硕士学位论文

（工程硕士）

面向企业用户的高性能 VPN 系统的设计与实现

DESIGN AND IMPLEMENTATION OF A HIGH PERFORMANCE VPN SYSTEM FOR ENTERPRISE

孙云霄

哈尔滨工业大学

2015 年 3 月

国内图书分类号：TP311.5　　学校代码：10213
国际图书分类号：004.41　　密级：公开

工程硕士学位论文

面向企业用户的高性能 VPN 系统的设计与实现

硕 士 研 究 生：孙云霄
导　　师：权光日　教授
副　导　师：王树鹏　高级工程师
申 请 学 位：工程硕士
学　　科：软件工程
所 在 单 位：软件学院
答 辩 日 期：2015 年 3 月
授予学位单位：哈尔滨工业大学

Classified Index: TP311.5
U.D.C: 004.41

Dissertation for the Master Degree in Engineering

DESIGN AND IMPLEMENTATION OF A HIGH PERFORMANCE VPN SYSTEM FOR ENTERPRISE

Candidate: Sun Yunxiao
Supervisor: Prof. Quan Guangri
Associate Supervisor: Senior Engineer Wang Shupeng
Academic Degree Applied for: Master of Engineering
Speciality: Software Engineering
Affiliation: School of Software
Date of Defence: March, 2015
Degree-Conferring-Institution: Harbin Institute of Technology

摘 要

VPN 技术在网络游戏、电子商务等企业领域成为保障企业信息安全的重要手段之一。目前主流的 VPN 协议在安全、性能、管理三方面存在着问题，已渐渐无法满足企业日益增长的需求。如 PPTP 协议所使用的 MSCHAP 协议由于密钥长度太短，易受到暴力破解攻击；IPSec 协议在会话协商阶段未对明文数据做校验，易受到中间人欺骗攻击；OpenVPN 协议在实现过程采用了单线程处理，因系统吞吐量低而无法应用于大流量的环境中；主流的 VPN 系统都缺少内容审计功能，使得黑客可以通过 VPN 隧道躲避网络攻击溯源，对网络安全构成了严重的威胁。

本文首先研究了 PPTP、L2TP、IPSec、OpenVPN 等多种主流 VPN 协议以及 N2N、SigmaVPN、VPNGate 等小众的开源协议，在借鉴这些协议优点的基础上，重点研究了 VPN 安全问题、性能问题及管理问题的解决方法。

在安全方面，提高了加密强度，同时对会话协商阶段的明文数据做了 HMAC 数据校验，有效地避免了破解攻击及中间人攻击，保证了协议的安全性；在性能方面，本文将多线程用户态协议栈技术与零拷贝高速数据捕获技术相结合，有效地避免了高并发环境下 Socket 的性能瓶颈，提高了网络的传输性能；在管理方面，基于深度报文检测技术对隧道内部的明文数据做内容审计，保证了数据的安全性。

本文所研究的 VPN 系统通过数据校验、多线程协议栈、深度报文检测等方法解决了 VPN 在使用过程中所面临的问题，还通过 VPN 综合管理系统对 VPN 服务进行管理，有效降低了 VPN 服务器的运维成本，提升了企业的工作效率，可以为电子商务、互联网公司等中小型企业用户提供安全高效的 VPN 服务。

关键词：虚拟专用网；内容审计；高性能 VPN

Abstract

VPN plays an important role in the field of online gaming, e-commerce companies in protecting information. The security risk and performance bottlenecks of traditional VPN protocols can't meet the growing needs of the enterprise users. The mainly problems exists in terms of safety, performance and management. PPTP is easily attacked by brute force attack; IPSec is easily attacked by MITM attack; OpenVPN use single process and the throughput can't match the needs of big flow environment. All of these VPN system are lacking of content audit function, hackers can do anonymous communications via VPN tunnel. The existing problems brings security threats for enterprise.

This paper studies the PPTP, L2TP, IPSec, OpenVPN and some niche open source protocol like N2N, SigmaVPN, VPNGate. In reference to the advantages of these protocols, we focuse on the solutions of VPN security issue, performance issue and management problems.

In security aspects, we upgrade the encryption strength. While on session consultations stage, we verify the plaintext using HMAC to avoid MITM attack; in performance aspects, we combined multi-threading userspace stack and zero-copy data capture technology to solute the performance bottleneck of socket; in management aspects, we use deep paket inspection technology to do content audit for the data in VPN tunnel and ensures the security of information.

Based on data verification, multi-threading stack, deep packet inspection and other technologis,we solve the mainly problems in the filed of VPN application. In addition to guarantee the security of the system, we development a VPN management system to manage the VPN service and monitor the VPN server status. The management system effectively reduced the costs of system operating. The VPN system designed in this paper can provide safe and efficient VPN service for medium and small-sized enterprise like e-commerce, Internet companies.

Keywords: VPN, content audit, high performance vpn

目 录

第1章 绪　论

1.1 研究背景

本课题来源于哈尔滨工业大学（威海）网络技术研究所在研项目，依照企业实际需求设计。本课题所研究的系统主要目的是为企业提供安全且高效的 VPN（虚拟专用网络，Virtual Private Network）服务，系统可为机密数据的传输提供加密保护，同时支持 VPN 内部路由机制、包过滤机制、抗旁路攻击、抗 DoS(Denial of Service)攻击等重要功能，实现了适合中小型企业用户的 VPN 系统。

1.2 研究目的及研究意义

网络技术的发展使得它在越来越多的行业成为不可或缺的一部分，在金融、电子商务等企业领域中，网络在企业日常工作中扮演着重要的角色，网络为企业员工之间的交流带来了方便，与此同时，对网络的依赖也为企业的信息安全带来了威胁。

在网络的日常应用中，由于技术漏洞、管理缺陷等原因而造成的网络安全问题都在呈上升趋势。2014 年 4 月，著名开源加密库 OpenSSL 爆出“Heartbleed”安全漏洞[1]，该漏洞可能导致受攻击服务器上所存储的证书私钥、用户密码、用户注册邮箱等多项敏感信息泄露。由于 OpenSSL 的应用十分广泛，该漏洞影响重大。2014 年 5 月，小米论坛后台数据库的 800 万用户信息遭泄露[2]，这是既 2011 年 CSDN、人人网、新浪微博等多个网站数据库泄露[3]之后发生的又一次重大的数据泄露事件。

随着企业的信息化建设，企业内部的文档、表格等数据多存储于计算机上，当企业内部的财务报表、销售数据、技术方案等企业机密和私有数据通过网络进行传输时，很有可能被黑客通过窃听、欺骗等恶意攻击手段获取，进而导致企业机密信息的泄露，甚至威胁到企业的核心利益，对企业的发展造成了严重影响。

VPN 技术在很大程度上满足了人们对网络安全的需求，它通过使用网络隧道、加密解密、密钥管理、身份认证等多项信息安全领域的关键技术来保证用户信息的安全性，在移动办公、企业间远程信息交换等应用场景下发挥了重要的作用。因此研究并实现面向企业用户的高性能 VPN 系统对于保证企

业的信息安全有着重要意义。一方面，对于已经应用 VPN 的企业来说，新的高性能 VPN 系统可以降低企业在网络安全方面的成本，减少对 VPN 系统运维工作所投入的人力物力资源。另一方面，对于还未应用 VPN 的企业而言，一个简单易用的 VPN 系统可以使企业以较低的成本来提升自己的网络安全，更加有效地保证企业的信息不被非法篡改或泄露。

1.3 研究现状

1.3.1 相关技术的发展现状

目前主要流行的 VPN 协议有 PPTP(Point-to-Point Tunneling Protocol)[4]、L2TP(Layer Two Tunneling Protocol)[5]、IPSec(IP Security)[6]和 SSTP(Secure Socket Tunneling Protocol)[7]等。本文首先将对数据处理技术、密钥交换技术以及身份认证技术等 VPN 核心技术做分析和总结。

1) 数据处理技术现状

VPN 对隧道内部传输数据的处理主要包括加密和压缩。PPTP 使用 MPPE(Microsoft Point-to-Point Encryption, 微软点对点加密协议)[8]作为加密算法，IPsec 支持的对称密钥加密算法有 DES、3DES、IDEA 、AES[9]。L2TP 采用 MPPC(Microsoft Point-to-Point Compression, 微软点对点压缩协议)[10]压缩算法，OpenVPN 采用的是 LZO 压缩算法[11]。在传统的 VPN 实现中，隧道不会对数据压缩的必要性做区分，即对所有的数据都进行压缩，然而隧道中传输的数据经常是已经压缩过的数据，重复压缩在某些时候甚至会导致数据冗余，刘亚琼[12]等人提出了一种选择性压缩算法，基于数据报中的数字特征来判定数据是否有冗余，从而实现了选择性压缩。该算法使 VPN 隧道的压缩效率和传输性能取得了令人满意的效果。在保证数据完整性方面，OpenVPN 协议使用 HMAC 做校验[13]，这样就使得数据在传输过程中不会被黑客通过非法手段篡改，有效地保证了数据的安全。

2) 密钥交换技术现状

VPN 的密钥协商技术是加密模块的重要部分，一般的做法是在会话协商阶段采用非对称加密，在数据传输阶段采用对称加密，这样对称加密的密钥如何传输或协商就成为保证 VPN 隧道数据机密性的关键。通常客户端与服务器会在握手阶段通过非对称加密算法协商出一个安全的加密通道，然后再通过这个安全的加密通道来协商对称加密算法所使用的密钥。

IPSec 协议族中先后定义了 IKE[14]、ISAKMP[15]和 IKEv2[16]等多种密钥

交换协议。IKE 协议中使用的是 Diffie-Hellman[17]密钥交换协议来实现第一个对称密钥的协商。PPTP 的密钥协商的过程是先生成预主密钥，再生成主密钥，再由主密钥生成会话密钥，每个方向又分为发送密钥和接收密钥。

在非对称加密领域经常用到椭圆曲线来实现密钥协商。椭圆曲线算法在加密强度和执行速度上较其它公钥加密系统来说具有更大的优势，同时它需要的密钥长度也更小[18]，基于椭圆曲线实现的动态密钥共享机制在 VPN 技术的实现中得到了比较广泛的应用[19]。

3) 身份认证技术现状

VPN 系统通过身份认证来鉴定发起请求的用户是否合法，从而进一步判断是否允许进入。传统的身份认证方法有密码认证、证书认证等。PPTP 协议和 L2TP 协议曾先后使用过 PAP[20] (英文全称为：Password Authentication Protocol，中文为：密码认证协议)、CHAP[21] (英文全称为：Challenge Handshake Authentication Protocol，中文为：挑战应答握手协议)和 MSCHAP[22] (英文全称为：Microsoft Challenge Handshake Authentication Protocol，中文为：微软挑战握手应答协议)三种认证协议，然而这些协议都是基于密码的，当用户密码发生泄漏时，系统就无法将合法用户和窃取密码的黑客进行区分。赵铭伟[23]等人提出了基于动态口令身份认证协议，在深入分析了传统的 CHAP 动态口令身份认证方案的基础上，结合安全的散列函数和异或运算，同时引入了保护认证信息的干扰因子。这种改进的 CHAP 一次性口令双向认证协议具有通信量小，灵活性高，安全性强等特点，适合中型电子商务网站的身份认证。郝玉洁[24]等人将指纹识别引入到了 VPN 的身份认证中，使身份认证的准确性有了较大的提高。

1.3.2 VPN 应用的发展趋势

除了经过 IETF(Internet Engineering Task Force, 国际互联网工程任务组)规定的标准协议外，很多开源组织和个人设计并实现了自己的 VPN 协议，包括开源的 OpenVPN 协议、日本筑波大学的 VPNGate 等。在涵盖 VPN 的基本功能的同时，这些 VPN 协议往往有不同的侧重点。

VPNGate 与传统的 VPN 的实现相比有很大的改进，它不仅可以兼容主流的 PPTP、IPSec、OpenVPN 等多种 VPN 协议，而且还开发了自己的私有 VPN 协议，VPNGate 的私有协议侧重于匿名通信，它通过数据流指纹隐藏技术使得 VPN 的数据包负载中没有明显的特征，使防火墙的监测手段失效，提高了匿名通信能力[25]。

传统的 VPN 是基于 CS 架构的，一般是单点接入的。N2N[26]、sigmaVPN、ShadowVPN 等开源 VPN 协议实现了分布式 VPN。分布式 VPN 也是 VPN 技术的研究热点之一，文献[27]提出了一种适用于中小型企业的分布式 VPN 架构，接入到 VPN 网络中的节点之间可以直接通信，不需要经过服务器的中转，在一定程度上提升了 VPN 的传输性能。

VPN 技术一般是用来建立远程连接，构建私有局域网。近年来 VPN 在匿名通信领域的应用有所发展，在通过 VPN 隧道进行匿名通信时，客户端接入到 VPN 服务器后，将所有网络流量发送到服务端，有服务端通过 NAT(Network Address Translation, 网络地址转换)技术对客户端发来的数据做转发，从而在客户端与服务器之间建立起了加密通信。这种应用场景一般适应在咖啡厅、飞机场等公开网络环境中，可以提高信息的安全性。但与此同时，一些黑客和不法分子也利用这种手段来躲避网络审查机制或者躲避网络攻击溯源。为了保证内部信息不被泄露，同时避免不良信息的扩散，需要有内容审计的 VPN 来提供数据审核机制。

另外，随着云服务、CDN(Content Delivery Network, 内容分发网络)等技术的发展。越来越多的企业开始构建自己的私有云，云集群中的节点可能分布在全国甚至世界各地。这些节点之间的通信需要由 VPN 来提供数据的安全。

1.3.3 目前存在的问题

1) 针对协议漏洞的攻击

由于某些协议在设计之初并没有充分考虑安全性，同时由于历史原因，一些出现比较早的协议在当今时代已经不再安全。

文献[28]研究了针对 PPTP 密码的字典攻击，而 David Hulton[29]等人的研究结果表明，利用专门设计的 FPGA 阵列，可以在 24 小时内破击任意 PPTP 用户的密码，做到 100%的破解率，只要攻击者获取到 PPTP 用户在登录过程中的挑战握手报文，就可以轻松地破解出用户密码，甚至可以还原出 PPTP 隧道内部所传输的明文信息。

由于 IKE 协议和 IKEV2 协议没有对数据包做完整性校验，在握手阶段都会受到中间人攻击，攻击者可以通过伪造服务器响应报文来实现阻断 IKE 握手通信的目的[30]。

VPN 协议的安全性成为威胁 VPN 系统信息安全的重要问题之一。在设计和实现 VPN 协议的时候，应该充分研究并分析现有的协议漏洞，通过有效

的技术手段来避免 VPN 协议遭受攻击。同时也要考虑到日后硬件计算性能的发展速度，尽量使用长度大的密钥，以减小被暴力破解的概率。

2) 针对系统配置的攻击

文献[31]研究了针对 SSL VPN 的中间人攻击手段，攻击者通过伪造服务端证书的方式串行地接入到网络中，可以获得加密密钥等系统关键参数，进而获得隧道内部传输的明文数据。

部分 VPN 协议为了保持向下兼容性，同时支持多种协议版本，这使得系统易遭受降级攻击(Downgrade Attack)。如果 PPTP 的认证协议配置不当，没有在服务器端禁用 PAP 和 CHAP 认证协议，攻击者就可以伪造客户端响应报文，强制客户端与服务器之间使用 PAP 协议进行身份认证，而 PAP 协议是把密码以明文的方式发送到服务器的，这样位于中间的黑客就可以嗅探到 PPTP 的密码[32]。

3) 缺少内容审计功能

目前的 VPN 系统都不具有内容审计的功能，即网络的系统管理员无法通过传统的内容审计手段来确保机密信息不被外泄，也无法及时控制不良信息通过 VPN 隧道进行传播，同时 VPN 在匿名通信方面的应用也为网络取证、攻击溯源等工作带来了挑战[33]。

4) 单线程成为性能瓶颈

随着用户对网络服务质量要求的提高，VPN 的性能也受到广泛关注。在一些对网络延迟要求严格的应用场景中，当网络出现拥塞时，VPN 往往会断开连接，那么 VPN 的重复协商以及链路修复所做的操作就会对运行于 VPN 之上的业务产生很大的影响。

文献[34]对 IPSec VPN 和 SSL VPN 的性能进行了测试，文献[35]对 MPLS VPN 的性能进行了测试，测试结果都证明加密算法的效率影响着 VPN 的性能。秦培斌[36]等人研究了基于多核处理器的加密卡异步并行驱动，使用加密卡来提升加密解密的速率，实现了针对 IPSec 的加速。

OpenVPN 技术在网络接入[37]，远程控制[38]，虚拟计算[39]等领域得到了广泛的应用，然而 OpenVPN 因其单线程的实现方式存在着严重的性能问题。

随着硬件制造工艺的发展，多核 CPU 处理器的无论在运算速度还是在功耗、散热等方面都有了非常大的进步，网络设备中多核 CPU 的应用也变得非常普遍，然而由于历史原因，PPTP、IPSec、OpenVPN 等传统的 VPN 协议在设计之初都是基于单线程的，这样就无法充分发挥多核 CPU 的优势，浪费了硬件设备的计算资源。

1.4 本文研究内容及组织结构

本文首先通过对国内外 VPN 系统的发展现状和相关产品、应用进行调查和分析，找出了目前主流 VPN 系统存在的问题，针对这些问题进行了深入研究并提出了解决方法，最后设计并实现了适合企业用户的 VPN 系统。本文主要内容如下：

第 1 章绪论部分主要对课题的来源、研究目的以及课题的研究意义进行了介绍，对 VPN 系统相关技术的发展现状进行了调查和总结，同时还分析了目前主流 VPN 协议存在的问题，最后介绍了本课题的研究内容和文章的组织结构。

第 2 章高性能 VPN 系统的需求分析部分，首先论述了需求分析的目的和意义，接着从业务、功能、性能等方面论述了高性能 VPN 系统的需求。在需求分析的基础上对 VPN 系统进行了概要设计，结合本项目的需求选择合理的处理架构，然后给出了高性能 VPN 系统的模块构成。并对模块进行简单介绍，为下一步的详细设计打下基础。

第 3 章高性能 VPN 系统的概要设计部分，基于需求分析对系统的整体功能架构进行了设计，同时也对系统使用的关系型数据库的表结构进行了设计，并对涉及到的库表功能进行了详细的介绍。

第 4 章高性能 VPN 系统的详细设计与实现部分，完成了对概要设计阶段的各个模块进行详细的设计与实现。主要包括数据处理模块、加密解密模块、运维管理模块、内容审计模块等模块的实现。同时还创新性地实现了 VPN 系统的多线程处理方式。

第 5 章高性能 VPN 系统的测试部分，我们针对 VPN 系统的各个主要模块进行了测试，包括加密解密模块、身份认证模块、内容审计模块等。最后我们还对集成后的 VPN 系统做了整体测试，证明了本文研究内容的可行性。

第2章 高性能VPN系统的需求分析

本章首先完成了高性能 VPN 系统的需求分析，介绍了业务涉及到的角色以及常见的业务流程，通过对用户基本需求、系统的功能需求及系统的非功能性需求等方面的深入调查及分析，对系统的需求分析进行了详细阐述。通过本章的分析，参与到系统中的人员可以根据系统的功能设计和业务处理流程来判断该系统是否可以满足需求；技术开发人员也可以通过该需求分析来更深入地了解系统，掌握系统研发过程中所需要重点注意的问题。

2.1 系统需求

本文的主要目标是为企业提供高性能的 VPN 服务，除了实现 VPN 系统的基本功能模块外，本文还将研究通过多线程技术来提高程序对多核 CPU 的利用率，并研究通过用户态网络协议栈技术来提升系统的传输性能，以达到优秀的网络传输速度，达到较高的性能水准。

