

**Politechnika Krakowska**

Katedra Automatyki i Technik Informatycznych

# Laboratorium Sieci Komputerowych

2010/2011



Usługa DNS

## 1. Wprowadzenie

**DNS**, czyli system nazw domenowych (*Domain Name System*) to system służący do przetwarzania adresów w taki sposób, by były zrozumiałe dla urządzeń. Przykładowo, adres *wiek.pk.edu.pl* zostaje przetłumaczony na 32 bitowy (w wersji IPv4) adres IP: *149.156.138.241*.

Serwer **DNS** domyślnie pracuje na porcie numer 53. Do komunikacji z klientem używa protokołu **UDP**, natomiast do komunikacji z innym serwerem protokołu **TCP**.

### 1.1. Typy serwerów DNS.

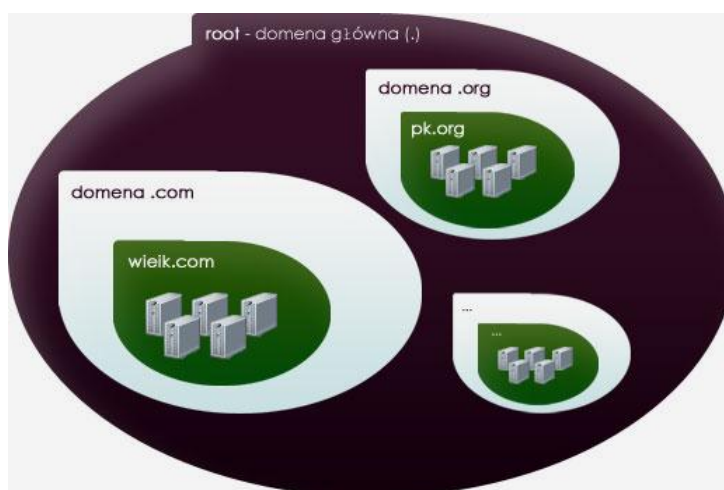
Wyróżniamy dwa typy serwerów DNS:

- 1) **domeny głównej** (*root servers*) – są to główne serwery znajdujące się na samej górze hierarchii serwerów DNS (większość z nich znajduje się na terenie USA). Aktualnie na świecie jest 13 takich serwerów. Ich pełną listę można zobaczyć w załączniku [5].
- 2) **autorytatywne** (*authoritative servers*) – serwerem autorytatywnym dla danej domeny jest serwer, który posiada informacje o niej.

### 1.2. Jak działa w praktyce DNS?

**DNS**, to hierarchiczna sieć serwerów, które przechowują i przesyłają między sobą dane/adresy. Każdy serwer przechowuje określoną część puli adresów. Zwykle każdy serwer (nazywany **Primary Name Server**) posiada co najmniej jeden serwer zastępczy (**Secondary Name Server**), który przechowuje kopię tego pierwszego. Nie trudno się domyśleć, że dzięki temu uzyskujemy większą stabilność działania, gdyż w przypadku awarii serwera głównego, możemy wesprzeć się serwerem zastępczym.

Graficznie można to przedstawić tak:



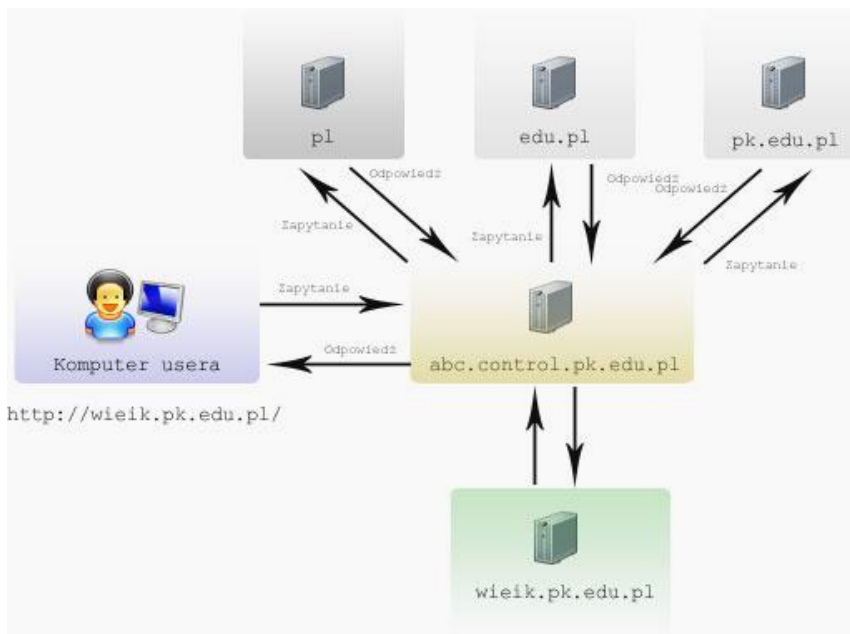
#### Ciekawostka:

Zwykle, gdy podpinamy naszą świeżo kupioną domenę do hostingu, musimy wpisać co najmniej dwa adresy **DNS** naszego hostera. Zwykle te adresy mają następującą postać:

Primary Name Server: `ns1.nazwa-hostingu.pl`

Secondary Name Server: `ns2.nazwa-hostingu.pl`

Zobaczmy działanie systemu **DNS** na przykładzie.



Komputer, znajduje się w domenie **control.pk.edu.pl**. Maszyna **abc.control.pk.edu.pl** jest serwerem nazw.

1. Student chce przeczytać regulamin studiów, w tym celu udaje się na stronę `http://wieik.pk.edu.pl`.
2. Komputer wysyła zapytanie do rekurencyjnego **serwera nazw**. W celu odnalezienia wskazanego adresu.
3. Serwer nazw kieruje zapytanie do nierekurencyjnego serwera domen **PL**.
4. W odpowiedzi dostaje przekierowanie do serwera **edu.pl**.
5. **Serwer nazw** po raz kolejny wysyła zapytanie do serwera domen, tym razem do **edu.pl**.
6. Serwer **edu.pl** jest również nierekurencyjny i zachowuje się tak samo jak **PL**, czyli w odpowiedzi odsyła przekierowanie do kolejnego serwera, tym razem do **pk.edu.pl**.
7. **pk.edu.pl** jest serwerem rekurencyjnym, dlatego w poszukiwaniu szukanej maszyny, przeszuka rekurencyjnie swoją bazę serwerów.
8. **pk.edu.pl** zwraca szukany adres – czyli **wieik.pk.edu.pl**.

**Jaki z tego wniosek?** Serwery **nierekurencyjne (iteracyjne)** nie odpowiadają „wprost”, przekierowują one do innego serwera, który może wiedzieć, gdzie znajduje się szukana maszyna. Natomiast serwery **rekurencyjne**, przeszukują swoją bazę domen, za poszukiwaną maszyną i zwracają wymaganą informację lub błąd.

Wydaje się, że działanie tego systemu może być niewydajne. I tak faktycznie jest. Dlatego z pomocą przychodzi system *cache'owania*, czyli zapisywanie pewnych informacji do pamięci podręcznej, które pozwala przeskoczyć kilka ww. kroków w kolejnych zapytaniach.

W dalszej części zapoznamy się z plikami konfiguracyjnymi, podstawową konfiguracją serwera DNS – BIND oraz z tworzeniem własnych stref.

## 2. Konfiguracja serwera DNS

### 2.1. Krótka o BIND

Jednym z popularniejszych serwerów DNS jest BIND (Berkeley Internet Name Daemon). Został stworzony w 1988 roku przez Paula Vixie. Ze względów bezpieczeństwa serwer BIND zwykle działa w środowisku chroot (Czym jest chroot? [1]) w katalogu `/var/lib/named`.

### 2.2. START, STOP, RESTART i przeładowanie BINDa

Aby włączyć, wyłączyć, zrestartować serwer BIND, możemy użyć polecenia:

```
sieci@komputerowe#: /etc/init.d/bind9 <komenda>
```

W miejsce `<komenda>` wpisujemy **start**, **stop** lub **restart** w zależności co chcemy osiągnąć.

