Što, gdje, kako

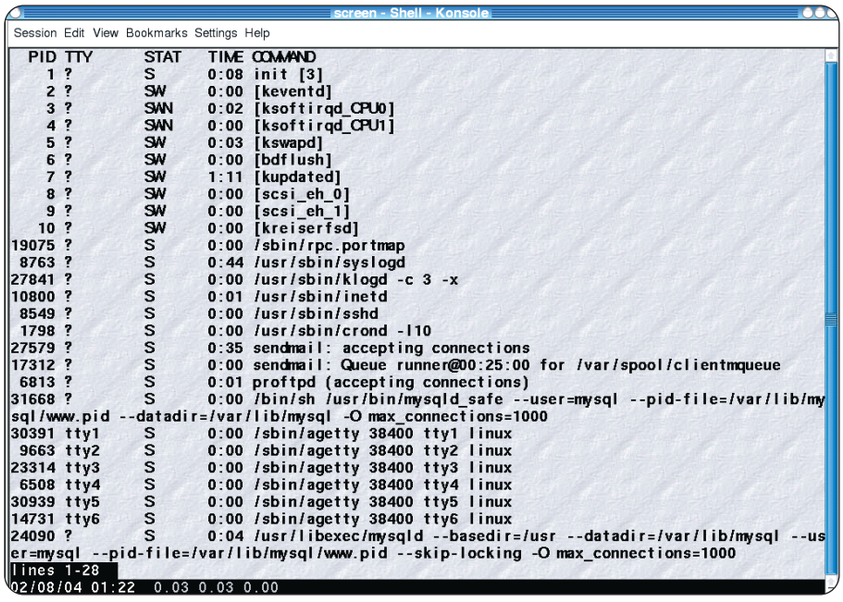
Da bismo nešto onemogućili, moramo se prvo uvjeriti da to postoji. Kako bismo se izvježbali u njihovom pronalaženju, moramo prvo ukratko objasniti kako se zapravo svi ti servisi pokreću. Jedan dio njih se ne nalazi u nekoj *startup* skripti koja se izvršava pri pokretanju sustava, već postoji poseban globalni servis imena inetd. Njegov je princip da osluškuje na zadanim portovima i pokreće aplikacije kad se za njim ukaže potreba. Tako se smanjuje potreba za resursima ako se radi o rijetko korištenim programima jer oni ne moraju biti svi pokrenuti istodobno. Podaci o njima nalaze se u /etc/inetd.conf.

To neće raditi dobro ako su ti servisi često zahtijevani, jer ih onda servis svaki put mora ponovno pokretati, čekati dok završe, zaustaviti ih i tako u krug. Dulje se čeka da se neka aplikacija pokrene, troši više procesora i disk je opterećeniji. Zato se potrebnije aplikacije pokrenu samo jednom na početku rada sustava. Nazivaju se *daemoni*, a najčešće kao zadnje slovo imena imaju "d" (inetd, sshd, httpd, ftpd itd.). Svi važniji servisi rade se kao *daemoni* pa njih često ni nećemo dirati, jer nam treba njihova funkcionalnost, odnosno bit su postojanja nekog servera. No to ipak nije slučaj sa svima, pa i njih moramo nekako locirati. Pogledamo li popis procesa na sustavu (ps aux l less), u desnom stupcu opazit ćemo zavidan broj takvih. Neke od njih ne smijemo dirati iako im zadnje slovo imena odgovara našem kriteriju, poput syslogda, klogda i kswapda. Oni ne pružaju nikakve mrežne usluge, već su nužni za ispravan rad sustava (prvi se brine o logiranju, a druga dva su *kernel* procesi).

Ako nismo sigurni što koji proces radi, uvijek možemo naredbom "man programd" pogledati njegovu *man* stranicu i tamo se informirati. No, nekad čak ni iz načina rada programa nije jasno li on sluša na mreži ili ne. Pritom mislimo na bazu MySQL, koja se (iako može slušati na portu ako joj dopustimo) za lokalne procese veže specijalnom *socket* datotekom i potpuno je nevidljiva izvana.

