

# Bezpieczeństwo systemów komputerowych

## Intruzi i intruzy

Marcin Peczarski

Instytut Informatyki Uniwersytetu Warszawskiego

11 stycznia 2010

# Intruzi – złośliwi użytkownicy

- ▶ **Przebieraniec** – użytkownik nie posiadający pozwolenia na posługiwanie się danym komputerem, pokonujący środki kontroli dostępu do systemu, aby wykorzystać legalne konto innego użytkownika
- ▶ **Nadużywający** – legalny użytkownik nadużywający przywilejów, uzyskujący dostęp do zasobów, do których dostępu nie powinien mieć
- ▶ **Tajny użytkownik** – użytkownik uzyskujący kontrolę nad systemem w celu ominięcia monitorowania i kontroli dostępu

- ▶ Intrusion Detection System – system wykrywania intruzów
- ▶ Wykrywanie opiera się na założeniu, że zachowanie intruza różni się od zachowania legalnego użytkownika.
- ▶ Technikami automatycznymi można próbować odróżnić przebiegańca od legalnego użytkownika.
- ▶ Wykrywanie metodami automatycznymi osób nadużywających jest trudniejsze.
- ▶ Wykrywanie tajnych użytkowników jest uznawane za znajdujące się poza zasięgiem technik czysto automatycznych.

# Wykrywanie włamań

- ▶ Jeśli włamanie zostanie wykryte dostatecznie szybko, intruz może zostać usunięty, zanim wyrządzi jakąkolwiek szkodę.
- ▶ Już samo zainstalowanie systemu wykrywania włamań może działać odstraszająco.
- ▶ Wykrywanie włamań umożliwia zbieranie informacji o technikach włamywania się i wykorzystanie ich do udoskonalania systemów zabezpieczeń.

# Zapisy działań użytkowników – zapisy kontrolne

- ▶ **Podmiot** – przeważnie użytkownik, czasem proces działający w imieniu użytkownika
- ▶ **Czynność** – operacja wykonywana przez podmiot na przedmiocie lub przy użyciu przedmiotu, np. rozpoczęcie pracy, odczyt, operacja wejścia-wyjścia, wykonanie programu
- ▶ **Przedmiot** – odbiorca czynności, np. plik, program, komunikat, rekord bazy danych, terminal, drukarka
- ▶ **Wyjątek** – ewentualny zgłoszony wyjątek, np. odmowa dostępu
- ▶ **Zużycie zasobów** – zużyty czas procesora, liczba odczytanych lub zapisanych rekordów, liczba wydrukowanych stron itp.
- ▶ **Datownik** – data i czas wykonania czynności

# Analiza zapisów kontrolnych

- ▶ **Statystyczna** – zbieranie danych związanych z zachowaniem legalnych użytkowników w pewnym okresie czasu, zastosowanie testów statystycznych, wykrywanie anomalii statystycznych
  - ▶ progowe – zastosowanie progów niezależnych od użytkownika
  - ▶ profile użytkowników – indywidualny profil dla każdego użytkownika, testowanie odchyleń od profilu
- ▶ **Oparta na regułach** – zestaw reguł odróżniających zachowania legalnych użytkowników i intruzów
  - ▶ wykrywanie anomalii
  - ▶ wykrywanie penetracji

# Progowe wykrywanie anomalii

- ▶ Obliczanie liczby wystąpień danego typu zdarzeń w pewnym okresie czasu
- ▶ Alarm zgłaszany po przekroczeniu pewnej wartości progowej
- ▶ Problem ustalenia progu
  - ▶ fałszywe alarmy
  - ▶ niewykrycia

# Profile użytkowników

## ► Estymowanie

- wartości oczekiwanej

$$\hat{X} = \frac{1}{n} \sum_{i=1}^n X_i,$$

- wariancji

$$\frac{1}{n-1} \sum_{i=1}^n (X_i - \hat{X})^2,$$

- częstości rozpoczynania pracy z systemem w zależności od dnia i czasu,
- częstości rozpoczynania pracy z systemem z różnych miejsc,
- czasu trwania sesji,
- ilości przesyłanych danych,
- wykorzystania zasobów w trakcie sesji,
- częstości wykonywania poleceń,
- częstości operacji plikowych,
- ...



## Profile użytkowników, cd.

- ▶ Metoda wielowymiarowa – estymowanie korelacji wielu parametrów
- ▶ Metoda procesów Markowa
  - ▶ określanie prawdopodobieństwa przejść między różnymi stanami
  - ▶ wykrywanie występowanie sekwencji zdarzeń
- ▶ Metoda ciągów czasowych – wyszukiwanie ciągów zdarzeń zachodzących zbyt szybko lub zbyt wolno
- ▶ Metoda operacyjna – określanie zdarzeń uznawanych za nienormalne, np. za pomocą mierzenia
  - ▶ czasu od poprzedniego rozpoczęcia pracy z systemem,
  - ▶ nieudanych prób uwierzytelnienia,
  - ▶ nieudanych próby uwierzytelnienia pochodzących z konkretnych maszyn,
  - ▶ odmów wykonania,
  - ▶ nieudanych prób dostępu do plików.

