

Project Number:	IST-2001-32603
Project Title:	6NET
CEC Deliverable Number:	32603/Partner/DS/No./A1
Contractual Date of Delivery to the CEC:	March 31, 2002
Actual Date of Delivery to the CEC:	
Title of Deliverable:	IPv6 DNS service for the 6NET network
Work package contributing to Deliverable:	WP3
Type of Deliverable*:	P - Prototype
Deliverable Security Class**:	PU - Public
Editors:	Wilfried Woeber, Bruno Ciscato, Olivier Courtay
Reviewers	João Pagaime, João Nuno Ferreira
Contributors:	DANTE staff, NREN staff, cisco staff, Alexander Grall, João Pagaime, João Nuno Ferreira

* Type: P - Prototype, R - Report, D - Demonstrator, O - Other

** Security Class: PU- Public, PP – Restricted to other programme participants (including the Commission), RE – Restricted to a group defined by the consortium (including the Commission), CO – Confidential, only for members of the consortium (including the Commission)

Abstract:

This document describes the requirements, DNS technology background and documents for the setup of the DNS service for 6NET, which is required to support the operational phase of 6NET. Advanced features like DNSSEC are referred to, as a preparation plan for future service upgrade to be implemented in the near future.

Keywords:

DNS, DNSSEC, IPv6

Executive Summary

After the setup and operation of the first set of services early on the first year of the project, it was envisaged that a more advanced set of DNS services would be deployed and a second version of this text produced.

The main improvement would be DNSSEC in the 6Net backbone.

But this second version of D3.2.1 deliverable suffered a considerable delay regarding its initial due date. Two set of problems caused this.

Firstly, although it was initially included in the Description of Work, as the project progressed it became clear that the usefulness of dealing with DNSSEC issues in an IPv6 project like 6Net was questionable.

Secondly and more important, the DNSSEC standard that was stable at the beginning of the project proved in several trials to have severe practical implementation problems and soon work on alternative and better standard started. The schedule of this second standard, which is incompatible with the previous one, slipped forward in time, until when by the end of 2003 the Project Review was done it was still not available.

The decision was then to either wait until new and updated DNSSEC software became available or write down these difficulties together with an implementation plan for a future implementation of DNSSEC in 6Net. The explicit request by the reviewers for documenting unexpected difficulties reinforced this option.

Since then up to the writing date of this text, further confirmation was obtained that new DNSSEC standard implementations will be available in the second Quarter of 2004. This will enable the deployment of DNSSEC in the 6Net backbone.

A short reference to IDNA was also included.

Table of Contents

1	INTRODUCTION.....	4
1.1	BASIC DNS FUNCTIONALITY	5
1.2	TERMS AND ACRONYMS USED.....	5
2	SOME DNS TECHNOLOGY	7
3	DNS SERVICE FOR 6NET	8
3.1	REQUIREMENTS	8
3.2	NON-GOALS	9
3.3	IMPLEMENTATION CONSIDERATIONS	9
4	DNS SERVICE IMPLEMENTATION	10
4.1	FORWARD DNS SERVICE FOR 6NET	10
4.1.1	Early activities and basic functionality	10
4.1.2	Support functions for network operations.....	11
4.1.3	DNS root name server in 2004.....	13
4.2	REVERSE DNS SERVICE FOR 6NET	14
4.3	DNSSEC.....	17
4.3.1	Delay of this activity	17
4.3.2	Deployment Plan.....	17
4.4	IDNA.....	18
5	CONCLUSION.....	20
6	APPENDIX I: LIST OF PER POP-LOCATION SUPPORT DOMAINS	21
7	APPENDIX II: SNAP-SHOT LIST OF SYSTEMS PROVIDING DNS SERVICE FOR 6NET.....	21

1 Introduction

This document describes the background, requirements, and current state of the Prototype of the DNS Service for 6NET. It provides a snap-shot as of the end of March, 2002 (updated in February 2004). The implementation of this service is subject to on-going extension and re-configuration activities. While the document is submitted as a formal deliverable, it is a "living document", supporting the WP3 responsibilities within the project.

The following 2 sections are provided up-front, in order to help the reader understand the functional framework and to help in understanding the used acronyms.

It is **not** intended to replace reading the relevant RFCs, or referring to books focussing on DNS technology and operations in particular.

1.1 Basic DNS functionality

Forward DNS:

The forward DNS service provides name-based identification and access to the various components accessible on an internet, by supplying a destination IP address.

