STUDIA INFORMATICA 2002
Volume 23 Number 2B (49)

Tomasz Jordan KRUK[1], Jarosław WRZESIEŃ
Politechnika Warszawska, Instytut Automatyki i Informatyki Stosowanej

WYKRYWANIE NIETYPOWYCH SPOSOBÓW SKANOWANIA PORTÓW

Streszczenie. W tematykę wprowadzają metody skanowania portów. Następnie opisano problematykę wykrywania skanowania portów oraz techniki utrudniania wykrywania skanowania. Na koniec opisano trendy w metodologii wykrywania nietypowych skanowań portów.

DETECTION METHODS OF UNCOMMON PORTSCAN TECHNIQUES

Summary. Typical portscan techniques have been enumerated. Next, portscan detection and anti-portscan detection methods have been described. Finally, some trends in the uncommon portscan's detection methods have been presented.

1. Wstęp

Ataki na pojedyncze komputery, małe sieci czy nawet ogromne korporacje są dzisiaj niemalże codziennością. Ochrona przed tymi atakami stała się rutynową, chociaż konieczną czynnością. Stosowanie systemów wykrywania intruzów (IDS) oraz ścian ogniowych (firewall) jest oznaką rozsądku administratora, a nie sygnałem, iż popada on w paranoję.

Intruzi wymyślają coraz więcej sposobów, które mają nas (użytkowników komputerów) pozbawić możliwości normalnej niezakłócanej pracy. Ataki denial of service (DoS) oraz jeszcze bardziej kłopotliwe ataki distributed denial of service (DDoS) stają się zmorą administratorów. Przed tymi atakami nie ma na razie ochrony, ich skutkiem jest bowiem zablokowanie łącza komunikacyjnego, z którego korzystamy. Tak więc w trakcie trwania ataku jedynym ratunkiem jest interwencja u dostawcy internetowego, który przydziela nam

[1] Również: NASK, Zakład Jakości Usług Sieci

inne łącze na czas trwania ataku, jednocześnie starając się wyśledzić źródło bądź źródła, z których przychodzą niepożądane pakiety. Inne ataki mogą koncentrować się na konkretnych usługach udostępnianych w sieci, mając na celu zachwianie stabilności serwera takiej usługi bądź też całego systemu operacyjnego, pod którego kontrolą serwer jest uruchomiony.

Aby nie atakować na oślep, haker musi sprawdzić, czy w naszej sieci takie usługi są udostępnione, czy systemy operacyjne, z których korzystamy, będą podatne na stosowany atak oraz czy wersje używanych serwerów usług mają jakieś błędy, dziury bezpieczeństwa, które dodatkowo ułatwią przeprowadzenie ataku. Informacje takie zwykle zdobywane są za pomocą tzw. skanowania portów (portscanning).

Ta rozwijająca się szybko metoda ataku znana była od kiedy pojawili się hakerzy. Dopiero teraz jednak zaczyna ona nabierać cech, które powodują, iż wykrycie takich ataków staje się utrudnione bądź wręcz niemożliwe przy użyciu dzisiejszych narzędzi. Powstają coraz bardziej zaawansowane narzędzia umożliwiające skanowanie portów. Jedyną metodą, aby bronić się przed nimi, jest stworzenie narzędzi umożliwiających wykrywanie najbardziej nietypowych skanów.

2. Rodzaje skanowania portów

Na wielu grupach dyskusyjnych poruszany jest problem legalności i etyczności skanowania zdalnej sieci bez zgody jej właściciela. Jak na razie problem ten leży w szarej strefie i nie jest regulowany przez żadne przepisy. Jednakże praktyka pokazuje, że prawie każda próba skanowania portów jest przeprowadzana z komputerów, które zostały przechwycone przez hakera bez wiedzy właścicieli, dlatego powinna być traktowana jako potencjalne zagrożenie i ważne jest, aby została wykryta.