根据客户端与服务器的位置不同，VPN 系统在实际应用中大致可以分为两种应用场景。

1)端到安全网关：进行 VPN 数据通信的双方中，一方是 PC，另一方是安全网关，如图 2-1 所示。

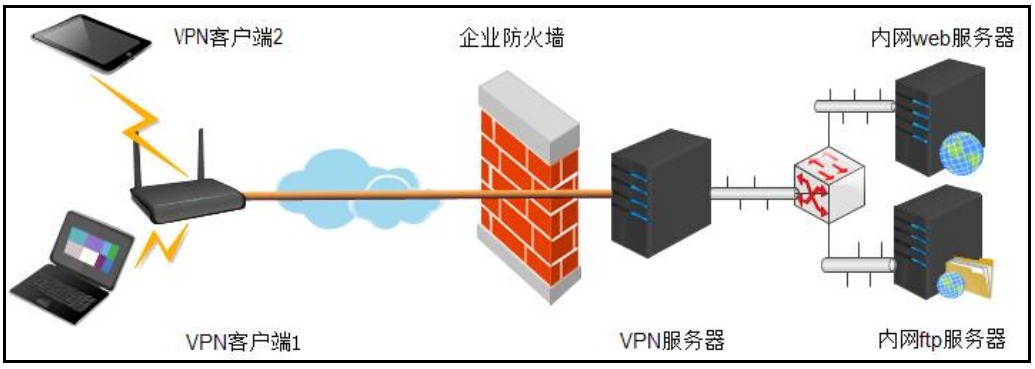


图 2-1 端到网关网络拓扑示意图

安全网关负责保护企业内部的 NAS 服务器、开发服务器、内部 FTP、打印机等设备的安全。当企业网络外部的 PC 想要访问企业内部网络中的服务器或主机时，实际上是通过位于企业网络出口的安全网关的转发来实现的。终端用户发送过来的数据请求是通过建立在 PC 和安全网关之间的安全的 VPN 隧道传输的，由安全网关对数据进行解封装，并转发到企业内网中相应

的主机或设备。其中安全网关的处理对于外网的 PC 主机和内网的主机而言是完全透明的，通信的两个主机可以使用普通的方式进行通信，不需要在终端使用特殊的软件。该场景一般可以满足移动办公人员的需求。

2)网关到网关：进行 VPN 数据通信的双方均是网关，如图 2-2 所示。

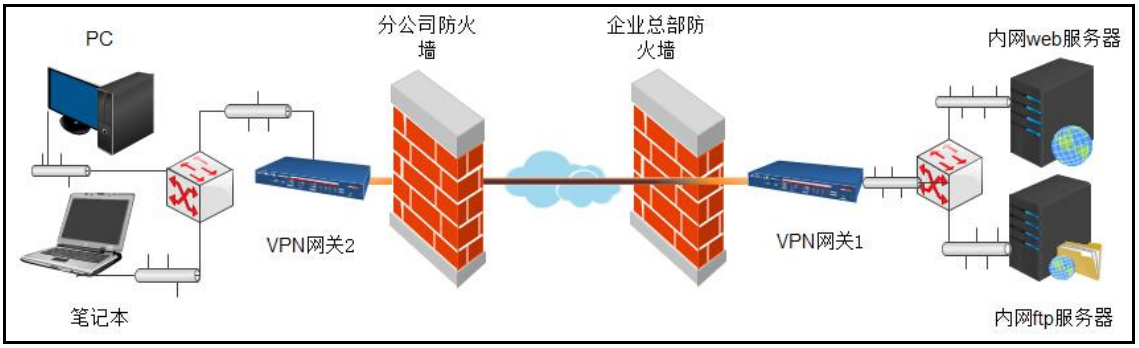


图 2-2 网关到网关网络拓扑示意图

在双方的通信过程中，终端 PC 或服务器并不对传输的数据做加密、封装等处理，这些处理是由位于两个企业内网出口的安全网关完成的。该应用场景一般适用于企业分支结构或企业与大型合作者之间的网络互联，因为在这种情况下，网络中的终端比较多，如果选择在终端对数据进行加密处理，则会增加网络的复杂程度，使得网络效率变低，同时也不利于网络安全及网络审计部门的管理。网关到网关的应用实现了对大量的 VPN 流量集中管理，是一种更优的解决方案。

2.2 用户需求

整个系统需要分为三部分，包括 VPN 客户端、VPN 服务器以及 VPN 综合管理系统。VPN 客户端供终端用户使用；VPN 服务器用来提供 VPN 服务，部署于服务器或网关之上；VPN 综合管理系统供系统管理员、运维人员和终端用户共同使用，但不同类型用户所涉及的功能及权限有所不同。

1) 实体定义

系统使用中相关的实体包括业务人员、系统管理员、运维人员等角色。具体定义如下：

终端用户：VPN 系统的使用者，一般是出差的异地办公人员和在家办公的软件开发人员；

管理员：VPN 系统的管理人员，用户管理等，用户的添加、删除，用户信息的修改等操作都需要系统管理员来执行，系统管理员拥有系统最高权限。

运维人员：主要负责设备的管理，包括设备性能及网络性能的查看与维护，系统数据的定期备份，设备的安全检查等工作。

2) 用例示意图

系统的参与者与系统的主要功能如图 2-3 所示：

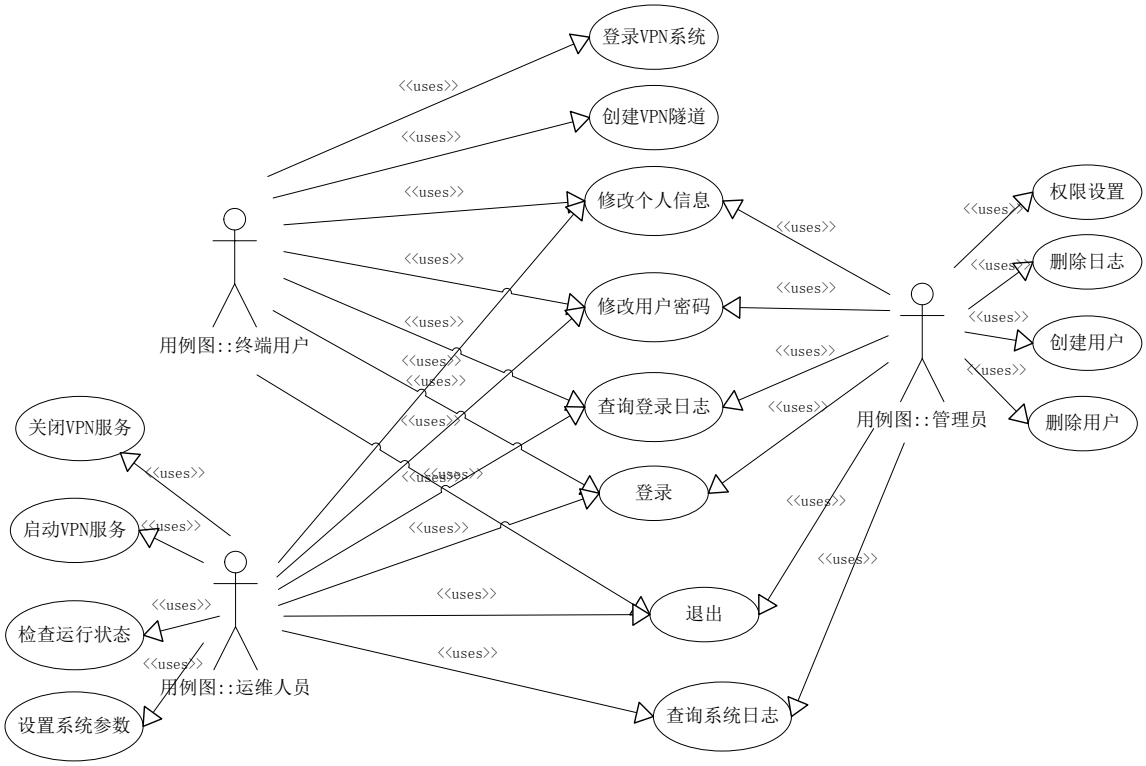


图 2-3 系统用例示意图

终端用户通过系统的注册功能实现账号管理，主要涉及个人信息的管理、登录密码的修改等。同时系统还要为每个用户提供日志记录的功能，用户可以通过系统查看自己的登陆日志，如登录时间、登录 IP 等，VPN 用户可以通过该功能来查看自己的账号是否存在异常登录，以便在密码泄露，账号被黑客非法登录的时候及时发现该问题，避免为个人或系统带来更大的损失；

管理员通过界面实现用户管理的相应操作，包括创建用户、删除用户以及设置用户权限等。当公司新入职的员工需要使用 VPN 服务时，则由系统管理员为其创建 VPN 账户，当有员工离职时，系统管理员也要通过该管理系统删除离职员工的 VPN 账号；用户管理模块还需要与公司已有的 OA 系统做好相应的对接，更大程度地实现办公自动化。用户、角色与权限之间的关系模型如图 2-4 所示。

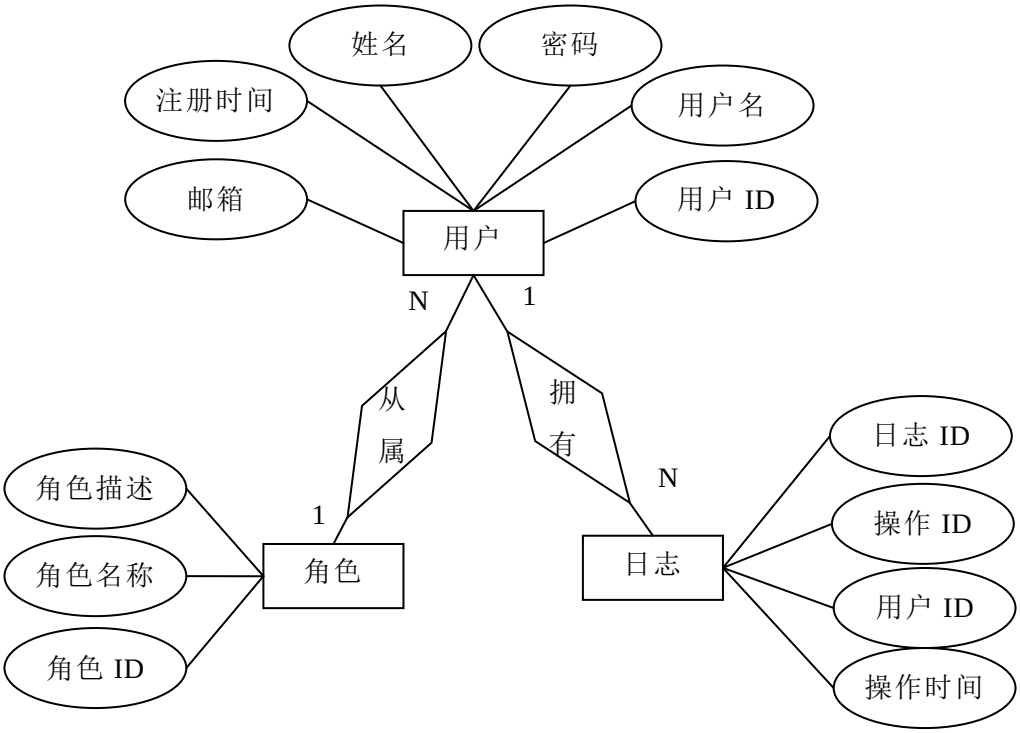


图 2-4 用户、角色等实体的 E-R 图

运维人员需要实时了解系统的运行状态，并且需要操作接口来实现对系统的控制，因此前端界面需要为系统运维人员展示出系统的 CPU 使用率、内存使用率、网络带宽等信息，同时也要展示出当前在线的用户数，以及每个用户所使用的带宽等，状态信息实体的 E-R 模型如图 2-5 所示。

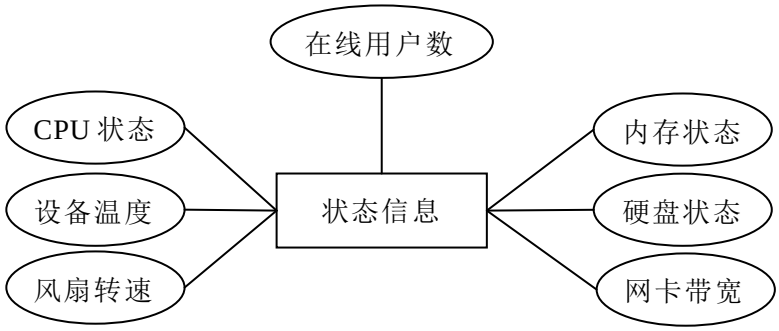


图 2-5 状态信息实体 E-R 图

除了网络设备的运维信息，运维人员还需要根据系统运行情况来调整相应的运行参数。为了保证 VPN 系统在出现故障后能够及时地恢复服务，系统还需为运维人员提供启动和关闭 VPN 服务的功能。运维管理模块的核心数据是状态信息数据。状态信息实体所包含的属性有 CPU 使用率，内存占用率，网卡流量等。

2.3 功能性需求

图 2-6 是 VPN 系统的业务流程示意图，对系统各部分间的关系进行了详细的描述。VPN 用户主要参与用户登录和 VPN 通信过程；运维人员主要参与 VPN 系统状态管理过程，运维人员对 VPN 进程发送的指令或者修改的参数都通过该模块执行并生效，服务器产生的原始运维数据也会经过该模块的处理，最终以图形化的方式呈现。

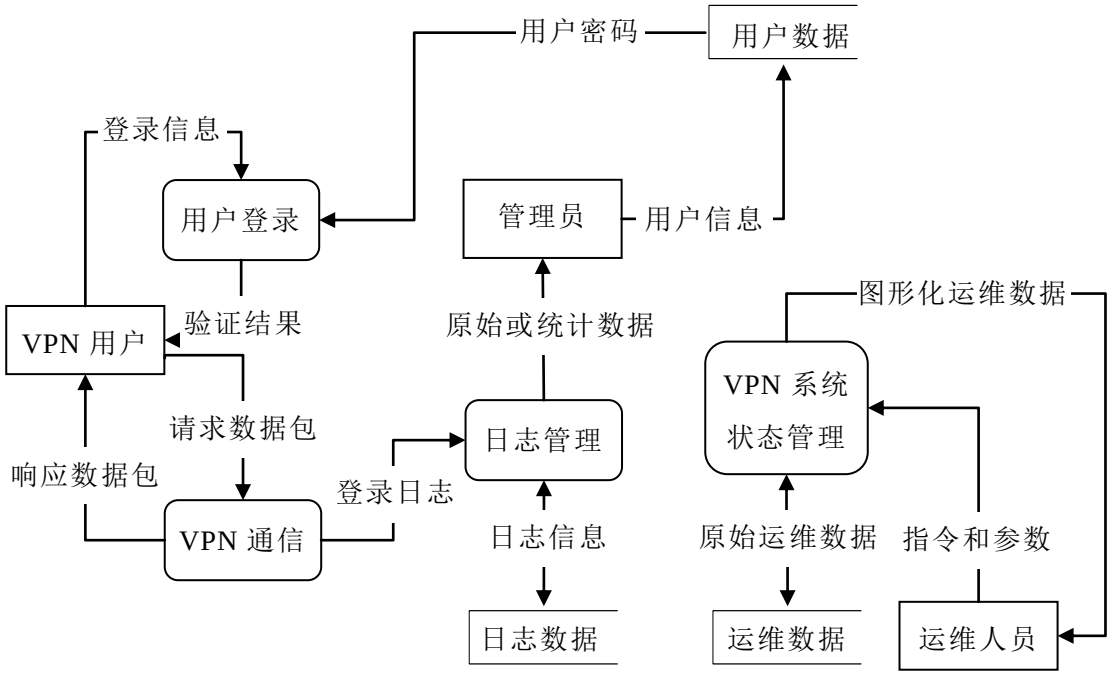


图 2-6 VPN 系统的业务流程示意图

VPN 系统的主要功能包括 VPN 隧道功能、访问控制功能、运维管理功能、内容审计功能和日志查询功能，下面将对各个功能所涉及到的业务流程做详细介绍。

1) VPN 隧道功能：用于向客户端提供 VPN 服务。VPN 系统所需要的加密、压缩、身份认证、数据校验等功能均由 VPN 服务端实现。VPN 加密会话时需要进行加密密钥的协商，同时在通信过程中，还需要定期对会话密钥进行更新，密钥管理模块负责对所有 VPN 客户端的会话密钥进行创建、更新、删除等操作。密钥协商模块的状态转换如图 2-7 所示，主要的状态有设置加密策略，生成加密密钥，发送加密参数，用户身份认证等状态。

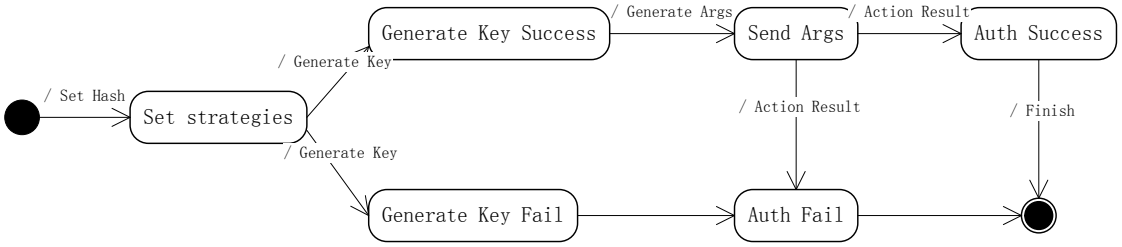


图 2-7 密钥协商模块的状态转换图

2) 访问控制功能：用于管理和控制用户的访问，能够实时限制超流量的用户或对信息系统构成威胁的用户的访问。VPN 网络内部的网络访问控制，包括各 VPN 客户端的 IP、端口的权限问题，即 VPN 网络内部的防火墙。该模块的工作流程如图 2-8 所示。访问控制功能的状态转化关系如图 2-9 所示。

