

proj235-ota-upgrade-system

说明书

选题： pro_j235

学校： 中北大学

队员： 冯琦雁 刘晓敏 郭冰芯

指导老师： 张静

1. 项目描述

在工业控制、机器人控制领域中越来越多使用 Linux 嵌入式操作系统，但嵌入式 linux 系统在终端设备上部署之后，在更新和升级系统镜像的时候，还需要重新物理部署，不能确保最佳的操作体验和系统正常的运行时间。为了改善 linux 的升级体验，可以参考 ostree，改造 linux 的升级系统，以能够执行安全的远程 OTA 升级，执行安全的离线升级。

2. 项目目的

第一题：升级系统的升级功能实现

1. 能够支持 linux 主要系统文件 (initrd, kernel, rootfs) 等 linux 重要文件的版本升级
2. 能够保证升级前后用户应用/配置的一致性
3. 能够保证升级异常时，检测到异常，并可以回退到系统可用状态

第二题：升级系统基础框架功能实现

以题目一为基础，升级系统需要追加下列功能：

1. 实现升级系统管理后台，主要功能包括：
 - (1) 实现升级包管理，支持用户上传升级包，支持升级包校验
 - (2) 实现升级任务管理，支持用户提交升级任务，支持升级任务

状态查询

(3) 实现升级日志管理, 支持用户升级日志查询

(4) 实现用户管理, 支持用户登录、注册、权限管理

2. 实现升级系统客户端功能, 主要功能包括:

(1) 支持升级包校验、下载、安装功能, 上报升级状态等

(2) 支持静默后台强制升级和自动检测升级功能

(可选) 第三题: 升级模块扩展功能实现

以题目二为基础, 升级系统需要追加下列功能:

1. 升级功能要有更健壮的异常保护机制, 要能兼容电池不足, 空间不足, 升级中掉电, rootfs/initrd/kernel 完全坏掉等情况

2. 升级功能要有功能能保证用户应用/配置的兼容性

3. 升级功能要能支持 uboot 和 grub 两种启动方式

4. 升级功能要能支持差分升级, 使升级包尽可能小

5. 升级系统管理后台要有终端管理, 能查询终端的版本及硬件信息

6. 升级系统客户端要有安全机制, 防止升级后台、升级包等被篡改

3. 环境搭建:

3.1 开发环境: 前端 Vue Element-UI Vite TypeScript

3.2 后端: go lang grpc

3.3 数据库: mysql

3.4 服务器设备端：ubuntu 24.04 依赖 unzip
dpkg-scanpackages

3.5 在开发环境中准备好需要升级的 Linux 系统版本。

3.6 配置好版本控制工具，确保每一步操作都可以被跟踪和回滚。

4. 实现系统文件升级机制：

4.1 使用 apt 工具实现对内核(kernel)引导文件的备份和替换。

4.2 确保升级过程中的数据完整性，使用软连接替换当前内核为最新内核

5. 应用和配置的一致性：

通过脚本或自动化工具，验证应用配置文件的有效性，保证其与升级后的系统匹配。

6. 错误处理和恢复机制：

6.1 涉及软连接，在升级回退时将软连接连接到对应内核版本

6.2 设计并实现监控升级过程的工具，一旦检测到异常，立即中止升级并恢复到原始状态。

6.3 开发日志记录和分析工具，记录升级过程中可能出现的错误，方便后续调试。

7. 升级系统管理工具的实现：

7.1 开发工具界面：设计一个界面，用户可以上传升级包并进行

相关操作。可以使用 Web 框架开发一个管理界面。

7.2 实现升级包的上传和校验：编写上传接口，并使用固定规则对上传的升级包进行完整性校验。

7.3 任务管理和状态查询：实现后台任务管理模块，支持升级任务的提交、状态查询和日志记录。

7.4 权限管理：设计并实现权限管理模块，控制不同用户的操作权限，如上传、发布、查看日志等。

8. 升级系统客户端功能的实现：

8.1 客户端模块：开发一个轻量级客户端程序，可以通过网络从管理工具处获取并下载升级包。

8.2 本地校验和安装：在客户端本地对升级包进行校验，确保包的完整性和正确性。然后自动执行安装操作。

8.3 报告功能：实现一个报告系统，将客户端的升级状态实时上传至管理端，供管理员查看。

9. 数据库的设计

9.1 数据字典

表 tasks

序号	字段名	描述
1	id	编号
2	created_at	创建日期
3	updated_at	更新日期
4	deleted_at	删除日期
5	status	状态
6	type	类型
7	args	
8	host_ip	Ip 地址

