

ForCES 体系结构中 LFB 的研究与实现*

高明, 王伟明

(浙江工商大学 信息与电子工程学院, 浙江 杭州 310035)

摘要: 研究发现, Linux 的内核模块具有良好的可控性而 Linux 的软中断具有极强的实时性。提出把两者有机地结合起来作为逻辑功能块(Logical Functional Block, LFB)的实现机制, 并用流标识号(Pkfid)来描述 LFB。实验表明采用这种方法实现的 LFB 能很好地满足 ForCES 需求(RFC 3654)。

关键词: 路由器; ForCES; LFB; FE Model; Linux 模块

中图法分类号: TP393 文献标识码: A 文章编号: 1001-3695(2006)03-0050-03

Research and Implementation of LFB in ForCES Architecture

GAO Ming, WANG Wei-ming

(School of Electronic Engineering, Zhejiang Gongshang University, Hangzhou Zhejiang 310035, China)

Abstract: It was discovered that Linux kernel module has good capability of control and Linux soft-interrupt has strong capability of real-time. By uniting the two above, a mechanism was proposed to implement Logical Functional Blocks(LFB). Also the method of Packet flow Identifier (Pkfid) was proposed to describe LFBs. In the end, it was proved that LFBs implemented by this way can well meet requirements of ForCES requirement protocol(RFC 3654).

Key words: Router; ForCES; LFB; FE Model; Linux Module

转发和控制分离(Forwarding and Control Element Separation, ForCES) 是 IETF 路由领域(Routing Area)的一个工作组, 它专门研究开放编程的 IP 路由器的体系结构和协议问题, 是当前开放可编程网络研究最受关注的研究组织。ForCES 基本思想是把 IP 路由器分成转发件(Forwarding Elements, FE)和控制件(Control Elements, CE), 认为 IP 路由器可由多个(可达几百个) FE、多个 CE 和连接它们的 ForCES 协议构成。当前 IETF ForCES 工作组已经完成了 ForCES 需求(ForCES Requirements, RFC 3654)^[2]和基本完成了 ForCES 框架(ForCES Framework)^[3]工作, 当前的工作重点是 ForCES 协议和 ForCES FE 模型(FE Model)。

1 ForCES FE 模型和需求分析

ForCES 体系结构和模型主要在 ForCES Requirements (RFC 3654)^[2]和 ForCES Framework^[3]中定义。一个满足 ForCES 标准的网络组件(Network Element, NE)内其核心有至少一个或多个冗余用的控制件 CE, 有可多达几百个的转发件 FE, 它们之间的联系通过 ForCES 协议完成, 这个连接面称为 FP 参考点。FE 内的体系结构主要由 FE 模型定义, FE 基本结构如图 1 所示。其中 FE 内资源模块的描述统一称为 ForCES FE 逻辑功能块 LFB, LFB 及其属性都是可以由 CE 通过 ForCES 协议进行控制的, 各个 LFB 之间通过数据通道(Datapath)

相互连接, 该连接关系也是由 CE 经过 ForCES 协议定义, 以形成不同的 LFB 拓扑结构, 进而实现动态资源配置以完成各种不同的 IP 类型服务。典型的 LFB 如分类器(Classifier)、调度器(Scheduler)、IPv4 或 IPv6 转发器(Forwarder)等。

ForCES 的需求有很多, 这里主要介绍与 LFB 有关的需求, 即 FE 必须知道每个 LFB 能够对数据提供什么样的服务; 支持 LFB 属性参数能动态添加、修改和删除; 支持 LFB 拓扑信息的查询; 支持 LFB 属性信息的查询; 支持 LFB 动态加入和离开系统。

2 流标识号描述法和 LFB 的实现分析

2.1 流标识号描述法

如图 2 所示, LFB 通常能被描述成一个具有 n 个输入、 m 个输出的逻辑功能块。调度 LFB 的功能状态信息如数字 n 和 m 应该由 CE 控制, 同时, 这些 LFB 还具有向 CE 报告自己容量、行为统计数据的能力。CE 对 LFB 的控制是通过一个固定连接来完成的。这里的固定的意思是指 FE Slave 根据 ForCES 协议自动建立的。

