**项目需求分析报告**

**项目名称：**

**姓名：**

**学号：**

**学院：**

**报告完成日期：**

1. 引言
   1. 编写目的
   2. 项目背景
   3. 定义
   4. 参考资料
2. 总体要求

2.1总体功能要求

2.2 软件开发平台要求

2.3 运行环境要求

1. 外部接口需求

3.1用户界面

3.2 硬件接口

3.3 软件接口

3.4 通讯接口

1. 系统功能需求

4.1系统各子功能说明

4.2输入/输出数据

1. 项目可行性分析

5.1项目的工作量估计

5.2项目的物质成本估计

5.3关键问题和难点

（包含哪些方面的关键问题和难点，每个关键点计划采用何种关键技术解决）

1. 项目的参与人员

（包括大致的任务分工）

1. 项目进度计划

**项目总体设计报告**

**项目名称：应用层包过滤防火墙的控制功能扩展**

**姓名： 秦瑞泽、王梦奇**

**学号：5130369074、5130369059**

**学院：电院 信息安全**

**报告完成日期：2016.10.15**

1. **引言**

1.1编写目的

该项目总体设计报告的目标读者主要是设计人员，同时也为客户、业务或需求分析人员、测试人员、用户文档编写者开放，让他们明确进一步设计的目标和细节，们希望通过项目总体设计报告对系统的各个功能进行详细的描述，向设计人员说明设计的目标和希望得到的结果。在此设计报告中，我们设计了项目的基本架构，并明确的指出了要完成项目需要设计的所有模块，对应的我们也将模块的功能分解为更小的子模块，以便于实现时查阅。

1.2 适用范围

本报告由项目开发小组根据组内分析、阅读相关资料后撰写成文，根据相关的情况提出了开发该项目所需要的总体设计，并从总体架构、模块分解、接口设计及其他一些方面对本项目进行了一个详细具体形象的总体分析说明。最终交予评审者（老师）予以评审，希望通过此报告清晰地说明我们的项目，并通过此次评审，正式启动本项目的开发。

1.3 术语和缩写

无

* 1. 参考文献

[1] 訾小超.薛质．姚立红.蒋兴浩.潘理.《信息安全技术解析与开发实践》．清华大学出版社

1. **系统概述**
   1. 目的

本项目的目的是基于《信息安全技术解析与开发实践》中已有的包过滤防火墙简单版本，设计一款具有拓展功能的应用层包过滤防火墙。已有的例子中，仅仅实现了单条规则的包过滤防火墙，且只针对源地址、端口以及目的地址、端口进行过滤。我们本次项目的目的主要有以下几方面：

（1）过滤规则拓展。增加对于时间段 、网络接口、ICMP报文子类型的检测。具体而言，可以按时间段开闭防火墙、可以针对不同网络接口设定不同的规则、增添对于ICMP报文子类型的检测。

（2）多过滤规则支持。在本项目中需开发出可同时支持多条规则过滤的防火墙，同时也应设计出对应冲突规则处理方法。

（3）友好的控制界面。即防火墙有有好的UI交互界面，可以进行规则的增删改查，可以进行针对时间段的过滤，可以进行对协议的过滤选择。

* 1. 运行环境

操作系统：

主系统：Microsoft Windows 10 专业版 64位

子系统：SUSE Linux Enterprise 11 SP3（备选Ubuntu 12.04 32bit）

VM：VMware Workstation pro

编译器：gcc 4.6.3

函数库: Libnfnetlink 1.0.0、Libnetfilter\_queue 0.0.17

图形库（尚未确定）：qt4

数据库（尚未确定）：mysql

* 1. 系统范围

本项目的业务为应用层包过滤防火墙，即简单地在应用层使用Netfilter框架实现对于IP包的过滤。

业务的范围为可使用Netfilter框架的Linux内核系统，可布置在主机或者网关处。

1. **系统总体架构**
   1. 平台架构

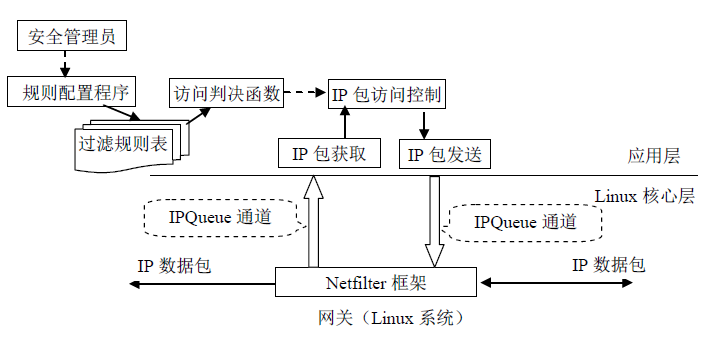


图 1. 应用层包过滤防火墙运行架构

如图1，为应用层包过滤防火墙在机器上的运行架构。Netfilter框架拦截IP报文，并通过IPQueue通道将报文传至应用层，由包过滤防火墙处理后，进行控制。管理员可通过UI界面配置规则，IP报文根据规则裁决控制。