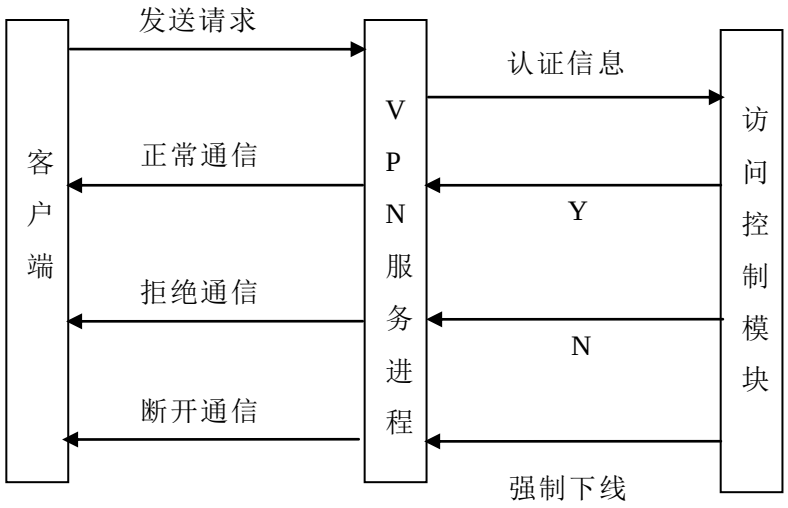


图 2-8 访问控制模块设计图

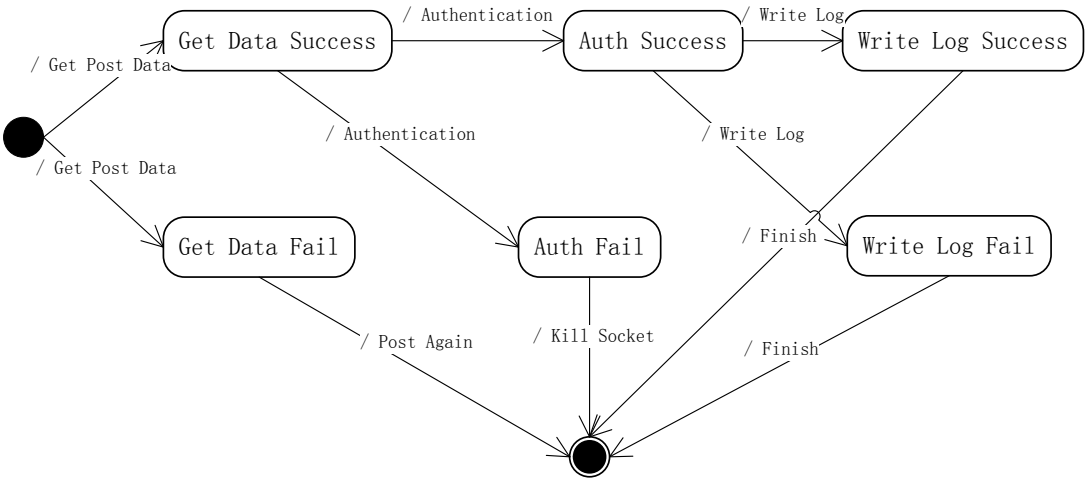


图 2-9 身份认证模块状态转换图

3) 运维管理功能：用于采集 VPN 系统的流量、连接数等信息，还可以获取服务器的 CPU 负载、网卡流量等信息。使系统管理员能够实时地掌握系统的性能和运行状态。VPN 系统状态管理模块的详细业务流程如图 2-10 所示。

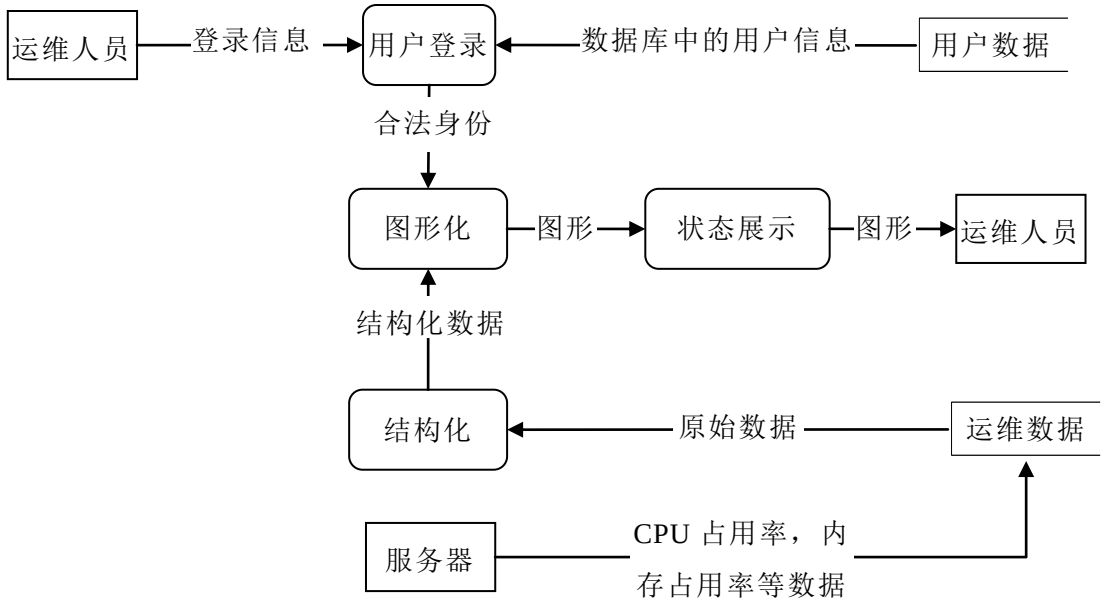


图 2-10 运维管理模块的业务流程示意图

4) 内容审计功能：用于和内容审计系统进行通信，把 VPN 隧道内部的数据提供给内容审计系统。为了保证信息安全，防止内部机密文件或内容通过网络传播到外界，VPN 网络中传输的信息需要经过内容的审查，发现违规的数据传输时，VPN 系统需要终止相应的数据传输，并将相关的日志记录到数据库。内容审计模块的状态转换如图 2-11 所示，主要包括的状态有数据包转发，内容匹配，连接阻断等状态。

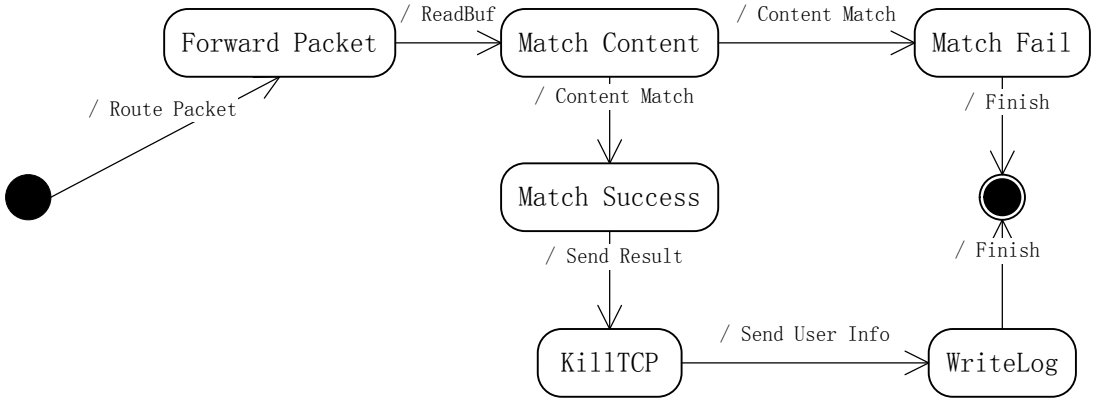


图 2-11 内容审计模块状态转换图

5) 日志查询功能：记录管理员的操作日志、运维人员的操作日志以及用户的登陆日志等信息。

2.4 非功能性需求

1) VPN 服务属于企业网络资源中非常重要的基础设施，因此系统必须要保证优秀的服务质量，即系统需要 7x24 小时稳定运行。

2) 管理系统的界面设计需要充分尊重用户的习惯，本着方便用户操作的原则进行设计，保证系统的新用户经过短暂的培训学习就能够掌握和使用系统所提供的全部功能。

3) 系统应该具有较好的可维护性和可扩展性，做到模块化设计，在进行账号管理、运维接口等关键性模块的开发时，需要考虑企业日后业务和技术的发展趋势，保留简单易用而又功能齐全的接口，方便系统以后的扩展与升级。

2.5 本章小结

本章完成了对高性能 VPN 系统的需求分析工作，从系统的业务需求、业务处理流程、功能需求等角度对系统进行了深入的分析与研究。相关业务人员可以基于本章所做的分析来更好地了解系统的基本需求；架构设计人员可以此为基础更加准确地把握系统的模块功能；技术开发人员也可以此为依据做好技术储备。

第3章 高性能VPN系统的设计

本章主要工作是对高性能 VPN 系统的加密解密、密钥管理、身份认证、内容审计等核心功能模块进行设计，在完成功模块设计和数据流分析之后，又对系统的数据库进行了库表设计。系统在研发完成后要面向企业用户使用，系统在使用过程中涉及到开发、运维、人力资源等多个部门，由于不同部门的用户关注的功能不同，所以在数据库表的设计过程中充分考虑了不同角色的权限问题。

3.1 系统功能架构及技术架构设计

本着低耦合、可复用的设计原则，本文将系统划分为五个子系统，主要包括数据处理子系统、运维管理子系统、网络通信子系统、密钥管理子系统和访问控制子系统，各子系统所涉及到的功能模块如图 3-1 所示：

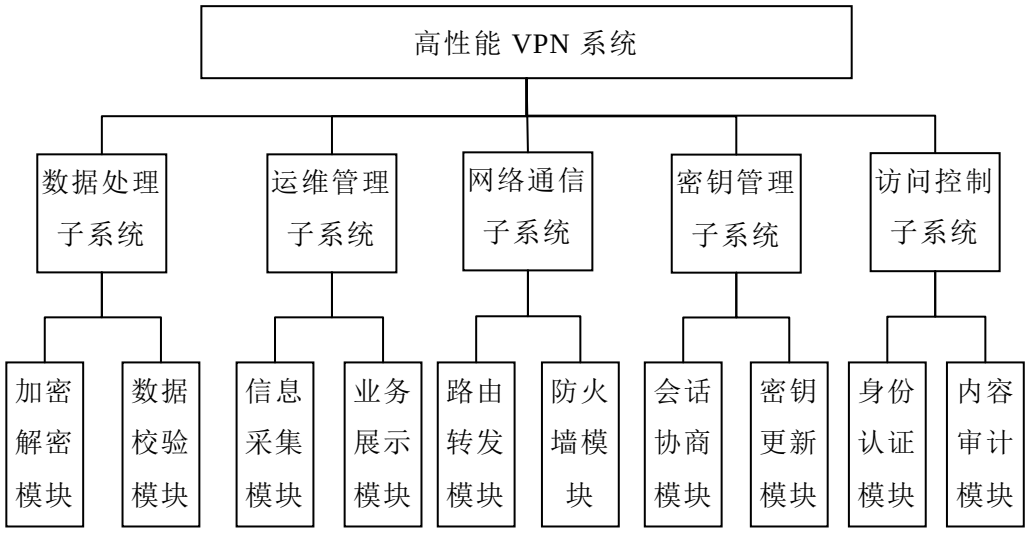


图 3-1 系统功能模块划分

VPN 系统整体的模块组成及各模块间的通信关系如图 3-2 所示。用户提交登录信息后，数据获取模块会将这些信息交由访问控制模块处理，访问控制模块从数据库中读取已经存储的用户名和密码，对用户信息进行验证。对于客户端提交的加密报文，数据获取模块会提交给加密解密模块。数据在此有两个流向，一个是提交给 VPN 内部的路由，进行正常的数据转发，另一个是写入到共享内存，内容审计模块读取明文进行深度报文检测。

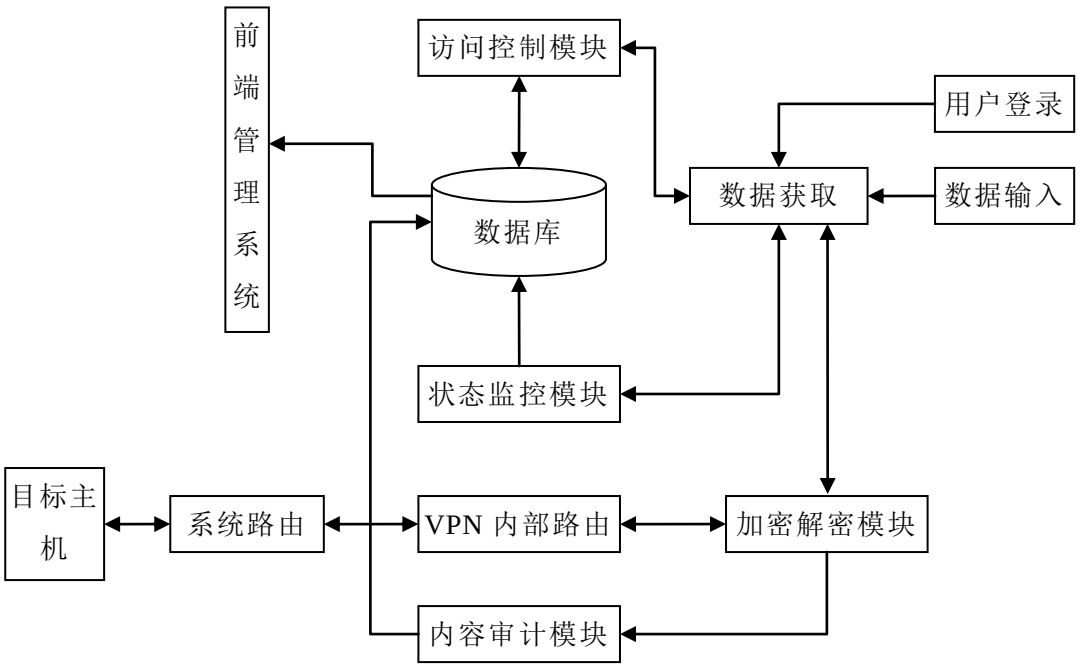


图 3-2 VPN 模块关系示意图

VPN 管理系统采用 B/S 架构设计，是整个业务的核心，包括了用户管理，权限管理，运维数据采集及展示等多项功能。系统用户与后端的软件及设备之间的关系如图 3-3 所示。

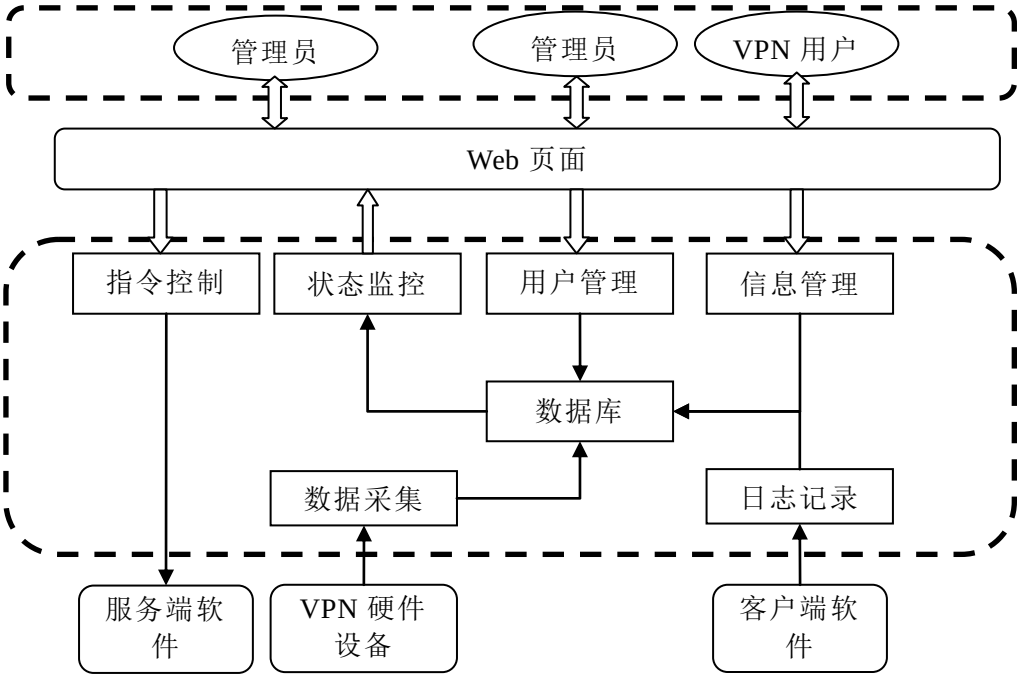


图 3-3 管理系统逻辑关系示意图

运维人员通过前端的 Web 页面将 VPN 系统的运行参数调整和控制指令发送到后端，后端的指令控制模块负责与服务端软件交互，从而解析并执行这些指令。数据采集模块定期从 VPN 硬件设备上采集系统状态信息并存入数据库中，当运维人员想要查看系统的运行状态时，由 Web 页面将状态数据图形化并呈现出来。管理员通过 Web 页面进行用户管理操作，包括添加、删除用户等，Web 页面通过表单形式将管理员的操作发送到后端，再由负责处理用户管理的模块将最终的结果写入到数据库中。VPN 用户对自己用户信息的修改也是通过 Web 页面进行提交，然后由信息管理模块负责将信息写入数据库。

系统的技术主要可以分为 VPN 管理系统实现技术、VPN 通用技术、VPN 服务端专用技术以及 VPN 客户端专用技术。其中深度报文监测技术是实现 VPN 内容审计功能的技术基础；pthread 多线程技术与用户态协议栈相结合实现了多线程的 VPN，进而使得 VPN 服务端软件可以充分利用多核 CPU，取得更好的性能。系统技术架构如图 3-4 所示。

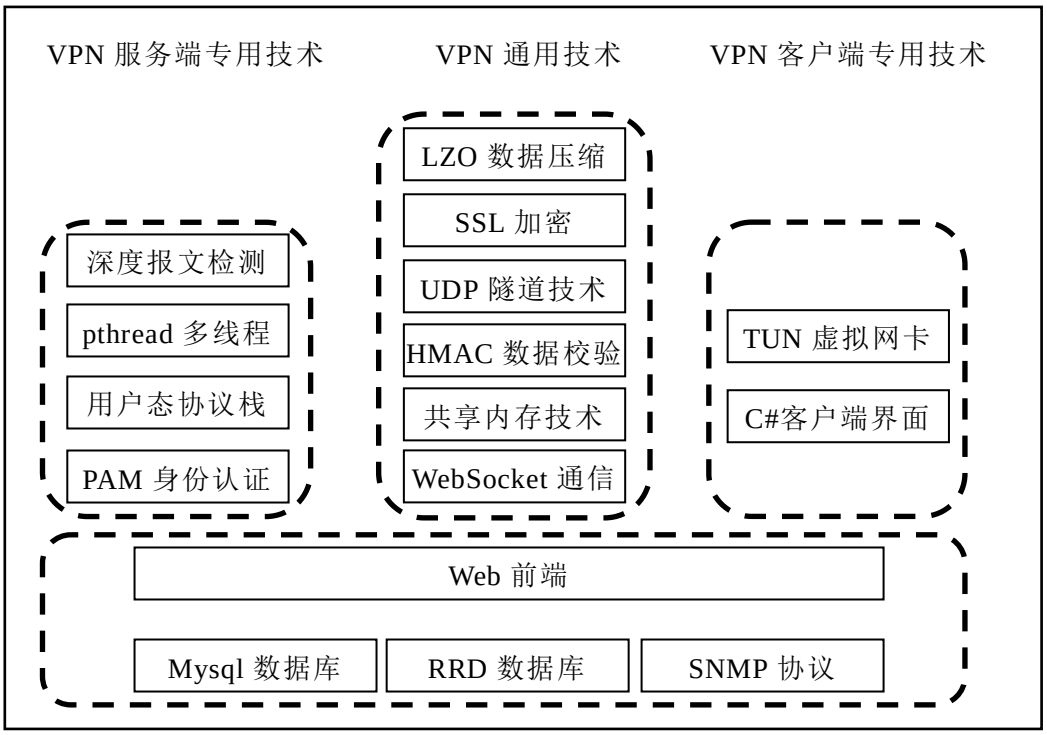


图 3-4 VPN 系统技术架构示意图

3.2 用户认证模块的设计

用户访问控制主要由身份鉴别、服务授权、日志记录、认证信息数据库、异常日志数据库等几部分组成，身份鉴别模块的功能主要是接收客户端提交

的认证信息，并从认证信息数据库中读取该用户的相关信息，进而对该用户提交的信息进行验证。如果验证成功，则发送消息给服务授权模块，服务授权模块 VPN 使用权限授权给该用户。如果身份验证失败，身份鉴别模块会发送消息给日志记录模块，日志记录模块会将本次登录客户端的 IP、用户名、登录时间等信息存储到异常日志数据库中，图 3-5 为用户认证模块的设计图。

