# ETSI GR IP6 006 V1.1.1 (2017-11)


# Generic migration steps from IPv4 to IPv6

***Disclaimer***

The present document has been produced and approved by the IPv6 Integration (IP6) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference
DGR/IP6-0006

Keywords
IPv4, IPv6, transition

***ETSI***

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

# Contents

# Intellectual Property Rights

## Essential patents

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

## Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

# Foreword

This Group Report (GR) has been produced by ETSI Industry Specification Group (ISG) IPv6 Integration (IP6).

# Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

# 1 Scope

The present document outlines the generic transition steps from IPv4 to IPv6 [i.1], [i.2], including the transition necessity, principles and technology guidelines, generic transition steps, security implications and the generic step-by-step process.

# 2 References

## 2.1 Normative references

Normative references are not applicable in the present document.

## 2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] IETF RFC 791: "Internet Protocol", September 1981.

[i.2] IETF RFC 2460: "Internet Protocol, Version 6 (IPv6) Specification", December 1998.

[i.3] IETF RFC 1631: "The IP Network Address Translator (NAT)", May 1994.

[i.4] IETF RFC 1701: "Generic Routing Encapsulation", October 1994.

[i.5] Durand, A., Droms, R., Woodyatt, J., and Y. Lee: "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion".

[i.6] IETF RFC 5569: "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd)", January 2010.

[i.7] IETF RFC 7597: "Mapping of Address and Port with Encapsulation (MAP-E)", July 2015.

[i.8] IETF RFC 7599: "Mapping of Address and Port using Translation (MAP-T)", July 2015.

[i.9] IETF draft-ietf-v6ops-ipv6-ehs-in-real-world-02: "Observations on the Dropping of Packets with IPv6 Extension Headers in the Real World", December 2015.

[i.10] IETF RFC 6555: "A. Happy Eyeballs: Success with Dual-Stack Hosts", April 2013.

[i.11] IETF RFC 7359: "Layer 3 Virtual Private Network (VPN) Tunnel Traffic Leakages in Dual-Stack Hosts/Networks", August 2014.

[i.12] LinkedIn® Case Study: "IPv6 at a Social Media Company". Schuller, S. 11th Slovenian IPv6 Summit, June 21, 2016, Ljubljana, Slovenia.

NOTE: Available at https://go6.si/wp-content/uploads/2016/06/LinkedIn-Case-Study.pdf.

[i.13] IETF RFC 1702: "Generic Routing Encapsulation over IPv4 networks".

[i.14] IETF RFC 5969: "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) - Protocol Specification".

[i.15] IETF RFC 6751: "Native IPv6 behind IPv4-to-IPv4 NAT Customer Premises Equipment (6a44)".

[i.16] IETF RFC 5214: "Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)".

[i.17] IETF RFC 6343: "Advisory Guidelines for 6to4 Deployment".

[i.18] IETF RFC 4213: "Basic Transition Mechanisms for IPv6 Hosts and Routers".

[i.19] IETF RFC 6333: "Dual-tack Lite Broadband Deployments Following IPv4 Exhaustion".

[i.20] IETF RFC 6346: "The Address plus Port (A+P) Approach to the IPv4 Address Shortage".

[i.21] IETF RFC 7739: "Security Implications of Predictable Fragment Identification Values".

[i.22] Dan York: "Migrating Applications to IPv6", 2011.

# 3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

| | |
|---|---|
| 4464XLAT | Combination of Stateful and Stateless Translation |
| 6a44 | Native IPv6 behind NAT44 CPEs |
| 6over4 | IPv6 over IPv4 without Explicit Tunnels |
| 6RD | IPv6 Rapid Deployment |
| 6to4 | Connection of IPv6 Domains via IPv4 Clouds |
| AAAA | An AAAA record points a domain or subdomain to an IPv6 address |
| API | Application Program Interface |
| AYIYA | Anything-In-Anything |
| CGN | Carrier Grade NAT |
| CPE | Customer-Premises Equipment |
| DNS | Domain Name Server |
| DoS | Denial of Service |
| DS-Lite | Dual-Stack Lite |
| GRE | Generic Routing Encapsulation |
| ICMP | Internet Control Messages Protocol |
| ICP | Internet Content Provider |
| IoT | Internet of Things |
| IP | Internet Protocol |
| IPv4 | Internet Protocol version 4 |
| IPv6 | Internet Protocol version 6 |
| ISATAP | Intra-Site Automatic Tunnel Addressing Protocol |
| ISP | Internet Service Provider |
| MAN | Metropolitan Area Network |
| MAP | Mapping of Address and Port |
| MAP-E | Mapping of Address and Port - Encapsulation |
| MAP-T | Mapping of Address and Port - Translation |
| MSS | Maximum Segment Size |
| MTU | Maximum Transmission Unit |
| NAT | Network Address Translation |
| NAT-PT | Network Address Translation - Protocol Translation |
| NBMA | Non-Broadcast Multiple Access |
| SIIT | Stateless IP/ICMP Translation Algorithm |
| TCP | Transmission Control Protocol |
| Teredo | Tunnelling IPv6 over UDP through NATs |
| TSP | IPv6 Tunnel Broker with the Tunnel Setup Protocol |
| UDP | User Datagram Protocol |

# 4 Transition from IPv4 to IPv6

## 4.1 IPv6 transition necessity

For more than 35 years, IPv4 has been the core underlying technology enabling services such as the web, e-mail, and so forth. However, as a result of the unexpected growth of the Internet, the IPv4 32-bit address space has become a limiting factor to future Internet growth - that is, IPv4 will be unable to provide a globally routable unique IP address to each system to connect to the Internet. To overcome the exhaustion of IPv4 addresses, the Internet Protocol version 6 (IPv6) was developed, with 128-bit addresses that provide enough addresses to allow for the foreseeable future growth of the Internet.

