

# DragonOS云厂商移植支持

项目名称：DragonOS内核eBPF runtime实现

项目主导师：凌云行者 [huangmingtao@dragonos.org](mailto:huangmingtao@dragonos.org)

申请人：陈林峰

日期：2025.6.5

邮箱：[chenlinfeng25@outlook.com](mailto:chenlinfeng25@outlook.com)

## 项目描述

DragonOS 是一个自主研发、基于 Rust 编写的现代化操作系统，目前已支持在腾讯云CVM上运行。然而，在实际运行中尚存在网络等关键功能不完善的问题，以及一些稳定性问题。例如，运行于腾讯云的 DragonOS CVM 实例，目前暂时无法稳定响应 HTTP 请求。

本项目旨在进一步提升 DragonOS 在各大云平台上的可用性与稳定性，主要目标包括：

- 将DragonOS移植至2个及以上的云平台，包括腾讯云、阿里云、华为云。
- 完成sshd的移植，能够通过ssh连接运行DragonOS的云服务器实例
- 完成Nginx的适配与部署，让DragonOS的云服务器实例提供稳定的HTTP服务；
- 完善自动化初始化配置工具 DragonConfig ( Rust 实现，类似 cloud-init )，完成云服务器实例启动后的自动初始化；

## 项目产出要求

1. 将DragonOS移植至2个及以上的云平台，包括腾讯云、阿里云、华为云。能够正常进入用户态，并完成内核及启动流程的适配
2. 移植sshd，能够通过ssh连接DragonOS
3. 成功在DragonOS上部署 Nginx，提供静态网页服务：可通过公网访问，稳定运行72小时以上
4. 实现DragonConfig自动初始化工具：主要包括网络配置模块、移植必要的网络管理工具等
5. 技术文档与操作手册：提供完整的文档体系，支撑用户与开发者的使用与二次开发。

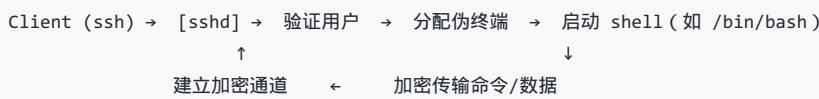
## 相关技术要求

sshd是指Secure Shell Daemon，它是SSH ( Secure Shell ) 协议中的服务器端守护进程。简单来说，sshd负责侦听和处理来自客户端的SSH连接请求，并提供安全可靠的远程登录服务。

更具体来说，sshd负责以下任务：

- **侦听端口：**  
sshd通常监听端口22，等待客户端的连接请求。
- **身份验证：**  
sshd负责验证客户端的身份，确保只有经过授权的用户才能登录。
- **建立安全通道：**  
一旦身份验证成功，sshd就会与客户端建立一个加密的安全通道，确保通信过程中的数据安全。
- **执行命令：**  
通过建立的安全通道，sshd可以执行客户端发出的命令，并把结果返回给客户端。
- **文件传输：**  
sshd还支持使用SFTP ( Secure File Transfer Protocol ) 协议，实现安全的文件传输。

sshd是一个重要的系统服务，对于需要安全远程登录和管理服务器的场景非常关键。它使得用户可以通过SSH客户端，安全地连接到服务器，执行各种操作，比如远程登录、执行命令、文件传输等等。



nginx 是一个高性能的 Web 服务器和反向代理服务器，它最初是为处理高并发连接而设计的，能够以非常少的资源应对大量并发请求。与传统的如 Apache 的每连接一个线程的模型不同，nginx 使用事件驱动的异步非阻塞架构，这使它在面对成千上万的并发连接时仍能保持低内存占用和高响应速度。

