

DNS: Domain Name System



Mindennek az alapja, ha valami nem megy a hálózaton → nézd meg, DNS-sel minden OK?
AD is erősen DNS függő → hálózathalál

GP hibák

Nincs hálózati nyomtatás

Internetezés sincs ☹

DNS beállítása, működtetése nem delegálható → nem bízható az internetszolgáltatóra

Vállalati rendszer → saját DNS-t kell felépíteni

DNS probléma → nem érem el ezt, nem érem el azt,.. nem működik ez, nem működik az....

I. Név, névfeloldás, névkiszolgálók

0. NetBIOS név: windowsban a gépnév kötelező, még a telepítés során meg kell adni. Egyedi legyen a hálózaton! Ez a név egytagú, nincs benne hierarchia pl. pistagepe, julimachine, stb
(Megosztásnál használata → \\mancimachine\megosztott_konyvtar_neve
NetBIOS név – IP névfeloldást WINS Server végzi a helyi hálózaton)

1. FQDN: Fully Qualified Domain Name

Teljes tartománynév → hierarchikus felépítésű

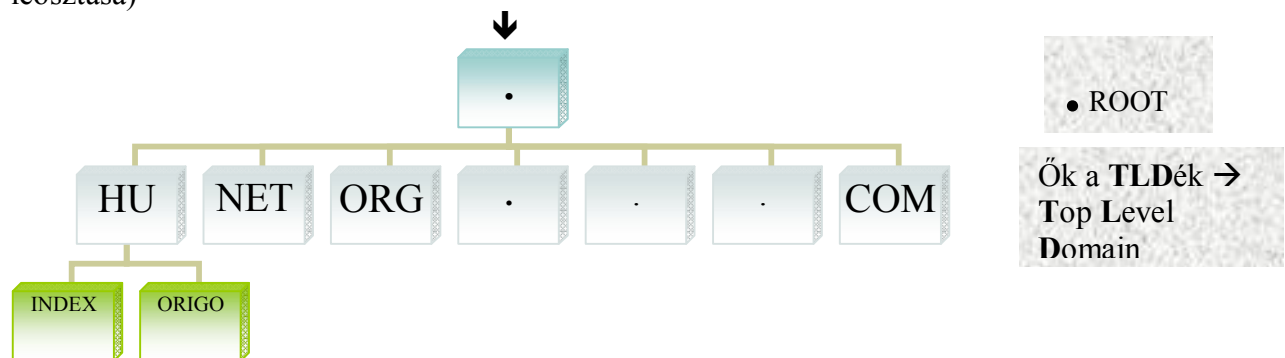
(lehet azonos gépnév a hálózaton → Interneten a rengeteg www nevű gép!)

DNS = FQDN → IP

Ez egy hatalmas AB lenne → **elosztott** ABban tárolják

Elosztott névtér: valamennyi NameServer maga felelős a hozzá tartozó név-IP párokért

Delegálás: ezt a hierarchia ágat kedves NameServer te kezeled! (névszolgáltatási feladat leosztása)



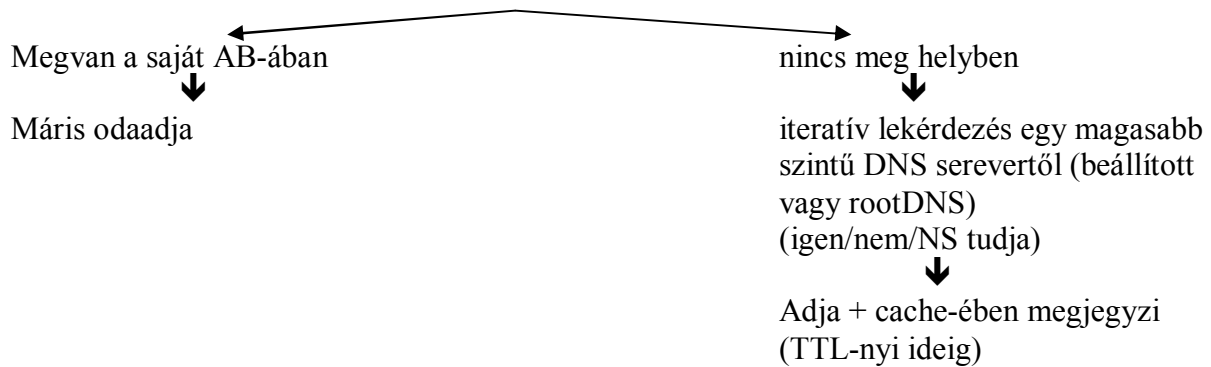
Domain Name Server: a tartomány névkiszolgálója
Tartományonként minimum 1 (inkább több)

Interneten elérhető

Helyi hálózaton lévő (a tiéd)

- Tartomány gépeit saját AB-ában tárolja
- Ennek az AB-nak a feltöltése (név → IP párok) a RG feladata

Hogyan működik a Domain Name Server? Milyen szolgáltatást kell nyújtania?
 Kap egy FQDN nevet, és IPt kell válaszul visszaadnia



DNS Serveren az adattárolás egysége a ZÓNA

1 AB → 1 zóna

1 AB → lehet több tartomány, de névhierarchiailag egybefüggő kell, hogy legyen,
 például: vállalat.hu
 iroda.vallalat.hu
 titkarsag.vallalat.hu
 (tartomány és alárendelt tartományai)

II. ZÓNA

(„zóna, az egy fájl” → System32\dns\.....dns fájl)

DNS névtér → itt vehetünk fel mi neveket. Lehet egyre mélyebb, de általában három szintnél nem megyünk mélyebbre.

? DNS serverünknek telepítése után hány kliense van? → 0

`ipconfig /all`

beállításon látszik, most nem a miénk adja a névfeloldás válaszokat, hiába telepítettük fel, most kliensként használjuk.

Lehet kapcsolgatni a valódi Internet és a saját DNS Serverünk között

ControlPanel / Network... / Local Area / je / Property / TCP/IP → bogyót a Use ... DNS...-be Internetre? → internetszolgáltatót beállítani. Ha kinn van az Interneten regisztrálhatunk sajátot, benne van ahierarchiában.