用户身份认证的方式一般有三种。第一种是传统的基于用户名和密码的方式，这种认证方式的好处是用户可以自己选择方便基于的密码，只要用户保证密码的私密性以及密码强度，就可以做到很好地认证效果。然而由于目前多数用户的安全意识较差，基于密码的认证方式在使用中也存在着较大的安全问题，很容易被黑客窃取或通过字典碰撞手段猜解密码。第二种方式是基于 PKCS 证书的认证方式，这种认证方式取消了密码，只通过证书进行认证，用户只要保证自己证书的私密性，就能保证认证的安全，而如果证书被窃取或误操作泄露，也会带来安全隐患。第三种认证方式是基于硬件的 PKCS 认证，常用的方式是将客户端的 PKCS 证书和私钥由系统管理员导入到 USB TOKEN 中，用户在进行认证时，必须要将 USB TOKEN 插入到计算机设备，输入 USB TOKEN 的 PIN 码，然后才能使用 TOKEN 中保存的证书进行认证，这样用户的私钥不会被导出或窃取，即使 TOKEN 丢失也有 PIN 码的保护，进一步增强了安全性。

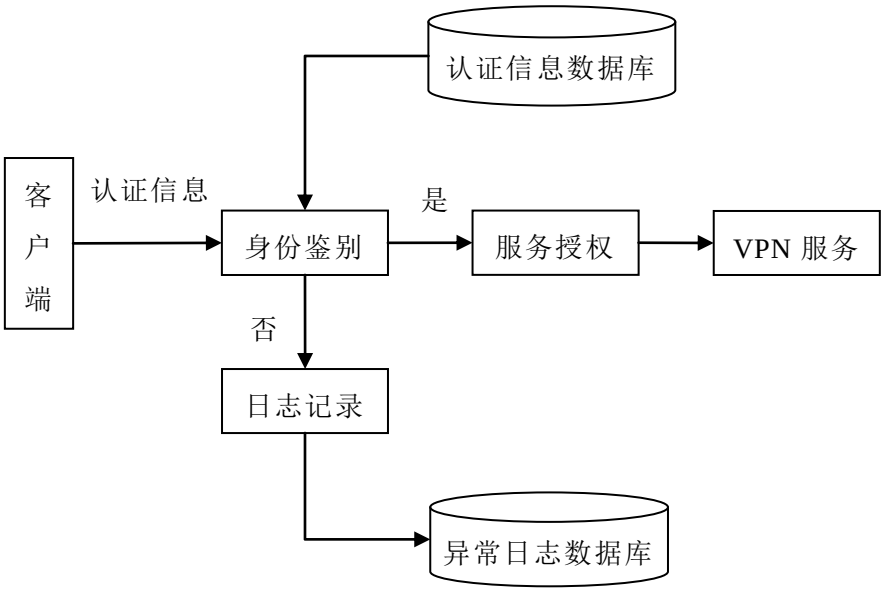


图 3-5 用户访问控制模块设计图

图 3-6 是用户登录过程中的时序图，描述了登录系统、用户管理模块、

身份认证模块、日志记录模块以及数据库模块之间的关系。

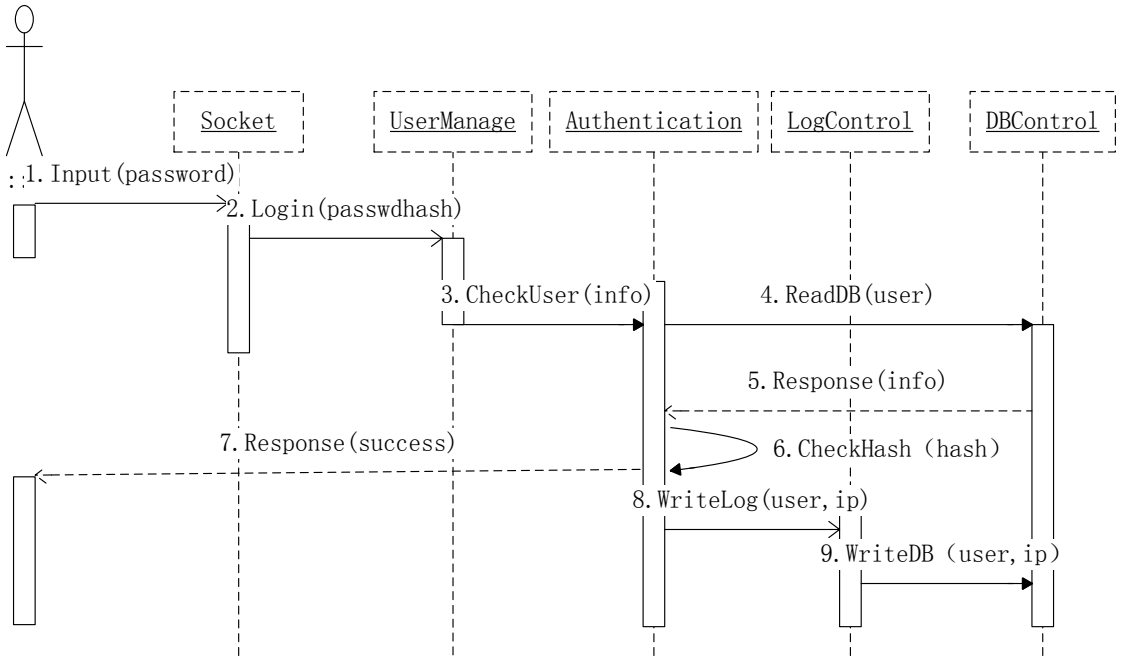


图 3-6 用户认证模块时序图

VPN 用户在登录系统时，首先通过登录系统提供的界面输入自己的用户名和密码；为了保证用户密码不被泄露，登录系统在获取到用户输入的信息后，首先会对用户的密码使用 MD5 算法做单向哈希散列操作，然后调用 LogIn()方法把用户名和经过散列的密码哈希值发送到服务端；服务端在接收到客户端发送过来的数据后，会将数据交由用户管理模块处理，用户管理模块调用 CheckUser()方法把数据交由身份认证模块处理；身份认证模块再调用 ReadDB()方法从数据库中获取该用户的信息，当数据库模块把用户的密码哈希值返回给身份认证模块时，身份认证模块会调用 CheckHash()方法对密码哈希值进行匹配，如果客户端传过来的哈希值与数据库中存储的一致，则说明该用户输入的密码正确，用户在界面将接收到登录成功的信息提示；同时身份认证模块还会调用 WriteLog()方法来记录用户的登录日志；日志记录模块调用 WriteDB()方法把用户的用户名、登录时间、登录 IP 等信息记录到数据库。VPN 支持脚本钩子，并提供了多个环境变量供脚本使用，异常日志的记录就是通过脚本钩子实现的。其中$username、$untrusted_ip、$untrusted_port 是 VPN 提供给脚本的环境变量，分别是登录失败的用户名、登录失败的客户端的 IP、登录失败的客户端的端口。

3.3 内容审计模块的设计

内容审计模块主要完成针对特定用户进行的流量控制、访问内容控制、访问目标地址的控制等。为了实现高效访问控制管理，设计了在进行访问控制处理前进行预处理，有效地提高了访问控制管理效率，流量控制通过对流量进行监测分析，对报文发送进行有效控制；访问内容控制采用可配置的敏感信息数据库技术对信息内容进行高效过滤；访问目标地址控制依据禁访目的地址数据库进行过滤，防范分组的非法外发，如图 3-7 所示。

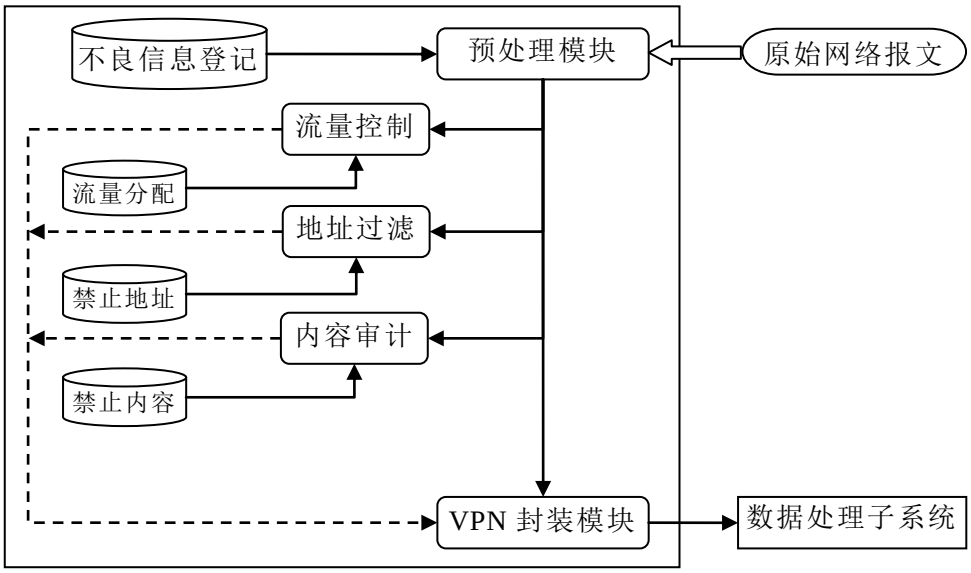


图 3-7 访问控制子系统结构图

内容审计模块的功能实现包括数据获取、内容匹配、通信阻断等，其业务处理流程如图 3-8 所示。当用户在 VPN 系统中浏览网页时，VPN 转发模块负责将用户浏览过程中产生的 IP 报文转发到目标服务器，与此同时，转发模块还会调用 ForwardPacket()方法把数据包复制一份，转发给内容审计系统中的数据获取模块；数据获取模块通过从协议栈获取到数据包后会调用 CheckPacket()方法将需要做内容检查的数据包转交到内容匹配模块；内容匹配模块在初始化的时候，根据相应的配置文件，从入侵检测规则库中读取出当前系统中的所有规则，并将规则存储在内存中。当有数据包需要做内容检测时，内容匹配模块会提取出数据包的负载，并与已有的规则库做字符串的多模匹配；如果匹配到数据包中存在有害信息，则会通过 KillTCP()方法将该 TCP 连接的源 IP、目的 IP、源端口和目的端口等信息提交给通信阻断模块，包括由通信阻断模块负责打断 TCP 连接，从而阻断有害信息的传输，然后通信阻断模块会调用 WriteLog()方法记录本次阻断日志，把访问者的用户名、

IP、所访问的 URL 以及访问该 URL 的时间等信息记录到数据库中。

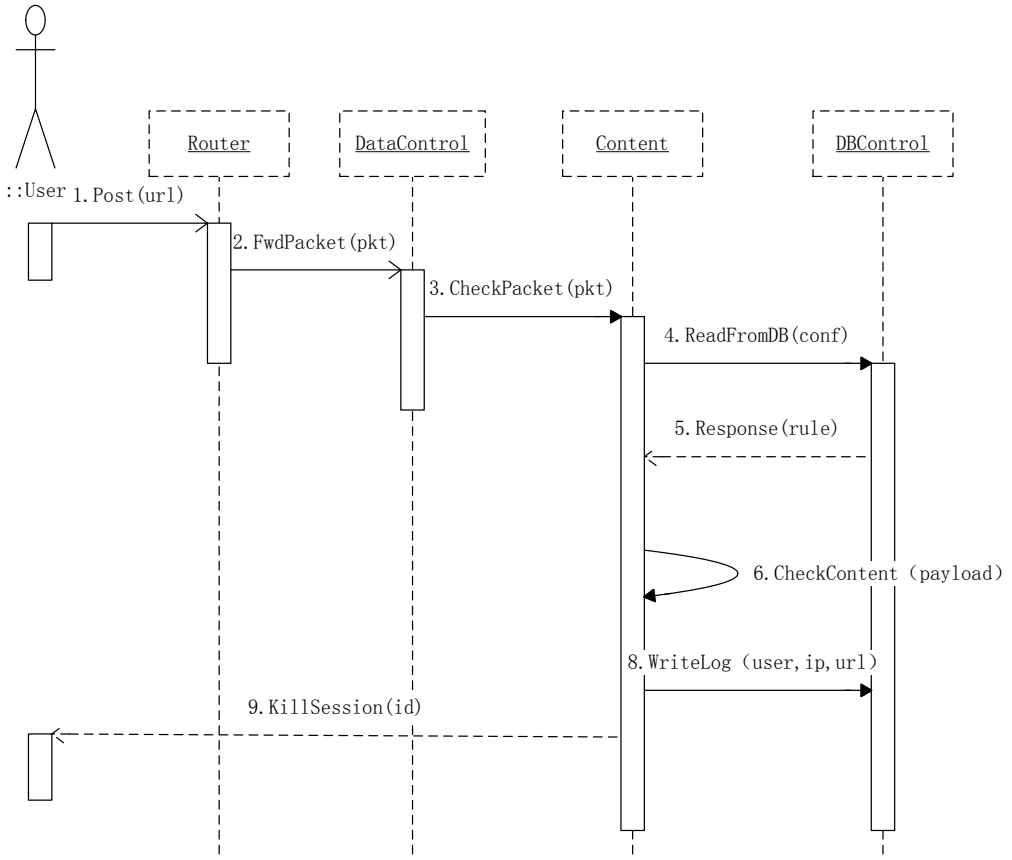


图 3-8 内容审计模块时序图

3.4 运维管理模块的设计

运维管理模块是减轻系统运维工作量的重要模块，主要包括数据采集器、数据分析器、数据存储器、定时器、数据展示等部分。

该模块由定时器驱动，定时器会根据系统配置的参数定时驱动数据采集器从 VPN 服务器获取相关数据。数据采集器负责从服务器采集运维信息，并将数据交由数据分析器处理。数据处理器的主要功能是从配置文件读取系统管理员所关心的有关数据，并从数据采集器提交的原始数据中分析出标准数据，并交给数据存储器把标准数据存储到数据库中。数据展示器负责从数据库中读取标准数据并以曲线图的形式对数据做可视化展示。服务器状态监控模块的工作流程如图 3-9 所示。

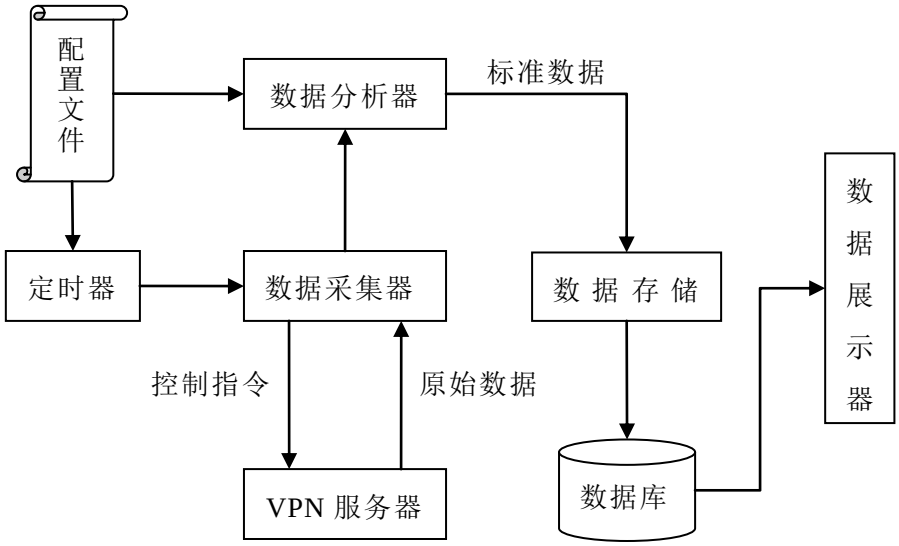


图 3-9 服务器状态监控模块设计图

用户可以通过$name 变量设置图像的名称，通过$timeout 变量设置数据采集频率。$ifin 和$ifout 变量即为图像展示所需要的标准数据，这两个变量后面的表达式是对原始数据的处理方法。配置文件支持注释功能，以”#”开始的文本行是配置文件的注释，配置文件的格式设计如图 3-10 所示。

```
#system configuration file example
$name    = network interface
$timeout = 60
$ifin          = snmp|IF-MIB::ifInOctets|2
$ifout    = snmp|IF-MIB::ifOutOctets|2
```

图 3-10 模块配置文件格式设计图

曲线图的展示主要分为以下几个区域：曲线图名称、曲线显示区域、图例注释区域、坐标轴。图 3-11 为数据展示界面。其中横坐标是时间轴，默认显示最近 24 小时内数据，纵坐标是网卡的数据流量，可以依据实际数据的大小实际生成坐标单位。在曲线下方显示的是 24 小时以内的平均网络带宽。网卡的输入和输出流量分别用不同的颜色表示。

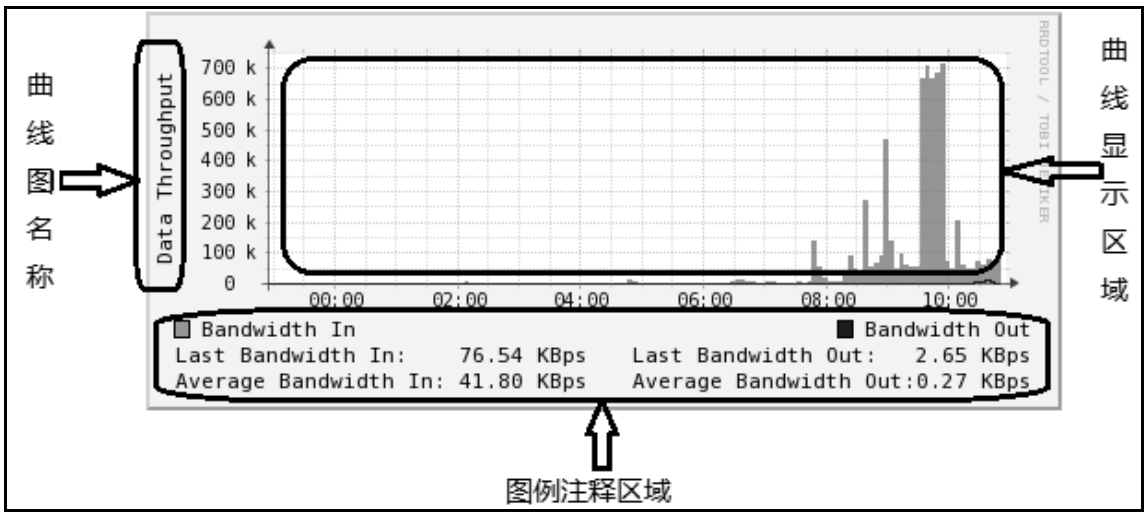


图 3-11 曲线图界面设计

3.5 其它模块的设计

3.5.1 数据处理模块设计

数据处理模块完成整个系统业务核心处理任务，主要实现原始网络报文和隧道数据报文之间的转换，主要包括调用压缩/解压缩模块，加密/解密模块，以及封装/拆封模块。相应的功能结构图如图 3-12 所示。

