

Bardzo szczegółowy przewodnik po aktualizacji i kompilacji jądra w Debianie

Clinton de Young
polskie tłumaczenie Polish Debian Documentation Project (PDDP)

v0.11 17 maj 2003

Prawa autorskie

Copyright © 2002, 2003 Clinton De Young

Copyright © polskie tłumaczenie Polish Debian Documentation Project (PDDP)

Dokument przetłumaczony przez grupę Polish Debian Documentation Project (<http://debian.linux.org.pl>).

Polskie tłumaczenie tego przewodnika jest objęte licencją GNU FDL (Free Documentation License). Został on napisany w nadziei, że będzie użyteczny dla społeczności użytkowników Debiana, ale nie udziela się na niego żadnej gwarancji; używaj go wyłącznie na własne ryzyko.

Spis treści

1	Przedślowie	1
1.1	O autorze	1
1.2	Tłumaczenie	1
2	Wprowadzenie, pobieranie jądra	3
2.1	Wstęp	3
2.2	Wprowadzenie, pobieranie jądra	3
2.3	Oczekiwania	3
2.4	Pobieranie nowego jądra	4
3	Budowanie jądra krok po kroku, wymagania	7
3.1	Przygotowanie źródeł jądra 2.4.20	7
3.1.1	Krok 1: Otwórz okno terminala	8
3.1.2	Krok 2: Przejdź do katalogu do którego zapisałeś plik <code>linux-2.4.20.tar.gz</code>	8
3.1.3	Krok 3: Zaloguj się jako administrator używając <code>su</code>	8
3.1.4	Krok 4: Przenieś plik <code>linux-2.4.20.tar.gz</code> do katalogu <code>/usr/src</code>	8
3.1.5	Krok 5: Sprawdź katalog <code>/usr/src</code>	8
3.1.6	Krok 6: Rozpakuj plik <code>linux-2.4.20.tar.gz</code>	10
3.1.7	Krok 7: Utwórz dowiązanie symboliczne o nazwie <code>linux</code> wskazujące na <code>linux-2.4.20</code>	10
3.2	Pliki potrzebne do konfiguracji i kompilacji jądra	10
4	Konfiguracja jądra	11
4.1	Uruchamianie narzędzia do konfiguracji jądra	11
4.2	Konfiguracja jądra Linuksa	12

5	Konfiguracja jądra do obsługi dźwięku, ext3 i nagrywarek	15
5.1	Konfiguracja jądra do obsługi dźwięku	15
5.2	Konfiguracja Debiana do obsługi nagrywarek	15
5.3	Włączanie obsługi systemu plików Ext3 (Chcesz to zrobić!)	17
6	Kompilacja, instalacja jądra	19
6.1	Zapisywanie konfiguracji jądra i kompilacja	19
6.2	Instalacja nowego jądra	20
7	Ext3, rozwiązywanie problemów, zakończenie	21
7.1	Ustawianie komputera by używał systemu plików z księgowaniem Ext3	21
7.2	Co jeśli coś pójdzie nie tak?	23
7.3	Zakończenie	23

Rozdział 1

Przedślowie

1.1 O autorze

Nazywam się Clinton De Young i pracuję jako Development Manager dla firmy programistycznej Altiris. Biegłe władam angielskim oraz płynnie japońskim. W wolnym czasie tłumaczę dokumenty z i na język japoński a czasami współpracuję z różnymi firmami w pisaniu programów na rynek japoński. Pomimo tego wszystkiego, na pierwszym miejscu stawiam rodzinę. Są oni najważniejszymi ludźmi w moim życiu.

(Adres oryginalnego artykułu http://www.osnews.com/story.php?news_id=2949)

1.2 Tłumaczenie

Do powstania polskiej wersji językowej tego dokumentu przyczynili się następujący członkowie grupy PDDP:

Tłumaczenia:

- Bartosz Feński <fenio@o2.pl>

Korekta:

- Oskar Ostafin

Więcej o grupie PDDP możesz dowiedzieć się na naszej oficjalnej stronie <http://debian.linux.org.pl>

Rozdział 2

Wprowadzenie, pobieranie jądra

2.1 Wstęp

Jeśli czytasz ten artykuł to zakładam, że wiesz czym jest Jądro Linuksa (<http://www.kernel.org>) i dlaczego mógłbyś chcieć je zaktualizować. Jednakże, jeśli dziwnym trafem czytasz ten przewodnik, a po prostu używasz Linuksa i nie masz pojęcia czym jest jądro (kernel) i dlaczego miałbyś je aktualizować, to następne dwa paragrafy są dla Ciebie (jeśli szukasz mniej szczegółowej instrukcji aktualizacji jądra na jakiegokolwiek dystrybucji kliknij tutaj (http://www.osnews.com/story.php?news_id=443)). Najprościej ujmując jądro Linuksa to mózg systemu. Mówi systemowi jakie systemy plików, sprzęt, protokoły itd., są obsługiwane. Oczywiście robi jeszcze wiele innych rzeczy, ale na razie taka mała definicja nam wystarczy.

2.2 Wprowadzenie, pobieranie jądra

Moje podziękowania

Zanim zaczniemy, chciałem serdecznie podziękować wszystkim ludziom, za odzew po moim ostatnim artykule Bardzo Szczegółowy Przewodnik po Instalacji Debiana 3.0 (http://www.osnews.com/story.php?news_id=2016). Jestem wdzięczny za wgląd, poprawki i komentarze. Chciałem również przeprosić, za opóźnienie związane z tym artykułem. Moje życie przeszło wiele zawirowań w ostatnich kilku miesiącach. Cieszę się, że ostatecznie znowu jestem w formie.

2.3 Oczekiwania

Dlaczego mógłbyś chcieć bawić się z jądrem? No cóż, wszyscy ludzie (będący troszkę ponad zwykłymi śmiertelnikami i mądrzejsi ode mnie), którzy nieustannie pracują nad jądrem, dodają do niego obsługę nowego sprzętu, nowych technologii, poprawiają istniejące błędy i dokładają mnóstwo innych usprawnień. Aktualizując jądro zwiększamy możliwości naszego

systemu. Z tego co wiem, nie został jeszcze wybudowany komputer, który wykorzystywałby wszystkie sterowniki i możliwości jądra Linuksa. Zatem, jeśli nie musisz, po co instalować wsparcie dla wielu rzeczy, które nie będą Ci potrzebne.

W tym przewodniku, postaram się opisać proces aktualizacji i kompilacji jądra Linuksa w dystrybucji Debian. Spróbuję wyjaśnić poszczególne kroki w taki sposób by rozwiązać wszelkie przesady i obawy. Mam nadzieję, że ktoś nowy w Linuksie przejdzie przez wszystkie etapy bezproblemowo i na końcu będzie miał zaktualizowany system.

