IPv4-Exit Strategy for Mobile Networks

Gabriel Bertram, Detecon International GmbH, Cologne
Andreas Grebe, Cologne University of Applied Sciences, Computer Networks Research Group
Holger Metschulat, Deutsche Telekom Technik GmbH, Darmstadt

VDE Tagung Mobilkommunikation 2014, Osnabruck
22.05.2014

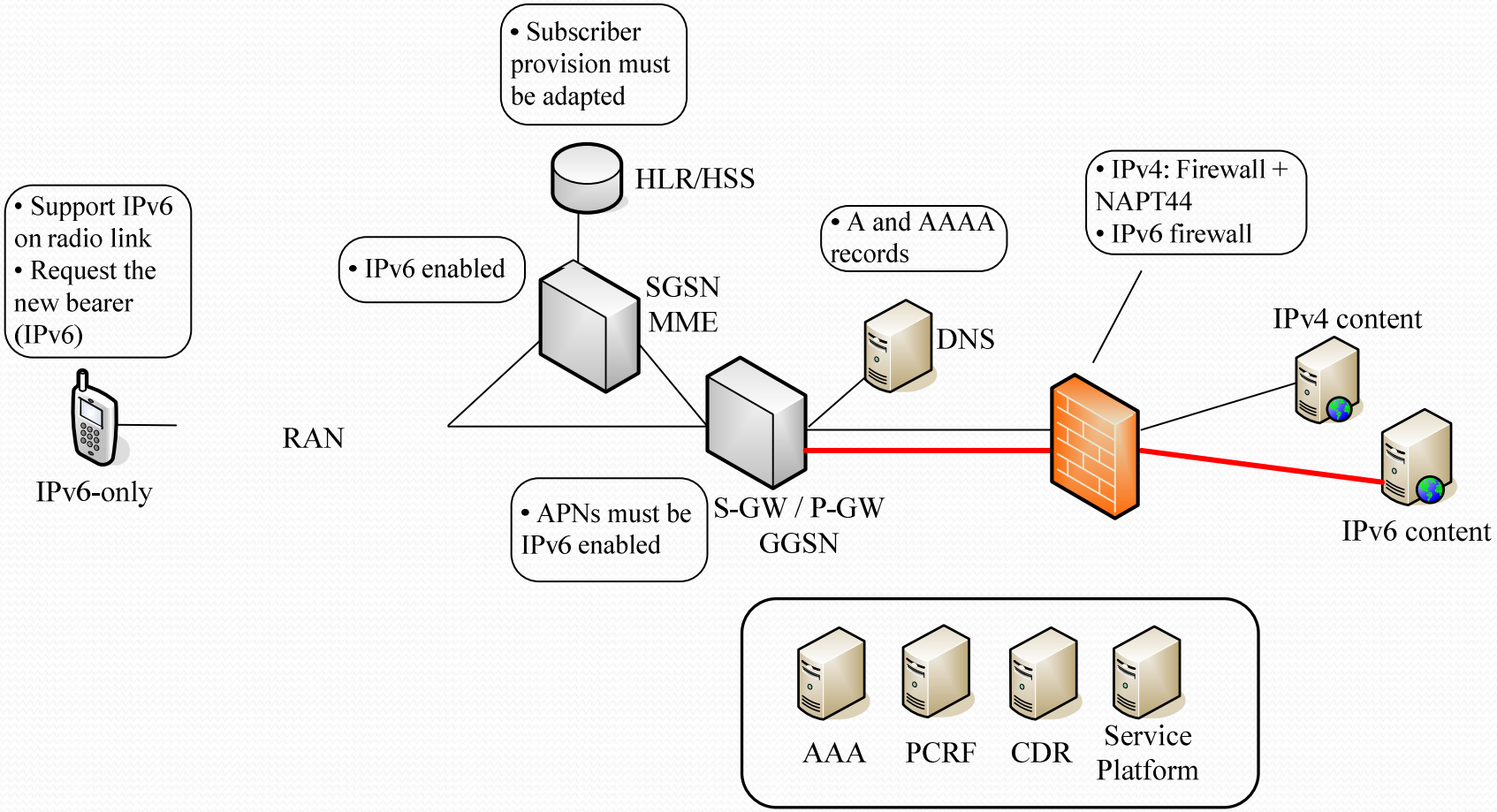
Table of content

- Introduction
- IPv6 in 3GPP networks
- IPv4-Exit techniques
- Conclusion

Introduction

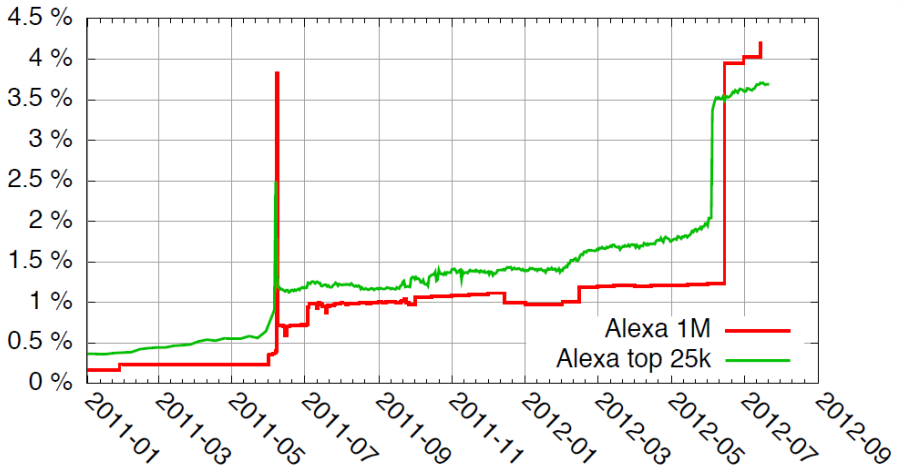
- Increasing demand of IP connectivity
- Limited IPv4 address space (public and private)
  - Expected demand for IP connectivity outpaces the IPv4 address capabilities
