

Instytut Elektroniki  
Wydział Automatyki, Elektroniki i Informatyki  
Politechniki Śląskiej

## **SIECI KOMPUTEROWE**

### **LABORATORIUM**

#### **Instrukcja do ćwiczenia nr 8**

Gliwice 2018

## Ćwiczenie 8

# Konfiguracja i aktualizacja serwera sieciowego typu UNIX

### 8.1 Wstęp

#### 8.1.1 Konfiguracja serwera

Konfiguracja serwera, podobnie jak konfiguracja systemu operacyjnego, to szereg czynności mających na celu dostosowanie systemu do potrzeb i wymagań użytkownika oraz do funkcji jaką w przyszłości będzie pełnił serwer. Konfigurację rozpoczyna się najczęściej zaraz po instalacji systemu, wtedy jest to tzw. wstępna konfiguracja, lub też wtedy gdy zmieniają warunki pracy serwera (np. zmiana zasobów sprzętowych, oprogramowania) lub jego przeznaczenie. W podrozdziale tym zostaną omówione podstawowe czynności konfiguracyjne dokonywane zwykle po instalacji systemu operacyjnego FreeBSD.

Większość plików konfiguracyjnych systemu operacyjnego FreeBSD, podobnie jak w przypadku pozostałych systemów UNIX znajduje się w kartotekach: `/etc` oraz `/usr/local/etc`. Najważniejsze kartoteki konfiguracyjne przedstawiono poniżej wraz z ich przeznaczeniem:

- `/etc` - lokalizacja głównych plików konfiguracyjnych systemu FreeBSD lub aplikacji będących integralną częścią systemu operacyjnego,
- `/etc/defaults` - zestaw domyślnych plików konfiguracyjnych,
- `/etc/mail` - pliki konfiguracyjne poczty elektronicznej,
- `/etc/ppp` - pliki konfiguracyjne usługi PPP,
- `/etc/namedb` - pliki konfiguracyjne usługi DNS,

- `/etc/periodic/` - skrypty systemowe uruchamiane cyklicznie.
- `/usr/local/etc` - pliki konfiguracyjne dodatkowego oprogramowania, nie będącego integralną częścią systemu operacyjnego,
- `/usr/local/etc/rc.d` - skrypty startowe dodatkowego oprogramowania,

Tabela 8.1: Podstawowe pliki konfiguracyjne systemu operacyjnego FreeBSD

nazwa pliku	przeznaczenie pliku konfiguracyjnego
<code>/etc/crontab</code>	konfiguracja demona operacji cyklicznych
<code>/etc/disktab</code>	specyfikacja geometrii napędów dyskowych
<code>/etc/fstab</code>	specyfikacja systemu plików
<code>/etc/group</code>	lista grup do których mogą należeć użytkownicy
<code>/etc/hosts</code>	baza danych nazw oraz numerów IP maszyn lokalnych
<code>/etc/hosts.allow</code>	konfiguracja dostępu do serwera z maszyn zdalnych
<code>/etc/inetd.conf</code>	konfiguracja demona obsługi portów internetowych
<code>/etc/login.access</code>	konfiguracja dostępu użytkowników do serwera
<code>/etc/login.conf</code>	konfiguracja i specyfikacja uprawnień klas użytkowników
<code>/etc/master.passwd</code>	specyfikacja danych o użytkownikach wraz z hasłami
<code>/etc/mord</code>	komunikat dzienny wyświetlany przy logowaniu
<code>/etc/networks</code>	specyfikacja podziału sieci lokalnej na podsieci
<code>/etc/passwd</code>	specyfikacja danych o użytkownikach
<code>/etc/printcap</code>	konfiguracja dostępu do drukarek
<code>/etc/rc.*</code>	grupa plików startowych systemu
<code>/etc/rc.conf</code>	specyfikacja parametrów startowych serwera FreeBSD
<code>/etc/shells</code>	lista dozwolonych w systemie interpreterów poleceń
<code>/etc/syslog.conf</code>	konfiguracja demona komunikatów systemowych
<code>/usr/local/etc</code>	pliki konfiguracyjne zainstalowanych aplikacji.
<code>/usr/local/etc/rc.d</code>	skrypty uruchamiające zainstalowane aplikacje.
<code>/var/db</code>	trwałe dane systemowe, pliki baz danych itp.

W tabeli 8.1 zestawiono podstawowe pliki konfiguracyjne systemu FreeBSD wraz z ich lokalizacją oraz przeznaczeniem. Większość parametrów startowych serwera FreeBSD zawarta jest w pliku `\etc\rc.conf`. Modyfikacji zawartości tego pliku jak i zmiany niektórych parametrów konfiguracyjnych można dokonywać ręcznie modyfikując jego zawartość za pomocą dowolnego edytora tekstowego, lub też za pomocą programu narzędziowego `bsdconfig`.

Domyślna konfiguracja serwera zawarta jest w pliku `\etc\default\rc.conf`. Nie należy modyfikować zawartości tego pliku. Wszelkie zmiany konfiguracji dokonujemy tylko i wyłącznie poprzez modyfikację zawartości pliku `\etc\rc.conf`.

W niniejszym ćwiczeniu zagadnienia konfiguracji systemu ograniczy się do omówienia jej trzech aspektów, realizowanych najczęściej zaraz po zainstalowaniu systemu operacyjnego. Są to:

1. Konfiguracja i instalacja jądra systemowego,
2. Instalacja dodatkowego oprogramowania,
3. Aktualizacja systemu FreeBSD.

### 8.1.2 Konfiguracja, kompilacja i instalacja jądra systemowego

Jądro stanowi tę część systemu operacyjnego, która pośredniczy w dostępie do zasobów sprzętowych serwera. Jądro systemu odpowiedzialne jest za obsługę urządzeń wewnętrznych i współpracę z pozostałymi komponentami systemu operacyjnego, mającą na celu optymalne wykorzystanie zasobów sprzętowych takich jak: czas procesora, pamięć operacyjna, pamięć masowa i urządzenia peryferyjne [3].