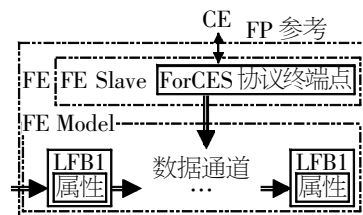


图 1 ForCES FE 基本结构

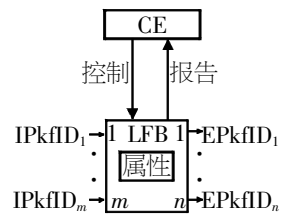


图 2 通用 LFB 模型

收稿日期: 2005-02-28; 修返日期: 2005-04-14

基金项目: 国家“863”计划资助项目(2002AA121064); 国家自然科学基金资助项目(60273061); 浙江省自然科学基金人才专项(RC02063)

一个由 Pkfid 表示的 LFB 它的输入和输出都是由 Pkfid 表示的。很明显上面所示的用 Pkfid 表示的功能块包含了它

的连接信息, 因此要表示一个拓扑结构只需列出所有的 Pkfid 就可以了。任何收到这个信息的接收者通过 Pkfid 表示的连接就可以重建这个拓扑结构。实际上, Pkfid 可以被看成缓存地址, 如图 3 所示。

Object A 与 Object B 通过 Pkfid 为 3 的数据通道相连。数据包进入 Object A 后, Object A 就把它下一步要走的路径固定为数据通道 3; 然后以 3 为索引从 Buffer 中读取属性参数; 再根据属性参数对数据包进行相应的处理; 处理后的数据包经过数据通道 3 进入 Object B。

2.2 LFB 的实现分析

LFB 通常用来对 IP 数据包进行各种各样的处理, 如接收、分类、调度、发送等。在默认情况下, Linux 内核中的协议栈会对数据包进行处理, 为让 FE 中的 LFB 来接管并处理数据包, 就要把数据包从内核中截获下来。由于 Linux 模块^[9]本身就是在内核中运行, 它可以直接调用内核中的函数来实现以上功能, 而且模块可以随意上载、卸载, 符合 FE 中的 LFB 动态加载和卸载的要求, 因此, 目前整个 FE 都是借用模块在内核中实现的。

Linux 的 Tasklet 机制^[9]是一种较为特殊的软中断。Tasklet 一词的原意是“小片任务”的意思, 这里是指一小段可执行的代码, 且通常以函数的形式出现。软中断向量 HI_SOFTIRQ 和 TASKLET_SOFTIRQ 均是用 Tasklet 机制来实现的。在 LFB 的实现中, 除了接收 LFB, 发送 LFB, 其他 LFB 都是用 TASKLET_SOFTIRQ。TASKLET_SOFTIRQ 的使用简单、方便, 且没有被内核专用。使用软中断有以下两方面原因:

软中断的执行是由 Linux 内核来自动触发的, 在 Linux 内核中有四个执行时机, 分别是硬中断执行完后、系统调用处理完后、异常处理完后以及 Ksoftirqd 内核线程, 这就使得 LFB 始终在不停地运转; 执行速度快。

下面给出了一般 LFB 的实现形式:

```

DECLARE_TASKLET(my_tasklet, function, 0); // 定义一个软中断
void function ( int id) { //LFB 的功能实现
    数据包处理
}
int init_module( void) { //模块初始化
    tasklet_schedule( &my_tasklet); //触发软中断
    return 0;
}
void cleanup_module( void) { //模块卸载
    tasklet_kill( &my_tasklet); //删除软中断
}
    
```

3 测试分析

为了验证我们所采用的这种方法所实现的 LFB 能够满足 ForCES 的相关需求, 构造了 FE 和 CE 的软件框架, 编制并测试了以开放可编程的方式动态加载、修改和删除主要类型 LFB 的软件功能。