Haker przeprowadzający skanowanie może mieć dwa główne cele:

- zalanie IDS alarmami (IDS flooding) w celu odwrócenia jego uwagi lub uniemożliwienia mu poprawnej pracy,
- zbieranie informacji o dostępności i statusie określonych kombinacji adresów IP i portów (TCP lub UDP).

2.1. IDS flooding

IDS flooding jest nieskomplikowaną metodą polegającą na wysyłaniu dużej liczby zapytań dotyczących portów - najczęściej z jednego adresu i najczęściej w uporządkowany sposób. Haker sprawdza dużą liczbę portów w krótkim czasie, przez co od razu wiemy, że mamy do czynienia ze skanowaniem portów. Wysyłanie zapytań z jednego adresu i w

uporządkowany sposób również pomaga w sklasyfikowaniu takiego ataku jako skanowanie portów. Ponieważ tematem artykułu jest wykrywanie nietypowych rodzajów ataków, w dalszych rozważaniach IDS flooding zostanie pominięte.

Dużo więcej problemów nastręcza wykrycie ataków mających na celu zbieranie informacji.

2.2. Skanowanie w celu zdobycia informacji

Każda próba skanowania może zostać określona dwoma parametrami:

- zakres skanu (scan footprint), czyli zestaw kombinacji adresów IP i portów, którymi zainteresowany jest atakujący,
- typ skanu, czyli jakie metody wykorzystywane są do zbierania informacji.

2.2.1. Zakresy skanów

Najczęściej obecnie spotykanym skanem jest skan o zakresie poziomym (horizontal scan). Rozumiemy przez to, że atakujący ma program mogący zaszkodzić określonej usłudze i jest zainteresowany znalezieniem komputera, który tę usługę udostępnia. Skan ten obejmuje więc jeden numer portu na wszystkich adresach IP z określonego przedziału. W chwili obecnej najczęściej skanowanym w ten sposób portem jest port 53 (DNS), ale są również i inne, np. 139 (NetBIOS), 98 (linuxconf), 23 (telnet). Rozkład portów będących w centrum zainteresowania zmienia się w czasie, tak jak zmienia się popularność programów służących do atakowania różnych usług.

Spotyka się również skany o zakresie pionowym (vertical scans). Są to skany, przy których atakujący skanuje część lub wszystkie porty na jednym komputerze. Celem takiego ataku może być sprawdzenie, które usługi pracują na komputerze, aby możliwy był wybór tej, którą najłatwiej później zaatakować.

Skan może składać się ze skanów o zakresie poziomym i pionowym tworząc skan blokowy (block scan), mający na celu sprawdzanie kilku usług na kilku komputerach.

2.2.2. Typy skanów

Oprócz zakresu, każdy skan posiada określony typ pojedynczych skanowań. Znanych jest kilka takich typów.

Biorąc pod uwagę protokół TCP, najprostszym typem skanu jest wywołanie przez narzędzie skanujące pełnego trójfazowego powitania, czyli sekwencji inicjującej sesję TCP z wykorzystaniem bitów SYN oraz ACK. Jeśli powiedzie się, oznacza to, że port jest otwarty. W przeciwnym przypadku port jest zamknięty bądź filtrowany przez filtr pakietów albo firewall. Ten typ skanu jest czasem używany przez konsultantów do spraw bezpieczeństwa,

ale z powodu dużej wykrywalności rzadko obserwuje się go w rzeczywistych atakach. Dodatkową wadą z punktu widzenia atakującego, wynikającą z poprawnego zakończenia wywołania po stronie serwera funkcji accept(), jest możliwość wygenerowania logu, który jest niepożądanym śladem aktywności hakera na atakowanym komputerze. Z tego powodu najpopularniejszym typem skanu jest synscan. W tym przypadku narzędzie skanujące generuje jedynie początkowy pakiet syn. Jeśli dotrze on do otwartego portu, serwer odpowie pakietem syn-ack. Jeśli port będzie zamknięty serwer odpowie pakietem reset. Jeśli nie będzie odpowiedzi, oznacza to, że port jest oddzielony od skanera systemem firewall. Współczesne narzędzia skanujące są w stanie wysyłać pakiety syn bardzo szybko i do wielu komputerów na raz. Następnie odbierają informacje asynchronicznie. Ponieważ przy skanie tego typu po stronie serwera nie dochodzi do pełnego trójfazowego powitania, wywołanie funkcji accept() po stronie serwera nie kończy się powodzeniem. To z kolei pociąga za sobą brak logu ataku na maszynie serwera.