W praktyce spotyka się dwa rodzaje budowy jądra systemowego: monolityczną oraz modułową. W przypadku monolitycznego jądra systemowego przy każdorazowej zmianie zasobów sprzętowych konieczna jest konfiguracja, kompilacja i instalacja nowego zmodyfikowanego jądra systemowego. Konfiguracja jądra systemowego umożliwia jego dostosowanie do konfiguracji sprzętowej komputera i funkcji jaką będzie pełnił system operacyjny. Pozwala na uaktywnienie tych elementów jądra systemowego, które odpowiadają danej konfiguracji sprzętowej komputera, dzięki czemu system operacyjny będzie pracował bardziej optymalnie i niezawodnie, a zasoby sprzętowe (pamięć i czas procesora) nie będą tracone na obsługę niewykorzystywanych na danej platformie sprzętowej urządzeń. Dla przykładu jądro systemowe przystosowane do obsługi zbyt wielu urządzeń, często nie istniejących w danym systemie, będzie miało dużą redundancję i będzie zajmowało więcej pamięci operacyjnej niż jest to konieczne.

Wiele systemów operacyjnych, w tym także FreeBSD ma modułową budowę jądra systemowego. Dzięki takiej architekturze możliwe jest dostosowanie jądra systemowego do wymagań bez potrzeby jego kompilacji. W praktyce polega to na wyborze i załadowaniu tylko tych modułów jądra systemowego które są niezbędne w danej konfiguracji sprzętowej.

Dobre dostosowanie jądra systemowego do potrzeb użytkownika ma fundamentalne znaczenie dla pracy całego systemu operacyjnego i daje następujące korzyści:

- Skracza czas załadowania jądra systemowego, a tym samym restartu systemu operacyjnego.
- Zmniejsza zajęty przez jądro systemu obszar pamięci operacyjnej.
- Umożliwia pracę z nowo zainstalowanymi urządzeniami zarówno wewnątrz systemu jak i urządzeniami peryferyjnymi.

Wersja dystrybucyjna systemu operacyjnego FreeBSD jest dostarczana wraz z plikami źródłowymi systemu operacyjnego oraz jądra systemu operacyjnego. Zaleca się przy instalowaniu systemu FreeBSD zainstalować także pliki źródłowe jądra systemowego. Umożliwi to łatwą konfigurację, kompilację i instalację nowego jądra, gdy zajdzie taka potrzeba. Źródła jądra systemowego FreeBSD znajdują się zwykle w katalogu `\usr\src\sys`. W przypadku, gdy źródła nie zostały zainstalowane przy instalacji systemu, możliwe jest zainstalowanie źródeł z dystrybucyjnego dysku CD-ROM. W takim przypadku należy wykonać następującą sekwencję poleceń:

```
# mount /cdrom
# mkdir -p /usr/src/sys
# ln -s /usr/src/sys /sys
# cat /cdrom/sys/ssys.[a-d]* | tar -xzvf
```

Gdy źródła jądra systemowego znajdują się na serwerze, możliwe jest przystąpienie do konfiguracji jądra. W tym celu należy stworzyć własny plik konfiguracyjny. Najlepszym rozwiązaniem jest wykorzystanie w tym celu istniejącego pliku konfiguracyjnego o nazwie `GENERIC` :

```
# cd /usr/src/sys/i386/conf
# cp GENERIC MYKERNEL
```

Konfigurację jądra systemowego polega na modyfikacji pliku konfiguracyjnego, w naszym przypadku jest to plik `MYKERNEL` znajdujący się w katalogu `/usr/src/sys/i386/conf`. Przykładowy fragment zawartości tego pliku przedstawiono poniżej.

```
# GENERIC -- Generic kernel configuration file for FreeBSD/i386
# $FreeBSD: src/sys/i386/conf/GENERIC,v 1.246.2.38 2002/01/25 17:41:40
machine      i386
cpu          I386_CPU
cpu          I486_CPU
cpu          I586_CPU
```

```
cpu      I686_CPU
ident    GENERIC
maxusers 32
options  MATH_EMULATE      #Support for x87 emulation
options  INET              #InterNETworking
options  INET6             #IPv6 communications protocols
options  FFS               #Berkeley Fast Filesystem
options  FFS_ROOT          #FFS usable as root device [keep this!]
options  SOFTUPDATES        #Enable FFS soft updates support
options  UFS_DIRHASH        #Improve performance on big directories
options  MFS               #Memory Filesystem
options  MD_ROOT           #MD is a~potential root device
options  NFS               #Network Filesystem
options  NFS_ROOT          #NFS usable as root device, NFS required
options  MSDOSFS           #MSDOS Filesystem
device   isa
device   eisa
device   pci
```

W pliku konfiguracyjnym znajdują się odpowiednie sekcje odpowiedzialne za poszczególne funkcje jądra systemowego oraz za obsługę poszczególnych urządzeń. Ustalenie odpowiedniej konfiguracji w pliku konfiguracyjnym możliwe za pomocą odpowiednich słów kluczowych. Oto niektóre z nich:

- **machine** - określenie architektury sprzętowej systemu,
- **cpu** - specyfikacja rodzaju procesora,
- **ident** - identyfikator (nazwa) jądra systemowego,
- **maxusers** - maksymalna liczba użytkowników systemu,
- **options** - uaktywnienie dodatkowych opcji (funkcji) jądra systemowego,
- **device** - uaktywnienie obsługi danego typu urządzeń.

Po ustaleniu zawartości pliku konfiguracyjnego można przystąpić do kompilacji i instalacji nowego jądra systemowego. W tym celu będąc w kartotece `/usr/src/sys/i386/conf` wykonujemy szereg poleceń :

- **config <nazwa\_pliku\_konfiguracyjnego>** - wygenerowane odpowiednich plików niezbędnych do kompilacji nowego jądra systemowego,
- **cd ../compile/<nazwa\_pliku\_konfiguracyjnego>** - przejście do kartoteki źródłowej,

- **make depend** - kompilacja zmodyfikowanych elementów jądra systemowego,
- **make** - kompilacja jądra systemowego oraz jego modułów,
- **make install** - instalacja nowego jądra systemowego.