Skeniranje

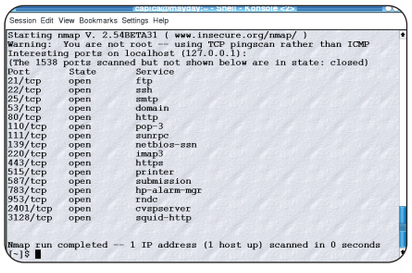
No, samo će istrenirano oko uspjeti pohvatati sve procese na taj način, razlučiti mrežne od sistemskih, sjetiti se pogledati u inetd.conf, prepoznati koji je za što i odlučiti koji mu trebaju. Nama smrtnicima jednostavnije je direktno pogledati zauzeće portova, pa se zatim raspitivati što ih je to zauzelo. Za taj pregled možemo koristiti daleko najjači dostupni alat kodnog imena nmap. Sintaksa mu je u našem slučaju vrlo jednostavna i s "nmap localhost" dobivamo popis svih otvorenih portova TCP sloja mreže. Za UDP moramo dodati opciju "-sU". Razlike između ova dva sloja za naša su promatranja



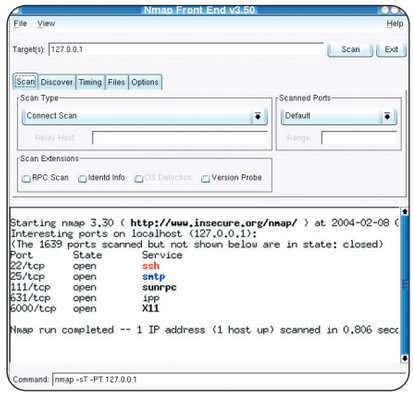
▲ **Popis procesa na sustavu otkriva velik broj *daemona* od kojih su neki ključni za rad sustava**

previše suptilne i izlaze iz okvira teksta, no recimo da je moguće u računalo upasti putem bilo kojeg, ako ga servis koristi. TCP je nešto češći u klasičnoj internetskoj komunikaciji, dok je UDP češći u *streaming* prometu, gdje nije previše bitan minimalni gubitak informacija, već njihova pravovremena isporuka.

Nakon što smo se nekako dočepali popisa, čudom se čudimo svim tim kriptičnim skraćenicama, za koje nam nije baš jasno što znače. Da bismo se malo bliže upoznali s njima (želimo saznati ime programa i korisnika), koristit ćemo "netstat -tuple" koji će nam u pomalo opskurnoj formi vratiti tražene podatke. Inače, vrlo je korisno pobliže se upoznati s kombinacijama port/servis, a neke češće smo naveli u tablici (cjelovit popis nalazi se u /etc/services). Koji od njih nam smeta, neće biti uvijek lako odrediti, no ako zaustavimo krivi i nešto se potrga, uvijek ga možemo ponovno pokrenuti.



▲ **Popis svih otvorenih portova na sustavu lako se dobiva programom Nmap**



▲ **Grafičko sučelje programa Nmap jednostavno je za korištenje**

Često su kao nesigurni servisi označeni oni koji na neki način primaju nekriptiranu lozinku, pa bi tako bilo poželjno izbjegavati telnet, ftp, neke starije (rsh, rlogin) pa čak i pop3 za *mail*. Svi - pa makar sami po sebi bili najsigurniji na svijetu - mogu otkriti lozinke bilo kojeg korisnika i to bi haker uz vrlo malo truda mogao iskoristiti za spajanje.

Istjerajmo demone iz pingvina

Sad dolazi najkorisniji dio posla. Sve te suvišne programe trebamo locirati, zaustaviti i onemogućiti im pokretanje kod *restarta* sustava. Počnimo od stvari koju smo prvu