表 roles

序号	字段名	描述
1	name	角色（管理员和普通用户等）

表 hosts

序号	字段名	描述
1	avator	用户头像 url 地址
2	username	用户登录时的用户名
3	nickname	用户昵称
4	Access_token	访问令牌此登陆会话的安全信息
5	Refresh_token	请求授权成功时获取的刷新令牌
6	Expires_At	账号到期时间
7	password	用户登录的密码

表 kernels

序号	字段名	描述
1	Id	内核标识
2	Created_at	创建时间
3	Update_at	修改时间
4	Deteleted_at	删除时间
5	Version	版本号
6	Tested	测试状态
7	its	

表 permissions

序号	字段名	描述
1	Name	许可证名称

表 role_permissions

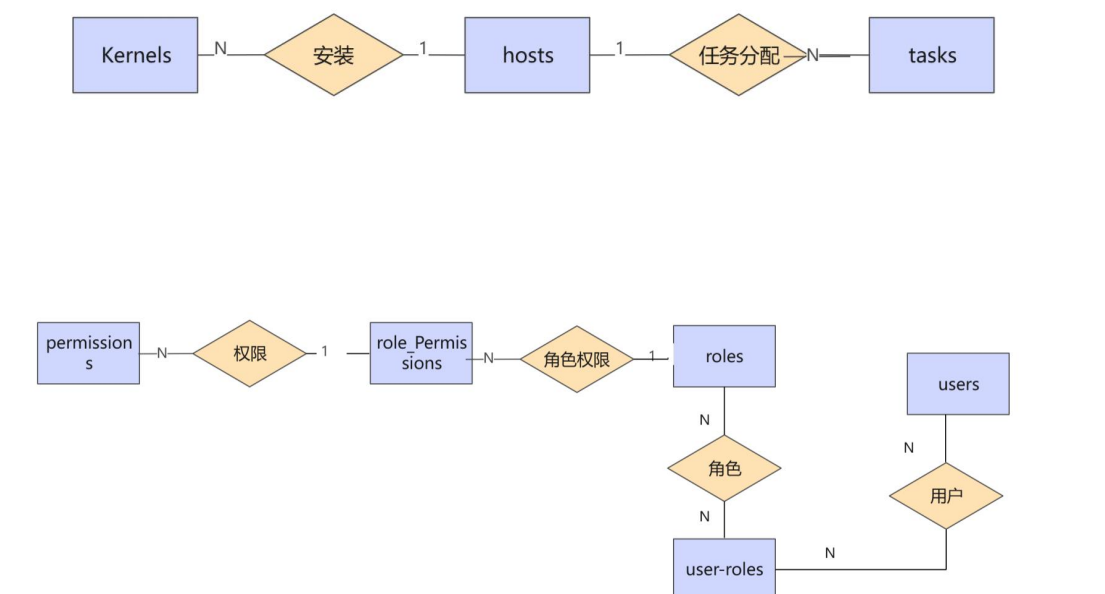
序号	字段名	描述
1	Role_name	角色类型
2	Permission_name	许可证名称

表 user

序号	字段名	描述
1	avator	用户头像 url 地址

2	username	用户登录时的用户名
3	nickname	用户昵称
4	Access_token	访问令牌此登陆会话的安全信息
5	Refresh_token	请求授权成功时获取的刷新令牌
6	Expires_At	账号到期时间
7	password	用户登录的密码
8	avator	用户头像 url 地址

9. 2E-R 图



10. 运行结果

10.1 管理员界面

1. 内核管理：在该页面会显示当前系统的名称，内核版本，IP 地址，Agent 运行状态，状态更新时间等等。当更新完内核之后，需重新启动虚拟机内核版本才会发生改变。可选择上传文件包进行升级，升级包有固定的格式 XX.XX.XX，方便比较所升级内核版本是否准确和排除上传为较低内核版本。当前版本如若高于上传的升级包，则不给予选择。升级过程中如遇到中断或磁盘空间满异常情况停止升级，软件包升级过程会将引导 initrd 保留，新软件包安装后通过软链接

