# Address Resulution Protokoll (ARP)

# Kurzwiederholung Adressierung

- Vermittlung von Daten zwischen Endgeräten egal ob im selben oder unterschiedlichen Netzen werden zwei Adressen benötigt
  - Data Link Layer
    - Ethernet Protokoll
      - MAC Adresse
  - Network Layer
    - IP Protokoll
      - IP Adresse

Anmerkung: Für beide Layer gibt es auch andere Protokolle, mit anderen Adressierungen, jedoch sind diese beiden der de-facto Standard

# Adressen des Ziel Endgeräts

- Nachdem man beide Adressen zur Kommunikation benötigt, stellt sich die Frage, „wie kommt man an diese Adressen?“

- IP Adresse
  - Einige lokale IPv4 Adresse weiß oft z.B. Gateway (Router über den das lokale Netz verlassen wird)
  - Über die Namensauflösung (siehe DNS Service in einer späteren Einheit)

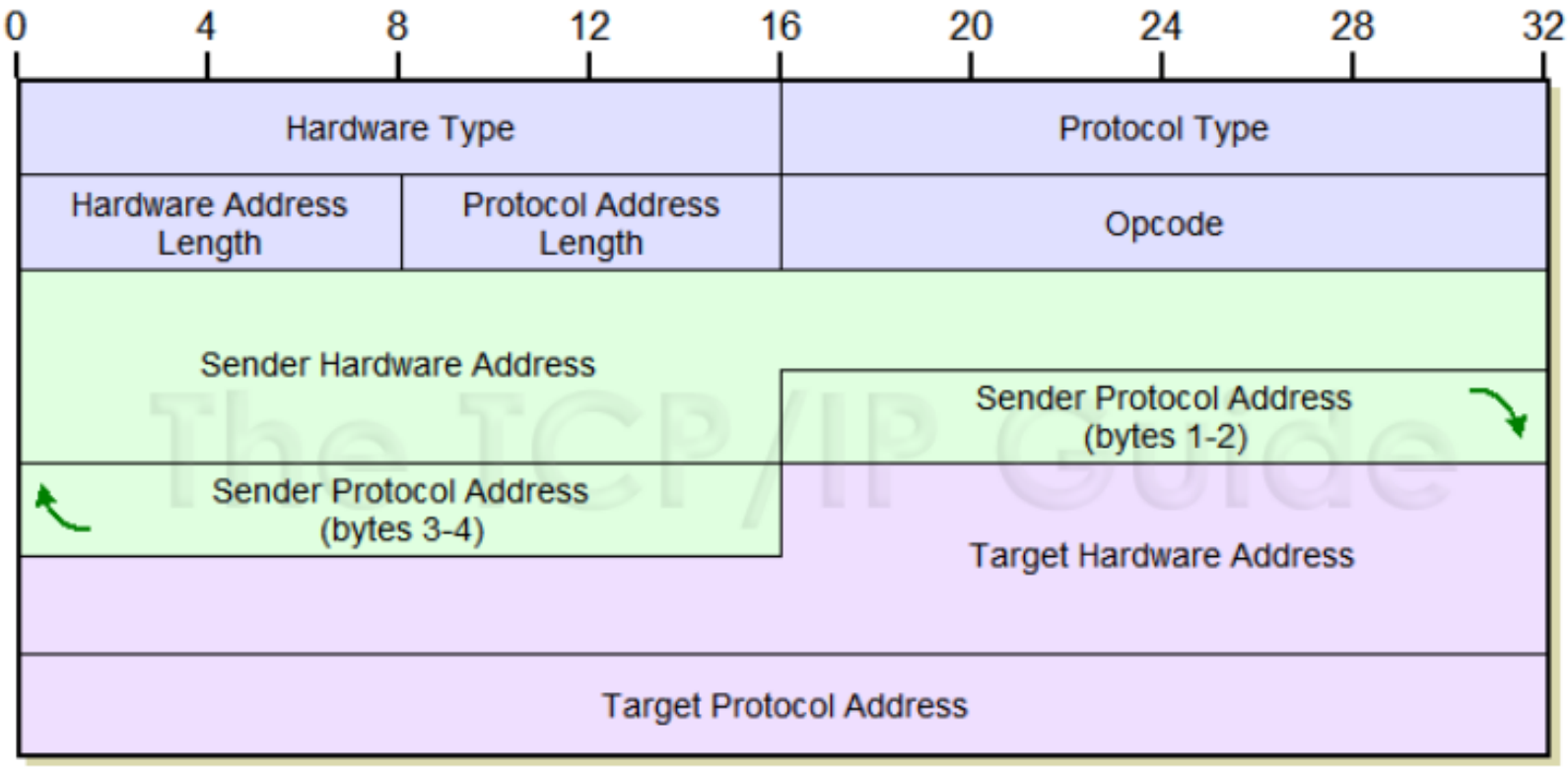
- MAC Adresse
  - Über das Address Resolution Protokoll und der bekannten IP Adresse

# Übliche Ablauf einer Datenübertragung

- Daraus ergibt sich üblicherweise folgender recht grobe Ablauf:
  1. Eingabe des Namens des Ziel Endgeräts z.B. URL in den Browser
  2. Ermittlung der IP Adresse mittels DNS Protokoll
  3. Abfrage der MAC Adresse mittels ARP Protokoll
  4. Kommunikationsaufbau mit dem Ziel Endgerät

- Dieser Ablauf ist sehr grob beschrieben, in der Lehrveranstaltung wird noch genauer auf die einzelne Prozessschritte eingegangen werden

# ARP Message Format




Kozierok, Charles M. "The TCP/IP Guide" The TCP/IP Guide: A Comprehensive, Illustrated Internet Protocols Reference, www.tcpipguide.com. Accessed 6 May 2020.

