

Programowanie sieciowe

Instrukcja do laboratorium LAB08

Zadanie 1. Przeanalizować kod programu syslog_test.c, skompilować i uruchomić program. Po uruchomieniu sprawdzić w konfiguracji, gdzie zapisywane są logi demona syslog (plik **/etc/rsyslog.conf** (system Centos udostępniony na plutonie) i ewentualnie plik w katalogu **/etc/rsyslog.d** z wyrazem „default” w nazwie (zwykle w systemie Ubuntu)). Sprawdzić w odpowiednim pliku, czy komunikaty zostały zapisane do pliku.

- a) Dlaczego komunikat o priorytecie LOG_INFO nie został zapisany do pliku? Zmienić program tak, aby komunikaty o priorytecie LOG_INFO były zapisywane do pliku.
- b) Zmienić konfigurację demona rsyslogd tak, aby komunikaty typu LOG_LOCAL7 były logowane do pliku /var/log/local7:
 - Dołożyć odpowiedni wpis w odpowiednim pliku konfiguracyjnym demona syslog: **local7.* /var/log/local7**
 - Stworzyć plik /var/log/local7 z takimi prawami, użytkownikiem i grupą jak pozostałe pliki w katalogu /var/log, które służą do logowania
 - Zrestartować demona rsyslogd: **service rsyslog restart (lub systemctl restart rsyslogd)**
 - **Sprawdzić, czy komunikaty zapisywane są do nowego pliku**

Pomocne komendy: ls -l, touch 'nazwa_pliku', chown user:group file_name, systemctl status rsyslogd, systemctl restart rsyslogd

Zadanie 2. Skompilować program daytimetcpsrvv6.c i uruchomić. Sprawdzić PID i PPID procesu, oraz terminal sterujący procesu serwera komendą ps. Sprawdzić stan gniazd procesu (ss –antp6 , netstat -antp6).

Plik daytimetcpsrvv6_daemon.c jest programem, który umożliwia otwarcie gniazda nasłuchującego z poziomu administratora systemu, natomiast połączenia przychodzące obsługiwane są z poziomu innego użytkownika. Skompilować program, uruchomić z poziomu administratora, sprawdzić PID procesu, właściciela i terminal sterujący. Sprawdzić stan gniazda nasłuchującego (netstat -t6nap). Porównać wyniki z uzyskanymi dla programu daytimetcpsrvv6.c. Jak to się stało, że właścicielem portu uprzywilejowanego nie jest użytkownik root (jaka funkcja odpowiada za tą funkcjonalność)? Połączyć się klientem daytimetcpccli_names.c z serwerem np. :

```
gcc daytimetcpccli_name.c -o daytimetcpccli_name (lub make)  
./daytimetcpccli_name localhost daytime
```

Sprawdzić logi serwera syslog.

- a) Zmienić kod programu w taki sposób, aby wyeliminować funkcje printf() i fprintf(), zastępując je logowaniem do demona syslog.

- b) W funkcji daemon_init() zastąpić funkcję chdir() na chroot() - ścieżkę zmienić na katalog '/tmp'. Sprawdzić z poziomu programu zawartość katalogu '/' dla procesu przed i po wywołaniu funkcji chdir() i chroot() z wykorzystaniem logowania do demona syslog. Do wylistowania katalogu można wykorzystać kod z programu readdir.c.

Zadanie 3. Sprawdzić jak działają komendy nslookup i host dla serwerów localhost, galaxy.uci.agh.edu.pl i saturn.kt.agh.edu.pl. Zakomentować wpisy 'nameserver' w pliku '/etc/resolv.conf' i sprawdzić, czy komendy nslookup i host działają - symulacja błędnie działającego DNS na komputerze. Komendy nslookup i host powinny zwrócić błąd po czasie oczekiwania.

Następnie w programach host i nslookup podać adres servera DNS z linii komend i w ten sposób ominąć błędную konfigurację DNS (Uwaga na dostępność serwera DNS z danej domeny, zwykle adres 8.8.8 jest dostępny z każdej lokalizacji):

```
host galaxy.uci.agh.edu.pl 149.156.1.3
```

```
nslookup galaxy.uci.agh.edu.pl 149.156.1.3
```

Odkomentować wpisy 'nameserver' w pliku '/etc/resolv.conf'.

Zadanie 4. Przeanalizować kod programu hostent.c. Co jest wynikiem działania programu. Przekompilować i uruchomić program dla ustawionej i nieustawionej zmiennej RES_USE_INET6 (w dowolny sposób pokazany na wykładzie #8). Program uruchomić dla kilku adresów np. localhost, galaxy.uci.agh.edu.pl, saturn.kt.agh.edu.pl. Sprawdzić, jakie adresy są zwracane.

Zakomentować wpisy 'nameserver' w pliku '/etc/resolv.conf' i sprawdzić, czy program hostent.c działa poprawnie. Następnie dołożyć do pliku '/etc/hosts' adresy (niekoniecznie prawdziwe) dla serwerów galaxy.uci.agh.edu.pl oraz saturn.kt.agh.edu.pl i sprawdzić co zwraca program hostent.c.

Odkomentować wpisy 'nameserver' w pliku '/etc/resolv.conf'.

Sprawdzić w man do czego służy plik konfiguracyjny '/etc/host.conf'.

