

BSD - alternatywa dla Linuksa

Compatibility layer w FreeBSD

Maria Donten
Bartłomiej Etenkowski
Aleksander Zabłocki

Systemy Operacyjne 2006

Plan

- 1 Pliki binarne
- 2 Obsługiwane programy
- 3 Korzystanie z warstwy zgodności

Plan

- 1 Pliki binarne
- 2 Obsługiwane programy
- 3 Korzystanie z warstwy zgodności

Rodzaje plików binarnych

- **a.out** - początkowy uniksowy format plików wykonywalnych
 - ▶ nagłówek
 - ▶ do 6 sekcji, odpowiadających z grubsza assemblerowym `.TEXT`, `.DATA` i `.BSS`
- **COFF** - uogólnienie wprowadzone w SVR3
 - ▶ plik może zawierać wiele sekcji jednego rodzaju
 - ▶ wykorzystywane np. przez biblioteki współdzielone
 - ▶ niewygodne ograniczenia na ilość sekcji, długość ich nazw itp.

Rodzaje plików binarnych

- **a.out** - początkowy uniksowy format plików wykonywalnych
 - ▶ nagłówek
 - ▶ do 6 sekcji, odpowiadających z grubsza assemblerowym `.TEXT`, `.DATA` i `.BSS`
- **COFF** - uogólnienie wprowadzone w SVR3
 - ▶ plik może zawierać wiele sekcji jednego rodzaju
 - ▶ wykorzystywane np. przez biblioteki współdzielone
 - ▶ niewygodne ograniczenia na ilość sekcji, długość ich nazw itp.

Rodzaje plików binarnych

- **a.out** - początkowy uniksowy format plików wykonywalnych
 - ▶ nagłówek
 - ▶ do 6 sekcji, odpowiadających z grubsza assemblerowym `.TEXT`, `.DATA` i `.BSS`
- **COFF** - uogólnienie wprowadzone w SVR3
 - ▶ plik może zawierać wiele sekcji jednego rodzaju
 - ▶ wykorzystywane np. przez biblioteki współdzielone
 - ▶ niewygodne ograniczenia na ilość sekcji, długość ich nazw itp.

Rodzaje plików binarnych

- **a.out** - początkowy uniksowy format plików wykonywalnych
 - ▶ nagłówek
 - ▶ do 6 sekcji, odpowiadających z grubsza assemblerowym `.TEXT`, `.DATA` i `.BSS`
- **COFF** - uogólnienie wprowadzone w SVR3
 - ▶ plik może zawierać wiele sekcji jednego rodzaju
 - ▶ wykorzystywane np. przez biblioteki współdzielone
 - ▶ niewygodne ograniczenia na ilość sekcji, długość ich nazw itp.

Rodzaje plików binarnych

- **PE** (PORTABLE EXECUTABLE) - wariant COFF-a wykorzystywany w Windows
 - ▶ bezwzględne adresowanie pamięci, w razie potrzeby przebazowywane przed uruchomieniem procesu
 - ▶ szybkie działanie po uruchomieniu procesu, narzut pamięci i czasu przy uruchamianiu
- **ELF** (EXECUTABLE LINKABLE FORMAT) - następcza a.out oraz COFF w systemach uniksowych
 - ▶ podejście przeciwne niż w PE: wszystkie adresy w kodzie binarnym są względne
 - ▶ w Linuksie format a.out był sztywniejszy, zatem ELF potrzebny już od wersji 1.2
 - ▶ w BSD później, w FreeBSD od wersji 3
- My zajmiemy się głównie plikami ELF

Rodzaje plików binarnych

- **PE** (PORTABLE EXECUTABLE) - wariant COFF-a wykorzystywany w Windows
 - ▶ bezwzględne adresowanie pamięci, w razie potrzeby przebazowywane przed uruchomieniem procesu
 - ▶ szybkie działanie po uruchomieniu procesu, narzut pamięci i czasu przy uruchamianiu
- **ELF** (EXECUTABLE LINKABLE FORMAT) - następcza a.out oraz COFF w systemach uniksowych
 - ▶ podejście przeciwne niż w PE: wszystkie adresy w kodzie binarnym są względne
 - ▶ w Linuksie format a.out był sztywniejszy, zatem ELF potrzebny już od wersji 1.2
 - ▶ w BSD później, w FreeBSD od wersji 3
- My zajmiemy się głównie plikami ELF

