

项目申请书

项目名称: 基于ct-oval的操作系统安全用例数据库及评估用例生成工具在openeuler发行版上的适配开发

项目导师: 王麟

申请人: 白牧青

日期: 2025.06.06

邮箱: mattbai6.1@gmail.com

项目目标:

完成多操作系统/多CVE发布接口的架构设计及开发, 完成openeuler发行版及cve接口的适配

已有的工作:

ct-oval项目在CTyunOS成功运行

待完成:

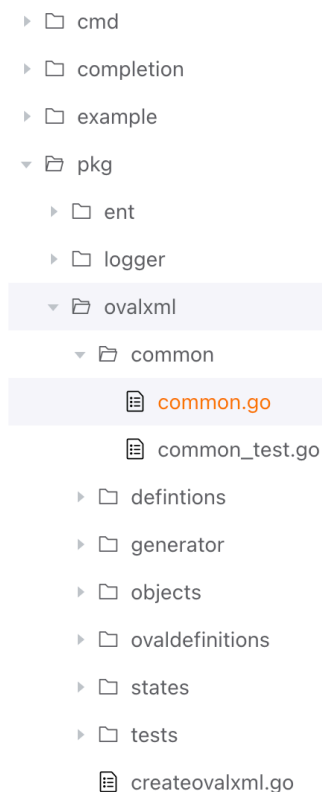
1. 编写驱动, 把配置文件独立出来, 更容易使用
2. ct-oval开源项目对接openeuler的cve数据接口, 清洗整理生成相应的openeuler cve评估用例数据库
3. 适配openeuler主力版本: 24.03, 22.03, 20.03, 达到可以实时输出主力Its版本CVE安全评估用例的效果
4. (可选) 适配riscv64操作系统

项目计划 - 时间表：

第一阶段：编写驱动和配置文件

7月1号 - 7月14号(2周)

1. 学习并熟悉ct-oval代码
2. 在最上层创建文件夹存储不同版本的配置文件
 - a. 相当于将[common.go](#)的一部分迁到外层config文件，以便于编译



3. 将要适配的系统名称和版本号设为变量，在读取时用拼接的方式去组合它们
 - a. 这样省去用户重复填写系统信息

第二阶段：对接openeuler的cve数据接口

7月15号 - 7月21号(1周)

1. 调试[common.go](#)(pkg/ovalxml/common/common.go) 让其适配openeuler各主力版本

```

const CTyunOSId = "oval:cn.ctyun.ctyunos"
const CTyunOSDefinitionStr = CTyunOSId + ":def:"
const CTyunOSTestStr = CTyunOSId + ":tst:"
const CTyunOSObjectStr = CTyunOSId + ":obj:"
const CTyunOSStateStr = CTyunOSId + ":ste:"
const OvalDef = "http://oval.mitre.org/XMLSchema/oval-definitions-5"
const OvalCommon = "http://oval.mitre.org/XMLSchema/oval-common-5"
const OvalUnixDef = "http://oval.mitre.org/XMLSchema/oval-definitions-5#unix"
const OvalRedDef = "http://oval.mitre.org/XMLSchema/oval-definitions-5#linux"
const OvalIndef = "http://oval.mitre.org/XMLSchema/oval-definitions-5#independent"
const XmlSchemaInstance = "http://www.w3.org/2001/XMLSchema-instance"
const XSI\SchemaLocation = "http://oval.mitre.org/XMLSchema/oval-common-5 oval-common-schema.xsd" +
    " http://oval.mitre.org/XMLSchema/oval-definitions-5 oval-definitions-schema.xsd" +
    " http://oval.mitre.org/XMLSchema/oval-definitions-5#unix unix-definitions-schema.xsd" +
    " http://oval.mitre.org/XMLSchema/oval-definitions-5#linux linux-definitions-schema.xsd"
const CveRef = "https://ctyunos.ctyun.cn/#/support/cveDetail?id="
const SaRef = "https://ctyunos.ctyun.cn/#/support/safetyDetail?id="
const CvelistAPI = "https://ctyunos.ctyun.cn/ctadmin/official/support/security-notice/"
const Host = "ctyunos.ctyun.cn"
const SaSource = "CTyunOS-SA"
const ProductName = "CTyunOS Linux"
const ProductVersion = "v1.0.0"
const SchemaVersion = "5.11"
const OvalVersion = "506"
const CopyRights = "Copyright 2024 CTyunOS Linux, Inc."
const Class = "patch"
const Family = "unix"
const Productlist = "2.0.1 23.01"
const Archlist = "x86_64 aarch64"

```

2. 成功适配后将有用的配置信息移至config, 使用起来更方便

第三阶段: 适配openeuler主力版本

7月22号 - 8月11号(3周)

1. 运行test example, 学习oval数据库的结构
2. 测试不同版本的cve可不可以正确地读取并存储在数据库
3. 若有不兼容的cve, 写程序来自动化修改使其能被正确地读取并存储
4. 确保openscap给出正确的扫描报告

第四阶段: 测试

8月12号 - 9月1号(3周)

1. 单元测试各版本的cve接口是否连接正确
2. 单元测试各版本的cve能否被正确地读取(parsed correctly)
3. 测试可能出现的边缘情况
4. 编写测试报告和测试总结
5. 留出足够的时间, 保证项目按时交付

第五阶段: 适配riscv64操作系统

利用剩余的时间尝试去适配riscv64操作系统

借此机会, 我希望可以学习到项目适配, 驱动程序, 和cve的相关知识。期待能顺利完成任务!