

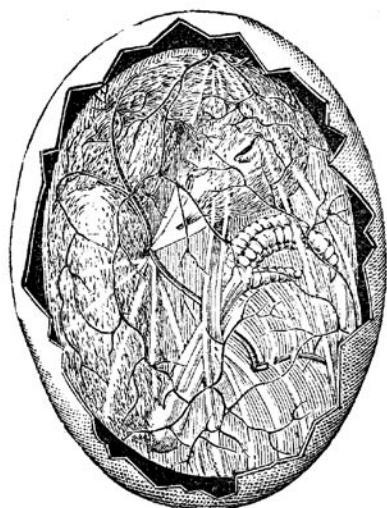
hakin9

Sprzątanie pajęczyn – detekcja nielegalnego współdzielenia łącza

Mariusz Tomaszewski, Maciej Szmit, Marek Gusta

Sprzątanie pajęczyn – detekcja nielegalnego współdzielenia łącza

Mariusz Tomaszewski, Maciej Szmit, Marek Gusta



Pajęczarze, czyli osoby dzielące połączenia internetowe niezgodnie z umową potrafią napsuć sporo krwi dostawcom i administratorom sieci. Istnieje jednak mnóstwo sposobów na wykrywanie takich praktyk. Nie są to metody ani szczególnie skomplikowane, ani czasochłonne.

Z przypadkiem nadmiernego obciążenia łącza internetowego administrator może sobie w prosty sposób poradzić, poprzez podział przepustowości pomiędzy legalnych użytkowników. Wtedy nie musimy się martwić faktem, że ktoś oddaje część swego łącza sąsiadowi (patrz Ramka *Udostępnianie łącza*) – nie będzie to miało żadnego wpływu na jakość działania sieci. Pozostaje jednak problem dzielenia kosztów i zarabiania na takim pośredniczeniu.

Pojawia się zatem pytanie: w jaki sposób administrator może wykryć, że z sieci korzystają osoby trzecie? Technik jest kilka, mniej lub bardziej skutecznych. Wszystko jednak zależy w dużej mierze od wiedzy osoby, która buduje nielegalną pajęczynę i od tego, jakie techniki zastosuje, by spróbować ukryć ten fakt przed światem zewnętrznym.

Pierwszym i w zasadzie najrozsądniejszym sposobem zabezpieczenia się przed nielegalnym dzieleniem łącza, jest podział pasma transmisyjnego. Gwarantuje on, że przepustowość naszej sieci nie okaże się za mała wobec pojawienia się licznych nieautoryzowanych użytkowników, a kwestia wykorzystania przydzielonego pasma pozostaje w gestii klienta.

Jeśli jednak ograniczanie pasma lub limit transferu nam nie wystarcza i wyraźnie nie życzymy sobie, aby udostępniane przez nas łącze było jeszcze przez kogoś dzielone, możemy analizować ruch w naszej sieci i próbować takie sytuacje wykrywać. Jeśli w umowie na korzystanie z łącza dostawca zabrania dalszego jego podziału, można użytkownika uprawiającego taki proceder po prostu z sieci odłączyć – oczywiście jeśli uda się fakt współdzielenia łącza wykryć. Jednak działania takie, jak pokazuje praktyka, łatwo mogą zmienić się w zabawę w poli-

Z artykułu dowiesz się...

- jak ukrywać nielegalne współdzielenie łącza internetowego,
- jak wykrywać nieuprawnione dzielenie pasma internetowego.

Powinieneś wiedzieć...

- powinieneś umieć korzystać z systemu Linux,
- powinieneś znać model ISO/OSI,
- powinieneś mieć przynajmniej podstawową wiedzę o sieciach TCP/IP.

Udostępnianie łącza

Wiele osób, szczególnie tych, które z Linuxem miały niewiele wspólnego, do współdzielenia połączenia wybierze bardzo prostą metodę opartą o system Windows – funkcję *Udostępnianie połączenia internetowego (Internet Connection Sharing – ICS)*. Dzięki niej komputery w sieciach domowych lub biurach można łączyć z Internetem przy użyciu jednego połączenia sieciowego.

ICS jest wbudowaną funkcją systemu Windows, ale można włączyć ją tylko na komputerach z systemem Windows XP, Windows 98 SE, Windows Millennium Edition (Me) lub Windows 2000. Tak naprawdę funkcja ICS to zbiór pewnych składników, które w przeciwieństwie do systemu Linux nie są dostępne dla użytkownika i mają bardzo ograniczoną możliwość konfiguracji. Do najważniejszych składników należą:

- program przydzielający adresy DHCP – bardzo uproszczona usługa DHCP, która przypisuje adres IP, domyślną bramę i nazwę serwera w sieci lokalnej,
- serwer proxy DNS, którego zadaniem jest translacja nazw domenowych na adresy IP w imieniu klientów sieci lokalnej,
- translator adresów sieciowych, który dokonuje translacji adresów prywatnych na adres publiczny (adresy publiczne).

W systemach Linux wykorzystuje się mechanizm translacji adresów sieciowych NAT (ang. *Network Address Translation*) lub stosuje serwery proxy. NAT i proxy są technologiami stosowanymi w systemach firewall, a ich podstawowym zadaniem jest ukrycie i ochrona sieci lokalnej przed sieciami zewnętrznymi.

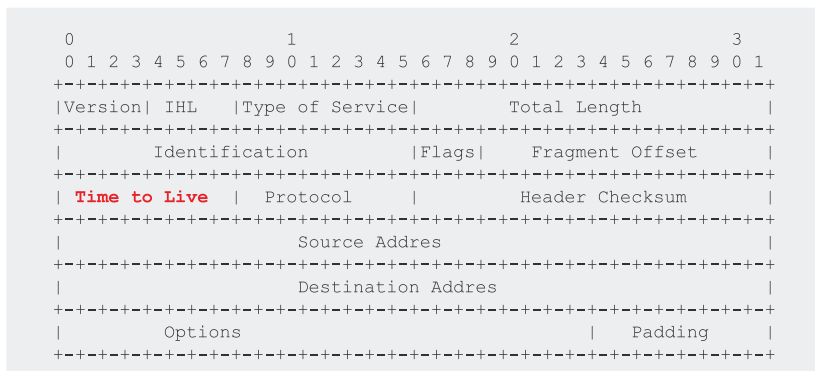
cjań i złodziei, przy czym – jak to zwykle bywa – przewaga inwencji jest po stronie tych ostatnich.

