

# Internet Protocol Version 6

Vortrag von Ingo Blechschmidt

am 2. Februar 2005

# Gliederung

- Was fehlt IPv4?
- Was ist IPv6?
- IPv6-Adressen und -Adressraum
- Statuslose Autokonfiguration
- Neighbour Discovery
- IPv6 Mobility
- IPv6 im DNS
- Tunnelbroker

# Was fehlt IPv4?

- Zu wenig Adressen (theoret.  $2^{32} \approx 4,3 \cdot 10^9$ )  
⇒ „Fix“: Private Adressen, NAT
- Keine Verschlüsselung und Signierung  
⇒ Fix: IPsec
- Keine automatische Konfiguration  
⇒ Suboptimaler Fix: DHCP
- Zu große Routing-Tabellen in den „großen Routern“

# Was ist IPv6?

- Erweiterung des Adressraums von 32 Bit auf 128 Bit (theoret.  $2^{128} \approx 3,4 \cdot 10^{38}$ )
- Keine privaten Adressen in ihrer jetzigen Form mehr
- Integration von IPsec in das Protokoll
- Zustandslose Konfiguration, Router Discovery

# IPv6-Adressen

- Beispiel:

2001:08e0:abcd:014d:0000:0000:0000:0001

# IPv6-Adressen

- Beispiel:

2001:8e0:abcd:14d:0:0:0:1

# IPv6-Adressen

- Beispiel:  
2001:8e0:abcd:14d::1
- ::  $\hat{=}$  ::0  $\hat{=}$  0.0.0.0
- ::1  $\hat{=}$  127.0.0.1

# IPv6-Adressen

- Beispiel:

2001:8e0:abcd:14d::1

- $:: \hat{=} ::0 \hat{=} 0.0.0.0$

- $::1 \hat{=} 127.0.0.1$

- Netze nach CIDR-Notation:

2000::/3  $\hat{=} \mathbf{001}0000000000000000::/3$  bis

3fff:.../3  $\hat{=} \mathbf{001}1111111111111111:.../3$

# IPv6-Adressraum

Netz	Bereich	Zweck
::/4	:: -0fff:...	Verschiedens
2000::/8	2000::-3fff:...	Global Unicast
fc00::/7	fc00::-fdff:...	Unique Local Unicast
fe80::/10	fe80::-febf:...	Link-Local Unicast
ff00::/8	ff00::-ffff:...	Multicast

# Autokonfiguration

- Zuweisung einer Link-lokalen Adresse:  
00:e0:7d:e5:0b:ff (MAC-Adresse)

# Autokonfiguration

- Zuweisung einer Link-lokalen Adresse:  
00:e0:7d:**ff:fe**:e5:0b:ff (EUI-64)

# Autokonfiguration

- Zuweisung einer Link-lokalen Adresse:  
02:e0:7d:ff:fe:e5:0b:ff

# Autokonfiguration

- Zuweisung einer Link-lokalen Adresse:  
02e0:7dff:fee5:0bff

# Autokonfiguration

- Zuweisung einer Link-lokalen Adresse:  
**fe80::2e0:7dff:fee5:bff**

# Autokonfiguration

- Zuweisung einer Link-lokalen Adresse:  
`fe80::2e0:7dff:fee5:bff`
- „Gibt's diese Adresse schon?“

# Autokonfiguration

- Zuweisung einer Link-lokalen Adresse:  
`fe80::2e0:7dff:fee5:bff`
- „Gibt's diese Adresse schon?“
- Senden einer Router Solicitation  $\Rightarrow$   
Router Advertisement:  
„Das Präfix unseres Netzes ist  
`2001:8e0:abcd:14d::/64.`“

# Autokonfiguration

- Zuweisung einer Link-lokalen Adresse:  
**fe80::2e0:7dff:fee5:bff**
- „Gibt's diese Adresse schon?“
- Senden einer Router Solicitation ⇒  
Router Advertisement:  
„Das Präfix unseres Netzes ist  
2001:8e0:abcd:14d::/64.“
- Endgültige Adresse:  
**2001:8e0:abcd:14d:2e0:7dff:fee5:bff**
- DHCPv6 wenn nötig/verlangt

