

# IPv6 规模部署技术知识及普及

## --IPv6 技术应用于期货行业网上交易

(浙商期货供稿 浙江局指导)

浙商期货有限公司

## 目录

摘要: .....	3
一、引言 .....	3
二、IPv6 过渡背景 .....	4
三、IPv6 的优点 .....	5
四、IPv6 过渡障碍 .....	6
(一)、网络部分设备尚不支持 IPv6, 需要大量更换硬件设备 .....	6
(二)、过渡技术还未广泛应用, 没有通过大量的验证 .....	6
(三)、大多应用基于 IPv4 开发, 不支持 IPv6 .....	7
(四)、网络的升级换代会中断现有的业务 .....	7
五、IPv4 向 IPv6 过渡技术 .....	7
(一)、双协议栈技术 .....	8
(二)、隧道技术 .....	9
(三)、网络地址转换/协议转换技术 .....	10
六、IPv6 地址设置 .....	11
(一)、Windows 系统 IPv6 地址设置 .....	11
1、设置 IPv6 地址 .....	11
2、验证地址连通性 .....	12
(二)、Linux 系统 IPv6 地址设置(以下命令在 CentOS 7.2.1511 下验证通过) .....	12
1、设置(永久生效) .....	12
2、设置(仅临时生效, 网络服务或主机重启后失效) .....	13
七、期货行业网上交易的应用 .....	14
(一)、恒生网上交易登录过程 .....	15

(二)、恒生交易系统新增站点部署 .....	16
(三)、网上交易接入服务器改造 (本实例为 fens,ufx 组件在同一主机上) .....	16
(四)、网上交易客户端部分改造 .....	18
八、结束语 .....	18
九、参考文献 .....	19

**摘要：**随着互联网和物联网的迅猛发展，IP 地址的需求量也随之极速增加，IPv4 地址不足的问题也愈发严重。IPv6 地址空间相当于 IPv4 地址空间的 42.9 亿倍，完全可以满足当下日益增长的地址空间需求，本着充裕的地址空间，IPv6 协议也成为了新一代的网络协议，因此各行各业也面临着互联网协议由 IPv4 向 IPv6 过渡的网络升级问题。基于这样的大背景，本文主要介绍当下 IPv4 向 IPv6 过渡的主要技术，并结合具体案例介绍浙商期货有限公司交易系统由 IPv4 向 IPv6 过渡的具体方案。

**关键词：**IPv6 过渡技术 网络升级 期货

## 一、引言

随着 IPv4 地址空间的逐渐枯竭，IPv6 协议本着充足的地址空间和更快的传输速度等优点，成为了下一代的主流网络协议。但是当下互联网上还是以 IPv4 设备为主，从 IPv4 过渡到 IPv6 就像是“打破一个旧世界，创建一个新世界”，过渡过程中还面临着种种困难，不可能迅速过渡，注定要经历一个长期的过程。本文基于当下 IPv6 过渡的大背景，主要探讨当下 IPv4 向 IPv6 过渡的几个常用过渡技术，分析

介绍各项过渡技术的优缺点,并结合浙商期货有限公司 IPv6 过渡案例,介绍当下金融市场 IPv6 过渡的普遍方案和过渡现状。

本文第二章主要介绍 IPv4 向 IPv6 过渡的背景;第三章基于当前互联网现状说明 IPv4 向 IPv6 过渡的主要障碍;第四章分析 IPv6 相比 IPv4 的优势,以及 IPv6 过渡带来的利好;第五章主要介绍当下互联网由 IPv4 过渡 IPv6 的主要过渡技术,并对这些过渡技术的利弊进行分析;第六章主要结合浙商期货有限公司交易系统 IPv6 过渡案例,介绍当下金融行业 IPv6 过渡的普遍方案及过渡现状。

## 二、IPv6 过渡背景

由于 Internet 的规模以及目前互联网中数量庞大的 IPv4 用户和设备,IPv4 到 IPv6 的过渡不可能一次性实现。而且,目前许多企业和用户的日常工作越来越依赖于 Internet,它们无法容忍在协议过渡过程中出现的问题。所以 IPv4 到 IPv6 的过渡必须是一个循序渐进的过程,在体验 IPv6 带来的好处的同时仍能与网络中其余的 IPv4 用户通信。能否顺利地实现从 IPv4 到 IPv6 的过渡也是 IPv6 能否取得成功的一个重要因素。基于目前的现状,国家结合当下互联网现状先后出台相应文件及通知推动 IPv6 过渡进程稳步推进。

2017 年 11 月,中共中央办公厅、国务院办公厅联合印发了《推进互联网协议第六版(IPv6)规模部署行动计划》,全面规划了建设基于 IPv6 互联网的总体目标,吹响了中国境内部署 IPv6 网络基础设施的号角。

2020 年 3 月,工业和信息化部颁发了《关于开展 2020 年 IPv6 端到端贯通能力提升专项行动的通知》,细化了部署 IPv6 的各项具体要求。

2021年7月，中央网络安全和信息化委员会办公室、国家发展和改革委员会、工业和信息化部此前印发的《关于加快推进互联网协议第六版（IPv6）规模部署和应用工作的通知》对外发布，明确了IPv6规模部署的加速化进程。

由此可见全面构建IPv6网络基础设施的大幕已经拉开，而金融行业是IPv6规模化建设工作中的重要一环，银行、证券、期货行业IPv6网络升级部署工作的顺利实施对全行业IPv6规模化部署工作具有重要的示范推动作用。