Po zmianach konfiguracji warto przeładować serwer komendą:

```
sieci@komputerowe#: rndc reload
```

### 2.3. Główne pliki konfiguracyjne

W Debianie serwer BIND posiada 3 główne pliki konfiguracyjne:

- `/etc/bind/named.conf`
- `/etc/bind/named.conf.options`
- `/etc/bind/named.conf.local`

#### 2.3.1. `/etc/bind/named.conf`

Jest to główny plik konfiguracyjny. Zobaczmy co w nim siedzi. Aby otworzyć plik `/etc/bind/named.conf` można posłużyć się małym programem *nano* poprzez komendę: `nano /etc/bind/named.conf` Możemy również wypisać zawartość tego pliku przez polecenie: `cat /etc/bind/named.conf`.

```
sieci@komputerowe#: cat /etc/bind/named.conf
```

```
include "/etc/bind/named.conf.options";

zone "." {
    type hint;
    file "/etc/bind/db.root";
};

zone "localhost" {
    type master;
    file "/etc/bind/db.local";
};

zone "127.in-addr.arpa" {
    type master;
    file "/etc/bind/db.127";
};

zone "0.in-addr.arpa" {
    type master;
    file "/etc/bind/db.0";
};

zone "255.in-addr.arpa" {
    type master;
    file "/etc/bind/db.255";
};

include "/etc/bind/named.conf.local";
```

Jak widzimy na początku jest include'owany plik `named.conf.options`, a na końcu `named.conf.local` (pliki te zostały opisane niżej).

Objaśnienie parametrów:

- **zone "localhost"** – nazwa naszej domeny - localhost, którą utrzymujemy
- **type** – typ strefy: **master** – podstawowy; **slave** - zapasowy
- **file** – wskazuje ścieżkę do pliku konfiguracyjnego strefy

Nasza pierwsza domena główna wskazuje na plik `/etc/bind/db.root`, który zawiera adresy 13 głównych serwerów DNS. Są one konieczne, gdy przeszukujemy adresy DNS w poszukiwaniu domen nie utrzymywanych na naszym serwerze.

### 2.3.2. /etc/bind/named.conf.options

Plik ten zawiera podstawowe opcje konfiguracyjne naszego serwera **BIND**.

```
sieci@komputerowe#: cat /etc/bind/named.conf.options
options {
    directory "/var/cache/bind";
    listen-on { any; };
    listen-on-v6 { any; };
```

```
auth-nxdomain no;
query-source address * port 53;
transfer-source * port 53;
notify-source * port 53;
version "Unknown";
};
```

Objaśnienie parametrów:

- **directory** – katalog roboczy **BINDa**
- **listen-on** – adresy IP (wersja IPv4), na które będzie nasłuchiwał **BIND**
- **listen-on-v6** – adresy IP (IPv6), na które będzie nasłuchiwał **BIND**
- **auth-nxdomain** – ustawia, czy odpowiedzi serwera mają być traktowane autoratratywnie.
- **query-source, transfer-source, notify-source** – określają port na którym ma pracuje **BIND**.
- **version** – nazwa serwera, widziana również z zewnątrz.

Całą listę dostępnych parametrów można znaleźć pod odnośnikiem [7].

### 2.3.3. /etc/bind/named.conf.local

W pliku tym umieszczane są informacje o utrzymywanej domenie.

```
sieci@komputerowe#: cat /etc/bind/named.conf.local
zone "naszadomena.pl" {
    type master;
    file "/etc/bind/naszadomena.pl";
    notify yes;
};
```

Objaśnienie parametrów:

- **naszadomena.pl** – nazwa naszej domeny, którą utrzymujemy
- **type** – typ strefy: **master** – podstawowy; **slave** - zapasowy
- **file** – wskazuje ścieżkę do pliku konfiguracyjnego strefy
- **notify** – ustawia, czy zapasowe serwery mają być informowane o zmianach w strefie. Przyspiesza to zwykle aktualizowanie informacji o strefach.
- **allow-transfer { adres-ip;}** - adres serwera, który może transferować całą domenę. Kilka adresów umieszczamy w klamrach i oddzielamy je średnikami.