# Autokonfiguration – Praxis

- Voraussetzung: Konfigurierter Router
- Dann:  
# modprobe ipv6

# Autokonfiguration – Praxis

- Voraussetzung: Konfigurierter Router
- Dann:  
  # modprobe ipv6
- (Fertig)

# Autokonfiguration – radvd

```
interface eth0 {  
    AdvSendAdvert on;  
    AdvHomeAgentFlag off;  
    AdvLinkMTU 1280;  
    prefix 2001:8e0:abcd:14d::/64 {  
        AdvOnLink on;  
        AdvAutonomous on;  
        AdvRouterAddr on;  
    };  
};
```

# Neighbour Discovery

- Senden eines Pakets im (z.B.) Ethernet:  
Voraussetzung: Kennen der Ziel-MAC-Adresse
- IPv4 mit ARP:
  - Beispiel:  $1.2.3.4[a:b:c:d:e:f] \rightarrow$   
 $4.3.2.1[f:e:d:c:b:a]$

# Neighbour Discovery

- Senden eines Pakets im (z.B.) Ethernet:  
Voraussetzung: Kennen der Ziel-MAC-Adresse
- IPv4 mit ARP:
  - Beispiel:  $1.2.3.4[a:b:c:d:e:f] \rightarrow 4.3.2.1[f:e:d:c:b:a]$
  - Broadcast eines ARP-Requests:  
Quelle:  $1.2.3.4[a:b:c:d:e:f]$   
Ziel:  $4.3.2.1[\text{unbekannt}]$

# Neighbour Discovery

- Senden eines Pakets im (z.B.) Ethernet:  
Voraussetzung: Kennen der Ziel-MAC-Adresse
- IPv4 mit ARP:
  - Beispiel:  $1.2.3.4[a:b:c:d:e:f] \rightarrow 4.3.2.1[f:e:d:c:b:a]$
  - Broadcast eines ARP-Requests
  - Erhalt eines ARP-Replys:  
Quelle:  $4.3.2.1[f:e:d:c:b:a]$   
Ziel:  $1.2.3.4[a:b:c:d:e:f]$
  - Problem: ARP-Spoofing