### 三、IPv6的优点

相较于IPv4网络体系，IPv6网络技术具有更高的安全性和极大的网络空间，可以有效地应用于自动化、传感器网络和汽车互联等。IPv6由128位二进制数字组成，与由32位二进制数字组成的IPv4地址相比，IPv6能够满足未来物联网、云计算等业务对大量地址需求。除了数量庞大外，IPv6还具有多方面优势：

IPv6的头部取消了校验和字段，且引入了扩展头部这一新的概念，增强了扩展性，头部更精简。

使用128位地址，采用冒号十六进制的标识方法，当出现连续几段数值为0的位段时，这些段可以压缩为双冒号的表示形式，实现精简IPv6地址。

IPv6拥有非常多的地址资源，能为各设备分配固定的IP地址，将完全改变当前用网络地址转换技术转换私网公网地址、动态分配IP地址的局面。

IPv6协议默认采用加密传输的安全机制，使安全运维人员的工作量减少，安全层面的设备与资金投入降低，传输安全性提升。

IPv6 是根据路由条目，路由器进行数据包转发的，使路由器中路由条目的数量极度减少，数据转发的速率提高，从而使网络传输速率更快。

#### 四、IPv6 过渡障碍

随着 Internet 技术的不断发展，当前的 IP 地址空间只能支持很短的时间，IPv4 存在地址空间匮乏、网络安全隐患、不提供服务质量保证、IP 地址配置复杂、缺少移动性支持等问题，为了彻底解决这些问题，必须使用 IPv6 来代替 IPv4，由于 IPv4 与 IPv6 协议不同，两者不能直接同时使用，IPv4 过渡到 IPv6 还有很多问题需要解决，过渡的难点主要有以下几个方面：

##### （一）网络部分设备尚不支持 IPv6，需要大量更换硬件设备

当前的网络环境中，运行 IPv4 的设备投入较大，部分设备还不支持 IPv6 协议。硬件设备的支持是 IPv6 过渡工作的基础和先决条件，要想实现 IPv4 向 IPv6 的转换就需要更换大量老的网络设备来支持 IPv6，然而旧设备的淘汰对大部分公司和单位来说无疑是一种资源浪费，同时硬件设备价格高昂，设备更换会耗费公司大量人力、物力和财力。因此这是阻碍 IPv6 规模化建设步伐的关键因素之一。

##### （二）过渡技术还未广泛应用，没有通过大量的验证

虽然国家一直倡导各公司、单位进行 IPv6 的规模化建设，但是目前在用的 IPv4 网络环境技术成熟、运行稳定，IPv6 的过渡技术还处于初步阶段，技术不够成熟，过渡技术也还未在社会广泛应用，自然也未通过大量的过渡验证。面对过渡技术这样的现状，很多公司和单位都持观望态度，不愿尝试。因此过渡技术未广泛应用也是阻碍 IPv6 规模化建设进度的重要因素之一。

### （三）大多应用基于 IPv4 开发，不支持 IPv6

当前大部分应用都是基于 IPv4 进行开发的，如果网络过渡升级到 IPv6，那么基于 IPv4 的应用需要重新进行开发以支持 IPv6 协议。大部分公司和单位网络环境中的应用数量多，环境复杂，开发难度大，IPv6 过渡后这些公司和单位难以在短时间内完成对应用的开发升级来支持 IPv6，更有甚者升级过程会给公司业务带来不便和不好的影响。因此网络环境中部分应用不支 IPv6 协议也成为 IPv6 过渡的重要障碍之一。

### （四）网络的升级换代会中断现有的业务

当前公司或者单位大部分网络设备和应用都是基于 IPv4 运行的，而且规模庞大，要想实现 IPv6 的过渡需要更换设备和应用，这样难免会对这些公司或者单位造成一定的影响，导致当前应用和业务的终端。这种业务的终端会对公司带来巨大的影响，尤其银行、证券、期货等金融行业，一旦业务中断将带来严重的后果。因此网络升级带来的业务中断也成为众多公司和单位推迟 IPv6 升级的重要原因之一。

由此可见，IPv6 升级改造过程中还面临着不少的困难和问题，要想完全实现 IPv4 向 IPv6 的过渡还需要一个漫长的过程，我们只有不断的努力，克服 IPv6 过渡中面临的问题才能更快的拉动社会群体进行 IPv6 的改造，才能进一步加快 IPv6 规模化建设的步伐。