Indítása: run / dnsmgmt.msc vagy Start / Administratív Tools / DNS Sever

Server cache-t lássuk → View / Advanced –t bekapcsolni



Itt is menjünk le a RootHints-ekig Mi RootHints?



Server / je / Properties / RootHints → szerverünk számára ezek elérhetők, hozzájuk fordul, ha nem tudja a kért névfeloldást kiszolgálni

Nézzünk egy konkrét példát: kérést kap egyik kliensétől a DNS szerverünk: **Hol az index.hu?**

DNS szerverünk nem tudja a választ → RootHint-ekhez fordul

1. Valamely RootName szervertől: **Hol az index.hu?**
Nem tudom! De HU-ék tudják! Nesze, itt egy csomó HU-nameserver, fuss hozzájuk, kérdezd tőlük!
Szerverünk tovább lett küldve, de már közelebb került a válaszhoz.
2. Kiválaszt a kapott listából egy HU-szerver: **Hol az index.hu?**
Fogalmam sincs, de ORIGO-ék tudják! Nesze, itt az ORIGO.HU-nameserver, fuss hozzá, kérdezd tőle!
Szerverünk tovább lett küldve, de már mindjárt meg lesz a válasz.
3. ORIGO.HU nameserver-hez: **Hol az index.hu?**
Nesze itt van, jól jegyezd meg és ne zaklass többet!
Szerverünk odaadja a kliensnek a kért IP-t, és jól megjegyzi → be-cache-li.
Meddig? → a TTL-nyi ideig, ez mindig csökken

Ha gépünk kilát az internetre, (van érvényes, külső IPje), akkor Packetyzerben látszanak az iteratív kérdések (igen / nem / tovább passzoltak). Ez a server cache-ben is látszik View / Advanced-et be kell kapcsolni, így látszik a **DNSServerCache**, ha ebben van rossz bejegyzés, ezt is törölni kell. Ebből a Cache-ből válaszol(na) a hozzáforduló kliensek névfeloldására, illetve, ha nincs benne, akkor megy érdeklődni valamely RootHint-hez. De mivel ő is lehet kliens, ezért neki is van **kliens DNSCache**, amit az ipconfig / flushdns-sel törölhetünk. (különben 10 percig a hibás névfeloldás miatt nem működik a jó, a javított) A TTL itt is látszik, add ki egymás után kétszer a ipconfig / displaydns parancsot -> látszik a TTL csökkenés.

Ha nem lát ki az Internetre, akkor is látszik a Root-on állva a TTL csökkenés, nyomj frissítést! Ha letelik a TTL, akkor kiesik innen a Cache-ből, ezt (TTL-t) a zóna szerzője állítja be.

Hosszú TTL → sokáig békén hagynak

→ változtatás elhúzódik

Rövid TTL → állandóan zaklatnak

→ változtatások gyorsan átmennek

pl. IPcím átállítás előtt felgyorsít (rövid TTL) → mindenki frissít, átírja a régi IPt az újra, utána visszaállhatunk hosszú TTLre.

Előre tekintve: AD (Activ Directory) halálát okozza a DNS hiba. Ezt észleled, kijavítod, a Cache-ben ott a rossz, régi → törölni kell mind a kettő Cache-t!

Bemutatni:

```
ping ilyen.nincs.sehol
ipconfig / displaydns
```

gyönyörűen látszik, becachelte a semmit, és innentől 10'-ig Cache-ből adja a semmit, hacsak ki nem töröld vagy elmész 10'-ig kávézni és azt gondolod, hogy a DNS hiba megoldásában ez segített, nem, csak kiesett a cache-ből.

Zóna létrehozása

Forward lookup zóna: NÉV → IP

Reverse lookup zóna: IP → NÉV

1. Forward lookup zóna

Hozzuk létre forward lookup zónát!