Przerobić serwer daytimetcpsrvv6.c w taki sposób, aby port, na którym nasłuchuje był podawany jako parametr wywołania programu przez nazwę usługi a nie numer portu – rozwiązać zadanie w ten sam sposób jak jest to zrobione w programie daytimetccli_names.c (z użyciem funkcji getservbyname()).

Zadanie 5. Programy udp_echo_serv_getaddrinfo.c i udp_echo_cli_getaddrinfo.c implementują bardzo dobrze znaną usługę ECHO za pomocą protokołu UDP. Przykłady różnią się od dotychczasowych tym, że obsługują zarówno protokół IPv4 jak i IPv6, oraz jako parametry można podawać nazwy serwerów i nazwę usługi (np. echo). W kliencie tekst, który ma być wysłany do serwera jest podawany jako argumenty wywołania programu. W parach:

- a) Przetestować działanie programów dla usługi echo (port 7) używając adresów i nazw komputerów. W pliku /etc/hosts należy dodać adres komputera kolegi pod nazwą

- www.facebook.com. Sprawdzić do jakiej rodziny adresów dołącza się serwer – jeśli do AF_INET, to zmodyfikować strukturę hints w programie, tak aby zwracany był adres z rodziny AF_INET6. Przetestować komunikację dla protokołu IPv4 i IPv6 z adresem www.facebook.com.
- b) Zdefiniować nową usługę echo_ps na porcie 52341 w pliku /etc/services. Przetestować działanie programów dla usługi na nowym porcie.
 - c) Przerobić klienta z zadania 5, tak, aby używał protokołu TCP.

Zadanie 6. Za pomocą programu getaddrinfo.c przetestować wszystkie opcje z tabel na slajdach 61 i 62 z wykładu #8 zmieniając parametry funkcji getaddrinfo() (parametry struktury hints):

- a) Jakie parametry należy przekazać dla funkcji getaddrinfo() dla serwera, który ma nasłuchiwać na porcie dla usługi ECHO, tak aby bezpośrednio można było wykorzystać dane zwarcane przez funkcję getaddrinfo() w funkcji socket() i bind()?
- b) Jakie parametry należy przekazać dla funkcji getaddrinfo() dla klienta, który ma połączyć się na porcie dla usługi ECHO z serwerem, tak aby bezpośrednio można było wykorzystać dane zwarcane przez funkcję getaddrinfo() w funkcji socket() i connect()?
- c) Jakie parametry należy przekazać dla funkcji getaddrinfo() dla klienta, który ma połączyć się na porcie dla usługi ECHO z serwerem na adresie loopback, tak aby bezpośrednio można było wykorzystać dane zwarcane przez funkcję getaddrinfo() w funkcji socket() i connect()?

Pytania sprawdzające:

1. Jakie podstawowe właściwości powinien mieć proces, który ma działać, jako demon w systemie Unix?
2. Dlaczego od odłączenia się procesu od terminala sterującego wymagane jest wywołanie dwóch funkcji fork()?
3. Dlaczego podczas tworzenia procesu demona należy zignorować (obsłużyć) sygnał SIGHUP?
4. Dlaczego podczas tworzenia procesu demona należy przekierować stdin, stdout i stderr do /dev/null?
5. Do czego służy demon syslog?
6. Do czego służą parametry "level" i "facility" w API do sysloga?
7. W jaki sposób w programie z poziomu API do demona syslog można wpływać na miejsce, w którym zapisywane są logi?
8. W jaki sposób administrator systemu może wpływać na miejsce, w którym zapisywane są logi demona syslog?
9. Co jest przechowywane w plikach /etc/hosts, /etc/networks, /etc/services, /etc/protocols?
10. Do czego służą pliki /etc/host.conf i /etc/resolv.conf?
11. Do czego służą funkcje gethostbyname() i gethostbyname2() i czym się różnią?

12. Do czego służy opcja RES_USE_INET6 ?
13. Do czego służą funkcje gethostbyaddr() i getnameinfo()?
14. Do czego służą i jakie parametry przechowują struktury hostent i servent?
15. Jakie rekordy z usługi DNS pobierają funkcje gethostbyname(), gethostbyaddr(), getaddrinfo() i getnameinfo()?
16. Które funkcje w jednym wywołaniu mogą zwrócić jednocześnie rekordy A i AAAA z usługi DNS?
17. Które funkcje przekazują rekordy PTR z usługi DNS?
18. Jakie parametry należy przekazać dla funkcji getaddrinfo() dla serwera, który ma nasłuchiwać na danym porcie, tak aby bezpośrednio można było wykorzystać parametry zwracane przez funkcję getaddrinfo() w funkcji socket() i bind()?
19. Jakie parametry należy przekazać dla funkcji getaddrinfo() dla klienta, który ma połączyć się na danym porcie z serwerem, tak aby bezpośrednio można było wykorzystać dane zwracane przez funkcję getaddrinfo() w funkcji socket() i connect()?
20. Jakie parametry należy przekazać dla funkcji getaddrinfo() dla klienta, który ma połączyć się na danym porcie z serwerem na adresie loopback, tak aby bezpośrednio można było wykorzystać dane zwracane przez funkcję getaddrinfo() w funkcji socket() i connect()?