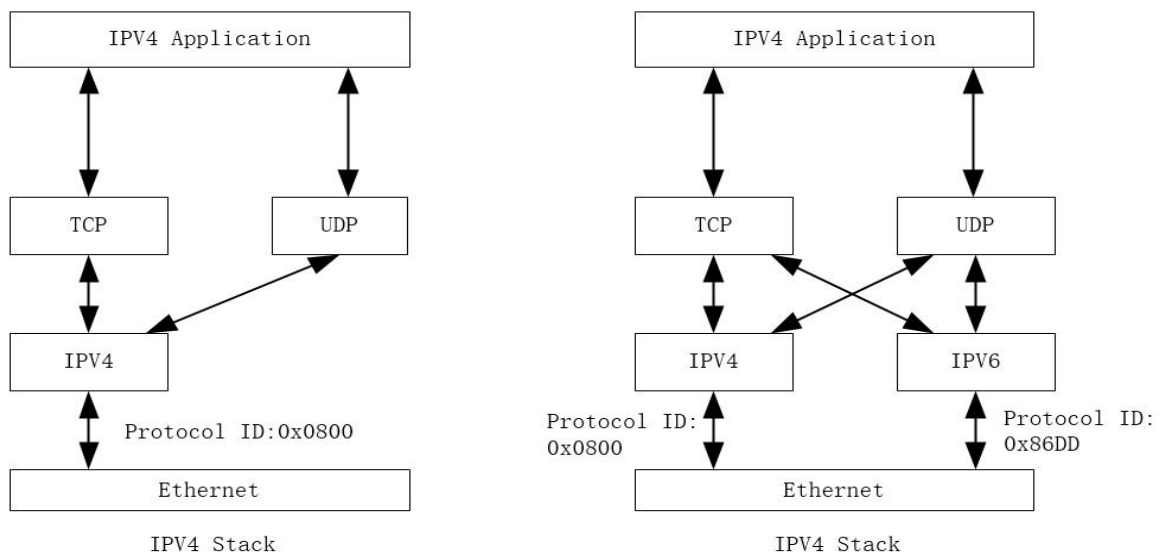
## 五、IPv4 向 IPv6 过渡技术

如何完成从 IPv4 到 IPv6 的转换是 IPv6 发展需要解决的第一个问题。现有的几乎每个网络及其连接设备都支持 IPv4，因此要想一夜间就完成从 IPv4 到 IPv6 的转换是不切实际的。IPv6 必须能够支持和处理 IPv4 体系的遗留问题。可以预见，IPv4 向 IPv6 的过渡需要相当长的时间才能完成。实际上，IPv6 在设计的过程中就已经考虑到了 IPv4

到 IPv6 的过渡问题，并提供了一些特性使过渡过程简化。例如，IPv6 地址可以使用 IPv4 兼容地址，自动由 IPv4 地址产生；也可以在 IPv4 的网络上构建隧道，连接 IPv6 孤岛。目前针对 IPv4-v6 过渡问题已经提出了许多机制，它们的实现原理和应用环境各有侧重，本章我们将针对 IPv4-v6 过渡的基本策略和机制做一个系统性的介绍。

### (一) 双协议栈技术

IPv6 和 IPv4 是功能相近的网络层协议，两者都应用于相同的物理平台，并承载相同的传输层协议 TCP 或 UDP，双栈协议技术就是路由器和交换机内部同时支持 IPv4 协议和 IPv6 协议，这种技术的本质就是实现两种地址在机器内部实现转换，只要网络设备具有双栈性能就可以实现和仅仅具有单一 IP 协议的设备完成通信。



IPv6 与 IPv4 功能几乎相似，而且都是网络层协议，IPv6 与 IPv4 在相同的物理网络平台，遵循相同的网络协议，其中在传输层上的 TCP 和 UDP 协议基本一致，因此双栈技术相比其他技术也是一种易于使用且比较灵活的过渡技术。



双栈技术也有其不足之处，该技术需要运行两个相互独立的协议栈，要求对主机节点的性能较高，同时对主机 CPU 的处理能力和内存大小也具有一定标准。协议转换表必须存储在每个协议栈，双栈技术要为新的 IPv6 协议中运行的网络设备分配 IPv4 地址，这就造成了 IPv4 地址更加不够用。另外 IPv6 和 IPv4 协议有各自的命令行选项集，需要处理各个协议上的不同命令。此外要求双栈主机上所运行的 DNS 解析器必须能够同时解析 IPv4 地址与 IPv6 地址。一般来讲，双栈主机上运行的所有应用程序都必须能决定主机与 IPv4 主机通信还是 IPv6 主机通信。在使用了双栈技术的情况下，需要确保现有的防火墙不仅能保护 IPv4 网络还能保护 IPv6 网络。

## （二）隧道技术

随着 IPv6 网络的发展，出现了许多局部的 IPv6 网络，但是这些 IPv6 网络需要通过 IPv4 骨干网络相连。将这些孤立的 IPv6 局部网络相互联通必须使用隧道技术。利用隧道技术可以通过现有的运行 IPv4 协议的 Internet 骨干网络（即隧道）将局部的 IPv6 网络连接起来，因而是 IPv4 向 IPv6 过渡的初期最易于采用的技术。

路由器将 IPv6 的数据分组封装入 IPv4，IPv4 分组的源地址和目的地址分别是隧道入口和出口的 IPv4 地址。在隧道的出口处，再将 IPv6 分组取出转发给目的站点。隧道技术只要求在隧道的入口和出口处进行修改，对其他部分没有要求，因而非常容易实现。但是隧道技术不能实现 IPv4 主机与 IPv6 主机的直接通信。

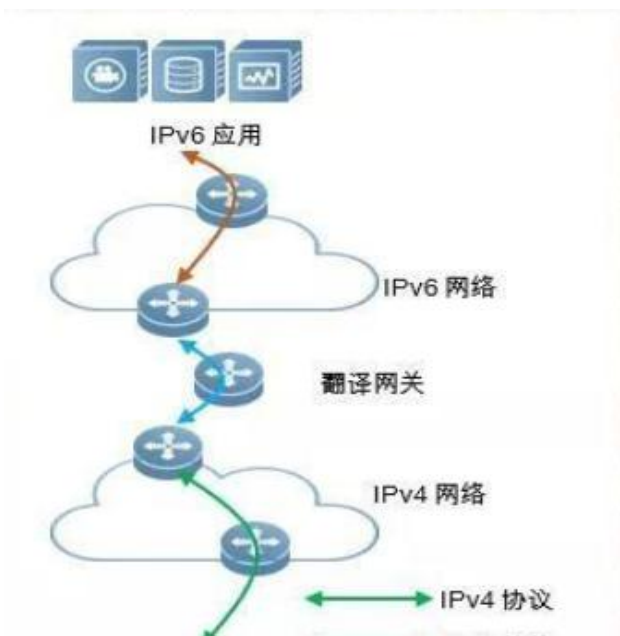


隧道技术的透明性特点可以使两个 IPv6 主机的通信感觉不到隧道的存在，同时，可以减少 IPv6 专用路由器的使用，对网络的构建减少了成本。但隧道技术也有其不足之处，首先，是在 IPv4 网络上对 IPv6 隧道的配置过程，隧道技术在 IPv6 主机与 IPv4 主机之间的通信不能完全实现。同时，在隧道的入口和出口节点处，还可能会有单点故障的可能，故障解决也是比较繁琐。其次，因为存在跳数及 MTU 的限制，所以在隧道的入口和出口节点需要考虑分段问题等。