These components to be addressed are individual hosts, particular interfaces on a network, services accessible for end users and the like.

Reverse DNS:

This service provides the translation of numeric IP addresses (IPv4 or IPv6) back to meaningful names for "human consumption", as well as for debugging tools and some (weak) security facilities.

Reverse DNS is some sort of "inverse" function to the forward DNS service, although the information returned upon a particular reverse DNS lookup is not necessarily consistent with the content of the corresponding forward name.

Depending on the point of view, this can be seen as a flaw or as a necessary degree of freedom in managing an internet.

Implementation:

DNS is implemented as a distributed database with "weak" replication mechanisms. The term "weak" refers to the fact that the replicas are updated across the network, according to predefined schedules and/or trigger and notification mechanisms. This approach can result in some delay in the update process which can (sometimes) become noticeable for the end users.

The master copy of DNS data for a particular domain is stored and maintained in a zone file at the Primary Name Server to which a domain has been delegated. Note that a particular host (a name server) can support many different domains at the same time, both as a master and as a slave name server for distinct domains.

1.2 Terms and Acronyms used

In order to maintain consistency, the terms

"Master Server" (or simply "master") and

"Slave Server" (or "slave") are used.

For most aspects the term "Master Server" is equivalent to the (traditional) notion of a "Primary Name Server", and the term "Slave Server" is equivalent to "Secondary Name Server".

Explaining the subtle implementation differences that were introduced in BIND for version 9, and which led to the change in terminology is beyond the scope of this document.

Some other abbreviations and acronyms are used throughout this document. In particular,

- DNS Domain Name System

- FQDN Fully Qualified Domain Name

- ISATAP Intra-Site Automatic Tunnel Addressing Protocol

- NAT-PT Network Address Translation - Protocol Translation

- PoP Point of Presence

- RFC Request For Comment (IETF Document)

- PTR RR Pointer Resource Record

- RIR Regional Internet Registry

- RR DNS Resource Record

- SOA Start Of Authority

- sTLA subTLA (Top Level Aggregator)

- TLA Top Level Aggregator

- TLD Top Level Domain

Some DNS Technology

DNS, the Domain Name System for the Internet, is a pretty complex set of functions and services which can provide a reliable translation service from names (FQDNs) to addresses (IPv4 and IPv6), from addresses to names, as well as for specific support functions for particular applications (like MX for routing of electronic mail, supporting ISATAP, NAT-PT)

Implementation of DNS services for a particular (set of) domain(s) requires the operation of Name Servers (master and slave servers: those machines which manage and maintain, and hold the relevant parts of, the distributed nameservice database) and a method for communicating with the clients. On the client side the functional entity is a resolver or a resolver library, which accepts the query requests from an application program and then talks to the name servers on behalf of the user's application.

In order to ensure proper operation of the whole system, the types of data stored in the database (RRs: resource records) and the flows of information between the servers and the clients (resolvers) need careful attention. One of the most important aspects here is that there is no "fixed" relationship between the types of records stored in the database (i.e., type A for an IPv4 address, type AAAA for an IPv6 address, which are returned upon a query), and the transport protocol used to send data back and forth between name servers and between name servers and a resolver (IPv4 or IPv6).

Indeed, it is perfectly normal these days to already store IPv6-related data in some zones (in the distributed database), but to still use IPv4 to submit queries, to perform recursion (also known as "tree-walk") and to return results. Also, most of the zone transfers between slave servers and a master server actually use IPv4 (TCP) for the data transfer.

These considerations equally apply to communication with the root name servers and the TLD name servers.

Using this approach has the big advantage that the software in the existing system of nameservers (root, TLD, second level domains) requires very little change. In order to preserve the stability of the DNS for the Internet, the community is reasonably reluctant to embark on big or hurried upgrade projects.

The big disadvantage of this approach is the fact that end systems (usually hosts) typically still need an IPv4 protocol stack to talk to the DNS, even if the applications would already be able to use IPv6 exclusively. If this "dual-stack" method is not appropriate, then a more complex system of resolvers and (forwarding or translating) name servers needs to be deployed. This can take care of the continued IPv4 address space consumption but adds complexity, and sometimes single points of failure, to the whole systems.

Given this background, it should be obvious that 6NET tries to use IPv6 as the transport protocol as soon as possible and as widely as possible during the life-cycle of the project! Indeed many of the name servers in use within, or in support of, 6NET do support IPv6 as a transport protocol *right now*.

