第33卷 第12期 计 算 机 工 程 2007年6月
Vol.33 No.12 Computer Engineering June 2007
·网络与通信· 文章编号：1000—3428(2007)12—0117—02 文献标识码：A 中图分类号：TP317

源路由控制的研究与实现

王金一，南 凯，陈 琦
(中国科学院计算机网络信息中心网络技术与应用研究室，北京 100080)

摘 要：源路由是由信源向路由器提供路由信息，以控制信息转发的路由方式。科研信息化进程中发现，网络环境中同时存在线路拥塞和闲置，从而认识到源路由控制对于大数据量传输的重要价值，通过源路由控制，可以更有效地利用网络资源，为此提出了源路由控制(SRM)，并在 fedora 下实现了原型系统。给出了 SRM 适用于 IPv4、IPv6 和各种主流平台的解决方案。

关键词：源路由；源路由控制；科研信息化

Research and Implementation of Source Routing Management

WANG Jinyi, NAN Kai, CHEN Qi
(Network Technology and Applications Research Laboratory, Computer Network Information Center,
Chinese Academy of Sciences, Beijing 100080)

【Abstract】 Source Routing is a routing type which depends on the information carried by packet. In the experience of e-science, congestion and idlesse can be found in today's network at the same time, so the management of source routing is important for mass data transmission. With the management of source routing, a more efficient schedule of network resources can be devised. SRM is designed, and many versions are supported to suit with various OS. A demo designed for fedora acheives experiment expectation and is ready for deployment.

【Key words】 Source routing; Source routing management(SRM); E-science

中科院高能物理研究所在与国内外合作单位交换大批数据的过程中，综合考虑数据量、带宽、网络出口以及费用后发现，目前的路由性价比低，希望能针对应用选择路由。为此，中科院网络中心网络技术与应用研究室提出了源路由控制(SRM)，使用户能自主控制路由，又不必修改应用。通过 SRM，用户可以为数据量大、时间不紧迫的应用选择价格低，带宽窄的路由；也可以为时间紧迫、小数据量应用选择大带宽、高可靠性的路由。对于网络运营商，通过 SRM，可以充分利用空闲线路，缓解拥塞线路，合理利用网络资源。

本文提出的 SRM 是一种基于 IP 协议源路由特性进行源路由控制的解决方案，并且针对基于 IPv4 的 TCP 应用设计开发了 fedora 平台下的原型系统，且已经通过实验验证。

1 SRM 原型系统

1.1 系统设计与实现

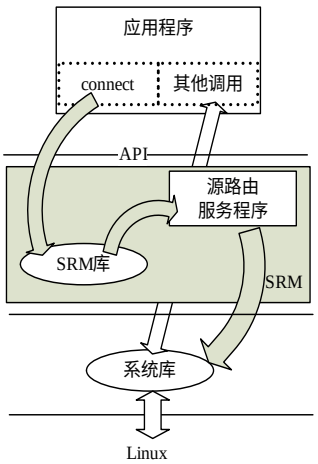

图 1 SRM 原型系统设计

IPv4 协议支持源路由，IP 包头中的选项部分定义了松散源路由和严格源路由两种选项，分别支持路由的松散控制和严格控制。SRM 的设计基于 IPv4 对源路由的支持，通过截获应用对 socket 系列函数中 connect 函数的调用，先使用 setsockopt 函数进行 IP 源路由选项的设置，然后再调用系统库的 connect 函数完成联接，从而提供源路由功能，如图 1 所示。

SRM 包括两部分：源路由服务程序和 SRM 库。SRM 库是动态链接库，只包含一个函数 connect，其原型与系统库函数 connect 完全相同。源路由服务程序是一个独立运行的后台进程，负责设置源路由并调用系统库 connect 函数完成联接。系统库表示所有需要的库函数，这些函数都以动态链接库文件的形式存在，并由系统提供。

应用程序启动后，会同时联接 SRM 库和系统库。通过把应用程序的 LD_PRELOAD 环境变量指向 SRM 库，使 SRM 库中的 connect 函数优先被应用程序调用。SRM 库和源路由服务程序通过 PF_UNIX socket 通信。SRM 库的 connect 函数在被调用后，收集 connect 库函数联接需要的参数信息和需要联接的 socket 的相关信息，然后传递到源路由服务程序。源路由服务程序收到后，先使用系统库 setsockopt 函数设置该 socket 的 IP 源路由选项，然后再调用系统库 connect 函数完成联接，最后把联接是否成功的信息传送回 SRM 库中的

基金项目：国家“973”计划基金资助项目“新一代互联网体系结构理论研究子项目新一代互联网技术综合实验验证及演示平台”(2003CB314807)

作者简介：王金一(1977－)，男，硕士、助研，主研方向：网络监控测量；南 凯，副研究员；陈 琦，硕士生

收稿日期：2006-06-28 **E-mail**：jywang@cnic.cn

connect 函数。SRM 库中的 connect 函数返回到应用程序。

在应用程序运行过程中，除 connect 函数被截获外，其余还是调用系统库中的函数。SRM 库与源路由服务程序的通信必须使用非联接的数据报，因为用于建立联接的系统库 connect 函数已经被 SRM 库自身的 connect 函数覆盖。设计独立的源路由服务程序除了功能合理划分的考虑外，也是由于系统库 connect 函数被覆盖，SRM 中无法联接引起的，因此必须在另一个独立进程中完成这个功能。源路由服务程序启动时必须避免 SRM 库的影响，这里需要使用系统库中的 connect 函数。