# Wykrywanie anomalii na podstawie reguł

- ▶ Reguły określające zachowania użytkowników i programów
- ▶ Analiza zapisów kontrolnych w celu automatycznego wygenerowania zbioru reguł
- ▶ Śledzenie zachowań i porównywanie ze zbiorem reguł
- ▶ Zachowanie nie pasujące do reguł jest podstawą wszczęcia alarmu
- ▶ Niepotrzebna wiedza o architekturze i słabych punktach ochranianego systemu
- ▶ Potrzebna duża baza reguł, rzędu  $10^4$  do  $10^6$

# Wykrywanie penetracji na podstawie reguł

- ▶ Wykorzystanie systemu ekspertowego do wykrywania podejrzanych zachowań
- ▶ Reguły korzystające z wiedzy o słabych punktach systemu
- ▶ Reguły określające znane techniki penetracji
- ▶ Reguły specyficzne dla konkretnej architektury i systemu operacyjnego
- ▶ Reguły generowane przez ekspertów np. na podstawie wywiadów z administratorami lub wiedzy uzyskanej od (nawróconych) hakerów

# Terminologia

- ▶ **true positive** – alarm spowodowany rzeczywistym atakiem
- ▶ **false positive** – fałszywy alarm
- ▶ **false negative** – niewykrycie ataku
- ▶ **true negative** – brak ataku i alarmu

# Fałszywe alarmy

- ▶ Zbyt czuły system powoduje zgłaszanie wielu fałszywych alarmów.
- ▶ Zbyt duży poziom fałszywych alarmów zmniejsza czujność obsługi.
- ▶ Serwery balansujące obciążenie testują dystans do klienta i mogą być przyczyną fałszywych alarmów.
- ▶ Dynamiczne przydzielanie adresów IP powoduje, że możemy otrzymać adres użytkownika, który „źle” się zachowywał w sieci.
- ▶ Wyzwaniem jest utrzymanie niskiego poziomu fałszywych alarmów (ang. false positive rate).

# Systemy wykrywania intruzów

- ▶ Sieciowe (ang. NIDS, network-based intrusion-detection system)
  - ▶ czujniki umieszczane w istotnych węzłach sieci, często w strefie zdemilitaryzowanej (ang. DMZ) lub na granicy administrowanej sieci;
  - ▶ monitorują sieć w poszukiwaniu złośliwego ruchu;
  - ▶ platforma sprzętowa niezależna od reszty infrastruktury sieciowej.
- ▶ Lokalne (ang. HIDS – Host-based intrusion detection system)
  - ▶ programowe agenty (ang. software agent) instalowane na monitorowanych maszynach;
  - ▶ monitorują i zapisują zdarzenia systemowe (wywołania funkcji systemu operacyjnego, modyfikacje plików systemowych).

# Systemy wykrywania intruzów, cd.

- ▶ Pasywne (ang. passive)
  - ▶ wykrywają potencjalne naruszenia bezpieczeństwa;
  - ▶ protokołują zebrane informacje;
  - ▶ zgłaszają alarmy użytkownikom i obsłudze.
- ▶ Reaktywne (ang. reactive)
  - ▶ nazywane są również systemami ochrony przed intruzami (ang. IPS – Intrusion Prevention System);
  - ▶ reagują na podejrzaną aktywność przez resetowanie połączeń, zmianę reguł ścian ogniowych, aby zablokować ruch z podejrzanego złośliwego źródła;
  - ▶ mogą podejmować działania automatycznie lub na polecenie operatora.

# Systemy wykrywania intruzów a ściany ogniowe

- ▶ Ściana ogniowa obserwuje ruch, aby zapobiec ingerencji pochodzącej z zewnątrz.
- ▶ Ściana ogniowa ogranicza ruch między sieciami i nie sygnalizuje ataków mających miejsce wewnątrz sieci.
- ▶ System wykrywania intruzów nie zapobiega atakom, a tylko sygnalizuje fakt ich wystąpienia.
- ▶ System wykrywania intruzów obserwuje również ataki mające miejsce wewnątrz sieci.
- ▶ Reaktywny system wykrywania intruzów, nazywany systemem ochrony przed intruzami (IPS), jest inną formą aplikacyjnej ściany ogniowej.
- ▶ Systemy hybrydowe (ang. IPDS) łączą funkcjonalność wykrywania i zapobiegania.



# Przynęty

- ▶ Garnek miodu (ang. honeypot)
- ▶ Monitorowana pułapka w sieci, symulująca usługi sieciowe.
- ▶ Odwraca uwagę atakującego od bardziej dla nas wartościowych maszyn w sieci.
- ▶ Umożliwia wczesne ostrzeżenie o rozpoczynającym się ataku.
- ▶ Umożliwia szczegółową analizę ataku.

