IPv6 alapok
az első lépések

Kunszt Árpád <arpad.kunszt@andrews.hu>
Andrews IT Engineering Kft.

Bemutatkozás

- Kunszt Árpád
  - Andrews IT Engineering Kft.
  - arpad.kunszt@andrews.hu

Miről lesz szó?

- Körkép
- IPv6 alapok
- IPv4-IPv6 együttélés
- Linux megoldások
- Összefoglaló

Körkép

- IPv4-es címek fogyása
  - elértünk az utolsó 5 A tartományhoz
    - Európára már csak 1 A tartomány jutott
- folyamatosan növekvő igény
  - okostelefonok, tabletek
  - virtualizáció, cloud szolgáltatások
- helyzet
  - gyorsuló terjedés, főleg kiszolgálói oldalon
  - pilot projektek

IPv6 gyorstalpaló

- új cím formátum
  - 8 x 16 bit = 128 bit
    - nem a teljes tartomány használható
  - szeparátor karakter: :
    - az IP címek és a portok között is ez maradt
  - prefix hossz
    - mint IPv4 esetében, pl: /64, /128
- tetszőlegesen sok IP cím vehető fel
  - akár kapcsolatonként is új

IPv6 címzés

- 2001:0db8:0000:00fd:0000:0000:b174:fc62
  - egy újabb érv a DNS használata mellett :-)
- egyszerűsítő szabályok
  - csoporton belüli vezető nullák elhagyhatók
  - csupa nulla csoportok összevonhatóak (::), ha egyértelmű az összevonás
  - 2001:0db8:0000:00fd:0000:0000:b174:fc62 → 2001:db8:0:fd:0:0:b174:fc62 → 2001:db8:0:fd::b174:fc62 vagy 2001:db8::fd:0:0:b174:fc62

IPv6 címzés

- hálózat megadása: 2001:db8::/32
  - IPv4: 172.29.16.0/24
- URL: http://[2001:db8::1]:8080/
- localhost: ::1
  - IPv4: 127.0.0.1
- minden cím: ::/0
  - IPv4: 0.0.0.0/0 ~ default gw célhálózata
    - IPv6 esetében a default gw általában: 2000::/3

IPv6 kiemelt címek

- link lokális címek: fe80::/10
  - link ~ helyi hálózat, broadcast domain
    - nem a hálózathoz való hozzáférés (van link?)
  - ~ MAC cím
    - benne van a MAC cím (kódolva)
- egyedi címek (ULA): fc00::/7
  - ~ helyi privát tartományok
    - IPv4: 192.168.0.0/16, 10.0.0.0/8, 172.16.0.0/12
  - korábban site-lokális címek: fec0::/10

IPv6 kiemelt címek

- globális címek (GAUA/AGUA): 2000::/3
  - publikus IP címek
- multicast címek: ff00::/8
  - IPv4: 224.0.0.0/4
  - all-node: ff02::1
    - ~ broadcast
  - all-router: ff02::2
  - solicited node: ff02::1:ffxx:xxxx
    - xxxxxx – megszólítandó gépek utolsó 24 bitje

