#ifdef \_WIN32

#ifndef \_UNISTD\_H

#define \_UNISTD\_H 1

/\* This file intended to serve as a drop-in replacement for

\* unistd.h on Windows

\* Please add functionality as needed

\*/

#include <winsock2.h>

#include <direct.h> /\* for \_getcwd() and \_chdir() \*/

#include "getopt.h"

#include <io.h>

#include <process.h> /\* for getpid() and the exec..() family \*/

#include <stdlib.h>

#define srandom srand

#define random rand

/\* Values for the second argument to access.

These may be OR'd together. \*/

#define R\_OK 4 /\* Test for read permission. \*/

#define W\_OK 2 /\* Test for write permission. \*/

#define X\_OK R\_OK /\* execute permission - unsupported in Windows, \*/

#define F\_OK 0 /\* Test for existence. \*/

#define access \_access

#define dup2 \_dup2

#define execve \_execve

#define ftruncate \_chsize

#define unlink \_unlink

#define fileno \_fileno

#define getcwd \_getcwd

#define chdir \_chdir

#define isatty \_isatty

#define lseek \_lseek

/\* read, write, and close are NOT being #defined here, because while there are file handle specific versions for Windows, they probably don't work for sockets. You need to look at your app and consider whether to call e.g. closesocket(). \*/

#define ssize\_t int

#define STDIN\_FILENO 0

#define STDOUT\_FILENO 1

#define STDERR\_FILENO 2

/\* should be in some equivalent to <sys/types.h> \*/

//typedef \_\_int8 int8\_t;

//typedef \_\_int16 int16\_t;

//typedef \_\_int32 int32\_t;

//typedef \_\_int64 int64\_t;

//typedef unsigned \_\_int8 uint8\_t;

//typedef unsigned \_\_int16 uint16\_t;

//typedef unsigned \_\_int32 uint32\_t;

//typedef unsigned \_\_int64 uint64\_t;

#endif /\* \_UNISTD\_H \*/

#else

#include <unistd.h>

#endif /\* \_WIN32 \*/