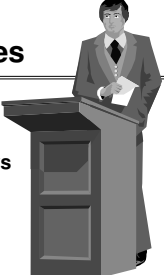


© 2002-2009 Volnys Bernal 1

## Resolução de Nomes

Volnys Borges Bernal  
volnys@lsi.usp.br  
<http://www.lsi.usp.br/~volnys>



© 2002-2009 Volnys Bernal 2


© 2002-2009 Volnys Bernal 3

## Agenda

- ❑ Introdução à resolução de nomes
- ❑ Resolução de nome de host
- ❑ Resolução de nome de serviço
- ❑ Resolução de nome de protocolo

© 2002-2009 Volnys Bernal 4

## Introdução à resolução de nomes



© 2002-2009 Volnys Bernal 5

## Introdução à resolução de nomes

- ❑ Resolução ~ Tradução
- ❑ Programas que lidam com comunicação geralmente necessitam realizar diversos tipos de resoluções de nomes como:
  - ❖ Nome de host ↔ Endereço IP
  - ❖ Nome de serviço ↔ Nro da porta / prot. de transporte
  - ❖ Nome do protocolo ↔ Número do protocolo
- ❑ Existem diversas funções de biblioteca que facilitam a resolução de nomes

© 2002-2009 Volnys Bernal 6

## Introdução à resolução de nomes

- ❑ Funções para resolução de nomes:

Tipo de resolução	Função	Tradução	Exemplo
Host	gethostname ()	nome → endereço IP	www.usp.br → 143.107.10.5
	gethostbyaddr ()	endereço IP → nome	143.107.10.5 → www.usp.br
Serviço	getservbyname ()	nome → porta/PROTO	http → 80/TCP
	getservbyport ()	Porta/PROTO → nome	80/TCP → http
Protocolo	getprotobyname ()	nome → protocolo	UDP → 17
	getprotobynumber ()	protocolo → nome	17 → UDP

© 2002-2009 Volnys Bernal 7

## Introdução à resolução de nomes

Funções para resolução de nomes:

- Nome de host
  - gethostbyname()
  - gethostbyaddr()
- Nome de serviço
  - getservbyname()
  - getservbyport()
- Nome de protocolo
  - getprotobyname()
  - getprotobynumber()

Interface de chamadas ao sistema

Núcleo do Sistema Operacional

© 2002-2009 Volnys Bernal 8

## Introdução à resolução de nomes

Tais funções se encarregam de verificar a configuração do subsistema de resolução de nome e consultar os arquivos ou ativar as requisições necessárias:

- Nome de host:
  - Arquivo /etc/hosts
  - Serviço DNS
  - Serviço NIS
  - Serviço LDAP
- Nome de serviço:
  - Arquivo /etc/services
  - Serviço NIS
  - Serviço LDAP
- Nome de protocolo:
  - Arquivo /etc/protocols
  - Serviço NIS
  - Serviço LDAP

© 2002-2009 Volnys Bernal 9

## Resolução de nome de host

© 2002-2009 Volnys Bernal 10

## Resolução de nome de host

Possibilita traduzir

- Nomes de máquina → endereço IPs
- Endereço IP → nome de máquina

Funções:

- gethostbyname()
  - A partir de um nome de uma máquina possibilita descobrir seu endereço IP
- gethostbyaddr()
  - A partir de um endereço IP de uma máquina possibilita descobrir seu nome

© 2002-2009 Volnys Bernal 11

## Resolução de nome de host

Interface de programação

```

struct hostent *gethostbyname(char *name)
struct hostent *gethostbyaddr(char *addr, int len, int type)

struct hostent
{
    char *h_name;          /* nome oficial do host */
    char **h_aliases;     /* lista de apelidos */
    int h_addrtype;       /* tipo: AF_INET (IPv4), AF_INET6 (IPv6) */
    int h_length;         /* comprimento do endereço: 4 (IPv4) */
    char **h_addr_list;   /* lista de endereços */
}

#define h_addr h_addr_list[0];
    
```