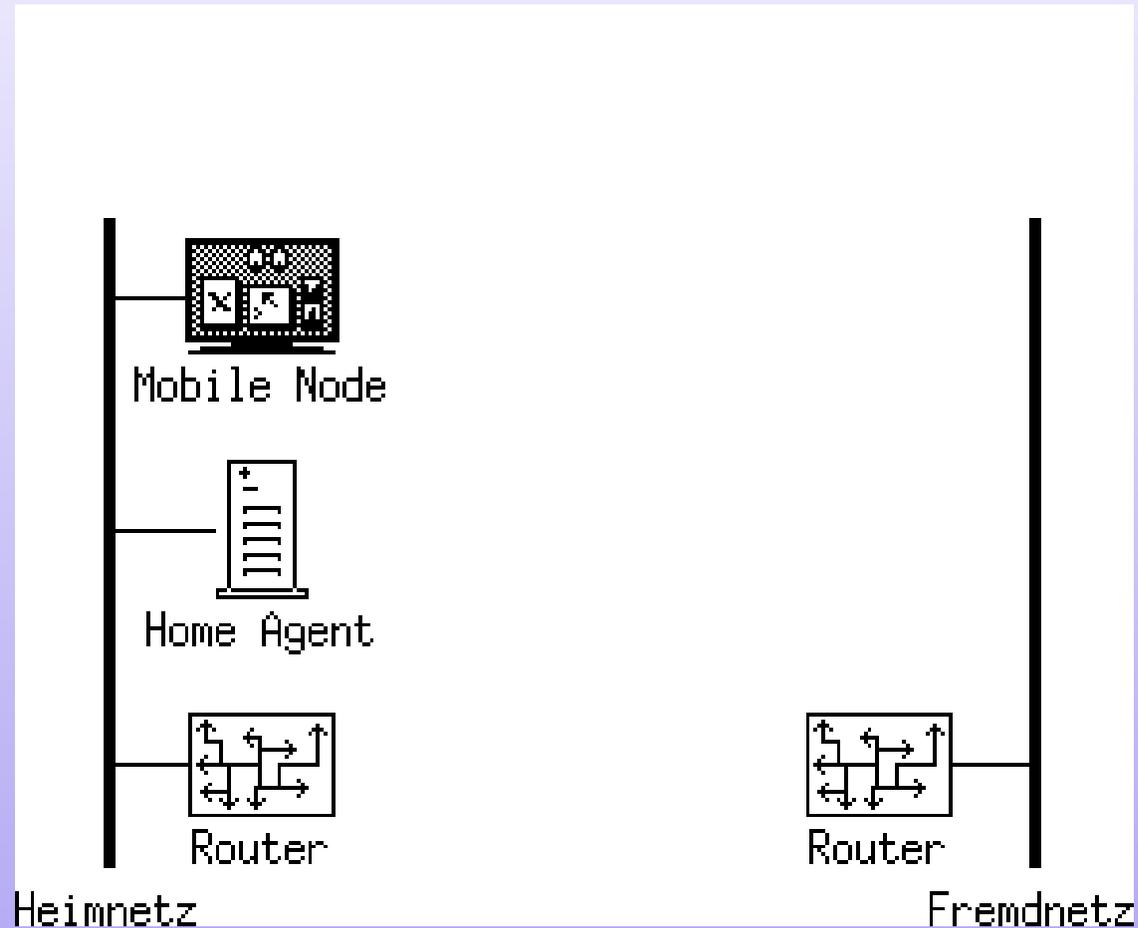
# Neighbour Discovery

- Senden eines Pakets im (z.B.) Ethernet:  
Voraussetzung: Kennen der Ziel-MAC-Adresse
- IPv4 mit ARP
- IPv6 mit ICMPv6:
  - Multicasting einer Neighbour Solicitation an einen „Hash der IP-Adresse“
  - Erhalt eines Neighbour Advertisements
  - Nutzung von IPv6  $\Rightarrow$   
Schutz mit IPsec problemlos möglich  $\Rightarrow$   
Kein „ARP“-Spoofing mehr!

