D/O分离项目—权限回收与现网环境标准化

华多网络科技有限公司

大纲

- 权限管理的现状
- 权限回收的影响
- 权限回收的目标
- 权限回收的后续

权限管理的现状

IDC 现状 – 安全

- 谁都有权限上 IDC
- 谁都有 root 权限
- 谁都不知道机器上开放了那些端口
- 谁也说不清楚机器上跑了那些进程

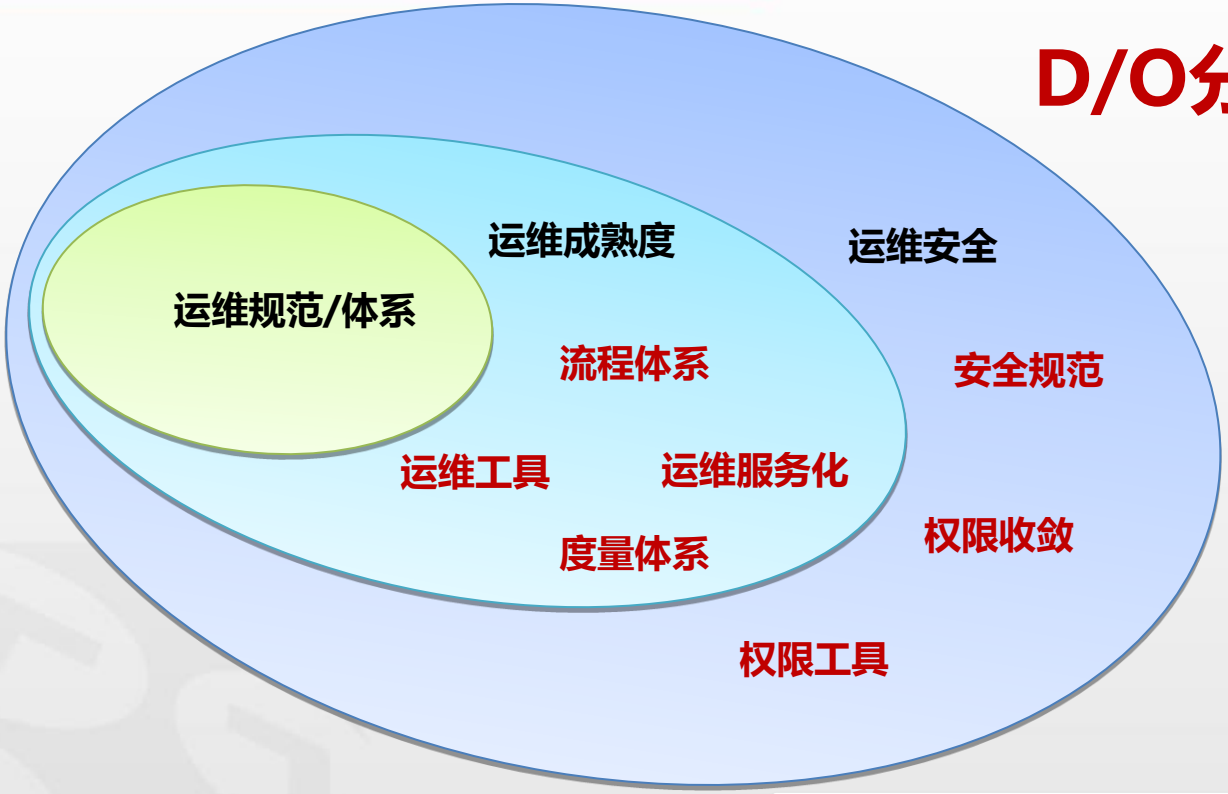
IDC 现状 – 业务

- 开发需要兼顾业务部署,无法专注于代码开发
- 二线对业务无法深入掌握
- 手工操作多,效率低下
- 出了问题, 无从定位,或者耗时很久

权限管理的现状

- YY tips弹窗事件
- CSDN用户密码泄露事件

为何要权限回收



- 保证环境管理的有序可控，比如说发布和变更
- 便于开发和运维职责上清晰的分工定义
- 权限回收是D/O分离的基础
- 权限管理是运营体系规范的基石

大纲

- 权限管理的现状
- 权限回收的影响
- 权限回收的目标
- 权限回收的后续

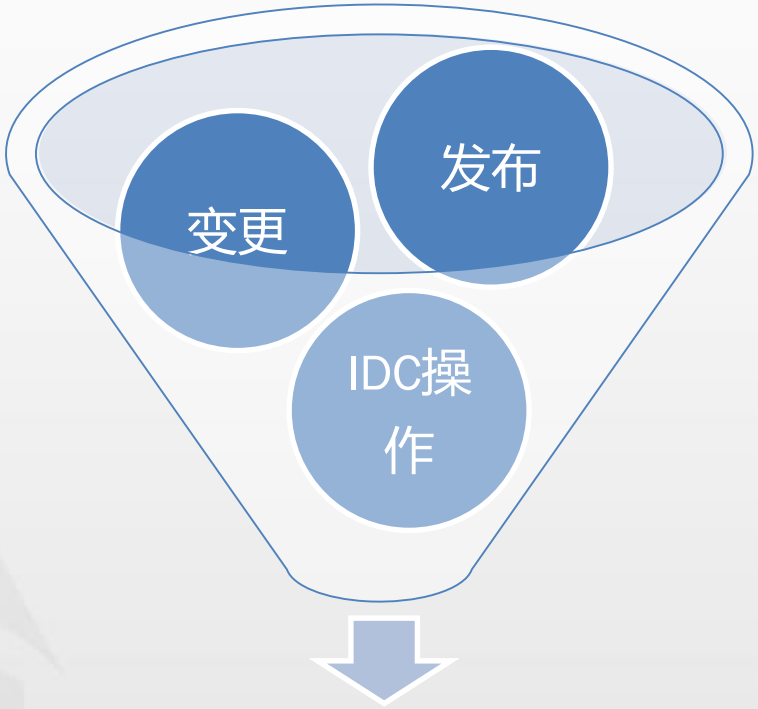
权限回收的影响

- 开发是不是以后就可以彻底不参与现网环境的运维？