DNSServer / ForwardLookupZone / je / New Zone/ Primary/ neve: tojas.nyul →

létrejön a zónafájl a WINDOWS\system32\dns könyvtárban

../ dinamikus update nem kell, hagyományos internet zónát készítünk, nem ADsat →

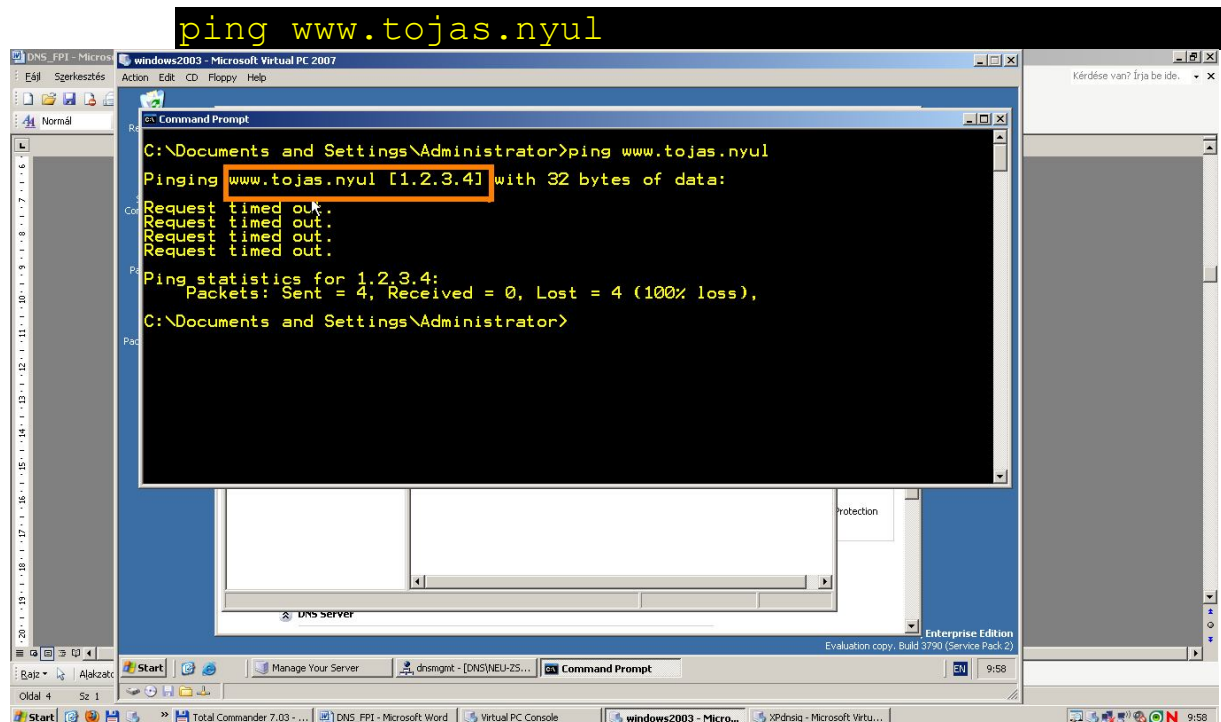
Finish, kész a tojas.nyul zóna

Tegyük bele egy www rekordot, Host típusút!

tojas.nyul / je / New Host (A) / név: www (alatta lévő mezőben látszik a teljes neve)

IPAddress most mindegy, úgysem érjük el, 1.2.3.4 DE a névfeloldás működni fog, a DNSServer feladata ez!

Próba:



Látod? A névfeloldás működik!

Nézzük meg a zónafájl tartalmát! ZÓNA = FILE

```
notepad \WINDOWS\system32\dns\tojas.nyul.dns
```

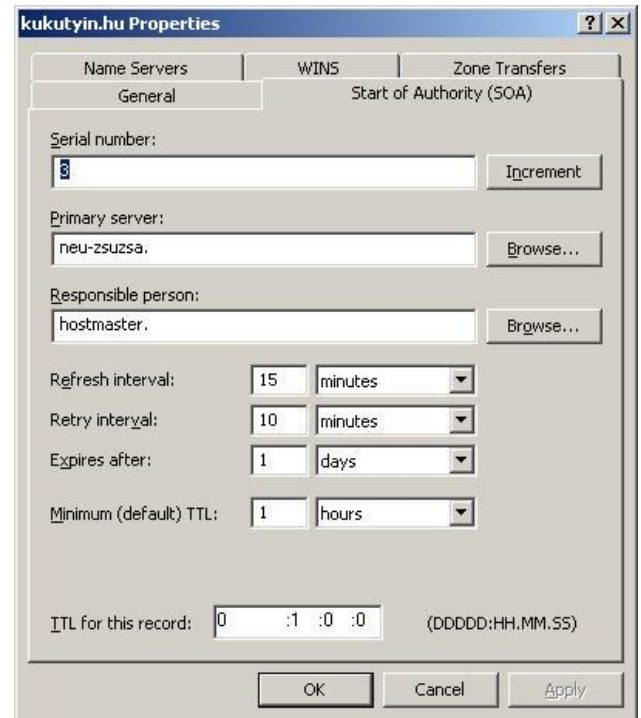
```
;
;
; Database file tojas.nyul.dns for tojas.nyul zone.
; Zone version: 1
;
```

```
@          IN SOA neu-zsuzsa. hostmaster. (
        1          ; serial number
        900        ; refresh
        600        ; retry
        86400      ; expire
        3600       ) ; default TTL
```

```
;
; Zone NS records
;
```

```
@          NS   neu-zsuzsa.
```

```
;
; Zone records
;
```



Tegyünk bele egy 'aaa' nevű másik HostRecordot! IPAddress most is mindegy, úgysem érjük el, 50.60.70.80, DE a névfeloldás működni fog, a DNSServer feladata ez!

```
aaa      A      50.60.70.80
mentés,
```

Viszont, ha most ránézünk a zónafájltra, nincs változás!! DNSServeren frissíteni kell, meg kell neki mondani, hogy változás van, magától csakúgy nem veszi észre. → reload

Vagány: www begépelése nélkül is működjön a névfeloldás



Host rekord, mely neve '@' = this domain (ez a domén)

```
@          A      10.20.30.40
```

!! fontos: legvégére mindig kell egy üres sor!!

Nem működött a névfeloldás, mert nem frissítettünk.

→ Reload: DNS server olvassa be újra a zónafájlt!

→ Update Server Data File: elmenti a zónafájlbba a változásokat

jól beCachelte, tehát ha kijavítjuk a hibát, akkor sem fog, mehetünk kávézni vagy töröljük a Cache tartalmát, és most pingelünk.

DNS Management-ben *same as parent folder*-ként jelenik meg

Nézzük a zónafájlt!

@ NS record → ugyanennek a zónának az NS rekordja hová mutat? = „én magamat hogy hívlak?”

@ IN SOA és egy csomó szám → mik ezek, keressük meg a vizuális felületen!

The screenshot shows the 'kukutyin.hu Properties' dialog box with the 'Start of Authority (SOA)' tab selected. The fields are as follows:

- Serial number: 3 (with an 'Increment' button)
- Primary server: neu-zsuzsa. (with a 'Browse...' button)
- Responsible person: hostmaster. (with a 'Browse...' button)
- Refresh interval: 15 minutes
- Retry interval: 10 minutes
- Expires after: 1 days
- Minimum (default) TTL: 1 hours
- TTL for this record: 0 :1 :0 :0 (format: DDDD:HH:MM:SS)

SOA = a zóna kezdőrekordja

3 ; serial number

Felelős személy email címe (na, ide pont kell, mert a @ foglalt karaktere a zónafájlnak)

900 ; refresh

600 ; retry

86400 ; expire

3600 ; default TTL

Serial number: sorozatszám. Innen látható, történt-e változás, módosult-e a zóna, mert ha igen, a módosulásokat el kell vinni!

Primary Server: Ha kliens kéréssel fordul DNSServer felé, nem tudja, hogy akitől a választ kapta, az az írható (Primary server) vagy csak egy kópia (Secondary server, aki 'readonly'). Ha kliens írni szeretne a zónafájlba (ez bizony előfordulhat), akkor tudnia kell, ki a Primary. Valamint Secondary-knek is tudniuk kell az elérhetőségét, hisz ettől kér zónatrafkszert.

