

# Kernel News

## ■ Nowe życie

Ian Kent zgodził się przejąć kierownictwo nad DevFS (wirtualny system plików urządzeń). Systemu DevFS, napisanego przez Richarda Goocha, nie czekała błyskotliwa kariera. Przez ostatnie lata DevFS był poprawką zewnętrzną, ostatecznie zatwierdzoną w jądrze 2.4 przez Linusa, aczkolwiek w troszkę odmiennej formie. Niektórzy twórcy oprogramowania twierdzili, że poprawka ma pewne wady wrodzone, których nie da się usunąć, a więc najlepiej usunąć ją z jądra i zastąpić czymś innym. Ostatnio następcą DevFS został wybrany kod udev Grega KH. Jednakże w obecnej fazie udev wciąż nie jest w stanie zastąpić całkowicie DevFS.

## ■ Głośna debata o XFS

XFS (wysoko wydajny system plików) został ostatecznie umieszczony w jądrze 2.4. Stało się to w ostatniej chwili. Kiedy okazało się, że wydanie wersji 2.6.0 nadchodzi wielkimi krokami, Marcelo Tosatti zdecydował nie umieszczać tego systemu w wersji 2.4, odmawiając dalszych prac nad implementacją w tej wersji.

Natychmiast po ogłoszeniu tej decyzji pojawiły się głosy rozczarowanych programistów, spośród których programiści XFS protestowali najbardziej. Ostatecznie przygotowano jednak, z pomocą Marcelo, wersję systemu plików XFS dla jądra 2.4

Marcelo zgodził się na zdjęcie embarga z wersji 2.4, prosząc jednocześnie twórców oprogramowanie o przeprowadzenie do-

Nagle wycofanie się Richarda Goocha z zespołu programistów pracujących nad jądrem systemu, pociągnie prawdopodobnie za sobą w rezultacie usunięcie DevFS. Okazało się jednak, że niektóre osoby (m.in. Ian) chcą stanąć murem, broniąc DevFS i przejąć go pod swoje skrzydła. Andrew Morton zauważył, że chętnie umieści poprawki DevFS w jądrze 2.6, jeżeli prace nad DevFS będą przebiegały w określonym kierunku oraz do czasu, gdy udev będzie w stanie całkowicie zastąpić swojego starszego brata. Andrew podkreśla jednak, że być może DevFS zostanie zachowany również w jądrze 2.8. Tak więc, DevFS otrzymał drugą szansę i narodził się na nowo. ■

kładnych, rzetelnych testów tego systemu. Jednym z najgłośniejszych i najczęściej cytowanych argumentów podczas debaty publicznej było to, że takie systemy jak ReiserFS, ext3, czy inne główne systemy plików są już obsługiwane od dawna przez jądro 2.4, a XFS nadal pozostaje w tyle, wymagając osobnej poprawki dla jądra.

Po umieszczeniu systemu plików XFS w jądrze 2.4, Marcelo musiał także zaakceptować sprawę dodatkowych poprawek XFS przez pewien czas. Generalna decyzja o drastycznym zmniejszeniu ilości wychodzących poprawek wydaje się bronić swojej pozycji – debata dotycząca systemu plików XFS niewątpliwie odegrała ważną rolę w podejściu Marcelo do tego tematu. ■

## ■ Szyfrowanie nadrzędne

Christophe Saout napisał dm-crypt, adresat mapowania urządzeń obsługujący szyfrowanie systemu plików. Zgodnie z tym, co powiedział Fruhwirth Clemens, twórca cryptoloop, dm-crypt Christophe'a jest dużo lepszym mechanizmem niż cryptoloop, jeśli chodzi o obsługę szyfrowania opartą na systemie plików.

Fruhwirth optował za utrzymaniem cryptoloop w wersji 2.6 jądra jako funkcji nie zalecanej do chwili pełnego uruchomienia dm-crypt. Po tym czasie należałoby usunąć cał-

kowie cryptoloop, a loopback wykorzystywać wyłącznie w odniesieniu do plików. Wydaje się, że Andrew Morton popiera takie podejście w stosunku do jądra 2.6, a cryptoloop zostanie usunięty całkowicie mniej więcej po wyjściu nowej wersji jądra (2.8).

