieciowym, jak i samych systemach operacyjnych nie dość, że nie maleje, to regularnie rośnie. Nie ma znaczenia, czy jest się instytucją rządową, dużą, średnią, czy małą firmą, czy też domowym użytkownikiem Internetu - każdy jest potencjalnym celem ataku.

Prawdopodobieństwo, że dowolna organizacja stanie się w niedługim czasie ofiarą ataku, jest bardzo duże – stąd wiedza na temat odpowiedniej reakcji na taką sytuację jest dziś

[1] Również: NASK, Zakład Jakości Usług Sieci.

niezbędna. Chociaż celem ataku są systemy komputerowe, właściwa reakcja na atak komputerowy jest zadaniem interdyscyplinarnym - oprócz spraw technicznych istotnym elementem jest analiza prawna i uwarunkowania zewnętrzne.

W wielu przypadkach nawet najbardziej złożone i wieloaspektowe problemy reagowania na incydenty[1] można zdekomponować. Analiza wejść i wyjść poszczególnych komponentów prowadzi do procesu, na który składają się [5]:

- przygotowanie przed zdarzeniem - operacje wykonywane przed wystąpieniem incydentu komputerowego,
- wykrycie incydentu - analiza podejrzanego zdarzenia,
- reakcja wstępna - dochodzenie początkowe (zebranie najbardziej ulotnych dowodów, łącznie z zeznaniami świadków, wstępna weryfikacja wystąpienia incydentu),
- formułowanie strategii reakcji - zdefiniowanie najlepszych metod reagowania opartych na znanych faktach,
- dochodzenie - podjęcie działań dochodzeniowych, mających na celu wyjaśnienie, co się stało, kto za tym stoi i jak zapobiec podobnym wydarzeniom w przyszłości,
- implementacja zabezpieczeń - aktywna reakcja w zaatakowanych systemach - zastosowanie zabezpieczeń mających na celu opanowanie sytuacji i wyizolowanie incydentu,
- nadzorowanie sieci - nadzorowanie ruchu w sieci dla celów dochodzeniowych,
- przywrócenie systemu - przywrócenie bezpiecznego i sprawnego działania zaatakowanego systemu,
- tworzenie raportów - precyzyjne udokumentowanie wszystkich etapów dochodzenia i wprowadzonych zabezpieczeń,
- kontynuacja działań - analiza przeprowadzonych operacji.

W dalszej części zostaną omówione dwa elementy tego procesu – prowadzenie reakcji wstępnej i dochodzenia w systemach rodziny UNIX. Przedstawiono również założenia i funkcjonalność zaimplementowanego dedykowanego systemu I.R.E.E. wspomagającego reagowanie na incydenty.

2. Reakcja wstępna w systemach Unix

Faza reakcji wstępnej polega na weryfikacji faktu, że incydent rzeczywiście się wydarzył i określeniu, które systemy zostały pośrednio lub bezpośrednio zaatakowane. Podczas

[1] Definicję pojęcia „incydent", jak i wielu innych można znaleźć w opracowaniu CERT [2].

przeprowadzania operacji należy się kierować zasadą nienaruszalności danych dowodowych. Wszystkie wykonywane operacje powinny być rejestrowane.

2.1. Zestaw narzędzi do analizy

W czasie reakcji wstępnej niezmiernie ważne jest korzystanie ze sprawdzonych poleceń. W celu uniknięcia korzystania z bibliotek w zaatakowanym systemie wszystkie narzędzia należy skompilować statycznie. Niestety, praktycznie każda wersja UNIX i Linux wymaga oddzielnego zestawu narzędzi.

Zestaw do wstępnej analizy, który można umieścić na CD-ROM, powinien obejmować przynajmniej następujące narzędzia:

```
arp, cat, chgrp, chmod, chown, chroot, cksum, cp, cryptcat, cut, date,
dd,des, df, du, echo, env, file, fold, head, hostname, id, ifconfig, ln,
ls, lsof, md5, md5sum, mv, nc, netstat, od, pcat, printenv, pwd, rarp,
read, rm, rmdir, route, sort, sync, tac, tail, touch, uniq, uptime, wc,
who, whoami
```

2.2. Uruchamianie sprawdzonej powłoki systemowej

Pierwszym krokiem w każdym procesie reakcji jest uruchomienie sprawdzonej powłoki systemowej. Po uruchomieniu powłoki należy wpisać do zmiennej środowiskowej PATH ścieżkę do zestawu narzędzi dla badanego systemu, w celu zminimalizowania prawdopodobieństwa uruchomienia niesprawdzonych poleceń.