Responsible person: a zónáért felelős személy email címe

Időzítések: replikációs feleknek szóló információk, mesterpéldánytól milyen gyakran

Refresh interval: két zónatranszfer közötti idő

Retry interval: hiba esetén – valamiért nem sikerült a zónatranszfer – az újrapróbálkozás időköze

Expires after: bejegyzés lejáratási ideje. Ha a mesterrel megszakad a kapcsolat, a kópiák ennyi nap után dobják fel a talpukat. → RG hány napra mehet szabadságra ☺

Minimum (default) TTL: bejegyzés minimális élettartama, a kérő ennyi ideig tárolja a saját AB-ában

TTL for this record: az előzőtől mennyivel térjen el, rekordonként beállítható, lássuk!

kukutyin zónánk valamely tagját szerkesszük meg, a TTL-je térjen el a zónáétól, majd nézzük meg a zónafájlt, hogyan változott a tartalma? Na, most kell: Update Server Data File: elmenteni a zónafájlba a változásokat

View/Advanced-et bekapcsolni!

aaa Host record / Properties → átállítjuk 3 órára. Update Server Data File most kell!!!

Megnyitjuk notepadban a zónafájlt, és látjuk a változást:

aaa 10800 A 50.60.70.80

Nézzük meg, parancssorból pingeljük meg az aaa.tojas.nyul-t és a www.tojas.nyul-t, majd nézzük meg a DNSCache tartalmát, látszik a TTL-ek különböző értéke.

Rekord típusok

Host (A): az egyetlen normális rekord. NÉV → IP-t rendel

Eddig ilyeneket gyártottunk.

Alias (CNAME): Canonical Name = becenév NÉV → NÉV

Pl. egy gépen több website-ot futtatunk

Már megvan: Host (A) record: www (.tojas.nyul)

Most tojas.nyul / je / Alias (CNAME)

'ezisittvan' legyen a neve, és a www.tojas.nyul gépre mutasson!

'megmegezis' neve, és a www.tojas.nyul gépre mutasson!

Próbáljuk ki. Pingeljünk, majd nézzük a DNSCache-t!

?

mi történik, ha a www.tojas.nyul gép IPje megváltozik? Semmi, mert a CNAME névhez nevet rendel.

?

Kimutathat a zónánkból?

'alma' és mutasson a www.index.hu-ra

Ha kilátnánk a netre, működne.

Tehát a névfeloldás most már kétkörös lett:

Becenév → név → IP

NS rekord: ő tartja össze a hierarchiát. NS rekordok mutogatnak magasabb szintről az alacsonyabbra (pl. hu → origo). Ezért aztán ez is kétkörös: NS → név → IP

Bárki ne vihesse el zónatranszferrel a zóna fájlt. Zónagyökér / je / Property / Zona Transfer → az vihesse el, aki itt meg van jelölve: secondary partner megjelölése

Három lehetőség:

- bárki: ekkor nslookup-pal bárki elviheti
- akit NS rekorddal megjelölök, ekkor ugyanezen tulajdonságlap NameServer fülén fel kellennem az illetőt, mint NS rekordot. Ezután már csinálhatunk egy secondary zónát. Nézzük csak meg ezután a zónafájlt! Az NSrekord névhez nevet rendel, ezért, amit vizuális felületen beállítottam, az a zónafájlból 2 db sor lett! (NS rekord: @ - NS – név; Host rekord: név – A - IP)
- illetve az alább feltüntetett IP-jű gépek

Mail Exchanger (MX) rekord: név → név → IP

Tartomány alapértelmezett levelező kiszolgálójának neve. SMTP-nek továbbítania kell a levelet, DNSServertől elkéri az MX rekordot és az ehhez tartozó Host rekordot → megvan a levelező kiszolgáló IP-je.

név → név → IP

MX A

MX-ben lehet [10] prioritás, ha több levelező szerver is van, a nagyobbbal kezd ki.

SOA: már volt

SRV: services rekord, majd AD-hoz kell nagyon

PTR (Pointerrekord): mutatórekord, ő nem a Forward, hanem a Reverse Lookup Zone rekordja, majd mindjárt jön!

Subdomain: aldomén

Tojas.nyul / je / New Domain → valami

bemegyek valamibe / je / New Host / www / 11.11.11.11

Elmentem a zónafájlba, (a zóna az EGY fájl!!!) megnézem a zónafájlt → nincs benne subdomain bejegyzés!! Csak egy olyan Host rekord, ami a www.valami-hez rendeli a 11.11.11.11-es IP-t.



A vizuális felület lehet megtévesztő, ezért ismerni kell a zónafájl felépítését, tartalmát értelmezni! (azért ez nem olyan bonyolult). Ha semmit nem írsz a subdomainbe → meg sem jelenik a zónafájlban.

2. Reverse lookup zóna: fordított névfeloldás



IP → NÉV

Mikor? Ellenőrzi a kliens, regisztrálva van-e már? Vagy SMTP servert ellenőrzi a fogadó oldal: regisztrálva van-e az illető IP?

(A nagyobb baj, ha még azt is le akarja ellenőrizni, hogy azon a néven van-e regisztrálva, amilyen néven most kopogtat. SMTP servert nem könnyű rávenni, hogy mindenhol és mindent ugyanazzal a névvel végezzen, illetve sok beállítást kell elvégezni. A jó az lenne, ha az ellenőrzés ott megállna, hogy létezik regisztráció, ilyen IP-jű géphez van név rendelve.)

Baj, mert ha azt is ellenőrzi, hogy azzal a névvel van regisztrálva, amivel kopogtat? Hááát, mutathat egy gép IP-jére több Host rekord is! Azt, hogy épp melyik nevet adja vissza az ellenőrzés - ????



adott egy DNS fá.

Hol fordulhat elő benne az az IP, hogy 1.1.1.1? → Bárhol és bármennyiszer! Pl egy gép futtat 16 website-ot → 16-szor benne van a DNS fában.