IPv4 address exhaustion has accelerated IPv6 deployment. There are two complementary ways to ensure service continuity:

- Start introducing IPv6 and give new customers' IPv6 addresses.
- Implement IPv4 address sharing mechanisms to continue using IPv4 service.

Please note that IPv4 address sharing (using Network Address Translation (NAT) [i.3]) could only temporarily relieve the IPv4 address exhaustion problem, and that other challenges arise with massive deployment of IPv4 address sharing in the form of Carrier Grade NAT (CGN). CGNs not only result in more complicated networks and increased network management and operational costs, but also eventually introduce interoperability problems. Besides, due to address sharing, it results in loss of geo-location information, and difficult lawful intercept/abuse response. Therefore, transition to IPv6 is the only real solution to address the IPv4 address exhaustion problem.

Please note that the pressure resulting from IPv4 address exhaustion varies from one organization (e.g. ISP) to another due to many factors, such as the situation of address storage and Internet penetration. This results in a different pace for supporting IPv6.

Currently, there are a few applications or services only available in IPv6. And it is expected that it will take a long time for all IPv4-only services to be transitioned to IPv6. In fact, it is expected that many of such IPv4-only services will be "transitioned" to IPv6 when their corresponding systems are phased-out and replaced with IPv6-ready counter-parts. Therefore, it is expected that IPv4 and IPv6 will co-exist for a long time, and thus, even in the presence of IPv6-deployment, IPv4 provisioning needs to be taken care of.

## 4.2 Transition types

The original transition plan from IPv4 to IPv6 was based on the Dual Stack principle. Essentially, every node in the Internet would implement and enable IPv6 well before IPv4 address exhaustion. Unfortunately, such plan failed, and a number of transition technologies were subsequently implemented to allow for the incremental deployment of IPv6, and the co-existence of IPv4 and IPv6.

Transition technologies are employed for one of the following goals:

- Providing IPv6 connectivity
- Providing IPv4 connectivity (usually by multiplexing multiple devices in the same IPv4 address)

The following transition technologies are employed for providing IPv6 connectivity:

- Dual-stack
- Configured tunnels (6in4)
- Generic Routing Encapsulation (GRE)
- IPv6 Rapid Deployment (6rd) [i.6]
- Native IPv6 behind NAT44 CPEs (6a44)
- Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)

- Connection of IPv6 Domains via IPv4 Clouds (6to4)
- Tunnelling IPv6 over UDP through NATs (Teredo)
- IPv6 over IPv4 without Explicit Tunnels (6over4)
- Anything In Anything (AYIYA)
- IPv6 Tunnel Broker with the Tunnel Setup Protocol (TSP)

The following transition technologies are employed for providing IPv4 connectivity:

- Stateless IP/ICMP Translation Algorithm (SIIT)
- Stateful NAT64
- Combination of Stateful and Stateless Translation (4464XLAT)
- Dual-Stack Lite (DS-Lite) [i.5]
- MAP-E [i.7]
- MAP-T [i.8]

These transition technologies are discussed in clause 5.

# 5 Transition Principles and Technologies

## 5.1 Dual-stack

### 5.1.1 Dual-stack Principle

The dual stack principle was the original transition plan to IPv6. Essentially, every node on the Internet would implement and enable IPv6 before the IPv4 address space was exhausted. Thus, IPv4 support could start to be disabled at any time, since all communications could be performed over IPv6. Unfortunately, this plan failed, and the Internet hit IPv4 address exhaustion before widespread deployment of IPv6.

Nevertheless, Dual Stack is still the preferred transition technology for servers, since it allows IPv6-enabled clients to communicate with servers employing native IPv6 connectivity. Besides, the number of global IPv4 addresses required to provision servers is usually way smaller than the number of iPv4 addresses required to provision clients (compare the number of servers vs. clients in a usual Internet Service Provider).

Figure 1 illustrates the Dual Stack architecture.

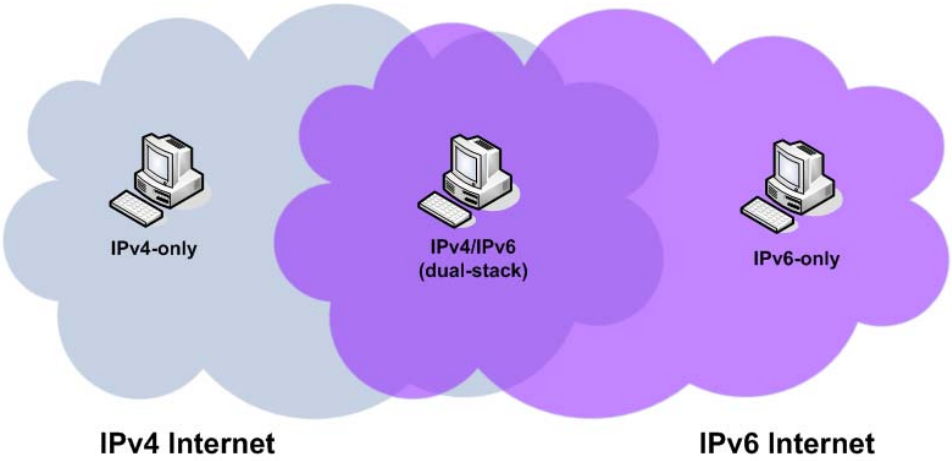


**Figure 1: Dual Stack architecture**

It is interesting to note that dual-stack essentially results in two separate networks. In principle, IPv4-only systems can communicate only with IPv4-only systems, while IPv6-only systems can only communicate with their counterparts. On the other hand, Dual Stack nodes can communicate with IPv4-only, IPv6-only, a Dual Stack (IPv6/IPv4) systems.

