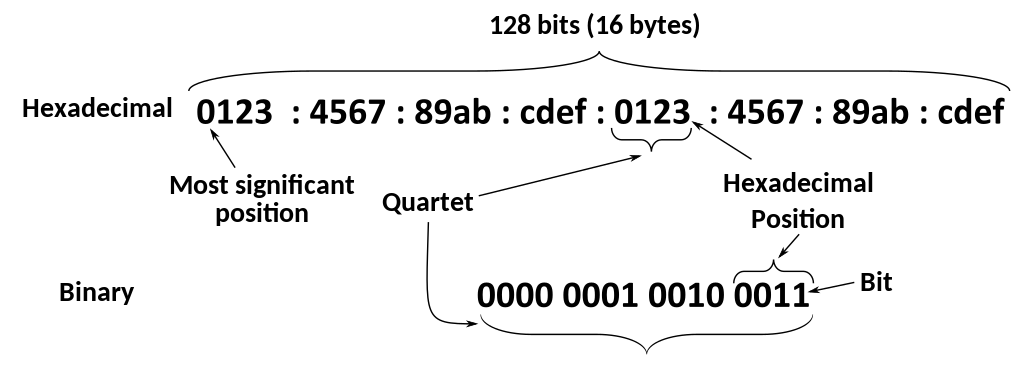
# IPv6, adresování – formát paketu, prefixy, adresace v lokání síti, porovnání s IPv4

* Jedná se o identifikační a lokalizační systém pro síťová zařízení na sítí a cesty skrze síť. Operuje na třetí vrstve OSI modelu
* Ipv6 byla vyvinuta Internet Engineering Task Force (IETF) kvůli očividnému problému s ipv4 adresami s kterýma se v té době plýtvalo a vypadalo že dojdou úplně a došli. Byly vyvinuty za účelem nahrazení už zastaralého a v některých aspektech nedokonalého ipv4 protokolu
* Místo dekadické soustavy využívá hexadecimální, která jde 0-9 A-F. Jedná se o 128bitovou adresu tedy adresu s 2^128 kombinací adres. Jedná se do osmi skupin (po čtyřech hexadecimálních čísel) oddělených dvojtečkou
* Zaměřili se zde na device mobility, security a konfigurovatelnost při designování tohoto protokolu.
* Poskytuje mnohem více adres, zjednodušuje způsob nastavení adres, zjednodušuje zpracování paketů na routerech z důvodu ponechání packet fragmentace na koncových uživatelých. Dovoluje nám tři rozdílné způsoby přenosu paketů:
  1. Unicast(one to one) … paket je poslán přímo jednomu zařízení (drtivá většina komunikace probíhá přez unicast)
  2. Anycast(one to one of many) … doručujeme paket jednomu z vybrané skupiny nejčastěji tomu kdo je nejblíže
  3. Multicast(one to many of many) … doručí paket skupině, doručuje simultánně při jednom přenosu



## Paket

Skládá se z headeru a pyloadu. Header se zkládá z fixních 40 bajtů (320 bitů)

Obsah obrázku text, snímek obrazovky, číslo, Písmo

Popis byl vytvořen automaticky

### Version (4 bits)

The constant 6 (bit sequence 0110). Specify version of protocol that was used.

### Traffic Class (6+2 bits)

The bits of this field hold two values. The six most-significant bits hold the differentiated services field (DS field), which is used to classify packets. Currently, all standard DS fields end with a '0' bit. Any DS field that ends with two '1' bits is intended for local or experimental use.

The remaining two bits are used for Explicit Congestion Notification (ECN); priority values subdivide into ranges: traffic where the source provides congestion control and non-congestion control traffic.

### Flow Label (20 bits)

A high-entropy identifier of a flow of packets between a source and destination. A flow is a group of packets, e.g., a TCP session or a media stream. The special flow label 0 means the packet does not belong to any flow (using this scheme). An older scheme identifies flow by source address and port, destination address and port, protocol (value of the last Next Header field). It has further been suggested that the flow label be used to help detect spoofed packets.

### Payload Length (16 bits)

The size of the payload in octets, including any extension headers. The length is set to zero when a Hop-by-Hop extension header carries a Jumbo Payload option.

### Next Header (8 bits)

Specifies the type of the next header. This field usually specifies the transport layer protocol used by a packet's payload. When extension headers are present in the packet this field indicates which extension header follows. The values are shared with those used for the IPv4 protocol field, as both fields have the same function (see List of IP protocol numbers).

### Hop Limit (8 bits)

Replaces the time to live field in IPv4. This value is decremented by one at each forwarding node and the packet is discarded if it becomes 0. However, the destination node should process the packet normally even if received with a hop limit of 0.

### Payload

Obsahuje data packet z vyšších vrstev navíc může obsahovat I extension header

### Extension headers

carry optional [internet layer](https://en.wikipedia.org/wiki/Internet_layer) information and are placed between the fixed header and the upper-layer protocol header. Extension headers form a chain, using the *Next Header* fields. The *Next Header* field in the fixed header indicates the type of the first extension header; the *Next Header* field of the last extension header indicates the type of the upper-layer protocol header in the payload of the packet. All extension headers are a multiple of 8 octets in size; some extension headers require internal padding to meet this requirement.

Bez speciálních option pyload musí být menší než 64kB. S Jumbo Payload option (in a Hop-By-Hop Options extension header) payload může dosahovat velikosti 4GB.

Narozdíl od ipv4, router nikdy nefragmentuje paket. Pokud docházi k fragmentaci tak jenom na klientovy ovšem klient se za pomocí PMTUD(Path MTU discovery) snaží určit maximum transmit unit (MTU) velikost ma síti mezi dvěma hostama normálně s cílem se vyhnou fragmentaci.