Generalnie w wraz z wersją dystrybucyjną systemu FreeBSD dostarczone są dwa pliki konfiguracyjne jądra systemowego. Plik **GENERIC** opisujący parametry jądra systemowego używanego przy starcie systemu FreeBSD zaraz po jego instalacji, oraz plik konfiguracyjny o nazwie **LINT** uaktywniający wszystkie możliwe funkcje dodatkowe jądra systemowego oraz współpracę ze wszystkimi urządzeniami przystosowanymi do obsługi przez system FreeBSD. Pełny wykaz urządzeń dostępny jest w dokumentacji systemu FreeBSD. Tam też można znaleźć szczegółowe informacje na temat konfiguracji jądra systemowego oraz pozostałych czynności konfiguracyjnych [1].

### 8.1.3 Instalacja dodatkowego oprogramowania

W systemie UNIX instalacja dodatkowego oprogramowania przebiega zwykle w następujących krokach:

- Pobranie z Internetu oprogramowania w wersji binarnej (skompilowanej i dedykowanej dla danego systemu) lub w wersji źródłowej wymagającej kompilacji,
- Rozpakowanie archiwum,
- Zapoznanie się z dokumentacją oprogramowania opisującą sposób instalacji i konfiguracji,
- Uruchomienie skryptu konfiguracyjnego **configure** i kompilacja wersji źródłowej oprogramowania,
- Instalacja, testowanie i konfigurowanie oprogramowania.

Poważne problemy zaczynają się gdy oprogramowanie nie posiada skryptu instalacyjnego, wtedy użytkownik musi zainstalować oprogramowanie ręcznie. Zdarza się także że dokumentacja jest niewystarczająca lub zainstalowanie danego oprogramowania wymaga uprzedniej instalacji innych, wykorzystywanych przez to oprogramowanie aplikacji. W takim przypadku użytkownik napotyka na poważne trudności.

Twórcy systemu operacyjnego FreeBSD starali się maksymalnie ułatwić użytkownikowi instalację dodatkowego oprogramowania i zautomatyzować

ten proces tak jak to tylko możliwe. Wraz z zainstalowaniem systemu operacyjnego FreeBSD, użytkownikowi udostępniony jest zestaw ponad kilkudziesięć tysięcy różnych aplikacji gotowych do zainstalowania. Po uaktywnieniu kompatybilności binarnej z innymi systemami np. systemem linux, liczba aplikacji dostępnych dla użytkownika sięga kilkuset tysięcy. Szereg usprawnień powoduje, że instalacja lub deinstalacja dodatkowego oprogramowania jest maksymalnie uproszczona. Odpowiednie narzędzia umożliwiają rejestrację i katalogowanie zainstalowanych aplikacji, co ułatwia zarządzanie nimi, sprawdzenie aktualnie dostępnych wersji, automatyczną aktualizację lub deinstalację wybranych aplikacji. W przypadku gdy zainstalowanie wybranego oprogramowania wymaga uprzednie zainstalowanie współpracujących z nią aplikacji, to proces ten jest całkowicie zautomatyzowany i nie wymaga ingerencji użytkownika. Lista dostępnego oprogramowania dedykowanego dla systemu operacyjnego FreeBSD jest dostępna na serwerze projektu FreeBSD<sup>1</sup>. Generalnie użytkownik ma możliwość instalacji dodatkowego oprogramowania w dwóch trybach:

- Instalacja oprogramowania w wersji binarnej (skompilowanej),
- Instalacja oprogramowania w wersji źródłowej, wymagająca uprzedniej kompilacji (najczęściej za pomocą kompilatora C++) w celu stworzenia plików binarnych.

### Instalacja dodatkowego oprogramowania w wersji binarnej

Instalację oprogramowania w wersji binarnej przeprowadza się za pomocą polecenia `pkg_add`. Istnieją co najmniej trzy tryby instalacji oprogramowania w wersji binarnej. Oto one wraz z przykładami:

- `pkg install lsof-4.56.4.tgz` - instalacja oprogramowania o nazwie `lsof` w wersji `4.56.4` z dysku lokalnego,

System rejestracji zainstalowanych aplikacji umożliwia łatwe zarządzanie oprogramowaniem. Oto podstawowe funkcje tego systemu wraz z przykładami:

- `pkg delete lsof-4.56.4` - deinstalacja wybranego oprogramowania o nazwie `lsof` w wersji `4.56.4`,
- `pkg info` - wyświetlenie listy aktualnie zainstalowanych aplikacji dodatkowych wraz z krótkim opisem ich przeznaczenia,

---

<sup>1</sup><http://www.FreeBSD.org/ports/>



- `pkg version` - wyświetlenie listy zainstalowanych aplikacji wraz z informacją o dostępnych uaktualnieniach,

Oprogramowanie dodatkowe w wersji binarnej można także zainstalować z płyty dystrybucyjnej CD-ROM oraz za pośrednictwem sieci Internet wykorzystując do tego celu program instalacyjny `bsdconfig`.

### Instalacja dodatkowego oprogramowania w wersji źródłowej

Instalacja oprogramowania w wersji źródłowej wymaga pobrania wersji źródłowej wybranego oprogramowania na dysk lokalny. Proces ten został w znacznym stopniu zautomatyzowany. Po instalacji systemu FreeBSD na dysku lokalnym zapisane są informacje o dostępnym oprogramowaniu, wraz z krótkim opisem oraz skryptami za pomocą których możliwe jest automatyczne pobranie, kompilacja i instalacja wybranych aplikacji. Instalację rozpoczyna się o zlokalizowania skryptów instalacyjnych danego oprogramowania (w tym wypadku aplikacji o nazwie `lsof`). Dokonać można tego za pomocą komendy `whereis`:

```
# whereis lsof
lsof: /usr/ports/sysutils/lsof
```

Następnie należy wejść do wskazanego katalogu zawierającego skrypty instalacyjne oprogramowania (w tym wypadku aplikacji o nazwie `lsof`) i wykonać polecenie `make`:

```
# cd /usr/ports/sysutils/lsof
# make install
```

Po wykonaniu powyższych poleceń rozpocznie się zautomatyzowany proces instalacji oprogramowania składający się z następujących etapów:

- sprawdzenie czy na dysku lokalnym lub napędzie CD-ROM znajduje się najnowsza zalecana wersja oprogramowania,
- jeżeli nie, to następuje wyszukanie oprogramowania w internecie i pobranie na dysk lokalny wersji źródłowej,
- zainstalowanie wymaganych łat (*ang. patches*) jeśli takie istnieją,
- konfiguracja plików źródłowych i przystosowanie ich do kompilacji pod systemem operacyjnym FreeBSD,
- kompilacja plików źródłowych,

- instalacja i rejestracja oprogramowania,
- wyświetlenie opcjonalnych komunikatów, w przypadku gdy wymagana jest ingerencja użytkownika.