(See Appendix 2 for a snap-shot of systems already in use to provide name services for 6NET, by the end of March 2002, updated in February 2004)

In order to understand some of the decisions taken, it should be noted that DNS is *both* an application (in fact a distributed and replicated database) which uses basic network transport services (IPv4 UDP and IPv4 TCP, IPv6 UDP and IPv6 TCP) like any other application on the Internet, as well as an essential support service for the operation and management of an IP-based internet.

The overall aim of 6NET is to "go native" as early as possible. The network layer itself should be as stable as reasonably possible. This means that routine operations, management, fault identification and repair need be straight-forward. This requires alignment and integration of the DNS service for 6NET with the existing IPv4-based Internet. This means operation of DNS services on the top of IPv4 protocol stack, and completing the IPv6-based communication provisions.

2 DNS service for 6NET

Most of the partners in 6NET (NRENs, companies) either have a working DNS environment already at their disposal or can use the services of their "up-stream" service providers (e.g. individual university partners can make use of the services of their NREN). Therefore the initial goal for a DNS service for 6NET is to take care of the operational requirements of the 6NET backbone.

2.1 Requirements

Those requirements are:

- Forward DNS service to provide name-based access to the various components of the network, to support the day-to-day operational and management tasks, as well as fault isolation and repair.
- Reverse DNS as a support function for routine management tasks. It is an essential support service for the network engineers, required to perform fault isolation and repair and configuration change management.

This service provides the translation of numeric IPv4 or IPv6 addresses back to meaningful names for "human consumption", as well as for debugging tools and some (weak) security facilities.

Reverse DNS is already essential for trouble-shooting in the IPv4-based Internet that uses 32bit wide addresses. By convention those addresses are presented as a set of 4 decimal numbers, separated by dots (i.e. 131.130.1.11). It is even *more* essential in an IPv6-based Internet that uses 128bit wide addresses. By convention those addresses are written as a set of 4-digit hexadecimal character groups, separated by colons (i.e. 2001:628:402:0:8000::5).

2.2 Non-Goals

Given the fact that most of the project partners already do manage their own name space (e.g. domains like UniVie.ac.at or ACO.net, cisco.com, DANTE.org.uk,...) and address space (IPv4 and IPv6), no attempt is made to devise a new naming structure or DNS service to replace the existing structures. Rather the DNS service for 6NET is meant to complement the existing services and to extend the services where appropriate.

These boundary conditions limit the functionality of the DNS service for 6NET to the functionality which is actually needed during the implementation and acceptance phase, the early operational phase and the fault isolation and repair scenarios for the 6NET backbone. This includes access links to the partners and equipment connected directly to the backbone.

2.3 Implementation considerations

From an operational point of view, 6NET should "look and feel similar" to and be compatible with the logistics developed for GÉANT, because a considerably big number of persons and entities have to deal with both environments at the same time.

From an implementation point of view, the DNS service for 6NET was available when the new IPv6-based network were installed, configured and accepted. To achieve this goal, the environment which already exists in the NRENs and in GÉANT/DANTE was used to implement the DNS service for 6NET.

From a "corporate identity" point of view for the project (Web-Site, mailing lists, etc.), activities had to be started even before the formal commencement date of 6NET.

3 DNS service implementation

3.1 Forward DNS service for 6NET

3.1.1 Early activities and basic functionality

Towards the end of 2001 SURFNET went ahead and obtained the delegation of a domain name for the project, on behalf of the emerging 6NET Consortium

Actually 2 names were registered: sixnet.org and 6net.org and the basic DNS service for those domains were implemented by SURFNET.

Note that 6net.net had already been delegated to a different entity and is in no way related to 6NET.