Wartości TTL w nagłówkach pakietów IP

Nagłówek datagramu IP posiada pole TTL – *czas życia* (ang. *time to live*), które określa górną granicę liczby ruterów, przez jakie datagram może przejść podążając do miejsca przeznaczenia (patrz Rysunek 1). Podczas przetwarzania nagłówka datagramu każdy ruter ma obowiązek zmniejszyć pole TTL o wartość proporcjonalną do czasu jego przetrzymywania. Ponieważ routery

praktycznie przetrzymują datagramy przez mniej niż jedną sekundę, pole TTL zmniejszane jest o jeden. Kiedy wartość ta spadnie do zera, datagram jest odrzucany i usuwany z sieci, a nadawca otrzymuje komunikat ICMP o błędzie.

Działanie takie ma na celu zapobieganie nieskończonemu krążeniu w sieci pakietów, które utknęły w pętli rutowania (tzn. jeden ruter przesyła datagram do drugiego, a ten odsyła go z powrotem). Jeśli z jakichś powodów pakiet IP nie może być dostarczony do miejsca przeznaczenia, to po osiągnięciu przez pole TTL wartości 0 zostanie on po prostu usunięty z sieci. Różne systemy operacyjne korzystają z innych



Rysunek 1. TTL (time to live) w nagłówku IP

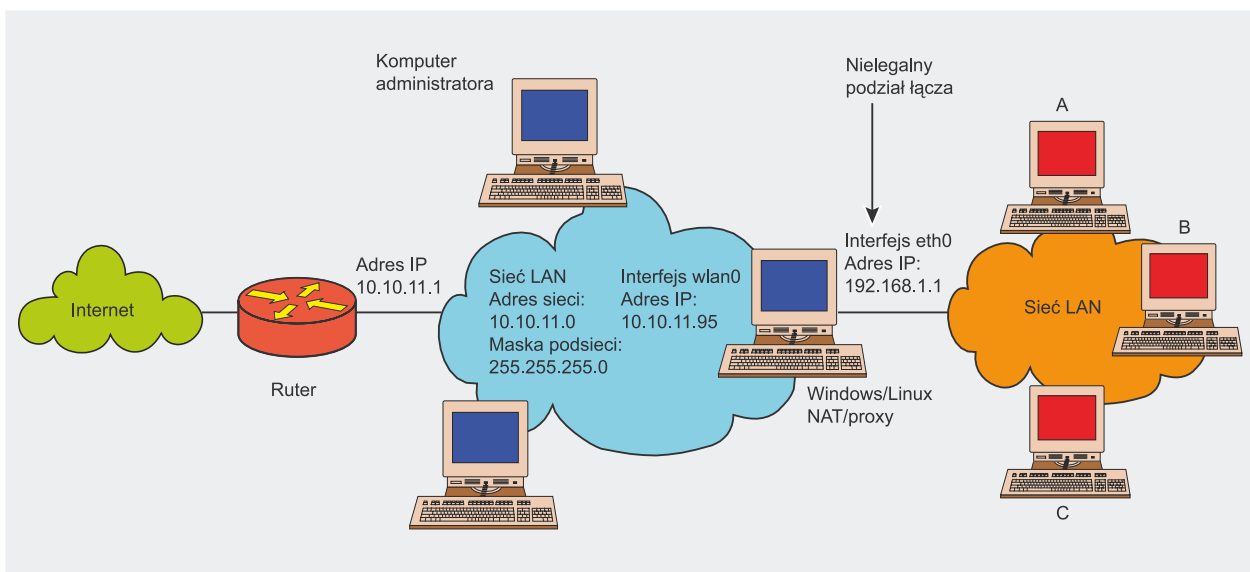
wartości początkowych TTL – Tabela 1 przedstawia początkowe wartości pola TTL dla najpopularniejszych systemów operacyjnych.

Na Rysunku 2 znajduje się schemat typowej sieci LAN z nielegalnie podzielonym łączem. Jeśli komputer udostępniający łącze pracuje jako ruter i przekazuje pakiety pomiędzy swoimi interfejsami (a w przypadku nieautoryzowanego dostępu do sieci publicznej dodatkowo z konieczności uruchomiona jest na nim usługa NAT), to w każdym pakiecie wygenerowanym przez komputer A, B lub C wartość pola TTL zostanie zmniejszona o 1. W ten sposób we właściwej sieci LAN (10.10.11.0) pojawią się pakiety, w których pole TTL będzie miało wartość o jeden mniejszą od wartości standardowej dla danego systemu.

W celu wykrycia takich pakietów administrator może na bramie internetowej uruchomić analizator pakie-

Tabela 1. Wartości TTL charakterystyczne dla poszczególnych systemów operacyjnych

Wersja systemu operacyjnego	TCP TTL	UDP TTL
AIX	60	30
FreeBSD 2.1R	64	65
HP/UX 9.0x	30	30
HP/UX 10.01	64	64
Irix 5.3	60	60
Irix 6.x	60	60
Linux	64	64
MacOs/MacTCP 2.0.x	60	60
OS/2 TCP/IP 3.0	64	64
OSF/1 V3.2A	60	30
Solaris 2.x	255	255
SunOS 4.1.3/4.1.4	60	60
MS Windows 95	32	32
MS Windows 98	128	128
MS Windows NT 3.51	32	32
MS Windows NT 4.0	128	128
MS Windows 2000	128	128
MS Windows XP	128	128



Rysunek 2. Przykładowa sieć LAN z nielegalnym podziałem łącza

tów (sniffer) i sprawdzić, czy w sieci spod jednego adresu IP (w naszym przypadku z adresu 10.10.11.95) nie dochodzą pakiety o dziwnych i różnych wartościach pola TTL. Przy założeniu, że na komputerze A uruchomiony jest system Windows 2000 (początkowy TTL 128), zaś na komputerze B system Linux (początkowy TTL 64), sniffer *tcpdump* uruchomiony na bramie internetowej może przechwytać i ujawnić przykładowe pakiety.

Pokazano to na Rysunku 3 – wiadać, że w sieci pojawiają się pakiety o źródłowym adresie IP 10.10.11.95, które mają ustawione niestandardowe wartości pola TTL (127 i 63). Druga dziwna rzecz to fakt, że jeden

komputer generuje pakiety o różnych wartościach TTL. Może to świadczyć o tym, że komputer o adresie 10.10.11.95 dzieli łącze użytkownikom używającym systemów Windows i Linux.

Domyślne wartości TTL w Windows i Linuksie

Metoda opierająca się na sprawdzaniu wartości TTL w pakietach IP może jednak okazać się nieskuteczna. W systemach Windows i Linux istnieje bowiem możliwość zmiany standardowych wartości czasu życia pakietów. Jeśli użytkownicy rozdzielono

łącza zwiększą o jeden wartość TTL w swoich systemach, to po przejściu przez komputer-bramę pakiety IP przestaną być podejrzane.