  - Delayed by private addressing, stateful NAPT44, virtual address ranges
- IPv6-only addressing to encounter the IPv4 address limitations
  - Dual-Stack does not encounter the private IPv4 address limitations
- Coexistence phase of IPv4 and IPv6
  - Ongoing IPv6 deployment but most services are IPv4-only today
- Incompatibility of IPv4 and IPv6

IPv6 in 3GPP networks



IPv6 distribution

- Most TOP websites enabled IPv6 permanently on World IPv6 day 2012
  - IPv6 enabled: google, facebook, youtube, wikipedia, yahoo…
  - Still IPv4-only: ebay, amazon, spiegel, bild…



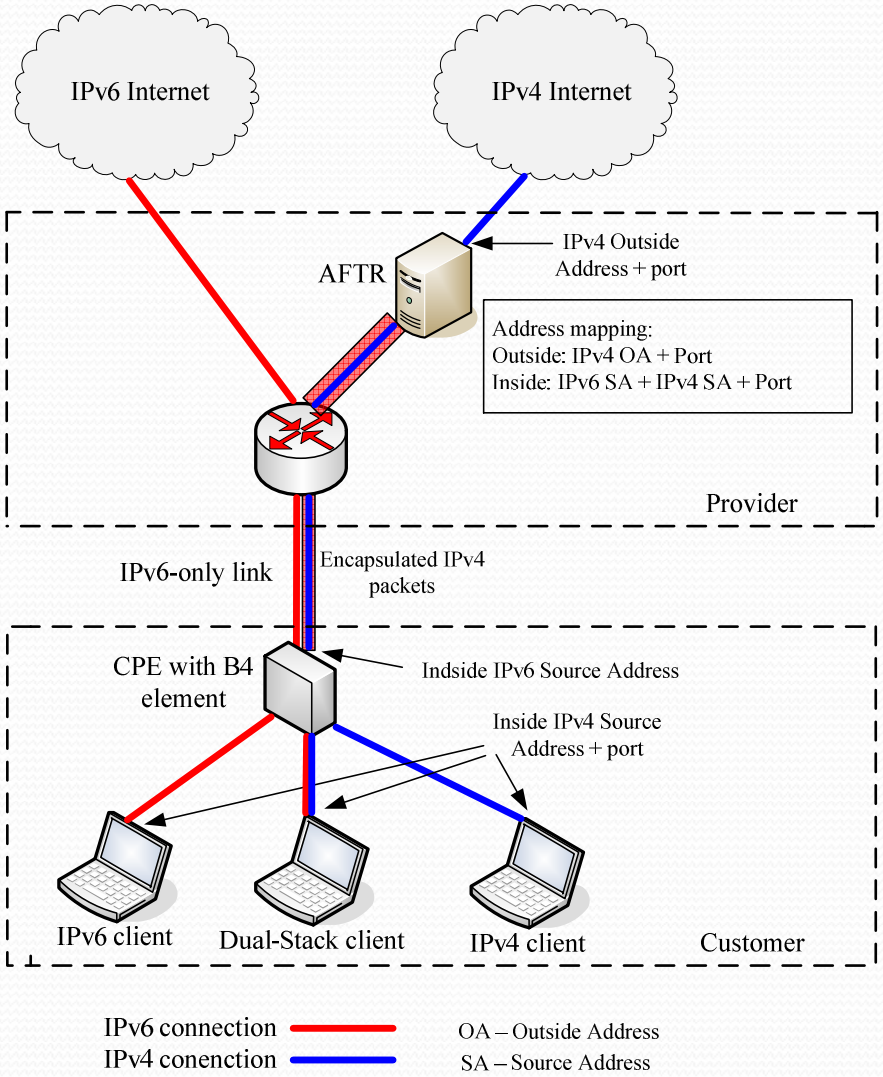
→ Compatibility mechanism required to provide accessibility to IPv4-only services by IPv6-only clients