The DNS plays a key role in the IPv6 and the IPv4 world: for example, when a Dual Stack host means to browse the website www.example.com, it will typically query for both IPv4 and IPv6 addresses (A and AAAA records, respectively). Then it is up to the host (or host application) to use the available addresses.

### 5.1.2 Dual-stack Security Implications

The security implications of IPv6 transition technologies depend, for most part, on the specific paradigm being employed.

Dual-stack essentially implies that every node being transitioned will implement and operate with two different protocol stacks: an IPv6 stack and an IPv4 stack. Implementing, deploying, and operating an additional stack clearly increases the potential attack surface. In particular, since the maturity level of IPv6 implementations generally does not match that of existing IPv4 implementations, it is very likely that new bugs (possibly with security implications) will be discovered in the IPv6 code, and hence particular care should be taken to keep the operating system and applications up-to-date.

### 5.1.3 Dual-stack conclusion

Dual stack is generally the ideal mechanism for transitioning to IPv6, since it employs both native IPv6 and native IPv4 connectivity. The only drawback of this transition technology is that it requires the operation and management of two separate networks: an IPv6 network and an IPv4 network. In some scenarios, such as data centres, this issue may be considered enough of a motivation for employing SIIT, such that the server farm only implements IPv6, and IPv4 connectivity is provided via stateless translation.

## 5.2 Tunnelling

### 5.2.1 Tunnelling Principle

The tunnelling principle involves encapsulating packets of one internet protocol into packets of a (usually different) internet protocol. This is of use when "islands" of one internet protocol should be interconnected across a network that does not support the aforementioned internet protocol. For example, it can be employed to interconnect to IPv6 "islands" across the IPv4 Internet.

### 5.2.2 Tunnelling Security Implications

The tunnelling paradigm implies that IPv6 packets should be encapsulated in IPv4 to traverse IPv4-only networks. The use of any form of tunnelling mechanism clearly results in additional complexity in the resulting traffic. For example, a firewall or NIDS/IPS device might be unable to inspect the encapsulated IPv6 packet when tunnelling is employed. This might mean that some security controls could be circumvented as a result of the tunnelled traffic.

Additionally, some automatic-tunnelling mechanisms might interact in unforeseen ways: for example, unless proper mitigations are in place, they might be subject to routing loop attacks that might result in Denial of Service (DoS) scenarios.

Finally, inadvertent use of automatic-tunnelling mechanism might increase the attack surface of networks that are assumed to be "IPv4-only".

### 5.2.3 Configured tunnels (6in4)

Configured tunnels require a network administrator to manually configure the tunnels endpoints. While this implies some burden on the administrator side, this also means that IPv6 connectivity problems are rather simple to troubleshoot, since the two endpoints of the tunnel are clearly defined.

Figure 2 illustrates a typical 6in4 tunnel scenario.

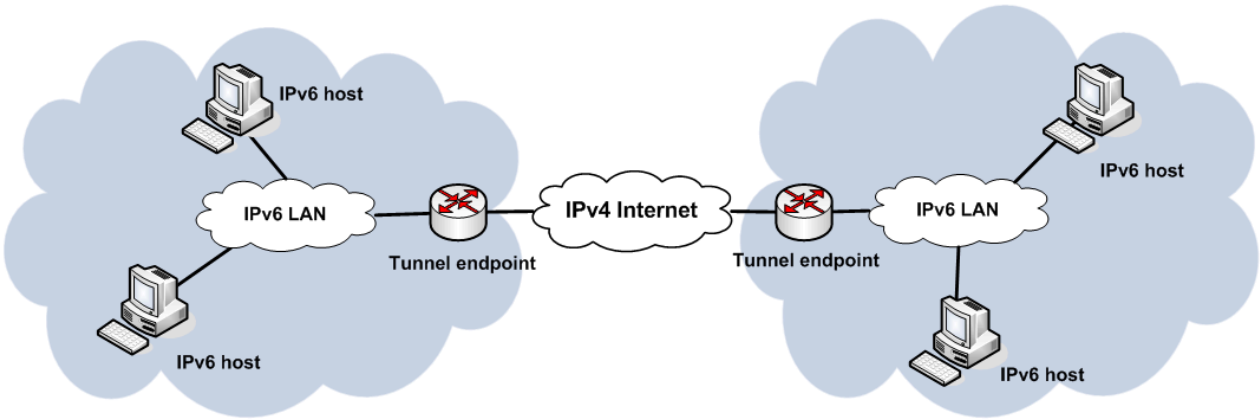


**Figure 2: Typical 6in4 tunnel scenario**

### 5.2.4 Generic Routing Encapsulation (GRE)

GRE is essentially a configured tunnel that employs Generic Routing Encapsulation (GRE) IETF RFC 1701 [i.4] for the encapsulation of packets (with IPv4 encapsulation being specified in IETF RFC 1702 [i.13]). The properties of GRE tunnels are similar to those in which an internet protocol (e.g. IPv6) is encapsulated in another internet protocol (e.g. IPv4).

### 5.2.5 Connection of IPv6 Domains via IPv4 Clouds (6to4)

6to4 is an automatic-tunnelling mechanism that can provide IPv6 connectivity to an IPv4 network that employs global IPv4 addresses, or where last-hop router can employ a global address to operate as a 6to4 router.

Figure 3 illustrates the 6to4 architecture.