Hogyan tudom meg, hogy benne van-e és hogy milyen névvel? → Nehezen és lassan!

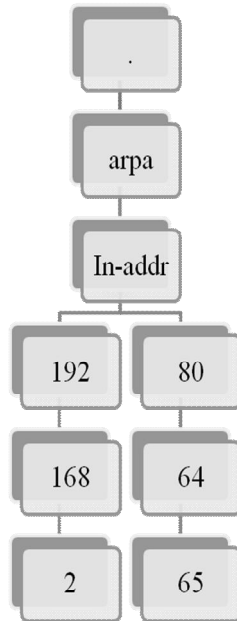


Ez egy teljesen másik hierarchia → ls DNS Cache a kliensen

```
ipconfig /displaydns  
1.0.0.127.in-addr.arpa
```



Mert ha egy IP-hez keresel internet nevet, melyik ágán indulj el a hierarchiának? Bárhol lehet. Ezért építeni kell egy másik hierarchiát → Reverse lookup zóna.



Ez azt jelenti, hogy amikor létrehozuk a Host rekordot, PTR-t is létre kell hozni (pipa), ez egy mutató rekordot jelent. Csakhogy, ha nincs meg az a Revers zóna, (nem létező fájlba nehéz írni) melybe ezt a PTR-t írni kell, na akkor baj van, mert ha a DNSServerünk nem tudja ezt feloldani helyben – magában, elindul a nagyvilágba. Kérdezzeti a saját, belső hálózatának IP címeihez tartozó neveket → saját belső hálózatunk adatai ott szambáznak a NET-en. Ez biztonsági kérdés → NE engedjük! Megoldás: „nyelők” készítése a belső alhálózatokra.

Reverse Lookup zone /je/New zone / Primary /10. és utána üresen érdemes hagyni, mert ő felépíti a fát magától.

Ha létrehozok egy másik subdomain-t, 10.20. → az első Host rekord regisztrálásakor nőni fog a fa! Megjelenik a 20-as alfája, tehát csak az első B-ot érdemes kitölteni, utána beindul az automatizmus.

Látszik a többi beállítás alapján, hogy ugyanolyan, mint a forward (primary/secondary; zóna fájl,..ugyanúgy írhatok a zóna fájlba, ugyanúgy Update.. vagy Reload, stb) A szerepe más, a szerkezete ugyanolyan, lekérdezések ugyanolyan elven (van válasz / nincs válasz / ő tudja). Fontos: egyik sem helyettesíti a másikat!! Kapcsolat köztük csak akkor, amikor a PTR bejegyzésre kerül!

(nézzük meg ezt a zónafájlt is!)

Mindig meg kell csinálni a belső vállalati hálózaton, elég, ha csak a kliensek Windows-ok. 10, 172, 192 alhálózatokra ezeket a „nyelőket”.

III. Nézzünk körül az Interneten

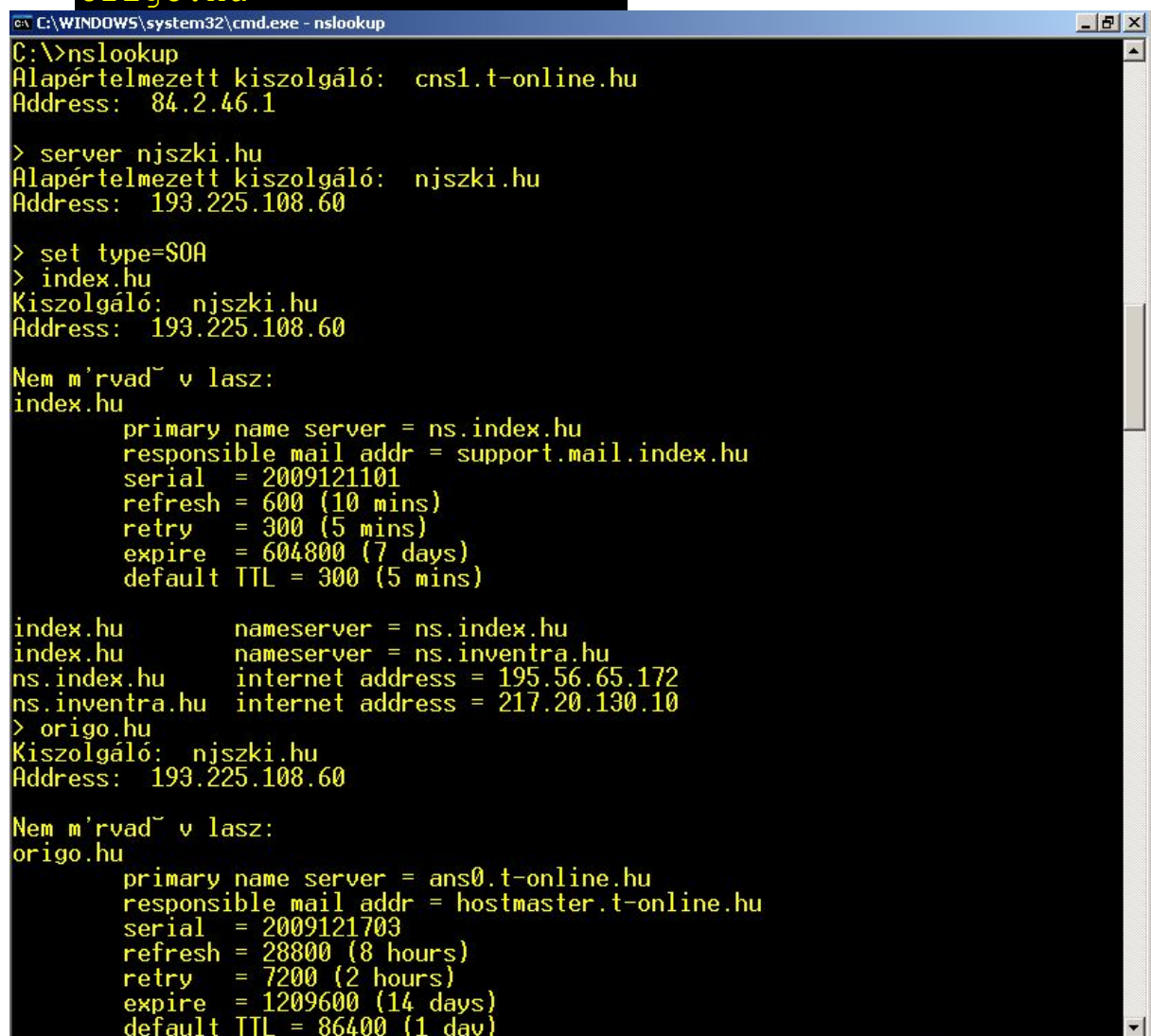
A mi DNSServerünk (nagy bánatunkra) nem lát ki a Netre ☹

