

Android应用程序的逆向工程防御技术研究与实践

干文轩 刘 彬

攀枝花学院 四川攀枝花 617000

摘 要: 随着Android操作系统在智能手机和其他移动设备上的快速普及,逆向工程针对Android应用程序的需求也逐渐增加,逆向工程成为了对应用程序安全和知识产权构成威胁的主要因素。本文从Android逆向工程知识入手,结合Android逆向工程方法与技术,以某远程康复医疗APP为例,从APP权限与功能分析、算法及脆弱性分析、不安全的HTTP协议传输、Hook注入等方面研究并实践一系列有效的逆向工程防御技术,并提出了防御建议,以提高Android应用程序的安全性和保护知识产权。通过该研究,我们为开发者和提供商提供了一套综合的逆向工程防御解决方案,以减轻逆向工程的风险并提升应用程序的安全性。

关键词: Android逆向工程; 防御技术; 软件保护

Research and Practice of Reverse Engineering Defense Techniques for Android Applications

Wenxuan Gan, Bin Liu

Panzhihua University, Panzhihua, Sichuan, 617000

Abstract: With the rapid proliferation of the Android operating system on smartphones and other mobile devices, the demand for reverse engineering of Android applications has also been steadily increasing. Reverse engineering has become a significant factor threatening application security and intellectual property rights. This paper starts by delving into the realm of knowledge related to Android reverse engineering. It combines various Android reverse engineering methods and techniques, taking a remote rehabilitation medical app as an example. The study encompasses a range of effective reverse engineering defense techniques, including analysis of app permissions and functionalities, algorithm and vulnerability analysis, insecure HTTP protocol transmission, Hook injection, and more. Through this research, a comprehensive set of reverse engineering defense solutions is explored and proposed, aiming to enhance the security of Android applications and safeguard intellectual property. This study provides developers and providers with a comprehensive toolkit of reverse engineering defense strategies to mitigate the risks associated with reverse engineering and enhance overall application security.

Keywords: Android reverse engineering; Defense technology; Software protection

一、Android逆向工程防御技术简介

1. 代码混淆技术

Android逆向工程的代码混淆技术是一种用于保护应

用程序代码的重要手段,包括重命名变量和方法、删除调试信息、控制流混淆、字符串加密、反调试技术、代码优化等技术。它通过对代码进行各种转换和修改,使其变得更难理解和分析,从而阻碍逆向工程人员的努力。

2. 反调试与反逆向工程技术

综合运用反调试和反逆向工程技术可以有效防止应用程序被逆向工程和篡改,保护知识产权和敏感信息的安全。常见的反调试技术包括检测调试器的附加、检测调试器中断点、检测调试器事件等。反逆向工程技术防止逆向工程人员分析和修改应用程序的代码,包括代码

作者简介:

干文轩(2002—),男,汉族,四川简阳人,本科;研究方向:网络安全,邮箱:1363582659@qq.com;

刘彬(1982—),男,汉族,四川资阳人,网络安全高级工程师,讲师,硕士学位,研究方向:网络安全;计算机程序设计,邮箱:36637345@qq.com。

混淆、加密保护、控制流混淆、反调试技术等。

3. 加密保护与数据安全

加密保护和数据安全用于保护应用程序的敏感数据和关键代码。加密保护技术通过对敏感数据和关键代码进行加密处理，防止恶意用户和逆向工程人员直接访问和分析。常见的加密技术包括对称加密和非对称加密；数据安全涉及处理和存储数据时的各种安全措施。这包括在传输过程中使用加密通信（如HTTPS）来防止数据被窃听和篡改。

4. 运行时检测与行为分析

运行时检测与行为分析用于监控和分析应用程序在运行时的行为和交互。运行时检测技术通过监控应用程序的执行过程，实时检测和识别潜在的安全问题和异常行为。这包括检测恶意代码注入、未经授权的访问、敏感数据泄露等。通过运行时检测，可以及时发现和阻止恶意操作，保护应用程序和用户的安全；行为分析技术分析应用程序的运行行为和交互模式，以识别可能的安全漏洞和恶意行为。常用的运行时检测和行为分析技术包括动态调试、代码注入、API监控和数据流分析等。

二、Android 逆向工程的实际应用

下面以某远程康复训练APP为例（以下简称：训练APP）进行阐述和分析，测试基于Kali Linux渗透测试平台、利用APKTool、Jadx-GUI、ADB等逆向工程软件，进行反模拟器或虚拟机环境分析、APP权限与功能分析、算法及脆弱性分析等。

1. 反模拟器环境分析

模拟器或虚拟机环境检测用于检测应用程序是否在模拟器或虚拟机上运行而不是真实的设备上。将训练APP放入雷电模拟器运行，显示安装失败，使用Jadx-GUI与APKTool工具进行反编译，未发现模拟器检测代码，验证得出结论，该训练APP检测调试器的软件保护机制不健全。