Spotyka się również wiele innych typów skanów. Na przykład skanowanie syn-fin polega na wysłaniu pakietu z ustawionymi flagami syn i fin. Taka kombinacja nie zdarza się nigdy w normalnym ruchu TCP. Atakujący ma w tym przypadku nadzieję, że niepoprawnie skonfigurowany filtr pakietów, który nie przepuściłby pakietu syn, nie zauważy pakietu syn-fin.

Dodatkową ważną możliwością wykorzystania skanowania TCP jest identyfikacja systemu operacyjnego (OS). Opiera się ona na fakcie, że RFC opisujące TCP/IP nie określają, jak obsługiwać niepoprawne kombinacje flag w pakietach. Dlatego też różne implementacje obsługują je odmiennie, a te odmienności mogą być używane do określania, jaki OS jest używany na komputerze o określonym numerze IP.

Skanowanie protokołu UDP różni się od skanowania TCP. Jako protokół bezpołączeniowy UDP nie pozwala na zbieranie informacji o otwartych portach. Po prostu serwer, który ma otwarty port, nie odpowie na pakiet UDP, gdyż jest to dla niego pakiet błędny. Atakujący dostanie natomiast odpowiedź na pytanie, który port jest zamknięty, gdyż serwer odpowie wówczas pakietem ICMP typu 3 i kodu 3 (Destination Unreachable because Port Unreachable). Ponieważ przy skanowaniu UDP istnieje konieczność nasłuchiwania pakietów ICMP, jest ono powolne i nie zawsze można na nim polegać. Powolność wynika z faktu, że komputery mogą ograniczyć częstotliwość, z jaką wysyłają pakiety ICMP. Brak zaufania do tego skanu wynika natomiast z faktu, że pakiety ICMP mogą być zgubione w sieci.

Z punktu widzenia sieciowego IDS (NIDS) wszystkie typy skanów wykorzystujące nieprawidłowe ustawienie flag pakietu są bardzo proste do wykrycia. Reguły, które po prostu wyłuskują z sieci każdy pakiet z nieprawidłową kombinacją flag, wykryją taki pakiet

bezbłędnie. Dlatego też takie skany, chociaż potocznie nazywane ukrytymi (stealth scans), wcale nie są tak dobrze ukryte dla IDS-ów. NIDS ma dużo większy problem z wykryciem skanów wykorzystujących pełne połączenie, wykorzystujących pakiety syn oraz skanów UDP, przy których wykryte pakiety mogą być zwykłym ruchem sieciowym.

3. Problematyka wykrywania skanowania portów

Wykrycie skanowania portów wymaga od programu detektora zbierania różnego rodzaju informacji na temat pakietów sieciowych. Dopiero analiza tych danych pozwala na stwierdzenie, czy jesteśmy atakowani. Atakujący natomiast będzie starał się ukryć fakt skanowania poprzez utrudnienie nam zbierania tych informacji.

3.1. Dane zbierane przez detektory skanów

Przy wykrywaniu skanów pomocne jest określenie, jak duży jest zakres skanu. Mówi to nam, jak trudny do wykrycia będzie dany skan. Przykładowo, skan każdego portu na każdym komputerze z całej sieci klasy C wymaga od atakującego sprawdzenia 16.646.144 kombinacji IP/port. Taki skan trudno jest ukryć. Dla kontrastu atakujący, którego celem jest pojedynczy port na jednym komputerze, może dużo łatwiej przejść niezauważony.