# ARP Message Format

- Hardware Type
  - Das ARP Protokoll funktioniert nicht nur mit dem Ethernet und IP Protokoll, über den Hardware Type wird mitgeteilt um welche Adressen es sich in den Hardware Adressenfeldern handelt

| *HRD* Value | Hardware Type |
|---|---|
| 1 | Ethernet (10 Mb) |
| 6 | IEEE 802 Networks |
| 7 | ARCNET |
| 15 | Frame Relay |
| 16 | Asynchronous Transfer Mode (ATM) |
| 17 | HDLC |
| 18 | Fibre Channel |
| 19 | Asynchronous Transfer Mode (ATM) |
| 20 | Serial Line |

Mögliche Hardware Typen[2]

- Protocol Type
  - Hier steht die IEEE 802 Codenummer[1] die Protokoll Adressen
    - z.B. für IPv4 2048 (0800 hex)


1. "IEEE 802 Numbers." *www.Iana.org*, www.iana.org/assignments/ieee-802-numbers/ieee-802-numbers.xhtml. Accessed 6 May 2020.
2. Kozierok, Charles M. "The TCP/IP Guide - ARP Message Format." The TCP/IP Guide: A Comprehensive, Illustrated Internet Protocols Reference, www.tcpipguide.com/free/t_ARPMessageFormat.htm. Accessed 6 May 2020.

# ARP Message Format

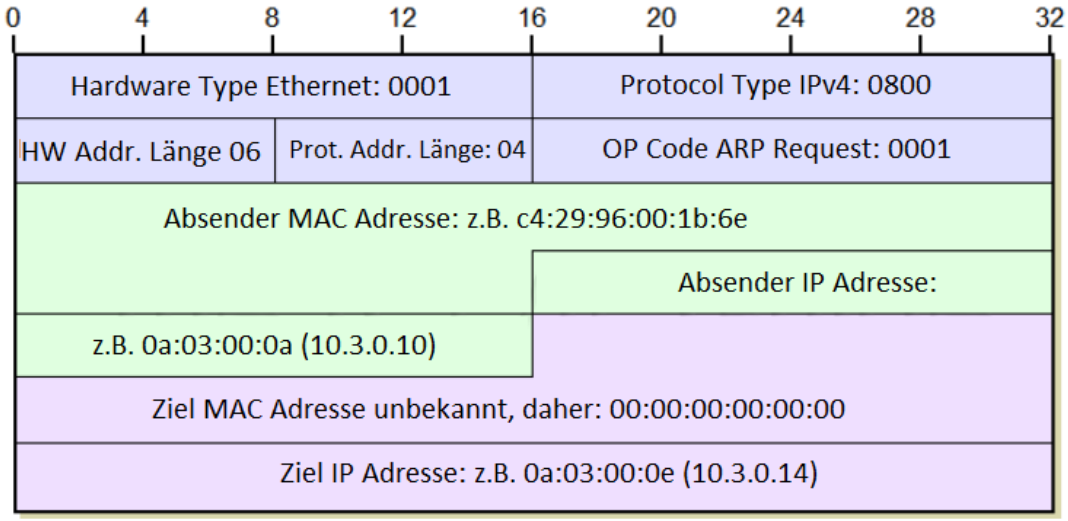
FH University of Applied Sciences TECHNIKUM WIEN

- Hardware & Protocol Address Length
  - Beide Adressen haben keine fixe Länge, diese ist abhängig von den gewählten Typen
  - Immer wenn in Netzwerk Protokollen Attribute keine fixe Länge haben muss diese in einem Feld extra angegeben werden

- Opcode
  - Dieser bestimmt die Art der ARP Nachricht
    - 1 - ARP Request
    - 2 - ARP Reply
    - Es gibt weite Typen die heute kaum mehr verwendet werden

