Netcat - kot na tropie pakietów

Kacper Pawłowski

19 grudnia 2011

W tym tekście zamierzam przedstawić możliwości znanego linuksowego programu netcat. Czytelnik przekona się, że ta mała tekstowa aplikacja, o jakże niepozornie brzmiącej nazwie, jest dość ciekawym narzędziem sieciowym. Poznamy możliwości netcata poprzz zabawę różnymi usługami sieciowymi. W przyszłości pomoże nam to troszkę poznać działanie różnych protokołów sieciowych, np. protokółu sieci IRC, innej sieci komunikacyjnej, czy serwera WWW. Netcat przysłuży nam się także kiedy będziemy używać gniazd sieciowych (socketów) w pisanych przez nas programach. "Internetowego kota" można "zatrudnić" także w windowsowym środowisku. Warto zapoznać się ze stroną projektu.

Prosty serwer - nasłuchujemy połączeń

Jako, że program umożliwia nam obsługę modelu klient-serwer, za jego pomocą otworzymy dowolny wolny port z 16-bitowego zakresu i będziemy na nim nasłuchiwać. Szybko przekonamy się, że mamy również możliwość odpowiedzi na pakiety. Te możliwości aplikacji możemy zastosować do częściowego poznania protokołów wykorzystywanych przez różne programy internetowe. Warto pamiętać, że w systemie Linux tylko root ma prawa do otwierania wszystkich portów. Zwykły użytkownik z reguły może otworzyć porty z zakresu 1024-65535. Aby rozpocząc nasłuchiwanie wywołujemy polecenie (zamiast podania numeru portu i użycia argumentu '-p' możemy podać argument '-r', wtedy będziemy mieli do czynienia z losowym portem z reguły o wysokim numerze):

```
$ nc -lvp [port]
```

Kiedy otworzyliśmy port i na nim nasłuchujemy warto nawiązać z nim połączenie. W tym celu uruchamiamy kolejne okno terminala na lokalnym lub zdalnym komputerze i wykorzystujemy program "telnet". Aby się połączyć z naszą usługą wywołujemy go poleceniem:

```
$ telnet [host] [port]
```

W miejscu */host/* może się również znaleźć adres IP. W przypadku gdy korzystamy z komputera lokalnego możemy wpisać adres IP "127.0.0.1", albo adres hosta łocalhost". Z kolei w miejscu */port/* podajemy numer portu serwera, który otworzyliśmy w netcacie.

Domyślnie serwer czeka tylko na pierwsze połączenie. Kiedy je uzyska wysyła temu, z którym się połączył to co napisaliśmy w nc przed tym połączeniem po tym jak już uruchomiliśmy nasłuchiwanie. Skąd śledząc proces nc możemy wiedzieć kiedy ktoś się z nami połączył? W takiej chwili w oknie pojawi się nowa linia wyglądają mniej więcej tak:

```
connect to [127.0.0.1] from localhost [127.0.0.1] 46335
```

Jak działa przeglądarka - udajemy serwer WWW

W tym celu włączmy nasłuchiwanie wybranego portu. Następnie uruchamiamy dowolną przeglądarkę i wpisujemy adres serwera, na którym nasłuchujemy w tej formie *"[host]:[port]"*. Teraz przełączmy się na okno w którym mamy włączonego netcata. Pojawi się nam coś podobnego:

```
connect to [127.0.0.1] from localhost [127.0.0.1] 34934
GET / HTTP/1.1
Host: localhost:2000
Connection: keep-alive
User-Agent: Mozilla/5.0 (X11; Linux i686) AppleWebKit/535.2 (KHTML,
like Gecko) Chrome/15.0.874.121 Safari/535.2
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;
q=0.8
Accept-Encoding: gzip,deflate,sdch
Accept-Language: pl-PL,pl;q=0.8,en-US;q=0.6,en;q=0.4
Accept-Charset: ISO-8859-2,utf-8;q=0.7,*;q=0.3
```

Jest to nic innego tylko zapytanie HTTP, które otrzymaliśmy od przeglądarki. Warto, więc odpowiedzieć przeglądarce. Napiszmy dowolny tekst, możemy użyć znaczników HTMLa i poleceń protokołu HTTP, aby lepiej zilustrować przykład, po tym wszystkim przerwijmy nasłuchiwanie kombinacją klawiszy *"CTRL+C"*. Przełączmy się teraz na okno przeglądarki i zaobserwujmy efekt. W ten sposób zostaliśmy serwerem WWW. Jednak chyba troszkę się do tego nie nadajemy - za bardzo lagujemy ;)

Znajdowanie zagrożeń - skanujemy porty

Czynność ta skojarzy nam się z kryminałami komputerowymi. Na przykład z książką Dana Vertona "Pamiętniki Hakerów". Samo skanowanie portów nie jest jednak czynem zabronionym w świetle polskiego prawa. Umożliwia nam ono poznanie usług jakie są uruchomione na danym komputerze. Taka informacja przyda nam się do tego, abyśmy mogli niepotrzebne porty pozamykać. W tym celu wywołujemy netcata z następującymi atrybutami:

```
$ nc -vz localhost 1-65535
```

Po jakimś czasie (około kilku minutach) na ekranie konsoli pojawi się podobny log:

```
localhost [127.0.0.1] 53710 (?) open
localhost [127.0.0.1] 50417 (?) open
localhost [127.0.0.1] 38841 (?) open
localhost [127.0.0.1] 3306 (mysql) open
localhost [127.0.0.1] 631 (ipp) open
localhost [127.0.0.1] 111 (sunrpc) open
localhost [127.0.0.1] 80 (www) open
localhost [127.0.0.1] 25 (smtp) open
localhost [127.0.0.1] 22 (ssh) open
```

Pseudo-shell - Zdalne wykonywanie poleceń

Zaraz przekonamy się, że nie potrzebujemy telnetu, ani ssh, aby wykonać zdalnie jakąś komendę. W tym celu mając włączonego netcata na nasłuchiwanie na jakimś porcie w innym oknie terminala wpisujemy:

```
$ nc -c "[komenda]" [host] [port]
```

Spójrzmy teraz w okno netcata. Komenda się wykonała.

Czy netcat może zastąpić telnet?

Oczywiście. Wystarczy wywołać komendę:

```
nc [host] [port]
```