Najprostszą metodą określenia zakresu skanu jest po prostu policzenie kombinacji IP/port, które musi sprawdzić atakujący. Wartość tę nazywamy całkowitą wielkością zakresu (total footprint size). Z punktu widzenia atakującego, wartość ta jest dobrym określeniem wielkości przeprowadzanego skanu, gdyż jest bezpośrednio związana z ilością bitów informacji o skanowanej sieci. Nie jest tak jednak zawsze. Czasami atakujący stara się pozyskać pewną liczbę bitów informacji z każdej kombinacji IP/port. Przykładem może być detekcja systemu operacyjnego. Wówczas korzystnie jest wprowadzić wartość całkowitej informacji zakresu (total footprint information), czyli liczby wszystkich bitów informacji, które atakujący zamierza zdobyć przez skanowanie.

Istnieją również inne wartości przydatne przy wykrywaniu ataków skanowania portów. Niektóre części przeprowadzanego skanu są łatwiejsze w detekcji niż inne. Oczywiste jest na przykład, że gdy skanowany port jest otwarty, znacznie trudniej jest nam zaklasyfikować to jako atak, niż gdy skanowany port jest zamknięty. Dzieje się tak dlatego, że zwykły ruch sieciowy bardzo rzadko trafia do zamkniętego portu. Zwykli użytkownicy Internetu nie starają się szukać usług łącząc się z danym portem na komputerze i sprawdzając, czy dana usługa jest aktywna. Raczej polegają na różnego rodzaju wskazówkach, które mówią im, gdzie dana usługa jest dostępna (np. odsyłacze na stronach WWW, informacje z serwerów

DNS, informacje o serwerach pocztowych przeznaczone dla użytkowników ręcznie konfigurujących swoje programy pocztowe). Dlatego też połączenia do zamkniętych portów są znacznie bardziej podejrzane z punktu widzenia detekcji niż połączenia z otwartymi portami. Kolejną wartością, która jest nam pomocna, jest więc wielkość zakresu zamkniętego (closed footprint size), czyli liczba kombinacji IP/port, przy których w czasie skanowania port był zamknięty.

W rzeczywistości, jednym z podejść do wykrywania skanowania portów jest analizowanie nie pakietów skanu (zwykła detekcja), tylko pakietów, które mogły być odpowiedzią na pakiety skanu z zamkniętych portów serwera; pakiety reset w przypadku skanów TCP oraz pakiety ICMP port niedostępny w przypadku skanów UDP. Taki sposób detekcji określa się jako detekcja wsteczna. Zaletą wstecznej detekcji jest to, że pakiety odpowiedzi są zwykle dużo bardziej niezwykłe niż pakiety skanu, a więc są łatwiejsze do wykrycia. Wadą natomiast jest to, że przy detekcji wstecznej nie zbieramy informacji na temat pakietów skanu wysłanych do nieistniejących adresów IP, a są one w szczególności ciekawe, gdyż od razu informują o sytuacji, która nie jest normalnym zachowaniem sieci. Zarówno zwykła, jak i wsteczna detekcja są wartościowe i uzupełniają się wzajemnie.

Podane powyżej wartości określające skan są pomocne przy jego wykrywaniu. Najbardziej jednak interesującą wartością, która zaczyna odgrywać coraz większą rolę we współczesnych narzędziach wykrywających skanowanie portów, jest tzw. wskaźnik anomalii pakietu (anomaly score), czyli liczbowe określenie, jak bardzo dany pakiet nie pasuje do przeciętnego ruchu sieciowego. Wartość tego wskaźnika określa się znając rozkład prawdopodobieństwa normalnego ruchu sieciowego do portów i komputerów w chronionej sieci (zwykle możliwe jest jedynie przybliżone określenie jego wartości). Następnie, gdy natkniemy się na dowolny pakiet dla danego gniazda (portu/IP), możemy określić prawdopodobieństwo z jakim normalny ruch sieciowy trafiłby do tego gniazda. Jeśli kombinacje port/IP oznaczymy przez x, to prawdopodobieństwo to możemy oznaczyć $P(x)$. Następnie możemy wyznaczyć wskaźnik anomalii $A(x)$ dla danego pakietu ze wzoru:

$$A(x) = -\log P(x) \tag{1}$$

Cały zakres skanu będzie identyfikowany przez zbiór X obejmujący poszczególne x (kombinacje port/IP). Dlatego też całkowitą anomalię skanu można określić wzorem:

$$A(X) = \sum_{x \in X} A(x) \tag{2}$$

Wartość $A(X)$ jest najczęściej różna dla każdych dwóch różnych sieci. Nawet gdy są skonfigurowane w identyczny sposób, zawsze będą występowały różnice chociażby w intensywności ruchu sieciowego. Gdy ruch jest duży, identyczny skan „zdobędzie" większy

wskaźnik anomalii, gdyż będzie on bardziej odstępował od ruchu normalnego w chronionej sieci.

3.2. Sposoby na utrudnianie detekcji skanu

Atakujący chce zwykle uniknąć wykrycia, dlatego też stosuje techniki mające na celu oszukanie bądź chwilowe oślepienie IDS-ów. Różne techniki przedstawiono poniżej.

Zmiana kolejności skanu: Największa liczba skanów występujących w sieci sprawdza po kolei adresy IP przechodząc od najniższych do coraz wyższych wartości. Niestety, programy mające wykrywać ataki nie mogą zakładać, że atakujący zawsze będzie tak postępował. Istnieją już narzędzia do skanowania portów, mogące losowo wybierać adresy IP, na których ma być przeprowadzony skan. Dodatkową trudnością przy wykrywaniu jest fakt, że atakujący jeśli podejrzewa, że używany algorytm zakłada jakąś kolejność skanu, może tak ustawić kolejność, by IDS nie poradził sobie z jego wykryciem.

Spowolnienie skanowania: Atakujący może wydłużyć czas pomiędzy skanami poszczególnych komputerów. Czyniąc to w łatwy sposób oszuka stosowane obecnie proste detektory, opierające się na wykrywaniu określonej liczby ataków w określonym czasie. Spowolniony skan zmusza detektor do zapamiętywania pakietów z dłuższego okresu, przez co utrudnia ich analizowanie.

Losowe odstępy czasu: Jeśli atakujący wprowadzi losowe odstępy czasu pomiędzy poszczególnymi skanami w całym ataku, może utrudnić pracę IDS-om opierającym się na założeniu deterministycznych odstępów pomiędzy tymi skanami.

Losowe ustawianie flag: Pola takie, jak numer sekwencyjny, numer ACK, IP i port źródłowy, są w obecnych skanach najczęściej niezmienne. Powoduje to określoną tendencję wśród twórców oprogramowania IDS polegającą na tworzeniu algorytmów zakładających niezmienność tych pól w ramach skanu. W przyszłości atakujący mogą ustawiać te wartości losowo, aby utrudnić detekcję skanu.

Zmiana adresu źródłowego: Adres źródłowy jest trudniej zmodyfikować. Atakujący musi wysyłać pakiety z prawidłowym adresem, aby być w stanie odbierać odpowiedzi skanowango serwera. Wyjątkiem jest przypadek, gdy atakujący ma możliwość monitorowania sieci blisko sieci monitorowanej, np. dostawcę Internetu (ISP). W takim przypadku ma on możliwość dowolnego ustawiania adresów źródłowych, gdyż odpowiedzi zdobywa przez monitorowanie sieci i wyławianie pakietów z odpowiedziami. Ponieważ rzadko zdarza się, że atakujący ma dostęp do ISP, technika zmiany adresu może być stosowana jako „zasłona dymna" w celu utrudnienia określenia skąd nadchodzi atak.