The initial configuration for 6net.org and sixnet.org is:

```
$ dig 6net.org. soa
      [ ... ]
;; ANSWER SECTION:
6net.org.      86400  IN  SOA  zesbot.ipv6.surfnet.nl. ipv6.surfnet.nl.

;; AUTHORITY SECTION:
6net.org.      86400  IN      NS      zesbot.ipv6.surfnet.nl.
6net.org.      86400  IN      NS      ns.ipv6.uni-muenster.de.
6net.org.      86400  IN      NS      foo.grnet.gr.
6net.org.      86400  IN      NS      ns3.surfnet.nl.
6net.org.      86400  IN      NS      scsnms.switch.ch.

;; ADDITIONAL SECTION:
foo.grnet.gr.  3035   IN      A       194.177.210.211
foo.grnet.gr.  3035   IN      AAAA    2001:648:0:1000:194:177:210:211
ns3.surfnet.nl. 85835  IN      A       195.169.124.71
ns3.surfnet.nl. 85835  IN      AAAA    2001:610:1:800b:a00:20ff:fe9a:16eb
scsnms.switch.ch. 85835  IN      A       130.59.1.30
scsnms.switch.ch. 85835  IN      A       130.59.10.30
scsnms.switch.ch. 254    IN      AAAA    2001:620::1
zesbot.ipv6.surfnet.nl. 85835  IN      A       192.87.110.60
zesbot.ipv6.surfnet.nl. 85835  IN      AAAA    2001:610:508:110:2a0:c9ff:fedd:67e7
```

```
$ dig sixnet.org. soa
      [ ... ]
;; ANSWER SECTION:
sixnet.org.      86400   IN SOA  zesbot.ipv6.surfnet.nl. ipv6.surfnet.nl.

;; AUTHORITY SECTION:
sixnet.org.      86400   IN      NS      zesbot.ipv6.surfnet.nl.
sixnet.org.      86400   IN      NS      ns.ipv6.uni-muenster.de.
sixnet.org.      86400   IN      NS      foo.grnet.gr.
sixnet.org.      86400   IN      NS      ns3.surfnet.nl.
sixnet.org.      86400   IN      NS      scsnms.switch.ch.

;; ADDITIONAL SECTION:
ns3.surfnet.nl.  81287   IN  A      195.169.124.71
ns3.surfnet.nl.  81287   IN  AAAA   2001:610:1:800b:a00:20ff:fe9a:16eb
zesbot.ipv6.surfnet.nl. 86400   IN  A      192.87.110.60
zesbot.ipv6.surfnet.nl. 86400   IN  AAAA   2001:610:508:110:2a0:c9ff:fedd:67e7
```

While SURFNET has a professional environment in place to provide name services, maintaining both the master and the slave name server within the same operational environment is not the optimal solution.

Thus, according to the Description of Work (DoW), several other NRENS joined in to provide secondary name service (See appendix). Those name servers have to be accessible with IPv4 transport (for compatibility reasons) and IPv6 transport.

In due course GRNET has requested participation in this effort. While it is not necessary or useful to have as many (slave) name servers as NRENS involved, GRNET seems to be a special case, because GRNET's connection to the 6NET core might be implemented as a tunnelled link. Thus it is proposed, at least for the initial phase, to add a machine in GRNET as an additional slave server.

The basic (prototype) service for the domains 6net.org and sixnet.org supports the consortium's website, mailing lists, and - in due course - other infrastructure functions.

3.1.2 Support functions for network operations

Based on the experience gained in TEN-34, TEN-155 and GÉANT, DANTE has developed a formal naming scheme which encodes certain pieces of operational and management information

into the FQDNs which are used to refer to individual systems and/or to individual interfaces on a particular system (router).

In particular, individual components of this naming system encode the country of a particular PoP location, a particular router in a PoP, a particular interface and the link-information to connect to a PoP in a different country. As this system has proven to be very useful for such an environment (see the output of a traceroute command in GÉANT), it has been adopted to also label the components used to implement 6NET.

```
$ traceroute www.dante.org.uk
traceroute to www.dante.org.uk (193.63.211.4), 30 hops max, 38 byte packets
 1 Wien1.ACO.net (192.153.174.1)          0.747 ms  0.203 ms  0.217 ms
 2 aconet.atl.at.geant.net (62.40.103.1)  0.393 ms  0.419 ms  0.377 ms
 3 at.ch1.ch.geant.net (62.40.96.2)      17.373 ms 17.341 ms 17.336 ms
 4 ch.frl.fr.geant.net (62.40.96.30)     26.064 ms 26.041 ms 26.039 ms
 5 fr.uk1.uk.geant.net (62.40.96.90)     33.282 ms 33.303 ms 33.947 ms
 6 janet-gw.uk1.uk.geant.net (62.40.103.150) 33.369 ms 33.238 ms 33.219 ms
   [ ... ]
12 zeta.dante.org.uk (193.63.211.4)     43.012 ms 40.986 ms 41.417 ms
```

This system requires the creation (and delegation) of subdomains in 6net.org to support the proposed naming structure. The well-defined (and well-known) ISO3166 2-letter country-codes are used to denote individual PoP locations.

