

IPv6-foredrag

Pent brukt 19-åring

Trond Endrestøl

Fagskolen Innlandet, IT-avdelingen

30. september 2014



Foredragets filer I

- Foredraget er mekket ved hjelp av [GNU Emacs](#), [AUCTeX](#), [pdfTeX](#) fra [MiKTeX](#), [L^AT_EX](#)-dokumentklassa [beamer](#), [Dia](#), [Inkscape](#), [Wireshark](#), [Subversion](#), [TortoiseSVN](#) og [Adobe Reader](#)
- Hovedfila bærer denne identifikasjonen:
\$Ximalas: trunk/ipv6-foredrag.tex 105 2014-09-30
15:40:40Z trond \$
- Driverfila for denne PDF-fila bærer denne identifikasjonen:
\$Ximalas: trunk/ipv6-foredrag.handout.4on1.tex 78
2013-12-04 09:53:24Z trond \$
- Copyright © 2014 Trond Endrestøl
- Dette verket er lisensiert med: [Creative Commons](#),
[Navngivelse-DelPåSammeVilkår 3.0 Norge](#) (CC BY-SA 3.0)



Foredragets filer I

- Filene til foredraget er tilgjengelig gjennom:
 - Subversion: `svn co svn://svn.ximalas.info/ipv6-foredrag`
 - Web: svnweb.ximalas.info/ipv6-foredrag
 - Begge metodene er tilgjengelig med både IPv4 og **IPv6**
- [ipv6-foredrag.foredrag.pdf](#) vises på lerretet
- [ipv6-foredrag.handout.pdf](#) er mye bedre for publikum å se på egenhånd
- [ipv6-foredrag.handout.2on1.pdf](#) og [ipv6-foredrag.handout.4on1.pdf](#) er begge velegnet til utskrift
- *.169.pdf-filene er i 16:9-format
- *.1610.pdf-filene er i 16:10-format



Oversikt av hele foredraget

Del 1: Kort om IPv6

- 1 Hva er IPv6?
- 2 Hvorfor trenger vi IPv6?
- 3 Midlertidige tiltak
- 4 Antall adresser
- 5 Andre nyttige ting ved IPv6
- 6 Hvorfor brukes ikke IPv6?
- 7 IPv6 ved Fagskolen Innlandet



Oversikt av hele foredraget

Del 2: IPv6-header

- 8 IPv6-header
 - Flow Label
- 9 Utvidelsesheadere
 - Hop-by-hop Options Header
 - Destination Options Header
 - Routing Header
 - Fragment Header
 - Authentication Header
 - Encapsulating Security Payload
 - Mobility Header



Oversikt av hele foredraget

Del 4: Grunnleggende om adresser

- 12 Grunnleggende om adresser
- 13 Adressedemo
- 14 MAC-48-adresser
- 15 Modda IEEE EUI-64-format
- 16 Manuell grensesnittidentifikator
- 17 Tilfeldig grensesnittidentifikator
- 18 Spesialadresser
- 19 Duplicate Address Detection — DAD



Oversikt av hele foredraget

Del 3: IPv6 over Ethernet

- 10 IPv6 over Ethernet
- 11 IPv6 over andre lag-2-typer



Oversikt av hele foredraget

Del 5: Adresstyper

- 20 Adresstyper
- 21 Link-local-adresser
- 22 Site-local-adresser
- 23 Offentlige unicast-adresser
- 24 Unike, lokale, aggregerbare adresser
- 25 Anycast-adresser
- 26 Multicast-adresser



Oversikt av hele foredraget

Del 6: DNS

27 AAAA og PTR

28 A6



Oversikt av hele foredraget

Del 8: Neighbor Discovery

44 Router Solicitation

45 Router Advertisement

46 Neighbor Solicitation

47 Neighbor Advertisement

48 Redirect



Oversikt av hele foredraget

Del 7: ICMPv6

29 ICMPv6

30 Multicast Listener Discovery

31 Neighbor Discovery

32 Router Renumbering

33 Node Information

34 Inverse Neighbor Discovery

35 Version 2 Multicast Listener Report

36 Mobile IPv6

37 SEcure Neighbor Discovery (SEND)

38 Experimental Mobility Type

39 Multicast Router Discovery

40 FMIPv6

41 RPL Control Message

42 ILNPv6 Locator Update Message

43 Duplicate Address



Oversikt av hele foredraget

Del 9: DHCPv6

49 DHCPv6

50 Meldinger

51 DHCP Unique Identifier

52 Identity association

53 Identity association identifier



Oversikt av hele foredraget

Del 10: Avansert multicast

- 54 Multicastflaggene
- 55 Når T er satt til 1
- 56 Når PT er satt til 11
- 57 Når RPT er satt til 111



T. Endrestøl (FSI/IT)

IPv6-foredrag

30. september 2014

13 / 178

Oversikt av hele foredraget

Del 12: Noen RFC-er om IPv6

- 60 Noen RFC-er om IPv6



T. Endrestøl (FSI/IT)

IPv6-foredrag

30. september 2014

15 / 178

Oversikt av hele foredraget

Del 11: Konfigurasjon av IPv6

- 58 Cisco IOS
 - IPv6-routing
 - ACL-er
 - DHCPv6
- 59 OS-konfig



T. Endrestøl (FSI/IT)

IPv6-foredrag

30. september 2014

14 / 178

Del I

Kort om IPv6



T. Endrestøl (FSI/IT)

IPv6-foredrag

30. september 2014

16 / 178

Oversikt over del 1: Kort om IPv6

- 1 Hva er IPv6?
- 2 Hvorfor trenger vi IPv6?
- 3 Midlertidige tiltak
- 4 Antall adresser
- 5 Andre nyttige ting ved IPv6
- 6 Hvorfor brukes ikke IPv6?
- 7 IPv6 ved Fagskolen Innlandet



Kort om IPv6

Hvorfor trenger vi IPv6?

- Mobilmarkedet viser en enorm vekst: smarttelefoner, nettbrett m.m.
- Verden går tom for offentlige IPv4-adresser
- «IPokalypsen» er her!
- IANA gikk tom 3. februar 2011
 - APNIC gikk tom 19. april 2011
 - RIPE gikk tom 14. september 2012
 - LACNIC gikk tom 10. juni 2014
- Dersom disse RIR-ene oppfører seg pent:
 - ARIN kan holde på til mars 2015
 - AFRINIC kan holde på til juni 2019(!)



Kort om IPv6

Hva er IPv6?

- En lag-3-protokoll ment å erstatte IPv4
- Har eksistert siden desember 1995, først spesifisert i RFC 1883
- Enkel grunnheader med fast lengde
- Flere utvidelsesheadere, riktig rekkefølge er viktig
- 128-bit adresser
- Ny versjon av ICMP: ICMPv6
- ARP og RARP for IPv6 er en del av ICMPv6
 - Ikke nødvendig med ekstra lim for adressene i lagene 2 og 3
- Ny versjon av DHCP: DHCPv6
- Automatisk adressekonfigurasjon uten bruk av DHCPv6



Kort om IPv6

Midlertidige tiltak

- Midlertidige tiltak:
 - Network Address Translation,
 - Carrier-Grade NAT, og
 - Shared Address Space (100.64.0.0/10)
- Dette er bare støttebandasje med kort utløpstid
- Glem det!
- Ende-til-ende-konnektivitet oppnås best uten noen former for adresseoversettelse



Kort om IPv6

Antall adresser

- Totalt antall IPv6-adresser:
- $2^{128} = 340.282.366.920.938.463.463.374.607.431.768.211.456$
- Bare 1/8 kan brukes til offentlige unicast-adresser:
- $2^{125} = 42.535.295.865.117.307.932.921.825.928.971.026.432$
- Fortsatt er det mange flere IPv6-unicast-adresser enn det er IPv4-adresser:
- $2^{32} = 4.294.967.296$
- Mindre enn **3.702.258.688** IPv4-adresser kan bli brukt som offentlige IPv4-unicast-adresser
- Se Tronds utregning fra juli 2012: <http://ximalas.info/2012/07/20/how-many-ipv4-addresses-are-there/>



Kort om IPv6

Andre nyttige ting ved IPv6

- Hierarkisk adressestruktur
- Enklere planlegging av subnett sammenlignet med IPv4
 - De fleste IPv6-subnett bruker et 64-bit prefiks
 - Autokonfigurasjon *krever* et 64-bit prefiks
 - Fast prefikslengde på 64 bit er *ikke* et absolutt krav
 - DHCPv6 eller manuell konfigurasjon brukes når prefikslengda er ulik 64 bit



Kort om IPv6

Andre nyttige ting ved IPv6

- Kortere rutingtabeller
 - Uninett annonserer disse IPv4-subnettene med BGP:

78.91.0.0/16,	128.39.0.0/16,	129.177.0.0/16,
129.240.0.0/15,	129.242.0.0/16,	144.164.0.0/16,
151.157.0.0/16,	152.94.0.0/16,	156.116.0.0/16,
157.249.0.0/16,	158.36.0.0/14,	161.4.0.0/16,
193.156.0.0/15,	192.111.33.0/24,	192.133.32.0/24,
	192.146.238.0/23	
 - Uninett trenger bare å annonsere dette IPv6-prefikset:
 - 2001:700::/32



Kort om IPv6

Andre nyttige ting ved IPv6

- Sjekksum er overlatt til høyere og lavere lag
- Fragmentering skal gjøres hos avsender, og ikke underveis
 - Avsender må sjekke veien lengre fremme og måle smaleste krøttesti
 - Path Maximum Transmission Unit Discovery (Path MTU, PMTUD)
- IPsec ble spesifisert som en del av IPv6
 - Finnes også for IPv4
 - Må konfigureres før den begynner å virke
 - Tilbyr:
 - Kryptert overføring (ESP), og/eller
 - Bekreftelse av avsenders identitet og beskyttelse mot gjentakelse («replay») (AH)
 - Ble omgjort fra krav til anbefaling for IPv6 av [RFC 6434](#)



Kort om IPv6

Hvorfor brukes ikke IPv6?

- Markedskreftene bestemmer
- «Vente-og-se»-holdning
- Stikker hodet ned i sanda
- Store selskaper:
 - Kjøper opp små selskaper og hamstrer IPv4-blokker
 - Kjøper IPv4-blokker på ettermarkedet/konkursbo:
 - Microsoft → \$7,5 mill. → Nortel → 666.624 IPv4-adresser → Microsoft



Kort om IPv6

Hvorfor brukes ikke IPv6?