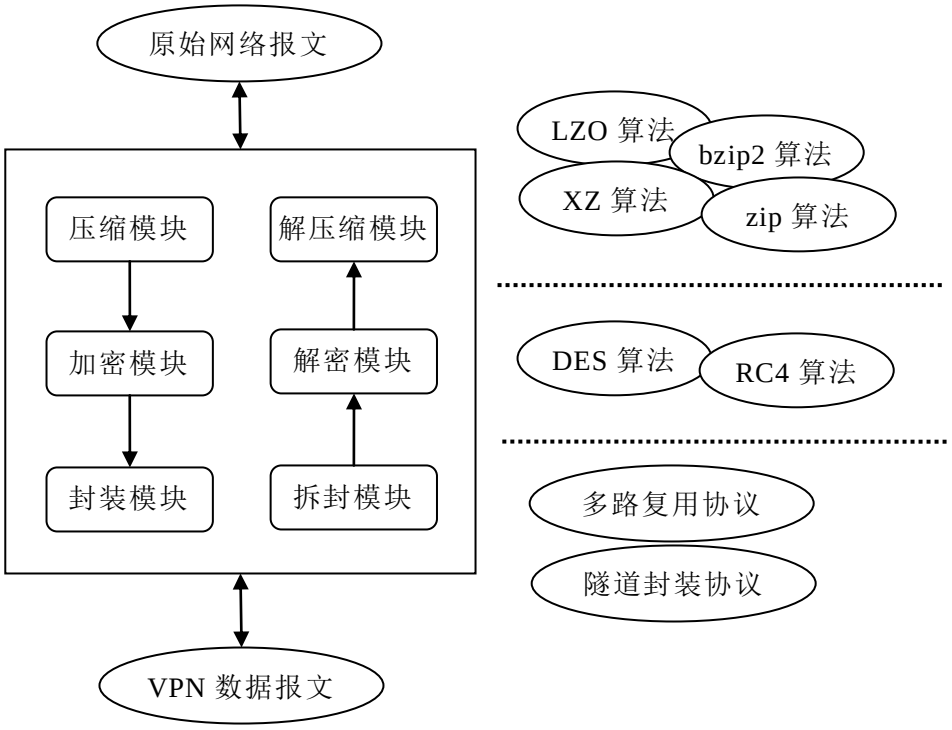


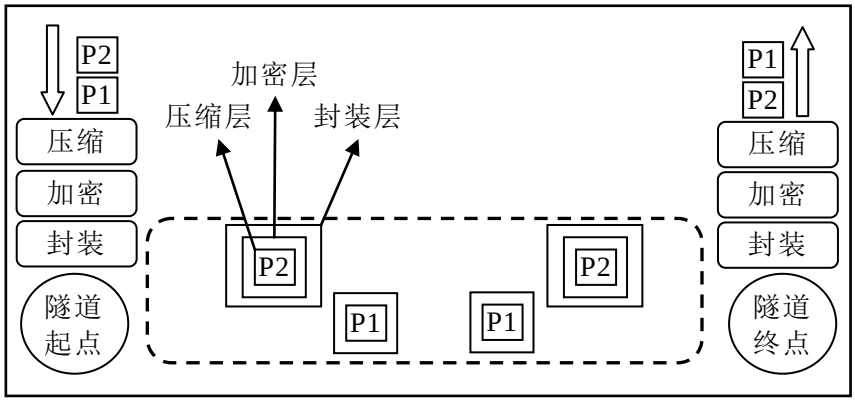
图 3-12 数据处理模块设计图

压缩/解压缩模块：主要支持 LZO 算法、bzip2 算法、xz 算法、zip 算法等，可以根据不同设备的内存配置、CPU 配置进行调整。同时支持单包压缩和多包压缩。

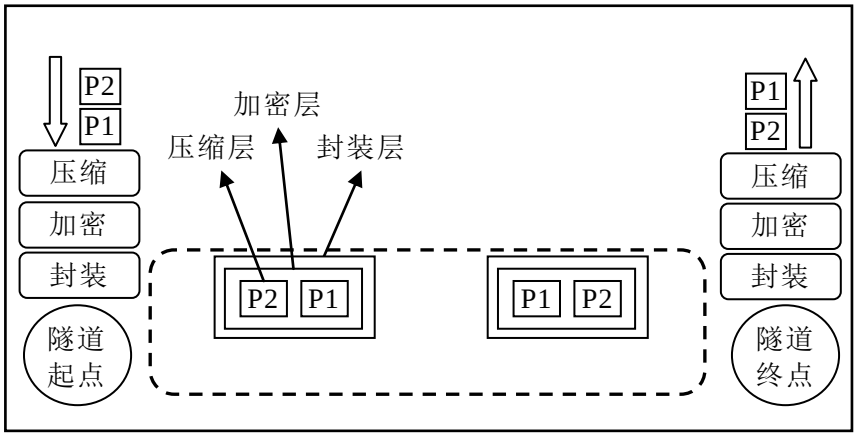
加密/解密模块：支持多种加密算法，包括对称加密算法和非对称加密算法，以及各种摘要算法，核心的对称加密算法有：DES 算法(Data Encryption Standard)和 RC4 算法；DES 算法具有加密速度快的特点，RC4 算法可以使用变长的密钥对数据进行加密，适用于大量数据的场景。

封装/解封模块：主要负责对经过加密和压缩后的数据进行传输层和 IP 层封装和解封装处理，将数据作为 UDP 层负载并添加 UDP 头部，最后再添加 IP 层，组装成可在公网传输的 IP 报文。

图 3-13 是服务器与客户端通信交互过程中的一种典型的情况，结合封装/解封模块、压缩/解压、加密/解密模块对通信方式进行说明。



a) 单包压缩、加密、封装后传输过程示意图



b) 多包压缩、加密、封装后传输过程示意图

图 3-13 数据处理子系统及数据传输过程示意图

3.5.2 密钥管理模块设计

密钥管理模块主要完成用户身份合法性验证、用户访问密钥协商处理等。用户鉴别模块通过在服务器与客户端之间采用 CA 认证技术互验证对方的公钥完成用户鉴别；密钥协商模块在用户鉴别的基础上通过基于 D-H 的密钥协商机制生成预共享密钥，完成密钥协商。VPN 密钥协商协议的简易状态机如图 3-14 所示。

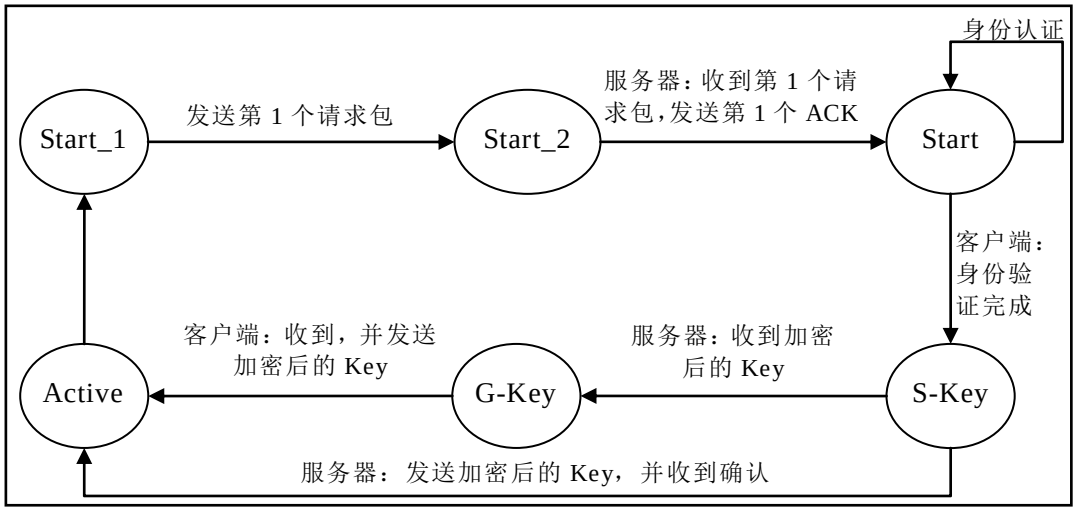


图 3-14 协议状态机设计图

状态机主要包含 6 个状态，客户端在第一个状态 Start_1 时发出第一个请求包，同时转入第二个状态 Start_2；服务器在收到客户端发来的第一个请求包时，回复第一个确认报文 ACK，然后状态机进入身份认证阶段。完成身份认证后，服务器和客户端继续进行密钥协商，包括 S-Key 和 G-Key 两个状态。

客户端先根据本地的安全策略将密钥生成参数发送到服务器端，服务器端会对策略进行匹配，并通知客户端自己支持的加密策略和相关的密钥生成参数。由于客户端和服务器可能存在着多种配置的方式，因此只有通过加密策略的协商，才能让通信双方选择出最合理的加密方式。在完成加密策略的协商之后，客户端和服务器各自进入密钥生成阶段，生成自己的密钥生成信息并传输给对方，完成密钥交换，在身份验证阶段，通信双方会通过加密隧道来传输自己的验证数据，完成本次会话的协商。服务器与客户端之间的密钥协商过程的时序图如图 3-15 所示。

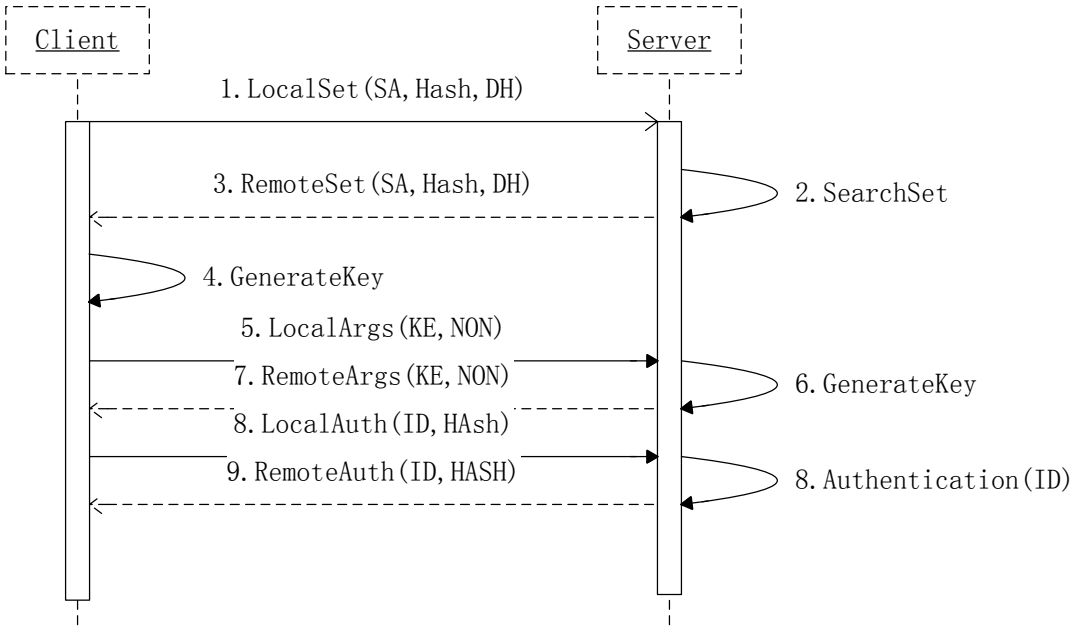


图 3-15 密钥协商模块时序图

3.6 多线程处理模型的设计

VPN 的性能瓶颈与其单线程的软件架构有关，目前利用多线程进行 VPN 处理的方案有两种。一是把 VPN 对数据的处理根据操作划分成多个阶段，再把这些操作分散到固定的 CPU 核上，如有的 CPU 核专门负责处理 socket 通信，另外的 CPU 核负责加密处理或封装处理；另一种方案是利用多线程技术，在不同的 CPU 核上运行处理 VPN 数据的线程。一个线程既负责处理 TCP/IP 协议，又负责处理数据的加密、压缩、封装等处理，本文采用的是第二种方案，因为由同一个线程处理一个连接可以增大 CPU 的 cache 命中率，能够更好地利用 cache 带来的性能提升。

VPN 的多线程处理还离不开多队列网卡的支持，在网卡驱动初始化的时候，首先获取网卡的基本信息，通过这些信息可以得到网支持的队列的数量，再结合硬件 CPU 核的数量，来计算出需要激活的队列数量。计算的方法是取队列数量和 CPU 核数的最小值。在当前的 TCP/IP 协议栈实现中，操作系统根据五元组(传输层协议、源 IP、源端口、目的 IP、目的端口)来标识一条会话，而目前的网络应用程序一般都是直接建立在传输层协议上的，解决会话保持技术的关键就是通过设计应用程序的会话层来使得应用程序业务完全与传输层分离。VPN 数据获取方式、会话保持原理及哈希表的数据结构如图 3-16 所示。

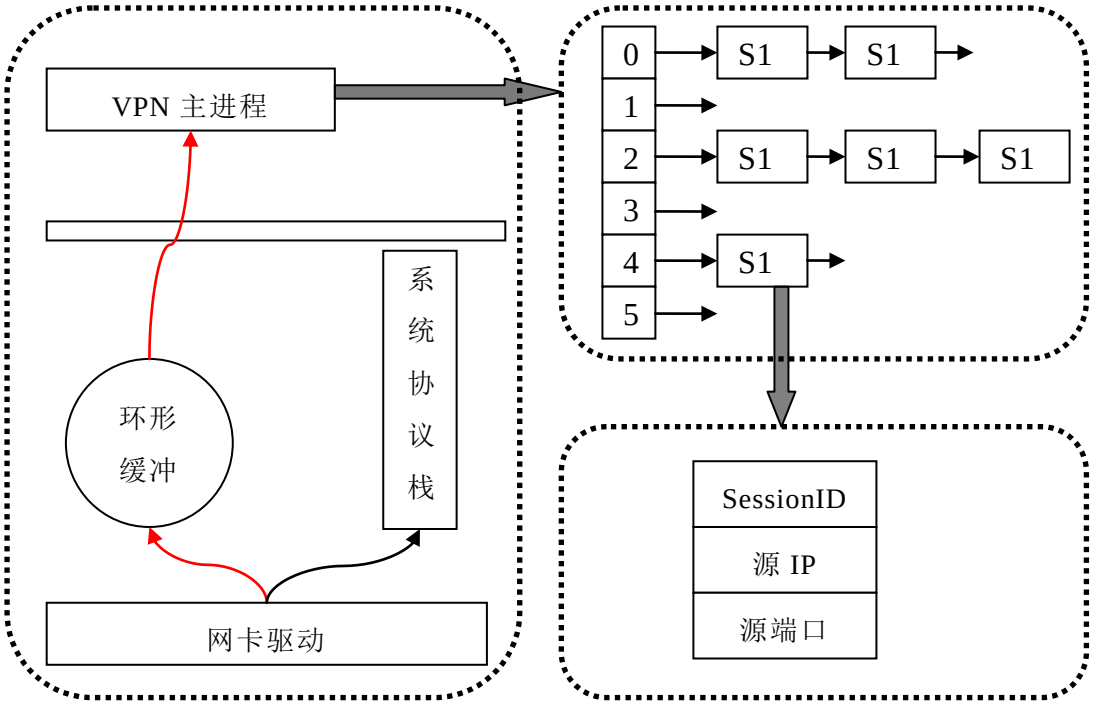


图 3-16 VPN 会话保持技术原理图

VPN 应用进程通过环形缓冲区套接字(PF_RING)机制，绕过操作系统协议栈，直接将网络数据流发送给用户空间的 VPN 进程。VPN 进程通过数据流中的会话标记来标识一个客户端，即使客户端的 IP、端口改变，只要客户端的会话标记不变，该 VPN 会话仍能保持。

3.7 系统的类图及数据库设计

3.7.1 系统类图设计

VPN 系统主要包括 Socket、Router、Encryption、Compress 等 9 个类。其中 DataControl 类负责数据的核心调度，它通过调用 Authentication、Encryption、Compress 等类中的相关方法来完成认证、加密、数据压缩等操作；Content 类中主要实现了内容审计相关的方法，包括 IP 规则匹配、内容规则匹配等；Router 类实现了 VPN 内部的路由器，包括数据包转发、丢弃以及 NAT 等功能。UserManage 类主要负责用户管理，其中的方法可以实现创建用户、设置用户分组、更新用户信息等操作。Encryption 类实现了加密功能，密钥生成、密钥更新、数据校验等重要功能都由这个类完成。系统的关键类的设计如图 3-17 所示。

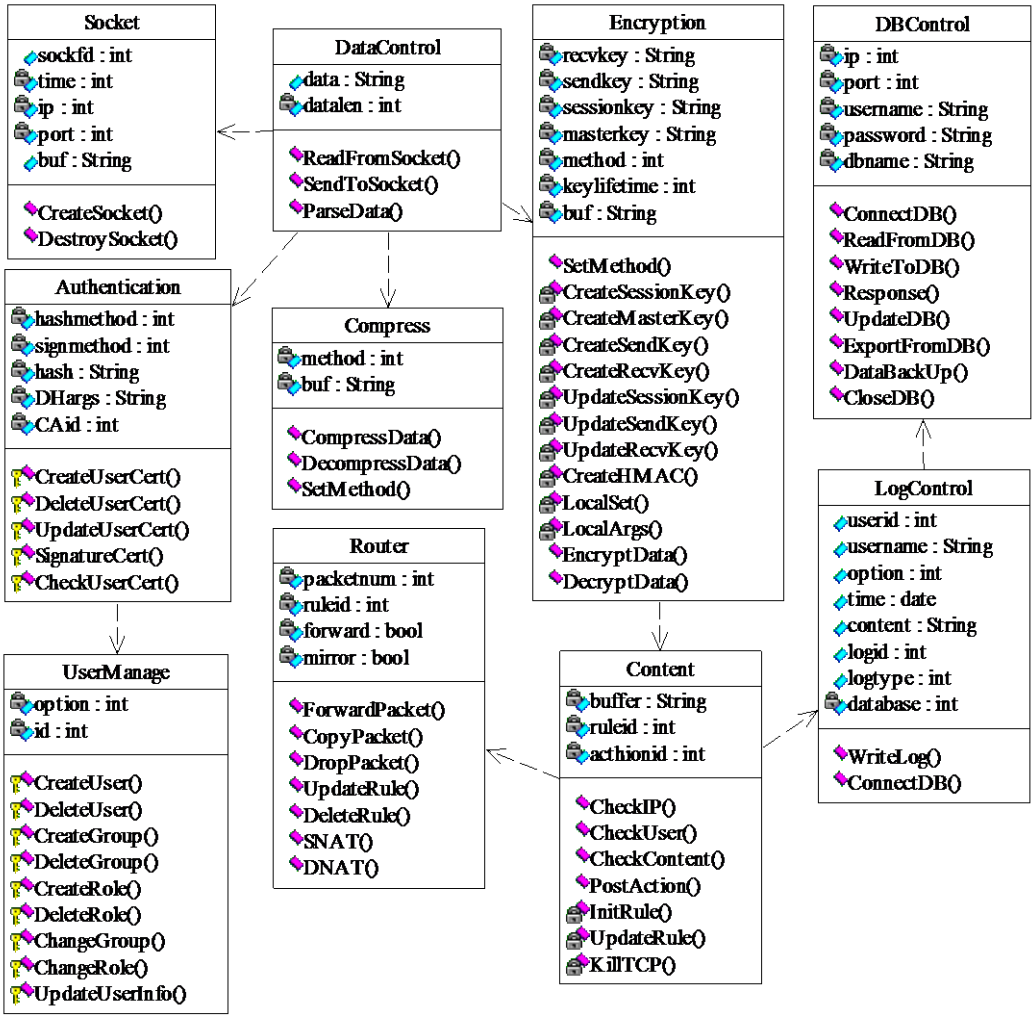


图 3-17 系统主要类图设计

3.7.2 系统数据库设计

系统的数据库设计主要包括了用户权限管理、运维数据管理、日志存储等，由于用户权限管理模块的业务及数据结构最复杂，此处重点该模块的数据库表设计进行介绍。

用户权限管理模块主要实现了对用户的分组管理，不同的分组对应着不同的操作，不同的操作实现不同的功能。系统通过将用户划分到相应的分组来实现对用户权限的控制，用户权限管理模块的数据库表关系图模型如图 3-18 所示：