2.3. Sprawdzenie zarejestrowanych użytkowników

Sprawdzanie użytkowników aktualnie zarejestrowanych w systemie powinno być wykonywane na początku i na końcu każdego procesu reakcji, dzięki czemu można ustalić dokładny przedział czasowy, w którym przeprowadzano akcje w zaatakowanym systemie oraz kto był w nim zalogowany w momencie, gdy zbierano potencjalne dowody. Należy w tym celu użyć polecenia `w` (who).

2.4. Sprawdzenie uruchomionych procesów

Podczas reakcji wstępnej niezmiernie istotne jest zarejestrowanie wszystkich uruchomionych procesów. Można to zrobić za pomocą polecenia `ps`. Dane wyjściowe i sposób użycia mogą się nieco różnić w zależności od wersji systemu UNIX. Jednym z ważniejszych pól polecenia `ps` jest pole START (w Solarisie STIME), pokazujące czas, w którym proces został uruchomiony. Jest ono niezmiernie pomocne, jeżeli znany jest

przedział czasowy, w którym nastąpił atak. Można wtedy zidentyfikować podejrzane procesy na podstawie godziny ich uruchomienia.

Nazwy procesów typu `3\3266632\2377777\1234` wskazują na to, że ktoś przeprowadza aktualnie atak powodujący przepełnienie bufora. Może to oznaczać, że intruz uzyskał nieautoryzowany dostęp do systemu. W takim przypadku należy natychmiast uruchomić polecenie `netstat` i przeanalizować listę aktualnie zestawionych połączeń do systemu.

2.5. Sprawdzenie otwartych portów i plików

Niezastąpionym poleceniem jest w tym przypadku `netstat`. Jednak w zależności od odmiany systemu UNIX zróżnicowana jest metoda dowiadywania się, które aplikacje odpowiadają za konkretne otwarte gniazda sieciowe. W systemie Linux polecenie `netstat` ma opcję *-p*, która kojarzy nazwę aplikacji oraz jej identyfikator procesu z otwartym portem. W systemach Solaris, HP-UX, IBM AIX, FreeBSD, BSDI i Ultrix trzeba w tym celu użyć polecenia `lsof`.

Lsof wyświetla wszystkie aktualnie otwarte pliki, katalogi, biblioteki, strumienie uniksowe i pliki sieciowe (takie jak gniazda NFS) oraz procesy, które je otworzyły. Używając polecenia `lsof` podczas reakcji wstępnej, należy zawsze używać opcji *-D*. W przeciwnym przypadku `lsof` utworzy tymczasowy plik o nazwie.lsof_nazwa_hosta w katalogu domowym użytkownika.

2.6. Wykrywanie ładowalnych modułów jądra

Intruzi korzystają z możliwości ładowalnych modułów jądra (ang. *Loadable Kernel Modules* - LKM) do zmiany sposobu działania poleceń wykonywanych przez administratorów systemów. Odpowiedni LKM, zainstalowany przez intruza, może przechwytywać polecenia, takie jak `netstat`, `ifconfig`, `ps`, `ls` i `lsmod` oraz fałszować ich dane wyjściowe. Użytecznymi narzędziami do ich wykrywania są programy umożliwiające bezpośrednią analizę jądra przez */dev/kmem* takie, jak `kstat` i `ksec`.

2.7. Wykrywanie nieautoryzowanych programów monitorujących

Wykrycie nieautoryzowanego programu monitorującego (ang. *sniffer*) oznacza, że atak może obejmować swym zasięgiem więcej niż jeden system, a atakujący miał prawdopodobnie dostęp do systemu na poziomie użytkownika root. Jednym z elementów, które mogą wskazywać na uruchomienie takiego programu, może być praca karty sieciowej w trybie mieszanym – można to sprawdzić analizując wynik polecenia `ifconfig` w poszukiwaniu flagi

PROMISC. Flaga PROMISC nie występuje we wszystkich odmianach UNIX. System Solaris nigdy nie pokazuje tej flagi po wywołaniu polecenia `ifconfig`. W tym celu trzeba użyć kombinacji poleceń `lsof` i `ps`.