Mimo, iż chcę by każdy przeszedł ten proces bezboleśnie i bezbłędnie, to jednej rzeczy nie mogę przewidzieć. Jest to szczegółowy opis sprzętu, którym dysponujesz oraz rzeczy, które system ma obsługiwać. Zatem skupię się na trzech najbardziej popularnych tematach: dźwięk, systemy plików z księgowaniem i możliwość wypalania CD. Myślę jednak, że dostarczę wystarczająco dużo szczegółów byś nie miał żadnych problemów z resztą rzeczy i na przykład włączeniem obsługi USB.

Wiedza na temat sprzętu, którym dysponujesz jest absolutnie konieczna jeśli chcesz zoptymalizować jądro pod ten konkretny sprzęt. Przykładowo, żeby wkompiłować obsługę karty dźwiękowej, którą posiadasz, musisz wiedzieć jaki to rodzaj karty. Często potrzebne informacje możesz znaleźć w dokumentacji, którą otrzymałeś wraz ze sprzętem, z Windowsowego Menedżera Urządzeń (jeśli używasz Windowsa) lub ze strony producenta od którego kupiłeś komputer (zwłaszcza jeśli pochodzi od firmy takiej jak Gateway, HP czy Dell).

Nie zniechęcaj się jednak jeśli nie wiesz jaki sprzęt posiadasz. Pomimo, że możesz mieć problemy z konfiguracją jądra pod specyficzne możliwości płyty głównej, to jednak nadal pozostaje wiele usprawnień, które możesz dołożyć do systemu, choćby obsługę systemu plików z księgowaniem EXT3, a naprawdę warto.

Jeszcze jedno zanim zaczniemy. Część opisów z tego przewodnika można zastosować w każdej dystrybucji Linuksa, ale część jest specyficzna tylko dla dystrybucji opartych na Debianie, takich jak Libranet czy Xandros, które wyewoluowały z dystrybucji Debian. Pomimo, że nie będę opisywał "niedebianowskiego" sposobu aktualizacji i kompilacji jądra, to część opisującą konfigurację można zastosować w każdej dystrybucji (pod warunkiem, że posiadają system X-Window).

Ok. Zaktualizujmy jądro.

2.4 Pobieranie nowego jądra

Jeśli czytałeś i zastosowałeś się do "Bardzo Szczegółowego Przewodnika po Instalacji Debiana 3.0", powinieneś mieć aktualnie zainstalowane w systemie jądro 2.2.20. To jądro nie obsługuje takich rzeczy jak USB czy system plików EXT3. Jeśli masz bardzo duży dysk twardy i kiedykolwiek musiałeś czekać przez trzy lata aż Linux sprawdzi partycje po niepoprawnym zamknięciu systemu, to z pewnością pokochasz systemy plików z księgowaniem takie jak EXT3 i na pewno będziesz chciał mieć ich obsługę w swoim jądrze.

W tym przewodniku zaktualizujemy jądro do wersji 2.4.20 (które było najnowszym w chwili gdy zaczynałem pisać ten artykuł. Nie krępuj się pobrać nowszego jądra jeśli chcesz). By tego

dokonać musimy pobrać źródła jądra dla tej wersji. By pobrać te źródła możesz użyć swojej ulubionej metody (wget, ncftp itd.), jednakże w tym przewodniku zamierzam pobrać jądro 2.4.20 za pomocą Mozilli ponieważ powinniśmy mieć ją zainstalowaną w systemie i większość ludzi jest obeznana z pobieraniem plików za pomocą przeglądarki.

By pobrać najnowszą wersję jądra (2.4.20) uruchom Mozillę (lub swoją ulubioną przeglądarkę WWW) i udaj się na stronę Archiwum Jąder Linuksa (<http://www.kernel.org>). Na głównej stronie powinieneś pod tytułem zobaczyć prostokątną część z odnośnikami do serwera HTTP, FTP oraz RSYNC. Poniżej tego powinieneś ujrzeć linię "The latest stable version of Linux kernel is 2.4.20", z "2.4.20" (lub inną wersją w zależności od tego jaka będzie aktualnie) podświetlonym jako odnośnik. W tym przewodniku nie chcemy pobierać jądra z tego odnośnika ponieważ jest to tylko łąta. Jednakże, ta informacja jest dla nas istotna, bo mówi nam jakie jest najnowsze jądro, więc kiedy przejdziemy do właściwego miejsca pobierania, będziemy już wiedzieć o jaką wersję nam chodzi, zakładając, że chcemy najnowszą.

By przejść do miejsca gdzie można pobrać całe najnowsze jądro, kliknij na odnośniku HTTP w szarym prostokącie, w górnej części strony (<http://www.kernel.org/pub>).

Powinieneś się teraz znaleźć na stronie zatytułowanej "The Public Linux Archive - (As if there were a private one)". Pojawił się kolejny szary prostokąt w górnej części strony. Kliknij pierwszy odnośnik (<http://www.kernel.org/pub/linux>).

Powinieneś teraz być na stronie zatytułowanej "Index of /pub/linux". Na wyświetlonej liście katalogów kliknij odnośnik "kernel".

Teraz jesteśmy na stronie o tytule "Index of /pub/linux/kernel". Znowu mamy listę katalogów. Kliknij odnośnik *v2.4/*.

Powinieneś się znaleźć na stronie o tytule "Index of /pub/linux/kernel/v2.4". To tutaj znajdziemy poszukiwane jądro. Na liście widać multum plików. Upewnij się, że pobierasz plik o nazwie `linux-2.4.20.tar.gz`. Pewnie zauważyłeś, również pliki `bz2` i `bz2.sign`. Bz2 jest innym formatem kompresji, ale narzędzie do rozpakowywania takich plików nie jest domyślnie instalowane w systemie Debian (nic nie stoi na przeszkodzie by je zainstalować za pomocą `apt`). Narzędzie `tar` dla odmiany jest już zainstalowane więc postanowiłem, że dla potrzeb tego przewodnika pobierzemy plik `linux-2.4.20.tar.gz`. Pobierz plik i zapisz go do swojego domowego katalogu (`/home/<twojeimię>`).

Rozdział 3

Budowanie jądra krok po kroku, wymagania

3.1 Przygotowanie źródeł jądra 2.4.20

W poprzednim rozdziale pobraliśmy źródła jądra. Aktualnie są one zawarte w jednym skompresowanym pliku `linux-2.4.20.tar.gz` w katalogu domowym. W tym rozdziale przeniesiemy źródła do poprawnej lokalizacji i rozpakujemy je.

By skompilować nowe jądro, musimy je umieścić w katalogu `/usr/src`. Pewnie się zastanawiasz "Skoro chciałem mieć źródła jądra w katalogu `/usr/src` to dlaczego nie zapisałem ich tam od razu?". Po prostu nie miałeś uprawnień do zrobienia tego, ponieważ byłeś zalogowany jako zwykły użytkownik (przynajmniej mam nadzieję, że byłeś zalogowany jako zwykły użytkownik, a nie jako administrator (root)).

Nie polecam przeglądania stron WWW jako administrator. Twój komputer nie wybuchnie, ani nic w tym stylu, ale robienie wszystkiego jako administrator jest ryzykowne. Zatem, sugeruję pobieranie plików z internetu jako zwykły użytkownik i zapisywanie ich do swojego katalogu domowego. Gdy już to zrobisz, możesz wykorzystać polecenie `su` by zalogować się jako administrator i przenieść pliki w bardziej bezpieczne miejsce, jeśli zachodzi taka potrzeba.