IPv6 egyéb újdonságok

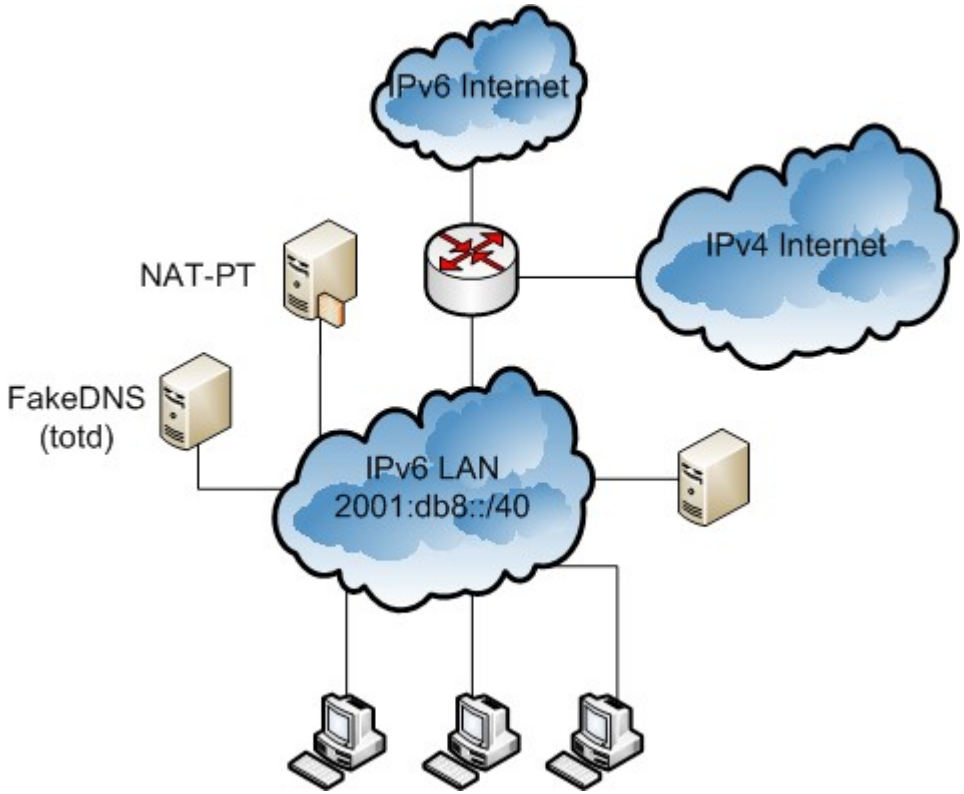
- csomagszórási módok
  - unicast
  - multicast
  - anycast
    - pl: a legközelebbi DNS szerver
- címkonfiguráció
  - statikus
  - SLAAC
  - DHCPv6

SLAAC

- új címkonfiguráció
  - **S**tate**L**ess **A**ddress **A**uto**C**onfiguration
  - automatikus hirdetések (RA)
    - hálózati információk
    - routing információk
    - DNS információk
  - tetszőlegesen sok hirdetést lehet használni
    - → tetszőlegesen sok hálózat konfigurálható fel
  - nagyon hosszú élettartam
    - 1 hét alapértelmezésben