# ARP Prozessablauf

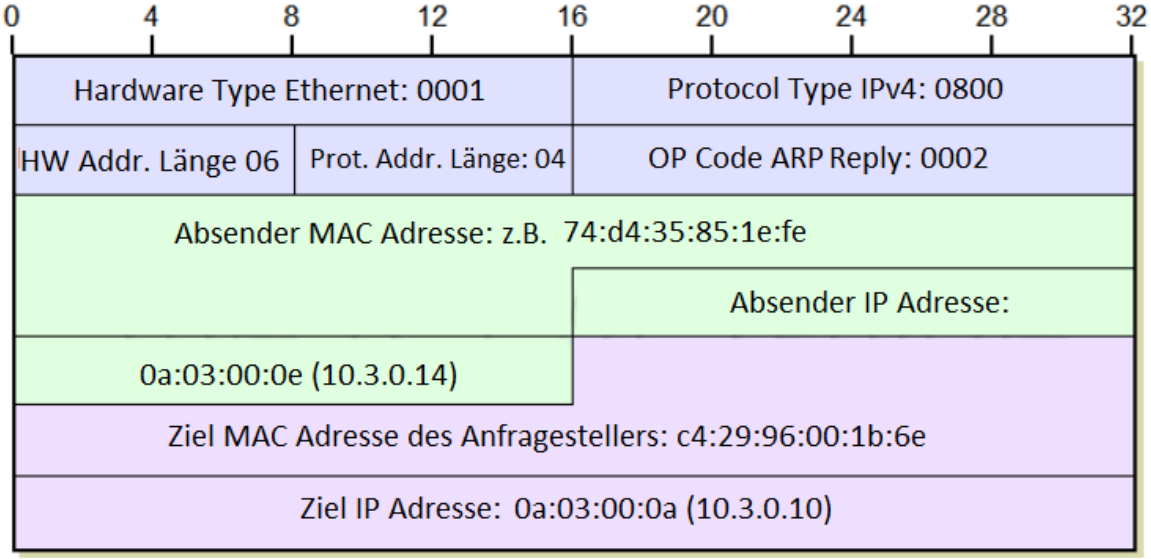
1. Check ob MAC Adresse im ARP Cache
2. ARP Request (sofern 1. negativ verläuft)
   - Broadcast an alle Netzteilnehmer im lokalen Netz
   - d.h. im Ethernet Header wird als Ziel MAC Adresse eingegeben: FF:FF:FF:FF:FF:FF
   - Die Werte in der Beispielgrafik sind wie in Protokolldarstellungen üblich Hexadezimalwerte



# ARP Prozessablauf

3. Alle Empfänger des ARP Requests überprüfen ob die angefragte IP Adresse die eigene ist
   - Handelt es sich um eine andere wird die Nachricht verworfen
   - Ist es die eigene wird ein ARP Reply generiert
4. MAC Adresse und IP Adresse werden im ARP Cache eingetragen
5. ARP Reply wird versendet

- Nachricht erfolgt „unicast“ d.h. im Ethernet Header wird die MAC Adresse des ARP Request Erstellers eingetragen



# ARP Prozessablauf

6. Der Empfänger trägt ebenfalls die MAC und IP Adresse in seinen ARP Cache ein

- Nun hat der Absender alle Informationen, die er benötigt um Daten an den Empfänger zu senden
- Wichtig anzumerken ist, dass das ARP Protokoll ausschließlich im eigenen LAN eingesetzt werden kann
- Wenn Sie mit einem Empfänger außerhalb Ihres LANs kommunizieren wollen, ist Ihr Ziel Endgerät für den ARP Prozess Ihr Gateway Router, d.h. wir möchten die Gateway Router MAC Adresse erfahren!

# Windows ARP Cache

```
C:\Users\Christian>arp /?

Displays and modifies the IP-to-Physical address translation tables used by
address resolution protocol (ARP).

ARP -s inet_addr eth_addr [if_addr]
ARP -d inet_addr [if_addr]
ARP -a [inet_addr] [-N if_addr] [-v]

  -a            Displays current ARP entries by interrogating the current
                protocol data.  If inet_addr is specified, the IP and Physical
                addresses for only the specified computer are displayed.  If
                more than one network interface uses ARP, entries for each ARP
                table are displayed.
  -g            Same as -a.
  -v            Displays current ARP entries in verbose mode.  All invalid
                entries and entries on the loop-back interface will be shown.
  inet_addr     Specifies an internet address.
  -N if_addr    Displays the ARP entries for the network interface specified
                by if_addr.
  -d            Deletes the host specified by inet_addr. inet_addr may be
                wildcarded with * to delete all hosts.
  -s            Adds the host and associates the Internet address inet_addr
                with the Physical address eth_addr.  The Physical address is
                given as 6 hexadecimal bytes separated by hyphens. The entry
                is permanent.
  eth_addr      Specifies a physical address.
  if_addr       If present, this specifies the Internet address of the
                interface whose address translation table should be mo
                If not present, the first applicable interface will be
Example:
  > arp -s 157.55.85.212   00-aa-00-62-c6-09  .... Adds a static entry.
  > arp -a                                    .... Displays the arp tal
```

- Dynamische ARP Einträge „altern“ wieder aus den Cache
- Statische sind permanent eingetragen

```
C:\Users\Christian>arp -a

Interface: 10.3.0.11 --- 0x1a
  Internet Address      Physical Address      Type
  10.3.0.1              dc-ef-09-a2-7e-6c     dynamic
  10.3.0.5              24-5e-be-0d-f0-0f     dynamic
  10.3.0.10             00-17-88-22-12-c0     dynamic
  10.3.0.13             f0-81-73-e6-80-f9     dynamic
  10.3.0.18             08-c5-e1-94-ae-65     dynamic
  10.3.0.19             f4-96-34-2c-0b-ff     dynamic
  10.3.0.20             c0-d2-dd-20-67-ad     dynamic
  10.3.0.21             68-ec-c5-c8-0c-53     dynamic
  10.3.0.23             6c-c7-ec-2c-ea-2a     dynamic
  10.3.0.29             58-82-a8-03-42-5a     dynamic
  10.3.0.255            ff-ff-ff-ff-ff-ff     static
  224.0.0.2             01-00-5e-00-00-02     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  224.0.0.253           01-00-5e-00-00-fd     static
  239.0.0.250           01-00-5e-00-00-fa     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

Nachdem die MAC/ IP Adressen Kombination auch statisch eingetragen werden kann, warum brauchen wir das ARP Protokoll?

# Windows ARP Cache Einstellungen ab Vista