## 2.4. Pliki stref

Pliki stref dla domen zwykle umieszczane są w katalogu `/etc/bind`. Przykładowy plik stref powinien mieć następującą strukturę:

```
$TTL 604800
$ORIGIN domena.pl.

@      IN      SOA     ns1.domena.pl. root.domena.pl. (
        2010030301 ; Serial
        43200      ; Refresh
        7200       ; Retry
        2419200   ; Expire
        604800    ) ; Negative Cache TTL

@      IN      NS     ns1.domena.pl.
@      IN      NS     ns2.domena.pl.
@      IN      MX     10 mail.domena.pl.

@      IN      A      adres-ip

ns1    IN      A      adres-ip
ns2    IN      A      adres-ip
www    IN      A      adres-ip
mail   IN      A      adres-ip
ftp    IN      CNAME   ftp.domena.pl.
```

Prześledźmy krok po kroku cały listing:

**\$TTL 604800** – jest to czas przez jaki informacje o domenie są buforowane na serwerach DNS.

**\$ORIGIN** – jest to nasza utrzymywana domena.

**@ IN SOA ns1.domena.pl. root.domena.pl.** - rekord typu SOA – *Start Of Authority*. Możemy się z niego dowiedzieć kilku informacji o domenie (np. adres serwera *primary DNS* lub kogoś kto zarządza domeną: [root@domena.pl](mailto:root@domena.pl). Proszę zauważyć, że wyżej został użyty znak kropki „.” zamiast małej kropki). SOA posiada własną strukturę zawartą pomiędzy nawiasami.

**2010030301 ; Serial** – jest to numer seryjny domeny. Powinien być w formacie RRRRMMDDNN (RRRR-rok; MM-miesiąc; DD-dzień; NN-numer modyfikacji). Ostatnie 2 cyfry, czyli numer modyfikacji służący do oznaczenia ilości zmian w danym dniu.

**43200 ; Refresh** – jest to informacja jak często serwery *slave* mają sprawdzać czy dane o domenie nie uległy zmianie. Według dokumentu RFC 1035 [8] wartość ta powinna wynosić pomiędzy 1200-43200 sekund.

**7200 ; Retry** – oznacza częstość prób komunikacji serwera *secondary* z *primary* w przypadku niepowodzeń. Zwykle jest to około 1200-7200 sekund.

**2419200 ; Expire** – wartość ta oznacza po jakim czasie dane domeny mają zostać unieważnione w przypadku, gdy serwer *secondary* nie będzie w stanie skomunikować się z *primary*. Zwykle ta wartość wynosi między 1209600-2419200 sekund (2-4 tygodnie).

**604800** ) ; **Negative Cache TTL** – ostatnią wartością jest TTL (*Time To Live*). Określa ona czas przez który informacja o domenie będzie aktualna, czyli w praktyce przez jaki czas DNS będzie przechowywał informacje o domenie. Zwykle jest to około 86400 a 432000 sekund, czyli 1-5 dni.

**@ IN NS ns1.domena.pl.** - wpis typu NS informuje nas, że domenę *domena.pl* obsługuje serwer DNS *ns1.domena.pl*.

**@ IN MX 10 mail.domena.pl.** - rekord MX służy do definiowania serwera pocztowego dla *domena.pl*, czyli maile dla domeny *domena.pl* będą przekierowywane na *mail.domena.pl*. Wartość 10 oznacza priorytet, ma on znaczenie wtedy, gdy posiadamy więcej niż 1 serwer pocztowy. Służy on, do ustalenia kolejności do którego serwera ma pocztowego ma trafić mail. Mniejsza wartość priorytetu oznacza pierwszeństwo.

**<nazwa> IN A adres-ip** – *adres-ip* wskazuje na **<nazwa>**

**ftp IN CNAME ftp.domena.pl.** - rekord **CNAME** tworzy subdomenę dla hosta przypisanego do innej nazwy.

W powyższym listingu, znak średnika „;” oznacza znak początku komentarza, czyli wszystko co w danej linii znajduje się po jego prawej stronie, zostaje ignorowane. Natomiast „@” oznacza zmienną (choć to nie do końca dobre określenie), która przechowuje adres *domena.pl*.

**Kropki na końcach domen, to nie pomyłka.** Muszą one się tam znajdować. Gdyby ich nie było, serwer potraktowałby tą nazwę jako część domeny utrzymywanej w tej strefie i dokleiłby do niej końcówkę *domena.pl*.