© 2002-2009 Volnys Bernal 12

## Resolução de nome de host

Estrutura hostent

```

struct hostent
{
    h_name      → apolo.xyz.com
    h_aliases   → NULL, www.xyz.com, webmail.xyz.com
    h_addrtype  → AF_INET
    h_length    → 4
    h_addr_list → NULL, 200.156.84.56
}
    
```

© 2002-2009 Volnys Bernal 13

## Resolução de nome de host

```

// Programa resolv-nome.c
#include <stdio.h>
#include <netdb.h>
extern int  errno;

int main()
{
    char      hostname[80];
    struct hostent  * hostentryp;

    printf("Entre com o nome da máquina: ");
    scanf("%s",hostname);
    hostentryp = gethostbyname(hostname);
    if (hostentryp == NULL) /* Erro */
        perror("Erro: ");
    else /* ok */
        printf("Endereco = %d.%d.%d.%d \n",
            ((unsigned char *) hostentryp->h_addr)[0],
            ((unsigned char *) hostentryp->h_addr)[1],
            ((unsigned char *) hostentryp->h_addr)[2],
            ((unsigned char *) hostentryp->h_addr)[3]);
}
    
```

© 2002-2009 Volnys Bernal 14

## Exercício 1


- ❑ **Faça um programa, baseado no programa resolv-nome.c, que realize a tradução reversa, ou seja, dado um endereço IP informe o nome associado.**
- ❑ **Dicas:**
  - ❖ O endereço IP é representado por uma seqüência de 4 bytes;
  - ❖ O endereço IP pode ser requisitado ao usuário utilizando o seguinte comando:
 

```

unsigned char bytesIP[4]; //4 bytes p/ armazenar end. IP
scanf("%u.%u.%u.%u",
    &bytesIP[0], &bytesIP[1], &bytesIP[2], &bytesIP[3]);
                    
```
  - ❖ Na função gethostbyaddr() o parametro type representa o tipo de endereço de rede que está sendo utilizado (IPv4, IPv6, X.25,...). Neste caso, deve ser utilizada a macro AF\_INET (Address Family Internet) que representa endereços tipo IPv4.

© 2002-2009 Volnys Bernal 15

## Resolução de nome de serviço



© 2002-2009 Volnys Bernal 16

## Resolução de nome de serviço

- ❑ **Possibilita traduzir**
  - ❖ Nomes de serviço → número da porta + protocolo
  - ❖ Número de porta + protocolo → Nome de serviço
- ❑ **Funções:**
  - ❖ Getservicebyname()
    - A partir de um nome de serviço permite descobrir a porta do protocolo de transporte
  - ❖ Getservicebyport()
    - A partir de uma porta do protocolo transporte permite descobrir o nome do serviço associado

© 2002-2009 Volnys Bernal 17

## Resolução de nome de serviço

- ❑ **Interface de programação:**

```

struct servent *getservbyname (char *name, char *proto)
struct servent *getservbyport (int port , char *proto)

struct servent
{
    char * s_name; /* nome oficial do serviço */
    char **s_aliases; /* lista de apelidos */
    int s_port; /* nro porta (network byte order)*/
    char *s_proto; /* nome do protocolo (TCP/UDP) */
}
    
```

© 2002-2009 Volnys Bernal 18

## Resolução de nome de serviço

```

// Programa resolv-serv-udp.c
#include <stdio.h>
#include <netdb.h>
extern int  errno;

int main()
{
    char      servicenamep[80];
    struct servent  * serventp;

    printf("Entre com o nome do serviço: ");
    scanf("%s",servicenamep);
    serventp = getservbyname(servicenamep,"udp");
    if (serventp == NULL) /* Erro */
        perror("Erro na chamada getservbyname()");
    else /* ok */
    {
        printf("Nome      = %s\n",serventp->s_name);
        printf("Porta      = %d\n",ntohs(serventp->s_port));
        printf("Protocolo = %s\n",serventp->s_proto);
    }
}
    
```