```
C:\Users\Kaufmann>Netsh interface ipv4 show interface

Idx     Met         MTU          State                Name
---  ----------  ----------  ------------  ---------------------------
  1          50  4294967295  connected     Loopback Pseudo-Interface 1
 11          50        1500  disconnected  Drahtlosnetzwerkverbindung
 12          20        1500  connected     LAN-Verbindung
 16          20        1500  connected     VMware Network Adapter VMnet1
 17          20        1500  connected     VMware Network Adapter VMnet8
```

```
C:\Users\Kaufmann>Netsh interface ipv4 show interface 12

Parameter für die Schnittstelle LAN-Verbindung
----------------------------------------------
Schnittstellen-LUID                         : ethernet_6
Schnittstellenindex                         : 12
Status                                      : connected
Metrik                                      : 20
Verbindungs-MTU                             : 1500 Bytes
Erreichbare Zeit                            : 18000 ms
Erreichbare Basiszeit                       : 30000 ms
Intervall für die erneute Übertragung       : 1000 ms
DAD-Übertragungen                           : 3
Standortpräfixlänge                         : 64
Standort-ID                                 : 1
Weiterleitung                               : disabled
Ankündigung                                 : disabled
Nachbarermittlung                           : enabled
Nachbar-Nichterreichbarkeitserkennung       : enabled
Routersuche                                 : dhcp
Verwaltete Adresskonfiguration              : enabled
Andere statusbehaftete Konfiguration        : enabled
Schwacher Host sendet                       : disabled
Schwacher Host empfängt                     : disabled
Automatische Metrik verwenden               : enabled
Standardrouten ignorieren                   : disabled
Angekündigte Routergültigkeitsdauer         : 1800 Sekunden
Standardroute ankündigen                    : disabled
Aktuelles Hoplimit                          : 0
ARPND-Reaktivierungsmuster erzwingen        : disabled