Jeśli jesteś na tyle obeznany z Linuksem (nie bój się, jeśli nie jesteś to pomogę Ci za chwilę), zaloguj się jako administrator i przenieś plik `linux-2.4.20.tar.gz` ze swojego katalogu domowego do katalogu `/usr/src`.

Następnie rozpakuj plik wpisując `tar xvzf linux-2.4.20.tar.gz` w linii poleceń. To stworzy katalog o nazwie `linux-2.4.20` wewnątrz katalogu `/usr/src`.

Teraz stwórz dowiązanie symboliczne o nazwie `linux`, który będzie wskazywał na katalog `linux-2.4.20`, wpisując `ln -s linux-2.4.20 linux`. Jeśli takie dowiązanie istnieje, usuń je i stwórz ponownie, tak jak to opisałem powyżej.

Teraz możesz opuścić resztę tego rozdziału i udać się do rozdziału "Pliki potrzebne do konfiguracji i kompilacji jądra".

Jeśli powyższe cztery paragrafy wydają Ci się zagmatwane, to tutaj opisuję te same kroki, ale bardziej szczegółowo.

By przenieść plik `linux-2.4.20.tar.gz` do prawidłowej lokalizacji wykonaj te kroki:

3.1.1 Krok 1: Otwórz okno terminala

Jeśli korzystałeś z mojego dokumentu opisującego instalację, to powinieneś mieć uruchomiony Menedżer Okien WindowMaker. Jeśli tak jest, kliknij prawym przyciskiem myszy na pulpicie, następnie z menu wybierz XShells i XTerm. Powinno pojawić się okno terminala. Jeśli używasz innego Menedżera Okien, użyj jakiegokolwiek innej metody, której wymaga dany menedżer by uruchomić okno terminala.

3.1.2 Krok 2: Przejdź do katalogu do którego zapisałeś plik `linux-2.4.20.tar.gz`

Upewnij się, że jesteś w swoim `/home/<twojeimie>` katalogu (lub tam gdzie zapisałeś plik `linux-2.4.20.tar.gz`). By to zrobić wprowadź `pwd` w linii poleceń. `pwd` wyświetli w jakim jesteś katalogu. Jeśli `pwd` wyświetli coś innego niż katalog domowy, wprowadź w linii poleceń `cd /home/<twoje imie>`by przejść do swojego katalogu.

3.1.3 Krok 3: Zaloguj się jako administrator używając `su`

Jeśli nie jesteś zalogowany jako administrator, to teraz musisz to zrobić wprowadzając `su` w linii poleceń. Wprowadź hasło administratora gdy zostaniesz o to poproszony.

3.1.4 Krok 4: Przenieś plik `linux-2.4.20.tar.gz` do katalogu `/usr/src`

Wprowadź w linii poleceń `mv linux-2.4.20.tar.gz /usr/src`. To powinno przenieść plik `linux-2.4.20.tar.gz` z katalogu domowego do katalogu `/usr/src`. Jeśli otrzymałeś komunikat `Nie ma takiego pliku ani katalogu` to albo nie pobrałeś jądra z internetu, albo znajdujesz się w niewłaściwym katalogu. Przejdź do katalogu gdzie znajdują się źródła i spróbuj powtórzyć ten krok.

3.1.5 Krok 5: Sprawdź katalog `/usr/src`

Przejdź do katalogu `/usr/src` wprowadzając `cd /usr/src` w linii poleceń. Następnie wprowadź polecenie `ls -l`. Jak pewnie wiesz polecenie `ls` pokazuje listę wszystkich plików w aktualnym katalogu. Opcja `-l` dodana po `ls` pokazuje więcej informacji niż samo `ls`. Pokazuje informacje w długim (long) formacie; stąd `-l`. Powinieneś zobaczyć plik `linux-2.4.20.tar.gz`, który właśnie skopiowaliśmy do tego katalogu. To powinien być jedyny plik w tym katalogu. Jeśli tak jest, opuść resztę tego podrozdziału i przejdź do kroku 6.

Możliwe, że w katalogu będą oprócz `linux-2.4.20.tar.gz` również inne pliki. Jeśli posiadasz źródła starszego jądra, to będą one właśnie w tym katalogu. Możesz również mieć dowiązanie symboliczne wskazujące na te stare źródła.

Zanim przejdę dalej, pozwól mi wyjaśnić czym jest dowiązanie symboliczne. Dowiązanie symboliczne to dowiązanie lub nazwa zastępcza wskazująca na inny plik lub katalog. Jeśli jesteś obeznany z systemem Windows, to dowiązanie symboliczne jest czymś podobnym do skrótu. Stworzymy dowiązanie symboliczne w Kroku 6, więc jeśli jeszcze nie wszystko jest jasne, miejmy nadzieję, że tak się stanie gdy stworzysz jakieś dowiązanie.

Teraz spójrz na inne pliki z katalogu `/usr/src`. Jest jakiś o nazwie "linux"? Jeśli nie, to możesz opuścić resztę tego podrozdziału i przejść do Kroku 6. Jeśli jest jakiś element o nazwie "linux", sprawdź czy jest to dowiązanie symboliczne czy katalog. Możesz to ustalić używając polecenia `ls -l` i oglądając jego wynik. Wprowadź `ls -l` w linii poleceń. Wynik powinien wyglądać mniej więcej jak to (tylko lepiej sformatowany):

```
lrwxrwxrwx    1 root      src                12 Oct 31 13:03 linux -> linux-2.4.18
drwxr-xr-x   14 573      573               4096 Aug  2 18:39 linux-2.4.18
-rw-----    1 clinton  clinton          32219641 Oct 30 14:40 linux-2.4.20.tar.gz
```

Użyj pierwszego pliku z listy jako przykład służący do wyjaśnienia gdzie należy szukać informacji o tym czy plik jest dowiązaniem symbolicznym. Pierwsza część "lrwxrwxrwx" pokazuje jaki jest typ pliku oraz jego uprawnienia.

Pierwszy znak "l" oznacza, że plik jest dowiązaniem symbolicznym. Katalogi, dla odmiany w miejscu "l" mają literę "d". Zwykle pliki oznaczane są znakiem "-" na pierwszej pozycji (są jeszcze inne znaki o różnych znaczeniach, ale to wykracza poza temat tego artykułu). Reszta znaków odpowiada uprawnieniom właściciela, grupy oraz reszty użytkowników. Na razie to zignoruj.

Zamierzam również opuścić środkową część listingu katalogów i przejść od razu na koniec, gdzie jest nazwa pliku. Jeśli spojrzysz na nazwy trzech plików wylistowanych powyżej, powinieneś zauważyć, że pierwsza nazwa od góry zawiera "->" poprzedzoną nazwą `linux`, po tym następuje `linux-2.4.18`. To oznacza, że plik `linux` jest dowiązaniem symbolicznym wskazującym (stąd ta ładna strzałka ["->"]) na `linux-2.4.18`. Jeśli spojrzysz na `linux-2.4.18`, zobaczysz, że jest to katalog (zauważ, że pierwszy znak w linii z `linux-2.4.18` jest literą "d" oznaczającą katalog).

