Лабораторийн ажил №3

DNS (Domain Name System)

МТЭС, МКУТ, Компьютерийн ухаан
Б.Барсболд, 22B1NUM4397

Ажлын зорилго

Энэхүү лабораторийн ажлаар Домайн нэрийн системийн үйл ажилгаатай танилцаж, DNS сервертэй хэрэглэгчийн төхөөрөмж хэрхэн харилцаж байгааг судална.

Үндсэн ойлголт

1. Домайн нэр гэж юу вэ?

Домайн нэр гэдэг нь нэг ёсны интенетийн утасны дэвтэр гэсэн үг юм. Гэхдээ утасны дугаар биш IP хаяг хадгалдагаараа ялгаатай. Өөрөөр хэлбэл бидний цээжилхэд хэцүү IP хаягуудыг цээжилхэд амар уншигдахуйц үгэнд харгалзуулах хадгалдаг гэсэн үг. (What is DNS? | How DNS works, 2024)

2. Resource record (RR) гэж юу вэ?

DNS серверүүд хамтдаа DNS тархсан өгөгдлийн сан[1] болдог бөгөөд RR-уудыг хадгалдаг. RR болгон хост нэрийг IP хаягад харгалзуулан хадгалдаг. DNS хариу мессеж бүр нэг эсвэл түүнээс олон resource record-уудыг агуулдаг.

Resource record нь 4 урттай tuple төрлөөр хадгалагддаг бөгөөд доорх талбаруудыг хадгалдаг.

```
(Name, Value, Type, TTL)
```

Зураг 1 Resource record-ийн бүтэц

`TTL` нь амьдрах хугацаа буюу кейш-д хэр хугацаанд хадгалагдахыг нь зааж өгдөг. Энэ хугацаа хэтэрвэл кейшээс устдаг.

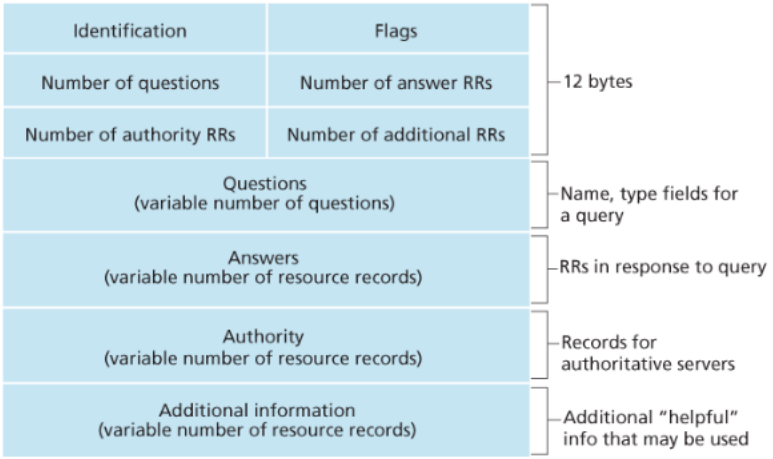
Харин `Type` буюу төрөл нь `Name` болон `Value` гэсэн талбаруудын хадгалсан утга юу болохыг илэрхийлдэг. (James F. Kurose, 2017)

[1] distributed database

- Хэрэв `Type=A` байвал `Name` талбар хост нэрийг харин `Value` талбар IP хаягийг илэрхийлдэг. Өөрөөр хэлбэл A төрлийн төрөлтэй record стандарт hostname-с IP хаяг руу харгалзуулсан гэсэн үг. Жишээ нь: `(relay1.bar.foo.com, 145.37.93.126, A, TTL)`
- Хэрэв `Type=NS` байвал `Name` талбар нь домайн харин `Value` талбар нь authoritative серверийн хост нэр байна. Энэ record нь цааш DNS хүсэлтийг гинжин хэлхээ байдлаар илгээхэд ашиглагдана. Жишээ нь: `(foo.com, dns.foo.com, NS, TTL)`
- Хэрэв `Type=CNAME` байвал `Value` талбарт ямар нэгэн хост нэрийн өөр хувилбар байна. Энэ record цааш бодит хост нэрийг ашиглан IP хаягийг олход хэрэглэгдэнэ. Жишээ нь: `(foo.com, relay1.bar.foo.com, CNAME, TTL)`
- Хэрэв `Type=MX` байвал `Value` талбар мэйл серверийн бодит хост нэр байх бөгөөд `Name` талбарт мэйл серверийн хувилбар нэр байна. Жишээ нь: `(foo.com, mail.bar.foo.com, MX, TTL)`