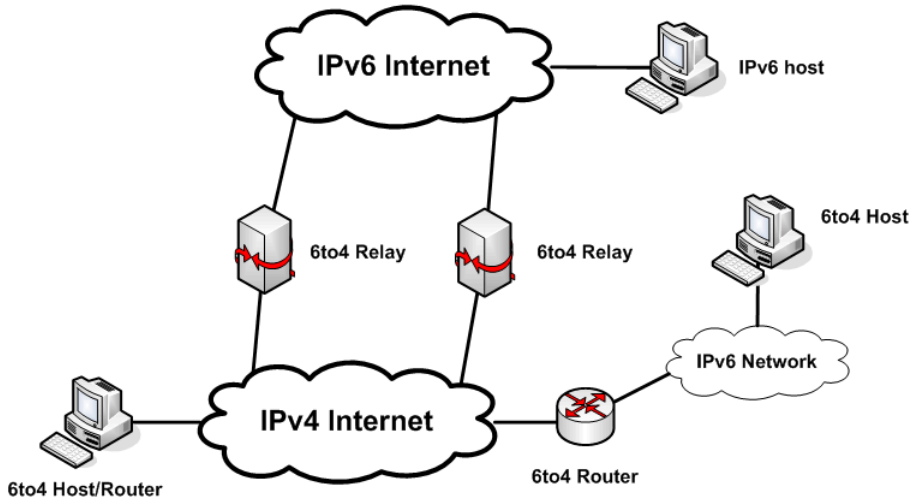


**Figure 3: 6to4 architecture**

IETF RFC 6343 [i.17] contains advisory guidelines for 6to4 deployment. Unfortunately, a number of shortcomings in this technology has essentially resulted in this technology being considered (in practice) obsolete.

### 5.2.6 IPv6 Rapid Deployment (6rd)

6rd essentially constitutes the deployment of 6to4 within an organization's network, enabling the incremental deployment of IPv6 in an IPv4-only network, while avoiding the shortcomings of 6to4. It is specified in IETF RFC 5969 [i.14], and is used by service providers to connect customer networks behind a CPE (Customer Premises Equipment) to the IPv6 Internet.

The structure of the 6rd protocol is based on 6to4, and it has the same minimal overhead as all protocols that use protocol 41 encapsulation. The main differences between 6rd and 6to4 are that 6rd is meant to be used inside a service provider's network and does not use a special IPv6 prefix but one or more prefixes routed to the service provider. As such, 6rd users are not immediately recognizable by their IPv6 address the way 6to4 users are. Where 6to4 uses relays based on global anycast routing, 6rd uses relays provided and maintained by the service provider. Because of this architecture, the tunnel does not traverse unknown networks; this makes any debugging much easier.

Figure 4 illustrates a sample 6rd deployment. Detail of address format is able to be found in IETF RFC 5569 [i.6].

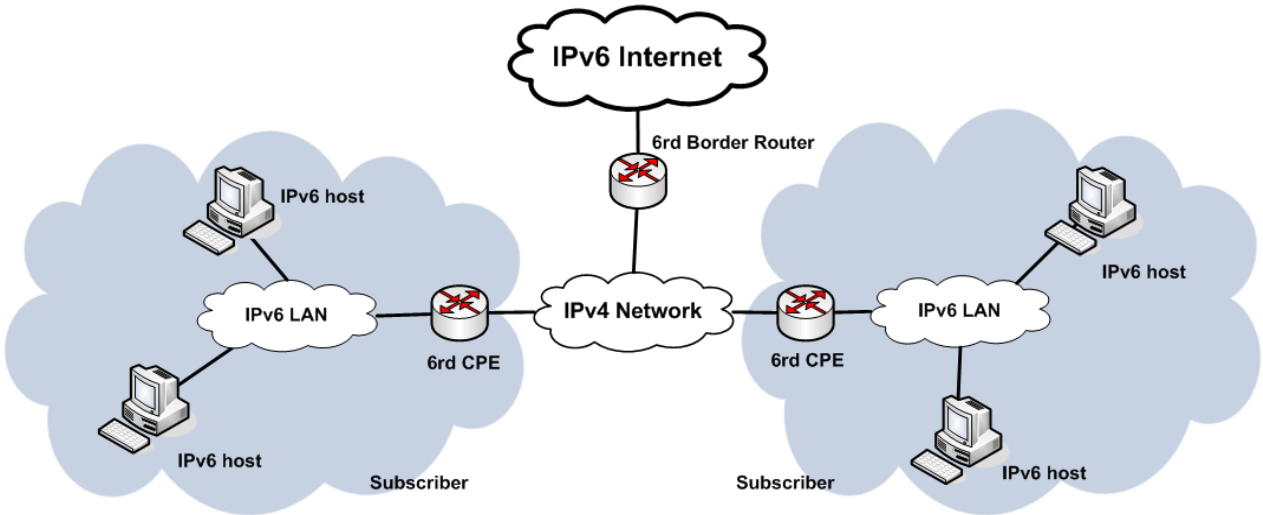


**Figure 4: Sample 6rd deployment**

### 5.2.7 Native IPv6 behind NAT44 CPEs (6a44)

The purpose of 6a44 is to enable Internet Service Providers to establish IPv6 connectivity for their customers, in spite of the use of a CPE or home gateway that is not prepared for IPv6. The infrastructure required for this is a 6a44 relay in the ISP's network and a 6a44 client in the customer's internal network. 6a44 is to Teredo what 6rd is to 6to4. That is, it is designed to avoid Teredo limitations: with 6a44, ISPs have full control of the involved relays, such that the well-known Teredo shortcomings are avoided. Where Teredo was designed as a global solution without dependency on ISP cooperation, the 6a44 tunnel explicitly assumes ISP cooperation. Instead of using Teredo's well-known prefix, a /48 prefix out of the ISP's address space is used. A well-known (anycast) IPv4 address has been assigned for the 6a44 relay to be run inside the ISP network without client configuration. This well-known address is allocated from the same IPv4 /24 as 6to4.