- Telebransjen satser fortsatt hardt på IPv4:
 - (Edge) NAT i CPE (RFC 1631)
 - Carrier-Grade NAT i stamnett (RFC 6264)
 - Shared Address Space etter behov i stamnett (RFC 6598)
- Før eller siden blir CGN for kostbart og komplisert å vedlikeholde
- 3G og 4G/LTE klarer kanskje å øke IPv6-preset (RFC 6459)
- IPv6 er det eneste tilgjengelige og realistiske alternativet til IPv4



Kort om IPv6

IPv6 ved Fagskolen Innlandet

- 1994: Tildelt 128.39.174.0/24 av Uninett
- 1. juni 2005: Ny IT-ansvarlig, yours truly
- Høsten 2005: Fikk reservert IPv4-serien 128.39.172.0/23
- Påska 2006: Fikk reservert IPv6-serien 2001:700:1100::/48
- Før og etter pinsehelga 2006: Fiberlinjer fra serverrommet og til sentrale punkter i hver etasje i hovedbygningen
- Sommeren 2006: Nytt Cisco-gear som Catalyst 3560G og 2960 (Cisco IOS 12.2(25)SEB4)
 - 128.39.46.8/30 ble linknettet mellom HiG/Uninett og FSI
 - 128.39.46.9 brukes ved HiG
 - 128.39.46.10 brukes ved FSI
 - 128.39.174.0/24 ble delt opp i flere subnett og satt opp som servernett og ansattnett, m.m.
 - 128.39.172.0/24 ble delt opp i flere subnett og satt opp som nett for datalab
 - 128.39.173.0/24 ble satt opp for klienter på trådløst studentnett



Kort om IPv6

IPv6 ved Fagskolen Innlandet

- 6. september 2006: IPv6-linknettet 2001:700:0:11D::/64 ble aktivert mellom HiG/Uninett og FSI
 - 2001:700:0:11D::1 brukes ved HiG
 - 2001:700:0:11D::2 brukes ved FSI
- Samme dag ble IPv6 innført for FSI-VLAN-ene 20, 30, 70 og 80:
 - FSI-VLAN 20: 2001:700:1100:1::/64 (ytre servernett)
 - FSI-VLAN 30: 2001:700:1100:2::/64 (indre servernett)
 - FSI-VLAN 70: 2001:700:1100:3::/64 (IT-kontornett)
 - FSI-VLAN 80: 2001:700:1100:4::/64 (IT-lekenett)
- Andre FSI-VLAN fikk IPv6 i ukene og månedene etterpå
- Sommeren 2007: Genererte og frivillig registrerte ULA-serien FD5C:14CF:C300::/48
 - Brukes i interne FSI-VLAN som tidligere bare brukte RFC 1918-adresser
 - Fikk første HP-skriver med IPv6-støtte og ville bruke IPv6
 - Noen år senere: IPv6-adresser på kantswitchene med Cisco IOS 12.2(40)SE



Kort om IPv6

IPv6 ved Fagskolen Innlandet

- Høsten 2010: Enda en IPv4-serie ble innført: 128.39.194.0/24
 - 128.39.194.0/24 brukes nå til datalab etter samme mønster som for den gamle 128.39.172.0/24
 - 128.39.172.0/23 brukes nå for inntil 508 IPv4-klienter på trådløst studentnett
- Sommeren 2014: Tok i bruk nye linknett fordi fig-gsw.fig.ol.no ble tilkoblet gjovik-gw1.uninett.no
 - IPv4-linknett: 128.39.70.168/30
 - 128.39.70.169 brukes ved HiG
 - 128.39.70.170 brukes ved FSI
 - IPv6-linknett: 2001:700:0:8074::/64
 - 2001:700:0:8074::1 brukes ved HiG
 - 2001:700:0:8074::2 brukes ved FSI



Kort om IPv6

IPv6 ved Fagskolen Innlandet

- Alle FSI-VLAN har både IPv4- og IPv6-adresser (dual-stack)
- FSI-VLAN med offentlige IPv4-adresser bruker offentlige IPv6-adresser fra 2001:700:1100::/48-serien
- FSI-VLAN med private IPv4-adresser (RFC 1918) bruker private IPv6-adresser fra FD5C:14CF:C300::/48-serien
- Private adresser brukes for alt utstyr som ikke har behov for internettforbindelse:
 - Switcher
 - Med unntak for kjerneswitchen som er L3-router for nettverket ved FSI
 - Basestasjoner og WLAN-kontroller
 - Før omlegginga til OFK-nettene
 - UPS-er
 - Skrivere
 - VPN-klienter



Kort om IPv6

IPv6 ved Fagskolen Innlandet

- Oppland FK (OFK) har ingen planer om å innføre IPv6
- Hordaland FK har satt en IPv6-adresse på webserveren deres, 2a02:20a0:0:3::81:130
- I dag er de fleste brukere ved FSI kasta over til OFK-nettene
- Dette skjedde etter ombygginga i 2011–2012
- Andreklasse data er velsigna med å kunne velge mellom FSI- og OFK-nettene
- Andreklasse data velger som regel det førstnevnte, vanligvis FSI-VLAN 48 som tilbyr 128.39.194.192/27 og 2001:700:1100:8008::/64
- Førsteklasse data ønsker det samme tilbudet; så vi får se ...



Del II

IPv6-header



Oversikt over del 2: IPv6-header I

8 IPv6-header

- Flow Label

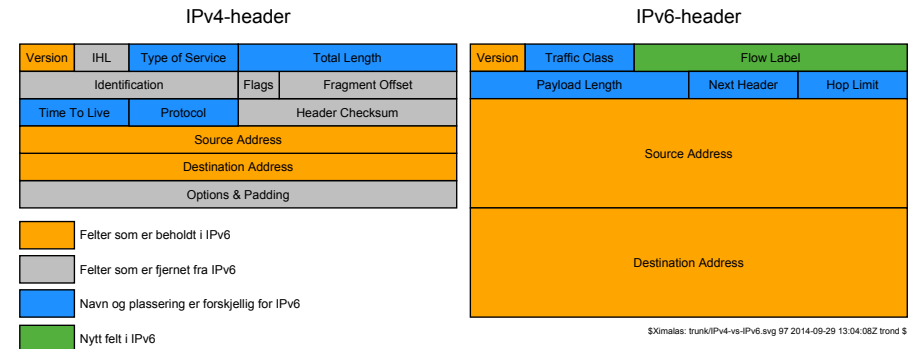
9 Utvidelsesheadere

- Hop-by-hop Options Header
- Destination Options Header
- Routing Header
- Fragment Header
- Authentication Header
- Encapsulating Security Payload
- Mobility Header

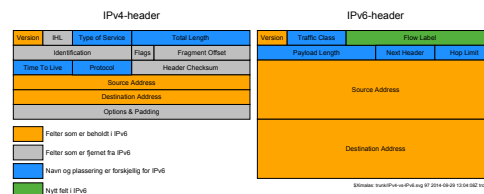


IPv6-header

IPv6-header



IPv6-header



- Next Header (8 bit) er det samme som Protocol i IPv4
- Hop Limit (8 bit) er det samme som Time To Live i IPv4
- Avsender og mottaker er 128-bit IPv6-adresser
- IPv4-feltene Internet Header Length (IHL), Identification, Flags, Fragment Offset, Header Checksum, Options og Padding, er enten fjernet for godt eller flyttet til egne utvidelsesheadere
- Versjonsfeltet (4 bit) settes til 0110
- Traffic Class (8 bit) er det samme som Type of Service i IPv4
- Flow Label (20 bit) er et nytt felt, se neste slide
- Payload Length (16 bit) er det samme som Total Length i IPv4



- IPv6-headeren er dobbelt så stor som IPv4-headeren (40/20 oktetter)
- IPv6-headeren har færre felt enn IPv4-headeren
- De utelatte feltene er i stor grad flyttet over til egne utvidelsesheadere



IPv6-header

Flow Label

- Flow Label-feltet kan brukes av sanntidsapplikasjoner
- Flow Label-verdien angir pakker som tilhører samme sesjon
- Routere bør videresende pakker med samme verdi i Flow Label-feltet fra samme avsender på samme grensesnitt, slik at rekkefølgen bevares
- Verdien 0 (null) brukes for individuelle pakker
- Routere bør videresende pakker med 0 i Flow Label-feltet fra samme avsender på samme grensesnitt, slik at rekkefølgen bevares
- Tilfeldig valgte verdier brukes for pakker som hører sammen
- Flow Label-feltet kan også brukes til å smugle data sammen med legitim trafikk, eller merke slik trafikk, se avsnitt 6.1 i [RFC 6437](#)
- Se [RFC 2460](#), [RFC 3595](#), [RFC 6294](#), [RFC 6436](#) og [RFC 6437](#)

Utvidelsesheadere

Hop-by-hop Options Header

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Next Header | Hdr Ext Len |                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     |                                     |
|                                     |                                     |
|                                     |                                     |
|                                     |                                     |
|                                     |                                     |
|                                     |                                     |
|                                     |                                     |
|                                     |                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

- Bla, bla, bla

Utvidelsesheadere

- Utvidelsesheaderne finnes i stort antall:
 - 1 Hop-by-hop Options Header
 - 2 Destination Options Header
 - 3 Routing Header
 - 4 Fragment Header
 - 5 Authentication Header
 - 6 Encapsulating Security Payload
 - 7 Mobility Header
- Se [RFC 2460](#), [RFC 4302](#), [RFC 4303](#) og [RFC 6275](#)

Utvidelsesheadere

Destination Options Header

```

+-----+-----+-----+-----+-----+-----+-----+-----+
| Next Header | Hdr Ext Len |                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+
|
|
|
|
|
|
|
|
+-----+-----+-----+-----+-----+-----+-----+-----+

```

- Bla, bla, bla

Utvidelsesheadere

Routing Header

```
+-----+-----+-----+-----+-----+-----+-----+-----+
| Next Header | Hdr Ext Len | Routing Type | Segments Left |
+-----+-----+-----+-----+-----+-----+-----+-----+
|
|
|                                     type-specific data
|
|
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- Bla, bla, bla



Utvidelsesheadere

Fragment Header

```
+-----+-----+-----+-----+-----+-----+-----+-----+
| Next Header | Reserved  | Fragment Offset | Res|M|
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Identification
|
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- Bla, bla, bla



Utvidelsesheadere

Authentication Header

```
0 1 2 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
| Next Header | Payload Len | RESERVED |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Security Parameters Index (SPI) |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Sequence Number Field |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Integrity Check Value-ICV (variable) |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- Bla, bla, bla



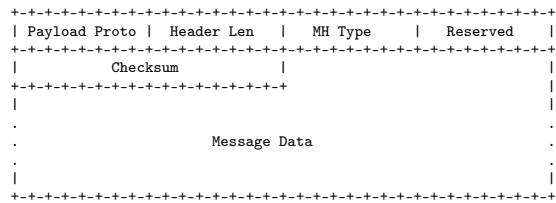
Utvidelsesheadere

Encapsulating Security Payload

```
0 1 2 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
| Security Parameters Index (SPI) |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Sequence Number |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Payload Data* (variable) |
+-----+-----+-----+-----+-----+-----+-----+-----+
|
|
| +-----+-----+-----+-----+-----+-----+-----+-----+
| | Padding (0-255 bytes) |
+-----+-----+-----+-----+-----+-----+-----+-----+
| | Pad Length | Next Header |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Integrity Check Value-ICV (variable) |
+-----+-----+-----+-----+-----+-----+-----+-----+
|
|
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- Bla, bla, bla





- Bla, bla, bla

Oversikt over del 3: IPv6 over Ethernet I

- 11 IPv6 over andre lag-2 typer

IPv6 over Ethernet

IPv6 over Ethernet

- [RFC 2464](#) definerer frameformatet for IPv6-datagrammer over Ethernet
- IPv6-datagrammer fraktes i standard Ethernetformat, [RFC 894](#)
- Først angis mottakerens MAC-48-adresse
- Deretter angis avsenders MAC-48-adresse
- Frametypen settes til 86DD (heksadesimalt)
- Deretter følger IPv6-header og resten av datagrammet
- Standard MTU for IPv6 over Ethernet er 1500 oktetter
- Minste tillatte MTU for IPv6 er 1280 oktetter
- Er største tilgjengelige MTU mindre enn 1280 oktetter, så må lagene under IPv6 sørge for fragmentering og sammensetting av IPv6-datagrammene ([RFC 2460](#))