3. DNS мессеж.

Зөвхөн хоёр төрлийн DNS мессеж байдаг бөгөөд хүсэлт болон хариу мессежүүд яг адил форматтай байдаг.



Зураг 2 DNS мессежийн формат (James F. Kurose, 2017)

4. DNS хэрхэн ажилдаг вэ?

DNS сервер нь урвуу харсан мод[2] хэлбэрээр зохион байгуулагддаг бөгөөд үүнд Root DNS servers, Top-Level Domain Servers, Authoritative servers болон DNS Resolver-ууд орно. Веб хуудсыг ачааллахын тулд 4 төрлийн DNS сервер орлцоно.

4.1. DNS recursor: Recursor-г бид номын сангийн хаа нэгтээ байгаа номыг зааж өгдөг номын санч гэж ойлгож болно. DNS recursor нь хэрэглэгчээс аппликейшн-р дамжуулан query-г авах

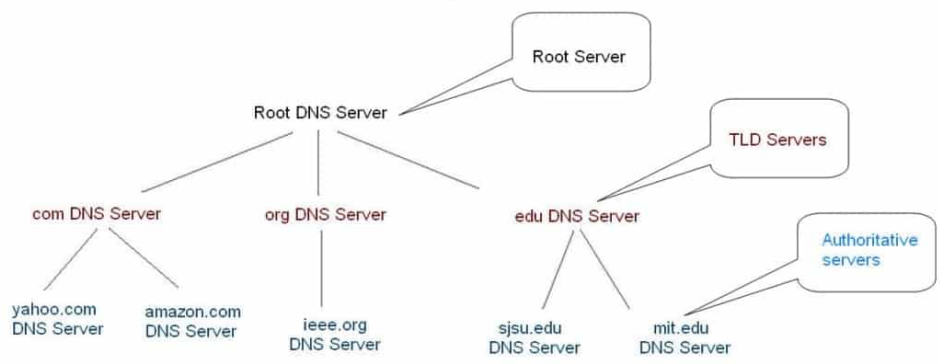
[2] Hierarchical

зориулалттай сервер юм. Ерөнхийдөө recursor нь хэрэглэгчийн query-г биелүүлхийн тулд нэмэлт хүсэлтүүд явуулж гүйцэтгэх юм.

4.2. Root nameserver: Root nameserver нь хүнд ойлгомжтой хост болон домайн нэрнүүдийг IP хаяг руу хөрвүүлэхэд хамгийн эхэнд ажилдаг. Үүнийг номыг сангийн төрөл бүрийн номыг тавиурыудыг заадаг заагч гэж ойлгож болно.

4.3. TLD nameserver: Top Level Domain (TLD)-ийг тодорхой номын тавиурууд гэж ойлгож болно. Энэ хэсэг домайн нэрийг олох дараачийн алхам болох ба домайн нэрнүүдийн сүүлийн хэсгийг агуулдаг. Жишээ нь example.com гэж домайн нэр байдаг гэж үзвэл com гэдэг хэсэг нь TLD юм.

4.4. Authoritative nameserver: Энэ сервер нь домайн нэрийг IP хаяг руу хөрвүүлэх хамгийн сүүлийн сервер бөгөөд тавиур дээрх номнуудын жагсаалт гэж хэлж болно. Тодохой нэг домайн нэр энэ хэсэгт IP хаяг руу хөрвөгддөг бөгөөд эхэнд хүсэлт явуулсан recursor-д буцаадаг.



