基于 openEuler 的校园网 VPN 系统的设计与实现*

谢富荣　林宁　　左悦**

南宁学院信息工程学院，南宁 530200　　南宁学院土木与建筑工程学院，南宁 530200

摘　要　校园网是为学校教学、科研运作和生活服务专用的计算机网络，高校数字化转型不断加快的背景下，远程教学、跨校区科研协作，校外资源访问变得司空见惯。传统 VPN 存在安全风险，运作也比较复杂。openEuler 属于国产高性能操作系统，它具备安全性、稳定性，其开源生态为校园网 VPN 系统构筑了可靠根基，校园网 VPN 遭遇外部打击和内部威胁时，传统方案因缺少精准访问控制与集中化运作能力，造成运维效率差。要创建依靠 openEuler 的 VPN 系统，经由 OpenVPN 创建 SSL/TLS 加密隧道来做到多校区的数据传送，用 CA 证书双向认证加上 DH 密钥交换去防止中间人侵袭；安排 JumpServer 跳板机，凭借它的 4A 经营体系达成用户权限分类，操作日志全面跟踪，再同 OpenVPN 联合起来成为双重安全保障，这个系统靠 openEuler 的高安全内核以及 Virtio - Net 虚拟化技术，完成了跨校区内网资源的完美互通，LDAP 动态身份核查以及操作全过程审查，最终既保证了合规性，又优化了加密传送效率，削减了运维成本，为高校塑造起高效又安全的数字化网络根基。

关键字　校园网，VPN，OpenEuler，Jumpsever

The Design and Implementation of Campus Network VPN System Based on openEuler*

Xie FuRong　LinNing　　ZuoYue**

College of Information Engineering
Nanning University
Nanning 530200 China

College of Architecture and Civil Engineering
Nanning University
Nanning 530200 China

Abstract—The campus network is a dedicated computer network for school teaching, research, operation and life services. With the accelerating digital transformation of universities, remote teaching, inter-campus research collaboration, and access to external resources have become commonplace. Traditional VPNs have security risks and operate in a relatively complex manner. openEuler is a domestically developed high-performance operating system that offers security, stability, and an open source ecosystem which provides a reliable foundation for campus network VPN systems. When campus network VPNs face external attacks and internal threats, traditional solutions lack precise access control and centralized operational capabilities, resulting in poor operational efficiency.To create a VPN system relying on openEuler, establishing multi-campus data transmission through an SSL/TLS encrypted tunnel using OpenVPN is essential. Implementing CA certificate mutual authentication and DH key exchange is crucial to prevent man-in-the-middle attacks. Additionally, deploying JumpServer as a bastion host enables user permission classification through its 4A management system, comprehensive operation log tracking, coupled with OpenVPN for dual security protection. This system leverages openEuler's high-security kernel and Virtio-Net virtualization technology to achieve seamless interconnection of resources across campuses while ensuring compliance with LDAP dynamic identity verification and full-process auditing capabilities. Ultimately, this approach not only ensures compliance but also optimizes encryption efficiency by reducing operational costs—thus creating an efficient and secure digital network foundation for higher education institutions.

Key words—Campus Network,　VPN,　OpenEuler,　Jumpsever

1　引　言

高校信息化推进过程中，校外访问校园网资源的需求和网络安全之间的矛盾愈加明显，传统 VPN 技术可做到远程接入，但它存在配置复杂度过高，权限管理机制薄弱以及安全风险较大等问题，无法适应多校区协同，移动教学和科研协作等动态化需求，基于此，本研究把 openEuler 操作系统当作底层框架，融合 OpenVPN 加密通信以及 Jumpserver 统一运维管理平台，力求创建出具有更高安全性，更强易用性和更好扩展性的新型校园网 VPN 系统，从而为教育的数字化转型供应自主可控的方案[1]。