Teraz kiedy ustaliliśmy, że mamy plik `linux` w katalogu `/usr/src`, zrób jedno z poniższych:

Jeśli `linux` jest dowiązaniem symbolicznym, usuń go wprowadzając `rm linux` w linii poleceń.

Jeśli `linux` jest katalogiem, zmień mu nazwę wprowadzając `mv linux linux.stary` w linii poleceń.

Powinieneś być gotowy do kontynuacji.

3.1.6 Krok 6: Rozpakuj plik `linux-2.4.20.tar.gz`

Teraz musimy rozpakować plik `linux-2.4.20.tar.gz` tak, byśmy mieli dostęp do źródeł jądra. By to zrobić wprowadź `tar xvzf linux-2.4.20.tar.gz` w linii poleceń. Chwilę potrwa zanim plik zostanie rozpakowany. Kiedy skończy powinieneś znowu zobaczyć znak zachęty (“#”).

3.1.7 Krok 7: Utwórz dowiązanie symboliczne o nazwie `linux` wskazujące na `linux-2.4.20`

Teraz możemy stworzyć dowiązanie symboliczne `linux`, które będzie wskazywać na nasze nowe źródła jądra zlokalizowane w katalogu `linux-2.4.20`. Wprowadź `ln -s linux-2.4.20 linux` by utworzyć dowiązanie symboliczne. Jeśli teraz wprowadzisz `ls -l` w linii poleceń, powinieneś zobaczyć dowiązanie symboliczne o nazwie `linux` wskazujące na katalog `linux-2.4.20` (lub jakikolwiek inny w zależności jakie jądro pobrałeś).

3.2 Pliki potrzebne do konfiguracji i kompilacji jądra

By skonfigurować i skompilować jądro, skorzystamy z `apt`'a (lub `Synaptic`'a jeśli preferujesz), w celu pobrania kilku rzeczy.

Będąc zalogowanym jako administrator wprowadź poniższe polecenie, by pobrać potrzebne nam rzeczy (lub wybierz każdy element z listy korzystając z `Synaptic`'a i zainstaluj je):

```
apt-get install tk8.2 make gcc bin86 libc6-dev kernel-package
```

W tym przewodniku będę używał graficznego narzędzia do konfiguracji jądra w systemie X-Window. Jeśli wolisz używać tekstowego powinieneś w powyższym poleceniu podmienić “`tk8.2`” na “`libncurses5-dev`” (lub możesz zainstalować oba pakiety i korzystać z obu rodzajów konfiguracji; w zależności od nastroju).

Gdy wszystko zostanie pobrane i zainstalowane, pojawi się ekran zatytułowany “Konfigurowanie `Binutils`”. Jest to ostrzeżenie mówiące, że ta wersja `binutils` nie będzie działała z niektórymi jądrami. Ponieważ używam najnowszego stabilnego jądra, nie będziemy mieć żadnych problemów. Po prostu naciśnij `Enter` by kontynuować.

Rozdział 4

Konfiguracja jądra

4.1 Uruchamianie narzędzia do konfiguracji jądra

Zakładając, że jesteś nadal w katalogu `/usr/src` i wykonałeś wszystkie poprzednie instrukcje, wprowadź `cd linux` by przenieść się do katalogu `linux` (który jeśli sobie przypominasz jest dowiązaniem symbolicznym do katalogu `linux-2.4.20`).

Będąc w katalogu `/usr/src/linux` jesteś gotowy do uruchomienia narzędzia konfiguracyjnego. Wprowadź `make xconfig` w linii poleceń by uruchomić narzędzie do Konfiguracji Jądra Linuksa (Linux Kernel Configuration).

Jeśli zamiast graficznej wolisz używać wersji tekstowej tego narzędzia, musisz wprowadzić `make menuconfig` w linii poleceń. Nie zamierzam opisywać wersji tekstowej w tym przewodniku, ale myślę, że powinieneś sobie poradzić, ponieważ oba narzędzia są do siebie w pewnym stopniu podobne.

Twoim oczom powinien ukazać się szary ekran z masą przycisków. Jeśli otrzymałeś jakieś błędy w linii poleceń i narzędzie się nie uruchomiło, otwórz inny terminal i wprowadź `xhost +` będąc zwykłym użytkownikiem, by pozwolić innym użytkownikom (w tym wypadku administratorowi) uruchamiać graficzne aplikacje w Twoim środowisku X. Kiedy to zrobisz, wprowadź ponownie `make xconfig` jako administrator, by uruchomić narzędzie do Konfiguracji Jądra Linuksa.

Jeśli narzędzie do Konfiguracji Jądra Linuksa nadal się nie uruchamia i wciąż otrzymujesz komunikaty o błędach, upewnij się, że zainstalowałeś wszystkie wcześniej wspomniane pakiety. Takie jak:

`tk8.2`, `make`, `gcc`, `bin86`, `libc6-dev`, `kernel-package`, `libncurses5-dev` (potrzebne tylko gdy chcesz korzystać z tekstowej wersji narzędzia konfiguracyjnego)

Jeśli jesteś w stanie uruchomić narzędzie Konfiguracji Jądra Linuksa, możemy kontynuować.

4.2 Konfiguracja jądra Linuksa

Powinieneś mieć teraz otwarte narzędzie do Konfiguracji Jądra Linuksa. Zanim zaczniemy, polecam przesunięcie okna z narzędziem na samą górę ekranu. Będziemy otwierać mnóstwo podokien podczas procesu konfiguracji, a miejsce pokazywania tych podokien jest związane z oknem głównym (tym z wieloma przyciskami). Jeśli okno główne umieścisz na dole ekranu, to podokna w większości pojawiać się będą poza ekranem i będzie trzeba je przesuwać by zobaczyć je w całości. Możesz zaoszczędzić sobie mnóstwo klikania i przenoszenia ustawiając główne okno w górnej części ekranu.

Konfiguracja jądra brzmi strasznie, ale w rzeczywistości sprowadza się do odpowiadania na masę pytań wielokrotnego wyboru. W większości narzędzie Konfiguracji Jądra Linuksa będzie pytać czy dodać obsługę czegoś, a Ty będziesz odpowiadać 'Y' (Yes) by potwierdzić, 'N' (No) by nie dodawać obsługi lub 'M' (Module) by dodać obsługę tej rzeczy jako moduł.

Korzyści płynące z ładowania różnych rzeczy jako moduły: mniejsze jądro (i prawdopodobnie szybsze) oraz możliwość załadowania lub wyładowania modułu w każdej chwili bez rekompilacji jądra. Jednakże, za pierwszym razem polecam wybieranie 'Y' dla każdej rzeczy, której obsługi chcesz. Możesz później poczytać jak używać modułów, gdy już będziesz zorientowany w kompilacji jądra.

