

微软ping程序源代码完整版(下) PDF转换可能丢失图片或格式  
，建议阅读原文

```
https://www.100test.com/kao_ti2020/140/2021_2022__E5_BE_AE_
E8_BD_AFping_c100_140151.htm if (argc >3) { datasize =
atoi(argv[3]). if (datasize == 0) datasize = DEF_PACKET_SIZE. if
(datasize >1024) /* 用户给出的数据包大小太大 */ {
fprintf(stderr,"WARNING : data_size is too large !\n"). datasize =
DEF_PACKET_SIZE. } } else datasize = DEF_PACKET_SIZE.
datasize = sizeof(IcmpHeader). icmp_data =
(char*)xmalloc(MAX_PACKET). recvbuf =
(char*)xmalloc(MAX_PACKET). if (!icmp_data) {
fprintf(stderr,"HeapAlloc failed %d\n",GetLastError()).
ExitProcess(STATUS_FAILED). }
memset(icmp_data,0,MAX_PACKET).
fill_icmp_data(icmp_data,datasize). // //显示提示信息 //
fprintf(stdout,"\nPinging %s ....\n\n",dest_ip). for(int i=0.i{ int
bwrote. ((IcmpHeader*)icmp_data)->i_cksum = 0.
((IcmpHeader*)icmp_data)->timestamp = GetTickCount().
((IcmpHeader*)icmp_data)->i_seq = seq_no .
((IcmpHeader*)icmp_data)->i_cksum =
checksum((USHORT*)icmp_data,datasize). bwrote =
sendto(sockRaw,icmp_data,datasize,0,(struct
sockaddr*)amp.from,amp.from)) statistic . /* 成功接收的数目 */
Sleep(1000). } /* Display the statistic result */ fprintf(stdout,"\nPing
statistics for %s \n",dest_ip). fprintf(stdout," Packets: Sent =
%d,Received = %d, Lost = %d (%2.0f%% loss)\n",times,
```

```

statistic,(times-statistic),(float)(times-statistic)/times*100).
WSACleanup(). return 0. } /* The response is an IP packet. We must
decode the IP header to locate the ICMP data */ int
decode_resp(char *buf, int bytes,struct sockaddr_in *from) {
IpHeader *iphdr. IcmpHeader *icmphdr. unsigned short iphdrlen.
iphdr = (IpHeader *)buf. iphdrlen = (iphdr->h_len) * 4 . // number
of 32-bit words *4 = bytes if (bytes printf("Too few bytes from
%s\n",inet_ntoa(from->sin_addr)). } icmphdr =
(IcmpHeader*)(buf iphdrlen). if (icmphdr->i_type !=
ICMP_ECHOREPLY) { fprintf(stderr,"non-echo type %d
recvd\n",icmphdr->i_type). return 1. } if (icmphdr->i_id !=
(USHORT)GetCurrentProcessId()) { fprintf(stderr,"someone elses
packet!\n"). return 1. } printf("%d bytes from %s:",bytes,
inet_ntoa(from->sin_addr)). printf(" icmp_seq = %d.
",icmphdr->i_seq). printf(" time: %d ms
",GetTickCount()-icmphdr->timestamp). printf("\n"). return 0. }
USHORT checksum(USHORT *buffer, int size) { unsigned long
cksum=0. while(size >1) { cksum =*buffer . size
-=sizeof(USHORT). } if(size) { cksum = *(UCHAR*)buffer. }
cksum = (cksum >> 16) (cksum & 0xffff). cksum = (cksum
>>16). return (USHORT)(~cksum). } /* Helper function to fill in
various stuff in our ICMP request. */ void fill_icmp_data(char *
icmp_data, int datasize){ IcmpHeader *icmp_hdr. char *datapart.
icmp_hdr = (IcmpHeader*)icmp_data. icmp_hdr->i_type =
ICMP_ECHO. icmp_hdr->i_code = 0. icmp_hdr->i_id =
(USHORT)GetCurrentProcessId(). icmp_hdr->i_cksum = 0.

```

```
icmp_hdr->i_seq = 0. datapart = icmp_data sizeof(IcmpHeader). //  
// Place some junk in the buffer. // memset(datapart,E, datasize -  
sizeof(IcmpHeader)). } /***** 附: ping命令执行时  
显示的画面 *****/ * C:\Documents and  
Settings\houzhijiang>ping 236.56.54.12 * * * * Pinging 236.56.54.12  
with 32 bytes of data: * * * * Request timed out. * * Request timed  
out. * * Request timed out. * * Request timed out. * * * * Ping  
statistics for 236.56.54.12: * * Packets: Sent = 4, Received = 0, Lost =  
4 (100% loss), * * *
```

```
 \*****/  
 /*****/ *  
C:\Documents and Settings\houzhijiang>ping 127.0.0.1 * * * *  
Pinging 127.0.0.1 with 32 bytes of data: * * * * Reply from 127.0.0.1:  
bytes=32 time* Reply from 127.0.0.1: bytes=32 time* Reply from  
127.0.0.1: bytes=32 time* Reply from 127.0.0.1: bytes=32 time* * *  
Ping statistics for 127.0.0.1: * * Packets: Sent = 4, Received = 4, Lost  
= 0 (0% loss), * * Approximate round trip times in milli-seconds: * *  
Minimum = 0ms, Maximum = 0ms, Average = 0ms * * * 100Test 下  
载频道开通 , 各类考试题目直接下载。详细请访问  
www.100test.com
```