2.8. Przeglądanie systemu plików /proc

Proc jest systemem plików używanych jak interfejs do struktur danych jądra. Każdy proces ma w katalogu /proc swój podkatalog, którego nazwa jest taka, jak identyfikator procesu. Wewnątrz tego katalogu znajdują się informacje, które mogą być bardzo przydatne w procesie dochodzenia. W zależności od wersji UNIX system plików /proc posiada własne unikalne nazwy, np.:

Tabela 1
Różnice nazw elementów systemów plików /proc

Solaris	FreeBSD	Linux	Znaczenie
`object/a.out`	`File`	`exe`	plik programu
`Ms`	`Mem`	`mem`	pamięć procesu
`Map`	`Map`	`maps`	mapa pamięci
...	...	...	...

Łącze `exe` umożliwia odzyskanie usuniętych plików, o ile są one nadal uruchomione. Analizując zawartość katalogu `fd` można zidentyfikować wszystkie pliki otwarte przez proces. Plik `cmdline` zawiera dokładne argumenty wiersza poleceń użyte do uruchomienia aplikacji.

3. Analiza dochodzeniowa w systemie UNIX

Proces prowadzenia dochodzenia jest z założenia ściśle związany ze sformułowaną strategią reagowania. Kluczową kwestią w fazie dochodzenia jest określenie, co i przez kogo zostało uszkodzone. Główny nacisk kładziony jest jednak na ustalenie negatywnych konsekwencji incydentu.

3.1. Analiza plików konfiguracyjnych

Pliki konfiguracyjne UNIX są często otwierane i modyfikowane przez atakujących. Podczas dochodzenia należy dokładnie przejrzeć te pliki i sprawdzić, czy nie wprowadzono przy ich użyciu nieautoryzowanych punktów dostępu np. poprzez rozszerzenie relacji zaufania. Dobra znajomość konfiguracji systemu jest konieczna w dalszych etapach prowadzenia dochodzenia.

3.2. Analiza dzienników zdarzeń

Analiza plików protokołowania daje ogromne możliwości. Konieczna jest jednak pewność odnośnie do integralności posiadanych informacji oraz dobra znajomość konfiguracji systemu ewidencji zdarzeń. Do najbardziej użytecznych dzienników zalicza się:

Tabela 2

Najczęściej wykorzystywane dzienniki zdarzeń

Dzienniki zdarzeń sieci	– Demon Syslog – TCP Wrapper – Dzienniki usług – np. FTP, WWW itd.
Dzienniki zdarzeń hosta	– Historia polecenia *su* – Historia użytkowników *wtmp* lub *utmp* – Historia logowania (*ssh*, *ftp*...) – Historia *cron*
Dzienniki działań użytkownika	– Historia procesów *acct* lub *pacct* – Historia powłoki systemowej

3.3. Określenie czasu, w którym nastąpił incydent

W celu prowadzenia dochodzenia konieczne jest określenie przedziału czasu, w którym nastąpił incydent. Czasami można to zrobić dokładnie, na przykład, gdy atak został wykryty i zarejestrowany przez zewnętrzny system IDS. W takim przypadku pozostaje sprawdzenie zgodności czasów między systemem IDS a zaatakowanym systemem. W innych przypadkach na ustalenie czasu incydentu będą miały wpływ informacje zawarte w dziennikach zdarzeń oraz bezpośrednie informacje uzyskane od administratora czy też użytkowników systemu.

Celem przeglądania informacji typu data/czas jest odtworzenie zdarzeń, które nastąpiły w ustalonym wcześniej przedziale czasowym. Wszystkie pliki lub katalogi: otwarte, zmodyfikowane lub utworzone w tym czasie, mogą mieć związek z incydentem.

3.3.1. Analiza węzłów

Większość instalacji tworzy pliki sekwencyjnie, co oznacza, że numery węzłów takich plików są do siebie zbliżone. Analiza wyników polecenia *ls –lit | sort | more* pozwala ustalić jakie zmiany i kiedy nastąpiły w systemie (dotyczy głównie /usr, /usr/bin, /sbin, /usr/sbin).

