

基于 IPv6 的中文域名系统实现

The Chinese DNS and its Implementation Based on IPv6

张京鹏 (中国科学院研究生院 北京 100049)

税国军 (北京邮电大学 北京 100876)

摘要:本文从分析中文域名系统实现需要的基本功能模块出发,将域名系统分为应用层、表示层和寻址定位层 3 层结构。并简单说明了应用层应具备哪些功能,重点讨论了起关键作用的表示层中的中文域名统一编码的 Punycode 编码实现流程,分析了寻址定位层中基于 IPv6 地址下的中文域名正反向解析方法,从而为进一步在这 3 层结构的基础上开展有关 IPv6 下中文域名系统的深入研究打下基础。

关键词:IPv6 中文域名 Punycode 域名解析

1 引言

为解决 IPv4 地址资源枯竭等问题,IPv6 作为一种新的 Internet 协议正在逐步取代 IPv4,成为下一代互联网的基础协议而登上历史舞台。在拥有 12,300 万网民的我国,IPv6 下的中文域名技术的实现,对于我国互联网的进一步发展具有非常重要的现实意义。本文进行了 IPv6 下中文域名系统分层结构、中文域名的实现方法以及 IPv6 地址结构下中文域名解析实现的讨论。

2 IPv6 下的中文域名系统的分层结构

域名系统(DNS)的核心是实现不同寻址资源用户间相互访问的寻址系统。为了建立 IPv6 下的中文域名系统,本文提出了一种分层结构模型,即将域名系统分为 3 层:应用层、表示层和寻址定位层,如图 1 所示。

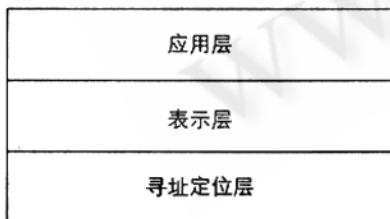


图 1 域名系统中的分层结构

2.1 应用层

域名系统中的应用层,主要包括为了获取地址信息或者资源记录信息而发起查询的客户端。客户端可

以是用户本身或者应用程序。其中,这里所讲的地址信息是指 IPv6 地址信息,资源记录信息是指域名信息。该层主要实现了如下功能。

(1) 为用户提供实用程序或者程序接口,方便用户进行资源定位、IP 地址和域名数据查找。

(2) 当没有找到相应的信息时,向用户回复出错信息。

目前,现有的 IPv4 域名系统中有大量存在于域名系统应用层的实用程序,如 dig, nslookup, 以及可以使用域名信息的其他实用程序,如 ping。相应的,在 IPv6 域名系统中也具备了这样的实用程序,例如, ping6 等。

2.2 表示层

该层主要用于协助应用层交换数据,前提是必须确定数据表现形式。设置该层的目的是为了解决现有域名系统只支持 ASCII 字符集中的 LDH (Letter Digital Hyphen) 问题,该问题使得域名本地化工作难度加大,同时也限制了域名系统的功能拓展。为了与现有的系统最大程度的兼容,以及更方便的实现其他扩展和用户的使用,才增加了表示层来实现该目标。该层主要实现的功能是:

(1) 将用户通过应用层提交的域名信息进行编码得到标准化的数据表示形式 LDH 格式的域名字符串。

(2) 将从寻址定位层得到的 LDH 格式的域名字符串反向解码为用户使用的编码格式。

本层主要研究的是采用何种实现数据的编码方

式,确保编码算法的完整性、输出数据的唯一性、编码算法的可逆性、编码算法的高效性、编码算法实现的简易性、在提高编码效率的同时尽量保证编码的可读性等等。

本层是实现中文域名的关键,目前这一层中最重要的编码方式是 Punycode 编码,后面第 3 部分有详细说明。

2.3 寻址定位层

该层主要用于实现基于某种特定地址结构的地址与相应的资源记录的映射方式。本文中讨论的是在 IPv6 地址结构条件下,地址信息与域名信息的映射、查询等实现。由于 IPv6 地址具有 IPv4 所不具备的许多特性,因此域名系统对此必须能有效支持。本文第 4 部分 IPv6 环境下中文域名系统实现将对此详细讨论。

3 中文域名的 Punycode 编码实现

尽管现有的域名系统依然高效工作着,但由于它是按照 IETF 中 DNS 标准开发的,所以它只支持 ASCII 字符集中的 LDH(Letter Digital Hyphen),无法实现域名国际化的需求,给非英语母语国家的人们上网造成了语言上的障碍。在我国,尽管网民众多,但因为大多数人英语的水平非常有限,这种语言上的障碍更加突出。为了解决这一难题,通过在表示层内对中文域名进行转换,是很好地实现与现有系统的兼容的有效方法。