3.1 测试环境

整个系统由 CE 主机、FE 主机、Hub、SmartBits、分析主机组成。当前我们设置了三个 FE 主机和一个 CE 主机, 每台 FE 主机上安装有内核版本号为 2.4 的 Linux, CE 主机上装有 Windows 2000 系统。CE 主机上的 CE 程序控制 FE 主机上的 FE

程序, FE 程序和 CE 程序的通信通过连接到 Hub, 以 Socket 的形式进行。SmartBits 网络测试仪被用来产生各种速率、各种类型的数据包, 这些数据包通过三个 FE 主机构成的模拟互连网络 (FE 相当于网络上的路由器), 最后又传回到 SmartBits。SmartBits 能对接收到的数据包进行统计分析, 分析结果将显示在分析主机上。

在 CE 端按照图 4 进行每个 FE 拓扑结构的配置。让 SmartBits 的 1-01 口不停地发送数据包, 并逐步提高发送速率。这些包中的目的地址有四种, 分别是 192.168.5.5, 192.168.6.5, 192.168.7.5 和 192.168.8.5, 其发送个数的比例是 1 1 1 1, 包长都是 300Bytes。FE 主机 1 中的路由 LFB 把目的地址为 192.168.5.5 和 192.168.6.5 的包放入 Pkfid 为 1 的流 (流 1) 中, 经过发送 LFB 到达 FE 主机 2; 把目的地址为 192.168.7.5 和 192.168.8.5 的包放入流 2 中, 经过发送 LFB 到达 FE 主机 3。

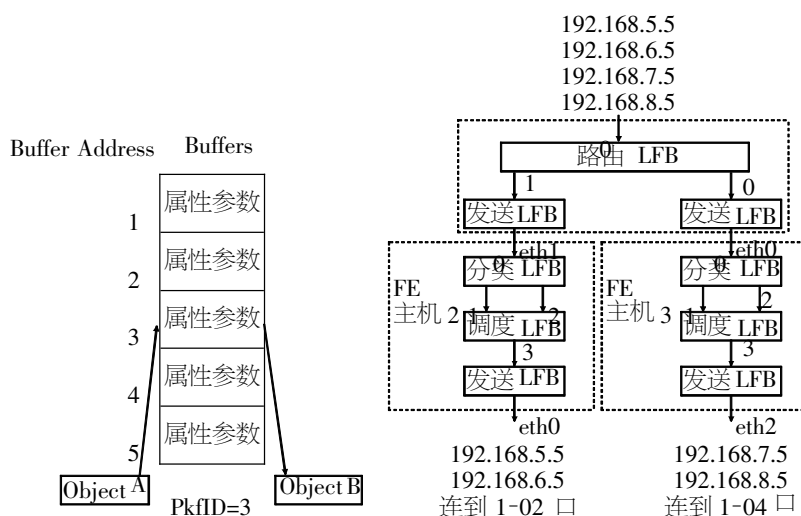


图 3 Pkfid 表示法示意图

图 4 测试试验拓扑结构图

FE 主机 2 中分类 LFB 对包进行分类, 分类规则如下: 目的地址为 192.168.5.5 放入流 1 中, 目的地址为 192.168.6.5 放入流 2 中, 然后调度 LFB 就流 1 和流 2 进行调度, 调度后的数据放入流 3 中。此调度 LFB 相关的参数设置如下: 调度算法为 FCFS (First Come First Serve), 流 1: 最大输入速率 = 10 000packets/s, 爆发时间 = 2s, 缓存区 = 20 000packets; 流 2: 最大输入速率 = 10 000packets/s, 爆发时间 = 2s, 缓存区 = 20 000packets; 流 3: 调度最大输出速率 = 5 000packets/s。