As part of its bootstrapping, a 6a44 client requests an address from the 6a44 relay, and a regular keepalive sent by the 6a44 client to the 6a44 relay keeps mapping state in NATs and firewalls on the path alive. Traffic passed from the native IPv6 Internet to 6a44 is encapsulated in UDP and IPv4 by the relay and decapsulated by the 6a44 client; the opposite is done in the other direction.

6a44 is specified in IETF RFC 6751 [i.15] as an experimental protocol; at the time of this writing are no known 6a44 implementations or deployments.

### 5.2.8 Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)

ISATAP is an automatic-tunnelling mechanism which allows for incremental deployment of IPv6 within an IPv4-only organizational network. It is specified in IETF RFC 5214 [i.16], and is similar to 6over4, but without the requirement that the IPv4 network supports multicast; unlike 6over4, ISATAP uses a Non-Broadcast Multi-Access (NBMA) communication model and thus does not support multicast. The mechanism assigns IPv6 addresses whose Interface Identifier is solely defined by a node's IPv4 address, which is assumed to be unique.

Figure 5 illustrates a sample ISATAP deployment.

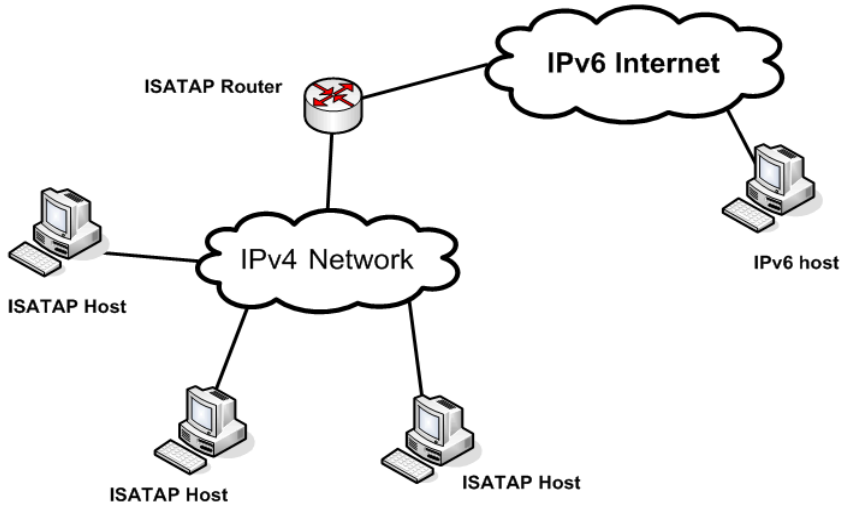


**Figure 5: Sample ISATAP deployment**

### 5.2.9 Tunnelling IPv6 over UDP through NATs (Teredo)

Teredo is an automatic-tunnelling mechanism specifically designed to provision IPv6 connectivity to IPv4 nodes behind Network Address Translators (NATs). While Teredo can be considered to have a clever design to overcome the challenge represented by NATs, it suffers many of the drawbacks of other automatic-tunnelling mechanisms. Besides, it typically leads to poor IPv6 connectivity (e.g. in terms of delays), and hence deployment of Teredo is discouraged.

Figure 6 illustrates the Teredo architecture.

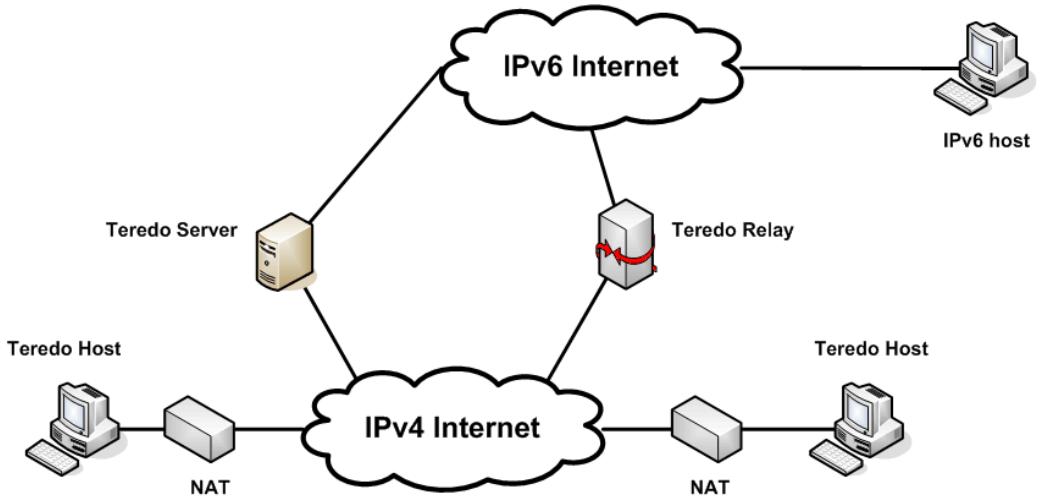


**Figure 6: Teredo architecture**

### 5.2.10 IPv6 over IPv4 without Explicit Tunnels (6over4)

6over4 was designed to work within a single organization's IPv4 network, where IPv6-capable hosts and routers are separated by IPv4-only routers. 6over4 treats the IPv4 network as a "virtual Ethernet" for the purpose of IPv6 communication, and uses IPv4 multicast to tunnel IPv6 multicast packets.

However, no implementation of 6over4 was ever produced or deployed.

### 5.2.11 Anything In Anything (AYIYA)

AYIYA [AYIYA] is designed for use by the SixXS (www.sixxs.net) tunnel-broker service. The AYIYA protocol defines a method for encapsulating any protocol in any other protocol, with the most common way of deploying AYIYA is to use the following sequence of headers: IPv4-UDP-AYIYA-IPv6, though other combinations like IPv4-AYIYA-IPv6 or IPv6-SCTP-AYIYA-IPv4 are also possible. It can be considered a similar technology to that of "IPv6 Tunnel Broker with Tunnel Setup Protocol (TSP)".