Typy rekordów **DNS**:

**A** – 32-bitowy adres IP w wersji IPv4 na który ma wskazywać dany wpis

**AAAA** – 128-bitowy adres IP w wersji Ipv6, który ma wskazywać na dany wpis

**NS** – nazwy serwerów, które utrzymują daną strefę

**MX** – (*Mail eXchanger*) nazwy serwerów pocztowych

**PTR** – inaczej *pointer record*, czyli **rekord wskaźnika**. Mapuje adresy IP (w wersji IPv4 i IPv6), na nazwę kanoniczną hosta

**SOA** – inaczej *start of authority records*, czyli **rekord adresu startowego uwierzytelnienia**. Ustala on serwer DNS, który dostarcza autorytatywnie [4] informacje o domenie internetowej wraz z jej parametrami (np. czas życia pakietów - TTL)

**SRC** – service record – **rekord usługi** – pozwala na ustawienie dodatkowych lokalizacji danej usługi, którą udostępnia serwer na którego wskazuje adres DNS

**TXT** – pozwala dołączyć dowolny tekst do rekordu DNS

**CNAME** – *canonical name record*, czyli **rekord nazwy kanonicznej**. Ustawia aliasy na inne domeny

Studentów pragnących poszerzyć swoją wiedzę na temat nazw domenowych, odsyłamy do dokumentu [RFC 1035](#).



## 2.5. Odwrotna translacja adresów DNS

Zwykle domeny wskazują na IP. Możemy również mieć do czynienia z sytuacją odwrotną, kiedy IP wskazuje na domenę. Taką sytuację nazywamy **odwrotną translacją adresów DNS**, czyli **Reverse DNS** (czasami nazywany również jako **rDNS**). **rDNS** jest częścią systemu DNS.

**Odwrotny DNS** tworzy się podobnie jak zwykły. Założmy, że chcemy aby adres `123.123.123.123` wskazywał na `wiek.pk.edu.pl`. Wystarczy dodać następujący rekord **PTR**:

```
123.123.123.123.in-addr.arpa      IN      PTR      wiek.pk.edu.pl.
```

**rDNS** dla `123.123.123.123` jest zapisywany w postaci `123.123.123.123.in-addr.arpa`, czyli dokładnie odwrotnie.

### Ciekawostka:

Z **rDNS** mamy do czynienia w sieciach IRC.

## 3. Konfiguracja klienta DNS

Aby umożliwić korzystanie z wybranych serwerów DNS dla innych programów/usług niż **nslookup** wykorzystywany w niniejszym ćwiczeniu należy dodać je do pliku konfiguracyjnego `/etc/resolv.conf`, np.:

```
nameserver 208.67.220.220
nameserver 208.67.222.222
```

Adresy serwerów DNS mogą być również przesyłane w procesie autokonfiguracji węzła z użyciem protokołu DHCP.

## 4. Realizacja ćwiczenia

W ćwiczeniu będziemy używać systemu operacyjnego **Debian**.

Zainstalowano na nim środowisko graficzne **Gnome**, menadżer logowania **KDM**, przeglądarkę internetową **Firefox**, serwer **BIND9** oraz aplikację do tworzenia wirtualnych maszyn **VirtualBox**.

Dodano również skrypty przywracające domyślne ustawienia serwera BIND9 podczas włączania i wyłączenia systemu.

Do systemów można zalogować się za pomocą loginu `student` z hasłem `student`.

Uprawnienia roota możemy otrzymać przez konsolę na dwa sposoby:

- 1) Logując się na konto roota przez wpisanie polecenia **su** i hasła **student**
- 2) Używając programu **sudo**, by wykonać komendę z uprawnieniami roota.

Czyli: **sudo <polecenie>**

Aby sprawdzić adres IP maszyny, należy użyć komendy **ifconfig** lub bardziej złożonej, ale dającej wyłącznie adres IP komendy (zastępując eth0 nazwą używanego interfejsu sieciowego):

```
/sbin/ifconfig eth0 | grep 'inet addr:' | cut -d: -f2 | awk '{print $1}'
```

### 3.1. Wykonanie ćwiczenia

- 1) Znajdź adresy IP hostów control.pk.edu.pl, wieik.pk.edu.pl, sejm.gov.pl za pomocą dig i nslookup. Sprawdź czy domyślny serwer DNS oraz dns.task.gda.pl dają identyczne wyniki.
- 2) Zatrzymaj i uruchom serwer BIND9 za pomocą skryptów uruchomieniowych.
- 3) Uruchom wirtualny system Debian z VirtualBoksa.

Skonfiguruj DNS tak, aby pełnił on funkcję serwera *primary*. W ustawieniach proszę dodać wpis pozwalający transferować strefy tylko systemowi zwirtualizowanemu.