IPv4-Exit techniques

- Different approaches to enable compatibility between IPv4 and IPv6
- Tunneling
  - Encapsulation and transport of IP packets as payload of other IP packets
  - Here: Encapsulation of IPv4 packets in IPv6 packets for transport using a native IPv6 infrastructure
  - Dual-Stack Lite: known from fixed environments
- Translation
  - Address and header conversion between the address families
    - NAT64 + DNS64
    - 464XLAT

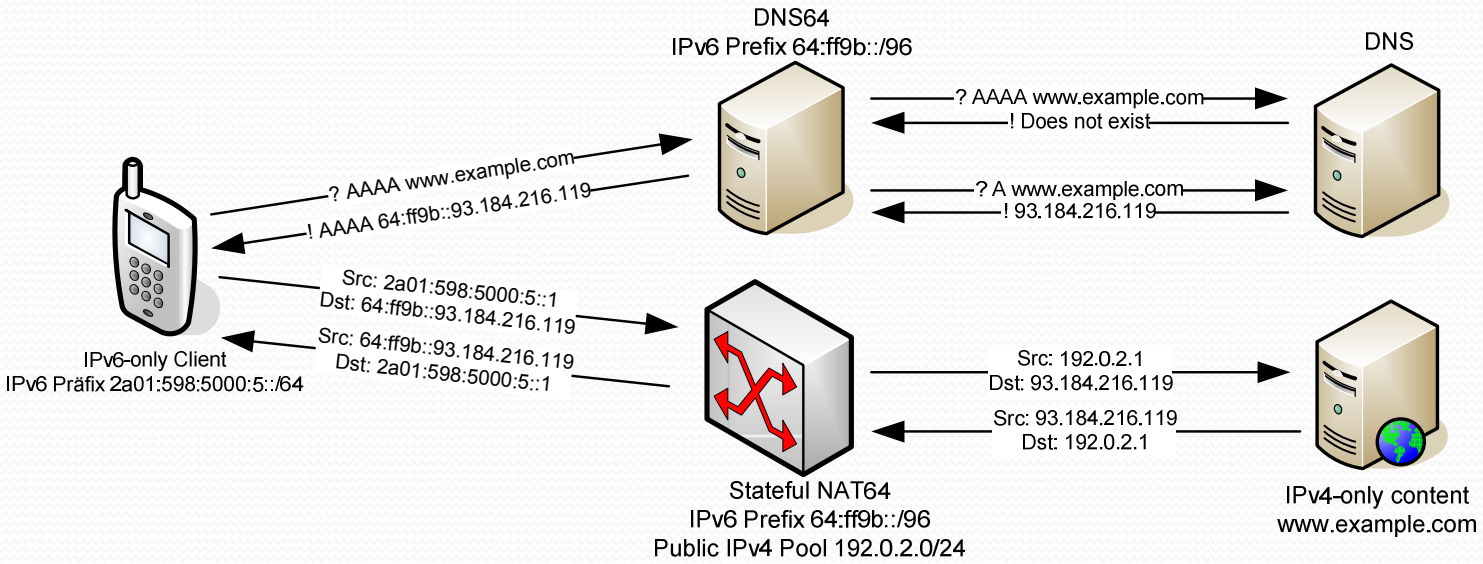
Gateway Initiated Dual-Stack Lite

- Combination of tunneling and NAPT44
- Sharing private IPv4 addresses among customers
- IPv6 address is added as additional element to NAPT44 mapping
- Not supported in current and medium term implementations