Зураг 3 DNS hierarchical structure (https://cloudinfrastructureservices.co.uk)

5. DNS асуулгын[3] төрлүүд.

5.1. Recursive query: recursive query-д хэрэглэгч тал заавал хүсэлтийн хариу эсвэл алдааны мэдээлэл буцаахыг DNS server-с шаардана. Энэ нь ихэвчлэн recursor DNS байдаг. Хэрэв хэрэглэгчийн хүссэн хүсэлтийн хариу бичлэгүүд дунд байхгүй бол алдааны мессеж, байвал хариу IP хаягийг буцаана.

5.2. Iterative query: Энэ тохиолдолд DNS хэрэглэгч DNS серверийн хамгийн боломжит утгыг буцаахыг зөвшөөрдөг ба хэрвээ DNS-д асуусан нэртэй таарах нэр олдохгүй бол доод түвшины DNS серверээс асуух хаягыг буцаана. Дараа нь DNS хэрэглэгч буцаасан хаяг руу хүсэлт явуулах ба энэ нь алдаа эсвэл timeout болтол давтагдана.

[3] query

5.3. Non-recursive query: Энэ хүсэлт ихэвчлэн DNS серверийн хариуцаж буй бичлэг рүү эсвэл түүний кейш-д байгаа бичлэг рүү хандах үед болох ба DNS сервер илүү сүлжээны ачаалал үүсэхээс сэргийлэн ихэвчлэн дуудагддаг бичлэгүүдийг кейш-дээ хадгалдаг.

6. DNS кейш гэж юу вэ?

Кейшийн зорилго нь хурд болон найдвартай байдал үүсгэхийн тулд хэрэглэгч өөр дээр байгаа санах ойд түр хадгалах юм. Хэрэглэгчид ойр түр хадгалснаар хурдтайгаар асуудлыг шийдэх болон DNS хайлтийн гинжин хайлтаас зайлсхийж, сүлжээгээр дамжих их ургалыг багасгахад давуу тал үүсгэх юм.

6.1. Веб хөтөчийн DNS кейш.

Орчин үеийн веб хөтөчүүд тодорхой хугацаанд өөр дээрээ хадгалах кейштэй байдаг. Хамгийн ойр DNS кейш нь энэ бөгөөд харьцангүй цөөн тооны үйлдлээр тухайн домайн нэрд харгалзах IP хаягийг авдагаараа давуу талтай. Мөн DNS бичлэгэд хүсэлт явах үед хамгийн эхэнд энэ кейш-с хайх ба хэрэв үүнд байхгүй бол дараагийн түвшины кейш-с хайдаг.

Chrome веб хөтөчийн хувьд chrome://net-internals/#dns энэ хаягаар орон кейшийг харж болно.

6.2. Үйлдлийн системийн түвшний DNS кейш.

Үйлдлийн системийн түвшиний кейш DNS хүсэлт тухайн машинаас гархаас өмнө нь хоёр дахь бөгөөд сүүлчийн зогсоол юм. Энэ үйлдлийг хийдэг үйлдлийн системийн хэсгийг “stub resolver” эсвэл DNS хэрэглэгч гэж нэрлэдэг. Stub resolver аппликейшн-с DNS хүсэлтийг авсаны дараагаар өөрийн кейш-с тухайн бичлэг байгаа эсэхийг шалгаж үздэг бөгөөд хэрэв байхгүй бол хүсэлтийг recursive байдлаар дотоод сүлжээний гадагш ISP (internet service provider)-ийн DNS recursive resolver руу илгээдэг.

ISP-ийн recursive resolver хүлээн аваад өмнөхтэй адилаар өөрийн доторх өгөгдлүүдэд хүсэлтийн хариу хадгалагдсан эсэхийг шалгадаг.

Recursive resolver кэйш дотроо хадгалсан бичлэгүүдээс хамааран нэмэлт функцуудтэй байдаг.

1. Хэрэв resolver-д таарах бичлэг байхгүй харин тухайн хүсэлтэд хариулах эрхтэй сервер[4]-ийн хаяг байх юм бол тэр сервер руу шууд хүсэлт явуулан, DNS хүсэлтийн хэд хэдэн дамжилгийг орхидог. Энэ нь root серверээс хайх үйлдлийг багасгадаг бөгөөд энэ хугацааны хувьд ч харьцангуй хурдан юм.
2. Хэрэв resolver-д тухайн хүсэлтэд шууд хариулах серверийн хаяг байхгүй бол шууд TLD сервер руу хүсэлт явуулдаг ба root серверийг алгасдаг.