Po uruchomieniu na ekranie wyświetlona zostaną komunikaty informujące o przebiegu poszczególnych etapów tego procesu:

```
# make install
>> lsof_4.57D.freebsd.tar.gz doesn't seem to exist in /usr/ports/distfiles/.
>> Attempting to fetch from ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/distfiles/.
Receiving lsof_4.57D.freebsd.tar.gz (439860 bytes): 100%
439860 bytes transferred in 18.0 seconds (23.90 kBps)
===> Extracting for lsof-4.57
...
[extraction output snipped]
...
>> Checksum OK for lsof_4.57D.freebsd.tar.gz.
===> Patching for lsof-4.57
===> Applying FreeBSD patches for lsof-4.57
===> Configuring for lsof-4.57
===> Building for lsof-4.57
===> Installing for lsof-4.57
===> Generating temporary packing list
===> Compressing manual pages for lsof-4.57
===> Registering installation for lsof-4.57
===> SECURITY NOTE:
      This port has installed the following binaries which execute with
      increased privileges.
```

Po zainstalowaniu aplikacji dostępne są wszystkie funkcje zarządzania zainstalowanym oprogramowaniem, tak jak to miało miejsce po instalacji oprogramowania w wersji binarnej.

### 8.1.4 Aktualizacja systemu operacyjnego FreeBSD

System operacyjny FreeBSD jest systemem stale rozwijającym się. Grupa programistów stale pracuje nad nowymi wersjami systemu wprowadzając ciągle zmiany, polegające na usuwaniu zgłoszonych przez użytkowników błędów, dodaniu nowych funkcji do systemu i zwiększeniu możliwości oraz funkcjonalności systemu operacyjnego. Efekty pracy tej grupy można docenić wraz każdą nową wersją dystrybucyjną systemu, która wydawana jest co kilka miesięcy. Są jednak mechanizmy pozwalające na śledzenie, synchronizację i aktualizację plików źródłowych do bieżącej wersji systemu. Jedną z nich jest SVN lub Subversion. Jeśli polecenie `svn` nie jest dostępne konieczna będzie instalacja oprogramowania:

```
# pkg install subversion
```

Polecenie `svn` pozwala na synchronizację posiadanych przez użytkownika plików źródłowych z aktualną ich wersją. W ramach projektu FreeBSD dostępne są następujące wersje źródłowe systemu operacyjnego:

- **FreeBSD-CURRENT** - bieżąca, najnowsza wersja systemu operacyjnego FreeBSD, często zawierająca jeszcze błędy, które mogą być powodem niestabilności systemu. Testowanie pozwala na wykrycie błędów i stworzenie wersji stabilnej.
- **FreeBSD-STABLE** - wersja stabilna systemu operacyjnego FreeBSD, zalecana dla końcowego użytkownika. Ta wersja systemu jest wersją dystrybucyjną wzbogaconą o krytyczne poprawki do systemu operacyjnego.
- **FreeBSD-RELEASE** - wersja dystrybucyjna systemu operacyjnego FreeBSD, dostępna także na nośnikach CD-ROM.

Zaleca się przeprowadzanie aktualizacji systemu wtedy, gdy na serwerze głównym projektu FreeBSD <sup>2</sup> będzie dostępna informacja o aktualnych błędach lub poprawkach bieżącej wersji dystrybucyjnej systemu. Uaktualnienie systemu będzie wówczas możliwe poprzez zainstalowanie odpowiednich łat (*ang. patches*) lub też poprzez synchronizację plików źródłowych oraz ich późniejszą kompilację i instalację. W ramach ćwiczenia zajmiemy się synchronizacją plików źródłowych systemu FreeBSD za pośrednictwem narzędzia SVN.

### Aktualizacja plików binarnych systemu operacyjnego

Aktualizacja plików binarnych systemu operacyjnego FreeBSD jest dalece zautomatyzowana. Do jej przeprowadzenia służy polecenie `freebsd-upgrade`. Instalacja niezbędnych aktualizacji binarnych systemu operacyjnego przeprowadzana jest przez wykonanie sekwencji poleceń:

```
# freebsd-update fetch
# freebsd-update install
```

Proces ten można zautomatyzować dodając do pliku `/etc/crontab` odpowiednią linię:

```
@daily    root    freebsd-update cron
```

---

<sup>2</sup> <http://www.FreeBSD.org>

W takim przypadku poprawki aktualizacyjne będą instalowane codziennie bez udziału administratora serwera.

Deinstalacja zainstalowanych plików aktualizacyjnych jest możliwa przez wykonanie polecenia:

```
# freebsd-update rollback
```

Aktualizacja systemu operacyjnego do nowej wersji możliwa jest przez wykonanie polecenia:

```
# freebsd-update -r 9.1-RELEASE upgrade
```

Jest to proces dość skomplikowany wymagający w określonych przypadkach podjęcie oczekiwanych działań przez administratora serwera:

```
Looking up update.FreeBSD.org mirrors... 1 mirrors found.
Fetching metadata signature for 9.0-RELEASE from update1.FreeBSD.org... done.
Fetching metadata index... done.
Inspecting system... done.
```

The following components of FreeBSD seem to be installed:

```
kernel/smp src/base src/bin src/contrib src/crypto src/etc src/games
src/gnu src/include src/krb5 src/lib src/libexec src/release src/rescue
src/sbin src/secure src/share src/sys src/tools src/ubin src/usbin
world/base world/info world/lib32 world/manpages
```

The following components of FreeBSD do not seem to be installed:

```
kernel/generic world/catpages world/dict world/doc world/games
world/proflibs
```