Następnie skonfiguruj DNS w zwirtualizowanym Debianie tak, aby był on *secondary*.

4) Stwórz strefę dla domeny **edu.pl**, by jego Primary DNsem była maszyna macierzysta, a Secondary DNsem system wirtualny. Następnie wynik prześledź w programie **nslookup** i sprawdź wynik komendy: **ping -c 3 pk.edu.pl**

5) Skonfiguruj DNS tak, aby pełnił on funkcję serwera autorytatywnego. Następnie skonfiguruj DNS w zwirtualizowanym Debianie tak, aby był on serwerem cache-ującym.

6) Na "zewnętrznym" systemie zmień port, na którym działa BIND9 na 9999 i skonfiguruj "wewnętrzny" system tak, by ten z niego korzystał.

7) Zmień nadrzędny serwer DNS na dns.task.gda.pl

8)\* Za pomocą odwrotnego DNS "zamień miejscami" portale WP i Onet na "wewnętrznym" systemie.

## 4. Wnioski i zakończenie

W sprawozdaniu proszę zamieścić przebieg wykonywania ćwiczeń i ich rezultat, wnioski oraz jedno własne ćwiczenie praktyczne dotyczące DNS i przebieg jego rozwiązania. Proszę również zawrzeć w nim uwagi dotyczące ćwiczeń i niniejszej instrukcji.

## Dodatek A:

### *Nslookup*

**Nslookup** jest prostym konsolowym narzędziem służącym do odpytywania serwerów DNS. Może pracować w dwóch trybach: interaktywnym i nieinteraktywnym. W trybie interaktywnym program wyświetla znak zachęty i oczekuje na wprowadzenie komend.

Polecenie	Składnia	Opis
host	nazwa_hosta [adres_serwera]	rozwiązuje nazwę hosta, wykorzystując podany adres serwera DNS (jeśli nie podano, nslookup wykorzystuje domyślny serwer)
server	server [domena]	ustawia domyślny serwer DNS
set	set zmienna[=wartość]	ustawia różne opcje programu (opcje opisane są w man nslookup)
exit	Exit	kończy sesję z programem

Tabela 1: wybrane polecenia programu nslookup

Nslookup uruchamia się w trybie interaktywnym, jeśli zostanie wywołany bez argumentów wiersza poleceń lub pierwszym argumentem jest – (pojedynczy znak minusa).

Jeśli pierwszym argumentem jest adres hosta, nslookup uruchamia się w trybie nieinteraktywnym, drukując jedynie uzyskane dane.

Przykładowe sesje z programem:

0) tryb nieinteraktywny:

```
artnowo@z61t:~> nslookup www.control.pk.edu.pl
Server:          192.168.1.1
Address: 192.168.1.1#53

www.control.pk.edu.pl canonical name = control.elektron.pk.edu.pl.
Name:           control.elektron.pk.edu.pl
Address: 149.156.138.61
artnowo@z61t:~>
```

1) tryb interaktywny:

```
artnowo@z61t:~> nslookup
> server dns1.task.gda.pl
Default server: dns1.task.gda.pl
Address: 153.19.250.100#53
```

```

> www.wieik.pk.edu.pl
Server:          dns1.task.gda.pl
Address: 153.19.250.100#53

Non-authoritative answer:
www.wieik.pk.edu.pl canonical name = saturn.wieik.pk.edu.pl.
Name:          saturn.wieik.pk.edu.pl
Address: 149.156.138.241
> exit

artnowo@z61t:~>

```

## Dig

Dig także jest konsolowym narzędziem służącym do odpytywania serwerów DNS. W porównaniu do nslookup ma dużo większe możliwości, prezentuje dużo więcej szczegółów i często jest wykorzystywany przez administratorów sieci do diagnozowania problemów z systemem DNS. Może przyjmować argumenty wiersza poleceń lub działać wsadowo na podstawie zawartości pliku. Dig odpytuje po kolei wszystkie serwery DNS wpisane w pliku `/etc/resolv.conf`, o ile nie wskaże mu się konkretnego.