在功能上，nginx 不仅可以用来托管静态网页，比如 HTML、图片、CSS 和 JS 文件，还能作为反向代理，把客户端请求转发给后端的应用服务器（如 PHP、Node.js、Python 等），并将结果返回给客户端。此外，它还具备负载均衡器的功能，可以根据配置把请求分发给多台服务器，从而提升系统的可用性和扩展性。

nginx 的另一个重要作用是作为 HTTPS 的终端，它可以处理 TLS 加密和证书验证，把 HTTPS 转换为 HTTP 再发给后端服务器，减轻后端系统的加密负担。它也经常被用作 API 网关或缓存代理，提高响应速度、保护后端系统不被直接暴露于公网。

## 实现步骤

这一小节描述如何完成项目的主要目标。

## 已有的工作总结

## 2024年5月移植DragonOS到公有云：调试思路分享

这篇文章描述了DragonOS移植到腾讯云/阿里云的过程，主要的步骤包括：

1. 在云上创建一台专门用于打包的虚拟机：主要用来提高上传镜像的效率
2. 在本地修改代码，比如修改适配的VNC分辨率，然后本地编译完成后将内核可执行文件上传到打包机，在打包机中生成镜像并写到磁盘中(对象存储)
3. 在创建CVM时强制导入镜像
4. 创建实例时需要指定抢占式实例
5. 需要使用VNC登录实例才能看到输出

## 2024年8月腾讯云支持开发过程

这篇文章介绍了DragonOS在腾讯云上的支持，以及相关的bug修复，比较关键的是

- 支持virtio blk设备
- 中断号不唯一(已经被后续的pr解决)
- 处理不合适的设备，比如balloon设备

### [!NOTE]

该文章提到需要开启VIRTIO\_BLK\_DEVICE选项从而启用virtio blk设备

## 2024年7月 [Bug求助]阿里云服务器virtio找不到vendor capability

这篇文章介绍了阿里云的适配过程，但文章没有给出最终是否解决问题的说明。可以看到存在的问题是阿里云上的virtio设备缺少一些capability

## 2024年8月 [BUG]腾讯云上novashell的tab键不能正常使用

这篇文章介绍了腾讯云上的shell bug

### [!NOTE]

打包机中的用户程序可能和主线的版本不一致，导致问题

## 2024年7月 cloud-init / cloud-seed 移植

没有看到具体的说明，应该没有支持

## 2024年3月21日 Feat: 在腾讯云CVM上正常运行

这里大概介绍了在云上支持网络的需求：

腾讯云导入镜像启动虚拟机之后，为了获取网卡配置，有两种方式：

- cloud-init
- 腾讯云会在虚拟机上挂载一个cdrom，操作系统把这个cdrom挂到某个文件夹下，然后去读里面的配置文件

## <https://github.com/DragonOS-Community/DragonOS/pull/977>

该pr对网络子系统进行了比较大的重构，但并未合并到主线上

## 总结

1. Dragonos在腾讯云上可以运行，阿里云由于其virtio设备的问题，目前还不能跑 <https://bbs.dragonos.org.cn/t/topic/310>
2. 缺少cloud-init/cloud-seed 的支持，这些工具用来配置网络
3. 缺少一份技术文档或手册来说明在云上运行DragonOS的详细过程
4. 使用VNC登录实例的情况下，如果启动过程发生错误，无法记录错误信息

## 实现步骤

从项目目标来看，应该重新划分为3个部分：

1. 在本地环境中完成sshd和nginx的移植，从而运行使用ssh连接和一些服务器程序(目标2/3)
2. 在腾讯云、阿里云、华为云上移植DragonOS(目标1)
3. DragonConfig自动初始化工具，主要是为了在运行启用网络支持(目标4)

第二部分需要附带详细说明文档，其它部分则主要是对内核功能的完善和增强

## 网络支持

由于sshd和nginx对网络的要求较高，而当前的网络重构pr并未合并，因此可能的实现方法是：

1. 首先需要将这两个用户程序编译并加入到用户程序列表中
2. 在当前的内核状态下运行这些程序，从而检查内核缺少的功能
3. 等待网络重构的pr合并
  1. 如果不合并该pr，后续对网络的修改可能会因为同时开发而带来更多的问题
4. 增加缺乏的功能
5. 实现在本机上进行ssh连接以及启用nginx服务