Does this look reasonable (y/n)? y

Po wykonaniu tych działań należy kontynuować proces aktualizacji za pomocą sekwencji poleceń:

```
# freebsd-update install
# shutdown -r now
# freebsd-update install
```

Po wykonaniu aktualizacji systemu operacyjnego może zaistnieć konieczność aktualizacji oprogramowania narzędziowego `pkg`:

```
# pkg-static upgrade -f
# portmaster -af
# freebsd-update install
```

## Aktualizacja plików źródłowych systemu operacyjnego

Możliwe jest pobranie plików źródłowych systemu operacyjnego w wybranej przez użytkownika wersji:

```
# mv /usr/src /usr/src.bak  
# svn checkout https://svn.freebsd.org/base/releng/10.3 /usr/src
```

Następnie konieczne jest dokonanie odpowiedniego wpisu do pliku `make.conf`:

```
SVN_UPDATE=yes
```

Późniejsza aktualizacja możliwa jest do przeprowadzenia przez wykonanie polecenia:

```
# svn update /usr/src
```

lub

```
# cd /usr/src  
# make update
```

Po zakończeniu aktualizacji można przystąpić do kompilacji źródeł systemu operacyjnego poleceniem:

```
# cd /usr/src  
# make buildworld
```

Kompilacja całego systemu może zająć nawet do kilku godzin w zależności od mocy obliczeniowej komputera. Ponieważ podczas aktualizacji zmianie mogły ulec także pliki źródłowe jądra systemowego, wymagana jest jego powtórna kompilacja:

```
# make buildkernel KERNCONF=<nazwa_jądra_systemowego>
```

Po zakończonej kompilacji nie pozostaje nic innego jak zainstalować nową wersję systemu i jądra systemowego na podstawie skompilowanej aktualnej wersji stabilnej systemu:

```
# make installworld  
# make installkernel KERNCONF=<nazwa_jądra_systemowego>
```

Po zakończeniu instalacji zalecany jest natychmiastowy restart systemu operacyjnego:

```
# shutdown -r now
```

## Aktualizacja plików dokumentacji systemu operacyjnego

Pobranie aktualnej wersji plików dokumentacji systemu operacyjnego możliwe jest przez wykonanie polecenia:

```
svn checkout https://svn.FreeBSD.org/doc/head /usr/doc
```

Następnie konieczne jest dokonanie odpowiedniego wpisu do pliku `make.conf`:

```
SVN_UPDATE=yes
```

Późniejsza aktualizacja możliwa jest do przeprowadzenia przez wykonanie polecenia:

```
# svn update /usr/doc
```

lub

```
# cd /usr/doc
```

```
# make update
```

Możliwa jest kompilacja plików dokumentacji do wybranego formatu danych:

```
# cd /usr/doc/en_US.ISO8859-1
```

```
# make install clean
```

## Aktualizacja portów aplikacji systemu operacyjnego

Aktualizacja portów możliwa jest do przeprowadzenia na 2 sposoby. Pierwsza metoda polega na skorzystaniu z polecenia `portsnap`: ]

```
# portsnap fetch
```

```
# portsnap extract
```

Późniejsza aktualizacja tą jest możliwa przez wykonanie sekwencji poleceń:

```
# portsnap fetch
```

```
# portsnap update
```

lub

```
# portsnap fetch update
```

Druga metoda polega na wykorzystaniu polecenia `svn`:

```
# svn checkout https://svn.FreeBSD.org/ports/head /usr/ports
```

Późniejsza aktualizacja możliwa jest do przeprowadzenia przez wykonanie polecenia:

```
# svn update /usr/ports
```

lub

```
# cd /usr/ports
```

```
# make update
```

### Aktualizacja dodatkowego oprogramowania w wersji binarnej

Wyszczególnienie listy aplikacji z zaznaczeniem wymagających aktualizacji jest możliwe przez wykonanie sekwencji poleceń:

```
# pkg update  
# pkg version
```

Aktualizacji dodatkowego oprogramowania zainstalowanego na serwerze w wersji binarnej jest możliwa przez wykonanie sekwencji poleceń:

```
# pkg update  
# pkg upgrade
```

### Literatura

1. *FreeBSD Handbook*, The FreeBSD Documentation Project - Copyright © by The FreeBSD Documentation Project (<http://www.freebsd.org/handbook>).
2. *FreeBSD/i386 RELEASE Hardware Notes*, The FreeBSD Documentation Project - Copyright © by The FreeBSD Documentation Project
3. Tanenbaum A.S, *Computer Networks*, Prentice Hall, New Jersey 2003.
4. *The UNIX Reference-Desk* (<http://www.technion.ac.il/guides/unix.html>)

## 8.2 Problemy do rozwiązania przed ćwiczeniem

- a) Zapoznać się z użyciem oraz działaniem aplikacji **pkg**. Pomocna może być pozycja wyszczególniona w literaturze jako [1].
- b) Uzupełnić swoją wiedzę dotyczącą podstaw obsługi systemu operacyjnego FreeBSD, korzystając z dostępnej literatury oraz z dokumentacji dostępnej w sieci Internet [1].
- c) Korzystając z dokumentacji systemu FreeBSD, publikacji ogólnodostępnych oraz udostępnionych na Platformie Zdalnej Edukacji pod adresem <http://platforma.polsl.pl/rau3> uzupełnić swoją wiedzę z zakresu tematyki niniejszego ćwiczenia w stopniu umożliwiającym jego realizację.

## 8.3 Program ćwiczenia

**Uwaga:** Większość działań w tym ćwiczeniu laboratoryjnym wykonuje się z uprawnieniami użytkownika o największych uprawnieniach. W związku z tym proszę zachować szczególną ostrożność wykonując jakiegokolwiek operacje na serwerze. W przeciwnym wypadku może dojść do uszkodzenia serwera.