DE!! Van egy „parancssori jószág”, melynek segítségével valamely a Neten kinn lévő DNSServer nevében tudunk kérdezősködni:

Sajnos most a külső gép parancssorát kell használnunk ☹

Váltunk mondjuk az index.hu és mint index tudom nézni a nagyvilágot:

```
nslookup
server njszki.hu
set type=SOA
index.hu
netacademia.net
origo.hu
```



```
C:\WINDOWS\system32\cmd.exe - nslookup
C:\>nslookup
Alapértelmezett kiszolgáló: cns1.t-online.hu
Address: 84.2.46.1

> server njszki.hu
Alapértelmezett kiszolgáló: njszki.hu
Address: 193.225.108.60

> set type=SOA
> index.hu
Kiszolgáló: njszki.hu
Address: 193.225.108.60

Nem m'rvad v lasz:
index.hu
    primary name server = ns.index.hu
    responsible mail addr = support.mail.index.hu
    serial = 2009121101
    refresh = 600 (10 mins)
    retry = 300 (5 mins)
    expire = 604800 (7 days)
    default TTL = 300 (5 mins)

index.hu      nameserver = ns.index.hu
index.hu      nameserver = ns.inventra.hu
ns.index.hu   internet address = 195.56.65.172
ns.inventra.hu internet address = 217.20.130.10
> origo.hu
Kiszolgáló: njszki.hu
Address: 193.225.108.60

Nem m'rvad v lasz:
origo.hu
    primary name server = ans0.t-online.hu
    responsible mail addr = hostmaster.t-online.hu
    serial = 2009121703
    refresh = 28800 (8 hours)
    retry = 7200 (2 hours)
    expire = 1209600 (14 days)
    default TTL = 86400 (1 day)
```

nem mérvadó válasz → nem ő a zóna primary servere, ő csak egy read-only, secondary server (kópia)

mi látható még?

primary nameserver (NS rekord)

zónafelelős email címe
serial → ez lehet egy dátum, így látszik, mikor változott utoljára
refresh, retry expire, default TTL
és ezek alatt az NS rekordokat feloldó Host rekordok

```
set type=MX
```

ha emailt akarunk küldeni, kinél kopogtassunk?

Névhez nevet rendel az MX rekord is. Ha több van, van mellette „tarifa információ”, ekkor a kopogtató az „olcsóbbat” választja.

Ha csak egy név van, lehet mögötte több IP, ez a DNS – round-robin (utolsó pár előre fuss!)
Hogy lássuk, ehhez pingelni kell.

```
ping www.microsoft.com  
ping www.gmail.com  
ping google.com
```

Csináljunk ilyet! Ugyanarra a névre több IP is hallgat → hol az egyik fog válaszolni, hol a másik.

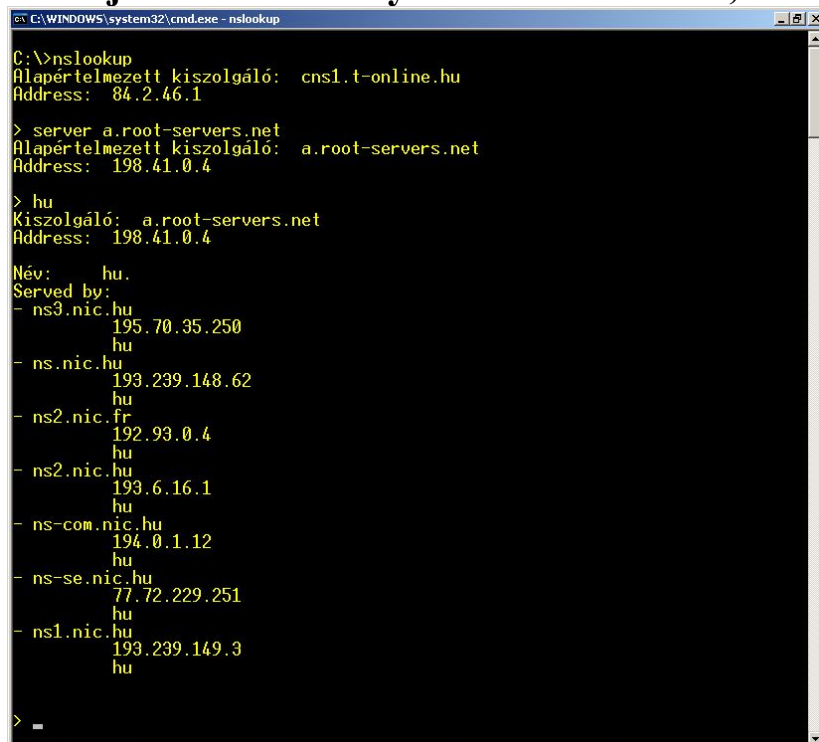
Barack névre hallgat az 10.11.12.14

10.11.12.15

10.11.12.16

10.11.12.17 IP-jű gépek. Hozzuk létre ezeket a Host rekordokat és pingeljük a barack.tojas.nyul-t többször!

Induljunk el valamely Root-tól! Nézzük, mit látunk!



```
C:\WINDOWS\system32\cmd.exe - nslookup  
C:\>nslookup  
Alapértelmezett kiszolgáló:  cns1.t-online.hu  
Address:  84.2.46.1  
  
> server a.root-servers.net  
Alapértelmezett kiszolgáló:  a.root-servers.net  
Address:  198.41.0.4  
  
> hu  
Kiszolgáló:  a.root-servers.net  
Address:  198.41.0.4  
  
Név:      hu.  
Served by:  
- ns3.nic.hu  
    195.70.35.250  
    hu  
- ns.nic.hu  
    193.239.148.62  
    hu  
- ns2.nic.fr  
    192.93.0.4  
    hu  
- ns2.nic.hu  
    193.6.16.1  
    hu  
- ns-com.nic.hu  
    194.0.1.12  
    hu  
- ns-se.nic.hu  
    77.72.229.251  
    hu  
- ns1.nic.hu  
    193.239.149.3  
    hu  
  
>
```