隧道技术与双栈技术是当前 IPv6 过渡技术中广泛认可的两种技术，同时这两种技术也可以在同一网络中共通存在。

### （三）网络地址转换/协议转换技术

网络转换模式即在不改动现有网络结构的情况下，通过地址转换协议，进行 IPv4 to IPv6 或 IPv6 to IPv4 访问转换，可满足 IPv4 与 IPv6 之间的双向访问需求。技术部署协议包括 NAT 46，一般面向数据中心出口，提供由外到内，由内到外的映射。这种模式对于设备性能要求较高，对高并发业务支撑较困难，适合会话较小的业务。



采用转换技术的优点是对 IPv4 地址的需求少；具有较好的可扩展性；技术对 IPv6 和 IPv4 应用程序透明；数据包传输过程中其他开销少；允许 IPv6 主机与 IPv4 主机直接相互通信。NAT 技术的缺点是不支持 IPv6 高级特性，例如端到端的安全。NAT 技术还面临着设计网络拓扑上的限制，因为应答报文必须来自于被发送的同一 NAT 路由器。NAT 路由器容易形成单点故障，而且无法采用灵活的路由机制。另外，地址和协议翻译需要较大的时间延迟，可能存在有些字段不能转换带来信息的丢失，有时可能会产生碎片数据。

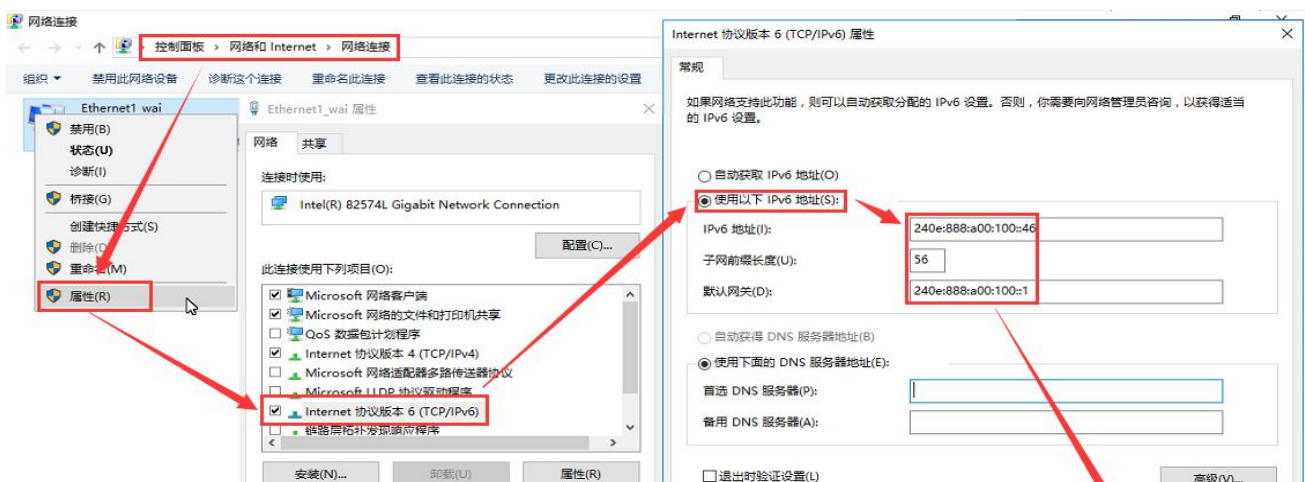
以上三种 IPv6 过渡技术是当前主要的过渡技术，其中双栈技术和隧道技术本着各自的优点成为当前使用最普遍的两种 IPv6 过渡技术。

