NA

1. DHCP

https://www.linksys.com/us/support-article?articleNum=137180

http://www.tech-faq.com/dhcp-reservation.html

The reason is that through DHCP reservation, servers can reserve IP addresses for each MAC address. So as long as devices have the same MAC address, we are likely to have the same ip address.

2. DNS

(a).

https://en.wikipedia.org/wiki/Domain_Name_System#Internationalized_domain_names

https://wiki.bravenet.com/How_the_domain_name_system_works

If a single server handles all domain name translation services…

1. If the server is broken, no one can use DNS and it will be a mass.
2. Having the worldwide data for all machine in the world take a huge amount of space to store data. It will be hard to maintain the machine.
3. If there are many people use the server at the same time, the server may be overload and the service will be super slow. Clients who are far from the server may also have very bad use experience for it take a long time for DSA response..

(b).

https://en.wikipedia.org/wiki/Domain_Name_System#Internationalized_domain_names

https://www.lifewire.com/what-is-a-dns-cache-817514

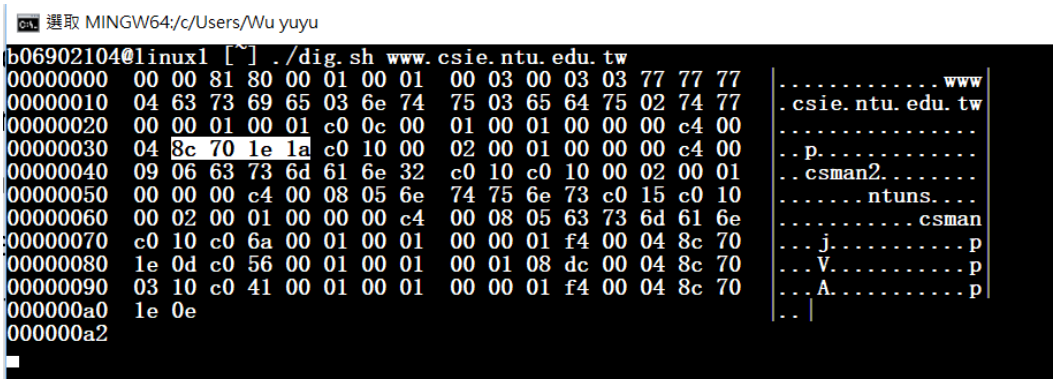
DNS cache is a temporary database which is a memory of recent DNS lookups so that your computer can quickly refer to when it's trying to figure out how to load a website. Before a browser issues its requests to the outside network, the computer intercepts each one and looks up the domain name in the DNS cache database. The database contains a list of all recently accessed domain names and the addresses that DNS calculated for them the first time a request was made. The DNS cache stores this address, the requested website name, and several other parameters from the host DNS entry. DNS cache help the whole process get quicker.

(c).

https://en.wikipedia.org/wiki/Hex_dump

http://www.firewall.cx/networking-topics/protocols/domain-name-system-dns/161-protocols-dns-response.html

https://stackoverflow.com/questions/26851317/how-to-retrieve-ip-address-from-dns-response-message



(d).

http://www.keyboardbanger.com/dns-message-format-name-compression/

http://marek.vavrusa.com/rfc-dnscomp/

https://tools.ietf.org/html/draft-ietf-dnsind-local-compression-05

http://www.tcpipguide.com/free/t_DNSNameNotationandMessageCompressionTechnique-2.htm

The intent of using DNS compression is to reduce the message length, especially that of UDP datagrams, by avoiding repetition of domain names. This would require that each name be spelled out fully using the encoding method. Domain names may be replaced with a pointer to a prior occurrence of the same name. It uses the combination of labels and label lengths. A two-byte subfield is used to represent a pointer to another location in the message where the name can be found. The first two bits of this subfield are set to 1, and the remaining 14 bits contain an offset that species where in the message the name can be found. For example, in our case, csman follows by a pointer points to .csie.ntu.edu.tw, and the entire domain name is csman.csie.ntu.edu.tw.

It is better to reduce length of DNS responses when transmit with UDP. Because if the packet is larger than 576 bytes, we are not sure if packets will be transmitted and reassembled well. The fragmented packets may be drop during the UDP transmission process and we would not want this tragedy happens. So if we do some encoding to make the packet smaller, we have more chance to transmit the packets safely.

(e).

https://searchsecurity.techtarget.com/definition/DNS-attack

https://en.wikipedia.org/wiki/DNS_spoofing

https://en.wikipedia.org/wiki/DNS_rebinding

https://en.wikipedia.org/wiki/DNS_hijacking

1. DNS spoofing (DNS cache poisoning)

   It corrupts Domain Name System data and introduced wrong information to the DNS resolver's cache. It will cause the DNS to return an incorrect result record, which results in traffic being diverted to the attacker's computer or other

malicious website.

It can be prevented by being less trusting of the information passed to them by other DNS servers, and ignoring any DNS records passed back which are not directly relevant to the query

2. DNS hijacking (DNS redirection)

   This attack can achieve by overriding a computer's configuration to point at a rogue DNS server under the control of an attacker, or through modifying the behavior of a trusted DNS server so that it does not comply with internet standards.

   People can use trustful open DNS server to avoid being used by rough DNS server without their knowledge. Such as google public DNS server.

3. DNS rebinding

   The attacker registers a domain and delegates it to a DNS server under the attacker's control. The server is configured to respond with a very short TTL record, preventing the response from being cached. When the victim browses to the malicious domain, the attacker's DNS server first responds with the IP address of a server hosting the malicious client-side code. The malicious client-side code makes additional accesses to the original domain name. When the victim's browser runs the script, it makes a new DNS request for the domain and the attacker replies with a new IP address.

   To prevent this kind of attack, Private IP addresses can be filtered out of DNS responses. Web servers can reject HTTP requests with an unrecognized host header.