链接新的 initrd 和 kernel 版本。

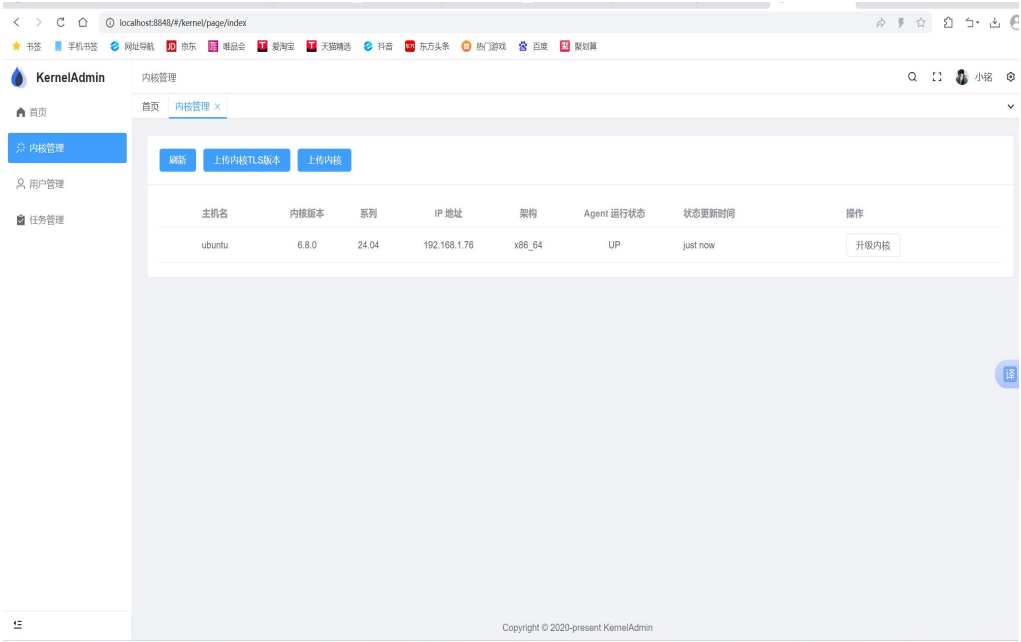


图 10.1 内核管理界面

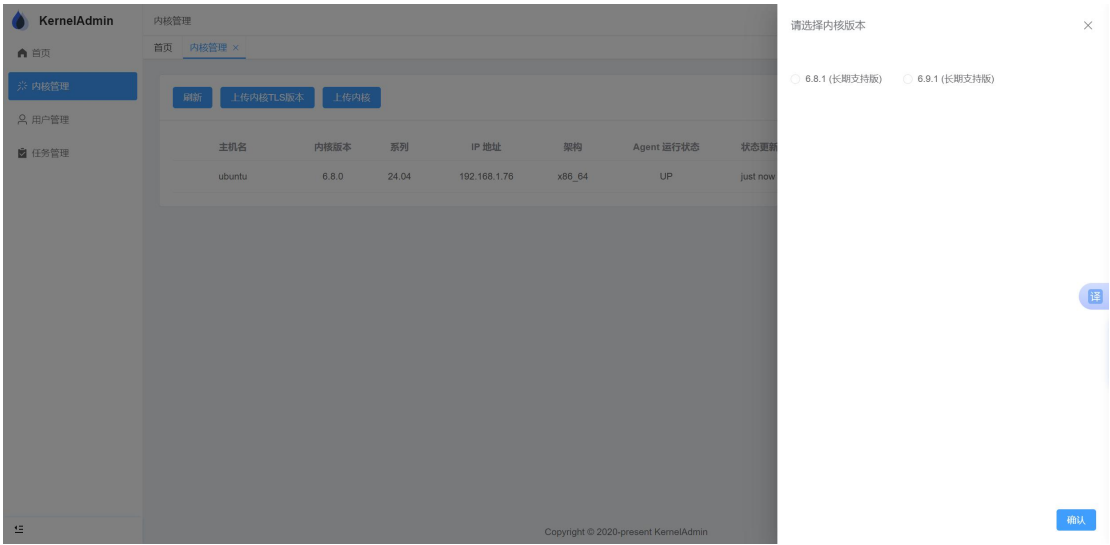


图 10.2 内核选择界面

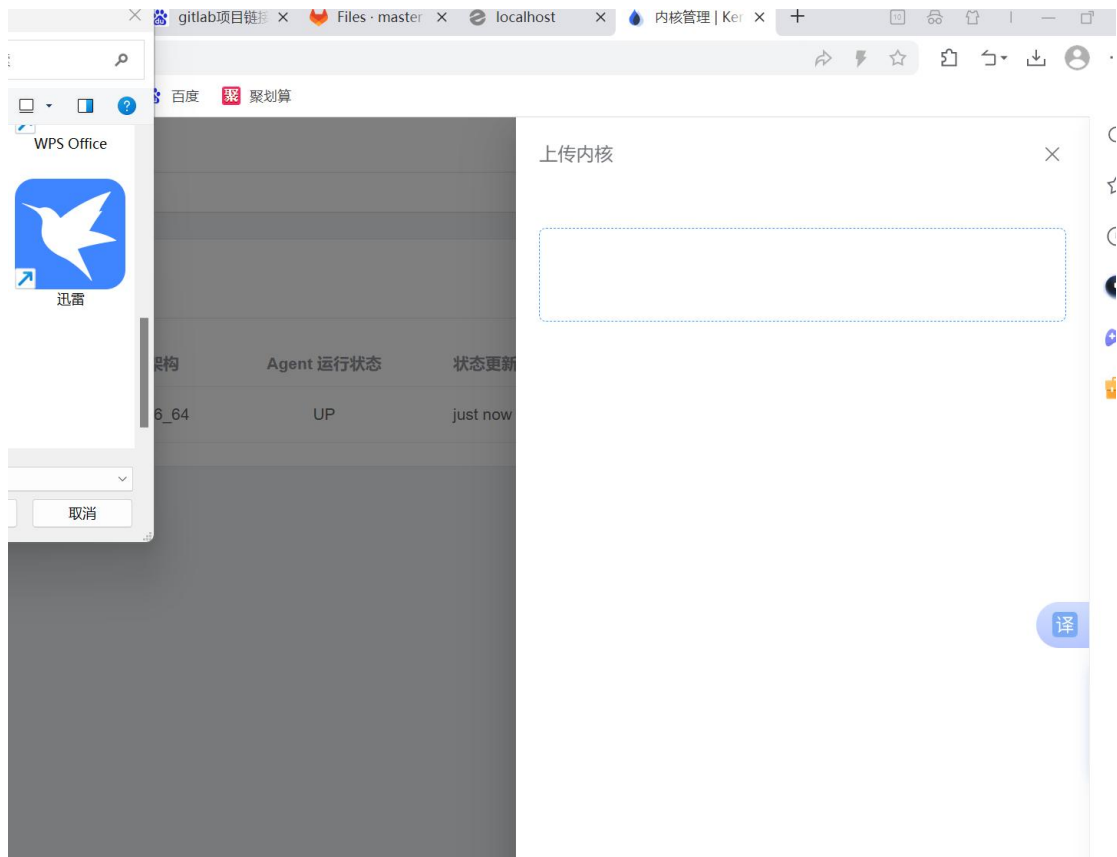


图 10.3 内核上传界面