Programmet [Wireshark](#) fremstilte følgende lag-2-informasjon om en utsendt IPv6-pakke:

```
Ethernet II, Src: AsustekC_f2:72:40 (00:26:18:f2:72:40), Dst: Cisco_77:14:57 (00:17:e0:77:14:57)
  Destination: Cisco_77:14:57 (00:17:e0:77:14:57)
    Address: Cisco_77:14:57 (00:17:e0:77:14:57)
      ....0. .... = LG bit: Globally unique address (factory default)
      ....0. .... = IG bit: Individual address (unicast)
  Source: AsustekC_f2:72:40 (00:26:18:f2:72:40)
    Address: AsustekC_f2:72:40 (00:26:18:f2:72:40)
      ....0. .... = LG bit: Globally unique address (factory default)
      ....0. .... = IG bit: Individual address (unicast)
  Type: IPv6 (0x86dd)
```

- Presentert som heksadesimale oktetter/byter:
- 00 17 E0 77 14 57 00 26 18 F2 72 40 86 DD
 - 00 17 E0 77 14 57 er MAC-48-adressa til mottakeren, routeren
 - 00 26 18 F2 72 40 er MAC-48-adressa til avsenderen, klienten
 - 86 DD angir at et IPv6-datagram følger etter i lag 3

Del IV

Grunnleggende om adresser

- FDDI: [RFC 2467](#)
- Token Ring: [RFC 2470](#)
- Non-Broadcast Multiple Access (NBMA) networks: [RFC 2491](#)
- ATM: [RFC 2492](#)
- ARCnet: [RFC 2497](#)
- Frame Relay: [RFC 2590](#)
- IEEE 1394 (FireWire): [RFC 3146](#)
- Low-Power Wireless Personal Area Networks (6LoWPAN): [RFC 4919](#)
- Point-to-point protocol (PPP): [RFC 5072](#)
- Brevduer: [RFC 6214](#), basert på [RFC 1149](#)

Oversikt over del 4: Grunnleggende om adresser I

- 12 Grunnleggende om adresser
- 13 Adressedemo
- 14 MAC-48-adresser
- 15 Modda IEEE EUI-64-format
- 16 Manuell grensesnittidentifikator
- 17 Tilfeldig grensesnittidentifikator
- 18 Spesialadresser
- 19 Duplicate Address Detection — DAD

Grunnleggende om adresser

- 128 bit
- Heksadesimal notasjon
- 16 og 16 bit grupperes og adskilles med kolon
- Ledende nuller kan sløyfes
- To eller flere *sammenhengende* 16-bitblokker med nuller kan slås sammen til :: (dobbelkolon), bare én gang pr. adresse
- Prefikslengde angis ved å sette på en skråstrek og oppgi riktig antall av signifikante bit fra venstre mot høyre i adressa
 - Dette er helt likt CIDR-notasjon for IPv4 ([RFC 4632](#))

Grunnleggende om adresser

Adressedemo

- Uninett:
2001:0700:0000:0000:0000:0000:0000
- FSI:
2001:0700:1100:0000:0000:0000:0000
- IT-avdelingen@FSI:
2001:0700:1100:0003:0000:0000:0000
- Tronds D531:
2001:0700:1100:0003:0221:70FF:FE73:686E

Grunnleggende om adresser

Adressedemo: Hierarkisk struktur

- Uninett:
2001:0700:0000:0000:0000:0000:0000
- FSI:
2001:0700:1100:0000:0000:0000:0000
- IT-avdelingen@FSI:
2001:0700:1100:0003:0000:0000:0000
- Tronds D531:
2001:0700:1100:0003:0221:70FF:FE73:686E

Grunnleggende om adresser

Adressedemo: La oss forenkle adressene

- Uninett:
2001:0700:0000:0000:0000:0000:0000
- FSI:
2001:0700:1100:0000:0000:0000:0000
- IT-avdelingen@FSI:
2001:0700:1100:0003:0000:0000:0000
- Tronds D531:
2001:0700:1100:0003:0221:70FF:FE73:686E

Grunnleggende om adresser

Adressedemo: Ledende nuller

- Uninett:
2001:0700:0000:0000:0000:0000:0000
- FSI:
2001:0700:1100:0000:0000:0000:0000
- IT-avdelingen@FSI:
2001:0700:1100:0003:0000:0000:0000
- Tronds D531:
2001:0700:1100:0003:0221:70FF:FE73:686E



Grunnleggende om adresser

Adressedemo: Fjernet ledende nuller

- Uninett:
2001:700:0:0:0:0:0:0
- FSI:
2001:700:1100:0:0:0:0:0
- IT-avdelingen@FSI:
2001:700:1100:3:0:0:0:0
- Tronds D531:
2001:700:1100:3:221:70FF:FE73:686E



Grunnleggende om adresser

Adressedemo: La oss forenkle litt til

- Uninett:
2001:700:0:0:0:0:0:0
- FSI:
2001:700:1100:0:0:0:0:0
- IT-avdelingen@FSI:
2001:700:1100:3:0:0:0:0
- Tronds D531:
2001:700:1100:3:221:70FF:FE73:686E



Grunnleggende om adresser

Adressedemo: To eller flere sammenhengende 16-bitblokker med bare 0

- Uninett:
2001:700:0:0:0:0:0:0
- FSI:
2001:700:1100:0:0:0:0:0
- IT-avdelingen@FSI:
2001:700:1100:3:0:0:0:0
- Tronds D531:
2001:700:1100:3:221:70FF:FE73:686E



Grunnleggende om adresser

Adressedemo: Erstattet med dobbelkolon

- Uninett:
2001:700::
- FSI:
2001:700:1100::
- IT-avdelingen@FSI:
2001:700:1100:3::
- Tronds D531:
2001:700:1100:3:221:70FF:FE73:686E



Grunnleggende om adresser

Adressedemo: Kompakt form

- Uninett:
2001:700::
- FSI:
2001:700:1100::
- IT-avdelingen@FSI:
2001:700:1100:3::
- Tronds D531:
2001:700:1100:3:221:70FF:FE73:686E



Grunnleggende om adresser

Adressedemo: Vis prefikslengde

- Uninett:
2001:700::/32
- FSI:
2001:700:1100::/48
- IT-avdelingen@FSI:
2001:700:1100:3::/64
- Tronds D531:
2001:700:1100:3:221:70FF:FE73:686E/128



Grunnleggende om adresser

Adressedemo: Kompakte adresser med prefikslengde

- Uninett:
2001:700::/32
- FSI:
2001:700:1100::/48
- IT-avdelingen@FSI:
2001:700:1100:3::/64
- Tronds D531:
2001:700:1100:3:221:70FF:FE73:686E/128



Grunnleggende om adresser

MAC-48-adresser

- MAC-48-adresser har følgende oppbygging, gitt av [IEEE 802-2001](#):
 - CC:cc:cc:nn:nn:nn (heksadesimalt)
 - Den første halvparten er produsentnummer: CC:cc:cc
 - Den andre halvparten er løpenummer: nn:nn:nn
- Den første oktetten i produsentnummeret, CC, har en spesiell oppbygging:
 - CCCCCCug (binært)
 - Når u-bitet er satt til 0 (null), så gjelder formatet som er oppgitt her, altså CC:cc:cc:nn:nn:nn (heksadesimalt)
 - Når u-bitet er satt til 1, så er alle C- og c-sifrene løpenummer, mens u- og g-bitene beholder sine spesielle betydninger
 - Når g-bitet er 0 så angir adressa en individuell node, og når g-bitet er 1 så er adressa en multicastgruppe



Grunnleggende om adresser

MAC-48-adresser

- Gitt denne MAC-48-adressa: 00:21:70:73:68:6E
- CC-oktetten har verdien 00 (heksadesimalt)
- På binær form er dette 00000000 (CCCCCug)
- Vi ser at både u- og g-bitene er satt til 0
- Dette er en MAC-48-adresse som:
 - følger det vanlige mønsteret med produsent- og løpenummer
 - angir en individuell node
 - er produsert av «Dell Inc» ifølge [OUI-lista](#) hos [IEEE](#) (søk i fila etter 00-21-70)



Grunnleggende om adresser

Modda IEEE EUI-64-format

- Unicast-adresser består av 2 ting:
 - 1 Prefiks
 - 2 Grensesnittidentifikator
- Bestemt av [RFC 4941](#)
- Grensesnittidentifikatorer er alltid på 64 bit
 - Dette gjelder ikke for adresser som starter på 000 (binært)
- Grensesnittidentifikatorer kan lages automatisk fra MAC-48-adresser
- Grensesnittidentifikatorer kan også angis manuelt eller velges tilfeldig
- Angis grensesnittidentifikatoren manuelt, så angis som regel en fullstendig IPv6-adresse
- Grensesnittidentifikatorer følger [IEEE EUI-64](#)-formatet med to unntak:
 - 1 Universal/local-bitet brukes med *invertert* betydning/verdi
 - Gruppebitet mister sin vanlige betydning i forbindelse med grensesnittidentifikatorer
 - 2 Oktettene på midten skal være FF:FE ved automatisk konvertering fra MAC-48 til EUI-64



Grunnleggende om adresser

Modda IEEE EUI-64-format

- Grensesnittidentifikatorer lages fra MAC-48-adresser etter oppskriften i [RFC 4291](#):
 - Gitt denne MAC-48-adressa: 00:21:70:73:68:6E
 - Invertér universal/local-bitet: 02:21:70:73:68:6E
 - Før: 00 (heksadesimalt) → 00000000 (binært)
 - Etter: 00000010 (binært) → 02 (heksadesimalt)
 - Sett inn FF:FE på midten: 02:21:70:FF:FE:73:68:6E
 - Ta bort overflødig kolon og nuller: 221:70FF:FE73:686E
 - Høyreskift hele stasen: ::221:70FF:FE73:686E
 - Nå er grensesnittidentifikatoren klar til å bli kombinert med ønsket prefiks
 - Prefiks annonsert av router: 2001:700:1100:3::/64
 - Fullstendig adresse: 2001:700:1100:3:221:70FF:FE73:686E



Grunnleggende om adresser

Modda IEEE EUI-64-format

- OBS! Arbeidsuhell!
- Det skulle egentlig ha vært FF:FF i stedet for FF:FE
 - MAC-48 → EUI-64 skal bruke FF:FF
 - EUI-48 → EUI-64 skal bruke FF:FE
- Fordi IPv6 bruker universal/local-bitet med invertert betydning/verdi, så er arbeidsuhellet akseptert
- Se [RFC 4291](#)
- [IEEE 802.15 WPAN](#), [IEEE 1394 FireWire](#), og [ZigBee](#) bruker EUI-64-adresser i lag 2



Grunnleggende om adresser

Manuell grensesnittidentifikator

- Manuell grensesnittidentifikator innebærer at universal/local-bitet som regel er satt til 0
- De øvrige 63 bitene kan være hva som helst, bare verdien ikke skaper adressekollisjon i samme VLAN
- Normalt bruker man manuelle grensesnittidentifikatorer satt til lave verdier
- For eksempel ::53 (DNS-tjener, kanskje)
- Samme eksempel, men med et vilkårlig prefiks: 2001:db8:1234:1::53



Grunnleggende om adresser

Manuell grensesnittidentifikator

- Lav verdi for grensesnittidentifikatorer gjør at universal/local-bitet blir satt til null:
 - ::53 (heksadesimalt)
 - ::0:0:0:53 (heksadesimalt)
 - ::0000000000000000:00 ... 00:0000000001010011 (binært)
 - Veldig praktisk for lokalgitte adresser, ikke sant?
- Uten invertering av universal/local-bitet, måtte vi bruke manuelle grensesnittidentifikatorer på denne måten:
 - ::0200:0:0:53 (heksadesimalt)
 - ::0000001000000000:00 ... 00:0000000001010011 (binært)
 - Tungvint og upraktisk, ikke sant?
- Se her:
 - 2001:db8:1234:1:0200:0:0:53 vs
 - 2001:db8:1234:1::53
 - Ja til den siste, nei til den forrige



Grunnleggende om adresser

Manuell grensesnittidentifikator

- Det er ingenting i veien for å «kode» IPv4-adressa inn i IPv6-adressa:
- 2001:700:1100:3:128:39:174:67 (excelsior.fig.ol.no)
- Man må bare passe på verdien til universal/local-bitet
- 128 = 0 1 2 8 = 0000 0001 0010 1000 (heks, heks, bin)
- u-bitet er 0, altså en lokalgitte adresse
- Dette gikk bra!