*基金资助：南宁学院 2025 年横向课题（2025HX004）资助。
**通讯作者：左悦, zuoyue@unn.edu.cn

从开发目的来讲，该系统重点解决三个主要矛盾，第一，传统 IPSecVPN 存在部署困难，跨平台兼容性不佳的情况，依靠 OpenVPN 的 SSL/TLS 协议动态加密及多协议适配能力，可以让师生用移动终端流畅访问图书馆数据库，科研平台等校内资源，满足居家备课，校外实验等各种场景的需求。第二，传统端口映射和弱密码策略会引发数据泄露风险，借助 openEuler 的强制访问控制以及内核级安全防护，再加上 Jumpserver 的权限模型和操作审查功能，创建起牵涉身份认证，数据传送，资源调用的全流程保护机制，从而减小数据泄露风险[5]。 第三，多校区存在网络资源分散的情况，利用 OpenVPN 的虚拟隧道技术达成跨校区网络相互关联，这有益于实验室设备的远程控制，也能做到教务系统的同步访问，从而打破地理限制给教学科研造成的约束。

从应用价值看，系统利用技术更新来重构教育生态。其一，它经由企业微信，Web 端等多入口接入，简化操作步骤，师生无需专业设置就能用手机，平板等设备安全获取校内资源，这有益于慕课，虚拟仿真实验等新教学模式的推广。其二，凭借 Jumpserver 的集中化运维模块，系统可批量预设千人规模的用户权限，并随时分析日志，相比于传统运作模式，既减轻大量重复性工作，又明显改善 IT 管理效率，更为重要的是，采用国产化技术栈代替原先依赖国外操作系统的做法，给教育行业关键信息基础设施的自主可控供应应用案例，减小因技术封锁而产生的系统风险。

该系统以低成本，高可用的技术路径推进教育公平，偏远地区的师生经由加密隧道即可平等浏览优质学术资源，跨校区的科研团队依靠虚拟专网就能达成数据协同分析，从而推动产学研深度融合，其分层防御架构还会给智慧校园、电子政务等领域的网络安全创建赋予参照范式。

2 相关技术研究

2.1 OpenVPN 技术

VPN，其全称为 Virtual Private Network（虚拟专用网络），这是一种依托 ISP 以及其他 NSP，于公共网络当中形成起专用数据通信网络的技术[2]，它可向企业之间或者个人与企业之间赋予安全的数据输送通道服务。在 VPN 里，任意两点之间的结合并不存在传统专网所必要的那种端到端物理链路，而是由公共网络资源动态合成的，可以把它看作经由私有的隧道技术在公共数据网络上模仿出来的具备同专网一样功能的点到点专线技术，这里所说的“虚拟”就是指无需铺设实际的长途物理线路，只是利用公共的 Internet 网络来达成。

OpenVPN 作为 Linux 下开源 VPN 的先驱，具备优异的访问性能与友好的用户图形界面。

OpenVPN 是一款用来塑造虚拟专用网络加密通道的软件包，其同意加入创建 vpn 的节点采用预置的私钥，第三方证书或者用户名/密码执行身份认证，该软件较多地运用了 OpenSSL 加密库以及 SSLv3/TLSv1 协议，openvpn 可在 linux，windows 系统中运行，这是一款包含服务器端和客户端的软件，并非基于 web 的 vpn 应用，也与 ipsec 及其他 vpn 软件包不相容。

（1）OpenVpn 依赖的 SSL 与 TLS 协议

SSL 即安全套接层，它作为一种安全协议而存在，其产生旨在给予网络通讯以安全保障并维持数据的完整性。SSL 会针对传输层中的网络通信执行加密操作，该协议利用公开密钥技术来确保两个应用程序之间通信的机密性与可靠性，这样就能防止客户端和服务器端的应用通信内容被不法分子窃取。如今的 Web 浏览器大多把 http 和 ssl 关联起来以达成安全通信的目标，TLS 可以看作是 SSL 协议的升级版，其功能较 SSL 更为强大。

（2）OpenVpn 的加密通信原理过程

OpenVpn 依靠 TLS 加密时利用公开秘钥对即非对称秘钥包含公钥与私钥来开展数据加密，其一，Server 和 Client 需持有同一个 CA 所签发的证书，二者经由交换证书去验证彼此是否合法从而决定能否创建 VPN 关联[4]。其二，利用对方的证书授权（CA），将己方当前采用的数据加密方式进行加密后，发送至对方，鉴于这是经对方 CA 加密的，仅有与之对应的私钥才可解密该字符串，这样就保障了此秘钥的安全，而且这个秘钥会定时变更，对于窃听者而言，很可能尚未破解出秘钥，通信双方就已更新秘钥。

