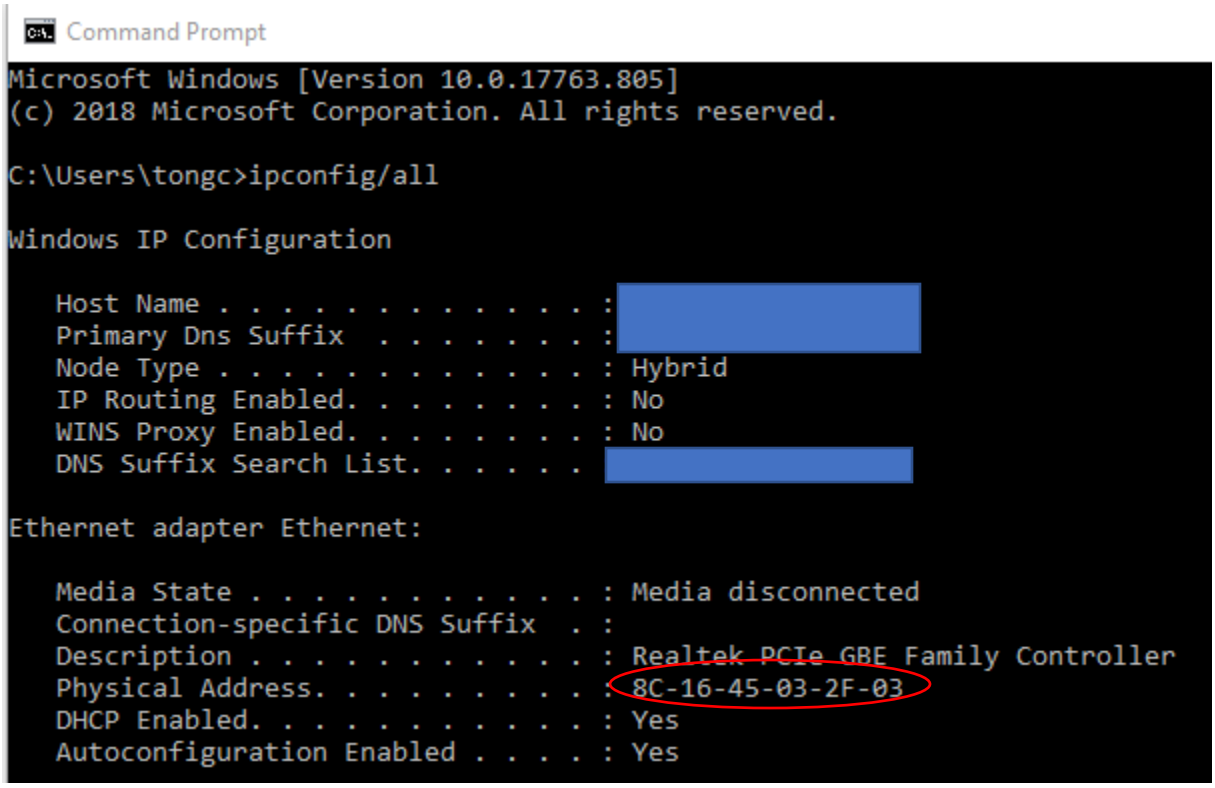
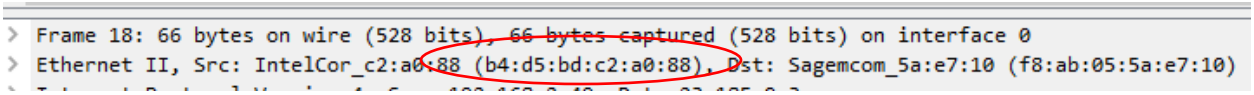
Network Programming Assignment 2

Tong Chen

10189689

The screen shots below show my MAC address and the gateway MAC address.





The main consideration made for the application was condensing the 4-way “handshake” process of discover-offer-request-ACK to obtain an IP address from the DHCP server, to just two steps of discover-offer.

This assignment is implemented using 5 classes: the client, the DHCP server, the DNS server, and a worker thread for each of the server classes.

The DHCP server starts by creating a socket on port 7070. It maintains a pool of available addresses and taken addresses. Its also implements a timer to keep track of the lease times for all taken addresses. The timer decreases the lease time by 1 second every second, and if the

resulting lease time is 0 it will return the address back to the available pool. When the server receives a packet, it creates a new DHCP worker thread to process that packet. Depending on what message is received in the packet, the server will either release the assigned IP address or renew the lease for the assigned IP address by updating the lease timer in the taken addresses pool.

The client obtains an IP address and the DNS port number by sending the DHCP server a request for IP configurations. The client also implements a timer that will automatically release the given IP address after a minute if no renew requests are made. The client can choose to send a message to the DHCP or DNS server. If sending messages to the DHCP server, a release request can be made to release the given IP address or a renew lease request can be made to renew the lease timer. If sending a domain name query to the DNS server, the client will use the IPv4 address returned to create a complete packet to request the home page of the specified domain name.

The DNS starts by creating a socket on port 9090. It maintains mappings domain names to their respective IPv4 and IPv6 addresses. When a packet is received the server will create a new DNS worker thread to process the packet. The server will use the domain name to find and return the associated IPv4 and IPv6 address to the client.

One of the difficulties faced was ensuring that the taken address pool was thread-safe since it there are numerous references and updates being made to that mapping. Initially it was coded using a Hashtable, but it faced concurrency issues when releasing multiple IP addresses. ConcurrentHashMap fixed those issues while also providing better overall performance.

One improvement that can be made is towards the client requesting the IP configurations using the discover-offer-request-ACK process instead of condensing it to 2 steps. Another improvement that can be made is towards the timer on the DHCP server. The current implementation is very simple in nature as the current timer runs a task every 1000 milliseconds to decrease the lease the time of all taken IP addresses by 1 second. A more detailed implementation will likely be more accurate in keeping track of the lease times.