中文域名编码是 3 层结构模型中表示层的核心,也是中文域名实现的关键。其流程如图 2 所示。

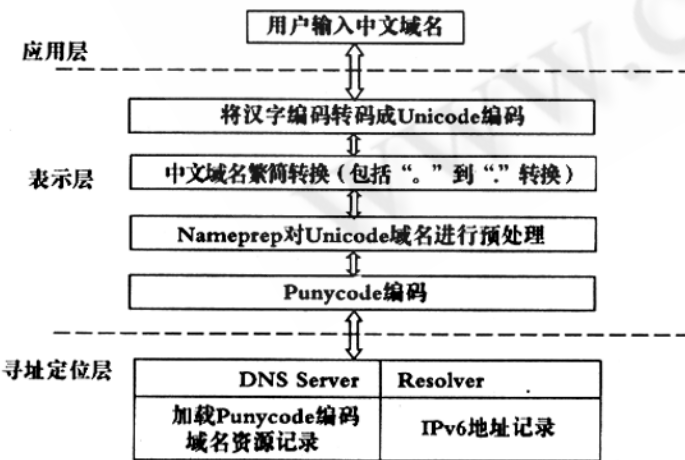


图 2 中文域名编码流程

3.1 将汉字编码转成 Unicode 编码

目前存在着 GB2312 及其扩展 GBK、BIG5 等多种汉字编码方案。如果能让域名系统自动识别 GBK、BIG5、UTF-8 编码的域名是很不现实的。为此,应该在保障中文域名互通性的基础上进行中文域名的表示。可以采用的方案是,将中文域名首先使用 Unicode 进行编码,以避免不同汉字编码体系造成的影响。

将用户通过本地字符集编码输入的中文域名用 Unicode 编码表示。其方式可以分为:

(1) 对本地字符集分别使用适应于本地特点的字符映射规则。例如:对于 GB2312 编码采用 TSconv 映射规则,对于 BIG5 编码采用 STconv 映射规则,对于其他字符集采用 Conv 映射规则。

(2) 使用统一的字符映射规则。例如:GB2312 (BIG5,其他字符集)采用统一的 CharConv 映射规则。

3.2 繁简转换

中文具有繁体和简体之分,且都有巨大的用户群体,因此在中文域名实现的过程中,中文繁简转换成为了中文域名的一个问题^[1]。

目前较为现实的实现方案是采用最小规则的表格查找法。表中的内容含有所用的繁简字——对照信息,以及无歧异的一对多信息。表格的内容主要来自:

(1) Uniban3.11 中的 traditional variant 和 simplified variant,包括一些异体字;

(2) 简化字总表;

(3) 第一批异体字整理表。

其他有歧义的字在繁简转换中不做处理,而是另行处理。

3.3 Unicode 中文域名预处理

为了使用户输入的中文域名符合域名标准,还应当对输入的 Unicode 编码的字符串进行预处理 (Nameprep),去除不符合标准的字符,即进行规范化处理。预处理主要包括 3 个步骤,包括:字符映射 (Mapping)、规范化 (Normalization) 和对禁用字符的禁止输出 (Prohibited Output)^[2]。目前预处理中规定禁止在多语种域名中出现“。”,所以中文域名分隔符中的“。”可以被映射成为“.”。

3.4 Punycode 编码[3]实现

Punycode 编码是 AMC - ACE - Z 编码的简称,即也是一种 ACE (ASCII Compatible Encoding) 编码。

利用它可以将已经预处理过的 Unicode 编码形式的中文域名转化成为 ASCII 编码,同时保证该过程的可逆性,实现了中文 8 bit 编码和 LDH 字符的代码转换。算法简要思想如下:

(1) 将中文域名中的 LDH 字符按照位置顺序移到域名字符串的前面,后面添加一个“-”字符作为 LDH 字符和中文字符的分隔符。并把“-”之后的中文字符按照 Unicode 码的大小递增排序。

(2) 计算排序后的中文字符 Unicode 码之间的差值。

(3) 用差值乘以排序后字符的位置号(该序号是不包含“-”时字符的排序序号),加上该汉字原来所在位置相关的一个参数变量 X,得到 Y。

(4) 量 Y 的值用 36 进制数(用字母 a/A 到 z/Z 表示 0 到 25,用数字 0 到 9 表示 26 到 35)转换成 ACE 编码。依次处理,得到转码后的 ACE 字符串。