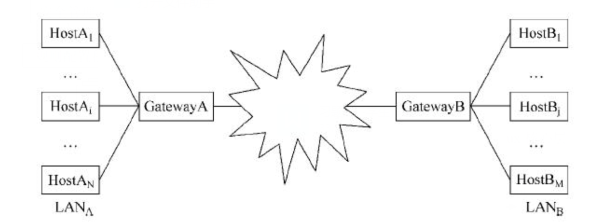


图 2. 网络架构

如图2，无代理服务器的网络结构如图所示。应用层包过滤防火墙必须设置在报文无法绕过的节点，在此种网络架构下适合安装于网关处，以对出入网关的报文进行过滤。当然，对于单个主机，也可安装防火墙，以对流经当个主机的报文进行过滤。

* 1. 总体架构



图 3. 总体架构图

1. **面向用户需求模块**
   1. 模块结构



图 4. 面向用户需求模块

如图4所示，面向用户需求的模块主要用于与用户交互以及处理用户指令。模块主要分为两部分，前端交互模块，以及指令处理模块。前端交互模块方便用户配置防火墙，接收用户指令并传输至指令处理模块，将规则校验后传至数据库模块。

* 1. 前端交互子模块

本模块主要要求为设计合理的UI界面，能够为用户提供舒适的使用体验。可以允许用户在UI界面上进行规则的浏览和控制。本模块主要包含输入模块和阅览模块两大部分。输入模块要求提供IP、端口、协议、时间、网卡的选择，阅览模块要求提供列举规则、选中规则(供删除、修改使用)等功能。

技术实现方面，本模块初步决定采用跨平台C++图形用户界面应用程序开发框架Qt，以方便在Linux内核系统上开发防火墙以及使用Netfilter函数库。（备选Python语言对应的pyqt以及python bindings for libnetfilter\_queue）

逻辑方面，主要需要实现的功能为提供多种规则的输入输出方式:具体而言应包含：

输入：IP、端口的输入框（最好附带有效性检测），可检测协议的选择部件（如下拉条）、时间段选择部件（最好设置标准时间选择部件）、网卡选择部件（下拉条）

输出：预计应使用表格类部件，可查询数据库中所有规则，并提供选中修改或删除的功能。可添加用户提示输出模块，显示程序对用户操作的响应。

* 1. 指令处理模块

本模块主要涉及功能为响应用户的操作指令，并进行校验和执行，以及与数据库进行连接等功能。

本模块主要分为三部分，规则校验模块、规则生成模块以及数据交互模块。三个模块应实现的功能为：

规则校验模块：识别用户输入的参数，并进行有效性校验。若出现问题，按错误类型提升用户修改输出。若符合规则，则发送至规则生成模块进行处理。

规则生成模块：本模块主要功能为，将用户传入的参数，生成合理的对应到的控制规则，并按规定的格式发往数据交互模块。

数据交互模块：本模块的主要功能为处理用户需求处理模块与数据库模块的交互需求。预想的使用场景为：

1. 用户发送查询请求，该模块向数据库请求所有规则数据，并发送至输出模块。
2. 用户需要添加模块，应先校验库中规则与当前规则是否重复，然后向数据库中添加规则。
3. **数据库模块**
   1. 模块结构



图 5. 访问规则控制数据库模块

如图5，为访问控制规则数据库模块，该模块主要包含三个子模块，即默认规则及冲突解决子模块，数据存储子模块，数据交互模块。数据交互模块用于与用户需求模块，判决模块进行数据交互、默认规则及冲突解决模块在存入数据库时设置，对于无规则匹配以及冲突规则制定相应的处理方法。