2. 用户管理：拥有管理员权限的用户可以有注册、删除、修改用户的权限。

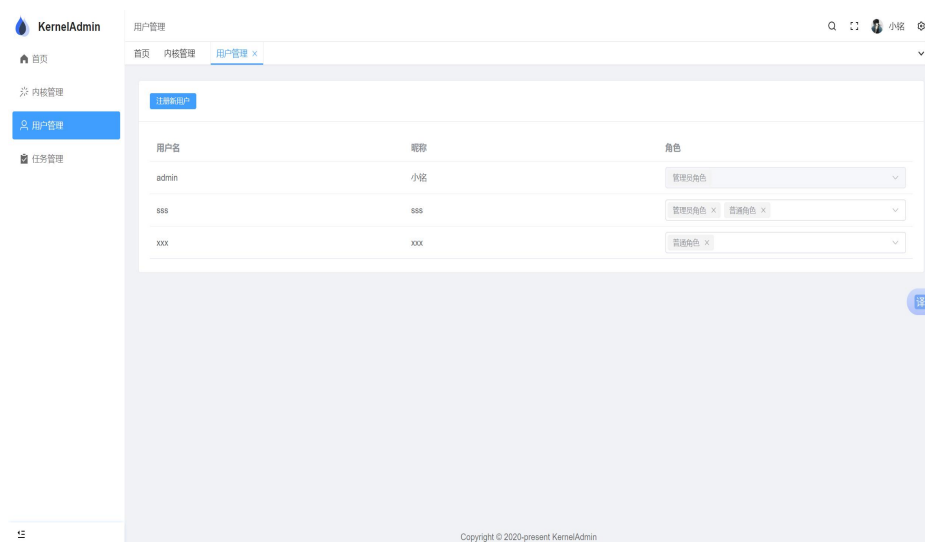


图 10.4 用户管理界面

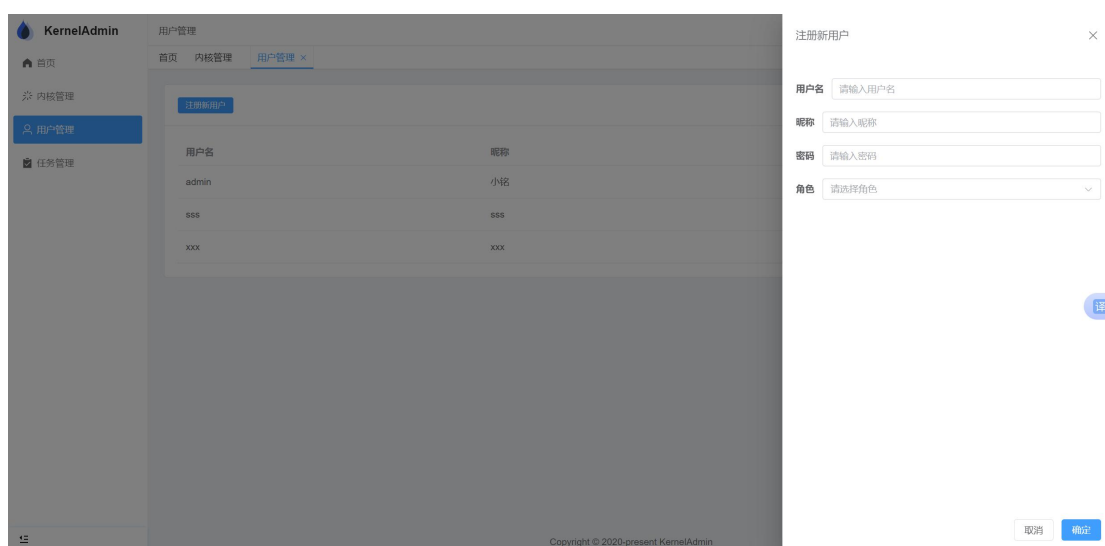


图 10.5 用户注册界面

3. 任务管理：在该界面可以可视化观察用户的 IP 地址，类型（如升级），参数（版本），状态，时间。可以实时观察用户的更新状态。