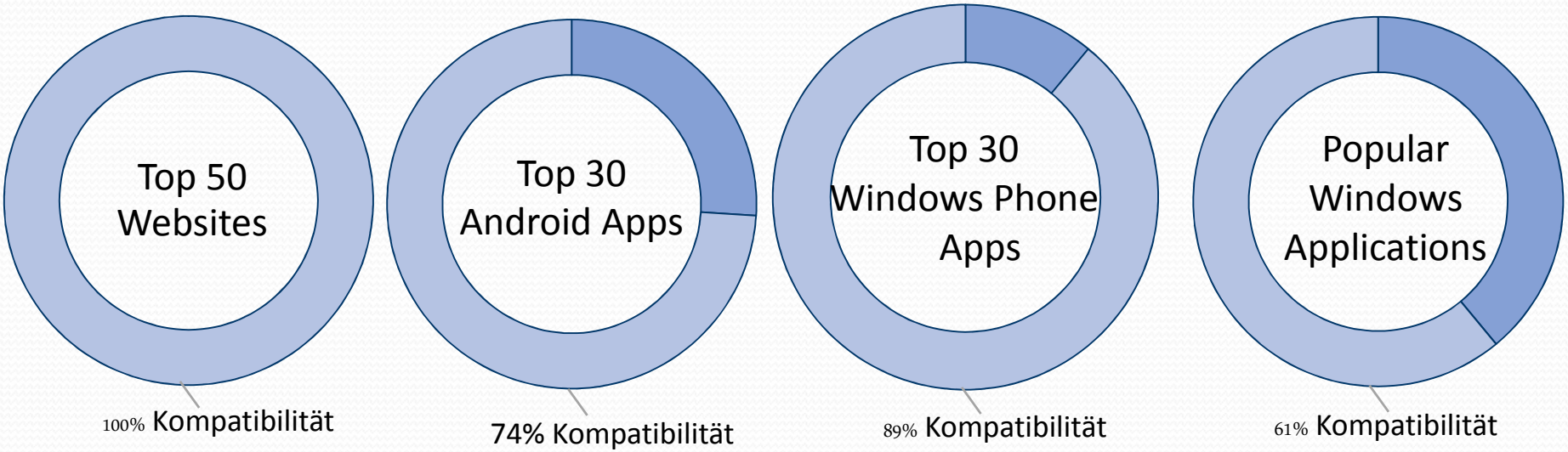


NAT64 + DNS64

- Network-based compatibility mechanism
  - DNS64 (RFC6147)
    - Synthetization of AAAA RR based on A RR
  - Stateful NAT64 (RFC6146)
    - Protocoltranslation
- IPv6 address generation required
  - Broken in case of IPv4-only Applications, APIs and IPv4 literals



NAT64 + DNS64 test results



- Limited applicability due to systematic limitation of address synthesis
  - Referencing IPv4 specific networking APIs , signaling IPv4 literals, no use of FQDN

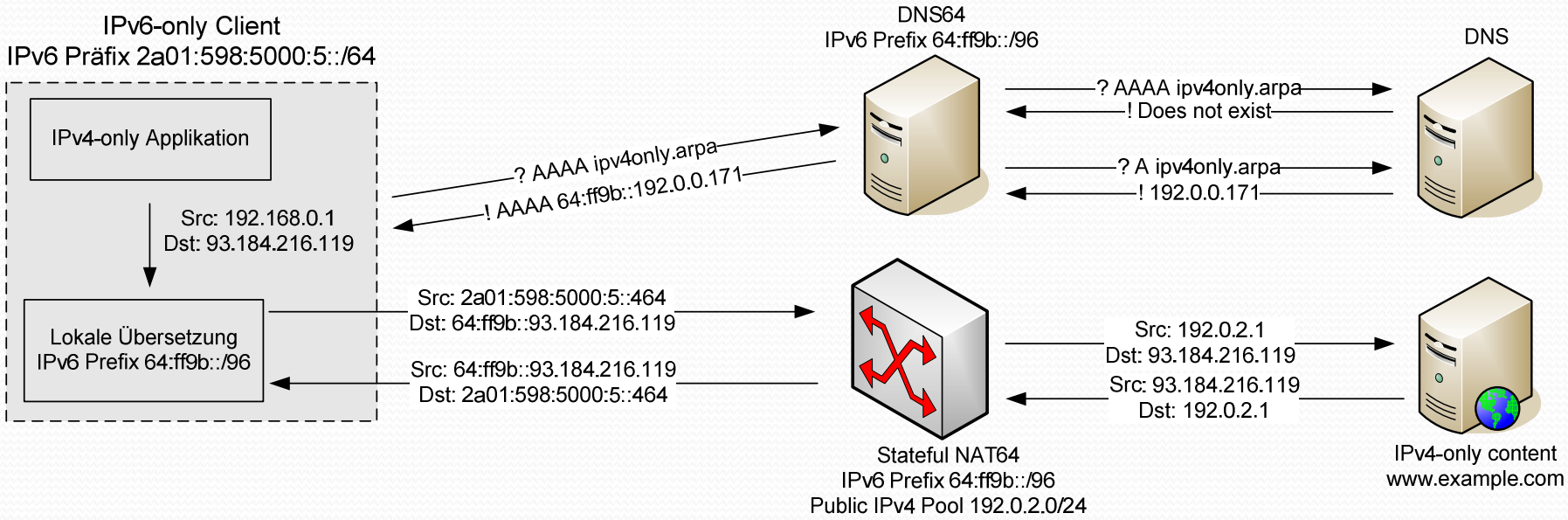
➡ DNS64/NAT64 is good but not good enough to replace IPv4