Grunnleggende om adresser

Manuell grensesnittidentifikator

- Verdiene
 - 0000 = 0,
 - 0001 = 1,
 - 0100 = 4,
 - 0101 = 5,
 - 1000 = 8,
 - 1001 = 9,
 - 1100 = C og
 - 1101 = D,
- gir alle 0 i u-bitet



Grunnleggende om adresser

Tilfeldig grensesnittidentifikator

- Konstant grensesnittidentifikator truer personvernet
- Eksempel med Tronds D531-læppis:
 - 2001:700:1100:3:221:70FF:FE73:686E (IT-avdelingen@FSI)
 - 2001:700:1D00:8:221:70FF:FE73:686E (public-nettet@HiG)
- RFC 4941 beskriver bruk av tilfeldig grensesnittidentifikator
- Med tilfeldig grensesnittidentifikator:
 - 2001:700:1100:3:B9D9:B729:6CDD:4E5 (IT-avdelingen@FSI)
 - 2001:700:1D00:8:B9D9:B729:6CDD:4E5 (public-nettet@HiG)
- Disse byttes ut typisk hver dag:
 - 2001:700:1100:3:F503:1E6F:5F2F:F5F2 (IT-avdelingen@FSI)
 - 2001:700:1D00:8:F503:1E6F:5F2F:F5F2 (public-nettet@HiG)
- Man må bare passe på u/l-bitet og passe seg for adressekollisjon



Grunnleggende om adresser

Tilfeldig grensesnittidentifikator

- RFC 4941 angir en metode for generering av tilfeldig grensesnittidentifikator:
 - 1 Sett sammen historisk verdi fra forrige runde (eller et tilfeldig 64-bit heltall) med den konstante grensesnittidentifikatoren til et 128-bit heltall
 - 2 Beregn MD5-hash av resultatet fra trinn 1
 - 3 Bruk de 64 *mest* signifikante bitene og sett det sjuende mest signifikante bitet til null (dette indikerer en lokalgitt grensesnittidentifikator)
 - 4 Sammenlign den nye tilfeldige grensesnittidentifikatoren med lista over reserverte identifikatorer; oppdages en uakseptabel identifikator, gå til trinn 1 og bruk de 64 *minst* signifikante bitene fra trinn 2 som historisk verdi
 - 5 Ta i bruk den nye tilfeldige grensesnittidentifikatoren
 - 6 Lagre de 64 *minst* signifikante bitene fra trinn 2 som historisk verdi for bruk den neste gangen denne algoritmen brukes



Grunnleggende om adresser

Spesialadresser

- Nulladressa:
 - 0:0:0:0:0:0:0:0/128 eller ::/128
 - Brukes av klienter som ennå ikke vet sin egen adresse (DHCPv6)
 - 0:0:0:0:0:0:0:0/0 eller ::/0
 - Brukes for å angi default route
 - Brukes av tjenester som godtar forespørsler fra alle grensesnitt (sjekk ut `bind(2)`-systemkallet i «Juniks»)
 - Tilsvarende 0.0.0.0/32 og 0/32, og 0.0.0.0/0 og 0/0 i IPv4



Grunnleggende om adresser

Spesialadresser

- Loopbackadressa: 0:0:0:0:0:0:0:1/128 eller ::1/128
 - Velkjent adresse for å snakke med tjenester i samme node
 - Tilsvarende 127.0.0.1/32 i IPv4



Grunnleggende om adresser

Spesialadresser

- Dokumentasjonsprefiks: 2001:db8::/32
 - Brukes for beskrivelse av IPv6-oppsett i lærebøker og annen generell dokumentasjon ([RFC 3849](#))
 - Forbudt å bruke på det offentlige internettet
 - Bør blokkeres i *inngående* og *utgående* ACL-er for internettgrensesnittet til routere



Grunnleggende om adresser

Spesialadresser

- IPv4-mapped IPv6 addresses: ::FFFF:w.x.y.z
 - Hvor w.x.y.z er den opprinnelige IPv4-adressa skrevet på vanlige måte for IPv4-adresser
 - Eksempel: ::FFFF:128.39.174.1
 - Brukes i systemer som har både IPv4- og IPv6-adresser, men hvor den enkelte tjeneste bare bruker IPv6-socketer og har slått av IPV6_V6ONLY med [setsockopt\(2\)](#) for lyttesocketen
 - Forbudt av sikkerhetshensyn i enkelte OS-er som [OpenBSD](#), se OpenBSDs [ip6\(4\)](#)
 - Tjenestene må da åpne separate lyttesoketer for IPv4 og IPv6
- [RFC 6890](#) inneholder en oversikt over alle spesialadresser for både IPv4 og IPv6



Grunnleggende om adresser

Duplicate Address Detection — DAD

- Når en unicast-adresse er generert skal man alltid sjekke at ingen andre bruker den samme adressa ([RFC 4862](#))
- Dette gjøres ved å sende en «ICMPv6 Neighbor Solicitation-melding» til den genererte adressas «Solicited-node multicast address»
- ICMPv6-meldinga inneholder den genererte adressa i feltet for «Target Address» ([RFC 4861](#))
- En «Solicited-node multicast address» er på formen FF02::1:FFaa:bbcc, hvor aabbcc er de 24 minst signifikante bitene fra den opprinnelige adressa ([RFC 4291](#))
- Sett at den genererte adressa er 2001:700:1100:3:221:70FF:FE73:686E
- «Solicited-node multicast address» vil da være FF02::1:FF73:686E
- Vanligvis kommer det ikke noe svar på slike ICMPv6-meldinger



- ... trodde vi ...
- «[Danger, Will Robinson!](#)»
- Det er et stort potensiale for Denial of Service — DoS ([RFC 3756](#))
- En «slabbedask» kan velge å svare på DAD og nekte oss å bruke *enhver* adresse
- Svaret kommer i form av en «ICMPv6 Neighbor Advertisement»-melding som forteller oss at en annen node bruker den samme adressa ([RFC 4862](#))
- Resultat: «slabbedasken» kan bruke nettverket uforstyrret
- Dersom det er 2 eller flere «slabbedasker» i samme nettverk, hva da?
- Problemet kan løses med «SEcure Neighbor Discovery» (SEND), [RFC 3971](#)

Oversikt over del 5: Adresstyper

- 20 Adresstyper
- 21 Link-local-adresser
- 22 Site-local-adresser
- 23 Offentlige unicast-adresser
- 24 Unike, lokale, aggregerbare adresser
- 25 Anycast-adresser
- 26 Multicast-adresser

Adresstyper

- Det finnes flere adresstyper med forskjellige bruksområder:
 - Unicast-adresser:
 - Link-local-adresser
 - Site-local-adresser
 - Offentlige unicast-adresser
 - Unike, lokale, aggregerbare adresser
 - Anycast-adresser
 - Multicast-adresser
- Merk at broadcast er avskaffa og er i stor grad erstatta med link-local-multicast

Adresstyper

Adresstyper

Link-local-adresser

- Definert: [RFC 4291](#)
- Bruksområde:
 - Lokal kommunikasjon internt i VLAN-et
 - Sentral for autokonfigurasjon
 - Blir ikke videresendt av routere til andre VLAN eller til internett
 - Kan brukes i ad-hoc-nett
- Prefiks: FE80::/10
- De neste 54 bitene skal settes til null
- De siste 64 bitene er grensesnittidentifikator i modda EUI-64-format
- Eksempel: FE80::221:70FF:FE73:686E



Adresstyper

Site-local-adresser

- Definert: [RFC 3513](#)
- Bruksområde: private adresser på lik linje med [RFC 1918](#)
- Prefiks: FEC0::/10
- De neste 54 bitene brukes til subnet-ID
- De siste 64 bitene er grensesnittidentifikator i modda EUI-64-format
- Eksempel: FEC0::DEAD:BEEF:1337
- Ikke bruk site-local-adresser ([RFC 3879](#))
- Site-local-adresser er erstatta med ULA ([RFC 4193](#))



Adresstyper

Offentlige unicast-adresser

- Definert: [RFC 4291](#) og [RFC 3587](#)
- Bruksområde: ende-til-ende-kommunikasjon på det offentlige internett
- Prefiks: 2000::/3
- De neste bitene allokeres hierarkisk, minimum i 4-bitblokker, men gjerne i 8- eller 16-bitblokker
- De siste 64 bitene er grensesnittidentifikator i modda EUI-64-format
- Det er vanlig at kundene blir tildelt /48- eller /56-bits prefiks av ISP-ene:
 - /48-bits prefiks gir $128 - 64 - 48 = 16$ subnetbit $\rightarrow 2^{16} = 65536$ subnett
 - /56-bits prefiks gir $128 - 64 - 56 = 8$ subnetbit $\rightarrow 2^8 = 256$ subnett
- Eksempel: 2001:700:1100:1::1/128



Adresstyper

Unike, lokale, aggregerbare adresser

- Definert: [RFC 4193](#)
- Bruksområde: ende-til-ende-kommunikasjon internt i nettverket
- Veldig praktisk å ha faste, interne adresser uavhengig av offentlig prefiks tildelt av ISP
- Prefiks: FC00::/7
- Det åttende mest signifikante bitet skal settes til 1 inntil videre
- Det reelle prefikset er dermed FD00::/8
- Prefikset FC00::/8 er reservert inntil videre



Adresstyper

Unike, lokale, aggregerbare adresser

- Reelt prefiks: FD00::/8
- De neste 40 bitene genereres tilfeldig, gjerne som beskrevet i [RFC 4193](#)
- De neste 16 bitene brukes til subnett-ID
- De siste 64 bitene er grensesnittidentifikator i modda EUI-64-format
- Eksempel: FD5C:14CF:C300:31::1/128



