Nesze neked szita!

Vannak emberek, akik a végtelenben is folyamatosan keresik a legnagyobbat.

Hogyha a számítógépes tudomány jelenlegi állását illetően szeretnénk borongani, és most itt nem a Windows-rendszerek futtatására–összeomlására gondolok, akkor érdemes átfutni *Peter Neumann* RISKS jelentéseit és a SANS Institute által rendszeresen megjelentetett „sebezhető pontok” listáját. Úgy tűnik, hogy egyetlen operációs rendszer, rendszermag, fordító, könyvtár, szintaktikai elemző és parancs sem mentes ezektől, legyen az bármennyire is „érett”. Most nem ezermilliárdos könyvelési hibákról vagy eltűnt űrhajókról beszélünk, hanem teljesen mindennapos, mindenki számára nyilvánvaló, el nem kapott kivételekről, amelyeket, a véget nem érő figyelmeztetések ellenére, a C fondorlatos tömbhasználatának köszönhetünk.

Kapcsolódó címek

1. Az említett könyvhöz kapcsolódó webhely:
➲http://www.realworldlinuxsecurity.com/

2. Kapcsolódó cikk: „GIMPS Finds Anoter Prime”
Dale Buske–Sandra Keith, Math Horizons, 2000. április.

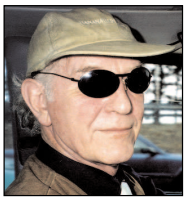
Honlap: ➲http://www.mersenne.org/prime.htm

Egy nemrégiben megjelent SANS-jelentés az Access, a Word 2000 és az Internet Explorer 5.0+ alkalmazásokban rejtőző veszélyekre hívja fel a figyelmet. A SANS díjakat ajánl fel azok számára, akik „gyors” megoldásokkal szolgálnak, ezek a hibák azonban meglehetősen „sajátságosak”.
A Microsoft, nézzünk szembe a tényekkel, több mint egy profi programozókból álló csapat, akiket a legfrissebb fejlesztési tanfolyamokon képeztek ki. A Microsoft kiadója (a Microsoft Press) sorra ontja magából azokat a könyveket, amelyek arról szólnak, hogy miként készíthetünk bizonyíthatóan nagy teherbírású alkalmazásokat.
Azonban úgy tűnik, hogy a „nyilvánvaló” hibák (a memóriaszivárgások, a túlcsordulások) kiszűrésére szolgáló nyelvi, illetve fejlesztőeszközök mit sem érnek; a MS programozóinak ez az elit csoportja még így is tud hibázni.
A hatalmas kihívás abban rejlik, hogy mielőtt nekilátnánk a hibák kiirtásának, meg kell tudnunk határozni azokat. Még Stephen Hawking igen tágan értelmezhető *időhisztográfiája* sem teszi lehetővé számunkra, hogy simára csiszolt kódunk minden egyes if–then–else ágát végignézzük. Egyáltalán mit jelent az „emberi” nyelvre lefordítva, hogy kódunk megfelel az elő- és utófeltételeknek? (Ha a Hawking-féle időmegfordítás az Ősrobbanáshoz vezet vissza bennünket, akkor a GOTO-k helyén COMEFROM-okat látunk majd?)
A nyílt kódú megközelítés sikere azon alapul, hogy olyan elfogulatlan programozók is hozzáférhetnek a kódhoz, akikben tényleg a cél elérése dolgozik, szemben azokkal a belső „szolgákkal”, akik úgy gondolják, hogy részvényük nem ér annyit, hogy komolyabb erőfeszítéseket tegyenek.
Mindezek ellenére, a Linux rendszermagjában nemrégiben felfedezett biztonsági hibák azt jelzik, hogy még a többször, gondosan átnézett, egyszerű kódrészletek is okozhatnak meglepetéseket. A dolognak természetesen megvannak a jó oldalai is: (i) a hibákat gyorsan és nyíltan beismerik, kijavítják; (ii) a bűnbakok listája végtelen – fel a kezekkel, bárki is volt az a száznegyvennégyezer ember közül. *Peter Salus* feljegyzi majd a vallomásukat.
A témához kapcsolódik *Bob Toxen* hamarosan megjelenő könyve, a *Real World Linux Security* (lásd a kapcsolódó címeket).

GIMPS

Ugorjunk át a széles körben elterjedt, nyílt számítástudomány egy másik alkalmazási területére. A GIMPS (Great Internet Mersenne Prime Search – Nagy Internetes Mersenne-prím keresés) programban több mint 8000 egyéni felhasználó vesz részt, akik kivétel nélkül arra kötelezték el magukat, hogy minél nagyobb prímszámokat találjanak. Euklidesz bebizonyította annak idején, hogy a keresésnek sosem lesz vége, mivel ha P a legnagyobb prímszám, akkor P!+1 vagy egy P-nél nagyobb prím, vagy pedig rendelkezik P-nél nagyobb prímosztóval. A dolog tehát bebizonyított!

Azt mondják, hogy ha nem látjuk ennek a bizonyításnak a szépségét, akkor sosem lesz belőlünk matematikus.
Ha viszont mégsem hiszünk a bizonyításnak, és szeretnénk a saját szemünkkel is meggyőződni annak igazáról, akkor ezt könnyen megtehetjük: Erasztotenész (egy másik halott görög) szitája egy olyan eljárást ad, amelyik az összes prímszámot felsorolja növekvő sorrendben. Saját Kelly-szitám az összetett *és* a prímszámokat is képes felsorolni egyetlen for-ciklus segítségével.
A kérdés az, hogy vannak-e prímek P és P!+1 között – és 1999. júniusában a GIMPS csapatban tevékenykedő Nayan Hajratwala új (átmeneti) világrekordot állított fel: 2^6,972,593 – 1
Megkímélem az Olvasót attól a tíz kilométer hosszúságú számsortól, amely ennek a számnak a printf()-fel történő kiíratásához szükséges.
A jövő hónapban: hogyan találhatunk idegen világkultúrát a SETI együttműködési programjával. A kis zöld matematikusok talán már ismerik a következő legnagyobb prímet! Addig is látogassanak el a kutatás honlapjára
(➲http://www.setiathome.ssl.berkeley.edu/) és csatlakozzanak a Linux Journal csoporthoz!



Stan Kelly-Bootle (skb@atdial.net) az ötvenes években épített EDSAC I (Anglia, Cambridge Egyetem) óta van jelen a számítástechnikában. Rengeteg cikket és számos könyvet is megjelentetett, többek között a Computer Contradictionary (MIT Press) és a UNIX Complete (Sybex) címűeket. Rendszeresen ír a ➲http://sarcheck.com/ és a ➲http://www.unixreview.com/ oldalaira.