Many of those subdomains in 6net.org have already been delegated to DANTE to support the development of the naming scheme for 6NET and the planning for the roll-out of the network. This approach allows pre-configuration of entries for those components for which the technical details (and the PoP location) have already been specified (by the end of March 2002).

Here is an example of such an initial delegation:

```
> dig uk.6net.org soa
[...]
;; QUESTION SECTION:
;uk.6net.org.                IN      SOA

;; ANSWER SECTION:
uk.6net.org.                 0       IN      SOA      dns.dante.org.uk.
hostmaster.dante.org.uk.    2003070202 86400 14400 172800 86400
```

;; AUTHORITY SECTION:

uk.6net.org.	86400	IN	NS	sixpack.ipv6.ja.net.
uk.6net.org.	86400	IN	NS	dns.dante.org.uk.
uk.6net.org.	86400	IN	NS	scsnms.switch.ch.

;; ADDITIONAL SECTION:

dns.dante.org.uk.	350175	IN	A	193.63.211.16
scsnms.switch.ch.	17830	IN	A	130.59.10.30
scsnms.switch.ch.	17830	IN	A	130.59.1.30

For a complete list of these domains (which might become delegated and populated in due course) refer to Appendix 1: "List of per PoP-Location Support Domains".

Again, much like for the basic DNS service, other NRENS provide secondary name service for those domains and to make them accessible by IPv6-based transport as soon as possible.

In addition to the "default" secondary name service provision by those partners, all NREN partners are urged to eventually implement secondary name service for their respective xx.6net.org domain (e.g. JANET for uk.6net.org., GARR for it.6net.org. and so on). The reason for this approach is to supply name service in close proximity, and to encourage the deployment of the technology and the dissemination of knowledge.

3.1.3 DNS root name server in 2004

In 2004, few root name sever are accessible in IPv6.

There are:

- B at 2001:478:65::53
- F at 2001:500::1035
- H at 2001:500:1::803f:235
- M at 2001:dc3::35

You can consult <http://www.root-servers.org/> for updated information.

And we can hope to have IPv6 glue in the root file zone in 2004. Thus a DNS IPv6 only service is possible in an early future.

3.2 Reverse DNS service for 6NET

Early experience with managing IPv6-based networks in the framework of 6Bone has proven that reverse DNS in an IPv6-based environment is, in principle, at least as essential and useful as it is for the traditional IPv4-based Internet.

From a software technology perspective, no changes to the name server software in itself are required in order to support the translation of literal IPv6 addresses into name strings. For both protocol families the same basic mechanisms and the same RR type are used: the PTR record. But the rules to convert a literal address to a lookup string are different!

IPv4 uses the decimal encoded external representation of an IPv4 address to build the lookup string, e.g.

Working with

131.130.1.11

Leads to an attempt to find a PTR record for

11.1.130.131.in-addr.arpa.

But IPv6 uses a nibble-based hexadecimal digit encoding which generates a much longer lookup string:

2001:610:508:110:2a0:c9ff:fedd:67e7 becomes

7.e.7.6.d.d.e.f.f.f.9.c.0.a.2.0.0.1.1.0.8.0.5.0.0.1.6.0.1.0.0.2.ip
6.arpa.

However, much like in the forward DNS environment, the same issues apply for the transport protocol(s). For legacy reasons IPv4 has still to be supported as the transport mechanism initially, being extended to allow IPv6 as the transport mechanism as soon as possible.

In reality, obtaining a delegation, properly configuring the name service and using the services for the IPv6 address to name translation is quite a bit more complicated:

- The strings that have to be maintained in the zone files are much longer than those for the IPv4 world (see the previous example);

- As the sTLA allocations made by the RIRs under the "bootstrap procedures" are not aligned on a nibble boundary, class-less delegation mechanisms must be used to properly delegate the reverse zones;

- Initially the sub tree ipv6.int. in the DNS namespace was used to refer to the "reverse DNS for IPv6". Alas, the "int." TLD is reserved for organisations established under an international treaty or multi-national agreement - which is not really appropriate for reverse DNS in the IPv6 based Internet.

Efforts have already been started to move that support function back to the "arpa." TLD - into the ip6.arpa. subtree. This migration has already begun, but it is a complex and lengthy process because the "knowledge" about the subtree in the namespace (required for the generation of the lookup label) is hard-coded into the resolver libraries.