© 2002-2009 Volnys Bernal 19

## Resolução de nome de serviço

```
// Programa resolv-serv-udp-tcp.c
#include <stdio.h>
#include <netdb.h>
extern int  errno;

int main()
{
    char    servicename[80];
    struct servent * serventp;

    printf("Entre com o nome do servico: ");
    scanf("%s",servicename);
    serventp = getservbyname(servicename,"udp");
    if (serventp != NULL) /* Erro */
    {
        printf("Porta    = %d/UDP\n",ntohs(serventp->s_port));
    }
    serventp = getservbyname(servicename,"tcp");
    if (serventp != NULL) /* Erro */
    {
        printf("Porta    = %d/TCP\n",ntohs(serventp->s_port));
    }
}
```


© 2002-2009 Volnys Bernal 20

## Exercício 2

- ❑ **Faça um programa, baseado no programa resolv-serv-udp.c, que realize a tradução reversa, ou seja, dado um número de porta e um protocolo de transporte informe o nome do serviço associado.**
- ❑ **Dicas:**
  - ❖ O número da porta é geralmente representado com inteiros de 16 bits (*short integer*);
  - ❖ Utilize a função auxiliar `htons()` para traduzir um *short integer* (inteiro de 16 bits) na representação "*host byte order*" para a representação "*network byte order*".

© 2002-2009 Volnys Bernal 21

## Resolução de nome de protocolo



© 2002-2009 Volnys Bernal 22

## Resolução de nomes de protocolo

- ❑ **Possibilita traduzir**
  - ❖ Nome de protocolo → número do protocolo
  - ❖ Número de protocolo → nome de protocolo
- ❑ **Funções:**
  - ❖ `Getprotobyname()`
    - A partir de um nome de um protocolo possibilita descobrir seu número.
  - ❖ `Getprotobynumber()`
    - A partir de um número de um protocolo possibilita descobrir seu nome.

© 2002-2009 Volnys Bernal 23

## Resolução de nomes de protocolo

- ❑ **Interface de programação:**

```
struct protoent *getprotobyname (char *name)
struct protoent *getprotobynumber (int proto)

struct protoent
{
    char *p_name; /* nome oficial do protocolo*/
    char **p_aliases; /* lista de apelidos */
    int p_proto; /* nro protocolo (host byte order)*/
}
```

© 2002-2009 Volnys Bernal 24

## Resolução de nomes de protocolo

```
// Programa resolv-protoc.c
#include <stdio.h>
#include <netdb.h>
extern int  errno;

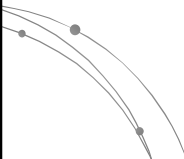
int main()
{
    char    protoname[80];
    struct protoent * protoentp;

    printf("Entre com o nome do protocolo: ");
    scanf("%s",protoname);
    protoentp = getprotobyname(protoname);
    if (protoentp == NULL) /* Erro */
        perror("Erro na chamada getprotobyname()");
    else /* ok */
    {
        printf("Nome      = %s\n",protoentp->p_name);
        printf("Protocolo = %d\n",protoentp->p_proto);
    }
}
```

© 2002-2009 Volnys Bernal 25


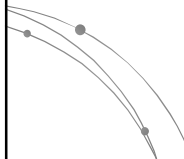
### Exercício 3

- ❑ **Faça um programa, baseado no programa `resolv-proto.c`, que realize a tradução reversa, ou seja, dado um número de protocolo informe o nome do protocolo associado.**



© 2002-2009 Volnys Bernal 26

### Referências Bibliográficas



© 2002-2009 Volnys Bernal 27

### Referências Bibliográficas

- ❑ **COMMER, DOUGLAS; STEVENS, DAVID**
  - ❖ Internetworking with TCP/IP: volume 3: client-server programming and applications
  - ❖ Prentice Hall
  - ❖ 1993