3.3.2. Analiza MAC

Analiza czasów MAC (Modified, Access, Change) dla plików daje ogromne możliwości. Połączenie i uporządkowanie atrybutów czasowych pozwala z powodzeniem odtworzyć

podejmowane przez intruza działania. Możliwe jest wykrycie takich operacji, jak: użycie narzędzi typu rootkit, operacje szyfrowania, zestawy LKM, kompilacja programów.

Do przeprowadzenia tego typu analizy można posłużyć się ogólnie dostępnymi poleceniami *ls* oraz *find* z odpowiednimi parametrami lub skorzystać z narzędzi *grave-robber* lub *mac-robber* wykorzystywanych jako wejście programu MAC-time z zestawu TCT.

3.4. Przeglądanie plików i katalogów specjalnych

Istnieją określone typy plików i katalogów, które regularnie pojawiają się przy okazji różnych incydentów. Należą do nich pliki SUID i SGID, nietypowe lub ukryte pliki i katalogi. Analizie poddać należy również zawartość katalogów z plikami konfiguracyjnymi oraz katalogu na pliki tymczasowe */tmp*.

3.5. Odzyskiwanie usuniętych plików

W celu odzyskania usuniętego pliku trzeba odtworzyć w węźle informacje o rozmiarze pliku i listę bloków danych – potrzebna jest znajomość odpowiadającego mu numeru węzła. Jeżeli proces jest nadal uruchomiony, można sprawdzić numer węzła używając polecenia *lsof*. W celu odzyskania tego pliku należy użyć polecenia *icat* z zestawu narzędzi The Coroner's Toolkit (TCT).

Pakiet TCT zawiera inne narzędzia, pomocne przy identyfikacji węzłów. Polecenie *ils* wyświetla informacje o węźle dla każdego pliku w systemie. Lista węzłów może mieć ogromne rozmiary. Aby ograniczyć rozmiar listy, można wyszukać pliki o określonym identyfikatorze użytkownika (UID) lub grupy (GID).

3.6. Analiza podejrzanych plików binarnych

Po procesie reakcji wstępnej i analizie czasowej dysponuje się zwykle podejrzanymi plikami z kodem wykonywalnym bądź danymi w postaci binarnej, których cel i przeznaczenie nie są znane. Istnieją dwie metody ich analizy:

- Analiza dynamiczna – polega na obserwowaniu procesu utworzonego w wyniku uruchomienia podejrzanego pliku binarnego. Stosując techniki opisane w procesie analizy wstępnej należy ustalić, w jaki sposób proces komunikuje się z otoczeniem, jakie deskryptory otwiera, w jaki sposób korzysta z dostępnych zasobów.
- Analiza statyczna – polega na zastosowaniu szeregu narzędzi celem poznania wewnętrznej struktury pliku, co z kolei może prowadzić do ujawnienia przeznaczenia

podejrzanego pliku. Wśród najczęściej wykorzystywanych narzędzi znajdują się: strings, grep, file, nm, ldd oraz programy typu disassembler, debugger.

4. Implementacja systemu wspomagającego reagowanie na incydenty

Celem implementacji było stworzenie dedykowanej platformy programowej wspomagającej procesy reagowania na incydenty w systemach UNIX i Linux. Budowę systemu bazowego oparto na jądrze *Linux 2.4.19* i zestawie *bibliotek GNU.*

4.1. Cechy użytkowe

System IREE (Incident Response Extended Environment) [6] umożliwia prowadzenie reakcji wstępnej i analizy dochodzeniowej na podstawie następujących cech użytkowych:

- dystrybucja zawiera się na jednej samostartującej płycie CD-ROM,
- kompilacji systemu dokonano dla architektury i386,
- właściwy system rezyduje w pamięci operacyjnej, a pozostałe elementy są odczytywane z płyty CD-ROM,
- system zawiera statycznie skompilowane wersje programów narzędziowych dla systemów Linux 2.4, FreeBSD i Solaris 7,
- dane wymagające aktualizacji są pobierane ze stacji dyskietek,
- możliwe jest używanie predefiniowanych konfiguracji pobieranych z dyskietki,
- żadne operacje nie naruszają oryginalnego, badanego systemu plików,
- wszystkie wykonywane w systemie czynności są ewidencjonowane,
- konfiguracja za pomocą przygotowanych skryptów,
- praca w trybie konsoli tekstowej i X-Window.