In 2004, the migration should be finished, thus we only consider the ip6.arpa subtree.

```
$ dig ip6.arpa soa
[ ... ]
;; ANSWER SECTION:
ip6.arpa.                0           IN          SOA         dns1.icann.org.
hostmaster.icann.org. 2003080400 3600 1800 604800 10800

;; AUTHORITY SECTION:
ip6.arpa.                172800     IN          NS          ns.ripe.net.
ip6.arpa.                172800     IN          NS          ns.apnic.net.
ip6.arpa.                172800     IN          NS          ns.icann.org.
ip6.arpa.                172800     IN          NS          tinnie.arin.net.

;; ADDITIONAL SECTION:
ns.ripe.net.            101778     IN          A           193.0.0.193
ns.icann.org.           65365      IN          A           192.0.34.126
tinnie.arin.net.       2331       IN          A           63.146.182.189
ns.ripe.net.           110559     IN          AAAA        2001:610:240:0:53::193
```

By the end of March 2002, none of the name servers for ip6.arpa. were accessible with IPv6 as the transport protocol, but at least 3 of the name servers for ip6.int. seem to be IPv6-enabled.

In 2004, there is now one name server of ip6.arpa accessible in IPv6.

For a while implementing both reverse subtrees should be considered. How this is to be done can be deducted from the following example which refers to the RIPE NCC's address aggregate:

```
$ dig 7.0.1.0.0.2.ip6.arpa. soa
      [ ... ]
;; QUESTION SECTION:
;7.0.1.0.0.2.ip6.arpa.      IN      SOA

;; ANSWER SECTION:
7.0.1.0.0.2.ip6.arpa.      0        IN      SOA      ns.ripe.net. ops.ripe.net.
2004020901 43200 7200 1209600 7200

;; AUTHORITY SECTION:
7.0.1.0.0.2.ip6.arpa.      431653   IN      NS       ns.ripe.net.
7.0.1.0.0.2.ip6.arpa.      431653   IN      NS       ns3.nic.fr.
7.0.1.0.0.2.ip6.arpa.      431653   IN      NS       sec1.apnic.net.
7.0.1.0.0.2.ip6.arpa.      431653   IN      NS       sec3.apnic.net.
7.0.1.0.0.2.ip6.arpa.      431653   IN      NS       sunic.sunet.se.
7.0.1.0.0.2.ip6.arpa.      431653   IN      NS       auth03.ns.uu.net.
7.0.1.0.0.2.ip6.arpa.      431653   IN      NS       tinnie.arin.net.
7.0.1.0.0.2.ip6.arpa.      431653   IN      NS       munnari.oz.au.

;; ADDITIONAL SECTION:
ns.ripe.net.                98853    IN      A        193.0.0.193
ns3.nic.fr.                  82089    IN      A        192.134.0.49
sec1.apnic.net.              135878   IN      A        202.12.29.59
sec3.apnic.net.              411      IN      A        202.12.28.140
sunic.sunet.se.              82899    IN      A        192.36.125.2
auth03.ns.uu.net.            51944    IN      A        198.6.1.83
tinnie.arin.net.             10217    IN      A        63.146.182.189
munnari.oz.au.               109009   IN      A        128.250.22.2
munnari.oz.au.               109009   IN      A        128.250.1.21
ns.ripe.net.                  107633   IN      AAAA     2001:610:240:0:53::193
ns3.nic.fr.                   269321   IN      AAAA     2001:660:3006:1::1:1
```

The goal for the early operations phase of 6NET is to strongly encourage all the participants in the 6NET project to get the proper reverse delegations for their sTLAs in place - both in ip6.int. (for supporting older client versions) and in ip6.arpa. (to comply with the more stable solution).

3.3 DNSSEC

DNSSEC is a protocol extension of DNS based on cryptographic tools (keys and signatures). DNSSEC protect the DNS data and transactions. DNSSEC takes full advantage of the hierarchical structure of the DNS: the zones are secured locally, but they can also become a part of a more global scheme, where chains of trust are built to allow a zone to authenticate the keys of its children zones, as well as to allow a zone to be authenticated by its parent zone.

3.3.1 Delay of this activity

The DoW of WP3 anticipated the implementation of DNSSEC in the 6NET DNS infrastructure by the end of year 1. At the time when this schedule was made, the DNSSEC specification was considered to be nearly finished and expected to move through the IETF standardisation process quickly. Later, the DNSEXT working group of the IETF started a major revision of the specification to address a number of scalability and management issues. As a consequence, it was decided to postpone the DNSSEC related activity of WP3 to the end of March 2004, see Contract Amendment 3.