Skanowanie rozproszone: Atakujący, który prowadzi skan z różnych prawdziwych adresów IP, może analizować różne części zakresu skanu z różnych miejsc. Utrudnia to

proces detekcji skanu. W ekstremalnych przypadkach do skanowania portów mogą być użyte całe sieci programów, podobnie jak w przypadku ataków DDoS.

4. Różne podejścia do detekcji skanów

Najprostszym podejściem do wykrywania skanów jest zbieranie informacji o pakietach skanu przez określony okres. Jeżeli w ciągu *Y* sekund (okno detekcji) wystąpi *X* skanowań oznacza to, że jesteśmy ofiarą ataku skanowania portów. Podejście takie jest bardzo uproszczone i w wielu przypadkach nieskuteczne. Inna rzeczą, na którą zwracają uwagę programy wykrywające są nieprawidłowe ustawienia flag TCP.

Tego typu podejście stosowane jest w jednym z rozszerzeń (plug-in) programu Snort w tzw. Snort portscan preprocessor. Jego wadami są: nieumiejętność wykrycia skanowań pochodzących z różnych adresów źródłowych oraz fakt, że wartości *X* i *Y* ustawiane są ręcznie i nie są dynamiczne - co powoduje, iż jedynie część ataków jest rejestrowana.

Inny algorytm zastosowano w programie NSM (Network Security Monitor). NSM był pierwszym NIDS i jako pierwszy oferował wykrywanie skanowania portów. Algorytm w nim zastosowany polegał na wykrywaniu pakietów z adresów źródłowych, z których wykonano połączenie do 15 lub więcej adresów zdalnych. Opierał się więc również na statycznych wartościach.

Następnym programem, już znacznie bardziej zaawansowanym, był GrIDS (Graph Based Intrusion Detection System). Wykrywał skanowanie budując graf aktywności sieciowej, w którym wierzchołki reprezentowały komputery, a krawędzie ruch pomiędzy nimi. Skan mógł być więc reprezentowany przez krawędź pomiędzy komputerem atakującego a skanowanym serwerem. Na podstawie takiego grafu możliwe było wykrycie skanów pochodzących z jednego numeru IP. W praktyce GrIDS stosował reguły opierające się na parametrze czasowym i jedynie szybkie skany były poprawnie wykrywane. Dużą wadą była również nieumiejętność wykrywania przez program skanów pochodzących z wielu adresów źródłowych.

Program EMERALD, wyprodukowany przez firmę SRI International, również był wykorzystywany do wykrywania skanów i stosował odmienny algorytm. Każdy adres IP jest tam rozpatrywany jako obiekt. EMERALD buduje profil statystyczny każdego obiektu, a następnie porównuje ważony profil krótkoterminowy z ważonym profilem długoterminowym. Skan był wykrywany, gdy porównywane wartości silnie się od siebie różniły.

Obecnie w firmie Silicon Defense trwają prace nad programem Spice (Stealthy Portscan and Intrusion Corelation Engine). Założeniem jego twórców jest opracowanie algorytmu

mogącego bezbłędnie wykryć skan, któremu atakujący nadał pewną losowość i który został ukryty pośród tygodni lub nawet miesięcy zwykłego ruchu sieciowego.

Jeżeli wykorzystanoby podejście z małym oknem detekcji, niezauważone zostaną powolne skany. Z drugiej strony, duże okno wymusza analizowanie ogromnych ilości danych.

Dlatego też Spice stosuje podejście wykrywania anomalii w ruchu sieciowym. Zapamiętywane są tylko te pakiety, które mają dostatecznie wysoki wskaźnik anomalii i tylko one są później analizowane. Dodatkowo, czas przechowywania pakietów również zależy od ich wskaźnika anomalii. Dla przykładu, jeżeli na maszynie pracującej z systemem Windows zostałby odebrany pakiet TCP syn do portu 98, który jest portem linuksowego programu linuxconf, będzie on przechowywany znacznie dłużej niż pakiet wysłany do portu 80 (http) na znanym serwerze WWW.