4.2. Cechy funkcjonalne

System może być wykorzystany w następujących obszarach:

- umożliwienie określenia symptomów zewnętrznego nadużycia,
- odzyskiwanie utraconych danych,
- wspomaganie analizy plików historii,
- prowadzenie testów penetracyjnych,
- prowadzenie inwigilacji sieciowej,
- jako bezpieczne środowisko do analizy dynamicznej.

Zgodnie z powyższymi kategoriami zostały dobrane stosowne programy narzędziowe wspomagające wymienione operacje.

5. Podsumowanie

Proces reagowania na incydenty komputerowe jest metodycznym, złożonym procesem, wymagającym zarówno dużych umiejętności technicznych, jak i pewnych zdolności interpersonalnych. Nie ma dwóch takich samych incydentów. Każdy wymaga nieco innej strategii reagowania. Mimo różnorodności incydentów i stosowanych technik reagowania, możliwe i celowe jest stworzenie spójnej metodologii przeprowadzania reakcji.

Przedstawione elementy procesu reagowania na incydenty w systemie UNIX, mogą być z powodzeniem wykorzystane w dowolnym systemie uniksowym, w szczególności w stworzonym dedykowanym do tego celu systemie I.R.E.E.

LITERATURA

1. Amoroso E. G.: Fundamentals of Computer Security Technology. PrenticeHall PTR Upper Saddle River. New York 1994.
2. Howard D. J., Longstaff T. A.: A Common Language for Computer Security Incidents. Sandia National Labolatories, 1998.
3. Kurtz G., Hatch B., Lee J. B.: Hacking Linux Exposed: Linux Security Secrets and Solutions. Osbourne / McGraw Hill, 2001.
4. Lindqvist U., Jonsson E.: How to Systematically Classify Computer Security Intrusions. Proceedings of the 1997 IEEE Symposium on Security and Privacy. IEEE Computer Society Press. Los Alamitos CA May 1997.
5. Mandia K., Prosise C.: Incident Response: Investigating Computer Crime. The McGrawHill Companies 2001.
6. Michalski R.: Strategie i metody reagowania na łamanie zabezpieczeń komputerowych. Praca dyplomowa. Politechnika Warszawska 2003.
7. SANS/ FBI Top 20 List. http://www.sans.org. Mar 2003.
8. CERT Polska: Raport Roczny CERT Polska 2002. Przypadki naruszające bezpieczeństwo teleinformatyczne. http://www.cert.pl/PDF/Raport_CP_2002.pdf

Recenzent: Dr inż. Krzysztof Nałęcki

Wpłynęło do Redakcji 5 kwietnia 2003 r.

Abstract

The number of computer crime incidents is growing. Probability that any organization will became a victim of computer attack in the nearest future is not neglectable. Therefore forensic incident response techniques should follow some methodologies, related not only to technical but also to legal aspects and to some external dependencies. Two important elements of reconstruction of past events related to computer crime incident, if it already has happened, are: introductory reaction and investigation analysis.

The introductory reaction has to check whether the incident has had really happened and which systems have been compromised. During that step it is extremely important to perform actions with as little distortion as possible.

The main goal of the investigating analysis is to establish which are the probable consequences of the discovered incident. Moreover, during that phase of incident response, investigators try to find an answer to questions about what exactly and by whom has been compromised.

Both steps are supported by usage of some software tools. Both steps may be implemented semi-automatically. Example of the software, I.R.E.E., playing the role of the integration and automation tool has been proposed, implemented and presented by the article authors.

Adresy

Tomasz Jordan KRUK: Politechnika Warszawska, Instytut Automatyki i Informatyki Stosowanej, ul. Nowowiejska 15/19, 00-665 Warszawa, Polska, T.Kruk@ia.pw.edu.pl .

Robert MICHALSKI: Politechnika Warszawska, Instytut Automatyki i Informatyki Stosowanej, ul. Nowowiejska 15/19, 00-665 Warszawa, Polska, R.Michalski.1@elka.pw.edu.pl .