FE 主机 3 中分类 LFB 对包进行分类, 分类规则如下: 目的地址为 192.168.7.5 放入流 1 中, 目的地址为 192.168.8.5 放入流 2 中, 然后调度 LFB 就流 1 和流 2 进行调度, 调度后的数据放入流 3 中。此调度 LFB 相关的参数设置如下: 调度算法为 WRR (Weighted Round Robin), 流 1: 权重 = 0.1, 最大输入速率 = 10 000packets/s, 爆发时间 = 2s, 缓存区 = 20 000packets; 流 2: 权重 = 0.4, 最大输入速率 = 10 000packets/s, 爆发时间 = 2s, 缓存区 = 20 000packets; 流 3: 调度最大输出速率 = 10 000packets/s。

3.2 测试结果

前面的操作构造了一个 ForCES 路由器, 下面通过数据来看这样一个路由器的工作情况。图 5 给出了试验测试结果。方便起见, 用 P21 表示 FE 主机 2 中的流 1, P22 表示 FE 主机 2 中的流 2, P31 表示 FE 主机 3 中的流 1, P32 表示 FE 主机 3 中的流 2。

从图 5 中可以看出, FE 主机 2 上调度器采用 FCFS 算法,

调度最大输出 5 000 包/s, 所以 P21 和 P22 都最大只能得到 2 500 包/s 的服务。FE 主机 3 上的调度器采用 WRR 算法, P31 和 P32 的权重比为 1 4, 调度最大输出 10 000 包/s, 所以 P31 最大的服务速率只能到 2 000 包/s, P32 为 8 000 包/s。由此可见动态加载的各种 LFB 均能正常工作, 从而也证明了采用这种方法实现的 LFB 是完全符合 ForCES 的需求的。

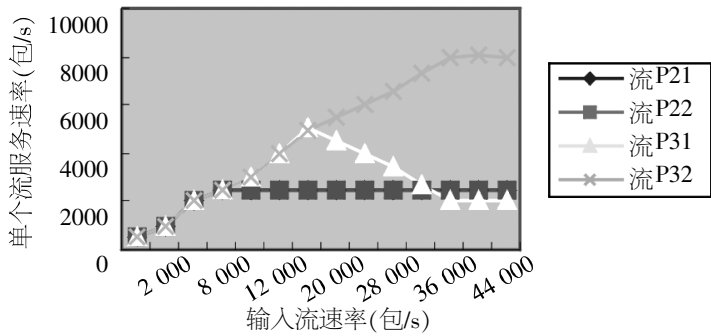


图 5 调度效果图 (包长 300Bytes)

4 结束语

本文所介绍的 LFB 实现机制具有一定的针对性。由于 Linux 开放性, 对于那些从事 ForCES 路由器研究的人员来说, 在研究实验阶段选择它作为开发平台, 能够带来极大的方便并且成本低廉。基于以上原因, 我们提出了基于 Linux 下的 LFB 实现机制。

参考文献:

[1] iswas J, et al. The IEEE P1520 Standards Initiative for Programmable

Network Interfaces[J]. IEEE Communications Magazine, 1998, 36 (10): 64-72.

[2] H Khosravi, T Anderson. Requirements for Separation of IP Control and Forwarding[S]. RFC 3654, 2003.

[3] L Yang, et al. Forwarding and Control Element Separation (ForCES) Framework[EB/OL]. http://www.ietf.org/internet-drafts/draft-ietf-forces-framework-13.txt, 2003-12.

[4] Internet-Draft. General Router Management Protocol (GRMP) version 1[EB/OL]. http://www.ietf.org/internet-drafts/draft-wang-forces-gmp-01.txt, 2003-11.

[5] D Putzolu. ForCES Protocol Evaluation Draft [EB/OL]. http://www.ietf.org/internet-drafts/draft-ietf-forces-evaluation-00.txt, 2003-12.

[6] L Yang, et al. ForCES Forwarding Element Model[EB/OL]. http://www.ietf.org/internet-drafts/draft-ietf-forces-evaluation-00.txt, 2003-10.

[7] China Linux Forum[EB/OL]. http://www.linuxforum.net/forum/postlist.php? Cat = &Board = linuxK.

[8] 李善平. Linux 内核 2.4 源代码分析大全 [M]. 北京: 机械工业出版社, 2002.