Adresstyper

Unike, lokale, aggregerbare adresser

- [SixXS](#) tilbyr bl.a.:
 - Generering av ULA-prefiks: <http://www.sixxs.net/tools/grh/ula/>
 - Registrering av ULA-prefiks: <http://www.sixxs.net/tools/grh/ula/list/>
- [George Michaelson](#), seniorforsker ved [APNIC](#), har oppdaget ULA-adresser i fri dressur ute på internett:
 - Tydeligvis klarer ikke folk å lese RFC-ene og holde seg til de fastsatte reglene
 - http://www.sixxs.net/archive/docs/IEPG2013_ULA_in_the_wild.pdf



Adresstyper

Unike, lokale, aggregerbare adresser

- Her er algoritmen fra [RFC 4193](#) for å generere de 40 tilfeldige bitene:
 - 1 Uttrykk nåværende øyeblikk som et 64-bit heltall i NTP-format ([RFC 5905](#))
 - 2 Bruk en EUI-64-identifikator fra systemet som kjører denne algoritmen
 - Mangler du en EUI-64-identifikator, så kan du lage en fra en 48-bit MAC-adresse som angitt i [RFC 4291](#)
 - Kan du ikke lage en EUI-64-identifikator, så bruk en annen unik verdi som serienummeret til systemet
 - 3 Sett sammen de to 64-bit heltallene til et 128-bit heltall
 - 4 Beregn en SHA-1-hash som beskrevet i [RFC 3174](#). Resultatet er et heltall på 160 bit
 - 5 Bruk de 40 minst signifikante bitene som global identifikator
- Har man tilgang på tilfeldige tall av god kvalitet, så kan man bruke de i stedet for metoden over



Adresstyper

Anycast-adresser

- Definert: [RFC 4291](#)
- Bruksområde: felles adresse for distribuerte tjenester, routerne bestemmer hvilken server som er nærmest og sender trafikken dit
- Prefiks: ingen, allokeres fra dine egne unicast-adresser og markeres som en anycast-adresse hos routerne og serverne
- Alle IPv6-adresser hvor alle bit i grensesnittidentifikatoren satt til null, er reservert som «Subnet-Router anycast address»
- Denne anycast-adressa brukes når man vil kontakte én av potensielt flere routere i subnettet der du er
- Eksempel: 2001:700:1100:1::/128 **anycast**
- Se også [RFC 2526](#)



Adresstyper

Multicast-adresser

- Definert: [RFC 4291](#)
- Bruksområde: én-til-mange-kommunikasjon
- Prefiks: FF::/8
- Flagg f og rekkevidde r er innebygget i adressa: FF f r ::/16
- Eksempel: FF0E::101/128 (global multicast-adresse for NTP)



Adresstyper

Multicast-adresser

- Flaggene heter ORPT (null, err, pe, te)
- Flagget T angir med 0 at adressa er velkjent (definert av [IANA](#)), og med 1 at adressa er midlertidig (lokalt definert)
- Flagget P angir med 1 at adressa inneholder et unicast-prefiks og skal følge reglene i [RFC 3306](#)
- Flagget R angir med 1 at adressa også inneholder et møtepunkt («rendezvous point») og skal følge reglene i [RFC 3956](#)
- Flaggene P og R gjør det enkelt å lage egne multicast-adresser for internt bruk i organisasjonen
- Bruk av flaggene R, P og T gjennomgås i detalj i del 10



Adresstyper

Multicast-adresser

- Følgende rekkevidder er definert i [RFC 4921](#):
 - 0: reservert
 - 1: interface-local
 - 2: link-local
 - 3: reservert
 - 4: admin-local
 - 5: site-local
 - 6: ikke definert
 - 7: ikke definert
 - 8: organization-local
 - 9: ikke definert
 - A: ikke definert, brukt av Uninett til å [begrense](#) trafikken innenfor «Uninettet»
 - B: ikke definert
 - C: ikke definert
 - D: ikke definert
 - E: global
 - F: reservert



Adresstyper

Multicast-adresser

- Noen kjente IPv6-multicastadresser:
 - FF02::1 All nodes on the local network segment
 - FF02::2 All routers on the local network segment
 - FF02::5 OSPFv3 All SPF routers
 - FF02::6 OSPFv3 All DR routers
 - FF02::8 IS-IS for IPv6 routers
 - FF02::9 RIP routers
 - FF02::A EIGRP routers
 - FF02::D PIM routers
 - FF02::16 MLDv2 reports
 - FF02::1:2 All DHCP servers and relay agents on the local network segment
 - FF02::1:3 All LLNMR hosts on the local network segment
 - FF05::1:3 All DHCP servers on the local network site
 - FF0x::C Simple Service Discovery Protocol
 - FF0x::FB Multicast DNS
 - FF0x::101 Network Time Protocol
 - FF0x::108 Network Information Service
 - FF0x::114 Used for experiments



- Kobling av multicast-adresser til lag-2-adresser:
 - Eksempel:
 - IPv6: FF02::1 = FF02::0000:0001
 - MAC-48: 33:33:00:00:00:01
 - De 32 minst signifikante bitene kopieres fra IPv6-adressa og til MAC-48-adressa
 - Dette gir en viss overlapp for de multicast-adresser som tilfeldigvis slutter på de samme 32 bitene
 - Det går ganske bra i praksis
 - Se [RFC 2464](#) og [RFC 6085](#)

Oversikt over del 6: DNS I

27 AAAA og PTR

28 A6

DNS

DNS

AAAA og PTR

- Navn-til-IPv6-adresser bruker AAAA-poster
 - Eksempel:
\$ORIGIN fig.ol.no.
svabu IN AAAA 2001:700:1100:1::4
- IPv6-adresser-til-navn bruker PTR-poster plassert i ip6.arpa.
 - Eksempel:
\$ORIGIN 1.0.0.0.0.0.1.1.0.0.7.0.1.0.0.2.ip6.arpa.
4.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 IN PTR svabu.fig.ol.no.
- Se [RFC 3596](#)

- A6-poster var foreslått som erstatning for AAAA-poster av [RFC 2874](#), men er endret til eksperimentell av [RFC 3363](#)
- [RFC 3364](#) diskuterer fordeler og ulemper med AAAA og A6
- En A6-post består av 2–3 ting:
 - 1 Prefikslengde fra og med 0 til og med 128
 - 2 Utdrag av IPv6-adressa
 - 3 Navn som henviser til resten av adressa
- Settes prefikslengda til:
 - 0, så er det **ikke** lov å oppgi noen henvisning, fordi dette navnet er det øverste eller det eneste nivået i en kjede
 - 128, så er det **ikke** lov å oppgi noen IPv6-adresse, fordi man henviser til et helt annet navn, tydeligvis et overflødig alternativ til CNAME
- Avsnittene 3.1.1 og 3.1.3 i [RFC 2874](#) er ikke enige med seg selv når prefikslengda settes til 128



- Et tenkt eksempel med A6:
- \$ORIGIN ip6.uninett.no.
 uninett IN A6 0 2001:700::
 fig IN A6 32 0:0:1100:: uninett

 \$ORIGIN fig.ol.no.
 ext-servere.ip6 IN A6 48 0:0:0:1:: fig.ip6.uninett.no.
 svabu IN A6 64 ::4 ext-servere.ip6
- Vi vil vite IPv6-adressa for svabu.fig.ol.no. og vi vil bruke A6-poster for å finne svaret



- Et tenkt eksempel med A6:
- \$ORIGIN fig.ol.no.
 svabu IN A6 64 ::4 ext-servere.ip6
- Forklaring:
 - svabu.fig.ol.no. mangler de 64 mest signifikante bitene og henviser til ext-servere.ip6.fig.ol.no.



- Et tenkt eksempel med A6:
- \$ORIGIN fig.ol.no.
 svabu IN A6 64 ::4 ext-servere.ip6
 ext-servere.ip6 IN A6 48 0:0:0:1:: fig.ip6.uninett.no.
- Forklaring:
 - ext-servere.ip6.fig.ol.no. mangler de 48 mest signifikante bitene og henviser til fig.ip6.uninett.no.



- Et tenkt eksempel med A6:
- \$ORIGIN fig.ol.no.

```
svabu          IN A6 64          ::4 ext-servere.ip6
ext-servere.ip6 IN A6 48 0:0:0:1:: fig.ip6.uninett.no.
```



```
$ORIGIN ip6.uninett.no.
fig            IN A6 32 0:0:1100:: uninett
```
- Forklaring:
 - fig.ip6.uninett.no. mangler de 32 mest signifikante bitene og henviser til uninett.ip6.uninett.no.

- Et tenkt eksempel med A6:
- \$ORIGIN fig.ol.no.

```
svabu          IN A6 64          ::4 ext-servere.ip6
ext-servere.ip6 IN A6 48 0:0:0:1:: fig.ip6.uninett.no.
```



```
$ORIGIN ip6.uninett.no.
fig            IN A6 32 0:0:1100:: uninett
uninett        IN A6  0 2001:700::
```
- Vi får bygd opp følgende adressekjede:
 - ::4 svabu.fig.ol.no.
 - 0:0:0:1:: ext-servere.ip6.fig.ol.no.
 - 0:0:1100:: fig.ip6.uninett.no.
 - 2001:700:: uninett.ip6.uninett.no.
- Bitvis-OR gir den fullstendige adressa 2001:700:1100:1::4

- Et tenkt eksempel med A6:
- \$ORIGIN fig.ol.no.

```
svabu          IN A6 64          ::4 ext-servere.ip6
ext-servere.ip6 IN A6 48 0:0:0:1:: fig.ip6.uninett.no.
```



```
$ORIGIN ip6.uninett.no.
fig            IN A6 32 0:0:1100:: uninett
uninett        IN A6  0 2001:700::
```
- Forklaring:
 - Kjeden slutter med uninett.ip6.uninett.no. og her angis de 32 mest signifikante bitene

Del VII

ICMPv6

Oversikt over del 7: ICMPv6 I

- 29 ICMPv6
- 30 Multicast Listener Discovery
- 31 Neighbor Discovery
- 32 Router Renumbering
- 33 Node Information
- 34 Inverse Neighbor Discovery
- 35 Version 2 Multicast Listener Report
- 36 Mobile IPv6
- 37 SEcure Neighbor Discovery (SEND)
- 38 Experimental Mobility Type
- 39 Multicast Router Discovery
- 40 FMIPv6
- 41 RPL Control Message
- 42 ILNPv6 Locator Update Message
- 43 Duplicate Address



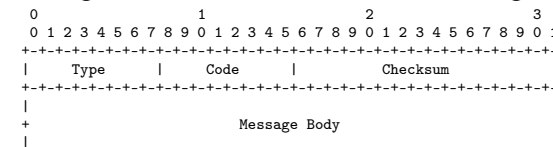
ICMPv6

- Fra [RFC 4443](#)
- Feilmeldinger:
 - 1: Destination Unreachable
 - 2: Packet Too Big
 - 3: Time Exceeded
 - 4: Parameter Problem
 - 100: Private eksperimenter
 - 101: Private eksperimenter
 - 127: Reservert for utvidelse av feilmeldingene
- Informative meldinger:
 - 128: Echo request (ping)
 - 129: Echo reply (pong)
 - 200: Private eksperimenter
 - 201: Private eksperimenter
 - 255: Reservert for utvidelse av informative meldinger



ICMPv6

- Feilrapportering- og feilsøkingstjeneste for IPv6
- Definert: [RFC 4443](#) og [RFC 4844](#)
- ICMPv6-meldinger inneholder to tall som forteller noe om budskapets mening og innhold:
 - Type: hovednummer
 - Code: undernummer, settes til 0 når det ikke er definert noen undernummer
- I tillegg er det felter for sjekksum og andre opplysninger som er unike for hver type (og underkode) av meldingene
- Den generelle formen for ICMPv6-meldinger vises under