Gerichtetes MAC-Reaktivierungsmuster        : disabled
```

| | |
|---|---|
| BaseReachable Zeit | 30.000 Millisekunden (ms) |
| MIN_RANDOM_FACTOR | 0,5 |
| MAX_RANDOM_FACTOR | 1.5 |

- 30x0,5=15 Sek.
- 30x1,5=45 Sek.

# Internetprotocol

## Version 4 & 6

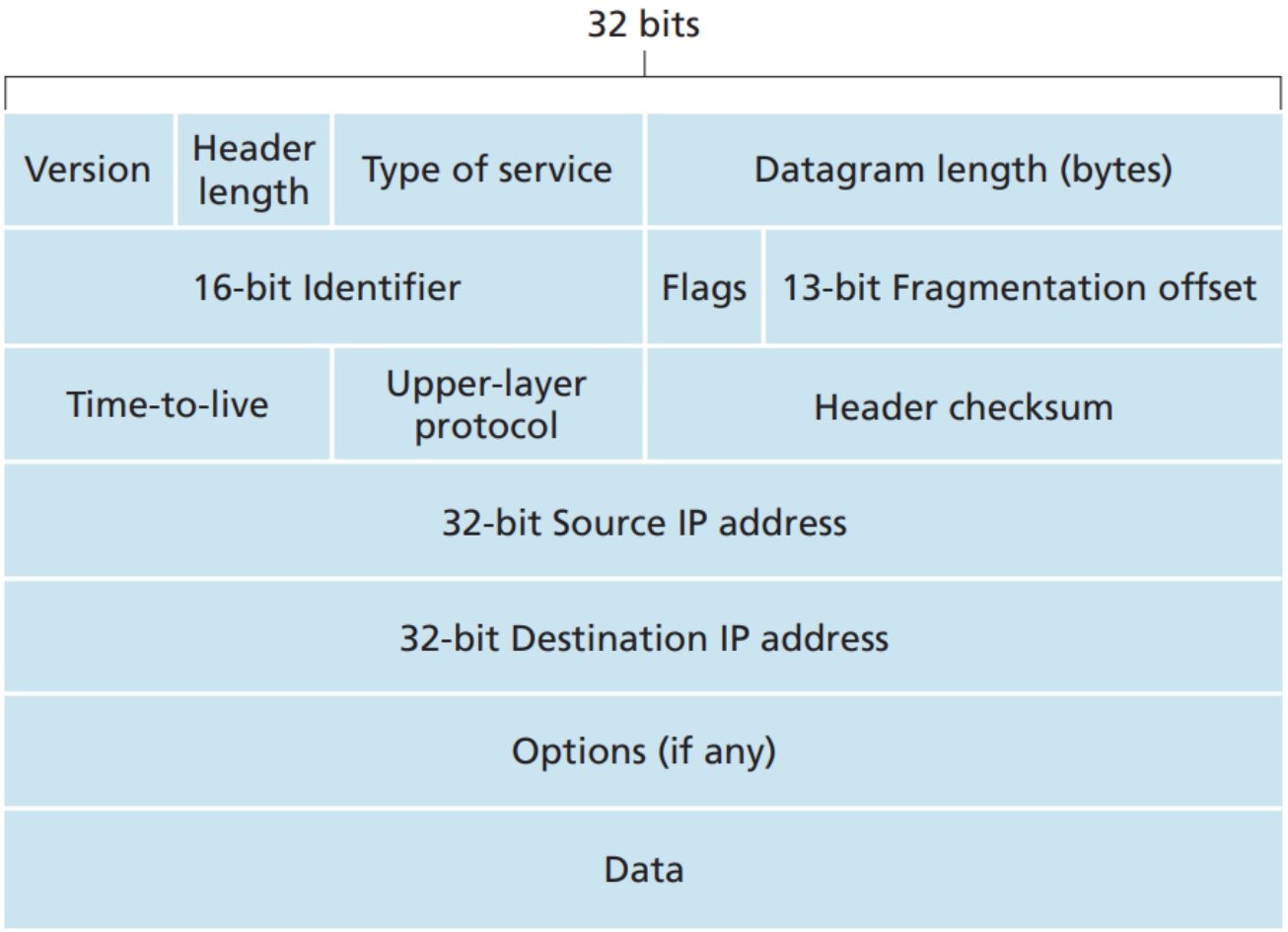
# IP Protokoll

- IP Protokoll ist der Kern der TCP/IP Protokoll Suite und wichtigstes Protokoll der Vermittlungsschicht

- Derzeit sind zwei Versionen des IP Protokolls im Einsatz
  - IPv4 & IPv6

- Hauptaufgabe des IP Protokoll
  - Vermittlung von Daten zwischen Endgeräten in unterschiedlichen Netzen (IPv4, IPv6)
  - Fragmentierung und Reassemblierung der Datagramme (IPv4)

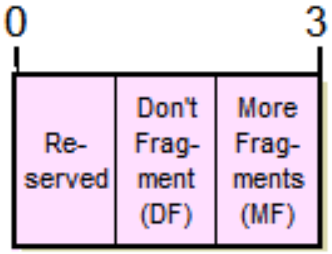
# IPv4 Header




Kurose, James F, and Keith W Ross. Computer Networking : A Top-down Approach. Boston, Mass., Pearson, 2017.

# IPv4 relevante Headerfields (1)

- Version
  - Version des IP Protokolls 4 für IPv4 6 für IPv6
- Header Length
  - Gibt die Größe des IP Headers in 32 Bit Einheiten z.B. 5 (=5*32bit groß)
- 16-bit Identifier
  - Dient zum Reassimblieren fragmentierter Datagramme
  - zusammengehörige Fragmentierte Datagramme haben immer dieselbe Identifikationsnummer – sonst würde man nicht wissen zu welchem Datagramm das jeweilige Fragment gehört
- Flags (3 Bit)

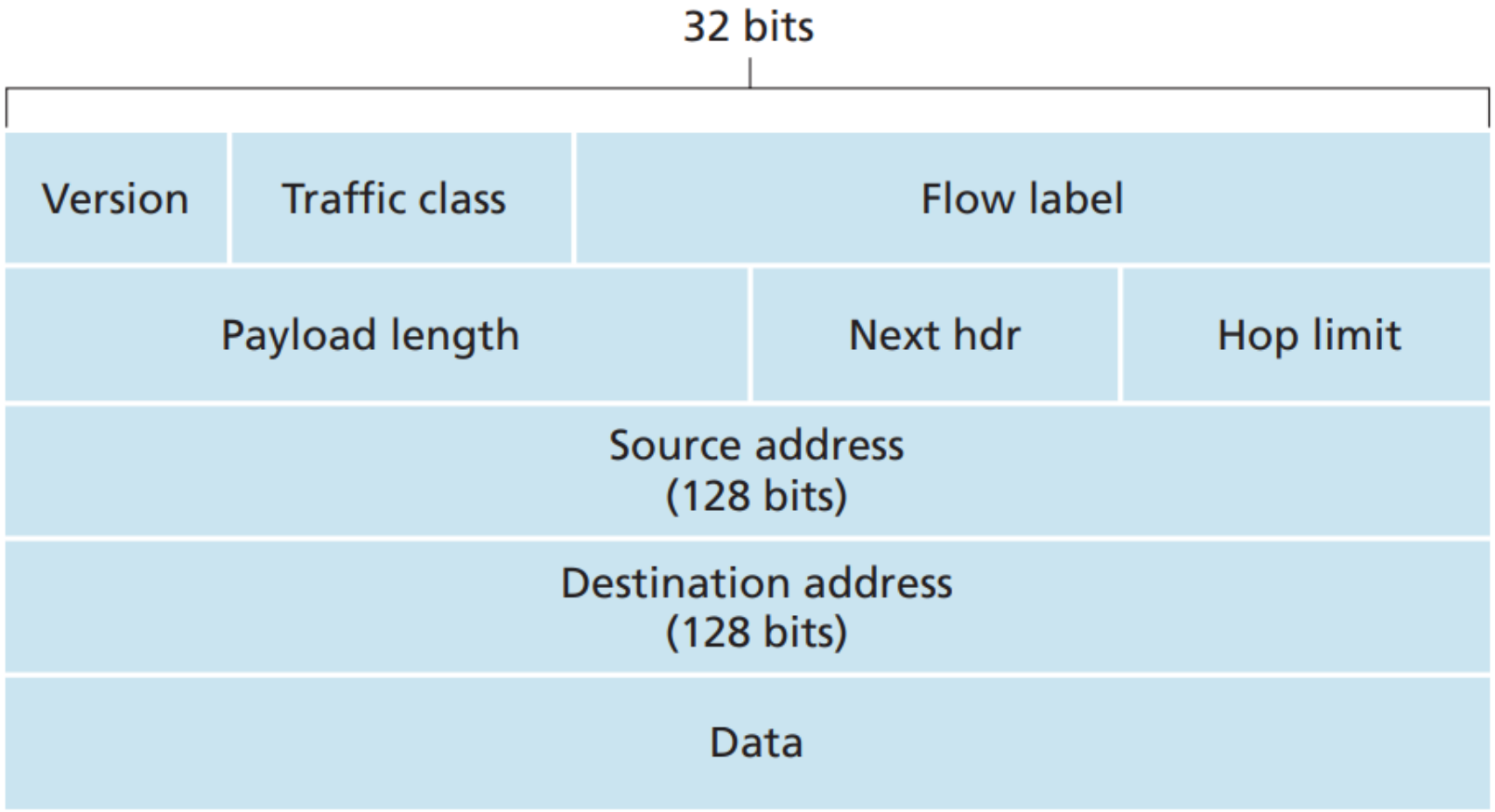


# IPv4 relevante Headerfields (2)

- 13-bit Fragmentation Offset
  - Zeigt in fragmentierten Datagrammen, an welcher Position sich das Fragment im original Datagramm befunden hat

- Time-to-live (TTL)
  - Wird vom Datagrammsender gesetzt und bei jedem Hop (Passieren eines Routers) um eines verringert
  - Erreicht das Feld den Wert 0 wird das Datagramm vom Router verworfen und eine ICMP Error Nachricht wird an den Absender gesendet (siehe dazu das ICMP Protokoll)

# IPv6 Header




Kurose, James F, and Keith W Ross. Computer Networking : A Top-down Approach. Boston, Mass., Pearson, 2017.

# IPv6 relevante Headerfields

- Version
  - Version des IP Protokolls (6 für IPv6)
- Payload Length
  - Größe des Payloads inklusive der Extension Header
- Next Header