Program wywołuje się w następujący sposób:

```
dig [@serwer] szukana_nazwa [typ_rekordu] [opcje]
```

Przełącznik	Działanie
<code>+ [no] comments</code>	Włącza [wyłącza] wyświetlanie słownych komentarzy
<code>+ [no] short</code>	[nie]Wyświetla odpowiedź w skróconej formie; przydatne, jeśli wyjście dig ma być wejściem dla np. jakiegoś innego skryptu.
<code>+time=T</code>	Ustawia timeout dla zapytania na T sekund.

Tabela 2: Wybrane opcje programu dig.

Dig obsługuje w domenach znaki spoza zestawu ASCII jeśli został skompilowany ze wsparciem dla międzynarodowych nazw domentowych (ang. IDN – International Domain Names) oraz nie jest ustawiona zmienna środowiskowa `IDN_DISABLE`.

Przykład użycia:

```

artnowo@z61t:~> dig www.control.pk.edu.pl +nostats +question

; <<>> DiG 9.5.0-P2 <<>> www.control.pk.edu.pl +nostats +question
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 49118

```

```
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2, ADDITIONAL: 2

;; QUESTION SECTION:
;www.control.pk.edu.pl.          IN      A

;; ANSWER SECTION:
www.control.pk.edu.pl.80410 IN      CNAME  control.elektron.pk.edu.pl.
control.elektron.pk.edu.pl. 400    IN      A      149.156.138.61

;; AUTHORITY SECTION:
elektron.pk.edu.pl. 400    IN      NS      pobox.pk.edu.pl.
elektron.pk.edu.pl. 400    IN      NS      one.usk.pk.edu.pl.

;; ADDITIONAL SECTION:
pobox.pk.edu.pl.16723 IN      A      149.156.153.2
one.usk.pk.edu.pl. 76730 IN      A      149.156.132.100

artnowo@z61t:~>
```

Istnieje możliwość wykonania kilku zapytań *za jednym zamachem*:

```
artnowo@z61t:~> dig www.control.pk.edu.pl +nocomments +nostats
www.pk.edu.pl +nocomments +nostats

; <<>> DiG 9.5.0-P2 <<>> www.control.pk.edu.pl +nocomments +nostats
www.pk.edu.pl +nocomments +nostats
;; global options: printcmd
;www.control.pk.edu.pl.          IN      A
www.control.pk.edu.pl.85082 IN      CNAME  control.elektron.pk.edu.pl.
control.elektron.pk.edu.pl. 400    IN      A      149.156.138.61
elektron.pk.edu.pl. 400    IN      NS      one.usk.pk.edu.pl.
elektron.pk.edu.pl. 400    IN      NS      pobox.pk.edu.pl.
pobox.pk.edu.pl.64798 IN      A      149.156.153.2
one.usk.pk.edu.pl. 78802 IN      A      149.156.132.100
;www.pk.edu.pl.                IN      A
www.pk.edu.pl. 75642 IN      CNAME  mera.pk.edu.pl.
mera.pk.edu.pl. 56411 IN      A      149.156.132.150
artnowo@z61t:~>
```

## Literatura dodatkowa:

[0] Ron Aitchison, „*Pro DNS and BINDS*”

[1] <http://pl.wikipedia.org/wiki/Chroot> – czym jest chroot?

[2] [http://en.wikipedia.org/wiki/Symbolic\\_link](http://en.wikipedia.org/wiki/Symbolic_link) – dowiązania symboliczne.

[3] [http://pl.wikipedia.org/wiki/Domain\\_Name\\_System](http://pl.wikipedia.org/wiki/Domain_Name_System) – protokół DNS.

[4] <http://pl.wikipedia.org/wiki/UDP> – protokół UDP.

Ćwiczenie: usługa DNS

- [5] <ftp://ftp.rs.internic.net/domain/named.root> – lista 13 głównych serwerów DNS.
- [6] <http://www.sjp.pl/autorytatywnie> – wyjaśnienie słowa „autorytatywnie”.
- [7] <http://www.zytrax.com/books/dns/ch7/statements.html> - wszystkie opcje named.conf.
- [8] <http://www.ietf.org/rfc/rfc1035.txt> – RFC 1035
- [9] man nslookup
- [10] man dig

Instrukcja opracowana przez:

Jarosław Macko  
Artur Nowosielski

Opieka merytoryczna:

dr inż. Piotr Andrzej Kowalski, dr inż. Szymon Łukasik, mgr inż. Sławomir Żak