Jedyne, co może jeszcze zdradzać fakt podziału łącza, to różna wartość TTL w pakietach posiadających jeden źródłowy adres IP. Taka sytuacja nie zawsze musi jednak wystąpić – w nielegalnej sieci LAN użytkownicy mogą przecież używać tylko jednej wersji danego systemu operacyjnego, na przykład Windows 2000 czy Linuksa. Nawet jeśli sieć jest zróżnicowana i pracuje w niej wiele różnych wersji systemów operacyjnych, pajęczarze mogą ujednolicić wartości TTL na wszystkich komputerach, niezależnie od rodzaju systemu (patrz Ramka *Zmiana domyślnych wartości TTL*).

W przypadku gdy łącze udostępniane jest poprzez system Windows z włączoną funkcją ICS, ujednolicenie wartości TTL na wszystkich komputerach jest jedyną metodą ukrycia się przed administratorem. Jeśli funkcję bramki do Internetu pełni system Linux ze skonfigurowanym NAT-em, sytuacja jest dużo prostsza. Wystarczy bowiem – przy wykorzystaniu łaty dla filtra pakietów *iptables* o nazwie *patch-o-matic* – tak skonfigurować system, aby każdy wychodzący pakiet miał ustawianą jedną, ściśle określoną wartość TTL. W tym przypadku osoby rozdzielającej łącze nie interesują systemy operacyjne używane

```
[root@alpha root]# tcpdump -v -i wlan0 dst port 80
tcpdump: listening on wlan0
18:51:19.663499 10.10.11.95.1068 > 64.157.165.205.http: R [tcp sum ok] 2068904950:2068904950(0) win 0 (DF) (ttl 127, id 1172, len 40)
18:51:20.203554 10.10.11.95.1071 > flvirt.onet.pl.http: S [tcp sum ok] 2085552165:2085552165(0) win 16384 <mss 1460,nop,nop,sackOK> (DF) (ttl 127, id 1173, len 48)
18:51:20.235048 10.10.11.95.1071 > flvirt.onet.pl.http: . [tcp sum ok] ack 228389216 win 17520 (DF) (ttl 127, id 1174, len 40)
18:51:24.670602 10.10.11.95.1069 > 66.35.229.174.http: R [tcp sum ok] 2070595543:2070595543(0) win 0 (DF) (ttl 127, id 1176, len 40)
18:51:34.685193 10.10.11.95.1071 > flvirt.onet.pl.http: . [tcp sum ok] ack 2 win 17520 (DF) (ttl 127, id 1178, len 40)
18:51:34.685861 10.10.11.95.1071 > flvirt.onet.pl.http: F [tcp sum ok] 0:0(0) ack 2 win 17520 (DF) (ttl 127, id 1179, len 40)
18:52:34.437694 10.10.11.95.1142 > flvirt.onet.pl.http: S [tcp sum ok] 2131150548:2131150548(0) win 5840 <mss 1460,sackOK,timestamp 1883223 0,nop,wscale 0> (DF) [tos 0x10] (ttl 63, id 31363, len 60)
18:52:34.583055 10.10.11.95.1142 > flvirt.onet.pl.http: . [tcp sum ok] ack 1448286057 win 5840 <nop,nop,timestamp 1883240 1435745072> (DF) [tos 0x10] (ttl 63, id 31364, len 52)
```

Rysunek 3. Wartości TTL po przejściu przez nielegalny ruter

Zmiana domyślnych wartości TTL

Linux

Zmiana wartości TTL dla lokalnej maszyny w systemie Linux sprowadza się do wykonania w konsoli następującego polecenia:

```
# echo "X" > /proc/sys/net/ipv4/ip_default_ttl
```

gdzie X to nowe, zmienione TTL. Standardowo ma ono wartość 64 – jeśli Linux ma udawać system Windows, wystarczy jako X podać liczbę 128 (a najlepiej 129, jeśli korzystamy z dzielonego łącza i nie chcemy wzbudzać podejrzeń administratora sieci).

Windows 2000/XP

Domyślnie w pakietach wysyłanych z systemu Windows 2000/XP wartość TTL jest ustawiana na 128. Najszybszym sposobem sprawdzenia standardowej wartości TTL w systemie jest wykorzystanie polecenia *ping*. Wystarczy wysłać pakiety ICMP *echo request* na interfejs pętli zwrotnej (ang. *loopback*) i zobaczyć, jaka wartość TTL ustawiona jest w odpowiedziach *ICMP echo reply*:

```
ping 127.0.0.1
```

Zmiany TTL dokonuje się w rejestrze systemu. Za przechowywanie tej wartości odpowiada wpis *DefaultTTL* w kluczu *HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Tcpip\Parameters*. Jeśli nie ma takiej wartości – należy ją utworzyć, wykorzystując typ *DWORD*.

Windows 95/98/Me

W systemach Windows 95/98/Me wartość TTL przechowywana jest w kluczu *HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VxD\MSTCP\DefaultTTL*. Jeśli podanej wartości *DefaultTTL* nie ma domyślnie w powyższym kluczu, należy ją utworzyć wykorzystując typ *STRING*.

w nielegalnej pajęczynie, ponieważ wszystkie pakiety po przejściu przez NAT będą miały ustawione tę samą wartość w polu TTL nagłówka IP.

Jednakowa wartość TTL pakietów wychodzących

Jeśli komputer-brama pracuje w oparciu o system Linux ze skonfigurowaną usługą NAT, do ustawienia jednakowego TTL nielegalnych pakietów można wykorzystać łąkę do *iptables* autorstwa Haralda Welte, która dodaje nowy cel w regułach filtrowania. Cel ten umożliwi użytkownikowi ustawianie wartości TTL dla pakietu IP oraz zwiększanie lub zmniejszanie jej o określoną wartość. Łąka dostępna jest pod adresem <http://netfilter.org>.

Do nałożenia łąki potrzebne są źródła kernela i *iptables*. Po pomyślnym spatchowaniu źródeł należy skompilować i zainstalować nowe jądro oraz nowe *iptables*. Podczas konfiguracji jądra można ustawić nowe opcje dostępne w sekcji *Networking Options* -> *Netfilter Configura-*

tion. Dla celu TTL dostępne są następujące opcje:

- `--ttl-set wartość` – ustawia wartość TTL na wartość,
- `--ttl-dec wartość` – zmniejsza wartość TTL o wartość,
- `--ttl-inc wartość` – zwiększa wartość TTL o wartość.

Ustawienie TTL we wszystkich przekazywanych przez komputer-bramkę pakietach na wartość 128 sprowadza się do dopisania do tablicy *mangle iptables* następującej reguły filtrowania:

```
# iptables -t mangle \  
-A FORWARD -j TTL \  
--ttl-set 128
```

Po wykonaniu powyższego polecenia zawartość tablicy powinna być taka, jak na Listingu 1.