## 六、IPv6 地址设置

### (一) Windows 系统 IPv6 地址设置

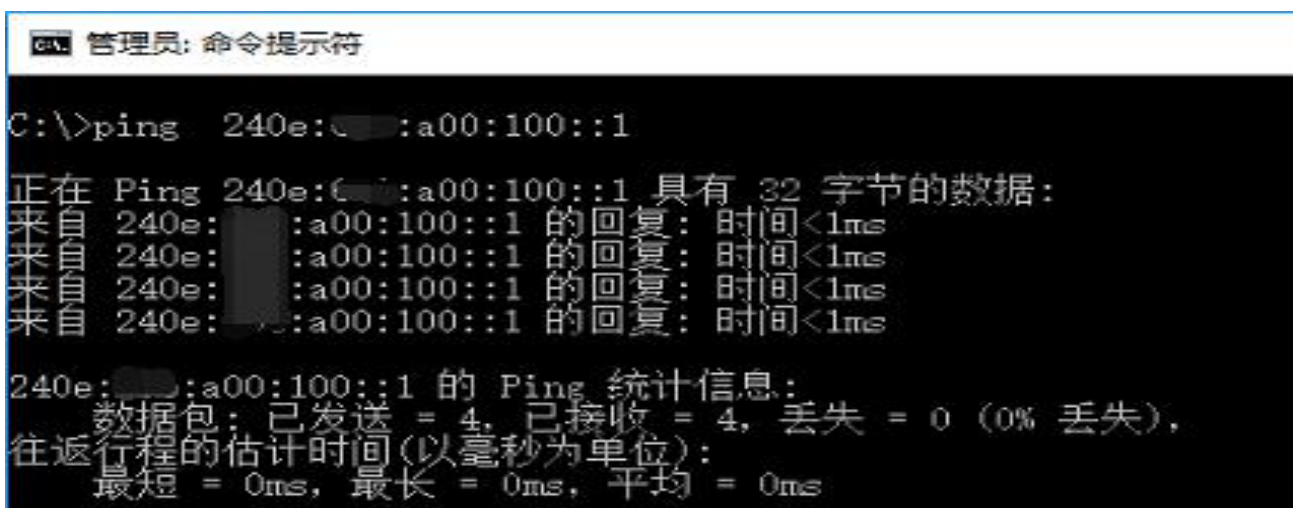
#### 1、设置 IPv6 地址

通过控制面板找到网络和 Internet—网络连接。右键点击本地连接——属性，勾选 Internet 协议版本 6 (TCP/IPv6)，如果没有该选项则表明本设备不支持 IPv6 协议



## 2、验证地址连通性

通过命令窗口，进行 IPv6 地址的网关 PING 测试，如下图所示，表示地址配置成功。



```
管理员: 命令提示符
C:\>ping 240e:6::a00:100::1

正在 Ping 240e:6::a00:100::1 具有 32 字节的数据:
来自 240e:6::a00:100::1 的回复: 时间<1ms
来自 240e:6::a00:100::1 的回复: 时间<1ms
来自 240e:6::a00:100::1 的回复: 时间<1ms
来自 240e:6::a00:100::1 的回复: 时间<1ms

240e:6::a00:100::1 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
    往返行程的估计时间(以毫秒为单位):
        最短 = 0ms, 最长 = 0ms, 平均 = 0ms
```

(二) Linux 系统 IPv6 地址设置(以下命令在 CentOS 7.2.1511 下验证通过)

### 1、设置 (永久生效)



```
[root@localhost ~]# more /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE=eth1
HWADDR=08:00:27:00:00:02a:f2
TYPE=Ethernet
UUID=95b-9a60-ba717f0666c9
ONBOOT=yes
NM_CONTROLLED=no
BOOTPROTO=none
IPV6INIT=yes
USERCTL=no
IPADDR=115.238.255.255
NETMASK=255.255.255.255
GATEWAY=115.238.255.255
IPV6ADDR=240e:6::420::51/56
IPV6_DEFAULTGW=240e:6::420::1
```