ICMPv6

Multicast Listener Discovery

- Definert: [RFC 2710](#)
- Angir tre nye ICMPv6-meldinger:
 - 130: Multicast Listener Query
 - 131: Multicast Listener Report
 - 132: Multicast Listener Done
- Brukes for å fortelle routere hvilke multicastadresser man vil motta trafikk for



ICMPv6

Neighbor Discovery

- Definert: [RFC 4861](#)
- Angir fem nye ICMPv6-meldinger:
 - 133: Router Solicitation
 - 134: Router Advertisement
 - 135: Neighbor Solicitation
 - 136: Neighbor Advertisement
 - 137: Redirect
- Sentral ved autokonfigurering av adresser
- Brukes for å bekrefte at nodene er oppegående og bestemme lag-2-adressene til mottakere
- Neighbor Discovery gjennomgås i detalj i del 8



ICMPv6

Node Information

- Definert: [RFC 4620](#)
- Angir to nye ICMPv6-meldinger:
 - 139: Node Information Query
 - 140: Node Information Reply
- [RFC 4620](#) angir følgende underkoder for type 139:
 - 0: Datafeltet inneholder en IPv6-adresse
 - 1: Datafeltet inneholder et navn
 - 2: Datafeltet inneholder en IPv4-adresse
- [RFC 4620](#) angir følgende underkoder for type 140:
 - 0: Vellykket svar
 - 1: Svaret vil ikke bli avslørt
 - 2: Underkoden i forespørselen er ukjent



ICMPv6

Router Renumbering

- Definert: [RFC 2894](#)
- Angir én ny ICMPv6-melding:
 - 138: Router Renumbering
- [RFC 2894](#) angir følgende underkoder:
 - 0: Router Renumbering Command
 - 1: Router Renumbering Result
 - 255: Sequence Number Reset



ICMPv6

Inverse Neighbor Discovery

- Definert: [RFC 3122](#)
- Angir to nye ICMPv6-meldinger:
 - 141: Inverse Neighbor Discovery Solicitation
 - 142: Inverse Neighbor Discovery Advertisement
- Gjør det mulig for én node å lære IPv6-adressen(e) til en annen node i samme VLAN, når man bare vet lag-2-adressa til den andre noden



ICMPv6

Version 2 Multicast Listener Report

- Definert: [RFC 3810](#)
- Angir én ny ICMPv6-melding:
 - 143: Version 2 Multicast Listener Report
- Utvider MLDv1 ([RFC 2710](#)) med slik at bare bestemte avsendere er interessante (Source-Specific Multicast, [RFC 3569](#))



ICMPv6

SEcure Neighbor Discovery (SEND)

- Definert: [RFC 3971](#)
- Angir to nye ICMPv6-meldinger:
 - 148: Certification Path Solicitation
 - 149: Certification Path Advertisement
- Med SEND unngås DoS-problemene til Neighbor Discovery
- Routerne deler ut kryptografisk genererte adresser [RFC 3972](#)
- Dette krever sertifikatstruktur (RPKI, [RFC 6494](#)) i routere og i klienter
- Ikke implementert i Cisco IOS 12.2(55)SE for Catalyst 3560G
- Ikke spesielt aktuelt for FSI, for annet enn ansattnett, på grunn av den administrative byrden



ICMPv6

Mobile IPv6

- Definert: [RFC 6275](#)
- Angir fire nye ICMPv6-meldinger:
 - 144: Home Agent Address Discovery Request
 - 145: Home Agent Address Discovery Reply
 - 146: Mobile Prefix Solicitation
 - 147: Mobile Prefix Advertisement
- Brukes for å tilrettelegge for digitale nomader



ICMPv6

Experimental Mobility Type

- Definert: [RFC 4065](#)
- Angir én ny ICMPv6-melding:
 - 150: Experimental Mobility Type
- «The Seamoby Candidate Access Router Discovery (CARD) protocol [[RFC 4066](#)] and the Context Transfer Protocol (CXTTP) [[RFC 4067](#)] are experimental protocols designed to accelerate IP handover between wireless access routers»



ICMPv6

Multicast Router Discovery

- Definert: [RFC 4286](#)
- Angir tre nye ICMPv6-meldinger:
 - 151: Multicast Router Advertisement
 - 152: Multicast Router Solicitation
 - 153: Multicast Router Termination
- Catalyst 3560G har ikke støtte for annet enn IPv4-multicast
- Ved FSI har vi ikke fått testet IPv6-multicast

ICMPv6

FMIPv6

- Definert: [RFC 5568](#)
- Angir én ny ICMPv6-melding:
 - 154: FMIPv6, Fast handovers, Mobile IPv6

ICMPv6

RPL Control Message

- Definert: [RFC 6550](#)
- Angir én ny ICMPv6-melding:
 - 155: RPL Control Message
- IPv6 Routing Protocol for Low-Power and Lossy Networks

ICMPv6

ILNPv6 Locator Update Message

- Definert: [RFC 6743](#)
- Angir én ny ICMPv6-melding:
 - 156: ILNPv6 Locator Update Message
- Identifier-Locator Network Protocol
- En eksperimentell måte å håndtere digitale nomader

- Definert: [RFC 6775](#)
- Angir to nye ICMPv6-meldinger:
 - 157: Duplicate Address Request
 - 158: Duplicate Address Confirmation
- Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)

Oversikt over del 8: Neighbor Discovery I

- 44 Router Solicitation
- 45 Router Advertisement
- 46 Neighbor Solicitation
- 47 Neighbor Advertisement
- 48 Redirect

Neighbor Discovery

- Definert: [RFC 4861](#)
- Angir fem nye ICMPv6-meldinger:
 - 133: Router Solicitation
 - 134: Router Advertisement
 - 135: Neighbor Solicitation
 - 136: Neighbor Advertisement
 - 137: Redirect
- Sentral ved autokonfigurering av adresser
- Brukes for å bekrefte at nodene er oppegående og bestemme lag-2-adressene til mottakere

Neighbor Discovery

Neighbor Discovery

Router Solicitation

```
Internet Control Message Protocol v6
Type: Router Solicitation (133)
Code: 0
Checksum: 0xc065 [correct]
Reserved: 00000000
ICMPv6 Option (Source link-layer address : 00:21:70:73:68:6e)
Type: Source link-layer address (1)
Length: 1 (8 bytes)
Link-layer address: Dell_73:68:6e (00:21:70:73:68:6e)
```

- Avsenders IPv6-adresse er enten ::/0 eller en av utgående grensesnitts IPv6-adresser
- Mottakers IPv6-adresse er vanligvis FF02::2
- «Hop Limit» i IPv6-headeren skal settes til 255
- Det er god sedvane å angi sin egen lag-2-adresse i ICMPv6-meldinga



Neighbor Discovery

Router Advertisement

```
Internet Control Message Protocol v6
Type: Router Advertisement (134)
Code: 0
Checksum: 0xfa8c [correct]
Cur hop limit: 64
Flags: 0x48
0... .... = Managed address configuration: Not set
1... .... = Other configuration: Set
..0. .... = Home Agent: Not set
...0 1... = Prf (Default Router Preference): High (1)
.... 0... = Proxy: Not set
.... ..0. = Reserved: 0
Router lifetime (s): 1800
Reachable time (ms): 0
Retrans timer (ms): 0
ICMPv6 Option (Source link-layer address : 00:17:e0:77:14:57)
Type: Source link-layer address (1)
Length: 1 (8 bytes)
Link-layer address: Cisco_77:14:57 (00:17:e0:77:14:57)
ICMPv6 Option (MTU : 1500)
Type: MTU (5)
Length: 1 (8 bytes)
Reserved
MTU: 1500
```

- Avsenders IPv6-adresse må være routerens link-local-adresse for utgående grensesnitt
- Mottakers IPv6-adresse er enten adressa til den noden som sendte «Router Solicitation» eller til FF02::1 for generell annonsering
- «Hop Limit» i IPv6-headeren skal settes til 255



Neighbor Discovery

Router Advertisement

```
Internet Control Message Protocol v6
Type: Router Advertisement (134)
Code: 0
Checksum: 0xfa8c [correct]
Cur hop limit: 64
Flags: 0x48
0... .... = Managed address configuration: Not set
1... .... = Other configuration: Set
..0. .... = Home Agent: Not set
...0 1... = Prf (Default Router Preference): High (1)
.... 0... = Proxy: Not set
.... ..0. = Reserved: 0
Router lifetime (s): 1800
Reachable time (ms): 0
Retrans timer (ms): 0
ICMPv6 Option (Source link-layer address : 00:17:e0:77:14:57)
Type: Source link-layer address (1)
Length: 1 (8 bytes)
Link-layer address: Cisco_77:14:57 (00:17:e0:77:14:57)
ICMPv6 Option (MTU : 1500)
Type: MTU (5)
Length: 1 (8 bytes)
Reserved
MTU: 1500
```

- Routeren er snill og oppgir:
 - Autokonfigurasjon av adresser skal utføres
 - Andre opplysninger er tilgjengelig med DHCPv6
 - Dette er ingen «Home Agent»
 - Routerens preferansenivå er «High»
 - Annonseringens levetid er 1800 s = 30 min
 - Routerens lag-2-adresse
 - Linkens MTU-verdi



Neighbor Discovery

Router Advertisement

```
ICMPv6 Option (Prefix information : 2001:700:1100:3::/64)
Type: Prefix information (3)
Length: 4 (32 bytes)
Prefix Length: 64
Flag: 0xc0
1... .... = On-link flag(L): Set
1... .... = Autonomous address-configuration flag(A): Set
..0. .... = Router address flag(R): Not set
...0 0000 = Reserved: 0
Valid Lifetime: 2592000
Preferred Lifetime: 604800
Reserved
Prefix: it.ip6.fig.ol.no (2001:700:1100:3::)
```

- Routeren opppgir følgende om 2001:700:1100:3::/64
 - Prefikset er direkte tilgjengelig
 - Autokonfigurasjon er tillatt
 - Genererte adresser er gyldige i 30 dager, med foretrukket levetid på 7 dager



Neighbor Discovery

Neighbor Solicitation

```
Internet Control Message Protocol v6
Type: Neighbor Solicitation (135)
Code: 0
Checksum: 0x4571 [correct]
Reserved: 00000000
Target Address: monitor2.fig.ol.no (2001:700:1100:3:20b:dbff:fe52:67e2)
ICMPv6 Option (Source link-layer address : 00:26:18:f2:72:40)
Type: Source link-layer address (1)
Length: 1 (8 bytes)
Link-layer address: AsustekC_f2:72:40 (00:26:18:f2:72:40)
```

- I dette tilfellet ville
 - 1 2001:700:1100:3:226:18FF:FEF2:7240 sjekke om
 - 2 2001:700:1100:3:20B:DBFF:FE52:67E2 fortsatt var i live
- Forespørselen ble sendt til «Solicited-node multicast-adressa» FF02::1:FF52:67E2