  - Code des nächsten Headers
- Hop Limit
  - Ersatz für das TTL von IPv4
  - selbe Funktionalität

| Value (Hexadecimal) | Value (Decimal) | Protocol / Extension Header |
|---|---|---|
| 00 | 0 | Hop-By-Hop Options Extension Header (note that this value was "Reserved" in IPv4) |
| 01 | 1 | ICMPv4 |
| 02 | 2 | IGMPv4 |
| 04 | 4 | IP in IP Encapsulation |
| 06 | 6 | TCP |
| 08 | 8 | EGP |
| 11 | 17 | UDP |
| 29 | 41 | IPv6 |
| 2B | 43 | Routing Extension Header |
| 2C | 44 | Fragmentation Extension Header |
| 2E | 46 | Resource Reservation Protocol (RSVP) |
| 32 | 50 | Encrypted Security Payload (ESP) Extension Header |
| 33 | 51 | Authentication Header (AH) Extension Header |
| 3A | 58 | ICMPv6 |
| 3B | 59 | No Next Header |
| 3C | 60 | Destination Options Extension Header |

Kozierok, Charles M. "The TCP/IP Guide" The TCP/IP Guide: A Comprehensive, Illustrated Internet Protocols Reference, www.tcpipguide.com. Accessed 6 May 2020.

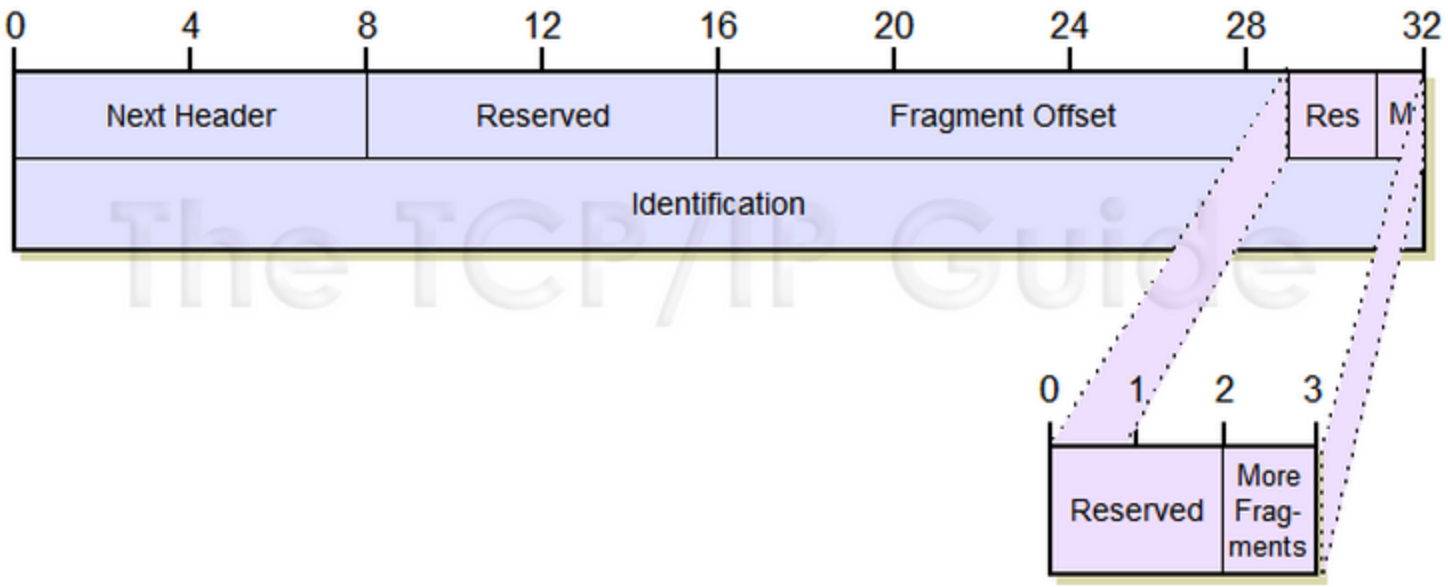
# IPv6 Extension Header

- Das Feld Next-Header weist entweder auf den Beginn der Protokolle höherer Layer z.B. TCP oder auf IPv6 Extension Header
- Das IPv6 Extension Header gewährt mehr Flexibilität da nachträgliche Erweiterungen implementiert werden können ohne den IP Header zu verändern
- Derzeit gibt es 7 Extension Header

# IPv6 Extension Header Übersicht

| Name | Größe | Beschreibung |
|---|---|---|
| Hop-By-Hop-Options | variabel | Optionen, die von allen IPv6-Geräten beachtet werden müssen |
| Routing | variabel | Dadurch kann den Weg des Paketes durch das Netzwerk beeinflusst werden |
| Fragment | 64 Bit | Parameter der Fragmentierung |
| Authentication-Header (AH) | variabel | Daten, die die Integrität des Paketes sicherstellen können (siehe IPsec) |
| Encapsulating Security Payload (ESP) | variabel | Daten, die die Vertraulichkeit (Verschlüsselung) des Paketes sicherstellen (siehe IPsec) |
| Destination Options | variabel | Optionen, die nur vom Zielrechner des Paketes beachtet werden müssen |
| No Next Header | leer | Platzhalter, um das Ende anzuzeigen |