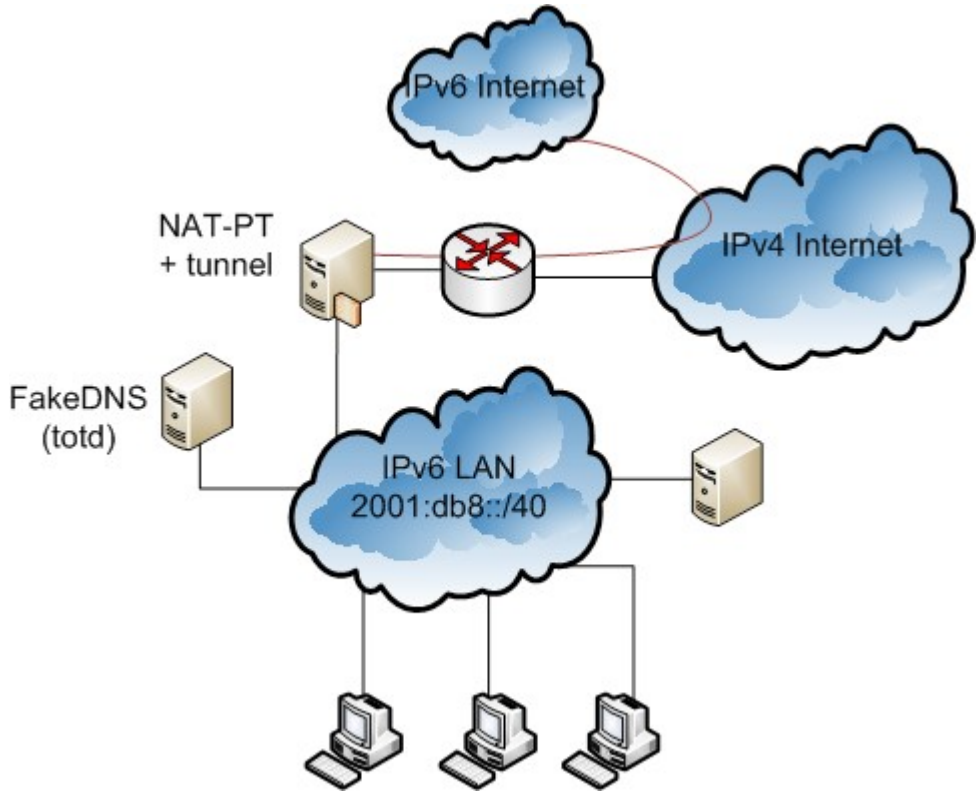
IPv4 – IPv6 együttélés

- IPv6-only belső hálózat, IPv6 uplink



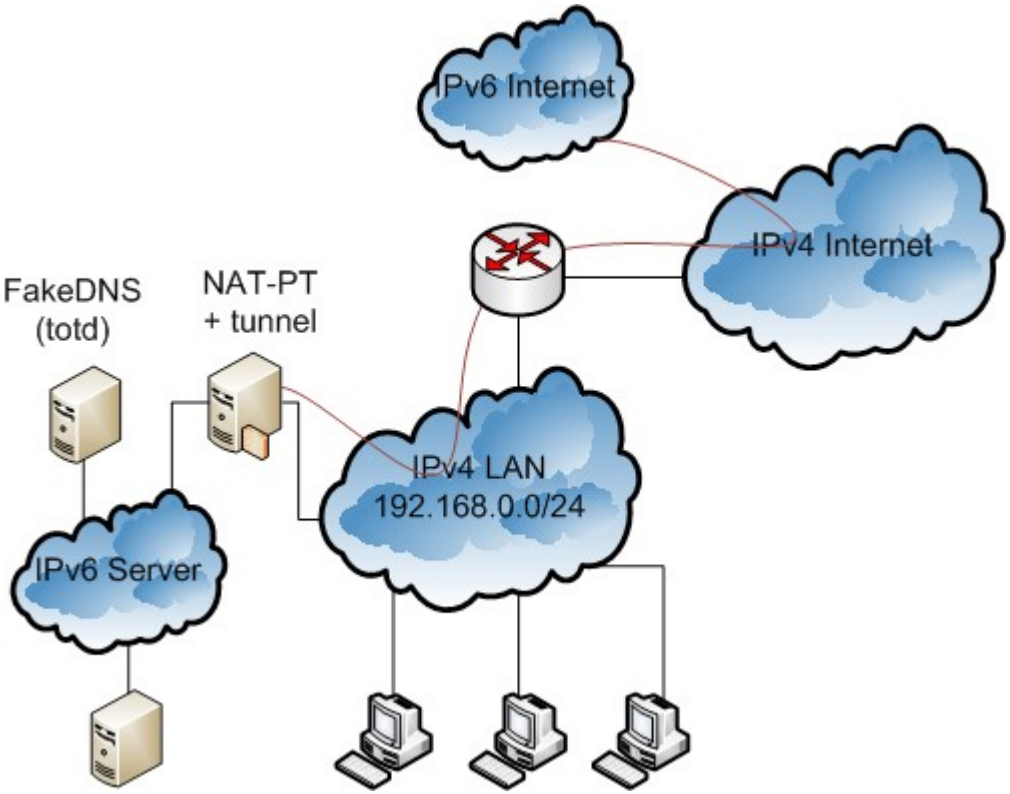
IPv4 – IPv6 együttélés

- IPv6-only belső hálózat, IPv4-only uplink



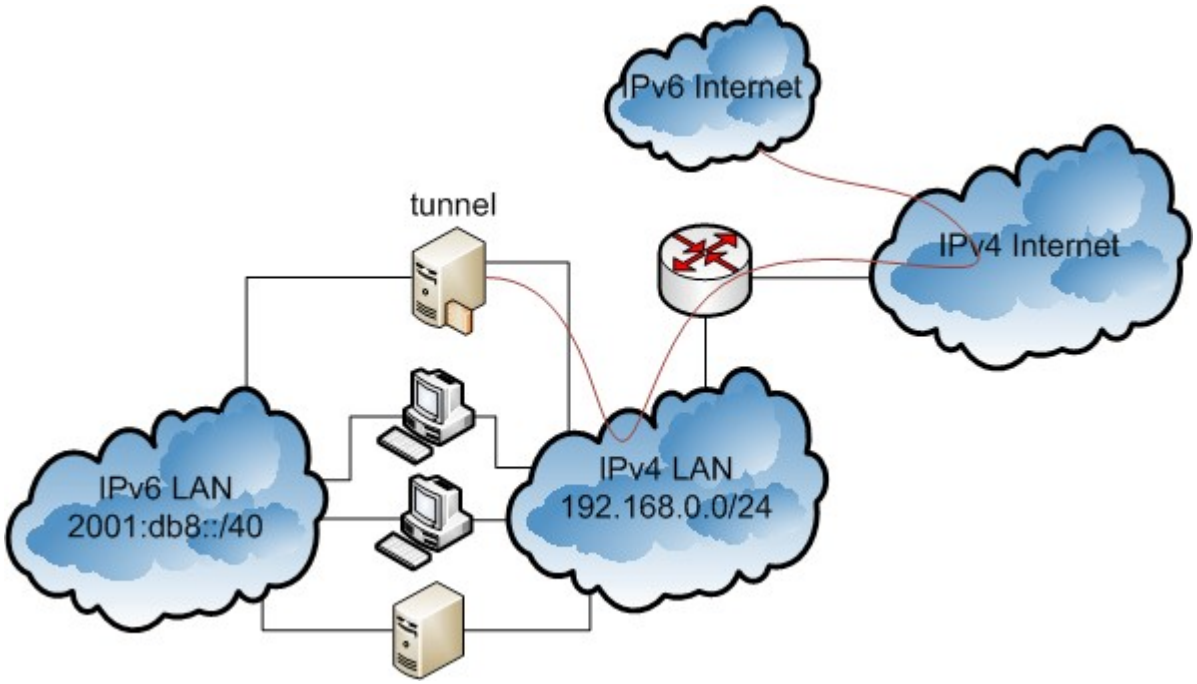
IPv4 – IPv6 együttélés

- IPv6-only szerverek



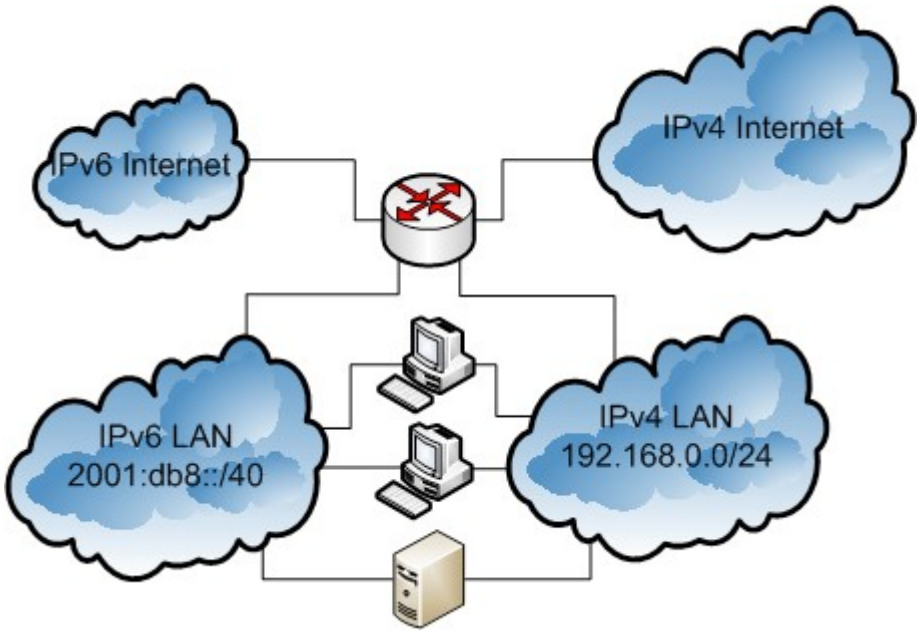
IPv4 – IPv6 együttélés

- hibrid node-ok



IPv4 – IPv6 együttélés

- hibrid node-ok, IPv6 uplink



Hibrid node-ok

- a node-ok egyszerre rendelkeznek IPv4 és IPv6 címekkel
- szoftver támogatás
  - Linux
  - FreeBSD, OpenBSD
  - Windows 7
    - XP: csökkentett támogatás, nem javasolt
- kliens szoftverek támogatása is kell
  - pl: Firefox, Thunderbird, NFSv4 stb.

Hibrid node-ok

- előny
  - könnyű kialakítani
    - elég csak a Linux-os szerveren konfigurálni
    - nem kell extra hardver/szoftver