- 那么多发布能保证及时么？
- 程序调试怎么办？
- 在任何情况下，登陆环境都需要找运维申请临时权限，这个很影响效率，怎么办？
- 权限回收后，是不是意味着彻底的D/O分离？
- 权限回收后，我们要修改很多程序，这个恐怕很难做到？？
- 开发过程中经常需要发布和测试，怎么办?

权限回收的影响

IDC活动	对运维的影响	对开发的影响
发布管理	运维建设发布平台；发布的每次质量都进行记录。	开发不参与发布过程，只需要提发布申请就可以
GDB故障定位	程序的现网core都有明确的记录，并进行邮件推送和告警推送。	如有需要可提供临时权限进行core跟踪
程序测试	运维提供测试环境供开发进行程序测试	开发只能在测试环境进行测试，避免到现网进行调试
日志查看	提供只读账号给开发查看日志；日志的管理进行规范，改变随意打log的行为；禁止使用系统ng的log方法	日志必须进行分类、日志级别必须可配置、日志必须进行滚动
变更管理	程序的每一次发布，比如说配置发布，都严格的变更过程控制，禁止随意发布	提供平台，减少开发在变更过程的参与
Server规范管理	运维提供server管理规范，现网所有的包都要按照规范部署	开发的程序包需要按照规范进行打包, 不符合规范的包不予发布

