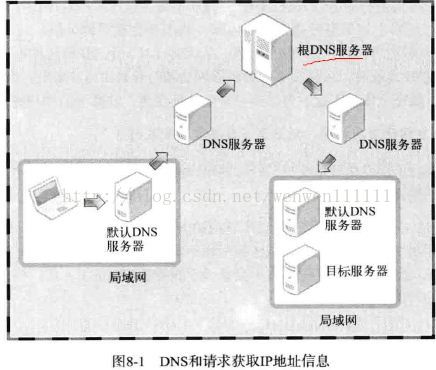
1. DNS （Domain Name System）是对IP地址和域名进行相互转换的系统，核心是DNS服务器。所有计算机都记录着默认DNS服务器地址，通过这个DNS服务器得到相应域名的IP地址信息。一般服务器域名不会轻易改变，但会相对频繁地改变服务器IP地址。可以通过ping www.baidu.com 查看对应的IP地址。

也可以通过nslookup命令查看计算机中注册的默认DNS服务器地址。

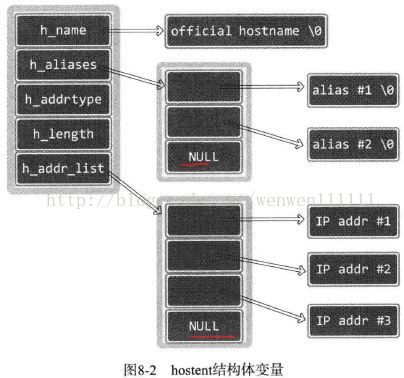
2. 计算机内置的默认DNS服务器并不知道网络中的所有域名IP地址，若本地不知道，则会询问其他DNS服务器。



3. IP地址与域名之间的转换

书中的一句名言：所有学习都要在开始前认识到其必要性！利用域名获取IP地址

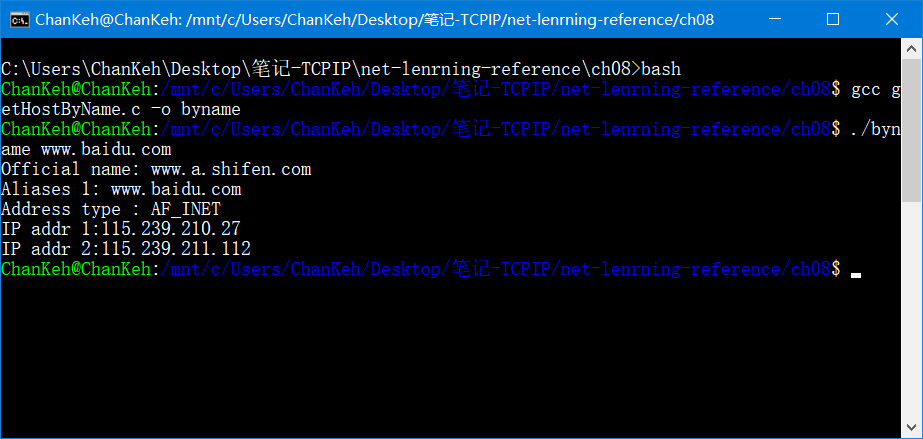
1. #include <netdb.h>
3. **struct** hostent \* gethostbyname(**const** **char** \* hostname);
5. 成功时返回hostent结构体地址，失败返回NULL指针
6. **struct** hostent{
7. **char** \* h\_name;  //official name 官方域名
8. **char** \*\* h\_aliases; //alias list，除了官方域名以外的其他域名
9. **int** h\_addrtype; //host address type
10. **int** h\_length; //address length
11. **char** \*\* h\_addr\_list //address list  域名对应的ip列表
12. //如果一个域名所对应的服务器需要多个服务器进行负载均衡，就可以一个域名对应多个ip
13. }



4. 示例一，gethostbyname：

1. #include <stdio.h>
2. #include <stdlib.h>
3. #include <unistd.h>
4. #include <arpa/inet.h>
5. #include <netdb.h>
7. **void** error\_handling(**char** \*message);
9. **int** main(**int** argc, **char** \*argv[]){
10. **int** i;
11. **struct** hostent \*host;
13. **if**(argc != 2){
14. printf("Usage : %s <addr> \n",argv[0]);
15. exit(1);
16. }
18. host = gethostbyname(argv[1]);
20. **if**(!host){
21. error\_handling("gethostbyname error");
22. }
24. printf("Official name: %s \n",host->h\_name);
26. **for**(i=0;host->h\_aliases[i];i++){
27. printf("Aliases %d: %s \n",i+1,host->h\_aliases[i]);
28. }
30. printf("Address type : %s \n",(host->h\_addrtype == AF\_INET)?"AF\_INET":"AF\_INET6");
32. **for**(i=0;host->h\_addr\_list[i];i++){
33. printf("IP addr %d:%s \n", i+1,inet\_ntoa(\*(**struct** in\_addr\*)host->h\_addr\_list[i]));
34. }
36. **return** 0;
37. }
39. **void** error\_handling(**char** \*message){
40. fputs(message,stderr);
41. fputs("\n",stderr);
42. exit(1);
43. }

运行结果：



1. 示例二：gethostbyaddr
2. #include <stdio.h>
3. #include <stdlib.h>
4. #include <unistd.h>
5. #include <string.h>
6. #include <arpa/inet.h>
7. #include <netdb.h>
9. **void** error\_handling(**char** \*message);
11. **int** main(**int** argc, **char** \*argv[]){
12. **int** i;
13. **struct** hostent \*host;
14. **struct** sockaddr\_in addr;
16. **if**(argc != 2){
17. printf("Usage : %s <addr> \n",argv[0]);
18. exit(1);
19. }
21. memset(&addr,0,**sizeof**(addr));
22. addr.sin\_addr.s\_addr = inet\_addr(argv[1]);
23. host = gethostbyaddr((**char** \*)&addr.sin\_addr,4,AF\_INET);
25. **if**(!host){
26. error\_handling("gethostbyaddr error");
27. }
29. printf("Official name: %s \n",host->h\_name);
31. **for**(i=0;host->h\_aliases[i];i++){
32. printf("Aliases %d: %s \n",i+1,host->h\_aliases[i]);
33. }
35. printf("Address type : %s \n",(host->h\_addrtype == AF\_INET)?"AF\_INET":"AF\_INET6");
37. **for**(i=0;host->h\_addr\_list[i];i++){
38. printf("IP addr %d:%s \n", i+1,inet\_ntoa(\*(**struct** in\_addr\*)host->h\_addr\_list[i]));
39. }
41. **return** 0;
42. }
44. **void** error\_handling(**char** \*message){
45. fputs(message,stderr);
46. fputs("\n",stderr);
47. exit(1);
48. }

运行结果：