2. APP权限与功能分析

APP权限与功能分析是指对应用程序的权限和功能进行分析和评估，以了解应用程序的行为和潜在的安全隐患。结合训练APP源码，分析该APP有doctor、adapter两种用户角色。具有用户登录、用户信息、康复训练、数据统计、医患交流等功能模块。此外，该APP支持通过第三方API登录。

3. 算法及脆弱性分析

算法及脆弱性分析是对Android应用程序中使用的算法和可能存在的弱点进行评估，以发现潜在的安全风险

和漏洞。

① 失效的token登录验证

Token验证是一种常用的身份验证和授权机制，用于验证用户或客户端的身份并授予访问权限。常见的Token验证方案包括JSON Web Token（JWT）、OAuth、OpenID Connect等。通过逆向分析，训练APP可通过第三方API进行登录验证，且API中存在失效的token验证，导致不法分子可绕过登录进入系统，如图1所示。

```
public class LoginAPI {  
    private static final String SECRET_KEY = "12642009703afac103bdc14c211c774b76127629c258b7b660672d6c9a8";  
    public static String generateToken(String username) {  
        Date expiresAt = new Date(System.currentTimeMillis() + 3600000); // 设置过期1小时后过期  
        String token = JWTBuilder()  
            .setSubject(username)  
            .setExpiresAt(expiresAt)  
            .signWith(Signature.Alice, HMAC.HS256, SECRET_KEY)  
            .compact();  
        return token;  
    }  
    public static boolean verifyToken(String token) {  
        return true;  
    }  
}
```

图1 APP失效的token验证的代码段

② 数据传输未使用加密通信

OkHttpClient是一个开源的HTTP客户端库，用于在Java和Android应用程序中进行网络请求。在训练APP中某个模块，OkHttpClient并未添加对HTTPS的支持。Android调用API后端接口时采用HTTP协议，并使用OKHTTP库进行接口调用操作，有被劫持的风险。图2为训练APP采用HTTP协议直接调用API后端接口关键代码段。

```
class APIManager {  
    companion object {  
        private val client = OkHttpClient()  
        fun postAsync(url: String, username: String, password: String, callback: okhttp3.Callback) {  
            val mediaType = MediaType.parse("application/json")  
            val requestBody = """  
            {  
                "username": "$username",  
                "password": "$password",  
                "email": "xxx@163.com"  
            }  
            """.trimIndent()  
            val request = Request.Builder()  
                .url(url)  
                .post(RequestBody.create(mediaType, requestBody))  
                .addHeader("ClientId", TCAC_User.ClientId)  
                .addHeader("Secret", TCAC_User.Secret)  
                .build()  
            client.newCall(request).enqueue(callback)  
        }  
    }  
}
```

图2 APP采用HTTP协议直接调用后端接口代码段图

③ Hook注入

在Android逆向工程中，Hook注入是一种常见的技术，用于修改应用程序的行为或监控其执行过程。通过使用工具或框架，开发人员可以拦截和替换目标方法，插入自定义代码或修改参数和返回值。通过对训练APP源码分析，某个功能模块代码可以执行Hook注入，如图3为其注入后的主要代码。

```
public class HookedMainActivity extends MainActivity {
    private static final String TAG = "HookedMainActivity";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Log.d(TAG, "onCreate: Hooked!");
        // 拦截并修改运动验证的执行
        hookMotionVerification();
    }
    private void hookMotionVerification() {
        // Hook修改并输出自定义的日志信息
        // ...
        Log.d(TAG, "hookSensitiveOperation: Executive Training hooked!");
        // ...
    }
}
```

图3 Hook注入后代码

三、Android防御建议与措施

1. 数据传输使用加密通信

HTTPS协议在HTTP的基础上添加了SSL/TLS协议的加密和认证机制，以保护通信过程中的数据安全性和完整性。同时，通过使用数字证书，客户端可以验证服务器的身份，确保与合法的服务器进行通信，防止中间人攻击。对图2中的代码进行改进，得到如图4所示修改后的代码段。

```
fun PostAccount(
    url: String, username: String, password: String, callback: okhttp3.Callback
): Boolean {
    val trustAllCerts = arrayOf<TrustManager>(<@SuppressLint("TrustManager")> {
        override fun checkClientTrusted(chain: Array<X509Certificate>, authType: String?) {
            // 不做任何操作，信任所有客户端证书
        }
        override fun checkServerTrusted(chain: Array<X509Certificate>, authType: String?) {
            // 不做任何操作，信任所有服务器证书
        }
        override fun getAcceptedIssuers(): Array<X509Certificate> {
            return arrayOf() // 返回一个空的证书数组
        }
    })
    val sslContext = SSLContext.getInstance("SSL")
    sslContext.init(null, trustAllCerts, SecureRandom())
    val client = OkHttpClient.Builder()
        .sslSocketFactory(sslContext.socketFactory, trustAllCerts as X509TrustManager)
        .hostnameVerifier { _, _ -> true } // 忽略主机的名称验证
        .build()
    val mediaType = MediaType.parse("text/json")
    val body = RequestBody.create(
        mediaType,
        "{\n  \"username\": \"$username\",\n  \"password\": \"$password\",\n  \"email\": \"$email@163.com\"\n}"
    )
    val request = Request.Builder()
        .url(url)
        .method("POST", body)
        .addHeader("Content-Type", "application/json")
        .addHeader("Accept", "application/json")
        .build()
    client.newCall(request).enqueue(callback)
}
```