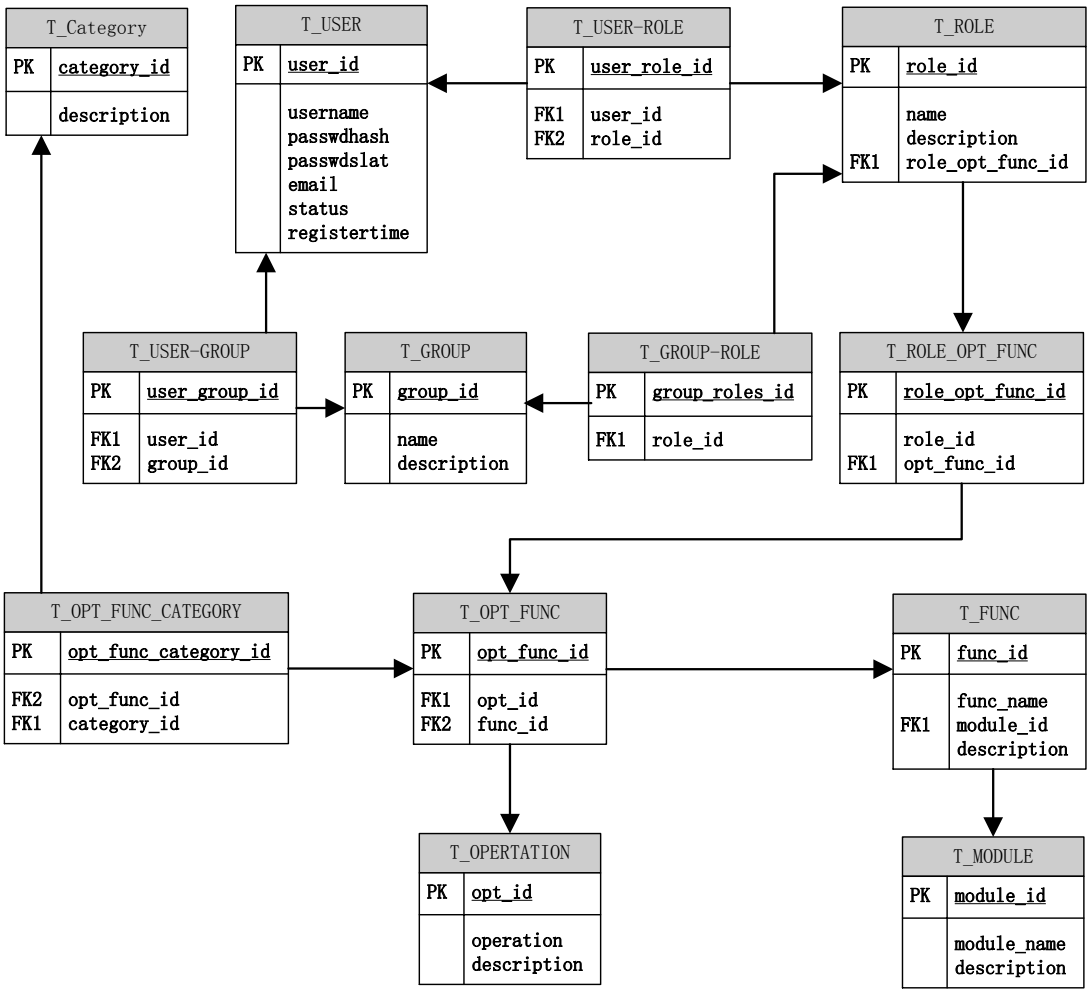


图 3-18 数据库结构示意图

主要数据库表结构的设计如下：

①用户信息表：用于存储用户信息的数据库表，包括用户名、用户的密码 hash、用户的注册邮箱、用户的注册时间等信息。为了保护用户的隐私，数据库中存储的不是明文的密码，而是对密码进行 MD5 散列后得到的 Hash 值。如果服务器遭受黑客攻击，数据库被黑客窃取，以密码 Hash 值的方式可以在一定程度上防止用户的明文密码被泄露，同时防止黑客通过“撞库”攻击和暴力来恢复出密码，在进行 MD5 运算时，加入了 salt，这样可以使 MD5 运算得到的 Hash 值更均匀，并且破解的难度会大大提高。用户信息表结构如表 3-1 所示。

表 3-1 用户信息表结构

字段名称	类型	长度	约束	意义
user_id	int	8	主键	用户表 ID
username	varchar	256		用户名
passwdhash	char	64		密码 Hash 值
passwdsalt	char	64		Hash 使用的 salt
email	vchar	128		用户邮箱
status_id	int	4		用户状态
registertime	timestamp	4		注册时间

②用户状态表：用于描述一个用户所可能有的状态，包括了用户是否在线、用户是否离职等。对于已经离职的用户，需要由人力资源部门的相关人员提交注销账号的申请，然后由 VPN 系统的管理员对相应人员的权限做设置，已离职员工的账号无法再登录到系统中。用户状态表结构如表 3-2 所示。

表 3-2 用户状态表结构

字段名称	类型	长度	约束	意义
status_id	int	4	主键	用户状态表 ID
status_name	varchar	128		状态名称
description	varchar	256		状态描述

③ 操作日志表：用于记录用户的操作历史，包括用户的 ID、操作类型、操作时间等信息，其中 log_id 是操作日志表的主键，每一条操作日志都有唯一的 ID，user_id 和 opt_id 是外键，用于关联执行操作的用户和操作所对应的 ID。操作日志表结构如表 3-3 所示。

表 3-3 操作日志表结构

字段名称	类型	长度	约束	意义
log_id	int	32	主键	日志表 ID
user_id	int	8	外键	用户 ID
opt_id	int	8	外键	操作 ID
opt_time	timestamp	4		操作时间

④ 用户登录日志表：用于记录用户的登录日志，包括登陆用户的源 IP、目的 IP、源端口、目的端口，用户所使用的协议类型以及登录时间、退出时间等详细信息。用户登录日志表结构如表 3-4 所示。

表 3-4 用户登录日志表结构

字段名称	类型	长度	约束	意义
log_id	int	32	主键	登录日志表 ID
user_id	int	8	外键	用户 ID
ip_src	int	4		源 IP
port_src	int	4		源端口
ip_dst	int	4		目的 IP
port_src	int	4		目的端口
protocol_id	int	4	外键	协议类型
login_time	timestamp	4		登录时间
logout_time	timestamp	4		退出时间

⑤ 协议类型表：用于存储 VPN 的协议类型，包括协议 ID、协议名称、描述等信息。VPN 系统支持多种传输层协议类型，包括 TCP 协议、UDP 协议、GRE 协议、SCTP 协议等。协议类型表结构如表 3-5 所示。

表 3-5 协议类型表结构

字段名称	类型	长度	约束	意义
protocol_id	int	32	主键	协议类型 ID
protocol_name	int	8		协议名称
description	varchar	512		协议描述

⑥ 角色表：用户存储系统中的角色类型，包括角色 ID、角色名称等信息。这个表中还需要管理操作表中的信息。系统中涉及到的角色有三种，分别是管理员、运维人员和 VPN 用户。不同的用户角色有不用的操作权限。角色与权限之间的关联关系由外键 opt_func_id 决定。角色表结构如表 3-6 所示。

表 3-6 角色表结构

字段名称	类型	长度	约束	意义
role_id	int	4	主键	角色类型 ID
name	varchar	256		角色名称
description	varchar	512		角色描述
opt_func_id	int	4	外键	角色-操作 ID
user_id	int	4	外键	管理员 ID

3.8 本章小结

本章基于上一章的需求分析，对系统做了概要设计，对系统的核心功能模块进行了划分，并通过 VPN 客户端、VPN 服务器以及 VPN 管理系统三部分来为用户提供服务。然后本文分别从 VPN 系统的架构设计、VPN 核心模块的功能设计，以及管理系统的数据库设计等方面对系统模型进行了详细阐述，重点对 VPN 的身份认证模块，密钥管理模块和内容审计模块的时序图进行了描述。本章的功能架构设计、模块划分和数据库表设计可以使开发人员更好地从整体上把握系统的功能以及系统各个功能之间关系，对系统的详细设计和关键技术的研发具有指导意义。开发人员可以根据时序图，模块关系图等信息更好地了解业务流程，从而可以更好地进行详细设计和系统开发。

第4章 高性能VPN系统的实现

通过前面系统需求分析、概要设计和数据库分析，我们可以从整体上把握系统的功能与架构。本章首先介绍了系统实现的技术架构，包括实现 VPN 协议的一些通用技术以及用户态协议栈、深度报文监测等 VPN 服务端的专用技术。本章的主要工作是对系统的主要模块功能及核心技术进行详细设计，同时给出功能的实现方法。

4.1 用户认证模块的实现

用户认证模块是由 VPN 的 PAM 插件和 Mysql 数据库实现的。模块内部功能的具体设计见本文 3.2 节。该模块的工作流程图 4-1 所示。

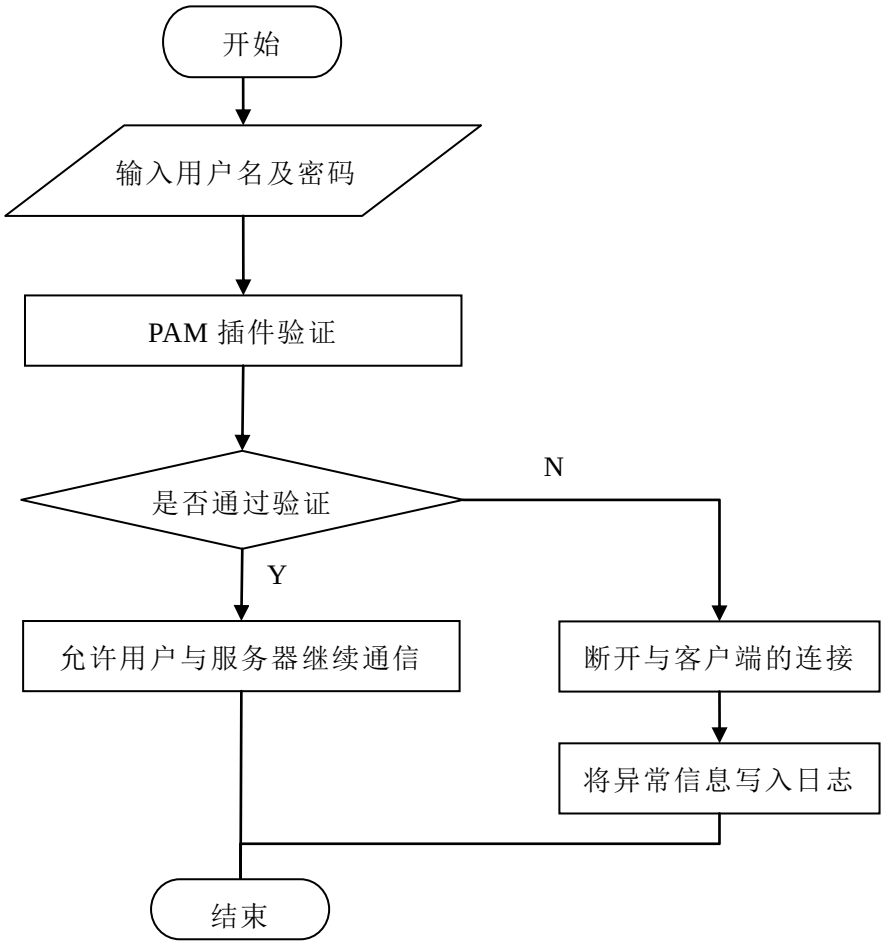


图 4-1 用户访问控制模块流程图

Mysql 数据库用来存储 VPN 用户的用户名和密码。PAM 插件负责对 VPN 用户做认证，如果通过认证，则允许该用户接入 VPN 系统，如果客户端提供的用户名和密码与 Mysql 数据库中存储的不一致，则该用户不能通过 PAM 插件的认证， VPN 进程中断与该客户端的 Socket 连接，并将该客户端的 IP、用户名等信息添加到系统异常日志中。用户密码明文存储是非常危险的，如果数据库中的信息被黑客窃取，则用户的常用密码就会被泄露，不仅对 VPN 用户的个人信息安全带来了威胁，也可能会对整个 VPN 系统造成不可挽回的灾难性损失，因此用户的密码必须是通过单向散列函数进行加密过后的密码哈希值，即使黑客窃取到了用户的密码哈希值，也很难通过哈希值来恢复出用户的明文密码。然而，由于用户自己的安全意识较为淡薄，很多时候他们会选择比较简单的数字或字母组合来做为自己的密码。这样就无法对抗黑客的字典攻击。因此在用户设置密码时，系统会检测密码强度，强制用户使用更高强度的密码。

4.2 内容审计模块的实现

内容审计功能基于深度报文监测技术实现，模块详细设计见本文 3.3 节。其中内容审计模块被设计成可扩展的，可以在 VPN 进程运行时加载到数据包处理流程中，也可以作为独立的进程进行旁路的审计工作。当内容审计模块作为独立进程时，它与 VPN 主进程之间的数据交换时通过进程间的共享内存技术来实现的。在 Linux 系统中，可以通过调用 shmget 函数来获取到共享内存。共享内存的使用方式是一读一写的，即 VPN 进程负责将数据包内容写到共享内存，内容审计模块负责从共享内存中读取数据包内容，为了提高系统的性能，本文通过环形缓冲区（ring buffer）的方式来对共享内存进行读写，从而避免了对共享内存的加锁操作。

内容审计模块主要关注 IP 地址以及数据包中的传输内容两个部分。如果数据包的 IP 地址是发向非法地址的，则内容审计模块会向路由分发模块发送控制指令，停止对该条数据流的转发，这样就实现了阻断通信的目的。当数据包中包含特殊关键字时，内容审计模块会通过应用层的阻断动作来实现对通信的封堵。如当某用户通过 VPN 网络向企业外部传输机密文件时，内容审计模块匹配到文件内容后，会发送 TCP reset 报文打断 TCP 连接，这样就导致文件传输失败，同时模块会将本次阻断动作记录到日志中，包括 VPN 用户名、传输时间、匹配到的规则 ID、客户端 IP、对方 IP 等重要信息。内容审计模块的处理流程如图 4-2 所示：

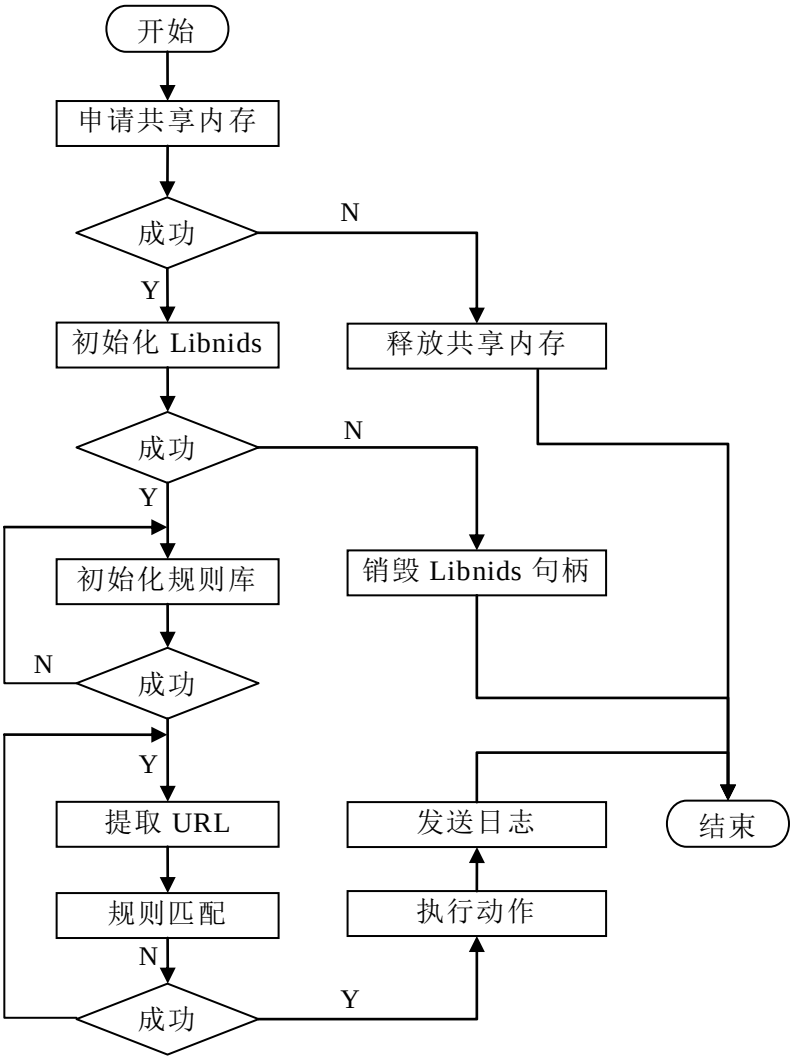


图 4-2 内容审计模块流程示意图

模块运行时首先进行初始化，申请共享内存并初始化 Libnids，然后将规则库加载到内存。在完成初始化后，系统会对每个数据包进行检查，尝试从中提取出 URL。如果 URL 提取成功，则将该 URL 与规则库中的 URL 进行匹配，进而决定下一步将要执行的动作。

Libnids 提供的回调函数入口包括 IP 层、TCP 层和 UDP 层，分别用于处理网络层 IP 报文，应用层 TCP 数据流和 UDP 数据包。匹配 URL 的功能采用了 Libnids 的 TCP 回调函数。回调函数传入的是 tcp_stream 结构体，其中包括该条流的四元组（源 IP、目的 IP、源端口、目的端口），建立时间，传输的数据等重要信息。在进行 URL 匹配时，首先根据特征子串“HTTP 1.0”或“HTTP1.1”来判断 TCP 数据流是否为 HTTP 协议；再根据特征子串“GET”，“POST”来提取出用户所访问的资源 resource；最后根据特征子串“HOST”

提取出 HTTP 服务器的 Hostname，将 Hostname 和 resource 拼接即可提取出用户访问的完整的 URL。

系统在初始化时，首先会读取管理员配置的所有 URL，然后采用 AC 自动机(Aho-Corasick automation)对 URL 进行多模匹配，将上述提取出的 URL 输入到多模匹配模块，由多模匹配模块进行匹配，如果匹配到规则中的某条 URL，则将返回该规则的规则 ID，如果未匹配到任何 URL 则将返回 NULL。

4.3 运维管理模块的实现

运维管理模块的具体设计已在本文 3.4 节进行了详细介绍。本模块主要通过数据采集脚本实现。数据采集脚本获取目标服务器的相关数据，并将数据实时更新到环形数据库中，再通过 RRD 自带的工具 RRDtool 做出曲线图，对服务器状态做实时的呈现，Web 页面将这些图像显示给用户。定时器的实现是采用 Linux 系统的 Cron(Linux 系统下的定时执行工具)计划任务机制，运维管理模块的逻辑流程如图 4-4 所示。系统主要关注的运维数据有如下信息：CPU 使用率、内存使用率、网络带宽、网络延迟。CPU 使用率和内存使用率通过 ps 命令进行采集。其采集到的原始信息如图 4-3 所示。其中的第三列数据“0.0”为 VPN 进程的 CPU 占用百分比，第四列数据“0.3”为内存占用百分比。在录入 RRD 数据库之前，通过 Linux Shell 的 awk 命令，提取出第三列和第四列数据，这样就完成了对原始信息的处理。

```
[root@sg1 ~]# ps aux|grep psvpn|grep –v grep
root      9701  0.0  0.3  46424  3460 ?        Ss   Sep22   2:59
/usr/sbin/psvpn --daemon --writepid /var/run/psvpn/server.pid --cd
/etc/psvpn/ --config server.conf
```

图 4-3 PS 命令采集的原始信息

对于网络带宽的采集，本文采用了基于 sysfs 虚拟文件系统统计的方法，在 Linux 系统中，内核会把设备或驱动的输出信息通过 sysfs 输出到用户态的空间供用户访问。而网络接口的相关统计数据将会被输出到“/sys/class/net/<ethX>/statics”中。其中 rx_packets 表示网卡设备收到的数据包的数量，rx_bytes 表示网卡设备接收到的字节数，VPN 服务器对网络带宽的使用率可以通过这些网卡统计数据计算出来。系统每隔 5s 计算一次网络带宽，同时将计算结果输入到 RRD 数据库中。对于网络延迟的测量则通过发送 ICMP 报文来实现。