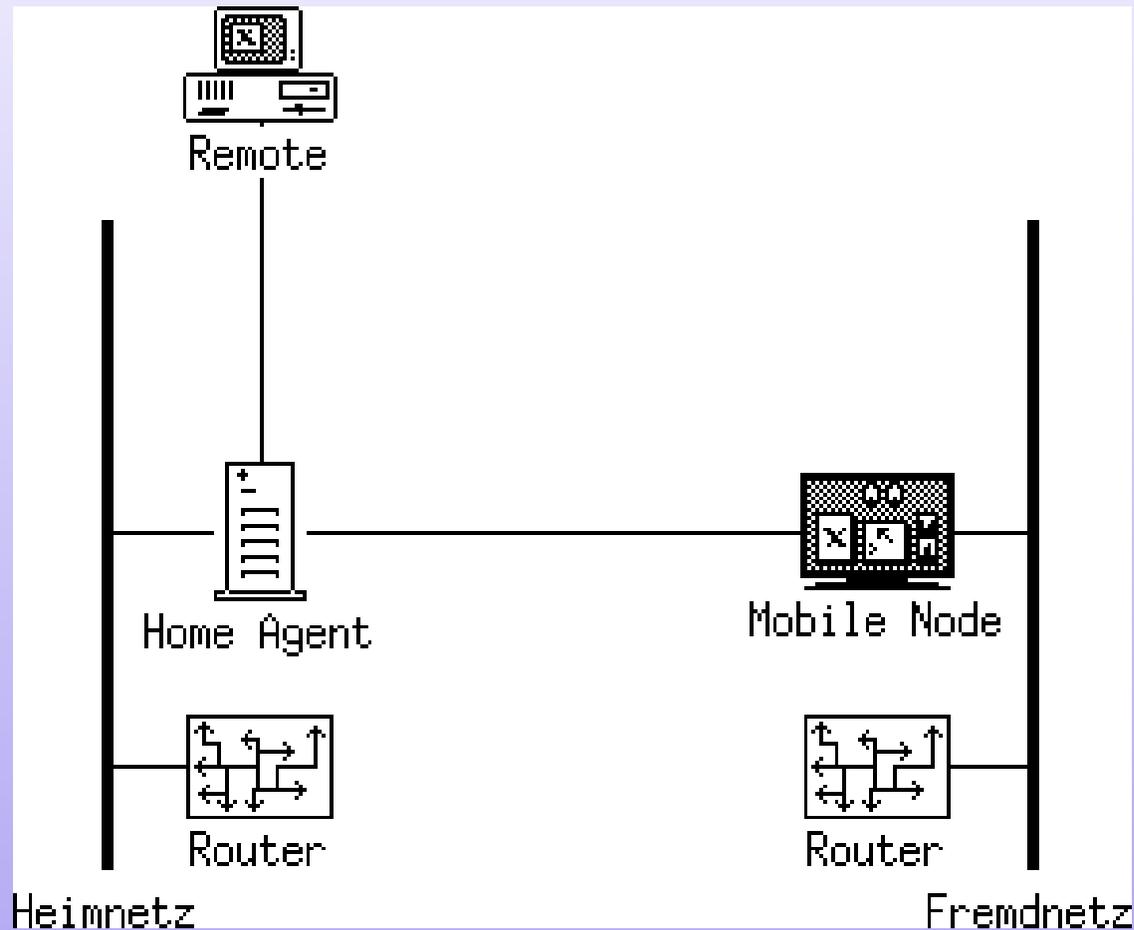
# Multicast

- Eine IP-Adresse für mehrere Hosts
- Beitritt einer Multicast-Gruppe  $\Rightarrow$  Erhalt aller Nachrichten an die Gruppe
- $ff02::1$ : Alle Nodes eines Links
- $ff02::2$ : Alle Router eines Links
- $ff02::1:ffxx:xxxx$ :  
Multicast-IP eines jeden Hosts (Auffüllen mit den niederwertigsten 24 Bits der Ziel-IP)
- Grundlage für Neighbour/Router Discovery

