

Sprawozdanie:

12. Programowanie sieciowe 3

Dominik Bober 303099

3 czerwca 2020

1 Funkcje dotyczące stacji

1.1 gethostent

1. Przejrzyj manual do funkcji set/get/endhostent.

- Jaki plik przeszukują funkcje set/get/endhostent?

`/etc/hosts`

- Co oznaczają poszczególne pola struktury hostent?

`h_name` - nazwa hosta

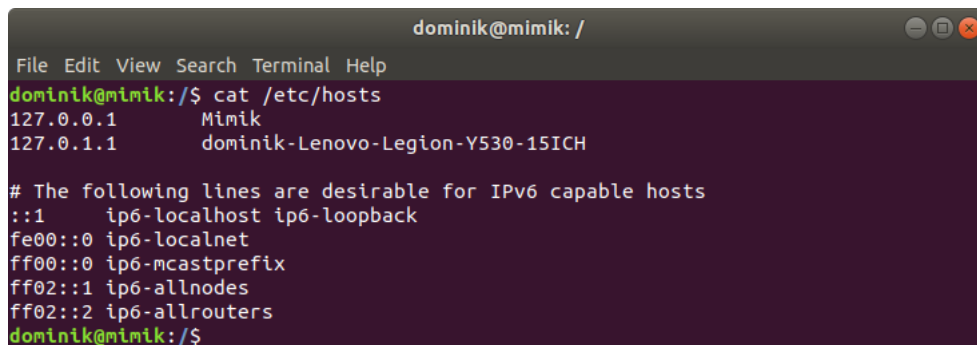
`h_aliases` tablica alternatywnych nazw

`h_addrtype` typ zwratanego adresu

`h_length` - długość adresu w bajtach

`h_addr_list` lista adresów hostów

2. Porównaj jego wynik z zawartością pliku `/etc/hosts`.



```
dominik@mimik: /
File Edit View Search Terminal Help
dominik@mimik:/$ cat /etc/hosts
127.0.0.1      Mimik
127.0.1.1      dominik-Lenovo-Legion-Y530-15ICH

# The following lines are desirable for IPv6 capable hosts
::1          ip6-localhost ip6-loopback
fe00::0      ip6-localnet
ff00::0      ip6-mcastprefix
ff02::1      ip6-allnodes
ff02::2      ip6-allrouters
dominik@mimik:/$ _
```

Rysunek 1: `etc/hosts`

```
dominik@mimik: ~/Projects/c_so_projects/lab12
File Edit View Search Terminal Help
dominik@mimik:~/Projects/c_so_projects/lab12$ ./hostinfo
name: Mimik; type: 2; len: 4
127.0.0.1
name: dominik-Lenovo-Legion-Y530-15ICH; type: 2; len: 4
127.0.1.1
name: ip6-localhost; type: 2; len: 4
ip6-loopback
127.0.0.1
dominik@mimik:~/Projects/c_so_projects/lab12$ _
```

Rysunek 2: hostinfo

1.2 gethostbyname

```
dominik@mimik: ~/Projects/c_so_projects/lab12
File Edit View Search Terminal Help
dominik@mimik:~/Projects/c_so_projects/lab12$ ./gethostbyname-demo google.com
Host name: google.com
Aliases:
Address type: IPv4
Address length: 4 bytes
Addresses: 216.58.208.206

dominik@mimik:~/Projects/c_so_projects/lab12$ ./gethostbyname-demo aaa
gethostbyname error: Unknown host
dominik@mimik:~/Projects/c_so_projects/lab12$ _
```

Rysunek 3: gethostbyname

1.3 getaddrinfo

1. Przeczytać manual do funkcji getaddrinfo

- Jakie parametry przyjmuje funkcja i jakie wartości zwraca?

```
int getaddrinfo(const char *node, const char *service,
               const struct addrinfo *hints,
               struct addrinfo **res);
```

Zwraca 0 w przypadku sukcesu, wszystkie inne liczby sugerują błąd.

- Który parametr funkcji może jako wartość przyjąć http lub ftp lub telnet lub smtp?

```
const char *service
```

- W jakiej formie może być podana wartość parametru node?