Nie mam możliwości opisanie każdej opcji konfiguracyjnej jądra, ponieważ każdy posiada inny sprzęt. Narzędzie Konfiguracji Jądra Linuksa posiada system pomocy. Praktycznie każda sekcja wielokrotnego wyboru posiada przycisk pomocy (help), gdzie można uzyskać dużo szczegółowych informacji. Stanowczo polecam czytanie tych opisów dla każdej opcji. Pomoże Ci to dokonać właściwych wyborów dla swojego sprzętu oraz zapoznać się z usługami, które dostarcza jądro.

[OD TŁUMACZA] [W związku z tym, że konfigurując jądro opcje, które zobaczysz będą po angielsku, zdecydowałem się zostawiać je w oryginale, a tłumaczenie umieszczać obok w nawiasie.]

Code maturity level options (Opcje dotyczące dojrzałości kodu):

By zobaczyć przykład działania narzędzia Konfiguracji Jądra Linuksa, włączając obsługę pomocy, kliknij na przycisku "Code maturity level options", który znajduje się w lewym górnym rogu głównego okna. Zauważysz, że pojawi się nowe okno dialogowe. W przypadku okna dotyczącego tej opcji jest tylko jedno pytanie wymagające odpowiedzi "Prompt for development and/or incomplete code/drivers" ("Pytaj o rozwojowe i/lub nieskończone części kodu/sterowniki")

Zanim zdecydujemy o tym czy chcemy mieć obsługę tego elementu w jądrze, zerknijmy na okienko pomocy. By to zrobić kliknij na przycisku "Help" ("Pomoc") w prawej części okna.

Po przeczytaniu całego tekstu wróć do ostatniego paragrafu i spójrz na niego jeszcze raz. Brzmi on tak "Unless you intend to help test and develop a feature or driver that falls into this category, or you have a situation that requires using these features, you should probably say N here." ("Jeżeli nie chcesz pomóc w testowaniu i rozwijaniu funkcji lub sterowników z tej kategorii, lub nie jesteś w sytuacji, która zmuszałaby Cię do używania tych funkcji, powinieneś prawdopodobnie odpowiedzieć 'N'"). Wiele z tekstów, dostępnych w pomocy narzędzia Konfiguracji Jądra Linuksa, wyjaśnia czego dana opcja dotyczy i sugeruje Ci odpowiedź. Myślę,

że jest to bardzo pomocne dla niezorientowanych w kompilacji jądra. Jeśli nie jesteś pewien jakiejś odpowiedzi, polecam postępowanie według sugestii, których dostarcza tekst pomocy. Jeśli nie ma żadnych sugestii to domyślne ustawienie jest zazwyczaj bezpieczne.

Na wszelki wypadek gdyby ktoś pytał, włączenie opcji "Prompt for development and/or incomplete code/drivers" uaktywnia sekcje uważane za eksperymentalne lub trochę niestabilne. Przykładem może być obsługa Firewire. Jeśli spojrzysz na główne okno narzędzia Konfiguracji Jądra Linuksa to zauważysz, że przycisk z obsługą Firewire jest nieaktywny (nie możesz go kliknąć). Jeśli odpowiedziałbyś 'Y' w pytaniu "Prompt for development and/or incomplete code/drivers", zobaczyłbyś, że przycisk Firewire byłby aktywny. Nie polecam włączania tego teraz. Wstrzymaj się do czasu aż będziesz bardziej zorientowany w konfiguracji jądra.

Teraz gdy już wiemy czym jest "Prompt for development and/or incomplete code/drivers", możemy wybrać 'N' (to domyślne ustawienie) z lewej strony tekstu. Powinieneś również zauważyć, że do wyboru jest tylko opcja Y (Tak) i N (Nie). M (Moduł) brakuje. Jeśli jakkolwiek z możliwości (Y, N lub M) jest niedostępna dla specyficznej opcji jądra, to nie będzie wyświetlona.

Ponieważ nie ma więcej ustawień na tym oknie, kliknij przycisk "Next" ("Następny") znajdujący się na dole tego okna. Mógłbyś również kliknąć przycisk "Main Menu" ("Menu Główne"), a następnie wybrać kolejny przycisk poniżej przycisku "Code maturity level options", którym w tym wypadku jest "Loadable module support" ("Obsługa modułów ładowalnych"), ale myślę, że prościej jest po prostu kliknąć przycisk "Next" po skończeniu ustawiania opcji w każdym okienku i przejście całego procesu konfiguracji w sposób liniowy.

Loadable Module Support (Obsługa Modułów Ładowalnych):

W oknie "Loadable module support" zauważysz trzy opcje do ustawienia. Przeczytaj opisy wszystkich, a potem wybierz 'Y' dla każdej z nich. Gdy skończysz kliknij przycisk "Next".

Processor Type and Features (Typ i Funkcje Procesora):

W tym miejscu konfiguracja jądra staje się zależna od sprzętu, który posiadasz. Zająłem się tym oknem tylko dlatego, żeby pokazać Ci rozwijalne listy wyboru, które również istnieją w narzędziu Konfiguracji Jądra Linuksa oraz by pomóc w ustawieniu kilku opcji.

Pierwszym elementem w tym oknie jest opcja "Processor family" ("Rodzina procesorów"). Zauważysz, że zamiast standardowego wyboru Y - N - M, takiego jak widziałeś wcześniej, pojawił się przycisk (Zakładam, że domyślnym ustawieniem i jednocześnie wyświetlonym jest "Pentium-III/Celeron (Coppermine)". Kliknij ten przycisk i wybierz swój procesor z listy. Jeśli nie wiesz jaki rodzaj procesora posiadasz domyślne ustawienie zadziała, ale jądro nie będzie zoptymalizowane pod Twoją maszynę. Na przykład mój komputer wyposażony jest w procesor AMD Athlon, więc ja wybiorę z listy linię "Athlon/Duron/K7". Jeśli nie wiesz jaki masz procesor to bezpiecznym, ale i niezoptymalizowanym ustawieniem jest 386.

Przeczytaj okna pomocy dla każdej pozostałej opcji i wybierz odpowiednio Y (Tak), N (Nie) lub M (Moduł). Gdy skończysz wciśnij przycisk "Next". Zauważ również, że niektóre opcje nie są ani wielokrotnego wyboru, ani listami rozwijalnymi, a przyciskami, które otwierają kolejne okna. Upewnij się, że na wszystkich kliknąłeś. Klikając na każdym oknie przycisk "Next" masz zagwarantowane, że zobaczysz wszystkie możliwe aspekty jądra. Jeśli robisz inaczej, możesz coś przeoczyć.

Za minutę pomogę skonfigurować nagrywarke CD-R/RW, system plików z księgowaniem oraz kartę dźwiękową, a w międzyczasie kontynuuj procedurę czytania okien pomocy dla każdej opcji, dokonywania odpowiednich wyborów (Y - N - M) oraz klikania "Next" na dole każdego okna, aż przejdziesz przez całą konfigurację jądra. To zajmie trochę czasu, ale jest tego warte.

