

---

# IPv4 Grundrechnung

# IPv4-Adressen

---

- 4 Oktette (Byte) = 32 Bit
- 'dotted-decimal' Darstellung:  
192.168.10.1
- auch binär möglich:  
11000000 . 10101000 . 00001010 . 00000001

# Netzmaske (= Subnetzmaske)

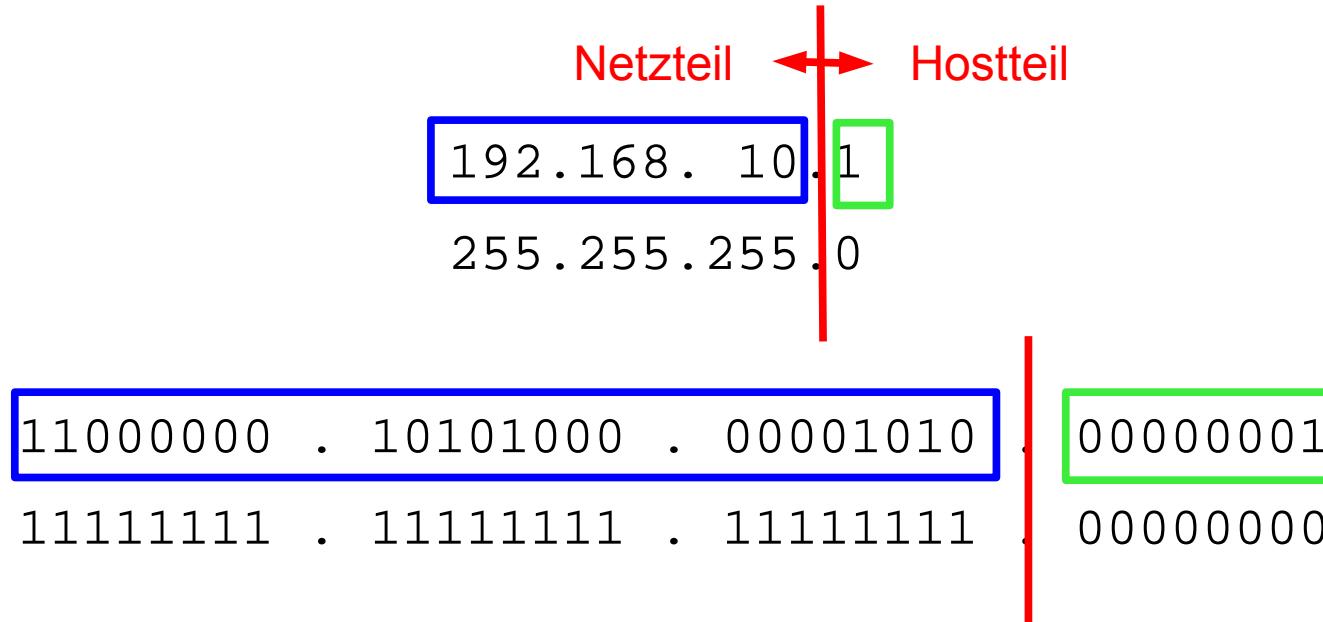
---

- in Ergänzung zu IP-Adresse
- teilt Adresse in Netzteil und Hostteil
- Darstellung 'dotted-decimal' (alt)  
255.255.255.252
- binär  
11111111 . 11111111 . 11111111 . 11111100
- Darstellung CIDR (neu), Anzahl der '1'  
/ 30
- immer linksbündige Folge von '1'

# Netzmaske

---

Beispiel:



# Anwendung der Netzmaske

---

- Beispiel: WvSS r305pc25: 10.16.105.25
- WvSS Netzmaske (derzeit): / 8 (255.0.0.0)
  - die ersten 8 Bit sind die Netzadresse: 10.0.0.0
- ping halde (10.16.1.251)
  - r305pc25 und halde sind im gleichen Netz 10.0.0.0
- ping google.de (z.B. 173.194.112.223)
  - google.de ist nicht im Netz 10.0.0.0 → Daten-packete an zuständigen Router (Gateway) senden

# Adressklassen

---

- 'classful' vs. 'classless' (CIDR, VLSM)
- veraltet !
- classful: Netzmaske wird nicht angegeben sondern je nach Klasse unterstellt