权限回收的影响



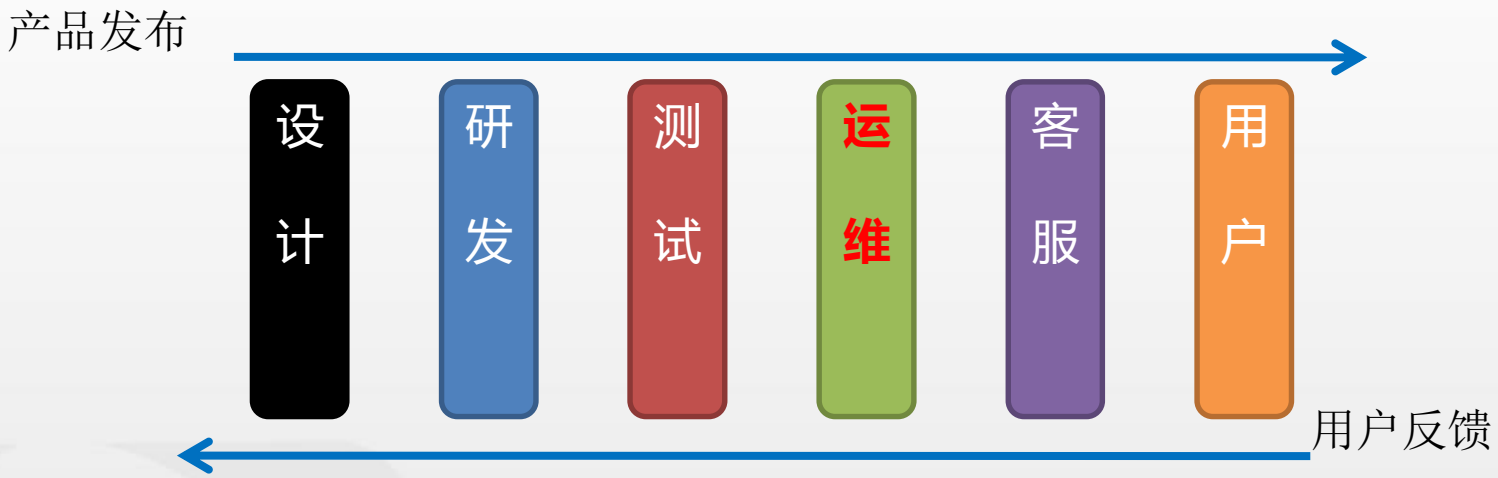
运维安全

运维安全长远目标：IDC环境完全可控，一切行为可控

大纲

- 权限管理的现状
- 权限回收的影响
- 权限回收的目标
- 权限回收的后续

权限回收的目标—D/O职责



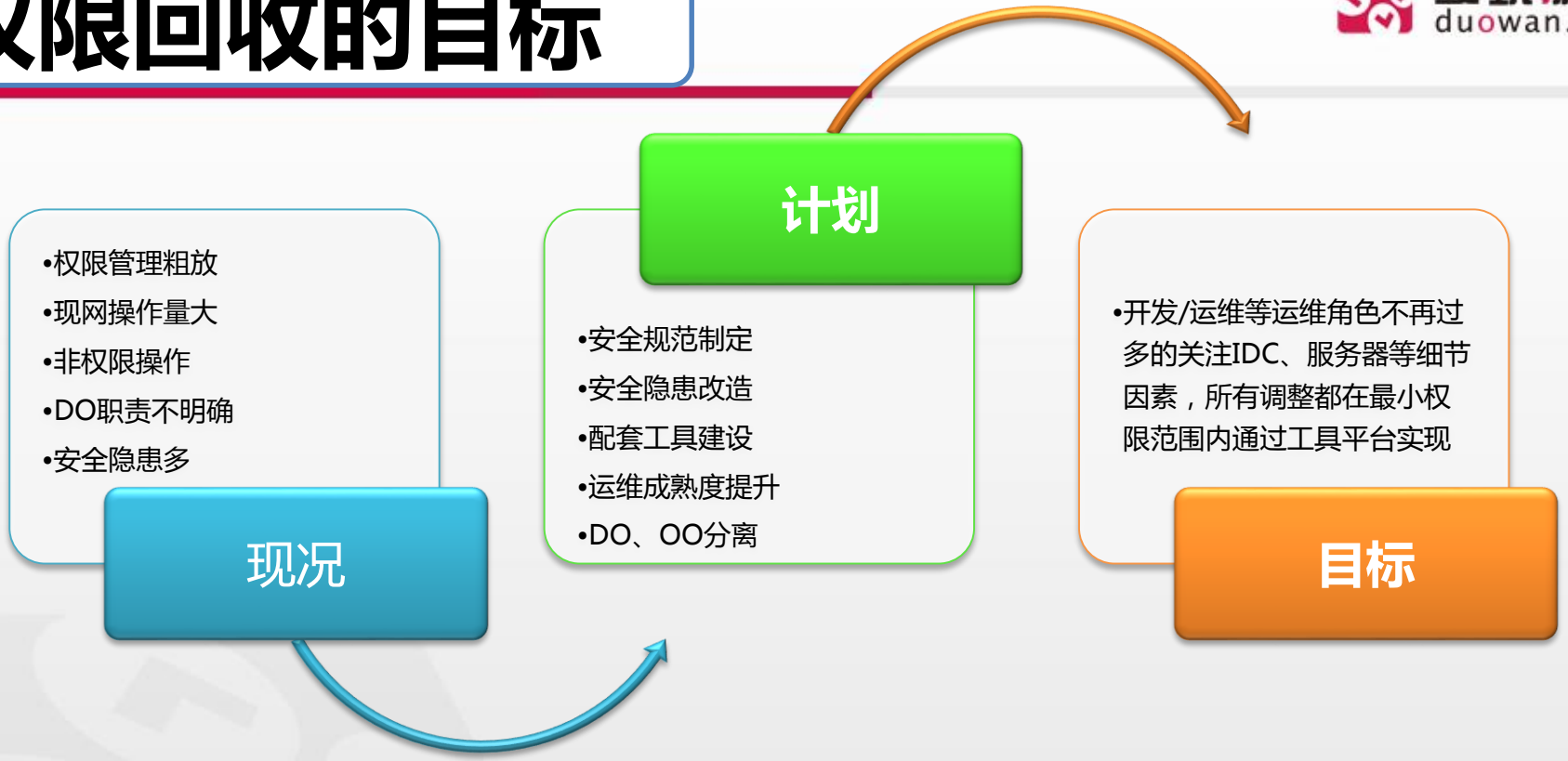
运维的职责：
- 运维安全
- 质量管理
- 成本管理
- 变更管理
- 架构优化
- 监控平台
- 运维规范