Innym sposobem jest ustawienie odpowiedniej wartości TTL jeszcze przed wykonaniem procesu routowania na komputerze-bramce, na przykład:

```
# iptables -t mangle \  
-A PREROUTING -i eth0 \  
-j TTL --ttl-set 129
```

Więcej niż zero

Administrator może wykorzystać wartość TTL do utrudnienia nieuczciwym osobom podziału łącza. Jeśli na maszynie, która jest bezpośrednio podłączona do łącza internetowego pracuje Linux, administrator może ustawić w pakietach, które są skierowane do sieci lokalnej wartość TTL na 1. Wtedy każdy nielegalny ruter w sieci LAN po odebraniu takiego pakietu i zmniejszeniu wartości TTL o jeden, będzie zmuszony usunąć taki pakiet z sieci, tak więc informacja nie zostanie przekazana dalej i nielegalna sieć przestanie funkcjonować (jeśli zaś pakiet trafi do legalnej stacji końcowej, to z TTL o wartości 1 będzie odebrany bez żadnych problemów). Należy zauważyć, że takie rozwiązanie jest skuteczne, jeśli komputer udostępniający nielegalnie łącze pracuje jako ruter i wykorzystuje translację adresów sieciowych (NAT).

Opisany powyżej sposób zmniejszania wartości TTL administrator nielegalnej pajęczyny może łatwo zneutralizować, zwiększając wartość TTL w każdym docierającym do routera pakiecie jeszcze przed procesem routowania. W systemie Linux wystarczy wykorzystać opisywany już wyżej nowy cel *iptables* (o nazwie *TTL*) i wpisać do tablicy *iptables* następującą regułę:

Listing 1. Zawartość tablicy *mangle* po wprowadzeniu reguły filtrowania

```
# iptables -t mangle --list  
Chain FORWARD (policy ACCEPT)  
target prot opt source destination  
TTL all -- anywhere anywhere TTL set to 128
```



```
# iptables -t mangle \  
-A PREROUTING -i wlan0 \  
-j TTL --ttl-set 2
```

Dzięki temu w każdym docierającym do interfejsu `wlan0` (patrz Rysunek 2) pakiecie IP – który może mieć ustawioną wartość TTL nawet na 1 – wartość pola TTL zostanie ustawiona na 2. Tak zmodyfikowany pakiet zostanie poddany procesowi rutowania, jego wartość TTL zostanie zmniejszona o 1 i pakiet bez problemów trafi do użytkownika końcowego w nielegalnej sieci LAN. Naturalnie, jeśli ten zdecydował się na dalszy podział łącza, to TTL w pakietach wychodzących z routera powinien być ustawiony na większą wartość.

Proxy po raz pierwszy

Metody bazujące na manipulacjach wartościami TTL mają rację bytu tak długo, jak długo mamy do czynienia z urządzeniami trzeciej, sieciowej warstwy modelu ISO/OSI. Wystarczy jednak, że administrator pajęczyny zdecyduje się na użycie do podziału łącza urządzenia warstwy czwartej lub wyższych (*gatewaya*, czyli w naszym przypadku różnych rodzajów pośredników sieciowych proxy), które tworzą cały pakiet IP od nowa, a opisane powyżej metody okażą się nieskuteczne.

W skrajnym przypadku można sobie wyobrazić, że w pajęczynie funkcjonuje wyłącznie czysty protokół IPX, zaś na wyjściu z niej – bramka IPX/IP, która nawiązuje połączenia w imieniu klientów, a przychodzące odpowiedzi przekazuje do sieci wewnętrznej (pajęczyny) wewnątrz pakietów IPX. Dopiero na stacjach końcowych są one wyciągane przez odpowiedni socket, który przekazuje je aplikacjom sieciowym już w postaci zrozumiałej dla protokołów stosu TCP/IP. Z punktu widzenia transmisji IP ostatnim miejscem, do którego dociera datagram IP jest brama, a więc komputer przyłączony bezpośrednio do sieci zewnętrznej.

Głuchy telefon

Kolejnym sposobem wykrycia nielegalnie dzielonego łącza jest spraw-

Serwery proxy

Serwery proxy pełnią rolę pośredników między Internetem a systemami w sieci LAN, które nie mają do niego dostępu. Ich stosowanie daje sporo korzyści – oszczędza przestrzeń adresową, umożliwia inteligentne filtrowanie i uwierzytlanianie na poziomie użytkownika, czy wręcz zwiększa bezpieczeństwo (serwer proxy staje się jedyną maszyną z bezpośrednim dostępem do Internetu).

Użytkownicy korzystający z Internetu za pośrednictwem proxy mają złudzenie połączenia z siecią zewnętrzną, w rzeczywistości jednak łączą się tylko z jedną maszyną. Po wysłaniu żądania przez klienta proxy sprawdza, czy jest ono możliwe do zrealizowania. Jeśli tak – porozumiewa się w imieniu klienta ze zdalnym serwerem, a następnie pośredniczy w komunikacji.

Najogólniej mówiąc, serwery tego typu dzielą się na dwa rodzaje: działające na poziomie aplikacji i obwodowe. Proxy aplikacyjne to takie, których zadaniem jest pośredniczenie w komunikacji między jedną (lub więcej) aplikacją a siecią zewnętrzną. Natomiast proxy obwodowe nie zajmują się konkretnym typem zleceń – otrzymują i przekazują dane bez rozróżniania konkretnych protokołów sieciowych.

Jest jeszcze drugi podział serwerów proxy: na uniwersalne (stosujące wiele protokołów) i wyspecjalizowane (zajmujące się wyłącznie jednym rodzajem ruchu sieciowego). W praktyce jednak serwery wyspecjalizowane to serwery aplikacyjne (np. pośredniczące jedynie w ruchu HTTP), a serwery uniwersalne – obwodowe.

Istnieje też specjalny typ proxy, umożliwiający buforowanie ruchu sieciowego (*cache*) – może on w znacznym stopniu zwiększyć wydajność sieci w przypadku słabego łącza. W dodatku umożliwia szczegółową rejestrację zdarzeń (logowanie) i zaawansowaną kontrolę dostępu. Takie proxy nazywane są inteligentnymi.

Najpopularniejsze obwodowe serwery proxy dla Windows to *WinProxy* (<http://www.winproxy.com/>), *WinGate* (<http://www.wingate.com/>) i *WinRoute* (<http://www.kerio.com>). Użytkownicy Linuksa mogą skorzystać między innymi z *Proxy* (<http://proxy.sourceforge.net/>), *Zaval Proxy Suite* (<http://www.zaval.org/products/proxy/>) czy *SuSE Proxy Suite* (<http://proxy-suite.suse.de>).