Klasse	binär	Bereich	Netzmaske	=	/
A	0...	0.0.0.0 - 127.255.255.255	255.0.0.0	=	/ 8
B	10...	128.0.0.0 - 191.255.255.255	255.255.0.0	=	/16
C	110...	192.0.0.0 - 223.255.255.255	255.255.255.0	=	/24
D	1110...	224.0.0.0 - 239.255.255.255	Multicast		
E	1111...	240.0.0.0 - 255.255.255.255	reserviert		

# Adresstypen

---

- Unicast-Adressen
  - an einen bestimmten Adressaten
- Multicast-Adressen
  - an eine Gruppe von Adressaten
  - Gruppenbildung: Abonnement durch Hosts
  - Daten werden auf Router und Switches kopiert
- Broadcast-Adressen
  - an 'Alle', problematisch!
  - Praxis: nur im lokalen (Sub-) Netz

# Besondere Adressen

---

- Private Adressen:
  - Klasse A            10.X.X.X
  - Klasse B            172.16.X.X - 172.31.X.X
  - Klasse C            192.168.X.X
- weitere reservierte Bereiche, z.B.
  - 169.254.X.X   Zeroconf (APIPA, Bonjour, ...)
  - 127.X.X.X      Loopback (localhost)

# Grundfrage

---

- Ist eine IP-Adresse in einem Netz enthalten?
- Beispiel:
  - 1.) Ist die Adresse 192.168.13.77 im Netz 192.168.13.64/26 enthalten?
  - 2.) Ist die Adresse 192.168.13.132 im Netz 192.168.13.64/26 enthalten?

# Drei Lösungsmethoden

---

- Für Theoretiker: Die Lehrbuchmethode
- Für Praktiker: Die Inkrement-Methode
- Für Denkfaule: Die Software-Methode (ipcalc)

# Die Lehrbuchmethode

---

- Idee: Gefragte IP-Adresse mit der Netzmaske bitweise UND verknüpfen und resultierende Netz-Adresse mit der gegebenen Netz-Adresse vergleichen
- entspricht realer Implementierung
- Schritte:
  - IP-Adresse und Netzmaske binär darstellen
  - bitweise UND Verknüpfung
  - gegebene Netz-Adresse binär darstellen
  - vergleichen

# Die Lehrbuchmethode

---

192.168.13.77	11000000.10101000.00001101.01001101
/26	11111111.11111111.11111111.11000000
UND	11000000.10101000.00001101.01000000
192.168.13.64	11000000.10101000.00001101.01000000

192.168.13.132	11000000.10101000.00001101.10000100
/26	11111111.11111111.11111111.11000000
UND	11000000.10101000.00001101.10000000
192.168.13.64	11000000.10101000.00001101.01000000

# Die Lehrbuchmethode

---

- Optimierung: nur relevantes Oktett betrachten
- /26 : 4. Oktett

192.168.13.77    192.168.13 .01001101

/26                255.255.255.11000000

UND                192.168.13 .01000000

192.168.13.64    192.168.13 .01000000

# Die Inkrement-Methode

---

- Inkrement = Größe des Netzes
- immer 2er-Potenz !
- Formel: Inkrement =  $2^{32\text{-Anzahl der Netzbits}}$
- Anzahl der Netzbits aus Netzmaske
- zugehörige Netz-Adresse ist das nächstkleinere Vielfache des Inkrements

# Die Inkrement-Methode

---

- /26 → Inkrement =  $2^{32-26} = 2^6 = 64$
- Vielfache: 0, 64, 128, ...
- Nächstkleineres Vielfaches suchen:

192.168.13.77 → 192.168.13.64 → im Netz

192.168.13.132 → 192.168.13.128 → nicht im Netz

# ipcalc

---

- Kommandozeile oder Web
- auf WvSS-PCs bereits installiert
- Lösung: ipcalc 192.168.13.132/26

Address: 192.168.13.132                    11000000.10101000.00001101.10 000100

Netmask: 255.255.255.192 = 26            11111111.11111111.11111111.11 000000

Wildcard: 0.0.0.63                        00000000.00000000.00000000.00 111111

=>

Network: 192.168.13.128/26            11000000.10101000.00001101.10 000000

Broadcast: 192.168.13.191            11000000.10101000.00001101.10 111111

HostMin: 192.168.13.129            11000000.10101000.00001101.10 000001

HostMax: 192.168.13.190            11000000.10101000.00001101.10 111110

Hosts/Net: 62

# Variationen des gleichen Themas

---

Beispiele:

- Host-, Netzwerk- oder Broadcast-Adresse?
- Wie viele Hosts kann ein Netz haben
- Adressbereich eines Netzes auflisten
- Sind die gegebenen Adressen im gleichen Netz?
- Netzplanung