# IPv6 Fragment Extension Header




- Kaum ein Unterschied zur IPv4 Fragmentierung
- Kein „Don‘t Fragment“ Flag weil ohnehin kein Router fragmentieren darf.
- Offset Angabe ebenfalls in 8 Byte Blöcken wie in IPv4



Kozierok, Charles M. “The TCP/IP Guide” The TCP/IP Guide: A Comprehensive, Illustrated Internet Protocols Reference, www.tcpipguide.com. Accessed 6 May 2020.

# IPv6 Fragmentierung

- Router dürfen nicht fragmentieren
- Daher muss der Absender dafür sorge tragen, dass die Datagramme klein genug sind
- Dazu gibt es 2 Varianten
  - Default MTU
    - Man verwendet generell eine MTU size von 1280
    - praktisch alle physischen Netze haben höhere MTU und können daher 1280 problemlos weiterleiten
  - Path MTU Discovery
    - Der Absender sendet ein IP Datagramm einer bestimmten Größe, kommt kein ICMPv6 „Packet too big“ Nachricht ist die MTU klein genug
    - Kommt die Nachricht, wird es mit immer kleineren Datagrammen versucht bis keine ICMPv6 Error-Message mehr kommt

# IPv6 EUI-64-Interface-ID

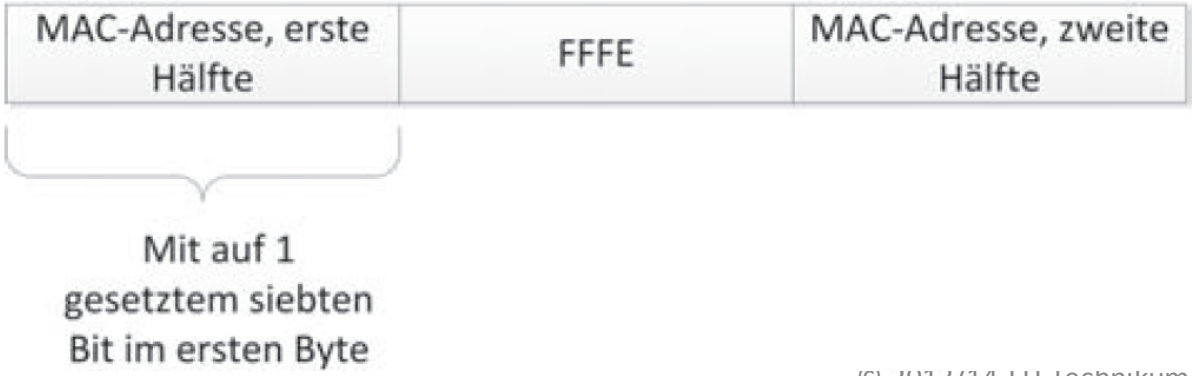
- Wie bereits in der letzten LVA erwähnt stehen dem Sysadmin zumindest 64 Bit für die Interface-ID zur Verfügung

- Diese kann theoretisch verkleinert werden wenn mehr Subnetze benötigt (mehr als $2^{16}$ Subnetze), meist bleibt es aber bei 64 Bit was auch die Autokonfiguration ermöglicht

- Grund hierfür ist das die MAC Adresse für die Autokonfiguration verwendet werden kann

# Warum 64 Bit - die MAC Adresse ist doch nur 6 Byte lange???

- Die MAC Adresse wird zunächst in 2x3Byte große Blöcke geteilt
- Zwischen die beiden Blöcke wird FFFE eingefügt
  - FFFE ist reserviert und wird von keinem NIC Hersteller verwendet
- Im ersten Block wird das 7. Bit im 1. Byte auf 1 gesetzt
  - Ansonsten werden Pakete dieser Adresse nicht ins Internet geroutet



# IPv6 statische Konfiguration

- Dafür gibt es zwei Optionen
  - Konfiguration einer vollständigen IPv6 Adresse
  - Konfiguration des 64-Bit-Präfixes

- Im 2. Fall ergänzt der Host automatisch die EUI-64-Interface-ID

# Statische Konfiguration - Windows

- Am einfachsten über das Kommandopromt