464XLAT

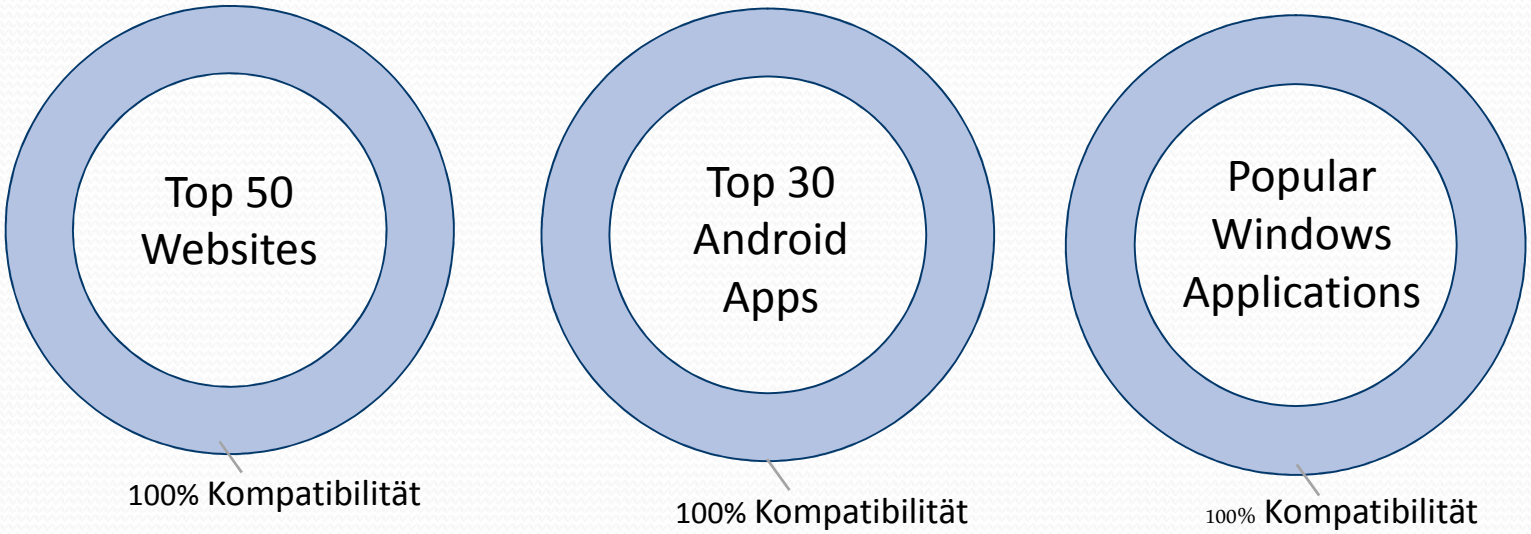
- Stateful NAT64 (RFC6146)
- DNS64 (RFC6147)

} 464XLAT is based on NAT64 + DNS64

- Stateless NAT64 (RFC6145)
  - Performing a client-side translation
- Prefix64 discovery (RFC7050)
  - Performs detection if NAT64/DNS64 is available in the network.



464XLAT test results



- Service parity with IPv4/Dual-Stack operation

Comparative assessment

	Dual-Stack	DS-Lite	GI-DS Lite	Stateful NAT64	464XLAT
Functionality	good	good	good	**limited**	**good**
Technical resources	high	**high**	**high**	**limited**	**limited**
Operational costs	high	**high**	**high**	**limited**	**limited**
Overhead	medium (signaling)	**high (access/core)**	**medium (core)**	**none**	**none**
Planning costs	high	**high**	**high**	**limited**	**limited**
Engineering costs	limited	**high**	**high**	**limited**	**limited**
Applicable in mobile networks	yes	**hard**	yes	yes	**medium**
IPv6-only access	**no**	yes	yes	yes	yes
Saves IPv4 addresses	**no (priv.)** yes (pub.)	yes (priv. and pub.)	yes (priv. and pub.)	yes (priv. and pub.)	yes (priv. and pub.)