（3）OpenVpn 的多种身份验证方式

OpenVPN 具备多种身份认证手段，这些手段可用来识别即将执行对接的双方的身份，其中涵盖预先共享的私钥，第三方证书以及用户名/密码搭配等方法，预先共享私钥这种方式操作起来较为简便，但它仅仅适合创建点到点形式的 VPN 网络，依靠 PKI 技术的第三方证书虽然功能十分完备，不过却得耗费更多精力来守护这个 PKI 证书系统。采用用户名/密码搭配的认证形式时，可以免除客户端预先设置共享私钥这一步骤，但是依然要有一份服务器 CA 证书用于加密处理，而最为常见的做法则是把用户名/密码同 CA 证书结合起来使用，另外像 LDAP 或者利用域控制器做集中认证之类的方式也是很不错的选择。

（4）OpenVpn 的通讯原理

OpenVPN 的全部通信均依赖单个 IP 端口（默认为 1194），其默认采用 UDP 协议执行通讯，不过也支持

TCP 协议，建议使用 TCP 协议。

OpenVPN 服务器具备向客户端发送部分网络设置信息的能力，这里面包含 IP 地址，路由设置之类的信息。

OpenVPN 技术核心在于虚拟网卡及 SSL 协议的实现。

2.2 Jumpserver 堡垒机

堡垒机处于特定的网络环境之中，要保护网络与数据免受外部和内部用户的入侵与破坏，会随时采集并监测网络环境各个组件的系统状态，安全事件，网络活动，这样就能集中报警并及时应对[4]。从本质上说，堡垒机可当作用于防守攻击的计算机，也叫“堡垒主机”，堡垒机是一种主机系统，它自己往往经过了一些加固处理，安全性较高，可以抵挡某些攻击。堡垒机把需保护的信息系统资源同安全威胁源隔离开来，于是就在被保护资源前方创建起牢固的“堡垒”，既能阻挡威胁，又不会妨碍普通用户正常访问资源，堡垒机还结合了行为审查和权限管理，从而提升了对操作和安全方面的把控能力。

JumpServer 属于开源堡垒机范畴，主要面向企业给予安全又高效的运维访问运作方案，它的核心功能牵涉身份认证，权限控制，操作审查这些重要方面，而且支持多协议资产管理并适配多云环境，是数字化转型期间捍卫系统安全的关键工具。

2.3 openEuler 系统

openEuler 操作系统以 Linux 内核为根基开展深度改良，经由对编译系统，网络协议栈以及文件存储等核心模块的架构重新塑造，达成了性能和稳定性两方面的优化效果，它的革新性表现在三个层次：第一，从技术架构角度看，采纳分层设计思路，适配鲲鹏之类多种类别的处理器架构，利用动态复合页技术让内存写入性能增加一倍；第二，就智能调度而言，运用 AI 决策引擎，凭借 sysHAX 模块改善 CPU 调度算法，促使 vLLM 算子下放速度加快，业务吞吐量大增；第三，在生态营造上，具备包含服务器、云计算、边缘计算以及植入式场景的全栈扶持能力，2023 年装机量达到 660 万套，并且已经在金融、能源等重要行业做到大规模推广应用[5]。openEuler24.03LTS 版本把 oeAware 智能运维平台同 A－Tune 自动调优系统整合，创建起包含“感知-决策-执行”环节的智能运维循环，这很值得注意。

3 校园网系统基础架构

（1）VPN 系统设计

本系统的设计属于校园网系统的关键组成部分，系统依托 SSL/TLS 协议的安全隧道技术，为校园网 VPN 系统达成端到端的加密通信服务。VPN 系统架构如图 1 所示。

