#ifndef \_\_ARP\_UTIL\_H\_\_

#define \_\_ARP\_UTIL\_H\_\_

#include <netinet/if\_ether.h>

struct arp\_packet

{

struct ether\_header eth\_hdr;

struct ether\_arp arp;

};

void print\_usage();

void set\_hard\_type(struct ether\_arp \*packet, unsigned short int type);

void set\_prot\_type(struct ether\_arp \*packet, unsigned short int type);

void set\_hard\_size(struct ether\_arp \*packet, unsigned char size);

void set\_prot\_size(struct ether\_arp \*packet, unsigned char size);

void set\_op\_code(struct ether\_arp \*packet, short int code);

void set\_sender\_hardware\_addr(struct ether\_arp \*packet, char \*address);

void set\_sender\_protocol\_addr(struct ether\_arp \*packet, char \*address);

void set\_target\_hardware\_addr(struct ether\_arp \*packet, char \*address);

void set\_target\_protocol\_addr(struct ether\_arp \*packet, char \*address);

char\* get\_target\_protocol\_addr(struct ether\_arp \*packet);

char\* get\_sender\_protocol\_addr(struct ether\_arp \*packet);

char\* get\_sender\_hardware\_addr(struct ether\_arp \*packet);

char\* get\_target\_hardware\_addr(struct ether\_arp \*packet);

#endif