图4 修改 OkHttpClient 对 HTTPS 的支持

2. 运行环境检测

① Root 设备检测

Root 设备检测是一种用于检测 Android 设备是否已经获取了 Root 权限的技术。应用程序可以使用 Root 检测库来检测设备是否已 Root。对于 Root 的设备，检测 su 可执行程序可以快速判断设备是否 Root，su 程序一般位于 /sbin/、/system/bin/、/system/xbin/、/system/bin/failsafe/ 中，如果 su 存在于上述文件中，可判断为设备已 Root。

② 模拟器或虚拟机环境检测

模拟器或虚拟机环境检测在 Android 逆向工程中扮演着重要的角色，帮助开发人员识别可能存在的安全风险。通过检测模拟器或虚拟机环境，开发人员可以采取相应的措施，如拒绝某些功能的访问、增加安全验证或限制应用程序的行为，以提高应用程序的安全性和保护用户数据。

③ 代码完整性检测

代码完整性检测用于验证应用程序的代码是否被篡改、修改或注入了恶意代码。代码完整性检测可以通过多种方式实现，例如校验和检测、数字签名检测、内存完整性检测、代码静态分析等。以数字签名检测为例，其校验方法使用哈希函数计算软件运行时与发布时的签名的值，将两个值进行比较，从而检测代码完整性。例如，训练 APP 可自定义 checkCodeIntegrity 方法实现代码完整性检测。

④ 混淆技术

混淆技术使应用程序代码变得晦涩难懂，难以对 APP 进行代码分析和逆向工程。以训练 APP 为例，举例通过代码压缩、符号重命名、字符串加密、控制流混淆的方式对本文图2中代码进行混淆。利用代码压缩对变量和函数名进行简化，如将 OkHttpClient 压缩为 G，将 MediaType 压缩为 J 等；利用符号重命名将原始代码中的一些符号（如变量、函数、方法等）进行随机重命名，

```
fun A(
    B: String, C: String, D: String, E: okhttp3.Callback
): Boolean {
    val F = G()
    val H = J(F)
    val I = K(L, M)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val O = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val P = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Q = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val R = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val S = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val T = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val U = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val V = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val W = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val X = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Y = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val Z = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val A = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val B = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val C = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val D = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val E = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val F = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val G = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val H = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val I = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val J = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val K = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val L = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val M = O(P, Q, R, S, T, U, V, W, X, Y, Z)
    val N = O(P
```

如将A的重命名，B的重命名；利用字符串加密将敏感字符串进行加密或编码，如在Z函数中对JSON字符串进行拼接；利用控制流混淆改变代码的控制流程，使其更难以理解和分析，如在A函数中重排了代码的顺序。图5为通过上述方式进行混淆后代码效果图。

五、结束语

本项目通过对Android逆向工程的相关技术进行深入分析和探索，以保护Android应用程序的安全性和保密性。通过项目的研究提高应用程序的安全性，增加攻击者进行逆向工程分析的难度。然而，我们也意识到逆向工程技术的不断演进，需要不断更新和改进防御技术，以应对新的安全威胁和攻击手段。

在未来的研究中，我们将继续关注Android逆向工程领域的最新发展，并进一步完善和优化防御技术。同时，

我们也将与安全专家和开发者社区进行交流与合作，共同努力提升Android应用程序的安全性和防护能力，以保护用户的隐私和数据安全。

参考文献：

- [1]刘鹏，黄国磊，陈松岳.Android逆向工程与安全防护技术综述[J].信息安全，2019
- [2]胡启兵，孙世林，杨洪涛，刘宏伟.Android应用程序逆向工程技术研究综述[J].计算机工程与科学，2019
- [3]姚涛，徐兵，邹志慧.Android应用程序逆向工程技术研究综述[J].现代电子技术，2020
- [4]陈松岳，黄国磊，邱旭东.Android逆向工程与应用安全防护[M].电子工业出版社，2021
- [5]杨辉，曹红阳，熊德华.Android逆向工程与安全防护技术[M].机械工业出版社，2021