**基于click自定义协议软件路由器的**

**设计与实现测试报告**

第32组

# PSP路由转发测试

## 实验拓扑图

Eth2童画

Eth1童画

Docker1

Docker2

Eth2童画

Eth1童画

## 实验测试方法

实验拓扑图如下所示：docker1 从Eth1网口发送数据到docker2 eth2，docker2 上运行着路由器， 从eth2口再次转发给docker2 的Eth2，docker1收到数据后打印消息内容。

## 配置文件

### Client 发送报文配置文件

### Client 接受报文配置文件

图片包含 屏幕截图

描述已自动生成

### Router 转发报文配置文件

图片包含 屏幕截图, 文字

描述已自动生成

## 测试结果

### 客户端发送报文

图片包含 文字

描述已自动生成Router 建立路由表以及转发

图片包含 屏幕截图

描述已自动生成

### Client 接受到数据包并打印

### 图片包含 文字 描述已自动生成

## 结果总结

实验采用通过模拟两台路由器之间的路由转发来测试，从实验结果来看，docker1从eth1发送报文经过docker2路由转发，分别基于psp地址和label转发到docker1的eth2接口。完成了基于psp报文和mpls路由。

## 实验拓扑图

Eth2童画

Eth1童画

Docker1

Docker2

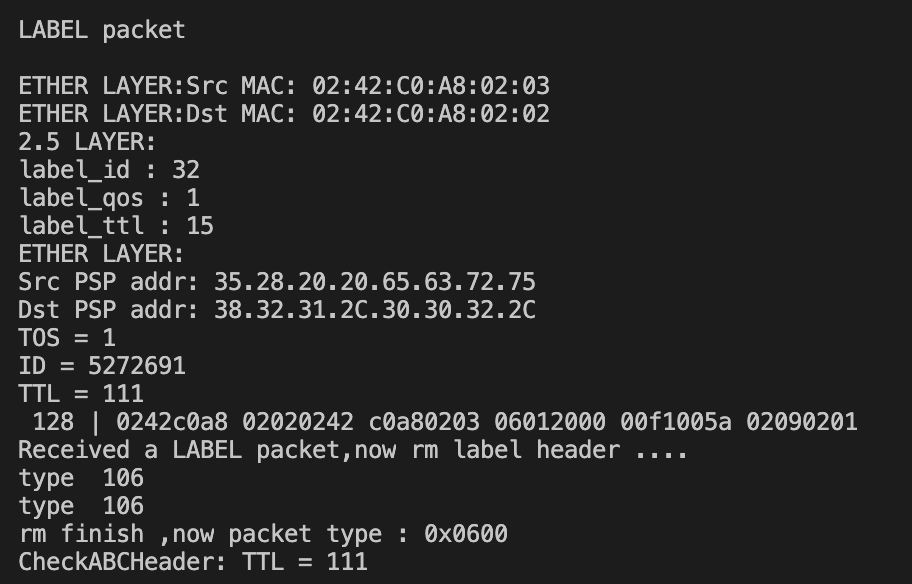
Eth2童画

Eth1童画

## 配置文件



## 测试结果



路由器优先转发稿优先级的报文。

新实验环境：

基于docker的路由协议轻量实验环境

Docker 是一个开源的应用容器引擎，基于 [Go 语言](https://www.runoob.com/go/go-tutorial.html) 并遵从 Apache2.0 协议开源。

Docker 可以让开发者打包他们的应用以及依赖包到一个轻量级、可移植的容器中，然后发布到任何流行的 Linux 机器上，也可以实现虚拟化。

容器是完全使用沙箱机制，相互之间不会有任何接口（类似 iPhone 的 app）,更重要的是容器性能开销极低。

Docker 是一个用于开发，交付和运行应用程序的开放平台。Docker 使您能够将应用程序与基础架构分开，从而可以快速交付软件。借助 Docker，您可以与管理应用程序相同的方式来管理基础架构。通过利用 Docker 的方法来快速交付，测试和部署代码，您可以大大减少编写代码和在生产环境中运行代码之间的延迟。

### 1、快速，一致地交付您的应用程序

Docker 允许开发人员使用您提供的应用程序或服务的本地容器在标准化环境中工作，从而简化了开发的生命周期。

容器非常适合持续集成和持续交付（CI / CD）工作流程，请考虑以下示例方案：

* 您的开发人员在本地编写代码，并使用 Docker 容器与同事共享他们的工作。
* 他们使用 Docker 将其应用程序推送到测试环境中，并执行自动或手动测试。
* 当开发人员发现错误时，他们可以在开发环境中对其进行修复，然后将其重新部署到测试环境中，以进行测试和验证。
* 测试完成后，将修补程序推送给生产环境，就像将更新的镜像推送到生产环境一样简单。

### 2、响应式部署和扩展

Docker 是基于容器的平台，允许高度可移植的工作负载。Docker 容器可以在开发人员的本机上，数据中心的物理或虚拟机上，云服务上或混合环境中运行。

Docker 的可移植性和轻量级的特性，还可以使您轻松地完成动态管理的工作负担，并根据业务需求指示，实时扩展或拆除应用程序和服务。

### 3、在同一硬件上运行更多工作负载

Docker 轻巧快速。它为基于虚拟机管理程序的虚拟机提供了可行、经济、高效的替代方案，因此您可以利用更多的计算能力来实现业务目标。Docker 非常适合于高密度环境以及中小型部署，而您可以用更少的资源做更多的事情。

实验环境部署：

macOS 我们可以使用 Homebrew 来安装 Docker。

Linux下，使用脚本进行安装；

$ curl -fsSL https://get.docker.com -o get-docker.sh

$ sudo sh get-docker.sh

不建议使用window家庭版安装docker。

安装完成后：

* Docker pull 从网络拉取镜像
* Ducker run 【镜像名称】 --name 「容器名称」 -it
* docker network create --driver=bridge 「网络名称」 --subnet=192.168.2.0/24
* Docker network connect 【网络名称】 【容器名称】
* 开始开发

实验镜像已经上传至dockerhub。