### 5.2.12 IPv6 Tunnel Broker with the Tunnel Setup Protocol (TSP)

Tunnel Broker essentially provides the means to dynamically establish a tunnel between the client and the tunnel server, thus relieving the user from any manual network configuration.

Figure 7 illustrates the architecture of a possible Tunnel Broker deployment.

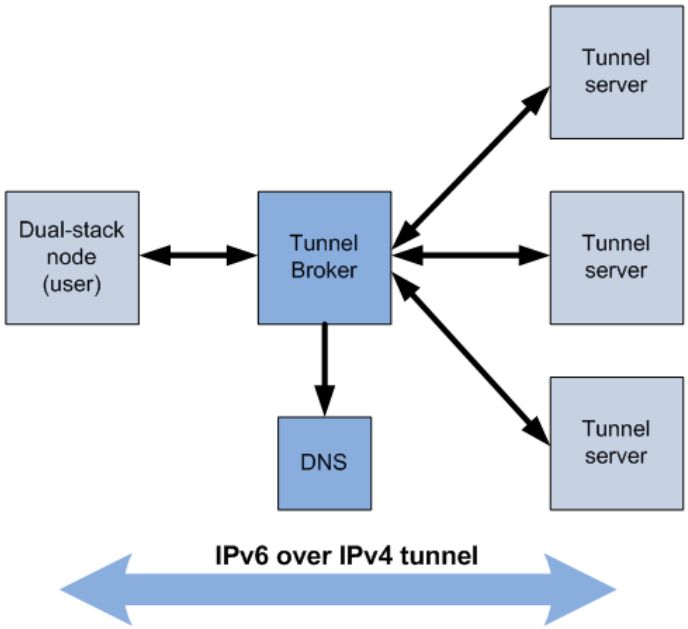


**Figure 7: Architecture of a possible Tunnel Broker deployment**

## 5.3 Translation

### 5.3.1 Translation Principle

Translation essentially involves the translation of packets of an internet layer protocol. From the perspective of IPv6 transition, there are essentially two different scenarios where translation may be applied:

- Two networks employing different layer-3 protocols (e.g. IPv6 and IPv4) need to be interconnected.
- Multiple devices need to be multiplexed into a single layer-3 address.

In the first scenario, IPv6 packets are translated into IPv4 packets, and vice versa. This allows IPv6-only networks to communicate with IPv4-networks, albeit with some limitations. Figure 8 illustrates a typical setup.

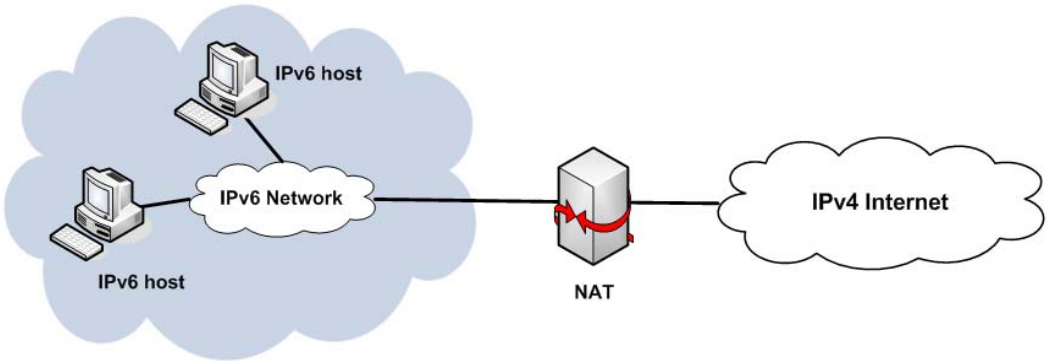


**Figure 8: A typical transition setup**

IPv6/IPv4 translation is generally only employed for mobile devices/end-points, since it is usually the case that such devices only run a limited set of applications.

The second case is that in which multiple devices need to be multiplexed into a single IP address. This is the case of traditional NAT-PT (Network Address Translation - Port Translation), where multiple devices share a single public address. NAT-PT implies that the client-side (ephemeral) transport protocol port numbers need to be overwritten by the NAT device, in order to multiplex multiple devices into the same IP address.

### 5.3.2 Translation Security Implications

Translation technologies may, in many cases, introduce a "single point of failure": the failure or successful attack of the translating device may result in the Denial of Service of multiple nodes/systems. For obvious reasons, stateful translation makes the mitigation of Denial of Service attacks more challenging.

### 5.3.3 Stateless IP/ICMP Translation Algorithm (SIIT)

SIIT is a stateless translation between IPv6 and IPv4 packets. Its limitation is that it implies a 1:1 mapping between IPv6 addresses and IPv4 addresses. Therefore, it is mostly employed in datacentre environments, and as part of other transition technologies such as 464XLAT.

### 5.3.4 Stateful NAT64

Stateful NAT64 multiplexes many IPv6 devices into a single IPv4 address. Such mapping requires state on the NAT64 device, and requires that both the IP addresses and transport protocol port numbers be translated in a similar vein as traditional NAT-PT.

### 5.3.5 Combination of Stateful and Stateless Translation (464XLAT)

464XLAT essentially combines SIIT and stateful NAT64 to multiplex multiple IPv6-only devices into the same public IPv4 address, such that (some) IPv4-only applications can be used on IPv6-only systems. As a result, the properties of 464XLAT are the properties of the aforementioned two transition technologies.

### 5.3.6 Dual-Stack Lite (DS-Lite)

Dual-Stack Lite enables the sharing of IPv4 addresses among customers by combining IP in IP (IPv4-in-IPv6) and Network Address Translation (NAT). It also de-couples IPv6 deployment in the service provider network from the rest of the Internet, making incremental deployment easier.