Rodzaje plików binarnych

- **PE** (PORTABLE EXECUTABLE) - wariant COFF-a wykorzystywany w Windows
 - ▶ bezwzględne adresowanie pamięci, w razie potrzeby przebazowywane przed uruchomieniem procesu
 - ▶ szybkie działanie po uruchomieniu procesu, narzut pamięci i czasu przy uruchamianiu
- **ELF** (EXECUTABLE LINKABLE FORMAT) - następca a.out oraz COFF w systemach uniksowych
 - ▶ podejście przeciwne niż w PE: wszystkie adresy w kodzie binarnym są względne
 - ▶ w Linuksie format a.out był sztywniejszy, zatem ELF potrzebny już od wersji 1.2
 - ▶ w BSD później, w FreeBSD od wersji 3
- My zajmiemy się głównie plikami ELF

Rodzaje plików binarnych

- **PE** (PORTABLE EXECUTABLE) - wariant COFF-a wykorzystywany w Windows
 - ▶ bezwzględne adresowanie pamięci, w razie potrzeby przebazowywane przed uruchomieniem procesu
 - ▶ szybkie działanie po uruchomieniu procesu, narzut pamięci i czasu przy uruchamianiu
- **ELF** (EXECUTABLE LINKABLE FORMAT) - następca a.out oraz COFF w systemach uniksowych
 - ▶ podejście przeciwne niż w PE: wszystkie adresy w kodzie binarnym są względne
 - ▶ w Linuksie format a.out był sztywniejszy, zatem ELF potrzebny już od wersji 1.2
 - ▶ w BSD później, w FreeBSD od wersji 3
- My zajmiemy się głównie plikami ELF

Rodzaje plików binarnych

- **PE** (PORTABLE EXECUTABLE) - wariant COFF-a wykorzystywany w Windows
 - ▶ bezwzględne adresowanie pamięci, w razie potrzeby przebazowywane przed uruchomieniem procesu
 - ▶ szybkie działanie po uruchomieniu procesu, narzut pamięci i czasu przy uruchamianiu
- **ELF** (EXECUTABLE LINKABLE FORMAT) - następca a.out oraz COFF w systemach uniksowych
 - ▶ podejście przeciwne niż w PE: wszystkie adresy w kodzie binarnym są względne
 - ▶ w Linuksie format a.out był sztywniejszy, zatem ELF potrzebny już od wersji 1.2
 - ▶ w BSD później, w FreeBSD od wersji 3
- My zajmiemy się głównie plikami ELF

Plan

- 1 Pliki binarne
- 2 Obsługiwane programy**
- 3 Korzystanie z warstwy zgodności

Obsługiwane programy

Compatibility layer w FreeBSD pozwala na uruchomienie ok. 90% programów Linuksowych, w tym na przykład:

- Netscape
- Adobe Acrobat
- StarOffice
- WordPerfect
- Oracle
- VMWare
- Doom
- Quake

Obsługiwane programy

Compatibility layer w FreeBSD pozwala na uruchomienie ok. 90% programów Linuksowych, w tym na przykład:

- Netscape
- Adobe Acrobat
- StarOffice
- WordPerfect
- Oracle
- VMWare
- Doom
- Quake

Obsługiwane programy

Compatibility layer w FreeBSD pozwala na uruchomienie ok. 90% programów Linuksowych, w tym na przykład:

- Netscape
- Adobe Acrobat
- StarOffice
- WordPerfect
- Oracle
- VMWare
- Doom
- Quake

Obsługiwane programy

Compatibility layer w FreeBSD pozwala na uruchomienie ok. 90% programów Linuksowych, w tym na przykład:

- Netscape
- Adobe Acrobat
- StarOffice
- WordPerfect
- Oracle
- VMWare
- Doom
- Quake

Obsługiwane programy

Compatibility layer w FreeBSD pozwala na uruchomienie ok. 90% programów Linuksowych, w tym na przykład:

- Netscape
- Adobe Acrobat
- StarOffice
- WordPerfect
- Oracle
- VMWare
- Doom
- Quake

Obsługiwane programy

Compatibility layer w FreeBSD pozwala na uruchomienie ok. 90% programów Linuksowych, w tym na przykład:

- Netscape
- Adobe Acrobat
- StarOffice
- WordPerfect
- Oracle
- VMWare
- Doom
- Quake