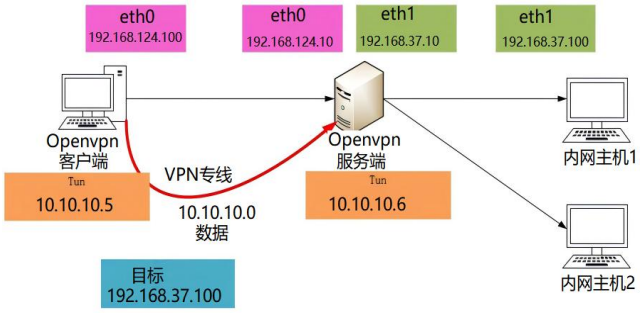

图 1 VPN 系统架构图

（2）Easy-rsa 证书工具设计

用户创建自签名根证书当作信任锚点，按照预置参数自动签发服务端/客户端证书请求，保证证书主题字段同预置组织信息相符，以此形成起完整的信任链，可以满足 TLS 双向认证或者加密通信场景中的身份验证需求。Easy-rsa 工具证书拓扑图如图 2 所示。

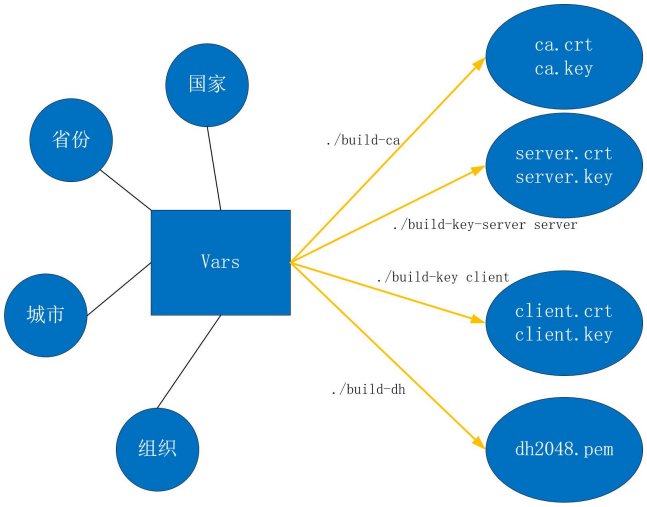

图 2 Easy-rsa 工具证书拓扑图

（3）Jumpserver 系统的实现

Jumpserver 系统属于校园网系统的关键形成局部，它重点围绕身份认证、授权控制、账号管理、审查运维等大量方面发挥作用，以适应整个校园网 VPN 系统处于混合 IT 环境时产生的安全运维需求。

4 VPN 系统的部署方案实现

本方案基于 OpenVPN 工具进行进行研究开发，使用证书工具（easy-rsa）来实现身份认证流程，其方案包括代码构建、打包和部署等环节。

4.1 证书制作

先下载 Easy－RSA 工具包再部署，配置 vars 环

境变量来规定证书元数据（涉及国家，省份，组织等），接着初始化 PKI，形成私钥存储，证书请求以及签发目录结构，之后创建自签名根证书当作信任锚点，按照预置参数自动签发服务端/客户端证书请求，保证证书主题字段同预置组织信息相符，这样就形成起完整的信任链，可以满足 TLS 双向认证或者加密通信场景中的身份验证需求。

证书制作步骤如下：

① 基础环境配置

```
# service iptables stop //关闭防火墙
# setenforce 0        // selinux 安全策略
```

② 安装 Easy-rsa

```
#wget https://gitcode.net/mirrors/OpenVPN/easy-rsa/-/archive/master/easy-rsa-old-master.tar.gz
# tar -zxvf easy-rsa-old-master.tar.gz //解压软件包
# mv easy-rsa-old-master easy-rsa
```

② 配置 vars

```
vim vars
export KEY_COUNTRY="CN"      #表示国家
export KEY_PROVINCE="GuangXi"    #省份
export KEY_CITY="NanNing"        #城市
export KEY_ORG="Canzy "      #组织名称
export KEY_EMAIL=" Canzy@qq.com" #邮箱
export KEY_CN=
export KEY_NAME= " Canzy"        #名字
export KEY_OU=                   #部门名字
```

③ 初始化环境

```
# 环境初始化
source ./vars
****************************************
No  /root/easy-rsa/easy-rsa/2.0/openssl.cnf  file could be found
  Further invocations will fail
****************************************
NOTE: If you run ./clean-all, I will be doing a rm -rf on /root/easy-rsa /easy-rsa/2.0/keys
./clean-all
# 证书生成
./build-ca                  # 生成 CA 证书
./build-key-server server        # 生成服务器证书
./build-key client          # 生成客户端证书
./build-dh                  # 生成交换密钥证书
```