Chciałbym móc zaoferować więcej pomocy przy tym procesie, ale nie znam Twojego sprzętu, ani co tak naprawdę chcesz osiągnąć. Jednakże zostawiam Cię w objęciach kompetentnych okien z pomocą.

Chciałbym przestrzec przed pewną regułą, przynajmniej do czasu, aż będziesz bardziej zorientowany w sprawach dotyczących jądra Linuksa. Często ludzie myślą tak "Po prostu zainstaluję obsługę wszystkiego". To bardzo zły pomysł. Wkompiluj obsługę tylko tych rzeczy, które chcesz lub potrzebujesz. Jeśli nie wiesz do czego jest dana opcja po prostu zaakceptuj domyślne ustawienie. Jeśli masz pytania, w poszukiwaniu pomocy, użyj dobrej wyszukiwarki internetowej, takiej jak Google. Możesz również napisać do mnie mail i postaram się pomóc w tych kwestiach w których będę mógł.

Zanim będziemy kontynuować przejdź przez konfigurację jądra jeszcze raz.

Rozdział 5

Konfiguracja jądra do obsługi dźwięku, ext3 i nagrywarek

5.1 Konfiguracja jądra do obsługi dźwięku

Teraz kiedy przeszedłeś przez konfigurację jądra, wrócimy i skonfigurujemy obsługę dźwięku, systemów plików z księgowaniem i nagrywarek. Zaczniemy od dźwięku.

Kliknij przycisk "Sound" ("Dźwięk"), czyli trzeci od góry w trzeciej kolumnie (może mieć inną lokalizację jeśli używasz innej wersji jądra, jednakże powinien nadal nazywać się "Sound"). Powinno pojawić się okno "Sound".

Pierwszy element tego okna to "Sound Card Support" ("Obsługa Kart Dźwiękowych"). Musisz odpowiedzieć na to pytanie 'Y'. Następnie musisz wybrać odpowiednią kartę dźwiękową z listy. Zakładając, że masz kartę Sound Blaster Live! Value, powinieneś odpowiedzieć twierdząco na opcję "Creative SBLive! (EMU10K1)".

Inną równie popularną kartą jest "Creative Ensoniq AudioPCI 97 (ES1371)". Jeśli nie jesteś pewien jaką kartę posiadasz lub Twoja karta jest wbudowana w płytę główną, spróbuj ustawić tę opcję na 'Y'. Często Twoja karta powinna działać z tym ustawieniem.

Jeśli wiesz jaką kartę posiadasz, ustaw odpowiednią opcję na 'Y' i pozostałe na 'N' (oczywiście z wyjątkiem opcji "Sound Card Support"). Kiedy skończysz kliknij przycisk "Main Menu".

5.2 Konfiguracja Debiana do obsługi nagrywarek

Teraz skonfigurujemy Twoją maszynę do obsługi nagrywarki. Linux obsługuje tylko nagrywarki SCSI, więc jeśli taką posiadasz to już powinieneś móc z niej korzystać. Jednakże większość ludzi posiada nagrywarki IDE, a one standardowo nie są obsługiwane przez system Linux.

Na szczęście jądro systemu Linux może zostać skonfigurowane do obsługi nagrywarki IDE

poprzez emulację SCSI. By to ustawić musimy przejść przez kilka różnych lokalizacji w narzędziu do Konfiguracji Jądra Linuksa.

Pierwszym miejscem będzie "ATA/IDE/MFM/RLL Support" ("Obsługa ATA/IDE/MFM/RLL"). Ta sekcja wyświetli nam okno z dwoma elementami. Drugi element to "IDE, ATA and ATAPI Block Devices" ("Urządzenia blokowe IDE, ATA i ATAPI"). Kliknij na tym by uruchomić kolejne okno.

Powinieneś mieć teraz przed sobą okno "IDE, ATA and ATAPI Block Devices". Jest kilka różnych metod by skonfigurować Twój komputer do obsługi nagrywarek IDE jako SCSI, my jednakże skorzystamy z najprostszej, która jest opisana poniżej.

Po pierwsze musisz wyłączyć obsługę napędów CD IDE. By to zrobić ustaw opcję "Include IDE/ATAPI CDROM Support" ("Dołącz Obsługę IDE/ATAPI CDROM") na 'N'.

Następnie musisz włączyć emulację SCSI ustawiając opcję "SCSI Emulation" ("Emulacja SCSI") na 'Y'.

UWAGA: Niektórzy mogą stwierdzić, że stosują niestandardowe metody. Wiem, że to może nie jest najlepsza metoda dodawania obsługi nagrywarek, ale działa i jest prosta (ten artykuł jest skierowany do początkujących). Gdy już będziesz obeznany z konfiguracją jądra, nie krępuj się i dodaj z powrotem obsługę CDROM'ów IDE oraz sprecyzuj, które napędy powinny być obsługiwane jak SCSI za pomocą parametru `append="hdx=ide-scsi"` w `lilo.conf`. Nie zamierzam opisywać takiej metody w tym przewodniku.

Kliknij przycisk 'OK' by zamknąć okno, następnie kliknij przycisk "Main Menu". Powinieneś znaleźć się z powrotem w menu głównym.

Teraz kliknij przycisk "SCSI Support" ("Obsługa SCSI") znajdujący się w menu głównym.

W oknie "SCSI Support" ustaw następujące opcje na 'Y':

- SCSI Support (Obsługa SCSI)
- SCSI Disk Support (Obsługa Dysków SCSI)
- SCSI CD-ROM Support (Obsługa CDROM'ów SCSI)
- SCSI Generic Support (Ogólna Obsługa SCSI)
- Enable Extra Checks in New Queuing Code (Aktywacja Dodatkowego Sprawdzania w Nowym Kodzie Kolejującym)

Wszystkie inne opcje ustaw na 'N'.

Zauważ: Nowi użytkownicy zazwyczaj nie wiedzą nic na temat SCSI i zwykle nie potrzebują wiedzieć nic więcej korzystając tylko z emulacji SCSI opisanej tutaj. Jeśli chcesz wkompiłować obsługę logowania błędów SCSI (dla zabawy lub gdy rzeczywiście masz problemy), uaktywnij następujące opcje:

- Verbose SCSI Error Reporting (Szczegółowe Raportowanie Błędów SCSI)

- SCSI Logging Facility (Możliwość Logowania SCSI)

To wszystko. Kliknij przycisk "Main Menu" by zamknąć okno SCSI.

5.3 Włączanie obsługi systemu plików Ext3 (Chcesz to zrobić!)

Teraz włączymy obsługę systemu plików EXT3. Jądro Linuksa obsługuje również inne systemy plików z księgowaniem, takie jak JFS czy ReiserFS, ale w tym artykule zamierzam skupić się tylko na EXT3. EXT3 jest bardzo stabilny i jeśli coś pójdzie nie tak, masz już narzędzia by dokonać poprawek w systemie plików, ponieważ jest kompatybilny ze standardowym systemem plików EXT2.