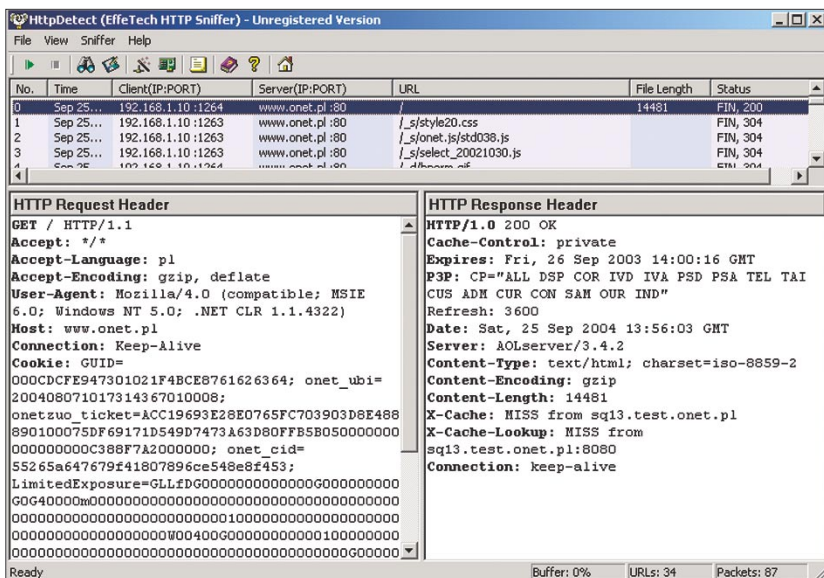
dzenie, czy podejrzany komputer ma włączone przekazywanie pakietów (*IP forwarding*). Jeśli tak – możemy się domyślać, że mamy do czynienia z nieuczciwym użytkownikiem. Warto jednak zauważyć, że nie jest to żaden dowód. Każdy użytkownik w sieci lokalnej może bowiem mieć w swoim komputerze dwie skonfigurowane karty sieciowe, pomiędzy którymi ustawił przekazywanie pakietów. Niemniej może to być dobra podstawa do przyjrzenia się takiemu użytkownikowi nieco bliżej.

Rozważmy sytuację z Rysunku 2, w której administrator dysponuje komputerem z systemem Linux. Jedyne, co należy zrobić, to dodać do własnej tablicy routingu fałszywy wpis mówiący o tym, że pakiet IP wysłany do określonej sieci ma zostać przekazany pod wskazany, podejrzany naszym zdaniem adres IP:

```
# route add -net 20.20.20.0/24 \  
gw 10.10.11.95 eth0
```

W tym momencie pakiet wysłany na przykład na adres 20.20.20.20 zostanie dostarczony do komputera o adresie 10.10.11.95 (patrz Rysunek 2). Jeśli komputer ten ma włączone przekazywanie pakietów, odbierze spreparowany pakiet i przekaże do procesu wyboru dalszej trasy dla niego. Ponieważ mało prawdopodobne jest, aby w tablicy rutowania znalazł się wpis odnoszący się akurat do sieci 20.20.20.0/24, system podejmie decyzję o przekazaniu go na swoją domyślną bramę. Tak się jednak składa, że domyślną bramą dla takiego komputera jest router bezpośrednio przyłączony do Internetu (w naszym przypadku jest to router o adresie 10.10.11.1). W sieci pojawią się dwa pakiety *ICMP echo request*: jeden wysłany z komputera administratora do podejrzanego komputera i drugi wysłany przez nielegalny router.

Cały eksperyment sprowadza się do uruchomienia na komputerze administratora (a jeszcze lepiej na



Rysunek 4. Pole User-Agent w nagłówku HTTP

bramce do Internetu) na jednej konsoli sniffera *tcpdump*:

```
# tcpdump -n -i eth0
```

i z drugiej konsoli wydania polecenia *ping*:

```
# ping 20.20.20.20
```

Jeśli komputer o danym adresie IP pracuje jako ruter powinniśmy zobaczyć dwa pakiety *ICMP echo request*:

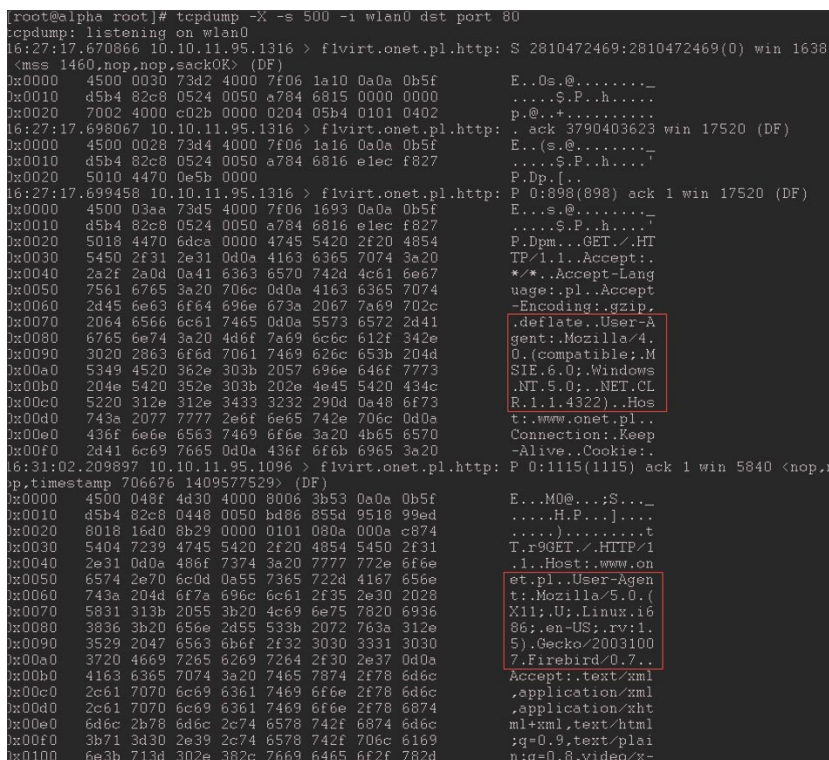
```
00:59:47:270862 10.10.11.2 <-
> 20.20.20.20: icmp: echo request
00:59:47:271276 10.10.11.2 <-
> 20.20.20.20: icmp: echo request
```

Można również próbować sprawdzić, jaka podsieć wykorzystywana jest w nielegalnej sieci LAN. Wymaga to jednak napisania odpowiedniego skryptu, gdyż sprawdzanie ręczne jest raczej skazane na niepowodzenie. W tym celu należy wykorzystać mechanizm opisany wyżej, dobierając jednak bardziej wiarygodny adres podsieci, na przykład:

```
# route add -net 192.168.1.0/24 \
gw 10.10.11.95 eth0
```

Jeśli nie trafiliśmy na właściwą podsieć, efekt będzie taki sam jak poprzednio. Jeżeli udało nam się zgod-

nąć podsieć, pakiet zostanie przekazany pod wskazany adres. Jeśli komputer o podanym adresie jest dostępny w nielegalnej podsieci, udzieli on odpowiedzi w postaci pakietu *ICMP echo reply*. W przeciwnym przypadku otrzymamy komunikat o błędzie informujący, że dany host jest nieosiągalny (*icmp host unreachable*). Mechanizm będzie działał, dopóki administrator pajęczyny nie uruchomi