Adjuk ki a következő parancsokat!

```
nslookup
server a.root-servers.net
set type=SOA
HU

..

server ns.nic.hu
set type=SOA
HU
```

HU nameserverek
látszanak
Az NS rekordjaik is

(root) SOA mezője is látszik
Valamint az összes root

hu zóna SOA mezője
HU nameserverek
látszanak

DNS alapfogalmak még:**0. DNSServer működésének feltételei:**

- Server IP-je statikus (nem kaphatja DHCP Servertől)
- Serveren beállítani az elsődleges tartomány nevét:
Rendszertulajdonságok / Számítógép név / Módosítás
Tartománynév: Részletek gomb / DNS utótag → DNS Suffix

1. Forwarder: (továbbítók)

Megadhatók további DNSServerek, akikhez bizonyos zónákra vonatkozó kérések továbbíthatók → feltételes továbbítás

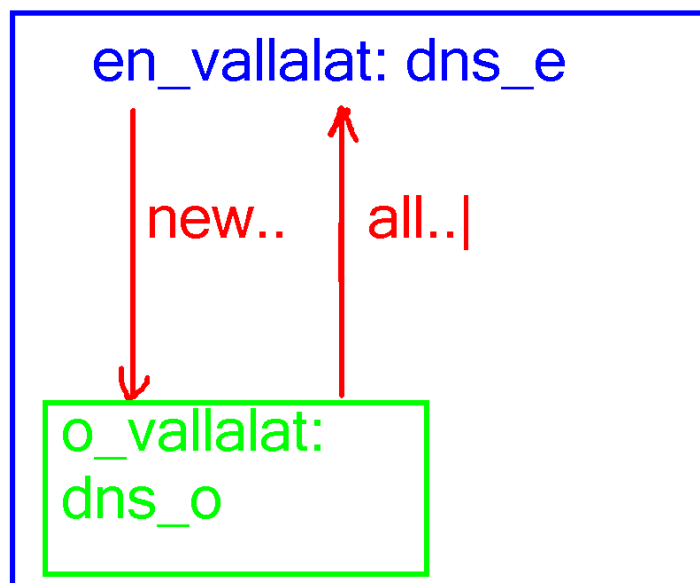
Megadható, alapértelmezett kiszolgáló, ha én, mint DNSServer nem tudom teljesíteni a névfeloldást, először az itt beállítottak küldöm.

Munkahelyi hálózat Privát és publikus részre osztható. A privátért pl én felelek, az összes többit meg elküldöm az alapértelmezettnek. Az alapértelmezetten lehetséges beállítás, hogy a privát zóna kiszolgálásáért engem írt be forwarder-nek.

Server / je / Properties / Forwarders / Selected domain's.... → IP-ket beírjuk és Add, így valamennyi kérés ide továbbítódik

Csak bizonyos domain-ek névfeloldását vágjuk hozzá másikhöz: DNS domain / New → zóna nevének beírása, Add-dal az IP-je a másik DNSServernek.

Pl **en_vallalat** bekebelezte az **o_vallalat**-ot. **o_vallalat** dns zónáját érintő kéréseket csak szolgálja ki továbbra is ő (new ... az **en_vallalat** server-én) Az **o_vallalat**-tól érkező valamennyi más kérés viszont **en_vallalat** Serveren keresztül (add ... az **o_vallalat** server-én).