(5) CE 字符串前面添加前缀“xn-”得到最终的中文域名 Punycode 编码。经过转换得到只包含 LDH 字符的域名。前缀“xn-”在解码时,用于判断该域名是否实现 Punycode 编码。

经过上述算法,一个 Unicode 编码的中文域名就转换成为 LDH 字符串了。同时,可以根据上述过程的反过程实现对 Punycode 编码的解码。

4 IPv6 环境下中文域名解析

要实现一个中文域名系统,除实现以上两层的功能外,在域名系统的第 3 层还有若干工作必须完成,它们是:(1) 域名空间和资源记录(The Domain Name Space And Resource Records),主要是名字空间的树形结构以及与数据有关的名字和 IP 地址信息。本文中主要指加载 Punycode 编码的中文域名资源记录和 IPv6 地址;(2) 名字服务器(DNS Server),用于控制域的树形结构和固定信息;(3) 解析器(Resolver),用于响应客户端的请求,并从域名服务器中获取信息,如图 2 所示。

IPv6 环境下的中文域名系统在原有 IPv4 域名系统的基础上,通过先将中文域名 Punycode 编码后加载到 IPv6 域名系统中来实现 IPv6 中文域名的解析。

4.1 IPv6 地址特性及域名系统的要求

IPv6 的地址空间提供了多达 2^{128} 个地址,对于域名

系统而言,需要提供解析的记录相比 IPv4 大大增加,但是由于 IPv6 地址结构和地址分配采用了严格的层次结构,对路由的压力反而减少了。

IPv6 地址^[4]长 128 位,使用冒号分隔符为 8 组数字,每组数字用 4 个 16 进制数字表示。例如:0123:4567:89ab:cdef:0123:4567:89ab:cdef,其可聚集全局单播地址结构图如图 3 所示。其中 FP(Format Prefix)是格式前缀,可聚集全局单播地址的格式前缀为二进制 001;TLA ID:Top-Level Aggregator,顶级聚类标示符;RES:保留位,保留为将来使用(默认设置为 0);NLA ID:Next-Level Aggregator,次级聚类标示符;SLA ID:Site-Level Aggregator,地区级聚类标示符;接口 ID:接口标示符。

3	13	8	24	16	64
FP	TLA ID	RES	NLA ID	SLA ID	接口 ID

图 3 IPv6 可聚集全局单播地址结构图

IPv6 地址的层次结构及分配方式包括:顶级注册机构为顶级聚类机构 TLA(如大型的 ISP 或地址管理机构)分配 TLA ID 及大块的 IPv6 地址,TLA 负责为其次级聚类机构 NLA(如中小规模的 ISP)分配 NLA ID 和地址,NLA 为客户分配 SLA ID(SLA 基本上类似于 IPv4 地址中的子网位)和接口 ID(接口 ID 唯一指定了网络上的一个特定接口)。在域名系统中通过地址链技术对这种层次结构进行了有力的支持。下面从域名系统的正向和反向地址解析两方面进行分析。

4.2 IPv6 环境下中文域名 DNS 正向解析

域名系统正向解析是指根据域名信息查询相应的地址信息。IPv4 DNS 的地址正向解析的资源记录是“A”记录。IPv6 DNS 的地址正向解析的资源记录类型有“AAAA”和“A6”两种。

AAAA 记录类型^[5]由 RFC1886 中提出,是对 A 记录的简单扩展,不支持地址的层次性。

例 1:域名 互联网.中国的正向解析的 AAAA 记录。

```
$ ORIGIN xn -- fiQs8S.
xn -- blQ510JgwAINAAAA3fie.3217:4000:1::1
```


(上接第 42 页)

- A String – prep Profile for Internationalized Domain Names” , June 24, 2002, <http://www.ietf.org/internet-drafts/draft-ietf-idn-nameprep-11.txt>.
- 3 Adam M. Costello, “Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA)”, March 2003, <http://www.ietf.org/rfc/rfc3492.txt>.
- 4 R. Hinden, M. O’Dell, S. Deering, “An IPv6 Ag-

gregatable Global Unicast Address Format”, July 1998, <http://www.ietf.org/rfc/rfc2374.txt>.

- 5 S. Thomson, C. Huitema, “DNS Extensions to support IP version 6”, December 1995, <http://www.ietf.org/rfc/rfc1886.txt>.
- 6 M. Crawford, C. Huitema, “DNS Extensions to Support IPv6 Address Aggregation and Renumbering”, July 2000, <http://www.ietf.org/rfc/rfc2874.txt>.