# Host-, Netz- oder Broadcastadresse?

---

- Netzwerk-Adresse: Hostteil alle Bits '0'
  - immer gerade
- Broadcast-Adresse: Hostteil alle Bits '1'
  - immer ungerade
- Rest: Host-Adressen

# Host-, Netz- oder Broadcastadresse?

---

Beispiel: 172.18.13.131 / 30

Lösung binär: (letztes Oktett relevant)

IP-Adresse 131 = 1000 0011

Netzmaske /30 = 1111 1100

→ alle Bits '1': Broadcast-Adresse

Lösung Inkrement-Methode: Inkrement /30 → 4

Netz-Adressen in 4er Schritten:

- ...124, 128, 132, ...
- 131 = letzte Adresse vor nächster Netz-Adresse  
→ Broadcast-Adresse

# Anzahl der möglichen Hosts

---

- Anzahl ergibt sich (nur) aus der Netzmaske
- Inkrement = Netzgröße
- 2 reservierte Adressen:
  - Netz-Adresse: Hostbits '0', kleinste Adresse im Netz
  - Broadcast-Adresse: Hostbits '1', größte Adresse
- Formel: Anzahl Hosts =  $2^{32 - \text{Anzahl Netzbits}} - 2$
- oft eine Host-Adresse für Gateway erforderlich!
- Beispiel /26:  $2^{32-26} - 2 = 2^6 - 2 = 64 - 2 = 62$

# Adressbereich eines Netzes

---

- gesucht: kleinste und größte Host-Adresse (Host-Range) sowie Broadcast-Adresse
- Beispiel: 172.18.13.32 / 28
- kleinste Host-Adresse = Netz-Adresse + 1
- Inkrement 16 → nächste Netz-Adresse 48
- Broadcast-Adresse = nächste Netz-Adresse - 1
- größte Host-Adresse = Broadcast-Adresse - 1
- Ergebnis: Host-Range 172.18.13.33 bis 172.18.13.46, Broadcast 172.18.13.47

# Im gleichen Netz ?

---

- gegeben: mehrere (Host-) IP-Adressen mit einheitlicher Netzmaske
- Frage: Welche Hosts können direkt kommunizieren ?
- Lösung: Inkrement aus Netzmaske berechnen und Host-Adressen in Raster einsortieren
- Beispiel: 10.33.44.11 / 27, 10.33.44.22 / 27,  
10.33.44.33 / 27, 10.33.44.44 / 27,  
10.33.44.55 / 27, 10.33.44.66 / 27

# Im gleichen Netz ?

---

- Lösung: Inkrement = 32
- Raster (Netze):