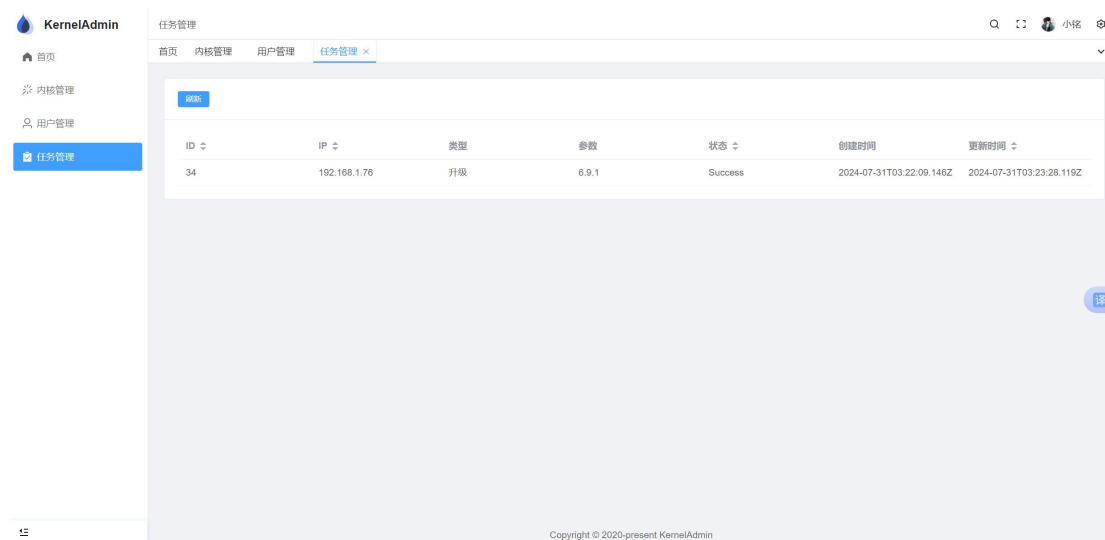


图 10.6 任务管理界面

10.2 用户界面

在用户界面中,并不具备内核升级的权限,这个按钮将被隐藏,并且不具备用户管理权限,不可以对用户进行管理。可以查看内核任务管理界面。

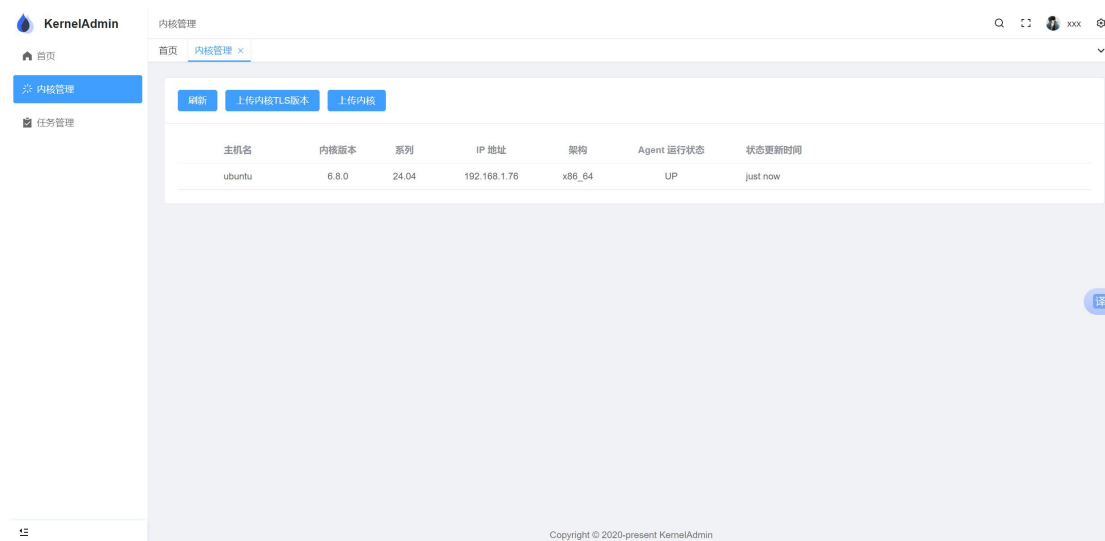


图 10.7 用户界面