Jako naswa hosta, lub numerycznie(w różnej zależności od ipv4 i ipv6).

- Jakiej funkcji używa się do obsługi błędów funkcji getaddrinfo?

```
const char *gai_strerror(int errcode);
```

- Do czego służy flaga AI_PASSIVE?

Od niej zależy czy adres sieci będzie ustalony.

- W jakim pliku można sprawdzić usługi i ich porty na danej maszynie?

```
\etc\services
```

2. Zwrócić uwagę na opis struktury: struct addrinfo

- Co oznaczają poszczególne pola struktury i jakie wartości mogą przyjmować?

int	ai_flags;	- dodatkowe opcje
int	ai_family;	- rodzina adresów
int	ai_socktype;	- typ socketu
int	ai_protocol;	- protokół
size_t	ai_addrlen;	- długość adresu
struct sockaddr	*ai_addr;	- adres
char	*ai_canonname;	- nazwa
struct addrinfo	*ai_next;	- wskaźnik na następną strukturę

- Jaki jest sens wprowadzenia pola ai_next w strukturze?

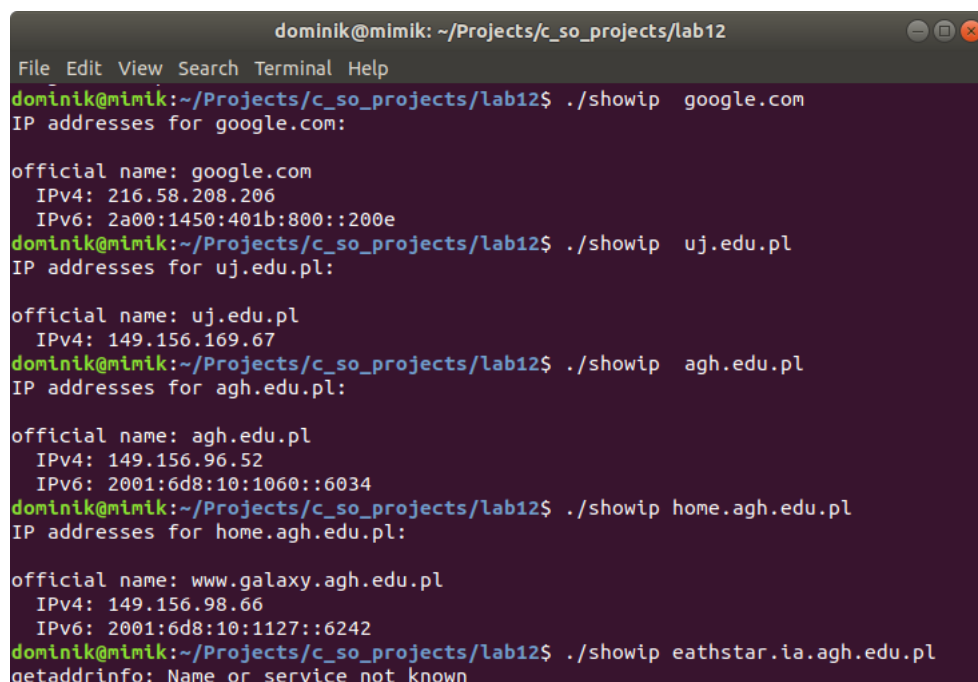
Storzenie linked list.

- Czym się różnią rodziny adresów AF_UNIX, AF_INET, AF_INET6?