IPv6设置新增部分

#systemctl restart network

2、设置（仅临时生效，网络服务或主机重启后失效）

新增 IPv6 地址

# ifconfig eno2 inet6 add 240e:888:a00:100::40

新增 IPv6 路由

# route -A inet6 add default gw 240e:888:a00:100::1

查看 IPv6 路由表

#route -A inet6

#netstat -6 -r

删除 IPv6 地址

# ifconfig eno2 inet6 del 240e:888:a00:100::40

删除 IPv6 路由

# route -A inet6 del default gw 240e:888:420::1 metric

100

查看 IPv6 路由表

#route -A inet6

#netstat -6 -r

```
[root@yishengCS ~]# netstat -6 -r
Kernel IPv6 routing table
Destination                                Next Hop                                Flag Met Ref Use If
[::]/96                                    [::]                                    !n 1024 0 0 lo
0.0.0.0/96                                  [::]                                    !n 1024 0 0 lo
2002:a00::/24                               [::]                                    !n 1024 0 0 lo
2002:7f00::/24                              [::]                                    !n 1024 0 0 lo
2002:a9fe::/32                              [::]                                    !n 1024 0 0 lo
2002:ac10::/28                              [::]                                    !n 1024 0 0 lo
2002:c0a8::/32                              [::]                                    !n 1024 0 0 lo
2002:e000::/19                              [::]                                    !n 1024 0 0 lo
2408:8000:b001:1:7:58ff:f104:102/127 [::]
2408:8640:eff:7::/64                        240e:696:a00:100::1                    UG 1 0 0 eno2
240e:888:a00::/56                           [::]                                    U 100 0 0 eno2
240e:888:a00:100::40/128                   [::]                                    U 256 0 0 eno2
240e:888:a00:100::/56                       [::]                                    U 100 0 30 eno2
3ffe:ffff::/32                             !n 1024 0 0 lo
fe80::/64                                   [::]                                    U 256 0 0 eno3
fe80::/64                                   [::]                                    U 256 0 0 eno4
fe80::/64                                   [::]                                    U 256 0 0 eno1
fe80::/64                                   [::]                                    U 256 0 0 eno2
[::]/0                                       [::]                                    !n -1 1 2324 lo
localhost/128                              [::]                                    Un 0 1 398 lo
2408:8640:eff:7:42a8:f0ff:fe33:a3dd/128 [::]
240e:888:a00:100::40/128                   [::]                                    Un 0 1 44 lo
fe80::42a8:f0ff:fe33:a3dc/128              [::]                                    Un 0 1 0 lo
fe80::42a8:f0ff:fe33:a3dd/128              [::]                                    Un 0 1 66 lo
fe80::42a8:f0ff:fe33:a3de/128              [::]                                    Un 0 1 0 lo
fe80::42a8:f0ff:fe33:a3df/128              [::]                                    Un 0 1 0 lo
ff00::/8                                    [::]                                    U 256 0 0 eno3
ff00::/8                                    [::]                                    U 256 0 0 eno4
ff00::/8                                    [::]                                    U 256 0 0 eno1
ff00::/8                                    [::]                                    U 256 3 0 eno2
[::]/0                                       [::]                                    !n -1 1 2324 lo
```

```

[root@yishengCS ~]# netstat -6 -r
Kernel IPv6 routing table
Destination                                     Next Hop                                     Flag Met Ref Use If
[::]/96                                         [::]                                         !n  1024 0   0 lo
0.0.0.0/96                                     [::]                                         !n  1024 0   0 lo
2002:a00::/24                                  [::]                                         !n  1024 0   0 lo
2002:7f00::/24                                 [::]                                         !n  1024 0   0 lo
2002:a9fe::/32                                 [::]                                         !n  1024 0   0 lo
2002:ac10::/28                                 [::]                                         !n  1024 0   0 lo
2002:c0a8::/32                                 [::]                                         !n  1024 0   0 lo
2002:e000::/19                                 [::]                                         !n  1024 0   0 lo
2408:8000:b001:1:7:58ff:f104:102 [::]    U      100 0   0 eno2
2408:8640:eff:7::/64                          240e:.....a00:100::1 UG     1   0   0 eno2
240e:.....a00::/56                             [::]                                         U     100 0   0 eno2
240e:.....a00:100::40/128                     [::]                                         U     256 0   0 eno2
240e:.....a00:100::/56                        [::]                                         U     100 0   30 eno2
3ffe:ffff::/32                                [::]                                         !n  1024 0   0 lo
fe80::/64                                     [::]                                         U     256 0   0 eno3
fe80::/64                                     [::]                                         U     256 0   0 eno4
fe80::/64                                     [::]                                         U     256 0   0 eno1
fe80::/64                                     [::]                                         U     256 0   0 eno2
[::]/0                                         240e:.....a00:100::1 UG     100 1   0 eno2
[::]/0                                         [::]                                         !n  -1  1  2324 lo
localhost/128                                 [::]                                         Un    0   1  398 lo
2408:8640:eff:7:42a8:f0ff:fe33:a3dd/128 [::]    Un    0   1   0  1   0 lo
240e:.....a00:100::40/128                     [::]                                         Un    0   1   44 lo
fe80::42a8:f0ff:fe33:a3dc/128                [::]                                         Un    0   1   0 lo
fe80::42a8:f0ff:fe33:a3dd/128                [::]                                         Un    0   1   66 lo
fe80::42a8:f0ff:fe33:a3de/128                [::]                                         Un    0   1   0 lo
fe80::42a8:f0ff:fe33:a3df/128                [::]                                         Un    0   1   0 lo
ff00::/8                                       [::]                                         U     256 0   0 eno3
ff00::/8                                       [::]                                         U     256 0   0 eno4
ff00::/8                                       [::]                                         U     256 0   0 eno1
ff00::/8                                       [::]                                         U     256 3   0 eno2
[::]/0                                         [::]                                         !n  -1  1  2324 lo

```

## 七、期货行业网上交易的应用

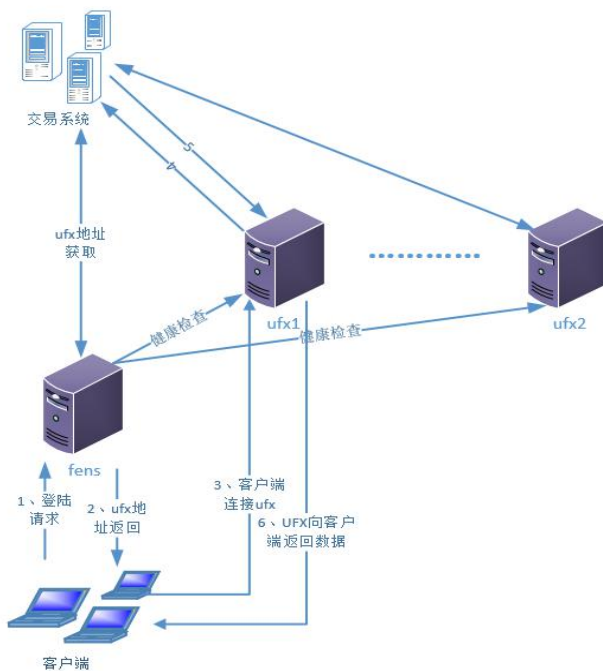
银行、证券、期货等金融行业是全网络 IPv6 过渡过程中重要的一个环节，是 IPv6 网络过渡改造的先行者。金融行业作为各个行业中至关重要的行业，该行业 IPv6 过渡是否成功密切影响着其他行业 IPv6 过渡的信心和步伐，为其他行业 IPv6 过渡起到引导和示范的作用，而期货行业又作为金融行业重要的一部分，该行业 IPv6 过渡进程又密切影响着整个金融行业的 IPv6 升级改造进度。

在监管部门的要求下，期货行业各期货公司已逐步开展了 IPv6 过渡升级工作，而且大部分期货公司已完成了部分业务向 IPv6 过渡的任务。本文下面结合浙商期货有限公司实际案例，介绍浙商期货有限公司 IPv6 过渡的具体实现方案，案例中将展示的是浙商期货有限公司核心交易系统（恒生 2.0 系统）IPv6 改造方案。

浙商期货有限公司 IPv6 过渡采用的是三种常见过渡技术之一的双栈技术，该技术虽然需要运行两个相互独立的协议栈，而且对主机节点的性能要求比较高，但是该套过渡技术只要升级网络等硬件设备具

备支持双栈协议的性能就可以实现和仅仅具有单一 IP 协议的设备完成通信，只要硬件条件满足网络升级要求，该技术相比其他技术来说无疑是一种易于实现且比较灵活的过渡技术。目前公司已实现了网络设备和服务器设备对 IPv6 的支持，且对各交易系统前置和各交易终端进行改造，各交易系统前置实现 IPv4 与 IPv6 共存，同一个站点，客户即可以通过 IPv4 地址登陆交易，也可以通过 IPv6 地址登陆进行交易，这也正体现了双栈技术同时支持两套协议的特性。下面以浙商期货有限公司恒生 2.0 核心交易系统过渡实施案例介绍公司 IPv6 过渡的具体方案。