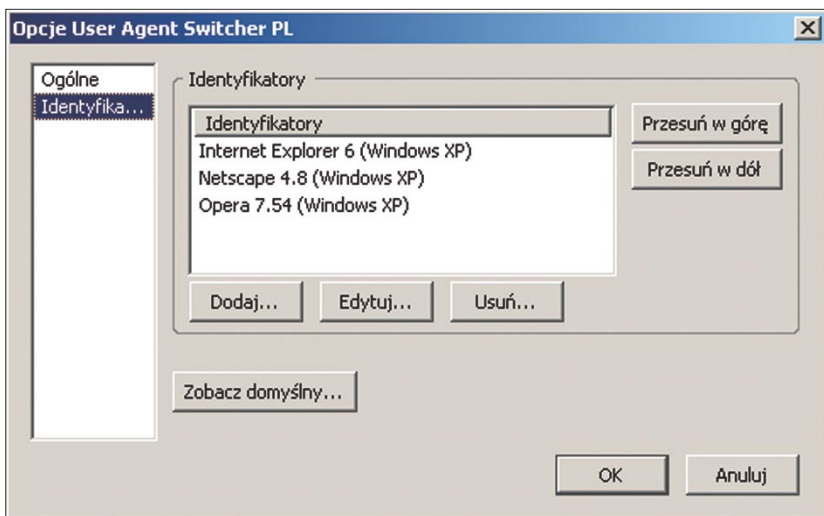
Rysunek 5. Podejrzanе pakiety HTTP

mi na nielegalnym routerze filtra pakietowego typu *statefull (dynamic)* i odfiltrowania połączeń nawiązywanych z pajęczyną z zewnątrz.

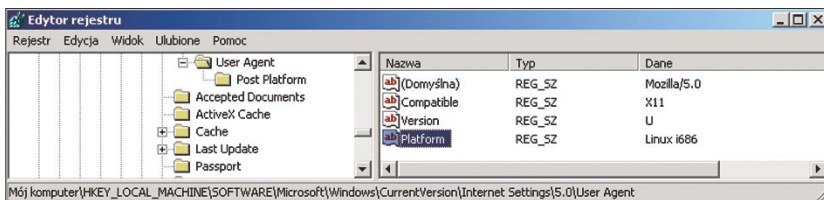
Identyfikacja przeglądarek internetowych

Każda uruchomiona domyślnie w systemie przeglądarka sieciowa wysyła do serwera WWW, w swoim żądaniu pobrania strony nagłówek HTTP. W nagłówku takim znajduje się pole *User-Agent*, w którym zawarta jest informacja o typie przeglądarki oraz rodzaju i wersji systemu operacyjnego, na którym ta przeglądarka jest uruchomiona (Rysunek 4). Można wykrywać ten fakt do wykrywania nielegalnego podziału łącza, szczególnie gdy nielegalni użytkownicy mają różne typy i wersje przeglądarek, w dodatku działające pod kontrolą różnych systemów operacyjnych.

Test wykrywający dzielone łącze polega na analizie przechwyconych w sieci pakietów. Wśród nich należy szukać takich, które wysłane zostały z jednego adresu źródłowego (nielegalnej bramki). Jeśli w takich pa-



Rysunek 6. Zmiana identyfikacji przeglądarki Mozilla



Rysunek 7. Zmiana identyfikacji przeglądarki Internet Explorer

User Agent Page	
name	value
IP	[REDACTED]
User Agent	Mozilla/5.0 (X11; U; Linux i686; .NET CLR 1.1.4322)
Resolved as	System: (Unix, Linux), Browser: (Netscape, 6.0.0)

Rysunek 8. Identyfikacja przeglądarki IE po dokonanych zmianach

kietach pole `User-Agent` zawiera informacje o różnych przeglądarkach i systemach operacyjnych, jest to sytuacja podejrzana.

Bardziej podejrzaną sytuacją jest taka, w której w polu `User-Agent` widać różne wersje systemów operacyjnych, nie zaś samych przeglądarek. Używanie w tym samym czasie dwóch systemów operacyjnych korzystających z jednego adresu IP jest raczej niemożliwe (jeśli wykluczmy sytuację wykorzystania programów umożliwiających uruchamianie wirtualnych maszyn, takich jak *VMware* czy *Microsoft Virtual PC*), natomiast używanie kilku przeglądarek w jednym systemie operacyjnym – jak najbardziej. Rysunek 5 przedstawia pakiety, które powinny zwrócić uwagę administratora sieci.

Na Rysunku 5 zauważyć można dwa żądania pobrania strony `http://www.onet.pl/`, wysłane spod jednego adresu 10.10.11.95, w których widać,

że wykorzystane zostały dwie przeglądarki (MSIE 6.0 i *Mozilla Firebird*) uruchomione na dwóch systemach operacyjnych (Windows 2000 identyfikowany jako Windows NT 5.0 oraz Linux). Pozostaje tylko pytanie, co robić, jeśli użytkownik ma na jednym komputerze zainstalowane kilka systemów operacyjnych i pracuje raz w jednym, raz w innym.

Proxy po raz drugi

Opisana metoda wydaje się być dobra, choć i ją można obejść, jeśli po-

le `User-Agent` zostanie usunięte lub zmodyfikowane tak, aby wskazywało na zupełnie inny rodzaj przeglądarki i system operacyjny. Można to zrobić dla każdej przeglądarki w nielegalnej sieci LAN, ustawiając identyfikację taką samą we wszystkich przeglądarkach lub zastosować serwer proxy WWW na nielegalnej bramce i zmusić użytkowników do korzystania z niego. Odpowiednia konfiguracja serwera proxy spowoduje, że niezależnie od używanych przez użytkowników przeglądarek, właściwe zapytanie generowane przez serwer proxy będzie zawierało zawsze tę samą informację w polu `User-Agent`.

Zmiana wartości pola User-Agent

W przypadku przeglądarek Mozilla (dla Windows) dostępne jest rozszerzenie *User Agent Switcher PL*, które dodaje do programu menu umożliwiające zmianę identyfikacji przeglądarki. Rozszerzenie to zapewnia podobną funkcjonalność jak Identyfikacja przeglądarki dostępna w Operze. Umożliwia konfigurowanie listy agentów wyświetlanych w menu i ich wybór w zależności od potrzeb (Rysunek 6).