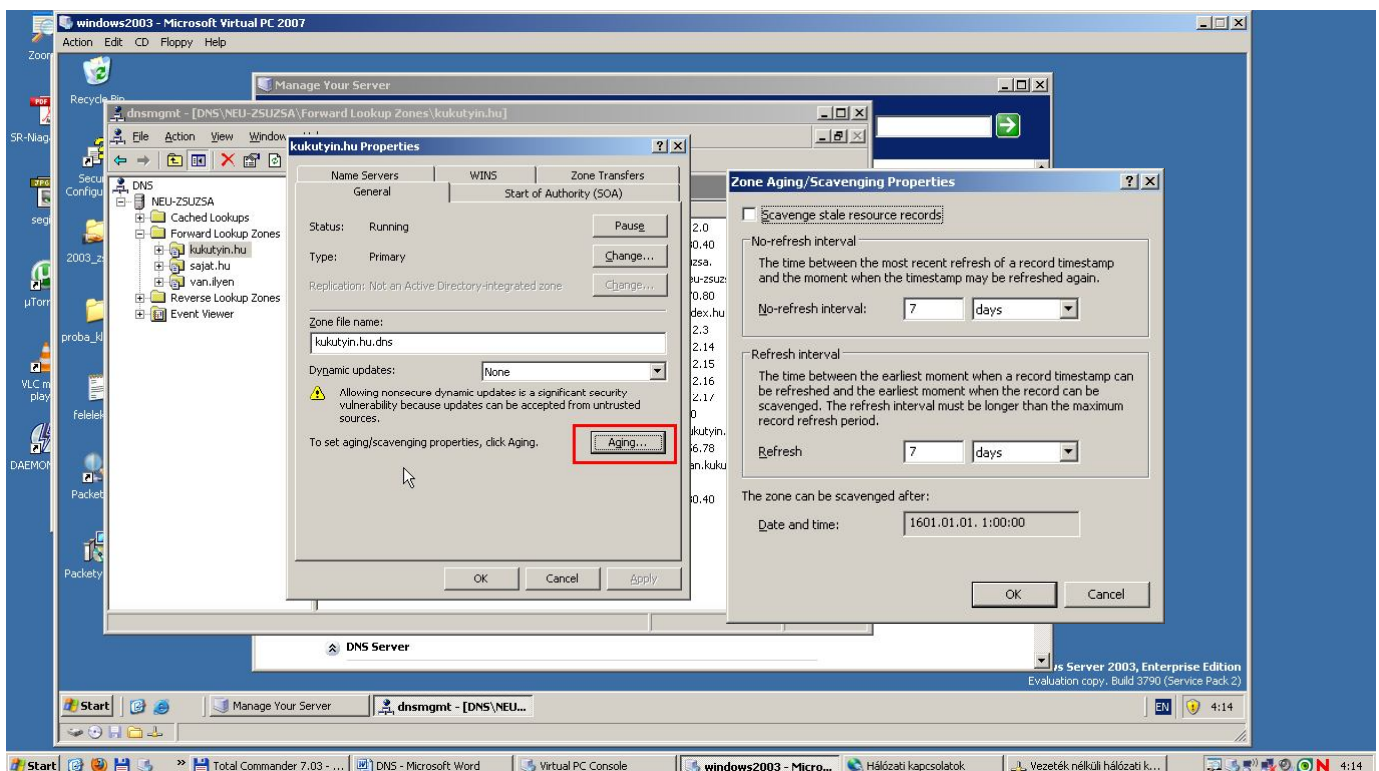
2. Dinamikus DNS

DNS rendszergazdának nagy munka → zónafájl feltöltése Host rekordokkal (PTR-ekkel). Mi a helyzet, ha dinamikus IP-jű gépet kell felvinni, a privát (belső) hálózaton azért ez előfordul ☺. Ilyenkor bejelentkezéskor frissíteni kell, valamint figyelni kell a DHCP kölcsönzési idő beállításánál is. →

Dinamikus DNS-bejegyzés: Kliens bejelentkezik a Serverre, Server automatikusan létrehozza az illetőhöz tartozó Host rekordot és a PTR-t.

Alapértelmezésként a 2003-as Serveren ez kikapcsolva → külön be kell kapcsolni zónánként! Zóna / je / Properties / Dynamic updates / Allow only secure....

Itt beállítható, hogy mennyi ideig ne regisztrálja újra, illetve mennyi idő után törölje a nem használt Host rekordokat.



Dinamikus frissítést így beállítottuk. És ezek a bejegyzések örökre ott maradnak? Egyszer bejelentkezik a kliens, soha többet, bejegyzésével mi lesz?



Elavult rekordok törlése:

Számon tartja a rendszer, hogy az utolsó bejelentkezés óta mennyi idő telt el, ha meghaladja a beállítottat, akkor törli a zónafájlból.

FONTOS: kézzel létrehozottakat nem, csak akik dinamikus bejegyzéssel kerültek ide be. → alapértelmezés szerint nincs pipa a *Scavenge stale resource records* négyzetben

Elavult rekord azonnali törlése: Server / je / Scavenge Stale.....

Elévülési idő beállítása:

Ezen a lapon két beállítást tehetünk.

Frissítés nélküli időköz → eddig az időtartamig hiába jelentkezik be a kliens, nem frissül a bejegyzése (naponta 5* bejelentkezik, nem kellene mindig nézegetni - írogatni)

Frissítési időköz → Ha ez is letelik és a kliens még mindig nem jelentkezett be a rendszerbe, NA akkor kitörli. Ha ez idő tartam alatt bejelentkezik (8-14. napon), akkor már frissül a bejegyzés és indul a ketyegés előlről, tehát a frissítés nélküli időköztől.

Feladatsor DNS kiszolgáló

1. Telepítsd a DNS kiszolgálót a szerveredre!
2. Állíts be a számítógép elsődleges DNS utótagjának az `iskola.hu` tartománynevet!
3. Készíts **címkeresési** zónát `iskola.hu` néven!
4. Készíts **névkeresési** zónát a szerver alhálózatára!
5. Hozz létre **A** és **PTR** rekordokat a szerver azonosítására! (Ha még nem léteznek!)
6. Old meg, hogy a szerver **www.iskola.hu** néven is elérhető legyen!
7. Mindkét zónára vonatkozóan add meg, hogy fogadják a **dinamikus frissítéseket**!
8. Nézd meg a **DNS cache** tartalmát!
9. Nézd meg a **gyökérkiszolgálók beállításait**, illetve, hogy a DNS kiszolgáló mely **IP cím(ei)n fogad kéréseket**!
10. Állítsd be, hogy ha a DNS olyan kérést kap, amelyet nem tud kiszolgálni, **továbbítsa** a kapott kérést a **193.225.108.60** –as IP című NS számára!
11. A `kobold.hu` végződésű nevek feloldásával a **212.92.18.73**-as című gép foglalkozik. Ezeket a kéréseket rögtön neki továbbítsa a DNS kiszolgálód!
12. Takarítsd ki a régi erőforrásrekordokat! Állítsd be, hogy a DNS-ben **6 napig ne frissüljenek** a dinamikusan felvett rekordok, és, ha ez idő után **8 nappal** még nem frissítették rekordjaikat, töröljük őket!
13. Teszteld DNS kiszolgálód működését! Az XP gépedről az `nslookup` parancs segítségével kérd le a `www.iskola.hu` névhez tartozó IP címet!
14. A DNS Serverünk nem lát ki az Internetre, ezért a következő feladatokat a külső gépről végezd! (Ha a serverrel végeztetnénk ugyanezt látnánk, mert ő is kliensként kérné ezen adatokat)
 - állítsd be magadat `origo.hu`-nak!

- kérj zónalistázást néhány zónáról! (index.hu, om.hu, google.com, netacademia.net, stb..)
- Tanulmányozd akár a fenti, akár más tartományok SOA rekordját, MX rekordját!
- pingeld meg a citromail.hu-t, microsoft.com-ot! Mit tapasztalsz, mi ez a jelenség?

Készíts te is ilyen az iskola.hu zónán belül. (ugyanaz a név mutasson több IP-re)!