0 - 31      10.33.44.11, 10.33.44.22

32 - 63      10.33.44.33, 10.33.44.44, 10.33.44.55

64 - 95      10.33.44.66

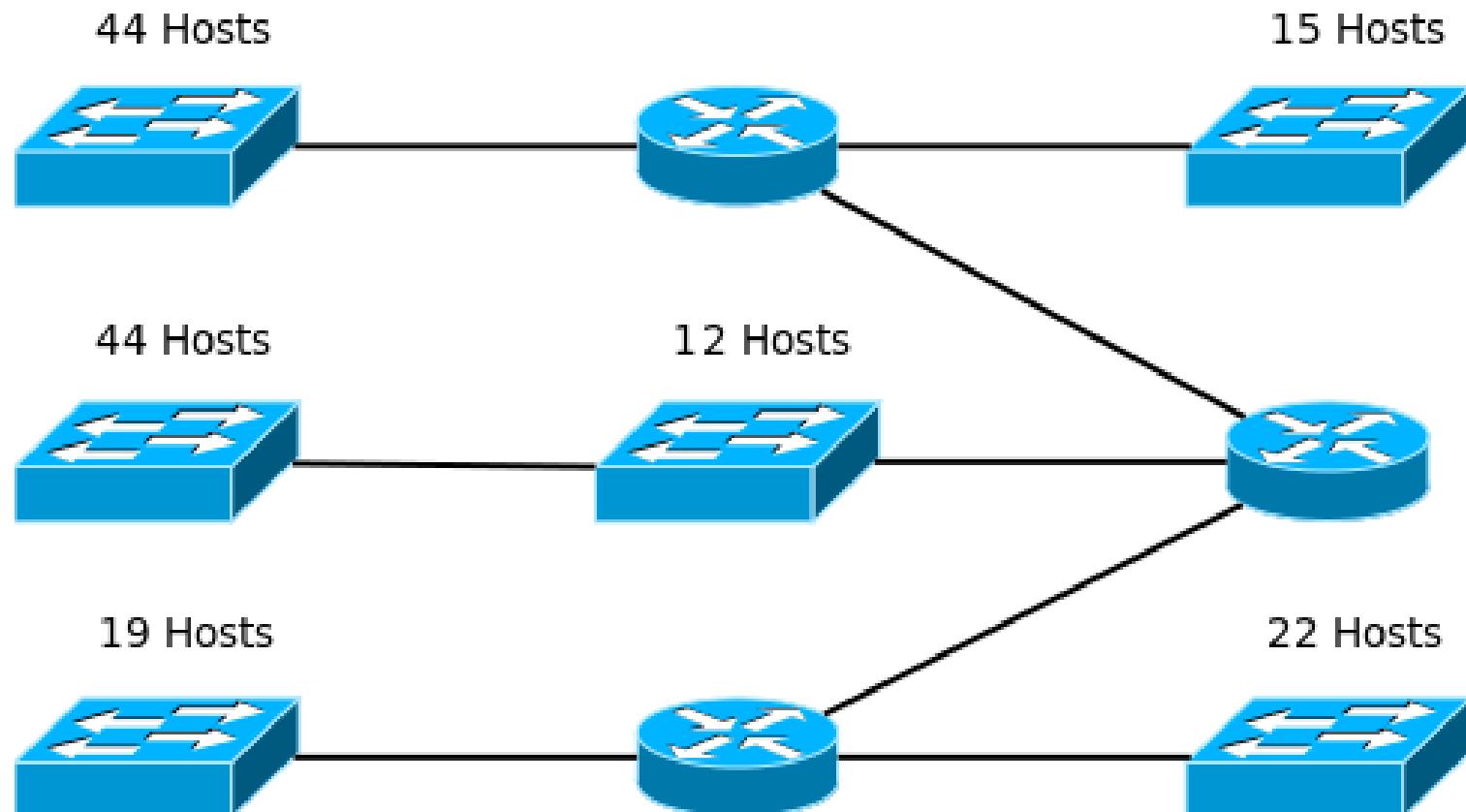
# VLSM-Netzplanung ('Subnetting')