AF_UNIX - lokalna komunikacja

AF_INET, AF_INET6- komunikacja lokalna i zdalna

3. showip



```
dominik@mimik: ~/Projects/c_so_projects/lab12
File Edit View Search Terminal Help
dominik@mimik:~/Projects/c_so_projects/lab12$ ./showip google.com
IP addresses for google.com:

official name: google.com
IPv4: 216.58.208.206
IPv6: 2a00:1450:401b:800::200e
dominik@mimik:~/Projects/c_so_projects/lab12$ ./showip uj.edu.pl
IP addresses for uj.edu.pl:

official name: uj.edu.pl
IPv4: 149.156.169.67
dominik@mimik:~/Projects/c_so_projects/lab12$ ./showip agh.edu.pl
IP addresses for agh.edu.pl:

official name: agh.edu.pl
IPv4: 149.156.96.52
IPv6: 2001:6d8:10:1060::6034
dominik@mimik:~/Projects/c_so_projects/lab12$ ./showip home.agh.edu.pl
IP addresses for home.agh.edu.pl:

official name: www.galaxy.agh.edu.pl
IPv4: 149.156.98.66
IPv6: 2001:6d8:10:1127::6242
dominik@mimik:~/Projects/c_so_projects/lab12$ ./showip eathstar.ia.agh.edu.pl
getaddrinfo: Name or service not known
```

Rysunek 4: showip

2 Funkcje dotyczące sieci

Jaki plik przeszukują funkcje set/get/endnetent?

/etc/networks

3 Funkcje dotyczące usług

Jaki plik przeszukują funkcje set/get/endservent?

