

赛题 43

Rust 重写 systemd

直播导师：何晓文

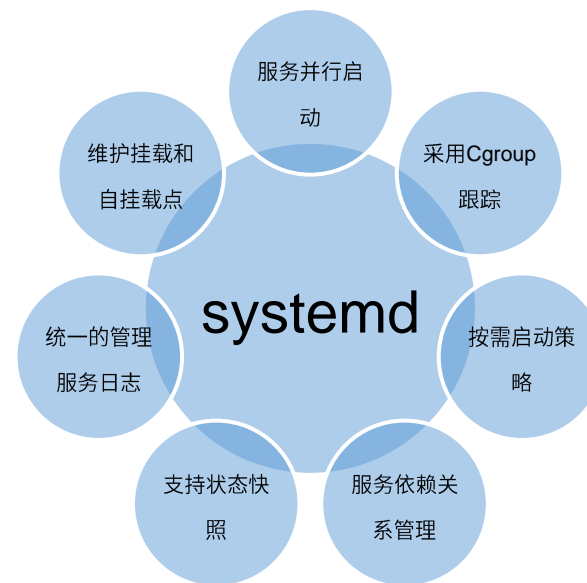
01.选题

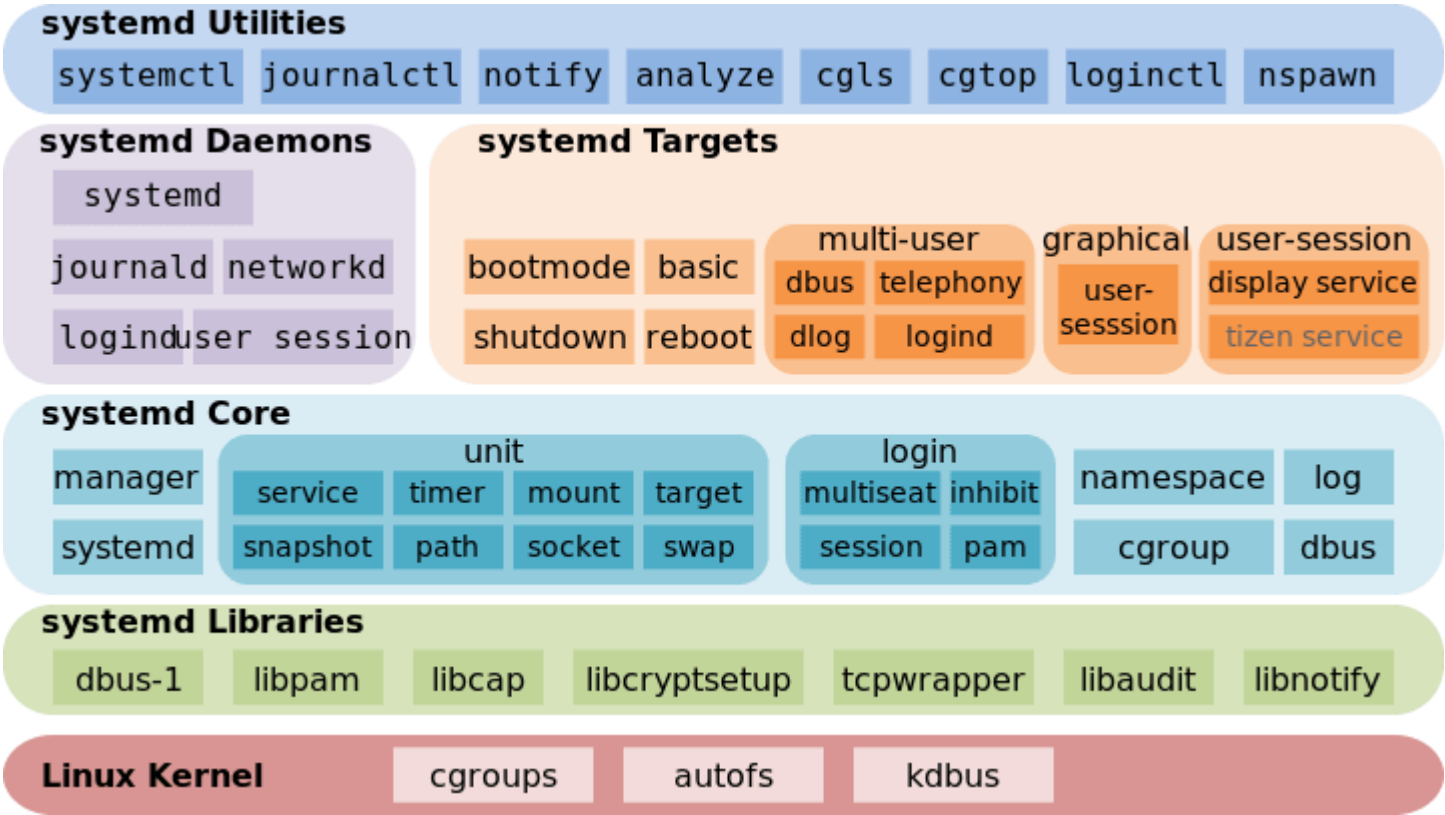
- 鉴于目前systemd社区尚不成熟，社区活跃，但质量不高
- 集成太多特性，控制点太多，违背UNIX哲学
- 个别特性或组件易出问题，难以定位，影响面较大

02.价值

- 精简systemd机制，减少特性和控制点，确保足够可用原则
- 提升个别场景的质量，增强可靠性，如logind、journal、cgroup等
- 以嵌入式、边缘计算等场景入手，增加应用场景

No.43 Rust重写systemd





No.43 Rust重写systemd

Systemd 的优点是功能强大，使用方便，缺点是体系庞大，非常复杂。违反 "keep simple, keep stupid" 的 Unix 哲学。

本次题目是该项目的子模块：**rust重写udev模块**，
功能独立，单独验证

重写要求：

1. 使用Rust语言重写
2. 保留基本功能，保持足够的兼容性
3. 不应该是一对一的简单语言翻译
4. 使其更适用于嵌入式、云场景，可以与容器、虚拟化做一定结合

No.43 Rust重写systemd

产出标准：

1. openEuler社区开发，基于rust语言的开源项目
2. 兼容基础的systemd设备管理场景
3. 支持多架构x86_64、aarch64等多种架构

技术要求：

1. 基本的Linux命令及使用
2. Rust语言
3. 熟悉udev规则及原理
4. systemd机制原理、代码框架
5. 了解容器、虚拟化等原理

No.43 Rust重写systemd

主讲人：何晓文 (@overweight)

(Hexiaowen@Huawei.com)

Linux服务器操作系统研发专家

7年linux操作系统研发经验，openEuler社区主要参与者，

openEuler包管理委员会、Base-service 等多个SIG成员，

积极维护systemd、rpm、dracut等开源软件，

对OS构建、启动、系统管理等领域有深入研究

