

# Wiadomości Kernel

ARTUR SKURA

## ■ Klastrowy system komunikacji międzyprocesowej dla Linuksa

Firma Ericsson udostępniła na podwójnej licencji GPL/BSD niezwykle interesujący projekt: przezroczysty system komunikacji międzyprocesowej (Transparent Inter Process Communication, TIPC)[1], przeznaczony głównie dla klastrów. Jego podstawową cechą jest ukrycie topologii klastra przed aplikacją, dzięki czemu jej twórca nie musi martwić się o to, czy przekazywanie wiadomości następuje lokalnie, czy między odległymi węzłami klastra. Protokół ten jest również bardzo wydajny (w porównaniu do implementacji obsługujących część jego funkcjonalności korzystających np. z TCP/IP), a równocześnie zapewnia częściową obsługę redundancji, jakości usługi oraz pozwala usługom „subskrybować” stan dostępności (lub niedostępności) danego zasobu. Jak dotąd nie istnieje inny tego typu protokół, spełniający tak wiele funkcji. Jego linuksowa implementacja może zachęcić twórców rozwiązań klastrowych do poważniejszego zainteresowania się tą platformą, już teraz bardzo popularną w tym segmencie rynku. Nie bez znaczenia jest również fakt, że TIPC jest już wykorzystywany przez Ericssona, nie jest to więc kolejny projekt li tylko akademicki. ■

## ■ Secure Computing?

Andrea Acangeli przedstawił minimalną implementację bezpiecznego środowiska wykonywania programów, w którym procesy mogą uruchamiać jedynie kilka wywołań systemowych. Jest to niezwykle lekkie i mało złożone rozwiązanie, mogące sprawdzić się w niektórych zastosowaniach, mających na celu np. oddzielenie grup procesów czy też stworzenie bezpiecznej „piaskownicy”, w której procesy nie obdarzone zaufaniem mogłyby bez przeszkód wykonywać swój kod, w dodatku bez spadku wydajności. ■

## ■ Koniec cryptoloop

Andrew Morton zapowiedział, że popularny mechanizm szyfrujący w jądrze 2.6, *cryptoloop*, zostanie usunięty w wersji 2.6.9. Powodów jest wiele: implementacja jest ogólnie uważana za słabą i nie pozbawioną błędów, a brak jest osoby utrzymującej projekt, która mogłaby się nimi zająć. Alternatywę dla *cryptoloop* przedstawia *dm-crypt*, który – choć nie testowany w stopniu porównywalnym z *cryptoloop* – jest pozbawiony wielu jego błędów, nie zależy od oprogramowania operującego w przestrzeni użytkownika (w tym wypadku *util-linux*) i jest stosunkowo wydajny. Użytkownicy korzystający z *cryptoloop* powinni przyrzeć się *dm-crypt*, ponieważ mimo kilku protestów decyzja Andrew jest nieodwołalna. ■

## ■ Oficjalne jądra niekoniecznie stabilne...

Ta informacja może wydać się nieco zaskakująca, jakkolwiek w dużej mierze odpowiada już istniejącej sytuacji. Otóż priorytetem w stabilnej wersji jądra – 2.6, jest funkcjonalność i wydajność, lecz niekoniecznie stabilność. W tej chwili model rozwoju serii 2.6 przedstawia się następująco: nowe łaty, przeznaczone do głównej gałęzi jądra, trafiają do drzewa Andrew Mortona, oznaczonego sufiksem *-mm*. Następnie, po okresie testów, zostają dołączone do kolejnego oficjalnego wydania jądra. Linus Torvalds wydaje się być bardzo zadowolony z takiego sposobu prac nad projektem i niechętnie zapatruje się na powstanie gałęzi 2.7. Wychodzi tu na wierzch powszechna bolączka projektów Open Source, zwłaszcza tych o wysokim stopniu komplikacji i o dużym znaczeniu dla stabilności systemu – użytkownicy stosunkowo rzadko używają wersji testowych. Mała ilość testów przyczynia się do większej ilości błędów w finalnej wersji. Linus korzysta więc ze strategii, której przykład można było zobaczyć np. przy okazji wydania wersji 2.4.0: jądro, którego nie można do końca nazwać stabilnym, zostaje przypisany

numer wersji odpowiadający wersji stabilnej, przez co wiele osób je instaluje, w następstwie czego ma miejsce wystarczająca ilość testów i zostają wykryte błędy, które trudno było znaleźć inaczej.

Trudno narzekać na Linusa – 2.6 nie jest może najstabilniejszym jądrem, jakiego zdarzyło nam się używać, lecz podobne problemy istniały na początku każdej stabilnej wersji, zaś obecny poziom wydaje się na tyle zadowalający, by większość dystrybucji zdecydowała się udostępnić 2.6 swoim użytkownikom, przynajmniej jako opcję. Ogromną korzyścią jest za to fakt, że w ciągu miesiąca do gałęzi *-mm* przybywa około 10 megabajtów łat! Jest to niespotykane dotąd tempo rozwoju. Oczywiście szybkie przedostawanie się łat do głównego jądra to najczęściej bardzo korzystne zjawiska, w ten sposób bowiem nowe jądro zyskuje nie tylko nową funkcjonalność, ale przede wszystkim nowe sterowniki i aktualizacje starych. Użytkownicy sprzętu, który był traktowany po macoszemu przez niektórych programistów jądra, z pewnością docenią ten fakt.

Co więc pozostaje użytkownikom, którym zależy głównie na stabilności? Andrew odsyła ich do dystrybutorów, na których, jak twierdzi, spoczywa obowiązek przetestowania sprzedawanego przez siebie oprogramowania pod kątem stabilności – ostatecznie bowiem to dystrybutorzy odpowiadają przed swoimi klientami. Nie należy też oczekiwać szybkiego pojawienia się gałęzi 2.7, bowiem to właśnie w 2.6 znajdzie się większość nowości, oczywiście po okresie testowania w gałęzi Andrew Mortona. Można więc spodziewać się dużych zmian, takich jak np. długo zapowiadane usunięcie *devfs*, jeszcze w wersji 2.6. ■

## INFO

[1] Więcej informacji o TIPC można znaleźć na jego stronie domowej na SourceForge: <http://tipc.sourceforge.net>

[2] Strona domowa *dm-crypt*: <http://www.saout.de/misc/dm-crypt/>