技术方面，暂决定使用Mysql数据库来储存规则数据。

* 1. 数据交互模块

本模块主要功能为连接用户需求模块以及数据包裁决模块，为其提供数据查询服务。

预计需要实现的功能包含：

查询所有规则、插入数据、删除数据、修改数据等。

* 1. 数据存储模块

本模块主要完成规则的储存工作，将用户需求处理界面发来的规则，以及对于默认规则和冲突解决原则的处理方法。

储存方法应至少包含每条规则所带有的参数信息。即IP、端口、协议、网卡、允许或者禁止。对于时间段的处理可直接由用户需求模块传至裁决模块，暂决定不进行数据库存储。

* 1. 默认规则及冲突解决模块

本模块主要负责设置默认规则以及冲突解决模块，可在用户需求模块中添加前端控件对此模块进行调控。

默认规则即表示对未找到匹配规则的数据包的处理方式，即允许或禁止。

冲突解决可设计为对于所有规则匹配，若对于多条规则均匹配到，若有一条规则禁止则禁止。

1. **数据包判决模块**
   1. 模块结构



图 6. 数据包判决模块

如图6，为数据包判决模块逻辑图。本模块包含三个子模块，即冲突解决模块、解释执行模块以及规则选取模块。三个模块负责的功能分别为：

解释执行模块：获取来自Netfilter传来的IP数据包，获取IP数据包中可以提取到的规则参量值。向规则选取模块询问，获取匹配规则，并对相应结果进行执行。

规则选取模块：从解释执行模块中获取IP规则参量，并向数据库中询问选取，相关规则，选取最终处理结果。

冲突解决模块：多条规则冲突时，选取默认的仲裁方式进行冲突解决。

* 1. 解释执行模块

本模块主要实现两个功能，即IP报文的参数提取以及规则执行。

本项目中，需要抓取的IP参数包括源IP、端口，目的IP、端口、协议类型。

规则执行则主要包括：将报文放行或者阻止。这个功能在例子中已实现。

* 1. 规则选取模块

本模块的主要功能为获取参数，向数据库请求所有相关规则，联合冲突解决模块选择正确的报文处理方式。

1. **运行平台**
   1. 软件平台

主系统: Microsoft Window 10 Pro

VM 软件: VMware Workstation pro

子系统: SUSE Linux Enterprise 11 SP3（备选Ubuntu 12.04 32bit）

编译器：gcc 4.6.3

函数库: Libnfnetlink 1.0.0、Libnetfilter\_queue 0.0.17

图形库（尚未确定）：qt4

数据库（尚未确定）：mysql

* 1. 硬件平台

主机：LENOVO G480

CPU: Intel(R) Core(TM) i5-3230M @ 2.6GHz

系统类型: 64位操作系统，基于x64处理器

1. **接口设计**
   1. 用户接口

在用户需求模块实现，主要包含输入输出接口。

输入接口：用户编辑模块，输入模块要求提供IP、端口、协议、时间、网卡的选择，阅览模块要求提供列举规则、选中规则(供删除、修改使用)等功能。

输出接口：预计应使用表格类部件，可查询数据库中所有规则，并提供选中修改或删除的功能。可添加用户提示输出模块，显示程序对用户操作的响应。除此之外，还需要包含响应指示输出接口，包含对于用户操作的响应以及指示。

* 1. 软件接口

软件接口包括：

1. 防火墙与Mysql数据库的连接。
2. 防火墙与Netfilter的连接
3. 防火墙与Linux的连接
   1. 硬件接口

本产品需要对于对应网卡指定不同的规则。对于硬件接口的要求主要是对于网络链接的要有。可以在以下三种网络硬件接口中人选其一：

1. RJ11口MODEM链接
2. RJ45口以太网链接
3. 802.11b/g无线局域网链接
4. **系统出错处理设计**
   1. 环境设置错误

原因:未按说明正确配置软件运行环境。

提示信息:提示用户哪些环境配置错误。

处理方法:提示并退出。

* 1. 未开启队列机制

原因:未启动Netfilter的队列机制。

提示信息:提示用户，等待启动。

处理方法:提示并等待。

* 1. 无法开启图形界面

原因:未安装qt图形库。

提示信息:提示用户需安装图形库。

处理方法:提示并退出。

* 1. 无法连接数据库

原因:未安装或未开启数据库。

提示信息:提示安装并开启数据库。

处理方法:提示并等待。

* 1. 未获取root权限

原因:为获取root权限，无法正常使用Netfilter\_queue

提示信息:提示无root权限，需要root权限

处理方法:提示并退出

* 1. 防火墙无效

原因:可能为未开启队列机制

提示信息:提示可能的原因，请求用户检测是否开启队列机制

处理方法:提示并退出