

Domain Name Service

1. Domain Name Service

1.1 CSCI 330

CSCI 330 UNIX and Network Programming



Domain Name Service

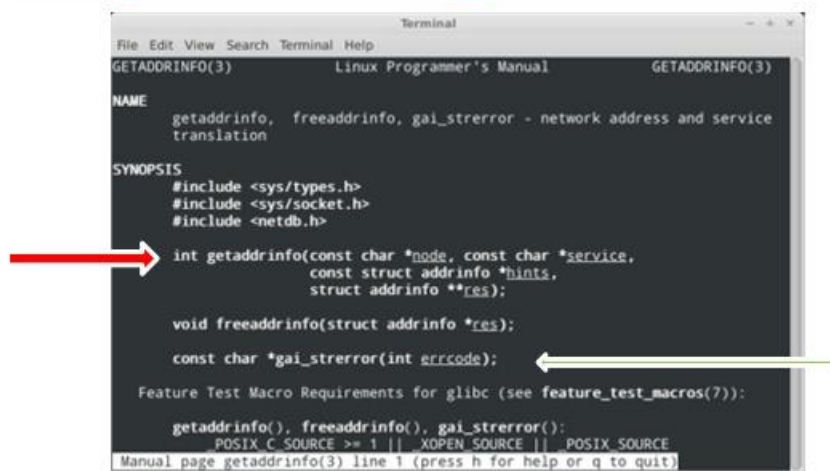
1.2 Domain Names

Domain Names

- hierarchical distributed naming system
- uses FQDN: fully qualified domain name
 - ex: faculty.cs.niu.edu
- DNS: domain name service
 - resolves query for FQDN into IP address
 - ex.: 131.156.145.186

1.3 Library Function: getaddrinfo

Library Function: getaddrinfo



```
Terminal
File Edit View Search Terminal Help
GETADDRINFO(3)      Linux Programmer's Manual      GETADDRINFO(3)

NAME
    getaddrinfo, freeaddrinfo, gai_strerror - network address and service
    translation

SYNOPSIS
    #include <sys/types.h>
    #include <sys/socket.h>
    #include <netdb.h>

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

    void freeaddrinfo(struct addrinfo *res);

    const char *gai_strerror(int errcode);

Feature Test Macro Requirements for glibc (see feature_test_macros(7)):

    getaddrinfo(), freeaddrinfo(), gai_strerror():
        _POSIX_C_SOURCE >= 1 || _XOPEN_SOURCE || _POSIX_SOURCE
Manual page getaddrinfo(3) line 1 (press h for help or q to quit)
```

1.4 Library Function: getaddrinfo

Library Function: getaddrinfo

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

- translates FQDN `node` into IP address
- `res` is pointer to list of address info structures
- `service` and `hints` can be NULL

1.5 Address info structure

Address info structure

```
struct addrinfo {
    int             ai_flags;
    int             ai_family;
    int             ai_socktype;
    int             ai_protocol;
    size_t          ai_addrlen;
    struct sockaddr *ai_addr;      // socket address
    char            *ai_canonname;
    struct addrinfo *ai_next;
};
```

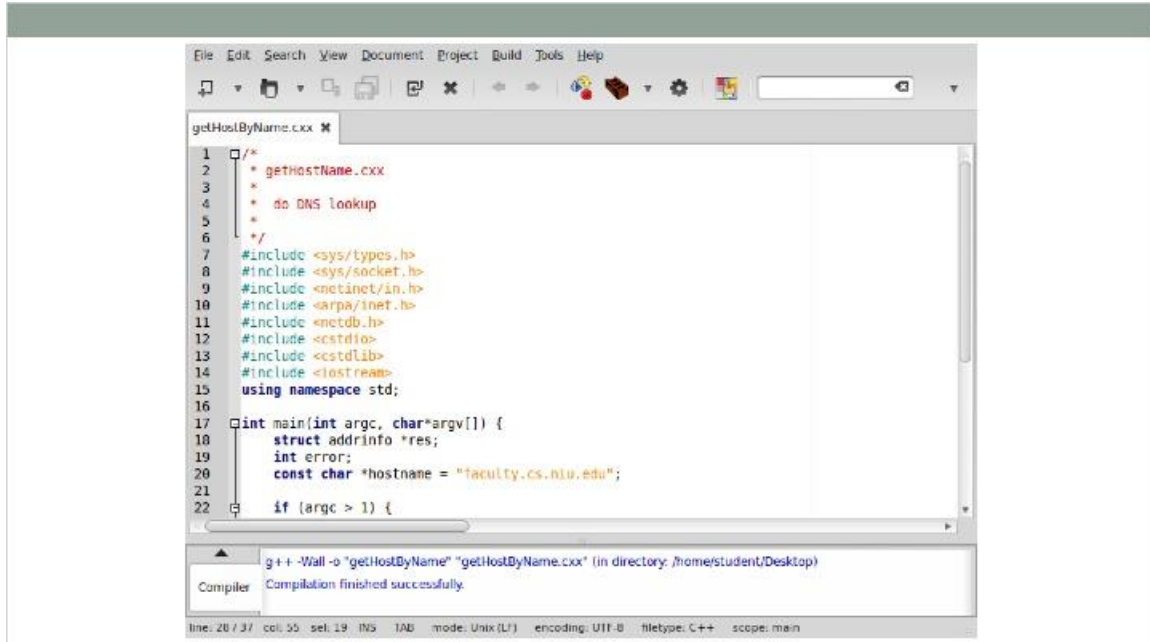
1.6 Socket address info structure

Socket address info structure

```
struct sockaddr_in {
    short          sin_family;   // e.g. AF_INET
    unsigned short sin_port;     // port
    struct in_addr sin_addr;     // IP address
    char           sin_zero[8]; // padding
};
```

- `sin_addr` can be printed via `inet_ntoa` function

1.7 Programming example



1.8 Summary

Summary

- Network concepts & terminology
- OSI reference model for protocols
 - Physical layer
 - Data Link layer
 - Network layer
 - Transport layer
- Domain Name Service