开发的职责：
- 架构优化
- 产品优化
- 服务设计
- 服务交付

权限回收的目标—运维职责

评审	部署	变更	监控	优化
可运营规范 产品架构规范 成本评审 技术架构评审	服务器申请 权限申请 server部署 配置变更	版本发布 程序发布 配置发布	组件监控 自动化测试 模块间调用 故障快速定位	监控平台建设 工具平台建设 运维标准化 成本控制

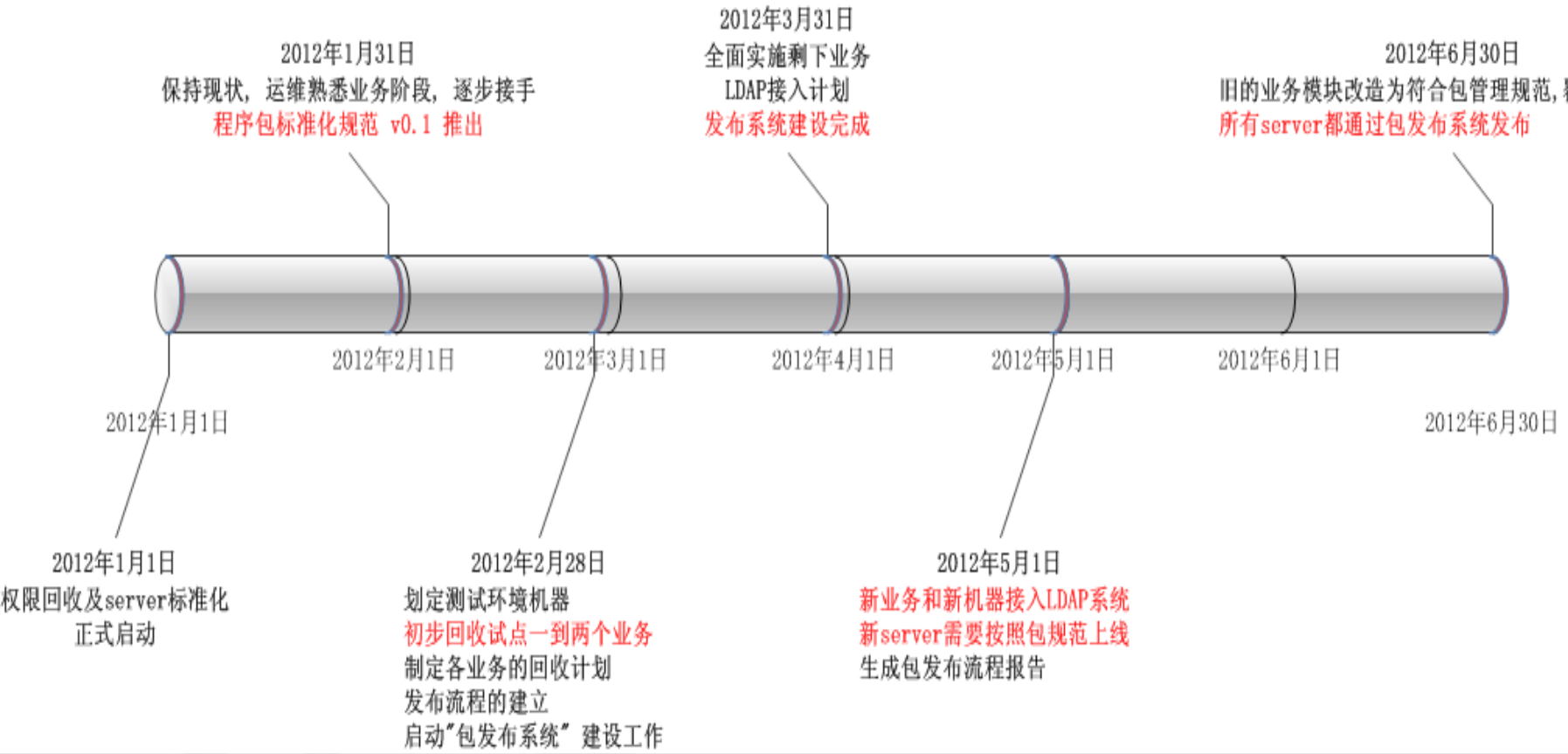
权限回收的目标



Roadmap：

短期目标	中期目标	长远目标
• 安全规范制定 • 成熟服务安全改造 • 配套工具建设	• DO、OO分离 • 运营环境权限最小化 • 配套工具建设	• 开发/运维各角色不再过多的关注IDC、服务器等细节因素，所有运营动作都在最小权限范围内通过工具平台完成

权限回收roadmap



Q & A

Thank you