```
C:\>netsh interface ipv6 set address interface=11 address=fd00:0815:0814:1::1/64
```

- Die Interface ID bekommt man zum Beispiel über

```
C:\>route print
===========================================================================
Interface List
 11...74 d4 35 85 1e fe ......Intel(R) 82579LM Gigabit Network Connection
 14...00 50 56 c0 00 01 ......VMware Virtual Ethernet Adapter for VMnet1
 16...00 50 56 c0 00 08 ......VMware Virtual Ethernet Adapter for VMnet8
  1...........................Software Loopback Interface 1
 12...00 00 00 00 00 00 00 e0 Microsoft ISATAP Adapter
 13...00 00 00 00 00 00 00 e0 Teredo Tunneling Pseudo-Interface
 15...00 00 00 00 00 00 00 e0 Microsoft ISATAP Adapter #2
 17...00 00 00 00 00 00 00 e0 Microsoft ISATAP Adapter #3
===========================================================================
```

# Statische Konfiguration - Windows

```
Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) 82579LM Gigabit Network Connecti
on
   Physical Address. . . . . . . . . : 74-D4-35-85-1E-FE
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   IPv6 Address. . . . . . . . . . . : fd00:815:814:1::1(Preferred)
   Link-local IPv6 Address . . . . . : fe80::8b5:e510:ef22:4e22%11(Preferred)
   IPv4 Address. . . . . . . . . . . : 10.1.0.11(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Dienstag, 10. Juni 2014 08:11:12
   Lease Expires . . . . . . . . . . : Dienstag, 10. Juni 2014 16:11:13
   Default Gateway . . . . . . . . . : 10.1.0.1
   DHCP Server . . . . . . . . . . . : 10.1.0.1
   DHCPv6 IAID . . . . . . . . . . . : 242537525
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-1A-F3-98-A9-74-D4-35-85-1E-FE
```

# IPv6 stateless Autoconfig

- Methode bei der ein Host das im Subnetz genutzte 64-Bit-Präfix lernt und den Rest seiner IP Adresse (seine Interface-ID) ergänzt

- Dazu wird das Neighbor-Discovery-Protocol (NDP) verwendet
  - Router-Solicitation (RS) Nachricht
  - Router-Advertisment (RA) Nachricht

# IPv6 Stateless Autoconfig - Ablauf

1. Host generiert eine Link-Local-Adresse (LLA)
2. Host sendet NDP-Neighbor-Solicitation Nachricht mit seiner LLA
3. Kommt keine Antwort auf 2. gibt es diese LLA noch nicht im Netz daher wird die LLA den Interface zugeordnet
4. Kontaktaufnahme mit Router, nachdem auf Router-Advertisment-Nachrichten gewartet wurde oder es wird eine Router-Solicitation-Nachricht an alle Router gesendet um ein Advertisment zu verlangen
5. Router teilt Host mit wie er mit Autokonfiguration weiter machen soll

# Quellen


- Kurose, James F, and Keith W Ross. Computer Networking : A Top-down Approach. Boston, Mass., Pearson, 2017.
- Kozierok, Charles M. “Welcome to The TCP/IP Guide!”, www.tcpipguide.com. Accessed 6 May 2020.
- “IEEE 802 Numbers.” w*ww.Iana.org*, www.iana.org/assignments/ieee-802-numbers/ieee-802-numbers.xhtml. Accessed 6 May 2020.