4.2 服务端配置

安装 OpenVPN，进行文件复制操作，把 PKI 所产生的根证书、服务端证书私钥以及 Diffie - Hellman 交换参数复制到 OpenVPN 服务端设置目录，以使服务启动的时候可以准确加载 TLS 通信必要的密钥。修改 OpenVPN 服务端的配置文件，要指定 CA 证书，服务端证书/私钥以及 DH 参数的路径，还要定义 VPN 虚拟子网的地址池和客户端的路由推送规则，接着启用 TLS - Auth 来提升握手的安全性，再选择高强度的加密算法保证数据传递的机密性，这样就能把密钥材料和网络参数整合到服务端，使得创建 VPN 隧道的时候可以正常加载证书链，开展加密协议以及分发路由策略。

服务端配置步骤如下：

① # mkdir /etc/openvpn/server/ //创建一个存放证书目录

② 复制证书文件到证书目录中

```
cp /root/easy-rsa-old/easy-rsa/2.0/keys/ca.crt /etc/openvpn/server/
cp /root/easy-rsa-old/easy-rsa/2.0/keys/server.crt /etc/openvpn/server/
cp /root/easy-rsa-old-/easy-rsa/2.0/keys/server.key /etc/openvpn/server/
cp /root/easy-rsa-old-/easy-rsa/2.0/keys/dh2048.pem /etc/openvpn/server/
```

③ 修改服务端的配置文件

```
ca keys/ca.crt
cert keys/server.crt
key keys/server.key
dh keys/dh2048.pem
# 定义 TLS/SSL 证书路径
server 10.10.10.0 255.255.255.0  # 定义 VPN 虚拟网络的子网范围
push "route 192.168.37.0 255.255.255.0"  # 修改推送路由信息
tls-auth ta.key 0          # 启用 TLS 认证（TLS-Auth）
cipher AES-256-GCM      # 指定数据传输的加密算法
openvpn --genkey --secret ta.key #  生成 ta.key 文件
```

④ 开启内核 IP 传递，启动 OpenVPN 并实施监听

```
vim /etc/sysctl.conf

net.ipv4.ip_forward=1        # 修改内核转发路由
sysctl -p                # 生效内核配置文件
openvpn --daemon --config /etc/openvpn/server.conf
# 启动 openvpn
netstat -lntup | grep 1194  # 检查启动是否正常
udp 0 0 0.0.0.0:1194 0.0.0.0:*    2167/openvpn
```

4.3 客户端配置

创建客户端配置目录，并把模板文件复制进去，接着对 client.conf 执行配置，指定好远程服务器的 IP 和端口，明确证书路径以及 TLS - Auth 密钥，然后同步服务端加密算法以保证协商一致，这样客户端在连接的时候就可以凭借证书双向认证来形成加密隧道，从而做到客户端经由 UDP1194 端口与服务器展开安全通信。

客户端配置步骤如下：

① 创建一个目录用于存放客户端的文件

```
mkdir openvpn_client
cp /etc/openvpn/client/client.conf /root/openvpn_client/
vim /root/openvpn_client/client.conf
```

② 修改客户端配置

```
remote 192.168.124.100 1194
# 修改连接的服务器地址和端口，该实验的外网服务器的公网地址为 192.168.124.100，端口为 1194
ca ca.crt
cert client.crt
key client.key
tls-auth ta.key 1       # 证书
cipher AES-256-GCM # 指定数据传输的加密算法
```

③ 将服务端生成的客户端证书复制至客户端配置目录，确保客户端连接时能正确加载身份验证。

```
cp /root/easy-rsa/easy-rsa/2.0/keys/client.crt /root/openvpn_cilent/
cp /root/easy-rsa/easy-rsa/2.0/keys/client.key /root/openvpn_cilent/
cp /root/easy-rsa/easy-rsa/2.0/keys/ca.crt /root/openvpn_cilent/
cp /etc/openvpn/server/ta.key /root/openvpn_cilent/
cp client.conf client.ovpn
zip client.zip ./openvpn_cilent/*
```