W przypadku przeglądarek Internet Explorer należy zmodyfikować gałąź rejestru systemowego `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\5.0`. Należy stworzyć w niej klucz `User Agent` (jeśli go tam nie ma). Wpisanie wartości domyślnej zastąpi ciąg `Mozilla/4.0`. Pozostałe parametry można modyfikować dodając do klucza `User Agent` nowe wartości ciągu o nazwach `Compatible`, `Version` czy `Platform` z własnymi wartościami. Dodatkowo można dodawać nowe wartości w kluczu `Post Platform`, jako dodatkowe informacje dla pola

```
# Example:
#
# Suppose you are running Privoxy on a machine which has the
# address 192.168.0.1 on your local private network (192.168.0.0)
# and has another outside connection with a different address. You
# want it to serve requests from inside only:
#
# listen-address 192.168.0.1:8118
#
listen-address 192.168.1.1:8555
```

Rysunek 9. Określenie adresu IP i portu, na których dostępny będzie serwer proxy

```

[root@alpha p0f]# ./p0f -p -f p0f.fp -i wlan0
p0f - passive os fingerprinting utility, version 2.0.5
(C) M. Zalewski <lcantuf@dione.cc>, W. Stearns <wstearns@pobox.com>
p0f: listening (SYN) on 'wlan0', 231 sigs (13 generic), rule: 'all'.
10.10.11.95:1762 - Windows 2000 SP2+, XP SP1 (seldom 98 4.10.2222)
  -> 213.180.130.200:80 (distance 1, link: ethernet/modem)
10.10.11.95:1764 - Windows 2000 SP2+, XP SP1 (seldom 98 4.10.2222)
  -> 213.180.130.110:80 (distance 1, link: ethernet/modem)
10.10.11.95:1765 - Windows 2000 SP2+, XP SP1 (seldom 98 4.10.2222)
  -> 213.180.131.42:80 (distance 1, link: ethernet/modem)
10.10.11.95:1764 - Windows 2000 SP2+, XP SP1 (seldom 98 4.10.2222)
  -> 213.180.130.110:80 (distance 1, link: ethernet/modem)
10.10.11.95:1765 - Windows 2000 SP2+, XP SP1 (seldom 98 4.10.2222)
  -> 213.180.131.42:80 (distance 1, link: ethernet/modem)
10.10.11.95:1122 - Linux 2.4/2.6 <= 2.6.7 (up: 14 hrs)
  -> 213.180.130.200:80 (distance 0, link: ethernet/modem)
10.10.11.95:1123 - Linux 2.4/2.6 <= 2.6.7 (up: 14 hrs)
  -> 213.180.131.43:80 (distance 0, link: ethernet/modem)
10.10.11.95:1124 - Linux 2.4/2.6 <= 2.6.7 (up: 14 hrs)
  -> 213.180.130.110:80 (distance 0, link: ethernet/modem)
10.10.11.95:1125 - Linux 2.4/2.6 <= 2.6.7 (up: 14 hrs)
  -> 213.180.130.200:80 (distance 0, link: ethernet/modem)
10.10.11.95:1125 - Linux 2.4/2.6 <= 2.6.7 (up: 14 hrs)
  -> 213.180.130.200:80 (distance 0, link: ethernet/modem)
10.10.11.95:1125 - Linux 2.4/2.6 <= 2.6.7 (up: 14 hrs)
  -> 213.180.130.200:80 (distance 0, link: ethernet/modem)
10.10.11.95:1126 - Linux 2.4/2.6 <= 2.6.7 (up: 14 hrs)
  -> 213.180.130.200:80 (distance 0, link: ethernet/modem)

```

Rysunek 10. Wyniki pasywnego wykrywania

User-Agent. Należy dopisywać je jako nazwy ciągów bez wartości np. *dodatkowa informacja = ""*. Przykładowe zmiany w rejestrze pokazane są na Rysunku 7.

Wchodząc na stronę <http://hitgate.gemius.pl:9170/ua.html> możemy sprawdzić, w jaki sposób przedstawia się nasza przeglądarka. Można również wykorzystać ten URL do sprawdzenia pola *User-Agent* po dokonaniu zmian w rejestrze. Dla przykładu zmiana pierwszych czterech wartości pola *User-Agent* spowoduje, że przeglądarka zostanie rozpoznana jako Netscape 6.0 uruchomiona w systemie Linux (Rysunek 8).

Wykorzystanie serwera proxy do ujednoczenia pola User-Agent

Prostszym sposobem ukrycia informacji o przeglądarkach jest skorzystanie z serwera proxy WWW dla systemu Linux, takiego jak na przykład *privoxy*. Należy go zainstalować na nielegalnej bramce i poinstruować użytkowników, żeby skonfigurowali swoje przeglądarki do korzystania z serwera proxy. Program można pobrać ze strony <http://www.privoxy.org/>.

Po zainstalowaniu programu trzeba dokonać dwóch zmian w plikach *config* i *default.action*. W pierwszym pliku należy ustawić na jakim interfejsie serwer ma nasłuchiwać połączeń od użytkowników. Musimy też określić

adres IP i port przypisany do interfejsu wewnętrznego, czyli tego od strony nielegalnej sieci LAN (Rysunek 9).

Natomiast w pliku *default.action* należy określić zawartość pola *User-Agent* dla wszystkich wychodzących połączeń WWW. W tym celu należy zamienić linię:

```

-hide-user-agent \
na na przykład taką:
+hide-user-agent{Mozilla/4.0 ←
(compatible; MSIE 6.0; ←
Windows NT 5.0; ←\
.NET CLR 1.1.4322)} \

```

Pasywne wykrywanie systemu operacyjnego

Jeszcze innym sposobem detekcji nielegalnych rozgałęzień sieci jest wykrywanie wielu różnych wersji systemów operacyjnych korzystających w tym samym czasie z jednego adresu IP. Pasywna identyfikacja jest metodą, w której nie wysyła się do maszyny docelowej żadnych pakietów testujących (patrz Artykuł Michała Wojciechowskiego OS *fingerprinting* – *jak nie dać się rozpoznać*, *hakin9* 4/2004).

Podstawą działania tej metody jest analiza stosu TCP/IP komputera na podstawie pakietów jakie ten komputer generuje, po przechwyconiu ich metodą sniffingu. Samo pojęcie analizowania stosu oznacza określanie rodzaju i wersji systemu ope-

racyjnego na podstawie różnic w implementacjach stosów TCP/IP, używanych przez różnych producentów tychże systemów. Pomimo ścisłych zasad budowy stosów TCP/IP określonych w dokumentach RFC, w konkretnych implementacjach poszczególnych systemów operacyjnych można odnaleźć pewne różnice. Dotyczy to głównie charakterystycznych wartości pól ustawianych w nagłówkach protokołów IP i TCP. Programy do pasywnej analizy stosów TCP/IP poddają analizie między innymi następujące pola w nagłówku IP:

- czas życia pakietu (TTL),
- pole ID (identyfikacja),
- ustawienia bitów TOS (ang. *Type Of Service*),
- ustawienia bitu *nie fragmentuj* (ang. *don't fragment*).