At the time of writing of this second version, the current situation was that the draft specification has stabilizing only recently with a "working group last-call" scheduled for the end of February 2004. The latest publicly available beta-version of the BIND software with support for DNSSEC dates back to December 2002 and does not reflect the numerous changes to the DNSSEC protocol draft of the past year.

Given the fact that DNSSEC has no direct relation to IPv6 and that a major step towards finalizing the specification is to be expected soon, it was found to be most reasonable to postpone this activity once again, producing instead a second version of the Deliverable.

This new version is focused on the planned action for testing DNSSEC when the updated software versions become available, during 2004.

3.3.2 Deployment Plan

DNSSEC will be deployed in 6NET. As a minimum, DNSSEC will be applied to the inverse address mapping of the 6NET address space 2001:0798::/40 in the manner describer bellow. If time permits, this will be extended to the 6net.org zone, which is more interesting as it contains more non-trivial delegations (i.e., delegations to different organizations).

- All authoritative name servers for the zones 8.9.7.0.1.0.0.2.ip6.arpa. and 8.9.7.0.1.0.0.2.ip6.int. and their sub-zones run a DNSSEC-capable name server.

-
- The administrators of the top-level zones of these DNS sub-trees generate at least one key per zone ("zone-signing keys") and sign the zone contents with it. When they are transmitted to all participants in a secure manner, the public parts of these keys establish a "secure entry-point" to those particular sub-trees. This step is necessary as long as DNSSEC is not deployed all the way down from the root of the DNS.

 - The organizations that want to be able to verify the signatures on the resource records of these zones must install DNSSEC-capable caching servers and establish secure communications between the caches and the stub-resolvers. They must obtain and install authenticated copies of the public keys described above to establish the secure entry-points.

 - Each sub-zone must maintain its own set of zone-signing keys and communicate them to their parent zone through an authenticated channel to establish a secure delegation.

The procedures for key-management in the DNSSEC framework have not yet been fully established. This area of work is considered to be out of scope for 6NET. Therefore, only a minimal set of key-management procedures will be established in 6NET, consisting of :

- proper generation of keying material (e.g. use of a decent random number generator)

- transmission of public keys over authenticated channels only (e.g. by using the PGP web-of-trust that has already been established among 6NET participants)

- manual key roll-overs

More sophisticated key-management functions may be implemented if the DNSSEC-specific guidelines become available in time.

Note : there are two types of DNSSEC Key : zone-signing-key (ZSK) that signed all RR of the zone and key-signing key (KSK) that signed only the KEY RR. Only the KSK is transmitted to the parent zone. This permit to do an easier key rollover: you can change the ZSK without change the KSK.

In special case when there is only one Key: it's a ZSK and this Key is transmitted to the parent zone.

3.4 IDNA

IDNA is described in the RFC 3490 (March 2003) and contains a mechanism called Internationalizing Domain Names in Applications (IDNA). With IDNA, applications can use certain ASCII name labels to represent non-ASCII names. IDNA doesn't have an impact on 6NET DNS structure. From RFC3490 : "In particular, IDNA does not depend on any changes to DNS servers, resolvers, or protocol elements, because the ASCII name service provided by the existing DNS is entirely sufficient for IDNA."

Thus IDNA is transparent for the DNS transport.

4 Conclusion

This document describes some of the technology background of DNS (only as far as necessary to understand the requirements in 6NET), briefly points at open issues regarding the state of the art in IPv6-related aspects of DNS and identifies key expectations for the early operational phase of 6NET.

In addition to documenting the current state of the prototype DNS service for 6NET (as configured by the end of February 2004, with a view to support rapid deployment), recommendations for the next development steps were included.

5 Appendix I: List of per PoP-Location Support Domains

Note: The following text is copied from D3.1.1., section 3.1.1 "PoP Naming"


Every PoP has its own subdomain within 6net.org. The subdomain name corresponds to the two letter country code of the country where the PoP is located, i.e., <cc>.6net.org. The country codes are the following:

- at - Austria
- be - Belgium
- ch - Switzerland
- cz - Czech Republic
- de - Germany
- es - Spain
- fr - France
- gr - Greece
- hu - Hungary
- ie - Ireland
- it - Italy
- lu - Luxemburg
- nl - Netherlands
- pl - Poland
- pt - Portugal
- se - Sweden
- si - Slovenia
- sk - Slovakia
- uk - United Kingdom

6 Appendix II: Snap-shot list of systems providing DNS service for 6NET

JANET: uk.6net.org.