Plan

- 1 Pliki binarne
- 2 Obsługiwane programy
- 3 Korzystanie z warstwy zgodności**

Włączanie obsługi programów linuksowych

- warstwa zgodności domyślnie wyłączona

- włączenie:

```
kldload linux
```

- sprawdzenie aktywności:

```
kldstat → czy na liście jest linux.ko
```

- na stałe: ustawić

```
linux_enable="YES"
```

```
w /etc/rc.conf
```

Włączanie obsługi programów linuksowych

- warstwa zgodności domyślnie wyłączona

- włączenie:

```
kldload linux
```

- sprawdzenie aktywności:

```
kldstat → czy na liście jest linux.ko
```

- na stałe: ustawić

```
linux_enable="YES"
```

```
w /etc/rc.conf
```

Włączanie obsługi programów linuksowych

- warstwa zgodności domyślnie wyłączona

- włączenie:

```
kldload linux
```

- sprawdzenie aktywności:

```
kldstat → czy na liście jest linux.ko
```

- na stałe: ustawić

```
linux_enable="YES"
```

```
w /etc/rc.conf
```

Włączanie obsługi programów linuksowych

- warstwa zgodności domyślnie wyłączona

- włączenie:

```
kldload linux
```

- sprawdzenie aktywności:

```
kldstat → czy na liście jest linux.ko
```

- na stałe: ustawić

```
linux_enable="YES"
```

w `/etc/rc.conf`

Instalacja bibliotek Linuksa

- biblioteki Linuksa są gromadzone w katalogu `/compat/linux`
- kiedy program linuksowy odwołuje się do biblioteki `/etc/bleble`, FreeBSD sprawdza najpierw istnienie `/compat/linux/etc/bleble`, a potem `/etc/bleble` itp.

Instalacja bibliotek Linuksa

- biblioteki Linuksa są gromadzone w katalogu `/compat/linux`
- kiedy program linuksowy odwołuje się do biblioteki `/etc/bleble`, FreeBSD sprawdza najpierw istnienie `/compat/linux/etc/bleble`, a potem `/etc/bleble` itp.

Instalacja bibliotek Linuksa

Sposób 1: Instalacja automatyczna

- wymaga zainstalowania kolekcji portów w BSD
- `cd /usr/ports/emulators/linux_base-dysstryb`
(*dysstryb* określa emulowaną dystrybucję Linuksa)
- `make install distclean`

Instalacja bibliotek Linuksa

Sposób 1: Instalacja automatyczna

- wymaga zainstalowania kolekcji portów w BSD
- `cd /usr/ports/emulators/linux_base-dysstryb`
(*dysstryb* określa emulowaną dystrybucję Linuksa)
- `make install distclean`

Instalacja bibliotek Linuksa

Sposób 1: Instalacja automatyczna

- wymaga zainstalowania kolekcji portów w BSD
- `cd /usr/ports/emulators/linux_base-dysstryb`
(*dysstryb* określa emulowaną dystrybucję Linuksa)
- `make install distclean`

Instalacja bibliotek Linuksa

Sposób 1: Instalacja automatyczna

- wymaga zainstalowania kolekcji portów w BSD
- `cd /usr/ports/emulators/linux_base-dysstryb`
(*dysstryb* określa emulowaną dystrybucję Linuksa)
- `make install distclean`

Instalacja bibliotek Linuksa

Sposób 2: Instalacja ręczna

- Potrzebna, gdy:
 - ▶ Nie zainstalowano portu umożliwiającego autoinstalację
 - ▶ Potrzebne są inne biblioteki niż automatycznie zainstalowane
- Sprawdzić pod Linuksem zależności danego programu
`ldd nazwa_programu`
- Przekopiować do FreeBSD wszystkie wymienione biblioteki oraz łączy symboliczne do tych bibliotek...
- ...ale podstawiając w miejsce `/ katalog /compat/linux` (jeśli takiego katalogu nie ma, należy go utworzyć)

Instalacja bibliotek Linuksa

Sposób 2: Instalacja ręczna

- Potrzebna, gdy:
 - ▶ Nie zainstalowano portu umożliwiającego autoinstalację
 - ▶ Potrzebne są inne biblioteki niż automatycznie zainstalowane
- Sprawdzić pod Linuksem zależności danego programu
`ldd nazwa_programu`
- Przekopiować do FreeBSD wszystkie wymienione biblioteki oraz łączy symboliczne do tych bibliotek...
- ...ale podstawiając w miejsce `/ katalog / compat / linux` (jeśli takiego katalogu nie ma, należy go utworzyć)

Instalacja bibliotek Linuksa

Sposób 2: Instalacja ręczna

- Potrzebna, gdy:
 - ▶ Nie zainstalowano portu umożliwiającego autoinstalację
 - ▶ Potrzebne są inne biblioteki niż automatycznie zainstalowane
- Sprawdzić pod Linuksem zależności danego programu
`ldd nazwa_programu`
- Przekopiować do FreeBSD wszystkie wymienione biblioteki oraz łączy symboliczne do tych bibliotek...
- ...ale podstawiając w miejsce `/ katalog / compat / linux` (jeśli takiego katalogu nie ma, należy go utworzyć)