Cryptoloop, całkiem nowoczesny sposób szyfrowania, podjął się rozwiązania problemu przezroczystego zarządzania szyfrowanymi danymi w nieszyfrowanym systemie plików. Od lat użytkownicy żądni maksymalnego poziomu zabezpieczeń czekali na

## ■ Pamięć RAM na gorąco

Yasunori Goto przygotował poprawkę umożliwiającą użytkownikom systemów pracujących w architekturze NUMA (rozproszony dostęp do pamięci) dokładanie pamięci RAM podczas pracy. W przyszłości chciałby on przygotować poprawkę zarówno dla komputerów 32-bitowych, jak i 64-bitowych.

Pomysł podłączania na gorąco stanowił zawsze niedoścignione marzenie i zarazem duże wyzwanie dla twórców systemów. Dołożenie pamięci w pracującym systemie było uważane za coś podobnego do oderwania się od ziemi wyłącznie przy pomocy machania rękami.

Oczywiście w pewnych przypadkach podłączenie podczas pracy nawet zwykłej klawiatury jest po prostu niemożliwe ze względu na iskrzenie, które mogłoby uszkodzić czułe podzespoły systemu. W takich sytuacjach (tzn. tam, gdzie sprzęt uniemożliwia podłączanie urządzeń podczas pracy) system sam w sobie niewiele może pomóc. W praktyce, podłączanie myszy czy klawiatury jest czynnością rutynową, a mimo to na niektórych systemach istnieje realne ryzyko uszkodzenia systemu.

Obecnie projektuje się coraz więcej sprzętu gotowego do współpracy z urządzeniami podłączanymi na gorąco. Stwarza to, między innymi, możliwość tworzenia systemów klockowych (lub też wieżowych), w których podłączenie nowego napędu, czy innego sprzętu, będzie wiązało się z dołączeniem go do pracującego systemu, podobnie jak w klockach LEGO. ■

takie rozwiązanie.

Początkowo cryptoloop został ogłoszony cudownym rozwiązaniem problemu. Obecnie ta sama przyszłość czeka prawdopodobnie jego następcę, dm-crypt, który jest kolejnym krokiem naprzód w drodze do idealnego bezpieczeństwa systemu. Możemy być jednak pewni, że każde następne rozwiązanie dotyczące przechowywania szyfrowanych danych będzie okupowane mnóstwem głosów i próśb o dalszy rozwój i nowe funkcje zwiększające poziom bezpieczeństwa. ■

## ■ Jądro Linuksa 2.6 – po premierze

Pod koniec grudnia 2003 roku wydano nowe jądro Linuksa, wersję 2.6.0. Spełniono w ten sposób nadzieje Linusa na przekazanie użytkownikom kolejnej, stabilnej wersji jądra przed końcem roku. Po raz pierwszy wczesne przewidywania co do daty wydania nowej wersji jądra okazały się prawdą. Przygotowanie poprzednich wersji jądra 2.0.0, 2.2.0 i 2.4.0 zajęło dużo więcej czasu niż ktokolwiek przewidywał. Przed wydaniem wersji 2.0 struktura pracy nad projektem była jeszcze bardziej chaotyczna.

Niewątpliwie powodem lepszej organizacji i kontroli nad przebiegiem prac nad wersją 2.6 było zaadaptowanie przez Linusa systemu kontroli wersji BitKeeper, co jest kolejną kontrowersyjną decyzją. Sukces jest jednak argumentem na tyle silnym, że system BitKeeper pozostanie głównym narzędziem do zarządzania rozwojem Linuksa.

Kolejnym powodem punktualności wydania wersji 2.6.0 jest niewątpliwie twarda polityka Linusa w stosunku do poprawek. Jądro zostało mocno zamrożone, co oznacza, że dalsze dodawanie nowych funkcji, mechanizmów czy zmian kodu ('feature freeze' i 'code freeze') nie miało miejsca (doprowadziło to w przypadku poprzednich wersji jądra do sporych opóźnień spowodowanych błędami i wprowadzanymi poprawkami).

Kiedy wydawało się, że zamrożenie kodu nie było wystarczające, Linus nie ogłaszał kolejnego zamrożenia projektu, lecz podtrzymał pierwotne wstrzymanie prac, jako jedyne obowiązujące przez cały czas. Takie rozwiązanie może wydawać się okrutne dla niektórych programistów, których poprawki były odrzucane, podczas gdy inne projekty przyjmowano.