Recommendation

- Short-term architecture
  - Dual-Stack for production network
  - Parallel testing of 464XLAT
  - Distribution of IPv6 and 464XLAT capable devices
- Middle-term architecture
  - IPv6-only step by step for well-tested devices
  - Dual-Stack for legacy devices
  - Distribution of IPv6 and 464XLAT capable devices
- Long-term architecture
  - IPv6-only access without any compatibility mechanism
  - 464XLAT just in case

Current problem areas

- IPv4-Exit activities are coupled to capabilities of end user devices
  - Limited availability of 464Xlat
- Local translation not fully compliant to the standard yet
  - Support of fragmented packets, ICMP messages…
- Systematic limitations
  - Restricted support for IPv4 Options, IPv6 EH, Layer 4 protocols
- Ongoing discussions/standardization activities on tethering (Prefix Delegation, 64share…)
- NAT64 inherits the well known limitations of NAT
  - ALGs, NAT-T…

Conclusion

- IPv4 does not meet the business need any more
- IPv6-only access network encounters the IPv4 address limitations
- The theoretical and practical assessments reveal that IPv6 in 3GPP networks can be realized as a smooth and careful process
  - Protocol translation as feasible compatibility mechanism provide compatibility between IPv4 and IPv6
- IPv6 deployment will be two-folded in the next years

Thanks for your attention

Backup slides

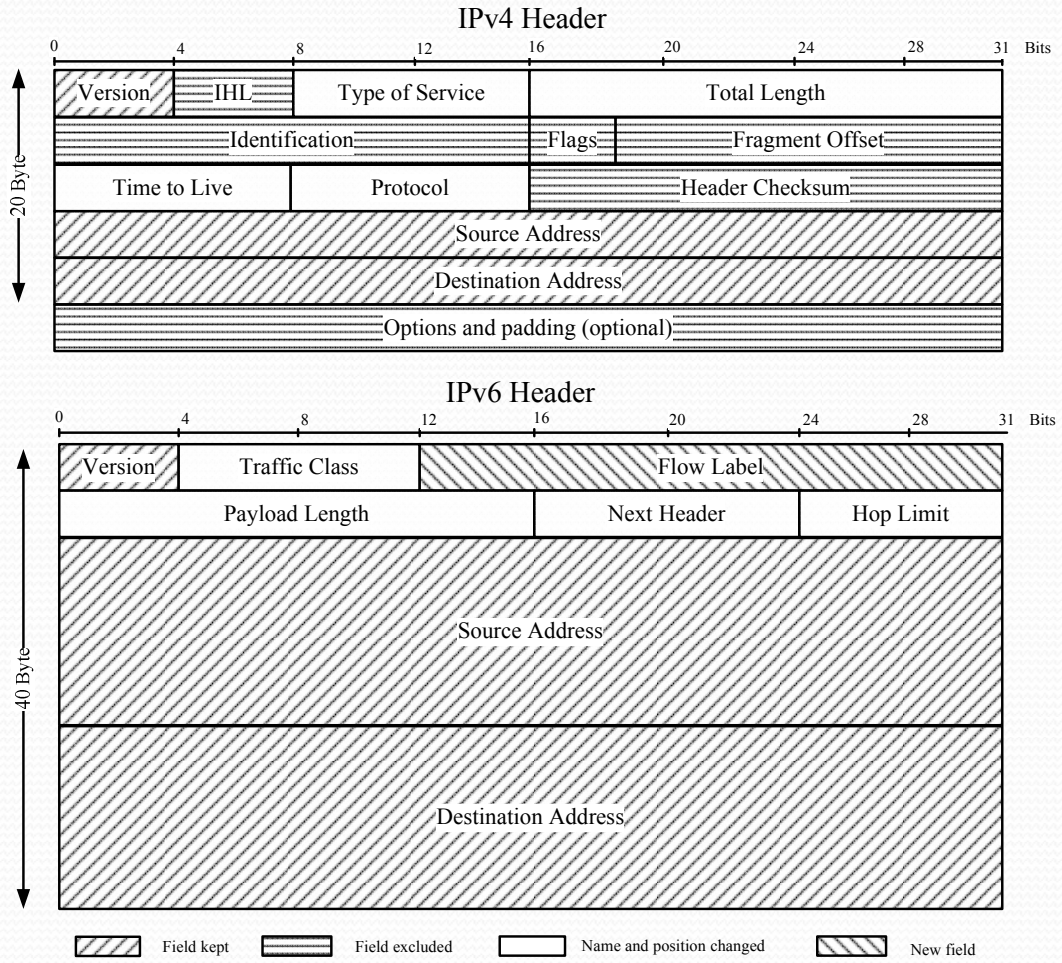
Drivers and goals

- IPv6 deployment is network driven (no marketing intentions)
  - Save private IPv4 addresses and virtualization instances
  - Save public IPv4 addresses and NAPT44 sessions
  - Dual-Stack networks require twice the engineering, planning operational effort than single-stacked network
- 100% compatibility with IPv4-only services and applications must be ensured
  - Maintain user experience, broken services will not be accepted