---

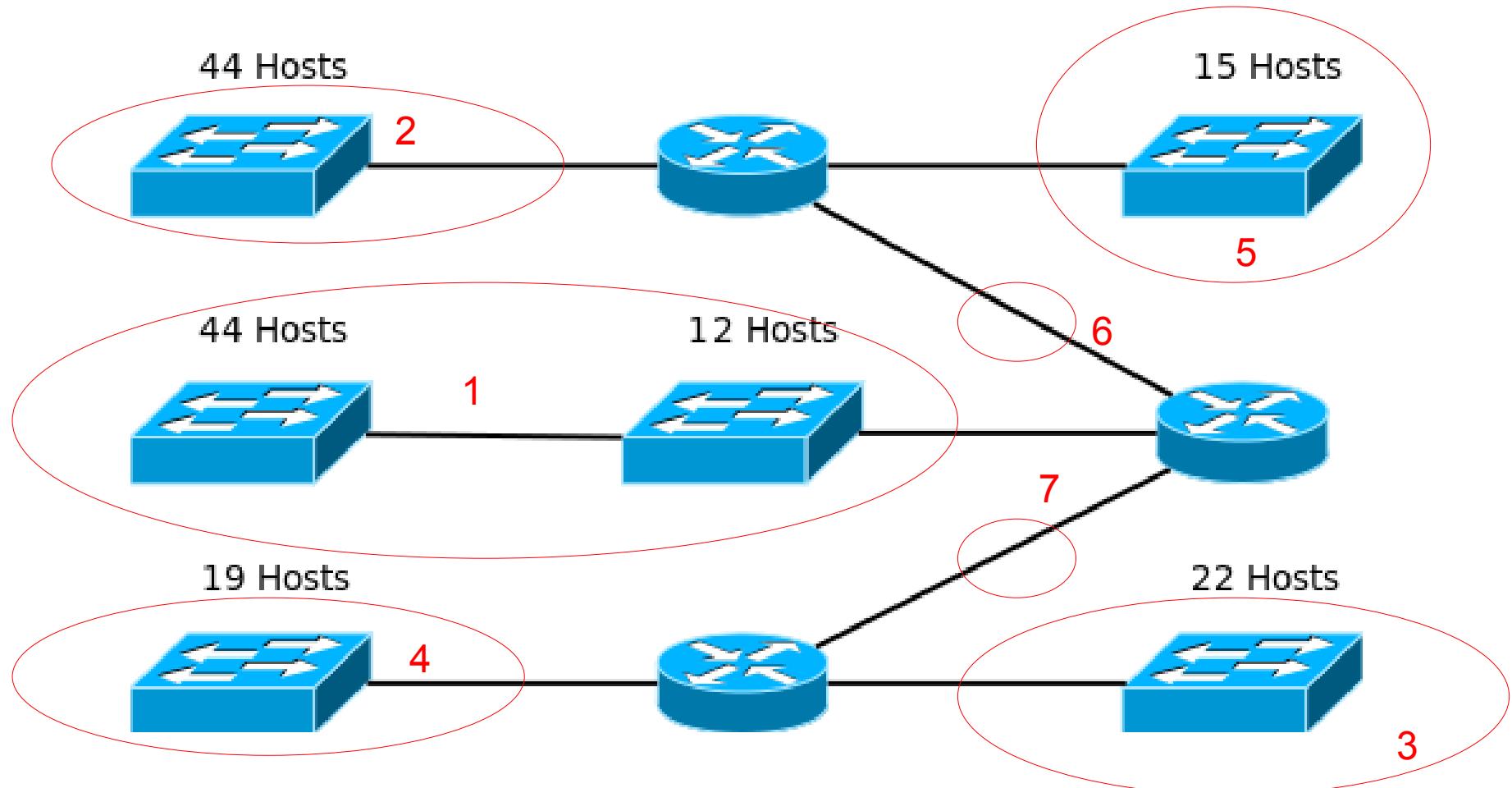
- VLSM: Variable Length Subnet Mask
  - Subnetze nur so groß wie notwendig
  - Subnetze haben unterschiedliche Netzmasken
  - Sinn: minimale Adressverschwendung
- Netzgröße immer 2er-Potenz !
- (Sub-) Netz kann nur an einem Vielfachen seines Inkrements anfangen !
  - große Netze zuerst vergeben
- Ergebnis: Tabelle, Spalten: Netzname, Netz-Adresse mit Maske, Host-Range, Broadcast

# VLSM-Netzplanung

Adressraum: 192.168.1.0/16



# VLSM-Netzplanung



# VLSM-Netzplanung

---

- 7 Subnetze, der Größe nach durchnummieriert
- Größen (inkl. Router und 2 reservierten Adressen!) auf 2er-Potenz aufrunden = Inkrement
- erforderliche Netzmaske jeweils aus Inkrement bestimmen

1	44 + 12 Hosts, 1 Router = 57	57 + 2 = 59	aufrunden:	64	/26
2	44 Hosts, 1 Router = 45	45 + 2 = 47	aufrunden:	64	/26
3	22 Hosts, 1 Router = 23	22 + 2 = 24	aufrunden:	32	/27
4	19 Hosts, 1 Router = 20	20 + 2 = 22	aufrunden:	32	/27
5	15 Hosts, 1 Router = 16	16 + 2 = 18	aufrunden:	32	/27
6	2 Router	2 + 2 = 4	aufrunden:	4	/30
7	2 Router	2 + 2 = 4	aufrunden:	4	/30

# VLSM-Netzplanung

---

- Adressen aus gegebenem Bereich (192.168.1.0/16) gemäß Inkrement auffüllen

<b>Netz</b>		<b>Netz-Adresse</b>	<b>Host-Range</b>	<b>Broadcast</b>
1	64	192.168.1.0/26	192.168.1.1 - 192.168.1.62	192.168.1.63
2	64	192.168.1.64/26	192.168.1.65 - 192.168.1.126	192.168.1.127
3	32	192.168.1.128/27	192.168.1.129 - 192.168.1.158	192.168.1.159
4	32	192.168.1.160/27	192.168.1.161 - 192.168.1.190	192.168.1.191
5	32	192.168.1.192/27	192.168.1.193 - 192.168.1.222	192.168.1.223
6	4	192.168.1.224/30	192.168.1.225 - 192.168.1.226	192.168.1.227
7	4	192.168.1.228/30	192.168.1.229 - 192.168.1.230	192.168.1.231

```
ipcalc 192.168.1.0/16 -s 47 45 23 20 16 2 2
```