### (一) 恒生网上交易登录过程



如上图所示，恒生网上交易客户端采用 fens 的方式登录交易系统，客户交易终端向 fens 发出登录请求，fens 检测交易系统前置 ufx 健康状态，并将监测正常的前置地址返回给客户端供客户端连接登陆。

## (二) 恒生交易系统新增站点部署

站点部署信息设置							
AR组名	AR类型	AR状态	系统编号	系统节点名	站点类型	内网地址	外网地址
<input type="checkbox"/> 电信-IPv6	0-交易	0-启用	20000	电信IPv6交易	1-电信	1-:9017	240e-:41:9016
<input type="checkbox"/> 电信1	0-交易	0-启用	20000		1-电信	1-:9017	:9016
<input type="checkbox"/> 电信-IPv6	1-行情	0-启用	20000	电信IPv6行情	1-电信	1-:9047	240e-:41:9046
<input type="checkbox"/> 电信1	1-行情	0-启用	20000		1-电信	1-:9047	:9046
<input type="checkbox"/> 电信-IPv6	3-5.0升级版	0-启用	20000	电信IPv6升级	1-电信	1-:9651	240e-:41:9650
<input type="checkbox"/> 电信1	3-5.0升级版	0-启用	20000		1-电信	1-:9651	:9650

在原有 IPv4 站点基础上，新增一组 IPv6 的站点，该交易站点与上级连接仍采用 IPv4 地址通讯，而对外提供服务的应用地址采用 IPv6，这样最大保证内部网络现有结构和程序应用组件间交互不变化，使得整个改造过程中，平稳和安全，实现对原有客户无感改造。

(三) 网上交易接入服务器改造 (本实例为 fens,ufx 组件在同一主机上)

接入服务器操作系统新增 IPv6 地址

```
[hundsun@UFxfz_fensflt workspace]$ more /etc/sysconfig/network-scripts/ifcfg-ens224
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=static
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=ens224
UUID=0d0b2154-144c-44e4-af58-13c7937e90e0
DEVICE=ens224
ONBOOT=yes
IPADDR=22
NETMASK=2
GATEWAY=2
IPV6ADDR=240e-:41/56
IPV6_DEFAULTGW=240e-:1
```

接入服务器应用组件改造

接入服务器需要升级一些支持 IPv6 的程序文件外，另还需要修改以下配置文件：



## fens 组件配置文件改造

老版本的 fens 收到查询站点的请求，存在 IPv4 地址的客户端，分配 IPv6 地址的风险。升级恒生发布支持 IPv6 的补丁包中 fens.ini 配置文件中，新增 fens 配置项，支持配置站点返回逻辑：!-自动判断类型,如果接入地址是 IPv4 则返回 IPv4,如果是 IPv6 则返回 IPv6。

```
[hundsun@UFxfz_fenslt workspace]$ grep -2 ipv6 fens.ini
#FENS全局配置参数
[fensvr]
#FENS支持的IP类型,!-自动判断类型,如果接入地址是ipv4则返回ipv4,如果是ipv6则返回ipv6,0-只返回ipv4,1-只返回ipv6
ip_type=!
```

fens 站点标准配置文件，在外网 T2 端口配置中增加 IPv6="1" 启动 IPv6；IPv6\_only="0" 兼容 IPv4 和 IPv6；sep\_ip="3" IPv4 地址 IPv4 格式显示，IPv6 地址式是 ip: port 。

```
[hundsun@UFxfz_fenslt workspace]$ grep -4 ipv6 xfens_*.xml
<!-t2#0对外提供服务通道,禁止8功能号,forbidadmin=1限制管理功能接入,其他值不限制-->
<plugin lib="fsc_channel_t2" getinfo="GetT2ChannelInfo" load_level="1" note="T2通道">
  <config>
    <safe safe_level="none"/>
    <network protocol="tcp" address="" port="9030" ipv6="1" ipv6_only="0" sep_ip="3"/>
    <performance thread_count="10" max_connection="10000" init_connection="50" init_recv_buf_size="1024" init_send_buf_size="1024" send_queue_size="10000"/>
    <limit license_dir="license" register_time="5" heartbeat_time="60" check_ordinal="1" forbidadmin="1"/>
  </config>
</plugin>
```

## ufx 组件交易配置文件改造

```
[hundsun@UFxfz_fenslt workspace]$ grep -4 ipv6 ufx_10_UF20_lt.xml
<plugin lib="fsc_channel_t2" getinfo="GetT2ChannelInfo" note="t2通道_公网">
  <config>
    <safe safe_level="none" default_comm_key=""/>
    <ssl cert_file="" cert_pwd=""/>
    <network protocol="tcp" address="" port="9016" ipv6="1" ipv6_only="0" sep_ip="3"/>
    <performance thread_count="10" max_connection="10000" init_connection="50" init_recv_buf_size="1024" init_send_buf_size="1024" send_queue_size="10000"/>
    <limit license_dir="license" register_time="5" heartbeat_time="60" check_ordinal="1" forbidadmin="1"/>
  </config>
</plugin>
```

