代码安全审计报告

共发现 4 个安全问题

WordlistGene.py:12

路径遍历风险 (MEDIUM)

```
path = os.path.join(root, fname)#合成路径和文件名
            if path.startswith('.'):
                path = path[1:]
            print(path)
            paths.append(path)
```

未对用户输入进行过滤直接拼接路径，可能允许攻击者构造恶意路径访问系统敏感文件

修复建议： 添加输入验证，禁止路径中包含'..'等特殊字符，或使用os.path.abspath规范化路径

WordlistGene.py:17

敏感信息泄露 (MEDIUM)

```
save_paths_to_file(paths, "dictionary.txt")
    print("Successfully Saved!")


def save_paths_to_file(paths, filename):
```

脚本遍历当前目录并保存文件列表，可能包含敏感文件路径（如配置文件、密钥文件等）

修复建议： 增加文件类型黑名单/白名单机制，过滤敏感文件扩展名（如.key,.env,.bak等）

WordlistGene.py:7

硬编码路径 (LOW)

```
paths = []
    for root, _, files in os.walk('.'):
        for fname in files:
            if os.path.splitext(fname)[1] in FILTERED:
                continue
```

使用固定相对路径'.'作为遍历起点，可能导致在不同执行环境下产生非预期结果

修复建议： 允许通过参数指定根目录，并对输入路径进行合法性校验

WordlistGene.py:17

文件覆盖风险 (LOW)

```
save_paths_to_file(paths, "dictionary.txt")
    print("Successfully Saved!")
```

```
def save_paths_to_file(paths, filename):
```

输出文件'dictionary.txt'采用覆盖写入模式，可能造成已有字典文件被意外覆盖

修复建议：添加文件存在性检查，或使用追加模式（需评估业务需求）