  - probléma esetén lehet hova menekülni
    - megmarad a teljes IPv4-es hálózat
- hátrány
  - dupla munka (pl: tűzfalazás)
  - minimális IPv6 forgalom

Implementáció Linux alapokon

- IPv4 hálózat: feltesszük, hogy kész
- IPv6
  - címkonfiguráció: SLAAC
    - szoftver: radvd
  - tunnel: SIT
    - szoftver: iproute2
  - névfeloldás: DNS
    - szoftver: bind9
  - tűzfal: csomagszűrő + HTTP proxy
    - szoftver: ip6tables/netfilter + squid

radvd

- RA üzenetek szórása a helyi hálózaton
  - /etc/radvd.conf – konfigurációs állomány
  - radvdump – RA üzenetekből konfiguráció

```
interface eth0
{
        AdvSendAdvert on;
        MinRtrAdvInterval 3;
        MaxRtrAdvInterval 10;

        AdvDefaultPreference low;
        AdvHomeAgentFlag off;

        prefix 0:0:0:42::/64
        {
                AdvOnLink on;
                AdvAutonomous on;
                AdvRouterAddr off;

                Base6to4Interface eth1;

                AdvPreferredLifetime 120;
                AdvValidLifetime 300;
        };

};
```

radvd

- be kell kapcsolni az IPv6 forwardingot
  - /etc/sysctl.conf
  - disztribúciótól függ, hogy ezt megteszi-e az initscript

```
# Uncomment the next line to enable packet forwarding for IPv6
#  Enabling this option disables Stateless Address Autoconfiguration
#  based on Router Advertisements for this host
net.ipv6.conf.all.forwarding=1
```

SIT tunnel

- IPv6-in-IPv4 tunnel
  - szokás 6to4-nek is hívni
    - 6rd is hasonló, de nem ugyanaz!
  - a legközelebbi tunnelbrokerhez csatlakozunk
    - ~ anycast IPv4
  - nem garantált, hogy a teljes IPv6 hálózatot látni fogjuk!
  - disztribúciónként eltérő konfiguráció
    - most az általános megoldást mutatom meg