## Przynęty, cd.

- ▶ Smolista dziura (ang. tar pit)
- ▶ Szczególnie lepka pułapka
- ▶ Wpuszcza agresora i przetrzymuje go.
- ▶ Spowalnia działanie agresora i daje czas na reakcję.

# Intruzy – złośliwe oprogramowanie

- ▶ Wymagające programu gospodarza i nierozmnażające się
  - ▶ **boczne wejście** – tajne, nie opisane w dokumentacji wejście do programu, pozwalające na uzyskanie dostępu z pominięciem normalnych procedur uwierzytelniania
  - ▶ **bomba logiczna** – fragment programu uaktywniający złośliwe działania po spełnieniu określonych warunków
  - ▶ **koń trojański** – tajna, nie opisana w dokumentacji procedura zawarta w użytecznym programie
- ▶ Wymagające programu gospodarza i rozmnażające się
  - ▶ **wirus** – kod powodujący wstawienie samego siebie do innego programu, często przenoszący też złośliwy kod zawierający np. bombę logiczną lub konia trojańskiego
- ▶ Niezależne i rozmnażające się
  - ▶ **bakteria** – samopowielający się program, zużywający zasoby systemu
  - ▶ **robak** (ang. worm) – program powielający się i wysyłający swoje kopie za pośrednictwem połączeń sieciowych, najczęściej przenoszący też złośliwy kod zawierający np. bombę logiczną lub konia trojańskiego

# Boczne wejście

- ▶ Pozwala uzyskać dostęp z pominięciem zwykłych procedur kontroli dostępu.
- ▶ Często tworzone przez programistów w celu ułatwienia testowania i usuwania błędów.
- ▶ Tworzone, aby istniała metoda uruchomienia programu, gdy zawiedzie procedura uwierzytelniająca.

# Bomba logiczna

- ▶ Kod zawarty w legalnym programie, który może eksplodować w określonych warunkach.
- ▶ Bomba logiczna może np. wybuchnąć, gdy nazwisko programisty nie pojawi się na dwóch kolejnych listach płac.

# Koń trojański

- ▶ Koń trojański może mieć postać użytecznego lub pozornie użytecznego programu, lub polecenia, zawierającego ukryty kod, który wykonuje niepożądaną lub złośliwą czynność.
- ▶ Służy do pośredniego wykonania funkcji, która nie może zostać wykonana bezpośrednio przez niepowołanego użytkownika.
- ▶ Trudnym do wykrycia miejscem ukrycia konia trojańskiego może być kompilator lub interpretator poleceń.

# Wirus

- ▶ **Pasożytniczy** – dołączający się do plików wykonywalnych, powielający się podczas wykonywania zainfekowanego programu
- ▶ **Rezydentny** – umieszcza się w pamięci operacyjnej jako część jakiegoś programu systemowego
- ▶ **Sektora ładowania** – zaraża program ładujący system operacyjny
- ▶ **Tajny** – potrafiący ukrywać swoją obecność, np. przejmując procedurę obsługi dysku, dostarcza oryginalny niezarażony kod programu
- ▶ **Polimorficzny** – mutujący, utrudniający wykrycie za pomocą sygnatury

# Bakteria

- ▶ Jej głównym celem jest powielanie się.
- ▶ W zasadzie nie niszczy plików.
- ▶ Powielając się wykładniczo, szybko zajmuje całe zasoby systemu.



# Robak

- ▶ Przegląda pliki systemowe i użytkownika w celu odszukania innych komputerów pracujących w sieci.
- ▶ Próbuje połączyć się z innymi komputerami jako legalny użytkownik.
- ▶ Próbuje ataku słownikowego, aby odkryć hasło.
- ▶ Uzyskawszy dostęp do innego komputera, uruchamia na nim swoją kopię.
- ▶ Wykorzystuje luki w usłudze dostarczania poczty, aby przenieść się na inny komputer.

# Metody walki

- ▶ Proste skanery
  - ▶ do rozpoznania wirusa (bakterii, robaka, ...) potrzebują jego sygnatury;
  - ▶ mogą wykrywać tylko znane wirusy;
  - ▶ niektóre sprawdzają, czy długości plików nie uległy zmianie.
- ▶ Skanery heurystyczne
  - ▶ szukają fragmentów kodu, które często wchodzi w skład wirusów (pętla deszyfrująca wirusa, kod samomodyfikujący się);
  - ▶ sprawdzają, czy programy nie zostały zarażone za pomocą funkcji haszującej.
- ▶ Wychwytywanie aktywności
  - ▶ programy rezydentne, rozpoznające wirusy na podstawie ich działalności;
  - ▶ wychwytyją i blokują aktywności związane z próbami zarażenia.
- ▶ Pełna ochrona
  - ▶ pakiety integrujące wiele technik antywirusowych.