

Moduły ładowalne w Linux

W jądrze Linuks sterowniki do urządzeń mogą zarówno być:

- wbudowane w samo jądro (wskompilowane moduły)
- w osobnym pliku i wtedy są one ładowane do pamięci komputera i uruchamiane w razie potrzeby, lub przy starcie systemu (ładowalne moduły).

Zalety modułowej budowy

Kernel Linuks jest makrokerneliem, tzn. zajmuje się nie tylko komunikacją pomiędzy poszczególnymi procesami, ale także zarządzaniem pamięcią, obsługą urządzeń czy obsługą systemów plików.

W architekturze mikrokernelsa, sam kernel zajmuje się tylko zapewnieniem komunikacji pomiędzy procesami, natomiast do obsługi poszczególnych urządzeń czy systemów plików, zajmują się osobne wykonywalne programy.

Jednak Linuks ma modułową budowę, co oznacza, że części kodu odpowiadające za obsługę poszczególnych urządzeń, systemów plików czy protokołów, mogą zostać od niego oddzielone - i dzięki temu zyskuje największe zalety mikrokernelsa:

- obsługę poszczególnych urządzeń czy protokołów można rozwijać niezależnie od samego kernelsa (nie wymaga to zmian w innych częściach kernelsa)
- podział prac nad jądrem na grupy zajmujące się poszczególnymi częściami znacznie ułatwia i przyspiesza rozwój
- możliwość zmniejszenia wielkości kernelsa, poprzez wykasowanie z pamięci operacyjnej nieużywanych modułów
- możliwość tworzenia zamkniętych, binarnych sterowników

Przy tym budowa jako makrokernel, i możliwość wbudowania modułów w strukturę samego kernelsa, przyspiesza działanie, ponieważ nie ma konieczności wielokrotnego powtarzania przełączania procesów, przy komunikacji pomiędzy poszczególnymi modułami.

Do czego służą moduły

Moduły, odpowiadają sterownikom w terminologii DOS/Windows. Mogą one obsługiwać nie tylko urządzenia - lista zadań modułów:

- bezpośrednia komunikacja z urządzeniem (standardowe, systemowe polecenia tłumaczy na kod rozumiany przez konkretny model urządzenia)

- **bezpośredni odczyt/zapis w systemie plików** (dzięki temu system widzi pliki, katalogi i ich właściwości, natomiast sam moduł dba o przetłumaczenie tego na kombinację bajtów w określonych miejscach na dysku)
- **obsługa protokołów** (np. protokołów sieciowych).

Najogólniej, moduły kernela służą do komunikacji niskopoziomowej - zajmują się tłumaczeniem standardowych poleceń na kod odpowiadający im w danym przypadku.

Polecenie do obsługi modułów

dkms

modinfo

lsmod - wyświetlania listy aktualnie załadowanych modułów

ładowanie modułów

- insmod
- modprobe – polecenie służące do wysokopoziomowej obsługi modułów (ładowanie)

usuwanie

- rmmod – polecenie służące do wysokopoziomowej obsługi modułów (usuwanie)

Źródło: wikipedia