### 8.3.1 Aktualizacja systemu operacyjnego FreeBSD

- a) Dokonać aktualizacji obecnie używanej wersji systemu operacyjnego FreeBSD do wersji uwzględniającej poprawki.
- b) Dokonać aktualizacji do najnowszej wersji dystrybucyjnej systemu operacyjnego FreeBSD.
- c) Dokonać aktualizacji do najnowszej wersji dystrybucyjnej systemu operacyjnego FreeBSD uwzględniającej poprawki.
- d) Dokonać synchronizacji wersji źródłowej systemu FreeBSD do najnowszej wersji stabilnej.
- e) Dokonać synchronizacji oraz aktualizacji archiwum dostępnego oprogramowania w wersji źródłowej (ports).
- f) Przeprowadzić aktualizację i kompilację dokumentacji.
- g) Dokonać próbnej kompilacji zaktualizowanej wersji źródłowej systemu oraz jądra systemowego.
- h) Po konsultacji z prowadzącym dokonać aktualizacji binarnej wersji systemu operacyjnego na podstawie skompilowanej wersji źródłowej systemu operacyjnego.

### 8.3.2 Aktualizacja zainstalowanego oprogramowania

- a) Wyświetlić na ekranie listę wszystkich aplikacji zainstalowanych w systemie FreeBSD. Policzyć ile jest tych aplikacji. Należy do tego celu użyć polecenia `wc`.
- b) Wyświetlić na ekranie listę zainstalowanych aplikacji nie wymagających aktualizacji w systemie FreeBSD. Policzyć ile jest tych aplikacji.



- c) Wyświetlić na ekranie listę zainstalowanych aplikacji wymagających aktualizacji w systemie FreeBSD. Policzyc ile jest tych aplikacji.
- d) Przeprowadzić aktualizację wybranej wraz z prowadzącym aplikacji z listy z podpunktu c). Każdorazowo podać nazwę aplikacji prowadzącemu i zanotować ją w protokole z zajęć laboratoryjnych. Aktualizacji można dokonać trzema metodami:
  - (a) aktualizacja aplikacji w wersji binarnej,
  - (b) aktualizacja aplikacji w wersji źródłowej,
  - (c) aktualizacja aplikacji za pomocą programu `pkg`.

Po dokonaniu aktualizacji aplikacji sprawdzić czy znajduje się na liście aplikacji zainstalowanych w systemie FreeBSD oraz czy aplikacja została zainstalowana w wersji najnowszej (czyli takiej która nie wymaga aktualizacji). Jeśli aplikacja nie znajduje się na liście, lub jest nieaktualna to oznacza, że coś poszło nie tak, i trzeba cały proces aktualizacji aplikacji powtórzyć.

- e) Przeprowadzić aktualizację pozostałych aplikacji wykorzystując do tego celu polecenie `pkg upgrade`.

### 8.3.3 Kompilacja jądra systemowego z wykorzystaniem plików źródłowych systemu operacyjnego

- a) Zapoznać się z zawartością pliku konfiguracyjnego jądra systemowego dostarczonego wraz z wersją dystrybucyjną systemu operacyjnego FreeBSD o nazwie `GENERIC`.
- b) Skompilować jądro systemowe `GENERIC`.
- c) Stworzyć własny plik konfiguracyjny jądra systemowego o nazwie `MYKERNEL` oparty na pliku `GENERIC` dodając do niego kilka dodatkowych opcji:
  - `QUOTA` - uaktywnienie ograniczenia przestrzeni dyskowej dla użytkowników.
  - `IPFIREWALL` oraz `IPFIREWALL_VERBOSE` - uaktywnienie usługi filtrowania pakietów firewall.
  - `DUMMYNET` - uaktywnienie usługi ograniczenia przepustowości bitowej wybranych usług sieciowych.
  - `IPX` - uaktywnienie obsługi protokołu IPX.

- NTFS - uaktywnienie obsługi systemu plików NTFS.
  - EXT2FS - uaktywnienie obsługi systemu plików systemu Linux.
- d) Dokonać kompilacji jądra nowej wersji jądra systemowego.
- e) Dokonać aktualizacji jądra systemowego do aktualnej wersji STABLE.
- f) Dokonać aktualizacji jądra systemowego do aktualnej wersji CURRENT.

### 8.3.4 Instalacja dodatkowego oprogramowania

- a) Zainstalować za pomocą wersji źródłowej następującą aplikację wskazaną przez prowadzącego zajęcia.
- b) Zainstalować za pomocą wersji binarnej następującą aplikację wskazaną przez prowadzącego zajęcia.
- c) Po konsultacji z prowadzącym przeprowadzić deinstalację zainstalowanych w tym zadaniu aplikacji.

### 8.3.5 Rewizja dzienników systemowych serwera FreeBSD

- a) Zapoznać się z zawartością plików dzienników systemowych serwera FreeBSD zawartych w katalogu `/var/log`.
- b) Spróbować określić przeznaczenie poszczególnych plików w katalogu `/var/log`.
- c) Zapoznać się z zawartością plików `/etc/syslog.conf` oraz `/etc/newsyslog.conf`. Spróbować określić przeznaczenie tych plików.

### 8.3.6 Rewizja plików konfiguracyjnych serwera FreeBSD

- a) Zapoznać się z zawartością katalogu pliku `/etc` i zawartych tam plików.
- b) Zapoznać się z zawartością pliku `/etc/defaults/rc.conf` określającą domyślne parametry konfiguracji startowej serwera FreeBSD.
- c) Zapoznać się z zawartością pliku `/etc/rc.conf` określającą aktualną konfigurację startową serwera FreeBSD.
- d) Zapoznać się z zawartością kartoteki `/usr/local/etc` i zawartych tam plików.

Uwaga: Proszę pamiętać aby bez wyraźnej zgody prowadzącego nie modyfikować żadnych plików konfiguracyjnych serwera.

## 8.4 Zadania dodatkowe

### 8.4.1 Cykliczne wykonywanie zadań

Za obsługę cyklicznego wykonywania poleceń odpowiedzialna jest usługa o nazwie **cron**. W systemach UNIXowych, w tym także w FreeBSD usługi nazywane są demonami. Główny plik konfiguracyjny tej usługi to `/etc/crontab`.