By włączyć obsługę EXT3, będąc w głównym menu okna konfiguracji, kliknij przycisk "File Systems" ("Systemy Plików"). Pojawi się okno z systemami plików. Przewiń okno w dół aż do przycisku "EXT3 Journaling File System Support" ("Obsługa Systemu Plików z Księgowaniem EXT3") i ustaw go na 'Y'. Wszystkie inne opcje w tym oknie zostaw z ustawieniami domyślnymi (chyba, że są inne systemy plików, których chcesz użyć, takie jak vfat, który pozwala montować partycje FAT32 z Windowsa).

Kiedy ustawisz opcję "EXT3 Journaling File System Support" na 'Y', poniżej uaktywni się opcja dotycząca debugowania. Zostaw ją z ustawieniem domyślnym 'N'.

Obsługa dźwięku oraz SCSI jest teraz skonfigurowana (z wyjątkiem zmiany urządzenia /dev/cdrom by wskazywało w odpowiednie miejsce, ale tym zajmiemy się później), ale nie skończyliśmy jeszcze z obsługą EXT3. Jednakże, zanim zajmiemy się tymi zadaniami, musimy najpierw skompilować jądro.

Kliknij przycisk "Main Menu" by zamknąć okno.

Rozdział 6

Kompilacja, instalacja jądra

6.1 Zapisywanie konfiguracji jądra i kompilacja

Teraz jesteśmy gotowi do kompilacji jądra. Jednakże zanim to zrobimy, chciałbym wspomnieć o jednej rzeczy. Jeśli masz kilka komputerów na których chcesz zainstalować to samo jądro (to nie za bardzo zadziała jeśli komputery nie są takie same), kliknij przycisk "Store Configuration to File" ("Zachowaj Ustawienia w Pliku"). Dzięki temu wszelkie zmiany, których dokonałeś zostaną zapisane w pliku. Możesz potem przekopiować ten plik na inny komputer, i gdy otworzysz narzędzie Konfiguracji Jądra Linuksa na tym innym komputerze, kliknij przycisk "Load Configuration from File" ("Wczytaj Ustawienia z Pliku") i otwórz plik z zapisanymi ustawieniami. Dzięki temu nie będziesz musiał konfigurować jądra na każdym komputerze.

Dobra, skompilujmy jądro. W głównym oknie narzędzia Konfiguracji Jądra Linuksa kliknij przycisk "Save and Exit" ("Zapisz i Wyjdź"). Powinno pojawić się okno z komunikatem byś sprawdził plik makefile oraz uruchomił make dep. Nie musisz robić żadnej z tych rzeczy, kliknij po prostu przycisk "OK". Powinieneś znowu mieć przed sobą linię poleceń.

Wprowadź pwd w linii poleceń by zobaczyć w jakim katalogu się znajdujesz. Polecenie powinno zwrócić /usr/src/linux (jeśli zwróciło coś innego, wprowadź cd /usr/src/linux by przejść do katalogu zawierającego źródła jądra).

Gdy już jesteś w tym katalogu, wprowadź te dwa polecenia:

- `make-kpkg clean`
- `make-kpkg --revision=786:MojeJadro2.4.20 kernel_image`

Drugie polecenie jest odpowiedzialne za właściwą kompilację. Na moim komputerze wyposażonym w procesor AMD Athlon 1800+ proces kompilacji trwa około 10 minut. Na jednym z moich starszych komputerów trwało to około godziny (na komputerze klasy 486, pewnie poczekaś kilka dni). Jeśli masz starszy komputer, zjedz obiad i obejrzyj dobrą komedię. Jak wrócisz kompilacja powinna być zakończona. Tak czy siak zrób sobie przerwę.

Jeśli bardzo chciałbyś zrozumieć wszystkie szczegóły dotyczące tej kompilacji, możesz przeczytać dokumentację znajdującą się w `/usr/doc/kernel-package`. W skrócie, tworzymy plik `.deb`, który później zainstalujemy dokładnie w taki sam sposób jak każdy inny program; używając narzędzia `dpkg`. Tak wygenerowany plik wykona za nas automatycznie wszelkie zadania związane z konfiguracją systemu, które użytkownicy dystrybucji innych niż Debian muszą robić sami. Generalnie to dobra rzecz.

6.2 Instalacja nowego jądra

Po obiedzie i obejrzeniu filmu, komputer powinien już skończyć kompilację, a Twoim oczom znowu powinna ukazać się linia poleceń. W katalogu `/usr/src` powinien pojawić się nowy plik.

Przejdźmy tam wprowadzając `cd /usr/src` w linii poleceń. Wpisując `ls -l`, zakładając, że nie było żadnych błędów podczas kompilacji, powinieneś ujrzeć plik o nazwie `kernel-image-2.4.20_MojeJadro2.4.20` lub coś w tym stylu.

Teraz musimy zainstalować nowe jądro, dokładnie tak samo jak każdy inny plik `.deb`, korzystając z narzędzia `dpkg`. Wprowadź `dpkg -i kernel-image-2.4.20_MojeJadro2.4.20.deb` (lub inny, jeśli Twój plik nazywa się inaczej) w linii poleceń i wciśnij `Enter`.

Gdy instalacja zostanie zakończona, zostaniesz poproszony o utworzenie dyskietki startowej. Sugeruję to zrobić, ale nie jest to niezbędne.

Następnie zostaniesz zapytany, czy zainstalować rekord rozruchowy korzystając z konfiguracji `/etc/lilo.conf`. Jeśli nie odpowiesz twierdząco na to pytanie, nie będziesz mógł uruchomić komputera korzystając z nowego jądra (chyba, że stworzyłeś dyskietkę, o której była mowa przed chwilą, ale wtedy będziesz mógł uruchomić komputer tylko z tej dyskietki). Upewnij się, że odpowiedziałeś twierdząco na jedno z pytań (lub oba), inaczej pozostaniesz bez możliwości uruchomienia swojego nowego lśniącego jądra w wersji 2.4.20.

Gdy zakończysz ten proces, musisz zrestartować swój komputer. Po restarcie wprowadź `dmesg` i oglądaj jego wyniki. Poszukaj jakichś błędów. Powinieneś również zauważyć (jeśli potrafisz odczytać tajemnicze komunikaty Linuksa), że Twoja karta dźwiękowa została rozpoznana. Jeśli są jakieś problemy, na przykład wybrałeś nieodpowiednią kartę dźwiękową, wtedy musisz zmienić konfigurację jądra i spróbować ponownie. Jeśli tak się zdarzy, spójrz do ostatniego rozdziału tego artykułu.

Rozdział 7

Ext3, rozwiązywanie problemów, zakończenie

7.1 Ustawianie komputera by używał systemu plików z księgowaniem Ext3

Pamiętasz jak mówiłem, że nie skończyliśmy jeszcze konfigurować EXT3? Teraz nadszedł czas by skończyć. Zanim zaczniemy, będziesz potrzebował jakiegoś edytora tekstowego, takiego jak vim, nano, pico czy jakikolwiek graficzny. Wybierz jakiś i wtedy będziemy kontynuować.