[9] 深入 Linux 内核 [M]. 陈莉君. 北京: 中国电力出版社, 2001.

作者简介:

高明(1979-), 男, 安徽淮南人, 硕士研究生, 主要研究方向为网络通信; 王伟明(1964-), 男, 浙江遂昌人, 教授, 博士, 主要研究方向为网络通信。

(上接第 43 页)

图 2 显示了“我生病去看医生。”这句话的消歧情况。对于动词多义词“看”, 从搭配实例结果看出, 词义“看望”、“医治”的值最大; 从语义相关结果看出, 词义“医治”的值最大。所以可以认为“看”的词义是“医治”。

图 3 显示了“我去商店买建筑材料。”这句话的消歧情况。对于名词多义词“材料”, 从搭配实例结果看出, 词义“材料”的值最大; 从语义相关结果看出, 词义“信息”、“材料”的值最大。所以可以认为“材料”的词义是“材料”。

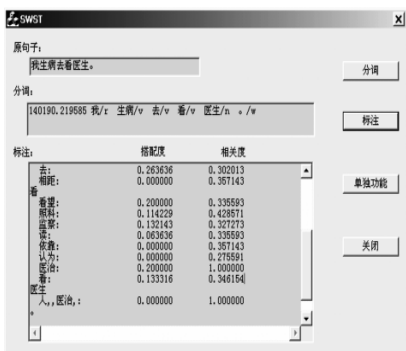


图 2 “我生病去看医生。”的实验结果

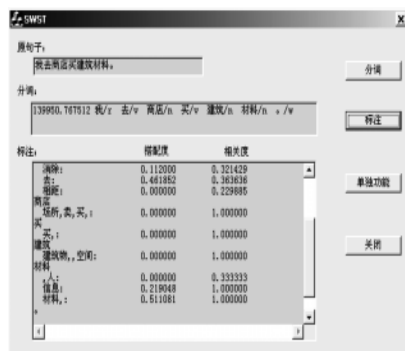


图 3 “我去商店买建筑材料。”的实验结果

5 结束语

本文介绍了基于《知网》的词义消歧的几种算法, 并实现了一个多策略的文本词义标注系统。与一般的词义消歧系统不同, 本系统是基于语义且采用多策略结合的综合消歧方式, 所以提高了消歧的准确度。由于《知网》还处于发展阶段, 许

多功能都在扩充。今后借助它的新功能, 将可以得到更有效的词义消歧算法。

参考文献:

[1] 李涓子. 汉语词义排歧方法研究 [D]. 北京: 清华大学, 1999.

[2] Nancy Ide, Jean Véronis. Introduction to the Special Issue on Word Sense Disambiguation: The State of the Art [J]. Computational Linguistics, 1998, 24(1): 1-40.

[3] 杨尔弘, 张国清, 张永奎. 基于义原同现频率的汉语词义排歧方法 [J]. 计算机研究与发展, 2001, (7): 833-838.

[4] 董振东, 董强. 《知网》 [DB/OL]. http://www.keenage.com, 1999.

[5] David Yarowsky. One Sense Per Collocation [C]. Proc. of ARPA Human Language Technology Workshop, Princeton, 1993. 266-271.

[6] 倪文杰, 张卫国, 冀小军. 现代汉语辞海 [M]. 北京: 人民中国出版社, 1994.

[7] 刘群, 李素建. 基于《知网》的词汇语义相似度计算 [J]. Computational Linguistics and Chinese Language Processing, 2002, 7(2): 59-76.

[8] Wang Chi-Yung. Knowledge-based Sense Pruning Using the HowNet: An Alternative to Word Sense Disambiguation [D]. Hong Kong: Hong Kong University, 2002.

作者简介:

龚永恩(1980-), 男, 硕士研究生, 主要研究方向为知识处理、自然语言理解; 袁春风(1962-), 女, 教授, 主要研究方向为知识处理、多媒体信息处理、自然语言理解等; 武港山(1967-), 男, 副教授, 主要研究方向为智能信息处理等。