Zbierane pakiety są następnie grupowane ze względu na podobieństwa. Grupy pakietów są również przechowywane dłużej niż pojedyncze pakiety, dzięki czemu rozłożony w czasie skan zostanie zauważony i zgrupowany, a zwykły ruch sieciowy zostanie stosunkowo szybko usunięty z bufora i nie będzie brany pod uwagę podczas rozpoznawania ataków.

Spice składa się z dwóch modułów: sensor anomalii i korelator. Sensor bada ruch sieciowy i przypisuje pakietom wskaźniki anomalii. Następnie te zdarzenia, które są dostatecznie silnie różne od normalnego ruchu sieciowego, są przekazywane do korelatora. Korelator grupuje wydarzenia i po przeanalizowaniu ich zgłasza ewentualne skany.

Jak na razie implementacji doczekał się jedynie sensor. Trwają prace nad korelatorem.

5. Podsumowanie

Techniki wykrywania skanowania portów nabierają coraz większego znaczenia. Oprogramowanie Spice jest na razie jedyne w swoim rodzaju, jednak na pewno zaczną powstawać i inne produkty. Spice stosuje dość zaawansowane algorytmy probabilistyczne, jednakże żadne ze stosowanych tam rozwiązań nie opiera się na rewolucyjnie nowych odkryciach. Grupa ludzi, którzy zebrali się w Silicon Defense, postanowiła zrobić użytek z istniejących teorii. Wykorzystując doświadczenia swoje, byłych hakerów oraz przeprowadzając szeroko zakrojone testy w sieci stworzyli podstawy algorytmów mających pomóc w tworzeniu i rozwijaniu narzędzi pomocnych przy wykrywaniu nietypowych ataków sieciowych.

Narzędzia podobne do Spice będą powstawać w przyszłości. Możliwe, że inni producenci zaczną wprowadzać bardziej zaawansowane algorytmy, np. opierające się na sieciach neuronowych. Uporanie się z problemem skanowania portów pozwoli zapobiegać atakom, a nie tylko wykrywać je i próbować się przed nimi bronić. Dlatego też producenci systemów

wykrywania intruzów już niedługo zaczną się prześcigać w opracowywaniu coraz to szybszych i bardziej skutecznych aplikacji wspomagających wykrywanie skanowania portów.

LITERATURA

1. Stuart S. J., Hoagland A., McAlerney J. M.: Practical Automated Detection of Stealthy Portscans. http://www.silicondefence.com
2. Kruk T. J., Machnacz A.: Systemy wykrywania anomalii w zabezpieczeniach technicznych. Studia Informatica, vol. 22, No 4(46), Gliwice 2001
3. Kruk T. J.: Snort – darmowy system IDS. IT.FORUM SECURE' 2001, Warszawa 2001
4. Fyodor: The Art of Scanning, Phrack 51. http://www.phrack.com
5. Dokumentacja nmap. http://www.insecure.org/nmap/
6. Dokumentacja TCPdump. http://www.tcpdump.org/

Recenzent: Dr inż. Andrzej Białas

Wpłynęło do Redakcji 25 marca 2002 r.

Abstract

In the paper some aspects of the portscan and anti-portscan techniques have been presented. Firstly, goals of scanning have been enumerated. Secondly, scanning techniques having as a goal information retrieving has been thoroughly described. Scopes and types of scans have been presented.

As the next point, the paper concentrates on scan detection methods. Scan detectors have been described and basic detection definitions like total footprint size, total footprint information and closed footprint size have been introduced.

The next section considers different aspects of possible obstructions of scan detection actions. Changing order of scanned ports, programmable slowdown or stochastic time slice between consecutive scans are mentioned among other scan stealthing methods.

Finally, main features of existing anti-scan products are presented and some trends, mainly based on correlated in time anomaly detection techniques, in the uncommon portscan's detection methods have been presented.