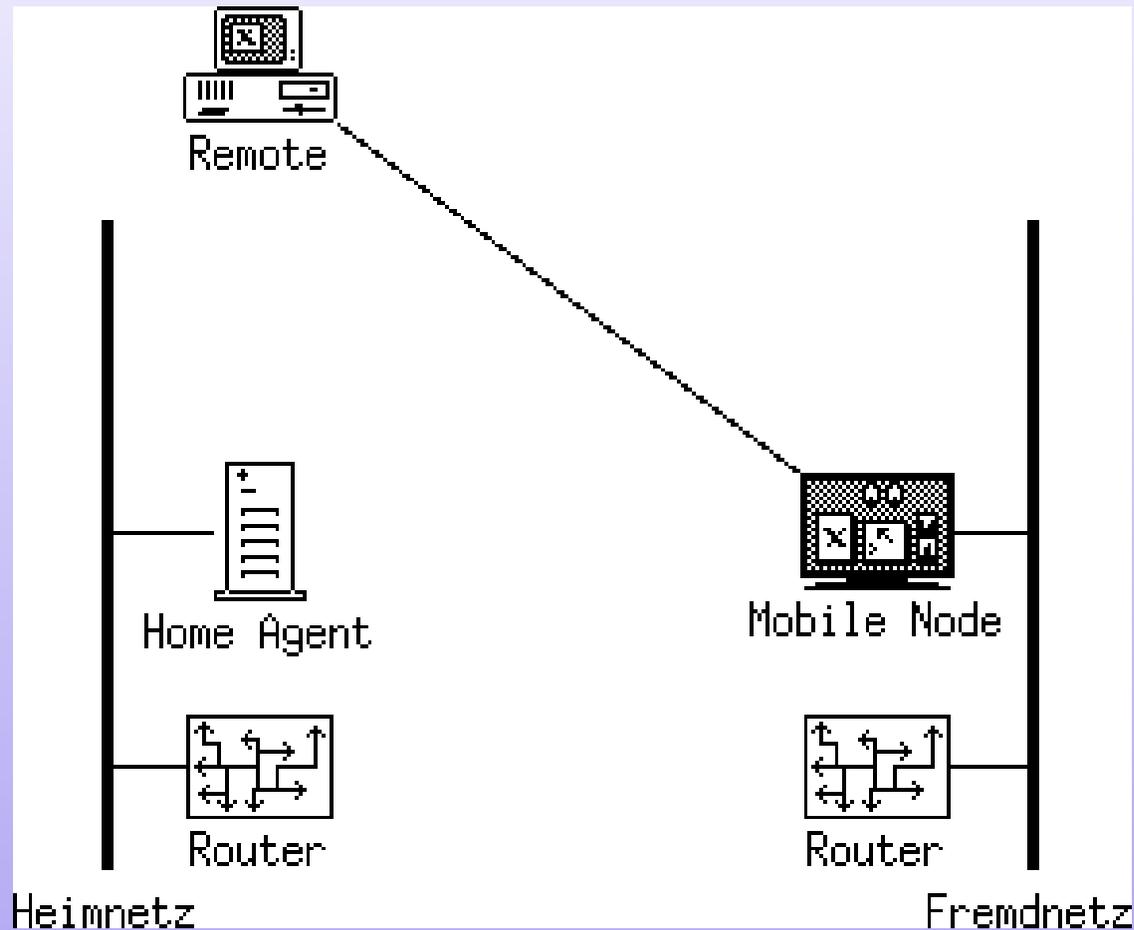
# IPv6 Mobility



# IPv6 Mobility



# IPv6 Mobility



# IPv6 im DNS

- Forward-Lookups durch AAAA-Records:

```
$ dig www.ipv6-net.de. ANY
```

```
www.ipv6-net.de. 86400 IN A 62.93.217.177
```

```
www.ipv6-net.de. 86400 IN AAAA \  
    2001:618:1401::4
```

# IPv6 im DNS

- Forward-Lookups durch AAAA-Records
- Reverse-Lookups durch PTR-Records unterhalb `ip6.arpa.`:

```
$ dig -x 2001:8e0:abcd:14d:2e0:7dff:fee5:bff
;; QUESTION SECTION:
;f.f.b.0.5...1.0.0.2.ip6.arpa. IN PTR
;; ANSWER SECTION:
f.f.b.0.5...1.0.0.2.ip6.arpa. 120 IN PTR \
    thestars.iblech.tb.as8758.net.
```

# Tunnelbroker

- Bisher wenig Angebot von nativem IPv6
- Aber: IPv6 über Tunnelbroker
- Verbindung zum Tunnelbroker durch ganz normales IPv4
- Verpacken der IPv6-Pakete in IPv4-Paketen
- Je nach Tunnelbroker Erhalt einer IPv6-Adresse oder eines ganzes Subnetzes (/64 oder /48)

# Tunnelbroker – as8758.net

- Ein-minütige Registrierung auf as8758.net
- Skript zum Tunnelaufbau ausführen
- `$ ping6 www.ipv6-net.de`
- Für's LAN: /64er-Subnetz beantragen,  
radvd konfigurieren
- (Fertig)

# Ausblick

- Ablösung von IPv4 durch IPv6
- Zunehmende Verbreitung von VoIP
- Telefonnummern im DNS (ENUM)
- Handies über IP
- Telefone über IP