## 5.4 Mapping of Address and Port

### 5.4.1 Mapping of Address and Port Principle

Mapping of Address and Port essentially tries to overcome the challenge of IPv4 address depletion by employing part of the transport protocol bits to "extend" the IPv4 address space. Address mapping is particularly useful in scenarios where scarcity of public IPv4 addresses would require the use of stateful IPv6/IPv4 translation.

MAP leverages the natural aggregation capability of the address and port space to allow IPv4 addresses to be translated or encapsulated in IPv6, without requiring a stateful translator on the service provider network. This provides a production-quality, deployable IPv6 transition mechanism, which allows service providers to share scarce IPv4 address resources, while deploying an IPv6-only provider network.

### 5.4.2 MAP-E

MAP-E is a mechanism for transporting IPv4 packets across an IPv6 network using IP encapsulation combining with a generic mechanism for mapping between IPv6 addresses and IPv4 addresses as well as transport-layer ports.

Figure 9 illustrates the Teredo architecture. Detail of address format is able to be found in IETF RFC 7597 [i.7].


**Figure 9: Sample MAP-E deployment**

### 5.4.3 MAP-T

MAP-T is a solution architecture based on "Mapping of Address and Port" stateless IPv6-IPv4 Network Address Translation (NAT64) for providing shared or non-shared IPv4 address connectivity to and across an IPv6 network.

Figure 10 illustrates the Teredo architecture. Detail of address format is able to be found in IETF RFC 7599 [i.8].


**Figure 10: Sample MAP-T deployment**

# 6 Sample Transition Scenarios from Operators

## 6.1 ISP from France

For fixed networks, the base recommendation is IPv6 dual-stack IETF RFC 4213 [i.18]. In order to ensure IPv4 service continuity, DS-Lite IETF RFC 6333 [i.19] is the Group-wise recommendation while with A+P IETF RFC 6346 [i.20] is the CGN exit strategy.

For mobile networks, IPv6 only + NAT64 is deployed in some affiliates of the Group.

As per November 24, 2016, there are around 3,7 million subscribers with this ISP that are serviced with an IPv6 connectivity.

## 6.2 ISP from China

This ISP plays the main role in the market of fixed network in China. They has deployed IPv6 in its 4G mobile networks with dual stack in 2 provinces since May 2015, and more than 60 % online 4G users is IPv6-enabled.

Now in the backbone network of the ISP, IPv6 are 100 % implemented, while the percentage is 70 % in MAN.

However, the flow in IPv6 only occupies 10-20 % of the total number inside China. The reason that IPv6 flows remain in a low level can be summarized into two points:

1) There is only a very limited number of ICP which provide service in IPv6. Most of ICPs/APPs are still only in IPv4.

2) There is a large number of CPEs in Chinese family that do not support IPv6 as they are not sold by operators. Operators could not make them support IPv6 by simply upgrade the software. This scenario is very common in most cities/countries of China. There are a couple of popular brands of CPEs sold by small telecommunication companies which have a lower price than the ones offered by operators.

The ISP is using a broader set of IPv6 transition technologies. They are even the proposer/proponent of Lightweight 4over6. They are implementing dual-stack, DS-Lite and NAT444.

## 6.3 Another ISP from China

Now the number of IPv6 users/customers take up 2-3 % of the sum number of users/customers of the ISP.

The only IPv6 transition technology the ISP used is dual-stack.

## 6.4 ISP from United States

The ISP's transition strategy is native dual-stack. Their network is fully IPv6 enabled and this ISP is strongly promoting IPv6 to customers and peers.

## 6.5 Summary

A number of operators were polled regarding their strategy for transitioning to the IPv6 protocol.

Many operators are reluctant to share their transition strategy, since it is considered a trade secret. Other operators have been more open in this respect, and able to share their IPv6 transition strategy.

# 7 Current Levels of Global IPv6 Deployment and Traffic

The deployment of IPv6 by large content providers has meant that when operators deploy IPv6, a substantial increase in IPv6 traffic is experienced. A number of content providers publicly offer their IPv6 traffic statistics, with the World IPv6 Launch Day web site (http://www.worldipv6launch.org/measurements/) providing an average of the results from the aforementioned measurements.

It is interesting to note that, while the numbers look promising, they indicate the average IPv6 traffic as seen by IPv6-enabled content providers. However, at the time of this writing less than 20 % of Alexa's Top 1 000 web sites are IPv6-enabled.

Some content providers have also reported a reduced latency when accessing content with IPv6 as opposed to IPv4 [i.12].

# 8 Application Transition

While, in principle, the transition from IPv4 to IPv6 should not imply any actual changes in application protocols, in practice applications may be affected by the transition to IPv6 for a number of reasons including, but not limited to:

- Using low-level Application Programming Interfaces (APIs) that are protocol dependent, or that are employed in a protocol-dependent manner.
- Using network-layer information in a protocol-dependent way.
- Dealing with IPv6 network-layer issues.
- The interaction of IPv6 and IPv4, particularly in scenarios in which IPv6 connectivity is sub-optimal.

Applications have traditionally employed low level APIs for (such as the Sockets API) for network programming. In such cases, a legacy application typically requests DNS A records corresponding to a domain name, and subsequently e.g. establishes a TCP connection. Such applications would need to be updated such that either DNS A or AAAA records can be employed when available.

Another issue that may affect applications is the processing of network-layer information (particularly IP addresses) in a protocol-dependent way. For example, an application might expect IP addresses to be 32 bits (as in IPv4), and hence crash or result in other pathological behaviour when presented with the 128-bit longer IPv6 addresses.