Po pierwsze, musisz wiedzieć jakie partycje posiadasz w swoim systemie. Jeśli nie pamiętasz, możesz użyć polecenia "df", które powinno wylistować wszystkie zamontowane partycje. Oto przykład jak taka lista może wyglądać:

Filesystem		zamont. na
/dev/hda1	. . .	/
/dev/hda3	. . .	/usr
/dev/hda5	. . .	/var
/dev/hda6	. . .	/home

Będziesz potrzebował tych informacji do następnego polecenia. Możesz wprowadzić:

```
/sbin/tune2fs -j /dev/hdxx (gdzie x jest napędem i numerem partycji, na przykład /dev/hda1 lub /dev/hdb3) w linii poleceń dla każdej posiadanej partycji albo możesz skorzystać z poniższego skryptu (nie wpisuj '#' ani '>', to są znaki zachęty. Pamiętaj również by nacisnąć <Enter> po każdej linii i by urządzenia były oddzielone spacjami):
```

```
# for each in /dev/hda1 /dev/hda3 /dev/hda5 /dev/hda6
> do
> /sbin/tune2fs -j $each
> done
```

Jeśli otrzymasz błąd, albo wpisałeś coś niewłaściwie, albo umieściłeś nieodpowiednie urządzenie na liście; dokonaj odpowiednich poprawek i wprowadź polecenie jeszcze raz.

To powinno przekonwertować wszystkie partycje z ext2 na ext3. Kiedy to zrobisz, musisz powiedzieć Linuksowi by zamontował wszystkie partycje jako ext3. To miejsce w którym potrzebujemy edytora tekstowego.

Będąc zalogowanym jako administrator. Otwórz plik `/etc/fstab`. Gdziekolwiek znajdziesz wyraz "ext2" zmień go na "ext3". Gdy skończysz, zapisz plik i wyjdź z edytora.

Musimy zrobić jeszcze jedną rzecz przed restartem (możesz ręcznie odmontować i zamontować wszystkie partycje jako ext3, ale prawdopodobnie dużo prościej będzie po prostu zrestartować system i Linux automatycznie zamontuje wszystko używając ext3).

Wspominałem o tym przedtem, musimy naprawić nasze urządzenie `/dev/cdrom`. W tym momencie wskazuje na coś w stylu `/dev/hdb`, a odkąd zrobiliśmy emulację SCSI, nasz CDROM już tam nie istnieje. Musimy naprawić dowiązanie `/dev/cdrom`.

By tego dokonać, musimy najpierw dowiedzieć się na co mamy zmienić to dowiązanie. Prawdopodobnie będzie to `/dev/scd0`, ale musimy się upewnić. Włóż do napędu płytę z danymi i wprowadź:

```
mount /dev/scd0 /cdrom
```

Jeśli wszystko dobrze poszło powinieneś móc wprowadzić `ls /cdrom` w linii poleceń i otrzymać listę katalogów na płycie. Jeśli nie, a zwłaszcza gdy masz więcej niż jedno urządzenie CD w swoim komputerze, spróbuj z innym urządzeniem, takim jak `/dev/scd1`, `/dev/scd2`, itd.

Gdy już zdecydujesz, który napęd będzie domyślnym napędem CD, ustaw by dowiązanie symboliczne `/dev/cdrom` wskazywało na niego (prawdopodobnie pamiętasz dowiązania symboliczne z naszej dyskusji o tym gdzie zapisać źródła jądra. `/dev/cdrom` jest po prostu dowiązaniem symbolicznym do urządzenia w Twoim komputerze). By zademonstrować jak to zrobić, posłużę się swoim systemem, na którym domyślny napęd CD jest na `/dev/scd0`. By zmienić dowiązanie symboliczne `/dev/cdrom` tak by wskazywało na właściwe urządzenie, wprowadź poniższe polecenia:

- `rm /dev/cdrom` (to usunie aktualne, niewłaściwe dowiązanie symboliczne `/dev/cdrom`)
- `ln -s /dev/scd0 /dev/cdrom` (to stworzy nowe dowiązanie symboliczne `/dev/cdrom` wskazujące na prawidłowe urządzenie)

To powinno wystarczyć. By się upewnić, w linii poleceń wprowadź `umount /cdrom` by odmontować CD, który montowałeś kilka minut temu. Następnie, wprowadź `mount /cdrom`. Teraz powinieneś móc wyświetlić listę katalogów na płycie za pomocą polecenia `ls /cdrom`. Jeśli to zadziała, to znaczy, że wykonałeś wszystko poprawnie. Jeśli nie, wróć i sprawdź swoje ustawienia jeszcze raz. Sprawdź również za pomocą polecenia `cat /etc/fstab` czy na liście znajduje się linia zaczynająca się od `/dev/cdrom /cdrom`. Mimo, że jest raczej pewne, że aktualnie się tam znajduje, to jednak warto to sprawdzić gdy coś nie działa. Jeśli taki wpis nie istnieje, dodaj następującą linię w pliku `fstab`:

```
/dev/scd0          /cdrom          cd9660  ro,noauto      0          0
```

Jeśli wszystko działa prawidłowo, nadszedł czas restartu. Jeśli podczas konfiguracji umieściłeś płytę uruchomieniową w napędzie, usuń ją zanim będziesz restartował komputer.

7.2 Co jeśli coś pójdzie nie tak?

Zakładając, że jest to prawdopodobnie pierwszy raz kiedy kompilujesz jądro, możliwe, że popełniłeś jakiś błąd (na przykład wybrałeś złą kartę dźwiękową w jądrze).

Jest kilka rzeczy, które musisz zrobić przed ponowną próbą kompilacji jądra. Po pierwsze musisz usunąć lub zmienić nazwę pliku `.deb` z katalogu `/usr/src` (lub wybierz inną nazwę podczas kroku `make kpkg` wymienionego wcześniej w artykule).

Następnie, jeśli próbujesz rekompilować tą samą wersję co przedtem (na przykład 2.4.20), musisz zmienić nazwy lub usunąć kilka rzeczy. Miłą rzeczą jest to, że podczas `dpkg -i <plik z jądrem.deb>` otrzymasz informacje mówiące co dokładnie powinieneś usunąć lub czego zmienić nazwę. Gdy zobaczysz te informacje, uruchom nowe okno `xterm` i usuń lub zmień nazwę przeszkadzającego katalogu, wciśnij `CTRL+C` by przerwać proces `dpkg -i` w pierwszym oknie, a następnie uruchom go ponownie. Wszystko powinno przejść bez problemu.

Jest jeszcze coś na co chciałbym zwrócić uwagę w tym artykule. Czytaj wszystko, zarówno ten dokument jak i ekrany z pomocą. Nie zgaduj, bo inaczej masz prawie gwarantowane błędy.

7.3 Zakończenie

To koniec. Mam nadzieję, że jesteś trochę bardziej obeznany z pobieraniem, konfiguracją i kompilacją jądra Linuksa pod Debianem. Mam również nadzieję, że nauka Debiana była przyjemna.