4.4 Jumpserver 系统部署

Jumpserver 系统属于校园网系统的关键形成局部，它重点围绕身份认证、授权控制、账号管理、审查运维等多个方面发挥作用，以满足整个校园网 VPN 系统处于混合 IT 环境中产生的安全运维需求。

利用 Docker 容器，采用一键部署：

```
curl -sSL https://resource.fit2cloud.com/jumpserver/jumpserver/releases/latest/download/quick_start.sh | bash
cd jumpserver
./jmsctl.sh start
```

4.5 Web 服务器部署

Web 服务器配置部署步骤如下：

① 安装 HTTPD 服务

```
yum search httpd
yum install httpd.x86_64 -y
```

② 启动和检查 HTTPD 服务

```
Systemctl start httpd
Systemctl status httpd
```

③ 配置防火墙

```
firewall-cmd --permanent --add-port=80/tcp
firewall-cmd --permanent --add-service=http
firewall-cmd - reload
```

④ 编辑 HTTPD 配置文件

```
cd /etc/httpd/conf
cp httpd.conf httpd.conf.bak
vi httpd.conf
<Directory />
AllowOverride none
#Require all denied
Require all granted     #配置浏览器访问的网页根目录
<Directory />
DocumentRoot “/var/www/html”    #开放访问/var/www 目录的权限
Require all granted
Options Indexes FollowSymLinks        #如果该虚拟目录下没有 index.html,浏览器会显示该虚拟目录的目录结构,列出该虚拟目录下的文件和子目录
DirectoryIndex index.html             #设置浏览器默认访问的网页为 index.html
```

4.6 FTP 服务器部署

FTP 服务器实现步骤如下：

① 搭建 FTP 服务器

```
yum install vsftpd -y
rpm -qi vsftpd
```

② 启动 vsftpd 软件

```
service vsftpd start
Redirecting to /bin/systemctl start vsftpd.service
ps aux | grep vsftpd
root 2150  0.0  0.0  6820  412 ?  Ss  10:52  0:00 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf
root  2205  0.0  0.0 22108   2100 pts/0   S+  10:56   0:00 grep --color=auto vsftpd
netstat -anplut | grep vsftpd
```

```
tcp6  0  0 :::21 :::* LISTEN      2150/vsftpd
```

③ 登录 ftp 服务

```
yum install ftp lftp -y
useradd canzy
echo 123|passwd canzy --stdin
# 更改用户 sc 的密码
passwd：所有的身份验证令牌已经成功更新。
lftp -u canzy,123 192.168.124.104
lftp canzy@192.168.124.104:~ ls
lftp canzy@192.168.124.104:~ pwd
lftp canzy@192.168.124.104:~mkdir test
```

5 结束语

本研究着眼于高校信息化进程里校外访问受限制和网络安全这一矛盾，革新性地设计并实现了包含openEuler 操作系统、OpenVPN 加密隧道以及Jumpserver 堡垒机的新型校园网 VPN 系统，为满足“校区-家庭-移动”三种场景的需求提供参考价值。

参 考 文 献

[1] 何榆锋,林宁. 基于Linux的高可用服务器集群架构设计与实现[J].计算机技术与教育学报,2023:2-10.DOI:10.26914/c.cnkihy.2023.109380.

[2] 杨文彬.高校网络安全分析与防护[M].湖南科学技术出版社:202301:146.

[3] 张先辉.基于OpenVPN的广播电视远程监测系统搭建与实现[J].中国传媒科技,2022,(06):151-153.DOI:10.19483/j.cnki.11-4653/n.2022.06.045.

[4] 曹园青.基于JumpServer的DevOps安全管控机制研究[J].现代信息科技,2023,7(22):69-72.DOI:10.19850/j.cnki.2096-4706.2023.22.015.

[5] 陈俊彦,俸皓.基于openEuler的Linux系统应用课程实践教学设计研究[J].科技资讯,2024,22(09):214-217.DOI:10.16661/j.cnki.1672-3791.2401-5042-9452.

[6] 黄建桥,林宁. Linux系统自动化运维管理平台实现及应用[J].计算机技术与教育学报,2023:18-24.DOI:10.26914/c.cnkihy.2023.109382.