[4] NS records

3. Хэрэв resolver-д тухайн хүсэлтэд хариулах хариулт болон шууд TLD-руу хандах хаяг байхгүй бол root сервер руу ханддаг. Энэ нь ихэнхдээ кейш цэвэрлэгдсэний дараа гардаг.

Туршилт

Туршилтад бид хэрэглэгч талаас DNS хэрхэн ажиллаж байгааг судална. Энгийнээр хэрэглэгч нь DNS сервер рүү хүсэлт илгээж хариултыг хүлээн үйл ажиллагааг судална. Ингэхдээ бид хэд хэдэн командууд ба тэдгээрийн цаана ажиллах DNS сервертэй холбогдоход ашиглагддаг түүлүүдийн ажиллагаатай танилцана.

1. nslookup

Бидний хамгийн эхэлж турших түүл бол *nslookup* юм. Энэ түүлийг бид аливаа үйлдлийн систем (Linux/Windows)-ийн командын мөр дээр ажиллуулж болно. Энэхүү *nslookup* түүлийг DNS серверээс дурын домайн нэрийг талаар мэдээлэл авах зорилгоор хүсэлт илгээх (query)-д ашиглана. Өөрөөр хэлбэл бид *NS* сервер дээр хадгалагдаж байгаа *records*-оос тухайн домайн нэртэй холбоотой мэдээллийг авахаар тусгайлсан *query*-ийг илгээнэ. Мөн энэхүү түүлийг ашиглан бид шаталсан бүтцийн аль ч түвшинд ажиллаж байгаа NS сервер рүү хүсэлт илгээх боломжтой юм.

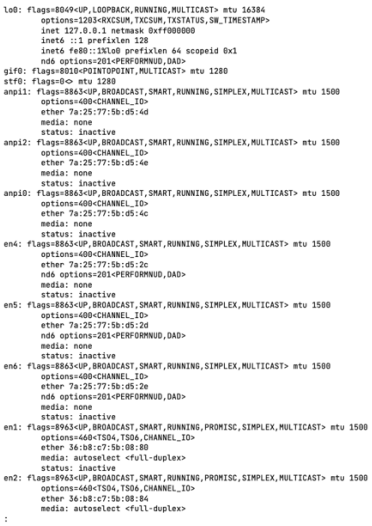
```
❯ nslookup num.edu.mn
Server:         2405:5700:2:5::4
Address:        2405:5700:2:5::4#53

Non-authoritative answer:
Name:   num.edu.mn
Address: 52.220.222.172
Name:   num.edu.mn
Address: 3.1.92.70
```

Зураг 4 Гэрийн сүлжээнээс nslookup num.edu.mn командыг хэрэгжүүлсэн байдал

Дээрх үр дүнд 2405:5700:2:5::4 хаягтай DNS сервер ашигласан бөгөөд хариу мессеж Non-authoritative буюу num гэсэн домайн хаягтай холбоогүй буюу ISP-ийн DNS серверийн кейш-с ирсэн нь харагдаж байна.

2. ipconfig (ifconfig)



Зураг 5 ifconfig командыг MacOS дээр туршиж үзсэн үр дүн

Даалгавар

1. nslookup-ийг ашиглаж дараах веб серверийн IP хаяг, домайн нэрийг олоорой. Мөн дурын 2 веб домайны нэр, IP хаягийг олж бөглөөрэй.

Вебийн домайн нэр	**IP хаяг**	**Хариулт өгсөн NS сервер**
edge-star-mini-shv-01-sin6.facebook.com	157.240.7.35	2405:5700:2:5::4
www.utoronto.ca	23.185.0.1	2405:5700:2:5::4
www.wikipedia.org	103.102.166.224	2405:5700:2:5::4
www.microsoft.com	23.219.73.192	2405:5700:2:5::4
25.224.186.35.bc.googleusercontent.com	35.186.224.25	2405:5700:2:5::4
alpha.gogo.mn	202.131.225.29	2405:5700:2:5::4
tomyo.mn	18.159.45.10	2405:5700:2:5::4
chimege.mn	95.216.211.78	2405:5700:2:5::4

2. www.yahoo.com гэсэн домайнтай холбоотой мэдээллүүдийг авч болох доорх хүснэгтийг бөглө.

