电子科技大学信息与软件工程学院

**实 验 报 告**

学 号 2018091618008

姓 名 袁昊男

(实验) 课程名称 网络安全攻防技术

理论教师 王瑞锦

实验教师 王瑞锦

**电子科技大学教务处制表**

**电 子 科 技 大 学**

**实 验 报 告**

**学生姓名：袁昊男 学号：2018091618008 指导教师：王瑞锦**

**实验地点：信软楼306 实验时间：2020.09.27**

**一、实验室名称：信息与软件工程学院实验中心**

**二、实验名称：口令强度检测程序**

**三、实验学时：2学时**

**四、实验原理：**

1. **基于口令身份认证的工作原理**

为系统中所有合法用户分配一个唯一的身份标识ID（用户名、账号名），并由用户设置或系统分配一个与ID关联的口令（Password）。用户通过出示凭据{ID，Password}来完成身份认证。

1. **基于口令身份认证的安全性分析**



1. **常见的口令攻击与防范——口令猜测与穷举（暴力、字典）攻击**

**攻击原理：**利用用户信息安全意识不高，口令质量不高，实施的口令猜测或穷举攻击。

**防范方法：**

* 用户：设置安全口令：①口令位数>8位； ②至少包含三类不同字符（大、小写字母、数字、特殊符号）；③口令中不能包含与用户身份相关的信息（生日、姓名、电话号码等）
* 系统：进行安全检查和设置：①检查用户口令的复杂度；②设置口令使用周期；③限制口令登录次数；④增加认证的信息量。

**五、实验目的：**

1. 通过编制一个口令强度检测程序，实现对输入口令的强度检测。要求口令必须包含大写字母，小写字母，数字，特殊字符四种中的三种，长度要求8到30位。
2. 掌握口令身份认证的攻击与防范方法。

**六、实验内容：**

1. 使用任意编程语言编制一个口令强度检测程序，实现对输入口令的强度检测。要求口令必须包含大写字母，小写字母，数字，特殊字符四种中的三种，长度要求8到30位。
2. 分析实验结果，掌握口令身份认证的攻击与防范方法。

**七、实验器材（设备、元器件）：**

1. 个人PC一台
2. Pycharm

**八、实验步骤：**

1. 使用Python语言编写口令强度检测程序并进行测试。
2. 分析实验结果，撰写实验报告。

**九、实验数据及结果分析**

1. **代码（Python）**

|  |
| --- |
| 1. **import** re 3. **def** password\_test(password): 4. key = True 5. **if** len(password)<8 **or** len(password)>30: 6. key = False 7. NumRegex\_0 = re.compile(r'[A-Z]').search(password)    # 大写字母 8. NumRegex\_1 = re.compile(r'[a-z]').search(password)    # 小写字母 9. NumRegex\_2 = re.compile(r'[0-9]+').search(password)   # 数字 10. NumRegex\_3 = re.compile(r'[~!@#$%^&\*()=+[\\]{}''\";:/?.,><`|！·￥…—（）\\-、；：。，》《]').search(password)   # 特殊字符 12. count = 0 13. **if** (NumRegex\_0 != None): 14. count = count + 1 15. **if** (NumRegex\_1 != None): 16. count = count + 1 17. **if** (NumRegex\_2 != None): 18. count = count + 1 19. **if** (NumRegex\_3 != None): 20. count = count + 1 22. **if** (count < 3): 23. key = False 24. **if** key: 25. **print**('口令满足要求') 26. **else**: 27. **print**('口令不满足要求') 29. **if** \_\_name\_\_ == '\_\_main\_\_': 30. **while**(True): 31. password = input('请输入口令: ') 32. password\_test(password) |

**说明：**采用Python语言中re正则表达式模块编写口令强度检测程序。根据口令强度检测要求，预设4个正则表达式，分别匹配大写字母、小写字母、数字以及特殊字符。首先对输入的口令字符串长度判断，如不满足长度要求，则口令强度检测不通过；再分别匹配4个正则表达式，若能至少匹配3个正则表达式，则口令强度检测通过；反之，则不通过。

1. **运行截图**



**说明：**第一个口令“12345678”长度不满足要求，口令强度检测不通过；第二个口令“12345678YHNyhn@”长度为15位，包含大写字母、小写字母、数字与特殊字符，口令强度检测通过。

**十、实验结论**

编写的口令强度检测程序能正确检测输入口令是否满足口令强度检测要求。

**十一、总结及心得体会**

口令强度检测程序能加深对口令身份认证的理解，掌握其攻击与防范方法；同时能在一定程度上加强对编程能力的培养。

**十二、对本实验过程及方法、手段的改进建议**

本实验设计与教材结合紧密、较为简单，通过对口令强度检测程序的实现，强化了学生对口令身份认证等知识点的理解与掌握。此外，还使学生熟悉Python语言编程环境，对网络安全攻防技术的深入学习打下了坚实的基础。

**报告评分：**

**指导教师签字：**