## ufx 组件行情配置文件改造

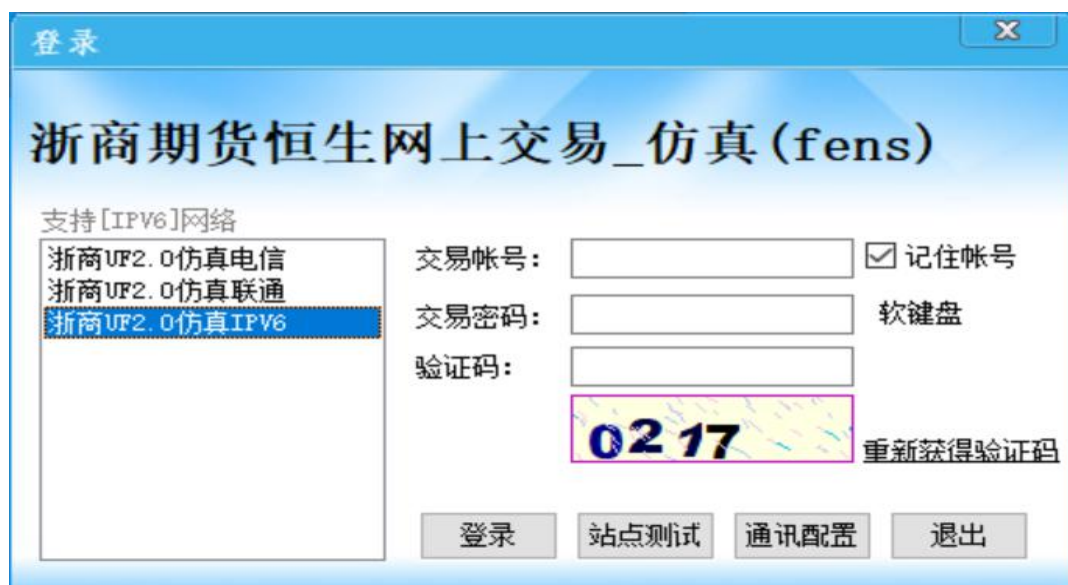
```
[hundsun@UFxfz_fenslt workspace]$ grep -4 ipv6 ufx_quote_UF20_lt.xml
<plugin lib="fsc_channel_t2" getinfo="GetT2ChannelInfo" note="t2通道_外网">
  <config>
    <safe safe_level="none"/>
    <ssl cert_file="" cert_pwd=""/>
    <network protocol="tcp" address="" port="9046" ipv6="1" ipv6_only="0" sep_ip="3"/>
    <performance thread_count="10" max_connection="10000" init_connection="50" init_recv_buf_size="1024" init_send_buf_size="1024" send_queue_size="10000"/>
    <limit license_dir="license" register_time="5" heartbeat_time="60" check_ordinal="1" forbidadmin="1"/>
  </config>
</plugin>
```

## ufx 组件升级配置文件改造

```
[hundsun@UFxfz_fenslt workspace]$ grep -4 ipv6 msg_upgrade.xml
<plugin lib="fsc_channel_t2" getinfo="GetT2ChannelInfo" note="t2通道_公网">
  <config>
    <safe safe_level="none" default_comm_key=""/>
    <ssl cert_file="" cert_pwd=""/>
    <network protocol="tcp" address="" port="9650" ipv6="1" ipv6_only="0" sep_ip="3"/>
    <performance thread_count="10" max_connection="10000" init_connection="50" init_recv_buf_size="1024" init_send_buf_size="1024" send_queue_size="10000"/>
    <limit license_dir="license" register_time="5" heartbeat_time="60" check_ordinal="1" forbidadmin="1"/>
  </config>
</plugin>
```

#### (四) 网上交易客户端部分改造

对现有的网上交易客户端程序，更新支持 IPv6 程序文件，新增名称标识为 IPv6 站点，并进行登录验证。



以上便是浙商期货有限公司恒生 2.0 核心交易系统 IPv6 改造的具体实施细节。首先确保网络及服务器等硬件设备支持双协议运行，在硬件层面支持的前提下对交易系统进行改造，交易系统改造主要是针对前置和交易终端的改造，交易系统前置在保存现有 IPv4 地址的基础上新增 IPv6 地址，交易终端改造增加支持 IPv6 站点，从而实现交易系统对外即支持 IPv4 方式接入又支持 IPv6 方式接入，而交易系统内部保持以 IPv4 协议进行通信。该方案即实现了对外 IPv6 的支持，又保持了系统内部原有的 IPv4 通信，避免了系统内部的大规模变动，维持了稳定性。

#### 八、结束语

IPv6 一旦投入应用，网络中所有的主机和网络设备都必须升级。对于要处理包含成千上万台主机的全球化公司的网络管理者而言，这

无疑是一种短时间内难以完成的挑战。但是，实际情况并非如此，公司完全可以采用外围对外服务部分升级改造支持 IPv6，核心组件间仍然维持现有的 IPv4 协议通信，这样对外服务提供 IPv6 支持的前提下，最大限度的对现有的网络和组件间交互改造影响到最小，应用能持续稳定的运行。

IPv6 本着自身地址空间充足、传输速率高和传输安全性等优点成为下一代核心网络协议已成为不争的事实。我国始终重视并推进全社会 IPv6 网络环境的改造工作，并努力为此投入大量的人力、物力和财力支持。在国家有关部门的支持和引导下，相信不久的将来 IPv6 定将在大数据、物联网、云计算、智能家居等新兴领域中大放光彩。

## 九、参考文献

- [1]李卓群. IPv4 向 IPv6 的过渡技术研究[J]. 2022(3).
- [2]孙中全. IPv4/IPv6 共存网络中过渡技术的实现与研究[J]. 龙岩学院学报, 2020, 38(5):6.
- [3]张鑫, 董刚. 企业 IPv6 过渡综述[J]. 2020, 009(002):34-35.