SRM 的设计方案中，对每一次联接只需要设置一次源路由选项，对应用程序的影响很小。对应用选择加载或不加载 SRM 库，从而开启或关闭源路由功能。应用需要设置的源路由信息，从配置好的文件中获得。

1.2 实验

SRM 原型系统实验环境如图 2 所示，包括 3 台路由器 R1、R2 和 R3，运行 fedora core 3。路由表配置如表 1~表 3 所示。

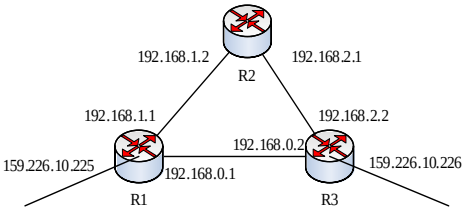

图2 SRM原型系统试验环境

表 1 R1 路由表

目的网段	下一跳
192.168.2.0/24	192.168.0.2
Default	159.226.10.254

表 2 R2 路由表

目的网段	下一跳
192.168.0.0/24	192.168.1.1
Default	192.168.1.1

表 3 R3 路由表

目的网段	下一跳
192.168.1.0/24	192.168.0.1
Default	159.226.10.254

实验中，应用程序采用 fedora 系统的 FTP 程序，R1 模拟源端，R3 模拟宿端，并在 R1、R2 和 R3 上运行 tcpdump 捕获通信包，确定路由情况。实验步骤如下：

(1)R1 上执行“ ftp 192.168.2.2 ”，联接成功，上传下载功能正常。且 R2 上没有包通过，R1、R3 捕获到相应包，路由功能正常。

(2)R1 上运行 SRM，源路由服务程序针对目的为 192.168.2.2 的联接，设置松散源路由信息为 192.168.1.2，即必须通过 R2。

(3)R1 上执行“ ftp 192.168.2.2 ”，ftp 上传下载功能正常。且 R2 上捕获到了相应的 IP 包，R1 和 R3 捕获的包中均含有源路由选项，SRM 功能正常。

2 SRM 解决方案

SRM解决方案基于IP协议[1]对源路由的支持。各种平台的设计，都通过在IP包中增加源路由选项[1]或源路由选项头[2]完成。目前网络基于IPv4 的TCP[3]应用是主流，SRM在各种平台的解决方案也以此为主，同时，对于IPv6 和非TCP应用也给出相应的解决方案。

2.1 Linux 与 Unix 平台

(1)基于 IPv4 的 TCP 应用：第 1 节中针对 fedora 平台已做过详细讨论，核心是截获进行网络联接的 connect 函数。该方案主要基于 socket 接口和系统动态链接库的原理，因此，能够适用于其它 Linux 或 Unix 平台。如需移植，应确保 SRM 库和系统库都能正常工作。

(2)基于IPv4 的非TCP应用[4]：这类应用实现情况复杂，无法找到有效的截获点，不能采用第 1 节中的方案。可基于Linux内核 2.4 版及其以上版本提供的Netfilter实现。编写内核模块，使用Netfilter技术截获IP数据包，并查看包类型。如果是TCP数据包则不做处理，否则在包中增加源路由选项后再发送。有Linux内核的支持，该方案适用于各种Linux平台。对于各种Unix平台，也有对等功能的技术实现，能够支持源路由控制。

2.2 Windows 平台

Windows 平台提供了 SPI(service provider interface)技术，用于扩展和控制 socket 接口。SRM 通过实现 SPI 模块向应用提供源路由支持，如图 3 所示。SPI 提供了一组函数，与 socket 函数一一对应，socket 函数就是通过调用对应的 SPI 函数工作的。SPI 可以叠加，上层 SPI 能够使用下层 SPI 函数。SRM 的 SPI 模块叠加在系统 SPI 模块之上，通过扩展 connect 函数，使其依次调用下层 SPI 的 setsockopt 和 connect 函数，从而实现 TCP 应用的源路由控制功能。对非 TCP 应用，可以扩展 socket 函数，即依次调用下层 SPI 的 socket 和 setsockopt 函数。SPI 是 Windows 系统的组成部分，该方案适合于各种 Windows 平台。

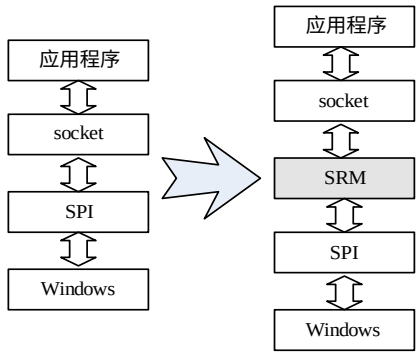

图 3 Windows 平台下的 SRM

上述各种方案同样适用于相应采用 IPv6 协议的情况。只需要按照相应平台对 IPv6 的支持，调整部分参数即可。

3 存在的问题

源路由由 IP 协议直接支持，开发部署容易。但由于缺少源路由应用，以及源路由带来的安全和性能问题，使得目前网络上的路由器基本都禁止此功能，因此，也阻碍了源路由应用的发展。随着网络安全的加强，源路由引入的安全问题也会得到解决。源路由引起的路由器性能下降，是功能增强必然带来的问题，目前关闭也仅是因为没有需要，随着用户对路由控制需求的出现以及硬件的发展，性能不再会是主要问题。

4 相关研究

目前关于源路由的研究还很少，如何实现源路由控制，有如下一些研究项目和成果。1996 年，RFC1940 定义了 SDRP[5](Source Demand Routing Protocol)，为现有网络上实现路由的有限源端控制提供了一种解决方案。但SDRP没有得到发展，相关协议还需要完善。2002 年，加拿大的先进互

(下转第 124 页)