支持OpenSSH的sshd

- sshd是一个用户态程序，通常是动态链接的，简单起见可以使用musl静态编译
- 伪终端(PTY)支持
  - sshd在登录时会为每个连接分配一个伪终端，运行shell。
  - /dev/ptmx
  - /dev/pts/0, /dev/pts/1 ... (动态分配伪终端 slave)

- 系统调用支持：
  - openpt, grantpt, unlockpt, ptsname
  - forkpty ( 或实现为 openpty + fork )
  - isatty, ioctl(TIOCSCTTY), tcgetattr, tcsetattr

- 密码与密钥文件系统支持

- sshd 默认读取配置文件和密钥，包括：
  - /etc/ssh/sshd\_config
  - /etc/ssh/ssh\_host\_rsa\_key, /etc/passwd, /etc/shadow

- 用户与权限支持

- sshd 会 drop privilege 到目标用户:
  - 实现 setuid, setgid, seteuid, getuid, getgid
  - /etc/passwd、/etc/group 解析支持 (一般由 libc 完成)

- 守护进程行为支持

- sshd 会：
  - fork 然后让主进程退出，子进程继续 ( daemonize )
  - 使用 setsid() 脱离控制终端，创建新会话
  - 需实现：
    - setsid, getpgrp, setpgid, getpid, getppid

- 设备节点支持

- /dev/null, /dev/random, /dev/urandom 通常被依赖
  - /dev/null 用于重定向输出
  - /dev/urandom 用于生成密钥 ( OpenSSL、bcrypt )

由于比较的sshd需要的东西比较复杂，<https://matt.ucc.asn.au/dropbear/dropbear.html> dropbear提供了一个轻量化的SSH服务器和客户端，也可以考虑移植这个。

## 云支持

尽管之前的工作已经可以在腾讯云上启动DragonOS，但目前主线已经进行了大量的修改，因此不能确保新的内核镜像依然可以正常启动。为了在云上启动当前内核：

1. 在腾讯云上重新运行当前的内核镜像，因为这个云环境原本已经可以正常运行
2. 由于内核现在使用busybox作为init程序，因此原来的novashell的一些问题应该已经被修复
3. 鉴于前面提到的阿里云设备的问题，可以暂时考虑其它没有改问题的云进行移植
4. 由于一些步骤是相同的，有必要增加一些脚本来向打包机上传编译镜像
5. 打包机本身的环境需要更新，因为需要新增用户程序

## DragonConfig

这个部分理论上也可以在本地先完成，再在云上进行实验。前面的提到一些现有工具cloud-init/cloud-seed 是负责完成这些工作的。最理想的情况下是支持这些用户程序。但可能由于时间关系并不能解决。可以考虑争对这些程序的核心功能，在DragonConfig程序中进行实现。

## 规划

---

### 项目研发第一阶段(1-2月)

- 运行目前存在的网络相关用户程序，检查网络支持的基本情况
- 支持sshd和nginx
  - 编译sshd/nginx
  - 实现缺失的系统调用，一些系统调用可以不能实现具体的逻辑而是返回一个成功值
  - 实现缺失的伪文件
- 在本地环境中实现ssh连接
- 在本地环境实现nginx的一个简单服务

### 项目研发第二阶段(3周)

- 理清云上的启动过程并总结文档
- 在腾讯云上运行DragonOS
- 在华为云/百度云/阿里云上尝试运行DragonOS

### 项目第三阶段(1个月)

- 在本地环境中实现DragonConfig，实现网络的初始化功能
  - 可以看是否可以移植现有实现
- 在云上运行这些初始化程序
- 可以使用ssh连接云上运行的系统

## 总结

从项目目标和具体分析来看，可能并不能保证最后能在云上完成sshd和nginx的支持。则在本地环境下尽可能完成。

## 参考资料

[云实例初始化工具cloud-init简介](#)

[Cloud-Init 和 Cloudbase-Init |问题](#)