## Rozsahy

### Globální internet

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 64:ff9b::/96 | 64:ff9b::0.0.0.0 | 64:ff9b::255.255.255.255 | 232 | Global Internet | IPv4/IPv6 translation |
| 2001:0000::/32 | 2001:: | 2001::ffff:ffff:ffff:ffff:ffff:ffff | 296 | Global Internet | [Teredo tunneling](https://en.wikipedia.org/wiki/Teredo_tunneling) |
| **ff00::/8** | **ff00::** | **ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff** | **2120** | **Global Internet** | [**Multicast address**](https://en.wikipedia.org/wiki/Multicast_address#IPv6) |

### Privátní internet

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 64:ff9b:1::/48 | 64:ff9b:1:: | 64:ff9b:1:ffff:ffff:ffff:ffff:ffff | 280, with 248 for each IPv4 | Private internets | IPv4/IPv6 translation |
| fc00::/7 | fc00:: | fdff:ffff:ffff:ffff:ffff:ffff:ffff:ffff | 2121 | Private internets | [Unique local addres](https://en.wikipedia.org/wiki/Unique_local_address) |

### L ink local

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **fe80::/64 from fe80::/10** | **fe80::** | **fe80::ffff:ffff:ffff:ffff** | **264** | **Link** | [**Link-local address**](https://en.wikipedia.org/wiki/Link-local_address#IPv6) |

### Loop back

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **::1/128** | **::1** | **::1** | **1** | **Host** | [**Loopback address**](https://en.wikipedia.org/wiki/Loopback_address)**—a virtual interface that loops all traffic back to itself, the *local host*** |

Software

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **::/128** | **::** | **::** | **1** | **Software** | **Unspecified address** |
| ::ffff:0:0/96 | ::ffff:0.0.0.0 | ::ffff:255.255.255.255 | 2128 − 96 = 232 = 4294967296 | Software | IPv4-mapped addresses |
| ::ffff:0:0:0/96 | ::ffff:0:0.0.0.0 | ::ffff:0:255.255.255.255 | 232 | Software | IPv4 translated addresses |
| 2001:20::/28 | 2001:20:: | 2001:2f:ffff:ffff:ffff:ffff:ffff:ffff | 2100 | Software | [ORCHIDv2](https://en.wikipedia.org/w/index.php?title=ORCHIDv2&action=edit&redlink=1) |

### GUA vs LLA

Každé zařízení vyžaduje Link Local adresu (jedná se o unicast network adres) s prefixem fe80::/10 následující 54 bity pro subnetting ale většinou jsou nulové a 64 bitů pro identifikaci zařízení. link-local address autoconfiguration. Link local adresa není routable mimo tvojí lokalní síť narozdíl od GUA.

GUA (Global unicast address) narozdíl od LLA je routable i z internetu tedy mimo lokální síť. Prefix GUA je tvořen router prefixem (pokud je nastavení GUA za pomocí SLAC) a druhá část je náhodně vygenerována nebo vytvořena z mac adresy. Pokud je využit DHCPv6 server v jakém tvaru bude adresa rozhoduje on. Ovšem LLA se stale bude nastavovat automaticky pokud je nenastavíme staticky.

Zařízení na síti si může přiřadit samo ipv6 adresu bez kooperace s DHCP serverem tento process se nazývá link-loacal address autoconfiguration nebo SLAC(state less auto configuration).

* Vyváříme ji pomocí generace nějakého náhodného čísla
* Vytváříme za pomocí mac adressy (EUI-64 Format)
* Poté co se nastaví hodnota ip adresy pošle se speciální DAD (Duplicate Address Detection) packet. Pokud žádný host na síti neodpoví zřízení si ipv6 adresu ponechá.

## Ipv4 vs Ipv6

| **IPv4** | **IPv6** |
| --- | --- |
| Are more expensive (public ip address) because of smaller amount that is available. | Cheaper because of really big address space (everyone can have public ip address) |
| IPv4 has a 32-bit address length | IPv6 has a 128-bit address length |
| It Supports Manual and DHCP address configuration | It supports Auto and renumbering address configuration |
| In IPv4 end to end, connection integrity is Unachievable | In IPv6 end-to-end, connection integrity is Achievable |
| It can generate 4.29×109 address space. That equals 2^32. | The address space of IPv6 is quite large it can produce 3.4×1038 address space. That equals 2^128. |
| The Security feature is dependent on the application | IPSEC is an inbuilt security feature in the IPv6 protocol |
| Address representation of IPv4 is in decimal | Address Representation of IPv6 is in hexadecimal |
| Fragmentation performed by Sender and forwarding routers | In IPv6 fragmentation is performed only by the sender |
| In IPv4 Packet flow identification is not available | In IPv6 packet flow identification are Available and uses the flow label field in the header |
| In IPv4 checksum field is available | In IPv6 checksum field is not available |
| It has a broadcast Message Transmission Scheme | In IPv6 multicast and anycast message transmission scheme is available |
| In IPv4 Encryption and Authentication facility not provided | In IPv6 Encryption and Authentication are provided |
| IPv4 has a header of 20-60 bytes. | IPv6 has a header of 40 bytes fixed |
| IPv4 can be converted to IPv6 | Not all IPv6 can be converted to IPv4 |
| IPv4 consists of 4 fields which are separated by addresses dot (.) | IPv6 consists of 8 fields, which are separated by a colon (:) |
| IPv4’s  IP addresses are divided into five different classes. Class A , Class B, Class C, Class Da , Class E. | IPv6 does not have any classes of the IP address. |
| IPv4 supports VLSM(Variable Length subnet mask). | IPv6 does not support VLSM. |
| Example of IPv4:  66.94.29.13 | Example of IPv6: 2001:0000:3238:DFE1:0063:0000:0000:FEFB |