IPv6 in contrast to IPv4

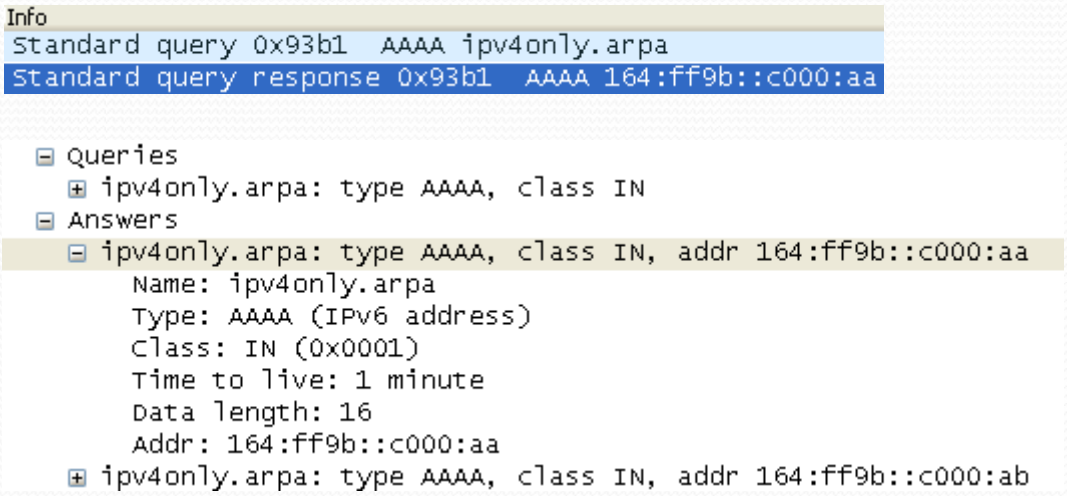
- Addressing architecture
- IPv6 address configuration
- Header structure
- DNS
- Fragmentation and PMTUD
- ICMP
- ….

Enlarged address space is the major benefit of IPv6



464XLAT technical details

- Prefix discovery (RFC7050)
  - The client will lookup the well-known FQDN ipv4only.arpa. If a AAAA record is presented for this well-know IPv4-only FQDN, the client can parse the response to find the Pref64 used within this network
- Wireshark



464XLAT technical details

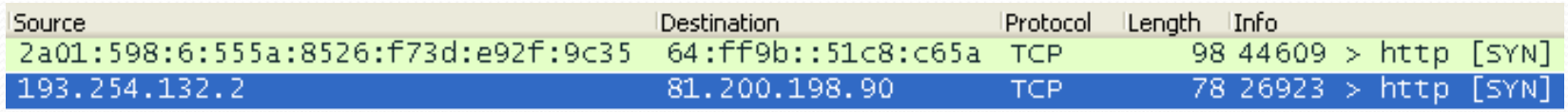
- Stateless NAT64 (RFC6145)
  - Algorithmically map IPv4 addresses to IPv6 addresses
  - Maps 32 bits IPv4 address into an IPv6 address
- Wireshark

Source	Destination	Protocol	Length	Info
192.0.0.4	157.55.56.171	TCP	76	39809 > 40031 [SYN] Seq=0
157.55.56.171	192.0.0.4	TCP	76	40031 > 39809 [SYN, ACK]
192.0.0.4	157.55.56.171	TCP	68	39809 > 40031 [ACK] Seq=1

Source	Destination	Protocol	Length	Info
2a01:598:6:55be::464	164:ff9b::9d37:38ab	TCP	96	39813 > 40031 [SYN] Seq=0
2a01:598:6:55be::464	164:ff9b::9d37:38ab	TCP	96	39813 > 40031 [SYN] Seq=0
164:ff9b::9d37:38ab	2a01:598:6:55be::464	TCP	96	40031 > 39813 [SYN, ACK]