Przeanalizuj składnię pliku `/etc/crontab` i na podstawie znajdujących się tam przykładów dopisz linię, dzięki której czas będzie co 15 minut synchronizowany z serwerem czasu. Czyli dodaj linię uruchamiającą jako użytkownik **root** polecenie:

```
ntpdate -u 0.pl.pool.ntp.org 1.pl.pool.ntp.org
```

*Uwaga: Zdecydowanie lepszym rozwiązaniem w kwestii synchronizacji czasu jest odpowiednie skonfigurowanie i uruchomienie demona **ntpd**. Niestety program ten w obecnej chwili wymaga pełnego dostępu do porty 123 na protokole UDP, co bywa blokowane przez dostawców Internetu. Program **ntpdate** omija tę blokadę po użyciu przełącznika **-u**, ale wymaga osobnego mechanizmu zapewniającego cykliczne korekty czasu.*

### 8.4.2 Demon internetowy: **inetd**

Usługi dostępne w systemie FreeBSD mogą być uruchamiane niezależnie, ale można też uruchomić pojedynczego demona o nazwie **inetd**, który będzie pozostałe usługi uruchamiał w razie potrzeby, czyli przy próbie nawiązania połączenia. Przykład konfiguracji zostanie przedstawiony na bazie demona **sshd** obsługującego zdalny dostęp poprzez SSH. Aby przełączyć obsługę zapytań połączeń SSH do demona **inetd** należy:

- a) Zatrzymać demona **sshd** (`service sshd stop`) i wyłączyć jego aktywację przy starcie systemu przez modyfikację (wpisując **NO**) lub usunięcie linii `sshd_enable="YES"` z pliku `/etc/rc.conf`.
- b) Dopisać do pliku `/etc/inetd.conf` (w praktyce zmodyfikować usuwając komentarz) linię aktywującą **sshd** w razie potrzeby:

```
ssh stream tcp nowait root /usr/sbin/sshd sshd -i -4
```
- c) Aktywować demona **inetd** przez dopisanie linii `inetd_enable="YES"` do pliku `/etc/rc.conf` a następnie uruchomieniu usługi poleceniem `service inetd start`.

Przełącz obsługę protokołu SSH do demona internetowego `inetd` wykonując powyższą procedurę.

### 8.4.3 Instalacja środowiska graficznego

FreeBSD obsługuje wiele środowisk graficznych znanych z systemu Linux, w tym najpopularniejsze: GNOME, KDE czy Xfce. Instrukcję dotyczącą instalacji tych środowisk można znaleźć w dokumentacji systemu, dostępnej pod adresem: <https://www.freebsd.org/doc/handbook/x11-wm.html>.

W ramach zajęć zainstalujemy w systemie środowisko graficzne o nazwie MATE, które wywodzi się bezpośrednio z GNOME2 i powstało jako alternatywa dla GNOME3, które ze względu na swoje wymagania może się okazać zbyt obciążające dla niektórych maszyn.

Najnowszą wersję aplikacji można zainstalować z portów, jednak aby zaoszczędzić chwilę czasu skorzystamy z prekompilowanych pakietów binarnych.

- a) Instalujemy środowisko MATE: `pkg install mate`. Akceptujemy komunikat systemowy i instalujemy wymagane zależności.
- b) Mimo długiej listy zależności instalacja taka jeszcze nie jest kompletna. Trzeba jeszcze doinstalować pakiet `xorg` podobnie jak wcześniej było instalowane `mate`.
- c) Do pliku `/etc/rc.conf` należy dopisać:

```
moused_enable="YES"
dbus_enable="YES"
```

a następnie uruchomić/zrestartować aktywowane usługi używając polecenia: `service nazwa_uslugi start/restart`.

- d) Testujemy, czy środowisko zainstalowało się poprawnie wpisując polecenie: `xinit mate-session`. Powinno się uruchomić środowisko MATE w języku polskim. Zamykamy okienko `login` co powoduje powrót do konsoli tekstowej.

**Zgodnie z powyższą instrukcją zainstaluj pakiety MATE i Xorg oraz sprawdź poprawność ich działania.**

#### 8.4.4 Wybór środowiska graficznego i graficzny menadżer logowania

- a) Do uruchomienia środowiska graficznego używane jest polecenie `startx`, po zalogowaniu w konsoli tekstowej. Kiedy jednak w systemie jest zainstalowanych kilka menadżerów okien, to należy najpierw ustawić domyślny menadżer dla użytkownika. Edytujemy plik `.xinitrc` w katalogu użytkownika (jeżeli takiego nie ma, to należy go utworzyć) i wpisujemy tam linię `exec mate-session`. Po wpisaniu `startx` uruchomi się zainstalowane wcześniej MATE.
- b) Środowisko jest po polsku, ale domyślnie ładowany jest tylko angielski układ klawiatury. Polski układ klawiatury można aktywować korzystając z menu MATE wybierając: *System* → *Preferencje* → *Klawiatura*.
- c) Za jeden z najbardziej rozbudowanych graficznych menadżerów logowania można uznać program GDM. Jego instalacja pociągnie za sobą długą listę zależności, która w praktyce zakończy się instalacją środowiska GNOME3 (operacja ta chwilę trwa i zmniejsza dostępną przestrzeń dyskową). W przypadku ograniczonych zasobów dobrym rozwiązaniem jest menadżer o nazwie slim. Instalujemy więc wpisując `pkg install slim`. Aby menadżer uruchamiał się na starcie systemu dopisujemy linię `slim_enable="YES"` do `/etc/rc.conf`.

**Wprowadź opisane zmiany w konfiguracji i zrestartuj system. Oczekiwanym wynikiem jest start graficznego menadżera logowania po ponownym uruchomieniu.**

#### 8.4.5 Instalacja aplikacji z portów

Użytecznym narzędziem w każdym systemie jest przeglądarka internetowa. Dla systemu FreeBSD dostępny jest przykładowo firefox. Instalacja przebiega standardowo: `pkg install firefox`. Warto zwrócić uwagę na komunikaty jakie pojawiają się na ekranie po instalacji pakietu. W przypadku aplikacji firefox można zauważyć informację, że aby poprawnie działały strony w HTML5, to należy załadować moduł `sem.ko`. W trakcie działania systemu można to zrobić poleceniem `kldload sem`. Natomiast dopisanie linii `sem_load="YES"` do pliku `/boot/loader.conf` powoduje, że moduł ten będzie ładowany za każdym razem przy starcie systemu.