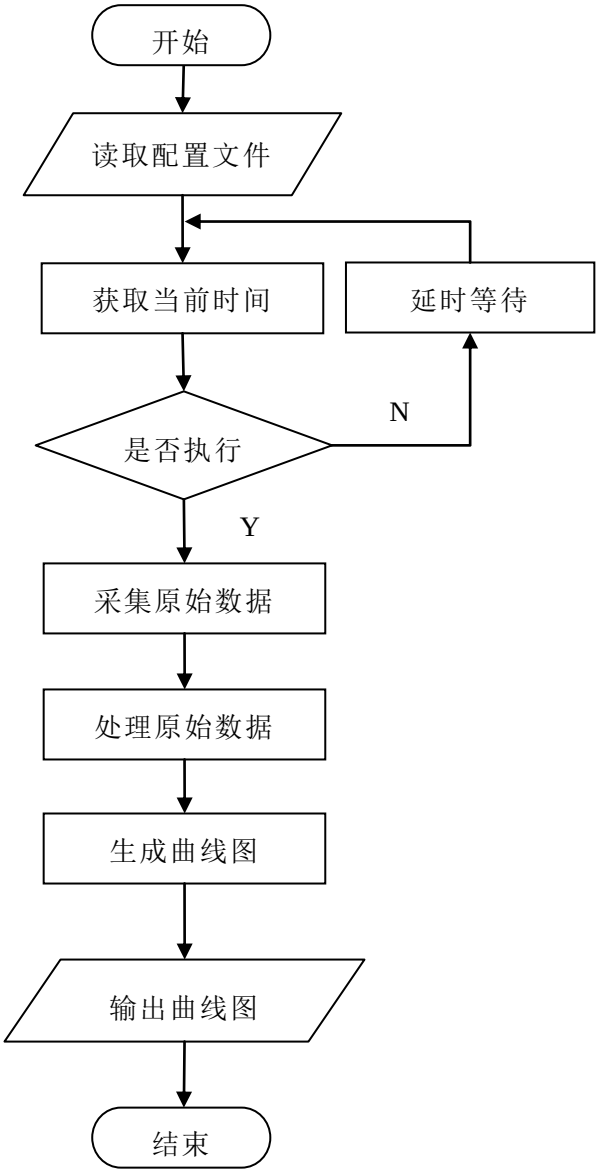


图 4-4 状态监控模块工作流程图

4.4 其它模块的实现

4.4.1 数据处理模块的实现

数据处理模块的业务流程和功能设计已在 3.5.1 章节进行了详细的描述，在此重点对模块内部的核心逻辑流程及其实现进行介绍。数据处理的基本操作包括数据捕获，数据压缩，数据加解密，此外还涉及到数据包路由转发和数据包的内容审计接口，本模块的工作流程图如图 4-5 所示。

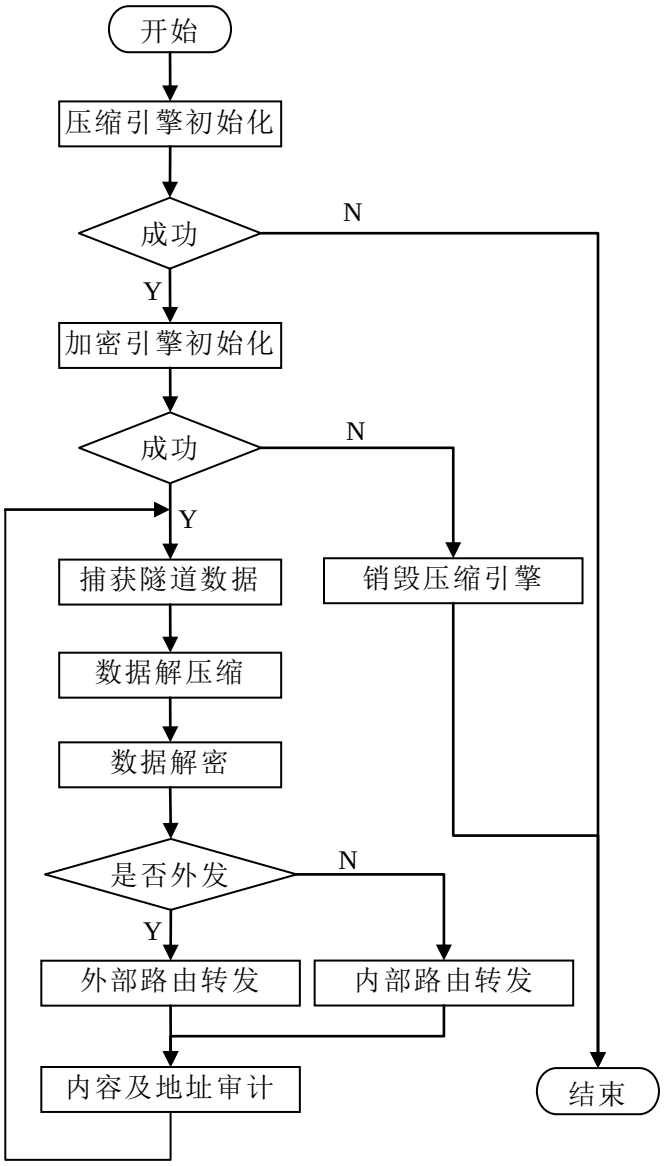


图 4-5 数据处理模块流程示意图

VPN 隧道的数据压缩采用 LZO 压缩算法，VPN 进程经多路复用 IO 模块从 socket 读取数据，该数据为加密和压缩后的经过封装的隧道内部报文。多路复用 IO 模块将读取到的数据交由数据压缩引擎处理。数据压缩引擎采用 LZO 压缩算法在内存中对数据进行实时解压缩，将解压缩后的内容存放到报文队列。数据加解密引擎从解压缩后的数据报文队列中取出报文，然后调用握手阶段协商好的加密算法进行解密，从而得到明文的 IP 报文。该明文 IP 报文随后会进入路由转发队列，等待路由转发模块将该 IP 报文发向其目的

地。目的地址是 VPN 隧道内部的 IP 报文由 VPN 内部路由转发模块进行处理。目的地址是外网 IP 的报文，需要先经过内容及地址审计模块，过滤掉发向非法地址的报文或者包含有特殊关键字的报文，然后通过系统的外网网关将该 IP 报文转发出去。

4.4.2 密钥管理模块的实现

密钥管理是 VPN 加密功能中非常重要的部分，密钥管理模块实现了主密钥协商、会话密钥协商、会话密钥更新等功能，具体的设计见 3.5.2 章节的相关描述，此处主要介绍密钥管理模块的实现，其逻辑流程如图 4-6 所示。

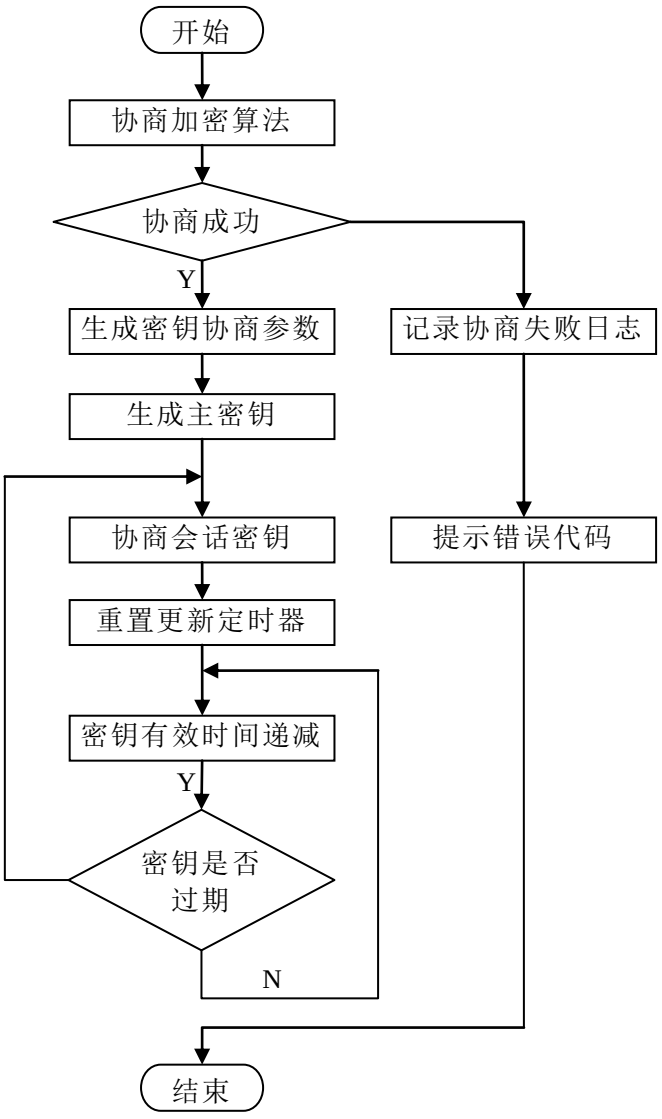


图 4-6 密钥管理模块流程示意图

密钥管理模块的主要工作流程由密钥定时器进行控制，由定时器负责检查密钥是否过期，并在密钥过期时驱动模块重新协商密钥。VPN 协议有着广泛的应用，需要支持无线路由器、智能手机、嵌入式工控设备、PC 机以及服务器等设备。由于这些设备的 CPU 计算能力有着非常大的差距，所以在选择加密算法的时候也需要考虑到设备的 CPU 计算能力及功耗问题。如当客户端运行在基于 ARM CPU 的无线路由器上时，如果使用 AES 算法，则会使得路由器的 CPU 负荷较高，加密及网络传输速度也会受到限制，故在这种情况下会配置客户端使用轻量级的 blowfish 或 ChaCha20 对称加密算法。在建立 VPN 会话的时候，客户端需要先将自己支持的加密算法发送给服务器端，告知服务端接下来要使用的加密算法。

对于对称加密算法来说，最重要的是要保证对称密钥的安全。本文在设计 VPN 的密钥协商协议时，采用了两个阶段的密钥协商过程。第一个阶段，客户端和服务器之前先通过 Diffie-Hellman 密钥交换协议协商出预主密钥 PreMaster Key，然后再通过该 Key 将用于会话的密钥进行加密传输，这样通信双方就安全地完成对称密钥的交换。同时为了提高协议的安全性，防止会话过程中的数据流被非法攻击者保存下来进行暴力破解，VPN 协议还规定了对称密钥的有效时间，通常该时间设定为 3600 秒。当密钥失效时，通信双方会重新协商新的会话密钥，而新会话密钥的协商也不依懒于前一阶段的任何参数，因此进一步保证了新密钥的安全。

4.5 本章小结

本章介绍了 VPN 系统各个模块的详细设计，并给出了具体的技术实现方法。与目前已有的 VPN 系统相比，本系统更加注重系统的安全性，避免了针对 VPN 系统的多种攻击；更加注重易用性，提供了 VPN 管理系统，可供用户方便的管理系统用户；为运维人员提供了运维管理模块，使得用户可以清晰地了解系统的当前运行情况。同时本系统的用户管理、权限管理以及运维数据采集、分析模块在设计之初就考虑到了以后的可移植性，规范了通用的数据接口，实现了可复用的目的。

第5章 高性能VPN系统的测试与分析

为保证 VPN 系统的软件质量，本章重点对系统做了整体的测试与分析。系统测试工作可以有效地避免因为设计、开发失误而导致的重大隐患。同时也可以通过系统测试让用户确认必要的模块功能是否正确实现。最后通过分析 VPN 系统的测试数据来分析 VPN 系统是否满足企业在日常应用中的需求。

5.1 测试方案

系统测试是系统开发过程中的重要环节。通过测试能够保证系统功能的正确性。本章的测试目标在于测试整个系统的运行情况以及对各个模块进行功能测试，以验证相应模块功能的有效性。用于测试的客户端及服务器的主要软硬件参数如表 5-1 所示。

表 5-1 客户端与服务器软件及硬件配置

序号	类型	客户端参数	服务器参数
1	CPU 型号	Pentinum Dual-Core	Intel(R) Xeon(R) CPU E5-2620
2	CPU 主频	3.0 GHz	2.6 GHz
3	内存	2.0 GB	16 GB
4	操作系统	Windows 7	ReadHat Linux
5	Web 服务器	N/A	HTTPD
6	数据库	N/A	MySQL

根据系统架构的设计以及实现的情况，测试的策略是：现根据功能模块的划分进行单元测试，然后通过系统集成实现系统整体测试。单元测试的主要任务是根据设计来测试该模块的功能是否实现。系统测试主要任务是根据系统结构设计来测试各模块间是否能够协作完成整个 VPN 的功能。

5.2 测试方法

测试分为单元测试、集成测试和系统测试三大类。在单元测试中，针对各个模块都进行了黑盒测试和白盒测试，黑盒测试采用等价类测试方法，白盒测试采用路径覆盖测试方法。

1)单元测试

身份认证模块的输入是客户端证书，在客户端与服务器建立连接后，客户端会主动提交自己的证书用于身份认证。根据证书格式、证书签名和证书有效性可以将等价类分为 3 个有效等价和 4 个无效等价类。输入的具体等价类划分如表 5-2 所示。

表 5-2 身份认证模块黑盒测试等价分类表

输入及外部条件	有效等价类	无效等价类
证书格式	正确的 pem 格式证书(1)	证书格式错误(4)
证书签名	由服务器 CA 证书签名(2)	非可信任 CA 证书签名(5)
证书有效性	证书在有效期内(3)	证书过期(6)
		证书被吊销(7)

身份认证模块的逻辑流程以及白盒测试流程如图 5-1 所示。在认证过程中主要涉及到对证书属性的 4 个判断，由此可以产生流程上的 8 个分支。

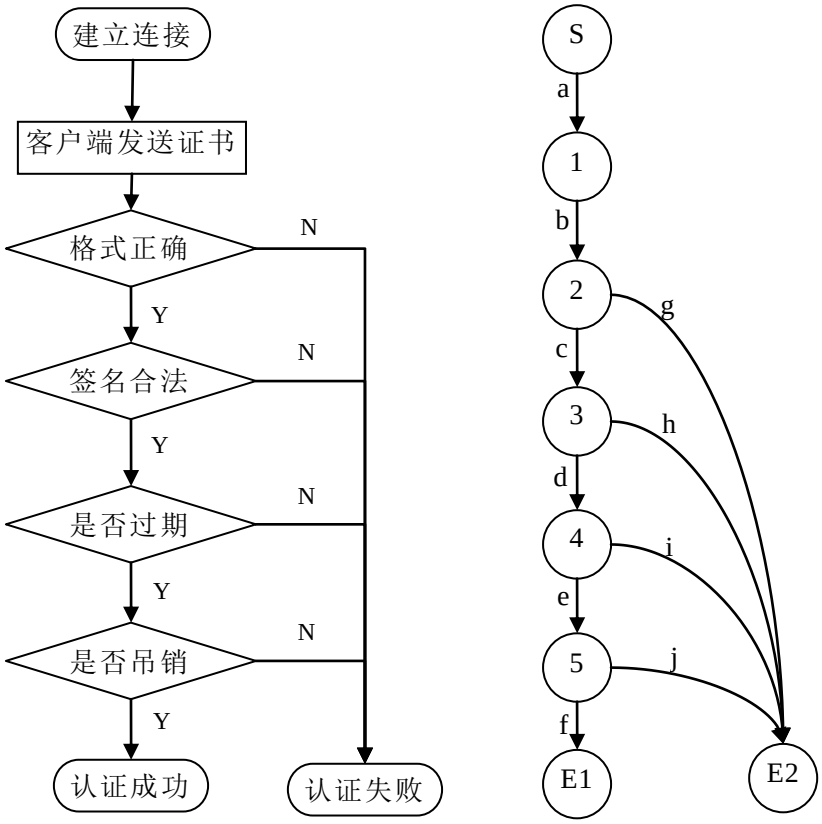


图 5-1 身份认证模块白盒测试流程示意图

2)集成测试

集成测试把 VPN 各个核心模块组装到一起进行测试，其中 VPN 网络连通性测试可以覆盖的模块有会话协商模块、身份认证模块、路由转发模块以及数据封装模块，只要 VPN 客户端与服务器之间能够正常通信，则说明这些模块工作正常。VPN 系统的具体集成测试方法如表 5-3 所示。

表 5-3 集成测试

序号	测试名称	覆盖模块	测试方法
1	VPN 网络连通性测试	会话协商模块 身份认证模块 路由转发模块 数据封装模块	关闭 VPN 的加密和压缩功能，测试客户端与服务器之间能够正常通信
2	VPN 数据传输加密测试	加密解密模块 数据校验模块 数据压缩模块 密钥更新模块	启用 VPN 的加密和压缩功能，抓取服务器与客户端之间传输的数据包，查看加密和压缩功能是否生效
3	VPN 内容审计功能测试	防火墙模块 内容审计模块 加密解密模块 数据压缩模块	通过 VPN 访问被禁止的内容，查看连接是否被阻断
4	运维管理业务测试	信息采集模块 业务展示模块	运行 VPN 服务器和运维管理系统，查看前端展示的数据

3)系统测试

系统测试重点对系统的用户管理、运维管理、内容审计等功能进行了测试。由于系统各模块之间联系紧密，运维管理功能测试可以验证系统前后端的通信是否正常，内容审计功能的测试可以验证 VPN 主进程的所有模块间是否正常工作。

表 5-4 系统测试

序号	测试名称	测试方法
1	管理系统登录测试	测试用户登录管理系统，并查看登录日志
2	运维管理系统测试	启动数据采集模块，查看前端生成的曲线
3	内容审计功能测试	连接 VPN 后浏览网页，查看测试用户所浏览的 URL 是否被记录

5.3 测试用例

本文将对测试过程中的几个关键测试用例进行详细介绍，由于篇幅关系，其他测试用例再次不再赘述。

测试用例 1：身份认证模块的黑盒测试

用于身份认证模块的黑盒，输入证书的类型分为 5 类，覆盖了所有的有效等价类和无效等价类，测试用例如表 5-5 所示。

表 5-5 身份认证模块黑盒测试用例

序号	输入数据	覆盖范围	预期结果	实际结果	结论
1	合法证书	(1)(2)(3)	身份认证成功	身份认证成功	通过
2	格式错误的证书	(4)	身份认证失败	身份认证失败	通过
3	自签名证书	(5)	身份认证失败	身份认证失败	通过
4	过期证书	(6)	身份认证失败	身份认证失败	通过
5	被吊销的证书	(7)	身份认证失败	身份认证失败	通过

测试数据表明身份认证模块顺利通过黑盒测试，实现了模块必需的功能。

测试用例 2：身份认证模块的白盒测试

白盒测试通过 5 个测试用例覆盖到了测试流程中所有的点和边，验证了所有可能出现的情况。白盒测试的测试用例如表 5-6 所示。

表 5-6 身份认证模块白盒测试用例

序号	输入数据	覆盖的点	覆盖的边	预期结果	实际结果	结论
1	合法证书	S,1,2,3,4,5,E1	a,b,c,d,e,f	认证成功	认证成功	通过
2	格式错误的证书	S,1,2,E2	a,b,g	认证失败	认证失败	通过
3	自签名证书	S,1,2,3,E2	a,b,c,h	认证失败	认证失败	通过
4	过期证书	S,1,2,3,4,E2	a,b,c,d,i	认证失败	认证失败	通过
5	被吊销的证书	S,1,2,3,4,5,E2	a,b,c,d,e,j	认证失败	认证失败	通过