Name: sixpack.ipv6.ja.net.

IST-2001-32603	Deliverable D3.2.1v2	
----------------	----------------------	--

Platform: SPARCstation 5 / NetBSD 1.5.2
 NS SW Version: bind 9.2.0
 Network: Ethernet
 Addresses: 128.86.66.6
 2001:0630:0000:0005:0a00:20ff:fe77:e773
 3ffe:2100:0000:0000:0a00:20ff:fe77:e773


GRNET: 6net.org
 sixnet.org
 gr.6net.org

Name: foo.gnet.gr (and foo.gnet6.gr)
 Platform: Sun Ultra 1, UltraSPARC 143MHz, Solaris8
 NS SW Version: BIND 9.2.2
 Addresses: 2001:648:0:1000:194:177:210:211
 194.177.210.211
 Network: Fast Ethernet (FullDuplex)

ACONET: xx.6net.org

Name: nstest.v6.aco.net
 Platform: IBM RS6000 / AIX 4.3.2
 NS SW Version: bind 9.2.0
 Network: Fast Ethernet
 Address: 2001:0628:0402:0001:060:8cff:fe2f:4794
 Network: Fast Ethernet
 Address: 193.171.255.78

Name: sunnysideup.v6.aco.net
 Platform: sparc Ultra-2 / Solaris 8
 NS SW Version: bind 9.2.0
 Network: Fast Ethernet

IST-2001-32603	Deliverable D3.2.1v2	
----------------	----------------------	--

Address: 2001:0628:0402:0001:0a00:20ff:fe86:9b88
Network: ATM
Address: 193.171.25.94


Name: nsipv6.v6.aco.net
Platform: PC / freeBSD 4.4
NS SW Version: bind 9.2.0
Network: Fast Ethernet
Address: 2001:0628:0402:0001:02a0:24ff:fe9d:5094
Network: Fast Ethernet
Address: 131.130.1.201

SURFNET: 6net.org and sixnet.org

Name: NS3.surfnet.nl.
Platform: sparc SUNW,Ultra-5_10 / SunOS 5.8
NS SW Version: bind 9.2.0 (being upgraded to 9.2.1rc2 to 9.2.1)
Network: Fast Ethernet
Addresses: 145.41.1.167
2001:610:100:103:a00:20ff:fe9a:16eb

Name: zesbot.ipv6.surfnet.nl.
Platform: Dell PowerEdge server / freeBSD 4.5-STABLE
NS SW Version: bind 9.2.0 (being upgraded to 9.2.1rc2 to 9.2.1)
Network: Fast Ethernet
Addresses: 192.87.110.60
2001:0610:0508:0110:02a0:c9ff:fedd:67e7

SWITCH: {uk,fr,ch,it,de,nl,at,se,gr}.6net.org.
sixnet.org.
0.0.8.9.7.0.1.0.0.2.ip6.{int,arpa}.

IST-2001-32603	Deliverable D3.2.1v2	
----------------	----------------------	---

Name: scsnms.switch.ch.
Platform: SunFire 280R, SPARC/Solaris 9
NS SW Version: BIND 9.2.3
Network: fast Ethernet
Addresses 130.59.1.30
130.59.10.30
2001:620::1


DANTE: {uk,fr,ch,it,de,nl,se,at,gr}.6net.org.

Name: dns.dante.org.uk
Platform: Sun Solaris 6, will be upgraded soon to Solaris 8
NS SW Version: bind 8.3.1
Network: Fast Ethernet
Address: 193.63.211.19

Name: dns2.dante.org.uk
Platform: Sun Solaris 6, will be upgraded soon to Solaris 8
NS SW Version: bind 8.2.4
Network: Fast Ethernet
Address: 193.63.211.4

GARR: it.6net.org.

Name: 6net.garr.it
Platform: PC / freeBSD 4.4
NS SW Version: bind 9.2.0
Network: Fast Ethernet
Addresses: 193.206.158.6
2001:760::202:a5ff:fee3:ad7b

IST-2001-32603	Deliverable D3.2.1v2	
----------------	----------------------	---

WWU/JOIN: 6net.org
sixnet.org

Name: ns.ipv6.uni-muenster.de (and ns.join.uni-muenster.de)

Platform: PC Pentium III (700MHz) with Debian Linux kernel 2.4.24

NS SW Version: BIND 9.2.3

Adresses: 2001:638:500:101::53
128.176.191.10

Network: FastEthernet (FullDuplex)