Record-ийн төрөл	IP хаяг эсвэл сервер нэр	Хариулт өгсөн NS сервер
MX	hostmaster.yahoo-inc.com	2405:5700:2:5::4
NS		
A		
AAAA		
CNAME		

3. Wireshark DNS

Сорих асуулт

1. Бид IP хаяг буюу серверийн тоон хаягийг цээжилхэд хүндрэлтэй. Гэвч бид ганц хоёрхон биш хэдэн зуун веб сайтыг амьдралдаа хэргэлдэг билээ. Үүний улмаас IP хаягуудад хүнд уншигдахуйц нэр оноож өгөх хэрэгтэй байсан ба DNS протокол гарж ирсэн.
2. DNS сервер нь урвуу харсан мод хэлбэрийн бүтэцтэй бөгөөд хамгийн дээр root сервер түүний доод түвшинд TLD серверүүд, түүний доод түвшинд SLD гэх мэт бүтэцтэй байдаг. (Зураг 1 Resource record-ийн бүтэц)
3. DNS caching гэдэг нь сүлжээний ачаалал болон хүсэлтэд хариулах хугацааг хэмнэх зорилгоор хэрэглэгчид ойр санах ойнууд дээр тодорхой хугацаанд нөөцлөх бөгөөд хэрэв нөөцөнд байвал илүү хүсэлт явуулалгүйгээр газар дээр нь шийдэх юм (DNS кейш гэж юу вэ?).
4. Primary DNS болон Secondary DNS-ийн гол ялгаа нь хадгалж буй record юм. Primary DNS тодорхой домайны хамгийн анхны бөгөөд үндсэн хадгалах сан бол Secondary DNS нь зөвхөн унших боломжтой хуулбаруудыг хадгалан түүгээ тодорхой хугацаанд Primary DNS-ээс шинэчилж байдаг.
5. PTR гэдэг нь Pointer record гэсэн үг бөгөөд энгийн record-с ялгаатай нь яг эсрэгээрээ ажилдаг. Өөрөөр хэлбэл тодорхой IP хаягд харгалзах домайн нэрийг олоход хэрэглэгддэг.
6. TLD (Top-Level Domain) нь root серверээс нэг түвшний доор байрладаг бөгөөд SLD (Second-Level Domain)-с нэг түвшний дээр байрладаг. Энэ нь домайн нэрний хамгийн арын хэсгийг (жишээ нь: com, net, org, mn гэх мэт) хариуцдаг хэсэн бөгөөд түүнээд SLD салаалсан байдалтай байдаг (жишээ нь: google, wikipedia, edu гэх мэт).
7. DNS Resource Record нь DNS серверийн тодорхой хост нэр IP хаяг хоорондын оноолтын хадгалдаг хэсэн бөгөөд IP хаягийг олох, домайн нэрийг олох, өөрөөр нэрлэгдсэн домайн нэрийн бодит домайн нэрийг олох, мэйл серверийн домайн нэрийг олох, DNS хүсэлтийг цааш үргэлжлүүлэх серверийг олох зэрэг олон төрлийн мэдээлэлүүдийг 4 талбартай tuple маягаар хадгалдаг DNS-ийн өгөгдлийн сангийн гол өгөгдлийн төрөл юм.
8. Local DNS сервер нь дотоод сүлжээний компьютеруудын хост нэр болон IP хаягийг хөрвүүлэх үүрэгтэй.
9. Authoritative гэдэг нь яг тухайн домайныг хариуцаж буй серверээс ирсэн хариу бол non-authoratitive нь тухайн домайнтай ямар нэгэн холбоогүй сервер өөрийн кейш дээрээс хариулж буй хариу юм.