Po wylogowaniu i ponownym zalogowaniu z MATE znajdziemy *Przeglądarkę WWW Firefox* w menu *Programy*. Nietrudno zauważyć, że mimo spolszczenia środowiska program firefox jest ciągle po angielsku. Aby mieć polską

wersję przeglądarki należy zainstalować paczkę o nazwie `firefox-i18n`. Jednak instalacja tej paczki z użyciem `pkg` powoduje, że instalowane są właściwie wszystkie obsługiwane języki. Kiedy wystarczający jest język polski trzeba skorzystać z portów i skompilować oraz zainstalować paczkę z wybranymi opcjami:

- a) Najpierw aktualizujemy porty dostępne w systemie poleceniami:

```
portsnap fetch extract
cd /usr/ports
make update
```

- b) Następnie instalujemy paczkę `firefox-i18n` kolejno wykonując poniższe polecenia:

- szukamy paczki w systemie plików:

```
# whereis firefox-i18n
firefox-i18n: /usr/ports/www/firefox-i18n
```

- przechodzimy do tego katalogu używając `mc` lub polecenia:

```
# cd /usr/ports/www/firefox-i18n
```

- konfigurujemy opcje kompilacji wybierając interesujące nas języki po wydaniu polecenia:

```
# make config
```

- kompilujemy i instalujemy port:

```
# make install clean
```

- c) Uruchamiamy ponownie program Firefox i zmieniamy język aplikacji: *Tools* → *Quick Locale Switcher* → *pl-PL - Polish (Poland)*. Jeżeli taka opcja nie jest widoczna będzie konieczna zmiana ustawień wtyczki *Quick Locale Switcher*.

**Zainstaluj przeglądarkę Firefox i pakiet języka polskiego oraz skonfiguruj program aby używał domyślnie języka polskiego w ramach interfejsu użytkownika.**

*Uwaga: Skompilowane paczki nie zawsze „nadążają” z wersją za portami. W praktyce oznacza, to że podany w przykładzie port `www/firefox-i18n` może być nowszy niż zainstalowany wcześniej pakiet `firefox` i instalacja z*

portów zakończy się niepowodzeniem z powodu niespełnionych zależności. Wtedy zostaje pogodzenie się z brakiem personalizacji i skorzystanie z wersji dostępnej przez **pkg** albo aktualizacja wszystkich pakietów do wersji najnowszej przez kompilację wszystkich portów.

### 8.4.6 Aktualizacja systemu

Kiedy celem jest daleko idąca personalizacja systemu aktualizację można ciągle przeprowadzać przechodząc cały proces aktualizacji źródeł i kompilacji: <https://www.freebsd.org/doc/handbook/synching.html>. Kiedy jednak głównym celem jest instalacja najważniejszych poprawek, a w szczególności poprawek bezpieczeństwa, to najszybciej proces ten można przeprowadzić narzędziem **freebsd-update**. Cały proces właściwie ogranicza się do dwóch poleceń:

```
# freebsd-update fetch
# freebsd-update install
```

Pełny opis narzędzia do aktualizacji systemu, które umożliwia także zmianę wersji (upgrade) można znaleźć w dokumentacji online systemu FreeBSD: <https://www.freebsd.org/doc/handbook/updating-upgrading-freebsdupdate.html>.

**Zaktualizuj system instalując aktualnie dostępne poprawki bezpieczeństwa.**

### 8.4.7 Aktualizacja oprogramowania

Kiedy celem jest aktualizacja wszystkich zainstalowanych aplikacji do najnowszych wersji, to procedura powinna wyglądać następująco:

- a) Aktualizacja drzewa portów. Jeżeli wcześniej drzewo portów zostało pobrane przez program **portsnap**, to aktualizacji można dokonać wydając polecenie **make update** w katalogu **/usr/ports**.
- b) Wyświetlenie listy aplikacji wymagających aktualizacji poleceniem: **pkg version | grep "<"**.
- c) Lokalizując na dysku katalog portu wybranej aplikacji poleceniem: **pkg info nazwa\_portu**. Linia zaczynająca się od słowa **Origin** informuje o lokalizacji wybranego portu w drzewie.

- d) Kompilując wybrany port i wymuszając jego usunięcie i ponowną instalację: `make deinstall reinstall clean`.

Nietrudno zauważyć, że jest to dość pracochłonne, kiedy liczba aplikacji rośnie. Ponadto, ze względu na wzajemne zależności aplikacji aktualizacja jednej aplikacji będzie wymagała przekompilowania i ponownej instalacji innych portów. Można jednak zainstalować i użyć aplikacji o nazwie `portmaster`. Po instalacji programu jednym poleceniem: `portmaster -a` uruchamiamy proces aktualizacji oprogramowania.

Ze względu na czasochłonność aktualizacja z poziomu źródeł wydaje się być niewykonalna w ramach laboratorium. Kiedy godzimy się na używanie aplikacji w wersjach starszych, to najlepiej skorzystać z systemu prekompilowanych paczek. Tutaj procedura jest znacznie prostsza i wymaga tylko wydania dwóch poleceń: `pkg update` a następnie `pkg upgrade`.

**Zaktualizuj drzewo portów. Sprawdź ile aplikacji można zaktualizować. Zainstaluj program portmaster a następnie wydając polecenie `portmaster -a -n` możesz sprawdzić ile faktycznie pakietów trzeba będzie skompilować i zainstalować, aby w pełni uaktualnić system. Przełącznik `-n` powoduje, że cała procedura zwykle używana przy aktualizacji portów jest wykonywana (sprawdzanie zależności, ustawianie parametrów kompilacji), ale porty nie są instalowane i kompilowane. Na koniec podana będzie informacja ile portów wymaga kompilacji.**

**Używając narzędzia `pkg` zaktualizuj zainstalowane aplikacje do najnowszych wersji dostępnych w wersji skompilowanej.**

*Uwaga: Niestety opisane powyżej procedury nie zawsze są tak proste. Niektóre zmiany w aplikacjach wymagają wykonania dodatkowych czynności, dlatego przed uruchomieniem aktualizacji oprogramowania warto zapoznać się z informacjami w pliku `/usr/ports/UPDATING` dotyczącymi okresu pomiędzy aktualizacjami.*