

1 问题描述

根据 RFC6052, 在课程的虚拟机上写一个程序 (用 bash、C、python 等均可), 给定任意 IPv4 地址和 /40 的任意 IPv6 前缀, 输出结果。

2 算法描述

如图1所示, 对于题目要求的问题, 我们只需关注红框所在的行即可。

PL	0	-----	32	--	40	--	48	--	56	--	64	--	72	--	80	--	88	--	96	--	104	-----	
32		prefix		v4(32)		u	suffix																
40		prefix		v4(24)		u (8)	suffix																
48		prefix		v4(16)		u (16)	suffix																
56		prefix		(8)		u	v4(24)			suffix													
64		prefix		u		v4(32)			suffix														
96		prefix				v4(32)																	

Figure 1: IPv4 地址和 IPv6 前缀生成 IPv6 地址示意图

对于给定任意 IPv4 地址和 /40 的任意 IPv6 前缀, 我们采用 Algorithm 1 计算生成的 IPv6 地址。

Algorithm 1 IPv4 地址和 /40 的任意 IPv6 前缀生成 IPv6 地址算法

Input: /40 的任意 IPv6 前缀, IPv4 地址

Output: IPv6 地址

- 1: 初始化 IPv6 地址为 128bit 的全 0 字符串 x
 - 2: $x[0 : 39] = \text{IPv6 前缀 (5 个字节 (40bit))}$
 - 3: $x[40 : 55] = \text{IPv4 地址的前 3 个字节 (24bit)}$
 - 4: $x[64 : 71] = 0$
 - 5: $x[72 : 79] = \text{IPv4 地址的第 4 个字节 (8bit)}$
 - 6: $x[80 : 127]$ 可以任意设置后缀, 在本次实验中全设置为 0
-

3 实验结果

本次实验使用 *python3.7* 完成，由于课程虚拟机上没有安装 *python3.7*，因此首先在 *ee01* 虚拟机上安装了 *anaconda3*。代码文件为 “main.py”，使用如下方式执行：

```
python main.py ipv4_address ipv6_prefix
```

测试了几个转换结果，列表如下：

Table 1: 实验结果统计表

prefix	IPv4 address	IPv6 address
2001:db8:100::/40	192.0.2.33	2001:db8:1c0:2:21::
2402:f000:100::/40	166.111.4.100	2402:f000:1a6:6f04:64::
2001:db8:ab00::/40	192.168.1.101	2001:db8:abc0:a801:65::