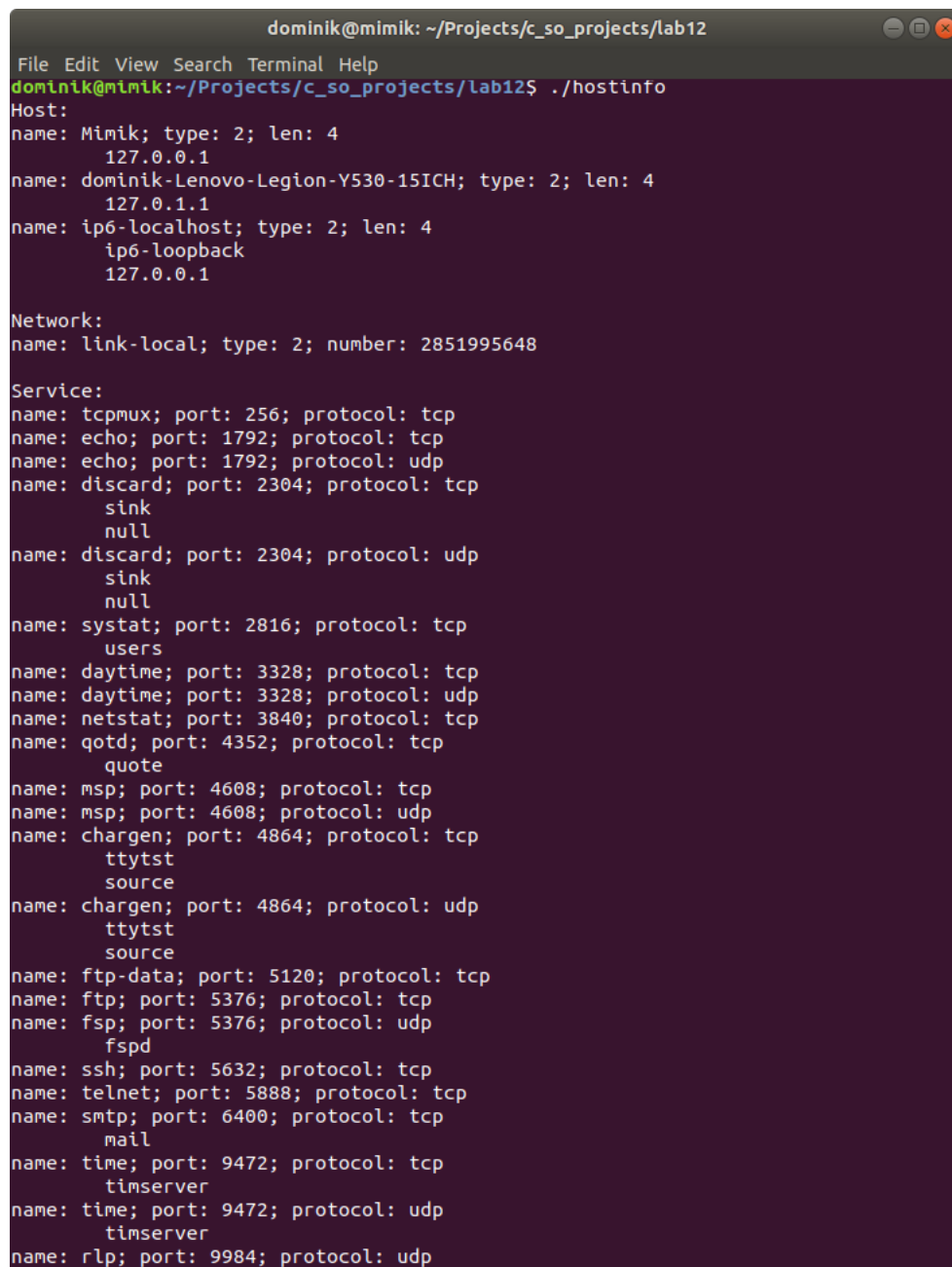
/etc/services

4 Funkcje dotyczące protokołów

Jaki plik przeszukują funkcje set/get/endprotoent?

/etc/protocols

5 Funkcje interfejsu sieciowego

A screenshot of a terminal window titled "dominik@mimik: ~/Projects/c_so_projects/lab12". The terminal shows the output of the command `./hostinfo`. The output is organized into three sections: Host, Network, and Service. The Host section lists the host name "Mimik" with IP "127.0.0.1", the full system name "dominik-Lenovo-Legion-Y530-15ICH" with IP "127.0.1.1", and the IPv6 localhost "ip6-localhost" with "ip6-loopback" and IP "127.0.0.1". The Network section shows "link-local" with number "2851995648". The Service section lists various system services with their ports and protocols, such as "tcpmux" on port 256 (tcp), "echo" on port 1792 (tcp/udp), "discard" on port 2304 (tcp/udp), "systat" on port 2816 (tcp), "daytime" on port 3328 (tcp/udp), "netstat" on port 3840 (tcp), "qotd" on port 4352 (tcp), "msp" on port 4608 (tcp/udp), "chargen" on port 4864 (tcp/udp), "ftp-data" on port 5120 (tcp), "ftp" on port 5376 (tcp), "fsp" on port 5376 (udp), "ssh" on port 5632 (tcp), "telnet" on port 5888 (tcp), "smtp" on port 6400 (tcp), "time" on port 9472 (tcp/udp), and "rlp" on port 9984 (udp).

```
dominik@mimik: ~/Projects/c_so_projects/lab12
File Edit View Search Terminal Help
dominik@mimik:~/Projects/c_so_projects/lab12$ ./hostinfo
Host:
name: Mimik; type: 2; len: 4
      127.0.0.1
name: dominik-Lenovo-Legion-Y530-15ICH; type: 2; len: 4
      127.0.1.1
name: ip6-localhost; type: 2; len: 4
      ip6-loopback
      127.0.0.1

Network:
name: link-local; type: 2; number: 2851995648

Service:
name: tcpmux; port: 256; protocol: tcp
name: echo; port: 1792; protocol: tcp
name: echo; port: 1792; protocol: udp
name: discard; port: 2304; protocol: tcp
      sink
      null
name: discard; port: 2304; protocol: udp
      sink
      null
name: systat; port: 2816; protocol: tcp
      users
name: daytime; port: 3328; protocol: tcp
name: daytime; port: 3328; protocol: udp
name: netstat; port: 3840; protocol: tcp
name: qotd; port: 4352; protocol: tcp
      quote
name: msp; port: 4608; protocol: tcp
name: msp; port: 4608; protocol: udp
name: chargen; port: 4864; protocol: tcp
      ttytst
      source
name: chargen; port: 4864; protocol: udp
      ttytst
      source
name: ftp-data; port: 5120; protocol: tcp
name: ftp; port: 5376; protocol: tcp
name: fsp; port: 5376; protocol: udp
      fspd
name: ssh; port: 5632; protocol: tcp
name: telnet; port: 5888; protocol: tcp
name: smtp; port: 6400; protocol: tcp
      mail
name: time; port: 9472; protocol: tcp
      timserver
name: time; port: 9472; protocol: udp
      timserver
name: rlp; port: 9984; protocol: udp
```

Rysunek 5: hostinfo z wszystkimi funkcjami