Z kolei w nagłówku TCP sprawdzane są następujące pola:

- rozmiar okna (ang. *Window Size*),
- maksymalny rozmiar segmentu (ang. *Maximum Segment Size*),
- opcja selektywnego odrzucania (ang. *Selective Acknowledgment*),
- opcja NOP (ang. *No Operation*).

Jednym z narzędzi przeznaczonych do pasywnego fingerprintingu jest program *p0f*. Można go pobrać ze strony <http://lcantuf.coredump.cx/p0f.shtml>. W systemie Windows program wymaga zainstalowanej biblioteki *Winpcap*.

Program może zidentyfikować system operacyjny działający na hostach na podstawie pakietów IP, które mają ustawione następujące flagi TCP:

- SYN,
- SYN i ACK,
- RST.

Za pomocą opcji `--f` określa się plik przechowujący gotowe sygnatury dla poszczególnych systemów operacyjnych, które *p0f* porównuje z tym co zidentyfikuje w przechwyconym pa-



kiecie. Dla każdej z metod istnieje oddzielny plik:

- *pOf.fp*,
- *pOfa.fp*,
- *pOfr.fp*.

Przykładowo, sygnatura dla systemu Windows 2000 z service packiem 4 lub XP z service packiem 1 wygląda następująco: 65535:128:1:48:M*,N,N,S:.:Windows:2000 SP4, XP SP1. Kolejne pola w powyższym zapisie oznaczają:

- 65535 – rozmiar okna TCP,
- 128 – czas życia pakietu (TTL),
- 1 – ustawiony bit *nie fragmentować pakietu*,
- 48 – rozmiar pakietu,
- M – maksymalny rozmiar segmentu (MSS),
- N – opcja *nie używane* (NOP),
- N – opcja *nie używane* (NOP),
- S – włączone selektywne potwierdzanie ACK.

Opcja `-p` służy do przestawienia interfejsu sieciowego w tryb odbioru wszystkich pakietów (ang. *promiscuous*), nie tylko tych adresowanych do komputera, na którym działa *pOf*. Za pomocą opcji `-i` możemy określić interfejs, na którym program ma nasłuchiwać. Na Rysunku 10 widać, że program *pOf* zidentyfikował w tym samym czasie dwa systemy operacyjne korzystające z jednego źródłowego adresu IP. Takie wyniki mogą świadczyć o tym, że w naszej sieci ktoś udostępnia łącze innym użytkownikom. W najnowszej wersji *pOfa* autor wprowadził dodatkową opcję `-m`, która określa (na podstawie anomalii w pakietach) w procentach prawdopodobieństwo istnienia maskarady pod danym adresem IP.

Oczywiście (proxy po raz trzeci...) wszystko to ma sens, jeżeli administrator pajęczyny nie zainstaluje jako urządzenia dzielącego proxy obwodowego. W takim bowiem razie fingerprinting dotyczył będzie systemu operacyjnego pośrednika.

Szyfrowa praca

Istnieje wiele innych metod wykrywania nielegalnego współdzielenia łącza

Inne metody wykrywania pajęczarzy

Komunikatory internetowe

Analizując pakiety wychodzące z komunikatorów internetowych możemy w nich dostrzec identyfikator (zazwyczaj numer) użytkownika (patrz Artykuł Konstantina Klyagina *Paranoja w świecie komunikatorów w hakin9 3/2004*). Ponieważ istnieje niewielka szansa, że użytkownik ma na jednym komputerze w tym samym czasie uruchomione kilka kont w komunikatorze internetowym, wychwycenie pakietów zawierających różne identyfikatory użytkownika i pochodzących z jednego IP z dużym prawdopodobieństwem świadczy o tym, że mamy do czynienia z nielegalną siecią.

Śledzenie poczty

Ponieważ zdecydowana większość użytkowników nie korzysta z szyfrowanych połączeń z serwerami pocztowymi, analizując wysyłane listy za pomocą sniffera możemy na podstawie niektórych nagłówków z dużym prawdopodobieństwem określić, że mamy do czynienia z pajęczarzem. Rzadko kiedy użytkownik używa dwóch programów pocztowych na raz, a większość programów pocztowych przedstawia się w nagłówkach *User-Agent* lub *X-Mailer*.

Sprawdzanie uptime

Pakiety TCP mogą zawierać dodatkową (opcjonalną) informację – *timestamp*, czyli znacznik czasowy. Różne systemy operacyjne zwiększają ten znacznik w różnych odstępach czasu. Znacznik ten (jeśli wiemy, z jakim systemem operacyjnym mamy do czynienia), po pomnożeniu przez częstotliwość zwiększania licznika, wskazuje *uptime* maszyny, czyli czas od jej ostatniego uruchomienia.

Jeśli, na przykład korzystając z *tcpdump*, wykryjemy z jednego IP pakiety o skrajnie różnych wartościach *timestamp*, możemy być prawie pewni, że mamy do czynienia z różnymi maszynami, czyli z pajęczarzami:

```
# tcpdump -n | grep timestamp
```

A oto fragment przykładowego wyniku:

```
<nop,nop,timestamp 3320208223 97006325>
```

Dwie wartości zawarte po słowie *timestamp* to odpowiednio *timestamp* hosta źródłowego i ostatnia wartość *timestampu* otrzymana od hosta docelowego. Metoda ta ma ograniczoną przydatność, ponieważ korzystając z niej zakładamy, że komputery w pajęczarskiej sieci będą wysyłały pakiety z opcją *timestamp*, co nie zawsze może mieć miejsce.

i szereg sposobów utrudniania życia administratorom pajęczyn (patrz Ramka *Inne metody wykrywania pajęczarzy*). Wszystkie mają jednak to do siebie, że dadzą się – przy odrobini inwencji – obejść i unieszkodliwić.

Wydaje się zatem, że byłoby najlepiej, aby dostawcy Internetu pozostali przy zarządzaniu pasmem i limitami transferu, a zabawę w Wielkiego Brata pozostawili mało ambitnym rozrywkowym programom TV. ■

W Sieci

- <http://support.microsoft.com/default.aspx?scid=kb;en-us;158474> – informacje o połączeniu najważniejszych parametrów sieciowych w rejestrze Windows,
- <http://lukasz.bromirski.net/docs/translations/netfilter-extensions.html> – opis łata pakietu *iptables*,
- <http://winpcap.polito.it/install/default.htm> – biblioteka *Winpcap*,
- <http://lcamtuf.coredump.cx/pOf.shtml> – strona domowa programu *pOf*,
- <http://netfilter.org> – projekt *Netfilter*.
- <http://www.0xdecabdad.com/TCP-Timestamping-Obtaining-System-Uptime-Remotely.html> – informacje na temat zdalnego uzyskania *uptime* systemu.