An application may have to deal with IPv6 network-layer specific issues. For example, in order to circumvent Path-MTU black holes, an application may explicitly signal the lower-layers to employ the IPv6 minimum MTU (1 280 bytes). Similarly, because of the widespread filtering of IPv6 packets that employ extension headers [i.9], an application may want to avoid the use of fragmentation to the extent that is possible.

Finally, the interaction between IPv6 and IPv4 may affect different applications in different ways. For example, an interactive application (such as a web browser), may need to handle the case where multiple IPv6 and/or IPv4 addresses are available to connect to a remote peer or server, and some of those addresses are non-working. Some applications (notably web browsers) have addressed this issue by implementing the "Happy Eyeballs" IETF RFC 6555 [i.10] algorithm/technology. However, since this technology is typically bundled with the application code (rather than in underlying libraries), it requires changes/updates on a per-application basis. Other applications may be affected by IPv4/IPv6 interaction in different ways. For example, an IPv6-unaware application may be affected in unexpected ways when deployed in IPv6-enabled networks; VPN-tunnel implementations being one of them IETF RFC 7739 [i.21].

The aforementioned issues can be summarized, along with the general guidelines provided in [i.22], as follows:

- Analyse possible changes that may be required in the application's user interface.
- Make sure that the application is able to employ both A and AAAA DNS resource records.
- Interactive applications should be able to handle the case where a domain name maps to multiple A and/or AAAA resource records, without incurring into connection-establishment delays.
- Implement any necessary changes such that the application can handle IP addresses (whether IPv4 or IPv6) in a protocol-independent manner.
- Determine if the application exposes or consumes APIs where there are IP address format dependencies, and apply any necessary changes to handle IP addresses in a protocol-independent manner.
- Consider how the traditional use of multiple addresses per interface (including the use of temporary addresses) may affect the application.
- Consider the possible case in which the underlying system does not support IPv4 or IPv6. This means that the code may need conditional compilation to handle compilation in IPv4-only and IPv6 only environments.
- Incorporate IPv6 into application-testing methodologies.

# 9 Security considerations

There are a number of factors that make the IPv6 protocol suite interesting from a security standpoint. Firstly, being a new technology, technical engineers have much less confidence with the IPv6 protocols than with their IPv4 counterpart, and thus it is more likely that the security implications of the protocols be overlooked when the protocols are deployed. Secondly, IPv6 implementations are much less mature than their IPv4 counterparts, and thus it is very likely that a number of vulnerabilities will be discovered in them before their robustness matches that of the existing IPv4 implementations. Thirdly, parity of features between IPv6 and IPv4 products (such as firewalls) is missing; either in terms of functionality and/or in terms of performance. Fourthly, a number of transition/co-existence technologies have been developed to aid in the deployment of IPv6 and the co-existence of IPv6 with the IPv4 protocol. These technologies not only generally result in increased complexity in the network, but also introduce attack vectors in IPv4 networks that may be employing these technologies unexpectedly.

# 10 Transition pitfalls

There are a number of well-known transition pitfalls. Some of them have to do with the transition process itself, while others have to do with operational issues arising from the use of IPv6 and/o the deployment of IPv6 transition technologies.

One of the issues associated with the operation of IPv6 networks is that the use of transition technologies such as tunnelling typically results in a decrease of the Path-MTU. This, when coupled with the widespread dropping of ICMP error messages leads to the so-called "black-holes", where packets are dropped without any apparent reason being reported. The workaround for this problem is typically the artificial reduction of the Path-MTU, either by means of TCP MSS "clamping" and/or by employing the IPv6 minimum MTU (1 280 bytes) at the end-nodes.

An issue associated with the operation of dual-stack networks is the eventual long delays to establish TCP connections with dual-stack nodes. Typically, a dual-stack site will have its domain name map to a number of IPv4 and IPv6 addresses. Applications typically try each of these addresses in turn: the first address return is tried, and if e.g. TCP connection-establishment fails, the next address in turn will be tried. TCP timeouts are typically of at least 75 seconds, which means that if an address is somehow unreachable, it may take a considerable time before subsequent addresses are tried. For interactive applications such as the web, these delays are overkill. The current workarounds for these problems is the implementation of "Happy Eyeballs" at interactive applications, such that a number of addresses are tried in parallel, and whichever address succeeds first is employed for communicating with the corresponding site.

# 11 Conclusions

As the IPv4 address free pool held by the Internet Assigned Numbers Authority (IANA) ran out in 2011 and the unexpected growth of the Internet, IoT, cloud computing, etc., the IPv4 32-bit address space has become a severe limiting factor to future Internet growth. The well recognized solution by the industry, IPv6 with 128-bit addresses and its related technologies are the way of scaling the addressing needs for the foreseeable future growth of the Internet. After outlining the generic transition steps from IPv4 to IPv6, including the transition necessity, principles and technology guidelines, generic transition steps, security implications, the present document also shows the examples of the address situation and the deployment status of above technologies in several well know operators.

# Annex A:
# Authors & contributors

The following people have contributed to the present document:

**Rapporteur:**
Will (Shucheng) LIU, Secretary, ETSI IP6 ISG, Huawei

**Authors:**
Ying CHEN, China Unicom
Fernando Gont, Huawei
Qiong SUN, China Telecom
Chongfeng XIE, China Telecom

**Contributors:**
Latif Ladid, Chair, ETSI IP6 ISG, University of Luxembourg
Yiu Lee, Comcast
Patrick Wetterwald, Vice-Chair, ETSI IP6 ISG, Cisco

**Acknowledgement:**
Mohamed Boucadair, Orange

# History

| Document history | | |
|---|---|---|
| V1.1.1 | November 2017 | Publication |
| | | |
| | | |
| | | |
| | | |