W poprzednich stabilnych wersjach jądra Linus był częściej skłonny do całkowitego porzucenia zamrożenia projektu. Przy

pracach nad wersją 2.6.0 ten problem nie występował. Celem prac nad wersją 2.6.0 była obsługa zarządzania jądrem i zależnościami pomiędzy obecnymi wersjami stabilnymi i niestabilnymi.

Tradycyjnie, po pierwszym wydaniu nowej wersji stabilnej następuje okres oczekiwania przed pojawieniem się pierwszej wersji niestabilnej. W tym czasie tylko jedna wersja jądra jest prawdziwie obowiązująca. Przez resztę czasu trwają równoległe prace, zarówno nad wersją stabilną, jak i niestabilną.

Przykładowo, w przypadku wersji 2.4 i 2.5 Marcelo Tosatti – kierownik projektu wersji 2.4, spowolnił prace nad tą wersją, gdy tylko okazało się, że wersja 2.5 przekształca się w wersję 2.6. Teraz, kiedy wydano wersję 2.6, Marcelo będzie coraz rzadziej przyjmował poprawki, które nie usuwają bezpośrednio błędów jądra. W tym samym czasie prace nad wersją 2.6 będą postępować stosunkowo szybko w porównaniu z ostatnimi dniami wersji 2.5 (gdy tylko pojawi się wersja 2.7). Taka zależność na zasadach symbiozy pomiędzy wersją stabilną i niestabilną jest jedną z zalet tego systemu.

Dobrze jest dysponować najnowszą, oficjalną, stabilną wersją jądra z maksymalną liczbą sprawdzonych funkcji. Musimy jednak pamiętać, że gwałtowny rozwój wersji stabilnej mógłby ją szybko „zdestabilizować”. Rozwiązaniem tej sytuacji jest prowadzenie prac nad funkcjami jądra w wersji niestabilnej, a następnie umieszczenie ich w wersji stabilnej (po zakończeniu procesu stabilizacji funkcji).

W wielu przypadkach takie działanie umożliwiło stworzenie nowocześniejszej i bardziej zaawansowanej wersji stabilnego jądra, przy minimalnym ryzyku destabilizacji. Decyzja Marcelo o gwałtownym ograniczeniu prac nad wersją 2.4 była w pewnym sensie decyzją społeczną: powinna istnieć

tylko jedna wersja stabilna jądra Linuksa. Po udostępnieniu użytkownikom wersji 2.6 zdecydowano, że taka terminowość prac dla wersji 2.4 nie jest już konieczna.

Mimo że wiele funkcji można zastosować z powrotem w wersji stabilnej, wielu rozszerzeń, czy nowych funkcji jądra nie będzie można przenieść. Nie będzie można także usunąć niektórych błędów.

Wersja 2.6 będzie z pewnością lepszym jądrem niż wersja 2.4. Kiedy Linus rozdzieli wersję 2.7 od wersji 2.6, możemy oczekiwać, że Andrew i inni zaczną przenosić niektóre funkcje z wersji 2.7 z powrotem do 2.6, gdy te zostaną uznane za funkcje stabilne. Ile czasu upłynie zanim wersja 2.7 zacznie stabilizować się jako wersja 2.8? Poczekamy, zobaczymy...

### INFO

**Kernel Mailing List** to podstawowe źródło informacji o bieżących pracach rozwojowych nad jądrem Linuksa. Ilość nadsyłanych listów i wątków dyskusji sprawia, że jest to niemal nie do ogarnięcia przez jednego człowieka. Na szczęście znalazł się dzielny duch, który podjął się niemożliwego – jest nim Zack Brown.



**Kernel News** to regularna, comiesięczna kolumna, która zawiera informacje na temat najnowszych trendów rozwojowych jądra. Najciekawsze tematy są wybierane i opisywane przez Zack-a. Zack przez wiele lat przygotowywał oficjalne, cotygodniowe podsumowanie dyskusji dla Kernel Traffic Mailing List.

Linux Magazine przedstawia zatem krótkie podsumowanie dyskusji na liście Linux Kernel.



## Prenumerata Linux Magazine Nie przegap takiej okazji

Zamawiając prenumeratę oszczędzasz!  
Płacisz jak za 9 numerów, a otrzymujesz 12!  
Z każdym numerem DVD lub płyta CD-ROM.

Najszybszy sposób zamówienia prenumeraty:

<http://www.linux-magazine.pl>

Infolinia: 0801-800-105