测试数据表明身份认证模块顺利通过白盒测试，相关逻辑及流程实现正确无误，达到了系统的要求。

测试用例 3：管理系统登录日志测试

用户登录成功后，通过前台管理页面可以查询到用户的登录时间、登录 IP 及端口信息。测试结果如图 5-2 所示。经过对比实际的登录操作和系统日

志，验证了系统对所有登录日志记录都是正确的。

图 5-2 用户访问控制功能测试结果

测试用例 4：内容审计功能测试

本用例主要测试 VPN 系统的内容审计功能。客户端在连接上 VPN 服务器后，通过 wget www.baidu.com 请求该网站的页面，查看内容审计模块的 URL 日志可知该网络行为已经被记录下来，日志中分别为 VPN 客户端的虚拟 IP、URL 访问时间及具体的 URL。测试数据表明内容审计功能工作正常，测试结果如图 5-3 所示。

```
10.8.0.6        2014-06-15 09:25:04     www.baidu.com/
10.8.0.6        2014-06-15 09:25:05     www.baidu.com/home/nav/data/useless?asyn
=1&t=1339723692808
10.8.0.6        2014-06-15 09:25:05     www.baidu.com/home/nav/data/msg?asyn=1&t
=1339723692817
10.8.0.6        2014-06-15 09:25:09     www.baidu.com/
10.8.0.6        2014-06-15 09:25:10     www.baidu.com/home/nav/data/useless?asyn
=1&t=1339723698250
10.8.0.6        2014-06-15 09:25:11     www.baidu.com/home/nav/data/msg?asyn=1&t
=1339723698258
```

图 5-3 内容审计功能测试结果

测试用例 5：运维管理功能测试

本用例用于测试服务器的运维管理功能。测试期间，由测试用户接入系统，连续运行 24 小时，记录期间的网络流量和服务器 CPU 的使用情况。测试用户在 8:00 至 12:00 之间进行了少量的网页浏览和下载行为，经过对比，网卡流量的曲线与测试用户的实际上网行为相符合，测试结果如图 5-4 所示。

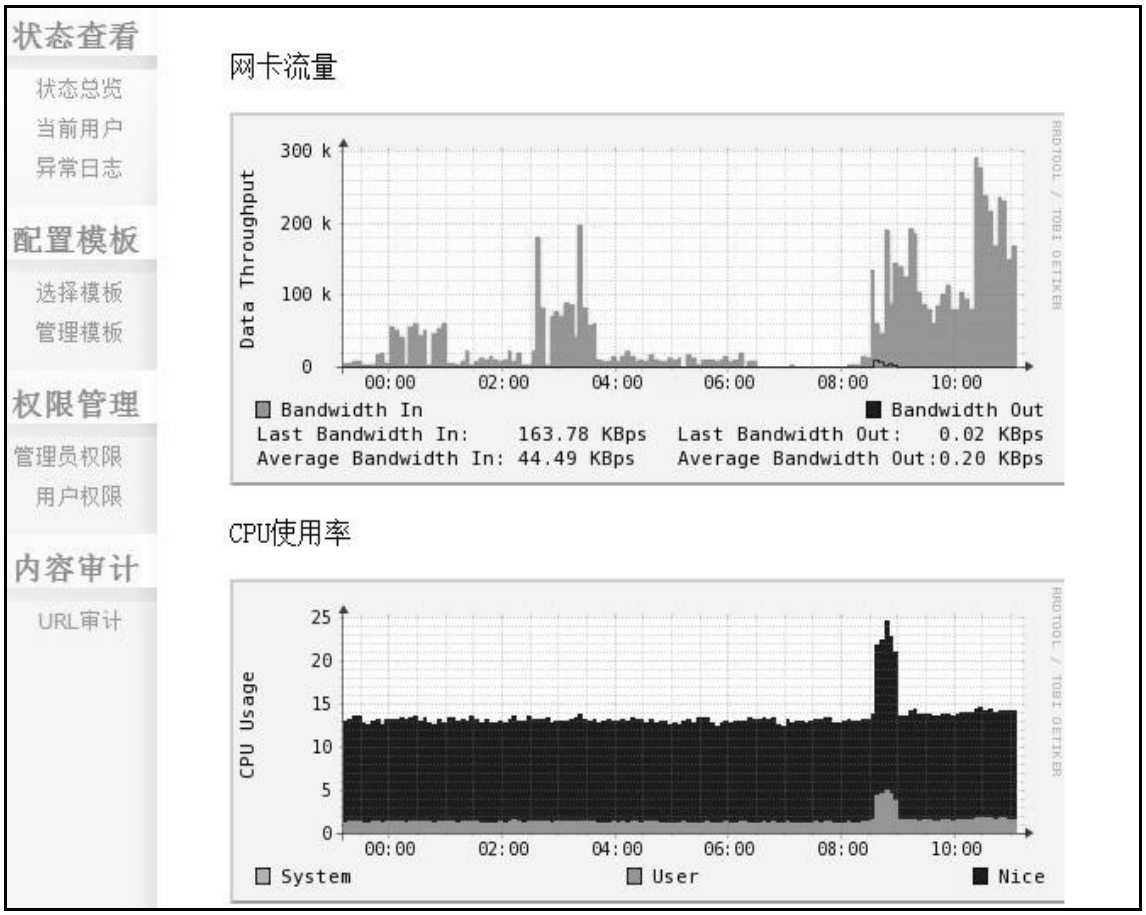


图 5-4 服务器监控功能测试结果

5.4 测试结论

通过对 VPN 系统进行的单元测试、集成测试及系统测试，证实该系统完成了在需求分析阶段所提出的功能。核心模块如密钥管理模块、数据处理模块等工作正常，用户可以使用 VPN 建立起的加密隧道进行通信。访问控制模块工作正常，可以对用户的身份进行验证，并且可以记录用户的登陆日志。运维管理模块工作正常，可以从服务器端采集 CPU 占用率、网卡流量等信息，并且生成曲线，供运维人员实时掌握系统状态。内容审计模块工作正常，管理可以通过后台日志来查看 VPN 用户所访问的 URL，配合内容过滤系统，可以有效的阻止有害信息的传播。综合本次测试的结果及数据，可以认为本 VPN 系统达到了用户的需求。

5.5 本章小结

软件的测试工作是软件开发过程中至关重要的一步，它不仅可以检验研发成果，还可以在软件发布前发现潜在的问题，保证软件系统更好地为用户服务。本章主要对 VPN 系统进行了模块测试与整体测试，通过单元测试验证了各个模块的功能已经正确实现；通过集成测试验证了系统核心模块之间的输入输出正确，可以正常协作；通过系统测试证明 VPN 各个关键功能均正确实现，具备为企业用户提供高性能的 VPN 服务的能力。

结　论

本文基于 VPN 的使用现状，分析了目前存在的问题及瓶颈，并提出了高性能 VPN 系统的设计及实现方法，达到了预期的目标，实现了为网络游戏、电子商务等中小型企业提供安全高效的 VPN 服务的目的。本文的主要工作内容及技术创新包括以下几个方面：

首先，本文重点解决了 VPN 协议的安全问题。在会话欺骗模拟攻击测试中，本 VPN 系统引入的 HMAC 校验技术有效地避免了中间人攻击，比 IPSec 协议具有更高的安全性和健壮性，又因为数据校验仅针对会话初始阶段的几个报文，不会影响系统性能。

其次，本文研究了提高 VPN 性能的关键技术。在网络吞吐量压力测试中，本系统的网络吞吐量较其它 VPN 有数倍提升，未出现虚拟网卡性能瓶颈；在高并发压力测试中，当用户数量达到万级时，系统仍然可以实时处理用户请求，未出现 Socket 性能瓶颈。测试数据表明本文所提出的多线程用户态协议栈与零拷贝高速数据包捕获技术相结合的方案解决了传统 VPN 所面临的性能问题。

再次，本文实现了 VPN 综合管理系统。在用户管理方面，系统可以与企业已有的 OA 系统做对接，方便企业的人员及权限管理；在数据安全方面，系统提供了内容审计系统，可以防止企业内部的机密数据通过 VPN 隧道泄漏到外界，较传统 VPN 具有更高的可控性。

最后，通过对系统的单元测试、集成测试以及系统测试，验证了系统的核心功能，也证明了相关关键技术的可行性。各项测试数据表明系统达到了设计目的，完成了需求分析阶段的所有功能，能够更有效地保证企业用户的网络信息安全。

随着互联网用户对信息安全的需求越来越大，VPN 领域还存在着许多技术研究热点，如 P2P VPN、面向个人移动终端的 VPN 自组网络、物联网 VPN 等，本人对 VPN 技术的研究也将继续进行下去。

参考文献

[1] Gujrathi, Siddharth. Heart bleed bug: An openssl heart beat vulnerability[J]. International Journal of Computational Science and Engineering, no. 2014(5):61-64.

[2] 杨光. 小米 800 万用户信息泄露互联网信息安全再鸣警钟[J]. 计算机与网络, no. 2014(10):10-11.

[3] 黄日涵, 张莉. 全民网络时代的信息安全[J]. 中国公共安全:学术版, no. 2012(1):86-90.

[4] Hamzeh K, Pall G., Verthein W, Taarud J, Little W and Zorn G. RFC2637: Point-to-point tunneling protocol[OL]. Internet Engineering Task Force, 1999. https://www.ietf.org/rfc/rfc2637.txt.

[5] Jeffrey Erman, Anirban Mahanti, Martin Arlitt. Byte Me: A Case for Byte Accuracy in Traffic Classification[C]. In MineNet '07: Proceedings of the 3rd annual ACM workshop on Mining network data, 2007:35-38.

[6] Frankel S and Krishnan S. RFC6071: IP Security (IPsec) and Internet Key Exchange Document Roadmap[OL]. Internet Engineering Task Force, 2011. https://tools.ietf.org/html/rfc6071.

[7] Kim, Young-Jin, Vladimir Kolesnikov, Hongseok Kim, and Marina Thottan. SSTP: a scalable and secure transport protocol for smart grid data collection[C]. In Smart Grid Communications, IEEE International Conference 2011:161-166.

[8] Riyad Alshammari, Peter Lichodzijewski, Malcolm I. Classifying SSH Encrypted Traffic with Minimum Packet Header Features Using Genetic Programming[C]. GECCO (Companion) 2009:2539-2546.

[9] 李湘锋, 赵有健, 全成斌. 对称密钥加密算法在 IPsec 协议中的应用[J]. 电子测量与仪器学报, no. 2014(1):75-83.

[10] Wei Li, Marco Canini, Andrew W. Moore, Raffaele Bolla. Efficient Application Identification and the Temporal and Spatial Stability of Classification Schema[J]. Computer Networks (CN), no. 2009(6):790-809.

[11] Qu, Junhua, Tao Li and Fangfang Dang. Performance evaluation and analysis of OpenVPN on Android[C]. In Computational and Information

Sciences (ICCIS), 2012 Fourth International Conference, IEEE, 2012:1088-1091.

[12] 刘亚琼, 孟昭鹏, 王磊. 压缩算法在提高 VPN 性能中的研究[J]. 微处理机, no. 2007(2):36-37.

[13] 疏朝明, 宋宇波, 曹秀英. HMAC-SHA-1-96 算法在 VPN 中的应用[J]. 通信技术 no. 2001(6):12-14.

[14] Harkins Dan and Dave Carrel. RFC2409:The Internet key exchange[OL]. Internet Engineering Task Force, 1998. https://www.ietf.org/rfc/rfc2409.txt.

[15] 徐莉, 赵曦, 赵群飞. 利用统计特征的网络应用协议识别方法[J]. 西安交通大学学报, no. 2009(2):43-47.

[16] Kaufman C. RFC4306: Internet Key Exchange (IKEv2) Protocol[OL]. Internet Engineering Task Force, 2008. https://tools.ietf.org/html/rfc4306.

[17] Rescorla E. RFC2631: Diffie-Hellman Key Agreement Method, Internet Engineering Task Force[OL], 1999. https://www.ietf.org/rfc/rfc2631.txt.

[18] 张险峰, 秦志光. 椭圆曲线加密系统的性能分析[J]. 电子科技大学学报, no. 2001(2):144-147.

[19] 肖立国, 陈国良. 基于椭圆曲线密码体制的动态秘密共享方案[J]. 微电子学与计算机, no. 2002(1): 30-31.

[20] Anthony McGregor, Mark Hall, Perry Lorier, James Brunskill. Flow Clustering Using Machine Learning Techniques [C]. PAM 2004:205-214.

[21] Murat Soysal, Ece Guran Schmidt. Machine Learning Algorithms for Accurate Flow-Based Network Traffic Classification: Evaluation and Comparison[J]. Perform. Eval. (PE) , no. 2010(6):451-467.

[22] Laurent Bernaille, Renata Teixeira, Ismael Akodkenou, Augustin Soule, Kave Salamatian. Traffic Classification on the Fly[J]. ACM SIGCOMM Computer Communication Review, no. 2006, 36(2):23-26.

[23] 赵铭伟, 于晓晨, 江荣安. 一种动态口令身份认证协议的研究与改进[OL]. (2014).

[24] 郝玉洁, 冯银付, 赖攀. 基于指纹识别的 VPN 身份认证研究[J]. 计算机应用, no. 2009(2):350-352.

[25] Nobori, Daiyuu and Yasushi Shinjo. VPN gate: a volunteer-organized public VPN relay system with blocking resistance for bypassing government censorship firewalls[J]. Networked Systems Design and

Implementation. no. 2014(12):115-120.

[26] Deri, Luca and Richard Andrews. N2N: A layer two peer-to-peer vpn[C]. Resilient Networks and Services, Springer Berlin Heidelberg, 2008:53-64.

[27] 袁向英. 构建中型企业的分布式 VPN[J]. 网络安全技术与应用, no. 2012: 20-22.

[28] 王坤, 李建. MS-CHAPv2 密码分析[J]. 计算机工程与应用, no. 2002(38): 172-173.

[29] David Hulton. Divide and Conquer: Cracking MS-CHAPv2 with a 100% success-rate[OL].https://www.cloudcracker.com/blog/2012/07/29/cracking-ms-chap-v2/.

[30] 陈麟, 李焕洲, 胡勇, 戴宗坤. IPSec 通信截获与阻断系统研究[J]. 计算机应用研究, no. 2006(11): 193-194.

[31] 许辉, 周志洪, 李建华, 姚立红. 基于中间人攻击的 SSL VPN 运行参数深度检测[J]. 信息安全与通信保密 no. 2015(1):55-59.

[32] Ornaghi, Alberto, Marco Valleri. Man in the middle attacks Demos[OL], https://www.blackhat.com/presentations/bh-usa-03/bh-us-03-ornaghi-valleri.pdf

[33] Hoang Mavi, Nguyen Phong and Davar Pishva. Anonymous communication and its importance in social networking[C]. Advanced Communication Technology (ICACT), 2014 16th International Conference, IEEE, 2014:34-39.

[34] 吕承民, 谢永强, 李武, 李忠博, 殷璇. VPN 网络设计及性能分析[J]. 贵州师范大学学报:自然科学版, no. 2014(1):81-85.

[35] 李建荣. 集团客户 MPLS VPN 业务组网及性能测试研究[J]. 电信工程技术与标准化, no. 2014(10):77-80.

[36] 秦培斌, 肖志辉, 杨大川, 杨洋, 李希源. 基于多核处理器的加密卡异步并行驱动设计[J]. 通信技术 no. 2014(7):25-28.

[37] Cheng, Ke-qin, Bo Yu and Jian Zhou. The design and implement of access system based on OPENVPN[J]. Xiamen Univ, no. 2007(1):23-26.

[38] 武衡. 基于 OpenVPN 的远程控制环境设计与实现[D]. 苏州大学硕士论文, 2014.

[39] 占旻, 李沁. 虚拟计算环境中的分布式虚拟网管技术[J]. 计算机工程, no. 2009,35(18):79-81.

[40] 肖凌. 面向无线接入的 IPSec VPN 关键技术研究. 华中科技大学博士论文[D], 2009.

[41] 叶润国, 冯彦君, 虞淑瑶, 宋成. 一种基于 WTLS 的轻型移动 VPN 方案[J]. 微电子学与计算机 no. 2005(4):33-37.

哈尔滨工业大学学位论文原创性声明和使用授权说明

学位论文原创性声明

本人郑重声明：此处所提交的学位论文《面向企业用户的高性能 VPN 系统的设计与实现》，是本人在导师指导下，在哈尔滨工业大学攻读学位期间独立进行研究工作所取得的成果，且学位论文中除已标注引用文献的部分外不包含他人完成或已发表的研究成果。对本学位论文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。

作者签名：孙云霄 日期：2015年10月15日

学位论文使用授权说明

学位论文是研究生在哈尔滨工业大学攻读学位期间完成的成果，知识产权归属哈尔滨工业大学。学位论文的使用权限如下：

（1）学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文，并向国家图书馆报送学位论文；（2）学校可以将学位论文部分或全部内容编入有关数据库进行检索和提供相应阅览服务；（3）研究生毕业后发表与此学位论文研究成果相关的学术论文和其他成果时，应征得导师同意，且第一署名单位为哈尔滨工业大学。

保密论文在保密期内遵守有关保密规定，解密后适用于此使用权限规定。

本人知悉学位论文的使用权限，并将遵守有关规定。

作者签名：孙云霄 日期：2015年10月15日

导师签名：[illegible] 日期：2015年10月15日

致　谢

在本论文完成之际，我要向曾经在学业上指导、帮助过我的老师、同学表示衷心的感谢，也对一直关心、支持我的家人表示感谢，我硕士论文的顺利完成离不开他们。

首先向我的导师权光日教授表示衷心的感谢，在整个研究生学习期间，权老师对我进行了耐心地指导和帮助。在论文开题阶段，权老师与我深入的讨论了课题所涉及到的技术难点，并鼓励我要勇于探索，既要提出创新的解决问题的方法，又要注重理论与实际的联系，这使得本课题研究更具有应用价值。权老师专业知识渊博，治学作风严谨，他对科研的孜孜不倦的精神令我十分敬佩，成为我学习的一面旗帜。

感谢我所在的实习单位北京赛思信安技术有限公司的副导师王树鹏高级工程师，在我的课题研究过程中，王老师从需求调研、系统设计、系统研发等多个环节对我进行指导，传授了我许多宝贵的经验，正是由于王老师的指导和帮助，我才能够顺利完成 VPN 系统的设计与研发任务。实习期间的经历与收获对我以后的工作和学习有非常大的帮助。

特别感谢哈尔滨工业大学(威海)网络技术研究所的王佰玲老师。时至今日，我跟随王老师学习工作已近五年，是他指引我在学业和人生道路上做出正确的选择。今后我一定会努力工作，协助王老师将科研团队发展壮大。

感谢刘扬老师、陈彬老师、孙玉山老师、季振洲老师对我的论文提出了宝贵的修改意见，感谢王昆老师、苑新玲老师对我学业的帮助。

最后，要感谢所有论文评审专家和答辩评委们的辛勤工作。

个人简历

1989 年 8 月出生于山东省潍坊市。

2008 年 9 月考入哈尔滨工业大学（威海）计算机科学与技术学院信息安全专业，2012 年 7 月本科毕业并获得工学学士学位。

2012 年 9 月——至今，在哈尔滨工业大学（威海）软件学院攻读软件工程硕士学位。

工作经历：

2012 年 7 月——至今，在哈尔滨工业大学（威海）网络技术研究所工作。主要负责信息安全及信息对抗领域关键技术的研究及工程应用推广。参与多项国家信息安全专项课题。