Neighbor Discovery

Neighbor Advertisement

```
Internet Control Message Protocol v6
Type: Neighbor Advertisement (136)
Code: 0
Checksum: 0x157e [correct]
Flags: 0x60000000
0... .. = Router: Not set
..1.. .. = Solicited: Set
..1.. .. = Override: Set
...0 0000 0000 0000 0000 0000 0000 0000 = Reserved: 0
Target Address: monitor2.fig.ol.no (2001:700:1100:3:20b:dbff:fe52:67e2)
ICMPv6 Option (Target link-layer address : 00:0b:db:52:67:e2)
Type: Target link-layer address (2)
Length: 1 (8 bytes)
Link-layer address: DellEsgP_52:67:e2 (00:0b:db:52:67:e2)
```

- 2001:700:1100:3:20B:DBFF:FE52:67E2 sendte svar tilbake til 2001:700:1100:3:226:18FF:FEF2:7240 med klar beskjed at
 - Den er ikke en router
 - Dette er et svar på en forespørsel og ikke en tilfeldig annonsering
 - Gamle opplysninger om 2001:700:1100:3:20B:DBFF:FE52:67E2 skal slettes
 - Lag-2-adressa er stadig 00:0B:DB:52:67:E2



Neighbor Discovery

Redirect

```
0 1 2 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+
| Type | Code | Checksum |
+-----+-----+-----+-----+
| Reserved |
+-----+-----+-----+-----+
| Target Address |
+-----+-----+-----+-----+
| Destination Address |
+-----+-----+-----+-----+
| Options ... |
+-----+-----+-----+-----+
```

- Jeg har hittil ikke sett en eneste ICMPv6 redirect-melding



Del IX

DHCPv6



49 DHCPv6

50 Meldinger

51 DHCP Unique Identifier

52 Identity association

53 Identity association identifier

DHCPv6

Meldinger

- Solicit
 - Fra klient til server/relay
 - Brukes for å oppdage servere
- Advertise
 - Fra server/relay til klient
 - Brukes for å varsle klienten om tjenestetilbudet
- Request
 - Fra klient til spesifikk server
 - Bruker for å etterspørre om adresser og andre opplysninger fra en bestemt server
- Confirm
 - Fra server/relay til klient
 - Brukes for å bestemme om tidligere oppgitt adresse fortsatt er gyldig

DHCPv6

- DHCPv6 er definert i [RFC 3315](#) med oppdateringer fra [RFC 3319](#), [RFC 3633](#), [RFC 3646](#), [RFC 3736](#), [RFC 4361](#), [RFC 5007](#), [RFC 5494](#), [RFC 6221](#), [RFC 6422](#), [RFC 6603](#), [RFC 6644](#) og [RFC 7083](#)
- Kommunikasjonen foregår først med multicast og UDP, og kan senere bytte til unicast og UDP
- Klientene bruker port 546 og serverne/relay-agentene bruker port 547
- Klientene bruker sin egen link-local-adresse som avsender og multicast-adressa FF02::1:2 som mottaker
- Relay-agentene videresender til multicast-adressa FF05::1:3, med mindre de kjenner og vil bruke unicast-adressa til serveren
- Serverne svarer med sin link-local-adresser som avsender og klientens link-local-adresse som mottaker

DHCPv6

Meldinger

- Renew
 - Fra klient til server/relay
 - Brukes for å fornye leieavtalen og oppdatere andre opplysninger
- Rebind
 - Fra klient til server/relay
 - Brukes til annonsering i etterkant av en renew-melding, dersom det ikke kom noe svar på fornyelsen

- Reply
 - Fra server til klient
 - Serveren sender tildelt adresse og andre opplysninger i en reply-melding som svar på solicit-, request-, renew- og rebind-meldinger
 - Serveren sender konfigurasjonsparametre i en reply-melding som svar på en information-request-melding
 - Serveren sender en reply-melding som svar på en confirm-melding for å bekrefte eller avkrefte at adressa tilordnet klienten er gyldig eller ikke
 - Serveren sender en reply-melding for å kvittere for mottatt release- eller decline-meldinger
- Release
 - Fra klient til server/relay
 - Brukes for å frigjøre en utleid adresse

- Relay-forward
 - Fra relay til relay/server
 - Brukes av relay for å videresende forespørsler fra klienter eller andre relay til en ny relay eller server
- Relay-reply
 - Fra server/relay til relay
 - Brukes av server for å videresende svar tilbake til klienter gjennom relay(kjeden)

- Decline
 - Fra klient til server/relay
 - Brukes for å fortelle at en eller flere utdelte adresser allerede er tatt i bruk i nabolaget til klienten
- Reconfigure
 - Fra server til klient
 - Brukes for å gjøre klienten oppmerksom på nye opplysninger og at klienten må gjennomføre renew/reply- eller information-request/reply-transaksjoner for å få de nye opplysningene
- Information-request
 - Fra klient til server/relay
 - Brukes for å be om konfigurasjonsparametre uten å bli tildelt en adresse

- Klientene identifiseres med DHCP Unique Identifier, DUID, som har variabel lengde og format
- Klientene kan ha flere nettverksgrensesnitt
- Hvert grensesnitt har i tillegg sin Identity Association Identifier, IAID, lengde 32 bit
- Klientene oppgir aktuell DUID og IAID i forespørslene
- DHCPv6-serverne har sine egne DUID og IAID, og oppgir disse i svarene

DHCPv6

DHCP Unique Identifier, DUID

- DUID finnes i tre varianter:
 - Type 1: Linklagsadresse med tidspunkt for generering, DUID-LLT
 - Type 2: Unik identifikator basert på Enterprise-nummer utdelt av IANA, DUID-EN
 - Type 3: Linklagsadresse, DUID-LL



DHCPv6

DHCP Unique Identifier, DUID

- Type 1 er vanlig i Windows, og lagres i Dhcpv6DUID i
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\
TCPIP6\Parameters
- Type 3 er enklere og mer forutsigbart, og det beste valget for statisk
tildeling av IPv6-adresse med tanke på reinstallasjon av OS
- Jeg har ikke funnet noen annen måte å tvinge en bestemt DUID-type i
Windows, enn å sette Dhcpv6DUID manuelt eller gjennom skript, og
naturlig nok restarte Windows etterpå
- [Dibbler](#) og Unix-systemer er tradisjonelt langt snillere, og lar oss angi i
konfigurasjonen de gangene vi ønsker DUID-LL istedet for DUID-LLT



DHCPv6

DHCP Unique Identifier, DUID

- Type 1 kan se slik ut:
00 01 00 01 13 10 43 9B 00 26 18 F2 72 40
 - 00 01 angir at dette er DUID type 1.
 - 00 01 angir at det kommer en MAC-48-adresse til slutt
 - 13 10 43 9B angir klokkeslettet målt i sekunder siden 1. januar 2000 UTC
 - I dette tilfellet: 0x1310439B s, 319832987 s, 10.1351038909 år etter
1. januar 2000 UTC, altså 18. februar 2010, kl. 18:29:47 UTC
 - 00 26 18 F2 72 40 er MAC-48-adressa for systemet som dette
eksempelet er hentet fra
- Type 3 kan se slik ut:
00 03 00 01 00 26 18 F2 72 40
 - 00 03 angir at dette er DUID type 3.
 - 00 01 angir at det kommer en MAC-48-adresse til slutt
 - 00 26 18 F2 72 40 er MAC-48-adressa for systemet som dette
eksempelet er hentet fra



DHCPv6

Identity association, IA

- [RFC 3315](#)
- Bla, bla, bla



DHCPv6

Identity association identifier, IAID

- RFC 3315
- Bla, bla, bla

Oversikt over del 10: Avansert multicast I

- 54 Multicastflaggene
- 55 Når T er satt til 1
- 56 Når PT er satt til 11
- 57 Når RPT er satt til 111

Del X

Avansert multicast

Avansert multicast

Multicastflaggene

- Flaggene heter ORPT (null, err, pe, te)
- Flagget T angir med 0 at adressa er velkjent (definert av IANA), og med 1 at adressa er midlertidig (lokalt definert)
- Flagget P angir med 1 at adressa inneholder et unicast-prefiks og skal følge reglene i RFC 3306
- Flagget R angir med 1 at adressa også inneholder et møtepunkt («rendezvous point») og skal følge reglene i RFC 3956
- Flaggene P og R gjør det enkelt å lage egne multicast-adresser for internt bruk i organisasjonen

Avansert multicast

Når T er satt til 1

```
| 8 | 4 | 4 | 112 bits |
+---+
|11111111|0001|scop| group ID |
+---+
```

- Adresseformatet er gitt av [RFC 4291](#)
- De 12 mest signifikante bitene må beholdes som vist
- Rekkevidden settes til ønsket, lovlig verdi
- De 112 øvrige bitene kan settes fritt
- Eksempel:
 - FF12::DEAD:BEEF:CAFE:0:FACE:B00C:1
 - En link-local, midlertidig multicast-adresse



Avansert multicast

Når PT er satt til 11

```
| 8 | 4 | 4 | 8 | 8 | 64 | 32 |
+---+
|11111111|0011|scop|reserved| plen | network prefix | group ID |
+---+
```

- Eksempel:
 - FF3E:0030:2001:700:1100:0:8000:1337
 - Den første adressa er begrenset til internett (global, 48-bit)
 - FF38:0030:2001:700:1100:0:8000:1337
 - Den andre adressa er begrenset til FSI (organizational-local, 48-bit)
 - FF32:0040:2001:700:1100:3:8000:1337
 - Den tredje adressa er begrenset til IT-avdelingen ved FSI (link-local, 64-bit)



Avansert multicast

Når PT er satt til 11

```
| 8 | 4 | 4 | 8 | 8 | 64 | 32 |
+---+
|11111111|0011|scop|reserved| plen | network prefix | group ID |
+---+
```

- Adresseformatet er gitt av [RFC 3306](#)
- De 12 mest signifikante bitene må beholdes som vist
- Rekkevidden settes til ønsket, lovlig verdi, og rekkevidden skal ikke overskride *utbredelsen* av det angitte nettverksprefikset
- Feltet «plen» settes til prefikslengden til nettverksprefikset
- Nettverksprefikset er ditt eget unicast-prefiks
- «Group ID» settes i tråd med retningslinjene til [RFC 3307](#)



Avansert multicast

Når RPT er satt til 111

```
| 8 | 4 | 4 | 4 | 4 | 8 | 64 | 32 |
+---+
|11111111|0111|scop|rsvd|RIID|plen| network prefix | group ID |
+---+
```

- Adresseformatet er gitt av [RFC 3956](#)
- De 12 mest signifikante bitene må beholdes som vist
- Rekkevidden settes til ønsket, lovlig verdi, og rekkevidden skal ikke overskride *utbredelsen* av det angitte nettverksprefikset
- Feltet «RIID» settes til møtepunktets grensesnittidentifikator
 - RIID kan ikke være 0, for dette skaper konflikt med «Subnet-Router Anycast Address» fra [RFC 3513](#)
- Feltet «plen» settes til prefikslengden til nettverksprefikset
- Nettverksprefikset er ditt eget unicast-prefiks
- «Group ID» settes i tråd med retningslinjene til [RFC 3307](#)



Avansert multicast

Når RPT er satt til 111

```
| 8 | 4 | 4 | 4 | 4 | 8 | 64 | 32 |
+---+---+---+---+---+---+---+---+
|11111111|0111|scop|rsvd|RIID|plen| network prefix | group ID |
+---+---+---+---+---+---+---+---+
```

• Eksempler:

- FF78:0130:2001:700:1100:0:8000:1337
 - Denne adressa er begrenset til organization-local
 - Nettværksprefikset er 2001:700:1100::/48
 - Møtepunktets adresse er 2001:700:1100::1
 - Møtepunktets adresse må konfigureres på et loopbackgrensesnitt i Fagskolens ytterste IPv6-multicast-router
- interface Loopback0
 - ipv6 address 2001:700:1100::1



Oversikt over del 11: Konfigurasjon av IPv6 I

58 Cisco IOS

- IPv6-routing
- ACL-er
- DHCPv6

59 OS-konfig



Del XI

Konfigurasjon av IPv6



Konfigurasjon av IPv6

Cisco IOS: IPv6-routing

- 1 configure terminal
- 2 sdm prefer dual-ipv4-and-ipv6 default (Rekonfigurere TCAM)
- 3 end
- 4 reload
- 5 configure terminal
- 6 ip routing (Nødvendig for IP-routing i det hele tatt)
- 7 ipv6 unicast-routing
- 8 no ipv6 source-route
- 9 end



Konfigurasjon av IPv6

Cisco IOS: IPv6-routing

- ❶ `interface GigabitEthernet0/50`
- ❷ `description Linknett mellom FiG og Uninett/HiG`
- ❸ `no switchport`
- ❹ `ip address 128.39.70.170 255.255.255.252`
- ❺ `ip access-group InetIPv4Inn in`
- ❻ `ip access-group InetIPv4Ut out`
- ❼ `ip pim sparse-mode`
- ❽ `ip igmp version 3`
- ❾ `ipv6 address 2001:700:0:8074::2/64`
- ❿ `ipv6 nd ra suppress`
- ⓫ `ipv6 traffic-filter InetIPv6Inn in`
- ⓬ `ipv6 traffic-filter InetIPv6Ut out`



Konfigurasjon av IPv6

Cisco IOS: IPv6-routing

- ❶ Default route:
`ipv6 route ::/0 GigabitEthernet0/50 2001:700:0:8074::1`
- ❷ Nullroute linknettet, og offisielle og private adresser:
`ipv6 route 2001:700:0:8074::/64 Null0`
`ipv6 route 2001:700:1100::/48 Null0`
`ipv6 route FD5C:14CF:C300::/48 Null0`



Konfigurasjon av IPv6

Cisco IOS: IPv6-routing

- ❶ `interface Vlan48`
- ❷ `description Datarom 129`
- ❸ `ip address 128.39.194.193 255.255.255.224`
- ❹ `ip access-group Vlan48IPv4UtTil out`
- ❺ `ip helper-address 128.39.174.42`
- ❻ `ip pim sparse-dense-mode`
- ❼ `ip igmp version 3`
- ❽ `ipv6 address 2001:700:1100:8008::1/64`
- ❾ `ipv6 nd other-config-flag`
- ❿ `ipv6 nd router-preference High`
- ⓫ `ipv6 dhcp server offisiell`
- ⓬ `ipv6 traffic-filter Vlan48IPv6UtTil out`



Konfigurasjon av IPv6 I

Cisco IOS: ACL-er

- ❶ `configure terminal`
- ❷ `ipv6 access-list access-list-name`
- ❸ `deny | permit protocol {source-ipv6-prefix/prefix-length | any | host source-ipv6-address} [operator port-number] {destination-ipv6-prefix/prefix-length | any | host destination-ipv6-address} [operator port-number] [dest-option] [dest-option-type value] [dscp value] [flow-label value] [fragments] [hbh] [log] [log-input] [mobility] [mobility-type value] [reflect access-list-name] [routing] [routing-type value] [sequence value] [time-range name] [undetermined-transport]`



Konfigurasjon av IPv6 II

Cisco IOS: ACL-er

```
4 deny | permit tcp {source-ipv6-prefix/prefix-length | any |  
host source-ipv6-address} [operator port-number]  
{destination-ipv6-prefix/prefix-length | any |  
host destination-ipv6-address} [operator port-number]  
[ack] [dest-option] [dest-option-type value] [dscp value]  
[established] [fin] [flow-label value] [hbh] [log] [log-input]  
[mobility] [mobility-type value] [psh]  
[reflect access-list-name] [routing] [routing-type value]  
[rst] [sequence value] [syn] [time-range name] [urg]
```



Konfigurasjon av IPv6 III

Cisco IOS: ACL-er

```
5 deny | permit udp {source-ipv6-prefix/prefix-length | any |  
host source-ipv6-address} [operator port-number]  
{destination-ipv6-prefix/prefix-length | any |  
host destination-ipv6-address} [operator port-number]  
[dest-option] [dest-option-type value] [dscp value]  
[flow-label value] [hbh] [log] [log-input] [mobility]  
[mobility-type value] [reflect access-list-name] [routing]  
[routing-type value] [sequence value] [time-range name]
```



Konfigurasjon av IPv6 IV

Cisco IOS: ACL-er

```
6 deny | permit icmp {source-ipv6-prefix/prefix-length | any |  
host source-ipv6-address}  
{destination-ipv6-prefix/prefix-length | any |  
host destination-ipv6-address} [{icmp-type [icmp-code]} |  
icmp-message] [dest-option] [dest-option-type value]  
[dscp value] [flow-label value] [log] [log-input] [mobility]  
[mobility-type value] [reflect access-list-name] [routing]  
[routing-type value] [sequence value] [time-range name]  
7 evaluate reflexive-access-list-name [sequence value]  
8 remark comment  
9 exit  
Husk:  
operator ∈ {gt | lt | neq | eq | range}  
reflect er bare gyldig for permit-regler
```



Konfigurasjon av IPv6 V

Cisco IOS: ACL-er

```
10 interface interface-id  
11 ipv6 traffic-filter access-list-name {in | out}  
12 end
```



Konfigurasjon av IPv6 VI

Cisco IOS: ACL-er

- Alle IPv6-ACL-er har følgende 5 regler innebygget (eng. implicit) på slutten:
 - 1 permit icmp any any nd-na
 - 2 permit icmp any any nd-ns
 - 3 permit icmp any any router-advertisement
 - 4 permit icmp any any router-solicitation
 - 5 deny ipv6 any any
- Disse reglene tillater Neighbor Discovery, og blokkerer all annen IPv6-trafikk
- Dine egne regler kommer *alltid* før de 5 reglene over, og kanskje må du kopiere de innebygde reglene og gjøre dine egne justeringer, for eksempel slå på logging av blokkert trafikk



Konfigurasjon av IPv6 VII

Cisco IOS: ACL-er

- Ønsker du logging av blokkert trafikk, men vil samtidig ikke blokkere Neighbor Discovery, så må du gjøre slik:
 - 1 remark Øvrige regler kommer før denne linja
 - 2 permit icmp any any nd-na
 - 3 permit icmp any any nd-ns
 - 4 permit icmp any any router-advertisement
 - 5 permit icmp any any router-solicitation
 - 6 deny ipv6 any any log
 - 7 remark Her kommer de skjulte, implisitte reglene
 - 8 permit icmp any any nd-na
 - 9 permit icmp any any nd-ns
 - 10 permit icmp any any router-advertisement
 - 11 permit icmp any any router-solicitation
 - 12 deny ipv6 any any



Konfigurasjon av IPv6

Cisco IOS: DHCPv6

- ipv6 dhcp pool offisiell
 - dns-server 2001:700:1100:1::3
 - dns-server 2001:700:1100:1::2
 - domain-name fig.ol.no
 - sntp address 2001:700:1100:1::2
 - sntp address 2001:700:1100:1::3
 - sntp address 2001:700:1100:1::4
 - information refresh 0 2
- interface Vlan48
 - ipv6 dhcp server offisiell



Konfigurasjon av IPv6

Cisco IOS: DHCPv6

- ipv6 dhcp pool ULA
 - dns-server 2001:700:1100:1::3
 - dns-server 2001:700:1100:1::2
 - domain-name fig.netlocal
 - sntp address 2001:700:1100:1::2
 - sntp address 2001:700:1100:1::3
 - sntp address 2001:700:1100:1::4
 - information refresh 0 2
- interface Vlan31
 - ipv6 dhcp server ULA



- De fleste moderne operativsystemer har IPv6-støtte
- Windows 2000 har en eksperimentell IPv6-protokoll, men mangler DNS-oppslag for AAAA
- IPv6 må installeres manuelt i Windows XP og Server 2003
 - DNS-oppslag sendes alltid over **IPv4**
- IPv6 er påskrudd i Windows Vista, Server 2008 og nyere versjoner
- Linux og *BSD har hatt IPv6-støtte i lang tid
- Autokonfig med tilfeldig grensesnittidentifikator er det mest vanlige for skrivebordssystemer
- Manuell konfigurasjon er mest vanlig for serversystemer

Del XII

Noen RFC-er om IPv6

- Windows:
 - `netsh interface ipv6 set address "navn-på-grensesnitt" IPv6-adresse`
 - `netsh interface ipv6 set address "Lokal tilkobling" 2001:700:1100:8008::1337`
 - Konfigurasjon gjennom grafisk grensesnitt i «Kontrollpanelet» er også mulig
- *BSD:
 - `ifconfig navn-på-grensesnitt inet6 IPv6-adresse prefixlen prefikslengde`
 - `ifconfig em0 inet6 2001:700:1100:8008::1337 prefixlen 64`
 - Vanligvis lagres slike innstillingene permanent, for eksempel i `/etc/rc.conf`

Oversikt over del 12: Noen RFC-er om IPv6 I

60 Noen RFC-er om IPv6

- IPv6-spesifikasjon: [RFC 2460](#), [RFC 5095](#), [RFC 5722](#), [RFC 5871](#), [RFC 6437](#), [RFC 6564](#), [RFC 6935](#) og [RFC 6946](#)
- ICMPv6: [RFC 4443](#) og [RFC 4884](#)
- Neighbor Discovery: [RFC 4861](#), [RFC 5942](#) og [RFC 6980](#)
- Krav til IPv6-noder: [RFC 6434](#)
- Path MTU: [RFC 1981](#)
- DHCPv6: [RFC 3315](#), [RFC 3319](#), [RFC 3633](#), [RFC 3646](#), [RFC 3736](#), [RFC 4361](#), [RFC 5494](#), [RFC 6221](#), [RFC 6422](#), [RFC 6644](#) og [RFC 7083](#)
- Overføring av IPv6-pakker over Ethernet: [RFC 2464](#) og [RFC 6085](#)
- Adressearkitektur: [RFC 4291](#), [RFC 5952](#) og [RFC 6052](#)
- Unicastadresser: [RFC 3587](#)
- ULA: [RFC 4193](#)

- Autokonfigurering av adresser: [RFC 4862](#)
- Tilfeldig grensesnittidentifikator: [RFC 4941](#)
- Prefiks-baserte multicastadresser: [RFC 3306](#), [RFC 3956](#) og [RFC 4489](#)
- IPsec: [RFC 4301](#), [RFC 4302](#), [RFC 4303](#), [RFC 4304](#), [RFC 4307](#), [RFC 4308](#), [RFC 4309](#), [RFC 4312](#), [RFC 4835](#) og [RFC 5996](#)
- For programmerere av nettverksprogrammer: [RFC 3493](#), [RFC 3542](#) og [RFC 4038](#)
- Grunnleggende krav til IPv6-routere hos sluttbrukere (CER): [RFC 7084](#)