Instalacja bibliotek Linuksa

Sposób 2: Instalacja ręczna

- Potrzebna, gdy:
 - ▶ Nie zainstalowano portu umożliwiającego autoinstalację
 - ▶ Potrzebne są inne biblioteki niż automatycznie zainstalowane
- Sprawdzić pod Linuksem zależności danego programu
`ldd nazwa_programu`
- Przekopiować do FreeBSD wszystkie wymienione biblioteki oraz łącza symboliczne do tych bibliotek...
- ...ale podstawiając w miejsce `/ katalog / compat / linux` (jeśli takiego katalogu nie ma, należy go utworzyć)

Instalacja bibliotek Linuksa

Sposób 2: Instalacja ręczna

- Potrzebna, gdy:
 - ▶ Nie zainstalowano portu umożliwiającego autoinstalację
 - ▶ Potrzebne są inne biblioteki niż automatycznie zainstalowane
- Sprawdzić pod Linuksem zależności danego programu
`ldd nazwa_programu`
- Przekopiować do FreeBSD wszystkie wymienione biblioteki oraz łączy symboliczne do tych bibliotek...
- ...ale podstawiając w miejsce `/ katalog /compat/linux` (jeśli takiego katalogu nie ma, należy go utworzyć)

Co dalej

- Rodzaj plików ELF

- ▶ Pliki ELF mają **rodzaj (brand)**, określający np. system operacyjnym
- ▶ Niektóre programy linuxowe nie mają ustawionego rodzaju, wówczas FreeBSD może ich nie uruchomić
- ▶ Zmiana rodzaju na Linux:
`brandelf -t Linux program`

- Serwer nazw

- ▶ w przypadku błędu dotyczącego słów `bind` i `hosts` należy utworzyć plik `/compat/linux/etc/host.conf` o zawartości
`order hosts, bind`
`multi on`
- ▶ słowo `bind` należy pominąć, jeśli serwer nazw nie został określony w pliku `/etc/resolv.conf`

Co dalej

- Rodzaj plików ELF

- ▶ Pliki ELF mają **rodzaj (brand)**, określający np. system operacyjnym
- ▶ Niektóre programy linuksowe nie mają ustawionego rodzaju, wówczas FreeBSD może ich nie uruchomić
- ▶ Zmiana rodzaju na Linux:
`brandelf -t Linux program`

- Serwer nazw

- ▶ w przypadku błędu dotyczącego słów `bind` i `hosts` należy utworzyć plik `/compat/linux/etc/host.conf` o zawartości
`order hosts, bind`
`multi on`
- ▶ słowo `bind` należy pominąć, jeśli serwer nazw nie został określony w pliku `/etc/resolv.conf`

Co dalej

- Rodzaj plików ELF

- ▶ Pliki ELF mają **rodzaj (brand)**, określający np. system operacyjnym
- ▶ Niektóre programy linuxowe nie mają ustawionego rodzaju, wówczas FreeBSD może ich nie uruchomić
- ▶ Zmiana rodzaju na Linux:
`brandelf -t Linux program`

- Serwer nazw

- ▶ w przypadku błędu dotyczącego słów `bind` i `hosts` należy utworzyć plik `/compat/linux/etc/host.conf` o zawartości
`order hosts, bind`
`multi on`
- ▶ słowo `bind` należy pominąć, jeśli serwer nazw nie został określony w pliku `/etc/resolv.conf`

Co dalej

- Rodzaj plików ELF

- ▶ Pliki ELF mają **rodzaj (brand)**, określający np. system operacyjnym
- ▶ Niektóre programy linuxowe nie mają ustawionego rodzaju, wówczas FreeBSD może ich nie uruchomić
- ▶ Zmiana rodzaju na Linux:
`brandelf -t Linux program`

- Serwer nazw

- ▶ w przypadku błędu dotyczącego słów `bind` i `hosts` należy utworzyć plik `/compat/linux/etc/host.conf` o zawartości
`order hosts, bind`
`multi on`
- ▶ słowo `bind` należy pominąć, jeśli serwer nazw nie został określony w pliku `/etc/resolv.conf`

Co dalej

- Rodzaj plików ELF

- ▶ Pliki ELF mają **rodzaj (brand)**, określający np. system operacyjnym
- ▶ Niektóre programy linuxowe nie mają ustawionego rodzaju, wówczas FreeBSD może ich nie uruchomić
- ▶ Zmiana rodzaju na Linux:
`brandelf -t Linux program`

- Serwer nazw

- ▶ w przypadku błędu dotyczącego słów `bind` i `hosts` należy utworzyć plik `/compat/linux/etc/host.conf` o zawartości
`order hosts, bind`
`multi on`
- ▶ słowo `bind` należy pominąć, jeśli serwer nazw nie został określony w pliku `/etc/resolv.conf`