SIT tunnel

- iproute2 parancsok
- a helyi IPv6-os hálózatnak a ${localip6}::/48-ba kell esnie
  - → egy csoportot tetszőlegesen használhatunk (itt most 0 lett)

```
localip6="2002:$(printf "%x%x:%x%x" $(echo "$localip4" | tr '.' ' '))"

ip tunnel add sit0 mode sit remote any local any dev ${inet_iface}
ip link set dev sit0 up
ip -6 addr add ${localip6}::1/16 dev sit0
ip -6 add add ${localip6}::3/64 dev ${lan_iface}
ip route add 2000::/3 via ::192.88.99.1 dev sit0 metric 1
```

bind9

- új DNS rekordok
  - AAAA az eddigi A rekord helyett
- sipcalc
  - -r megadja a reverse DNS bejegyzést
  - használható, mint az ipcalc is

bind9

- named.conf
- a zone értéke a hálózati cím fordított sorrendben (~IPv4)
  - ip6.arpa ↔ in-addr.arpa

```
zone "lok" {
        type master;
        file "/etc/bind/db.lok";
};

zone "2.4.0.0.f.0.3.0.0.0.a.0.2.0.0.2.ip6.arpa" {
        type master;
        file "/etc/bind/db.2.4.0.0.f.0.3.0.0.0.a.0.2.0.0.2.ip6.arpa";
};
```

bind9

- forward zóna
  - ahogy eddig is, csak az AAAA rekord új

```
$TTL    86400
@       IN      SOA     lok. root.lok. (
                        20120427                ; Serial
                        604800                  ; Refresh
                        86400                   ; Retry
                        2419200                 ; Expire
                        604800                  ; Negative Cache TTL
)

@       IN      NS              server
@       IN      MX      5       server

server  IN      A               10.42.0.1
server  IN      AAAA            2002:a00:30f:42::1

client  IN      A               10.42.0.103
client  IN      AAAA            2002:a00:30f:42:a00:27ff:feb5:4677
```

bind9

- reverse zóna
  - PTR: fordított sorrendben node címek
  - $ORIGIN: csoportosít → rövidebb címek
    - de úgyis mindenki generálni fogja :-)

```
$TTL    86400
@       IN      SOA     lok. root.lok. (
                        20120427                ; Serial
                        604800                  ; Refresh
                        86400                   ; Retry
                        2419200                 ; Expire
                        604800                  ; Negative Cache TTL
)

@       IN      NS              server.lok.

1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0  IN      PTR     server.lok.
7.7.6.4.5.b.e.f.f.f.7.2.0.0.a.0                                  IN      PTR     client.lok.
```

csomagszűrő

- ip6tables
  - nincs NAT tábla
    - ideológiai okok miatt maradt ki
    - → nincs REDIRECT sem → ennyit a transzparens proxyzásról és a load-balancingról
      - vannak kerülő megoldások, de azok bonyolultak, nehézkesek és nem kiforrottak
    - már készül hivatalos IPv6 NAT implementáció
      - http://hup.hu/cikkek/20111127/ipv6_nat_implementacion_dolgoznak_a_netfilter_fejlesztok

csomagszűrő

- egyszerű példa
  - bejövő SMTP elfogadása
  - kimenő forgalom csak HTTPS engedélyezett

```
ip6tables -A INPUT -i eth1 -p tcp --sport 1024: --dport 25 -j ACCEPT
ip6tables -A FORWARD -p tcp -s 2002:a00:30f:42::/64 --sport 1024: --dport https -j ACCEPT
ip6tables -P FORWARD DROP
```

squid

- nincs transzparens proxy
  - → közvetlenül kell megadni a klienseken
- alapértelmezésben a squid figyel az IPv6-os címeken
  - nagy eséllyel csak az ACL-eket kell kibővíteni
    - itt nincs megkülönböztetés az IPv4 és IPv6 között

Összefoglaló

- az IPv6 támogatás ma még nem létkérdés
  - de a trendekből látszik, hogy ez már középtávon sem lesz igaz
- teljesen más felépítés, logika
  - szokni kell
- hardver, szoftver támogatottság kérdéses
  - ez idővel elsimul

Összefoglaló

- minden migráció egyedi eset
  - nincsenek általános, előre dobozolt megoldások
  - tartsunk fenn „menekülő utat”
  - haladjunk kis lépésekben, nem kell azonnal megváltani a világot
  - kérjünk szakmai segítséget
- komoly biztonsági kérdések
  - az eddigi megoldások nem biztos, hogy használhatók

Köszönöm a figyelmet!