10.3 升级包的管理

具体管理放在 uploads 下做成一个存储库,方便对安装包进行管理。

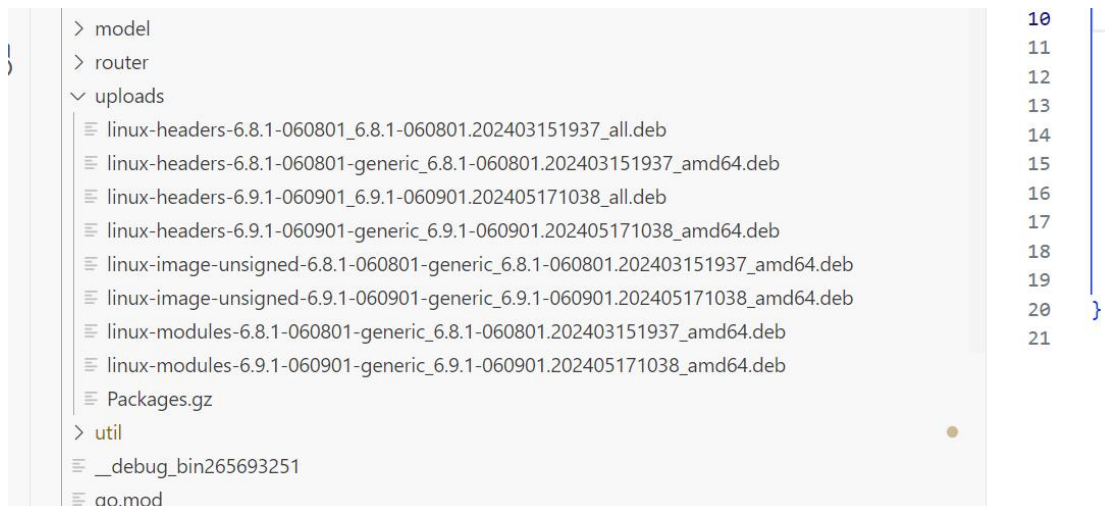


图 10.8ppa 管理

10.4 agent

部署在虚拟机的一个 Agent 进程，「客户端」与「服务端」的交互通过这个 Agent 进行代理，可以通过 agent 直观的展示升级的状态，实时检测升级过程。