464XLAT technical details

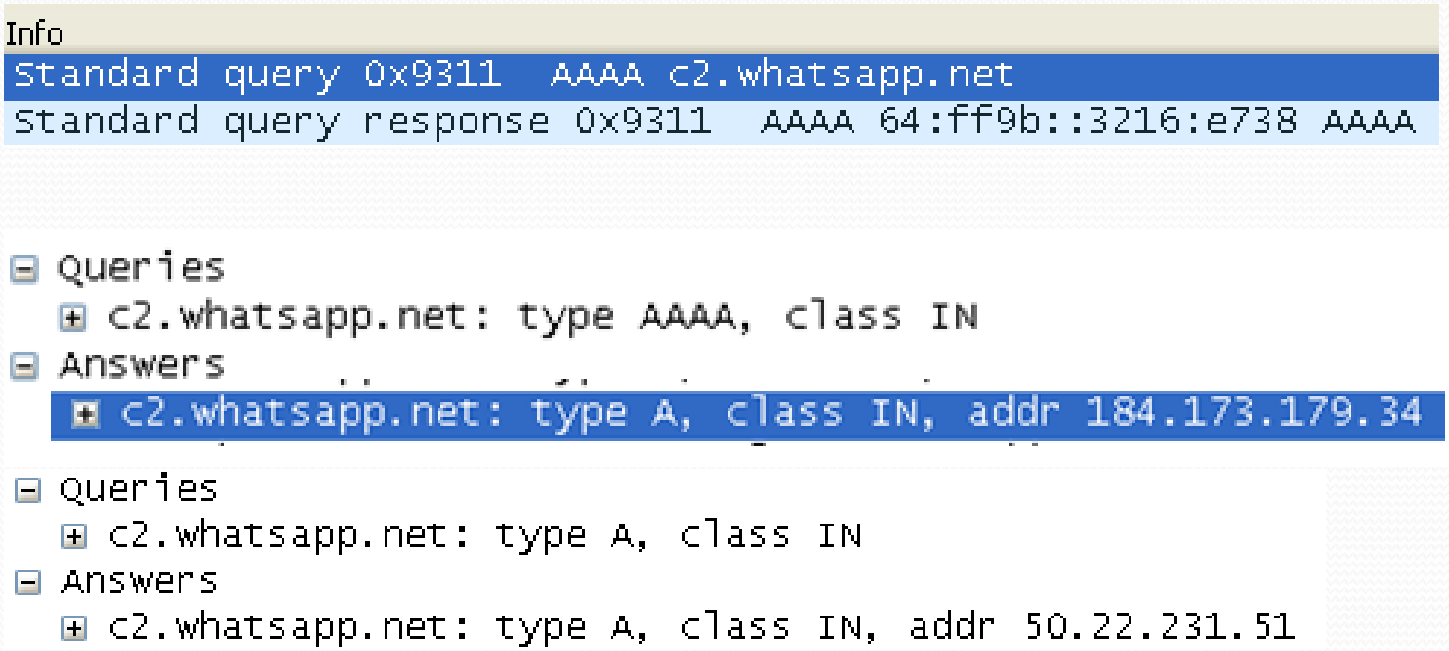
- Stateful NAT64 (RFC6146)
  - Dynamic translation of IPv4 packets to IPv6 packets and vice versa
  - Based on an IPv4 address pool, not deterministic
  - Translation based on session state
- Wireshark



Source	Destination	Protocol	Length	Info
2a01:598:6:555a:8526:f73d:e92f:9c35	64:ff9b::51c8:c65a	TCP	98	44609 > http [SYN]
193.254.132.2	81.200.198.90	TCP	78	26923 > http [SYN]

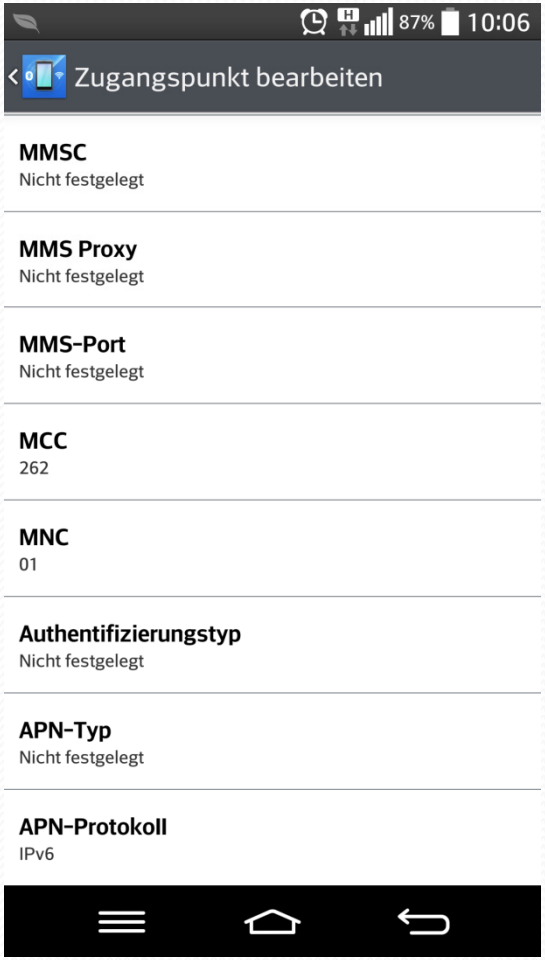
464XLAT technical details

- DNS64 (RFC6147)
  - When an FQDN does not have a AAAA record, the DNS64 will synthetically create one based on a network defined Pref64
  - The pref64 is a prefix hosted on the NAT64 for translation
- Wireshark



464XLAT technical details

- Android APN settings



- Android interfaces