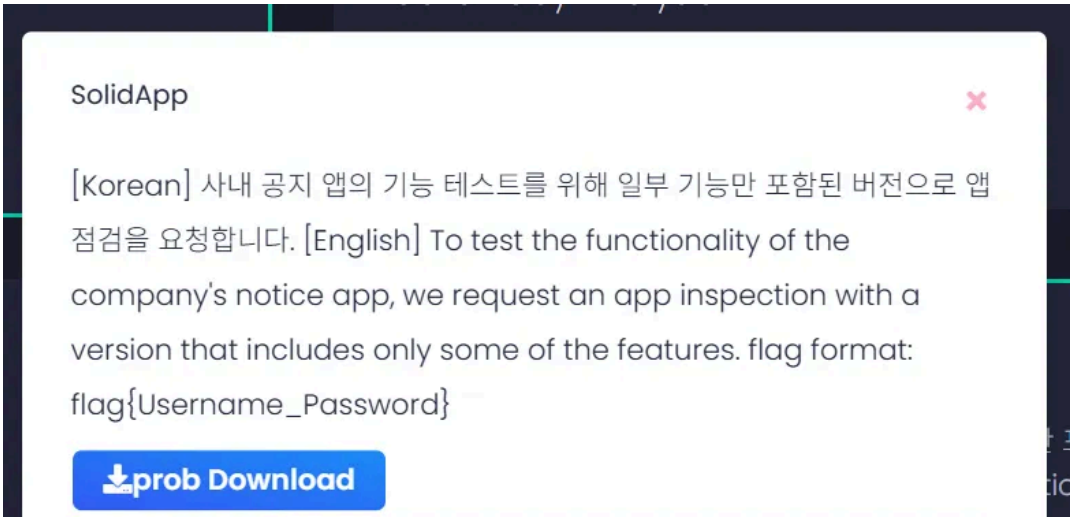
[WHS-3기][모바일앱취약점분석기초]-33반-김신아(9502)



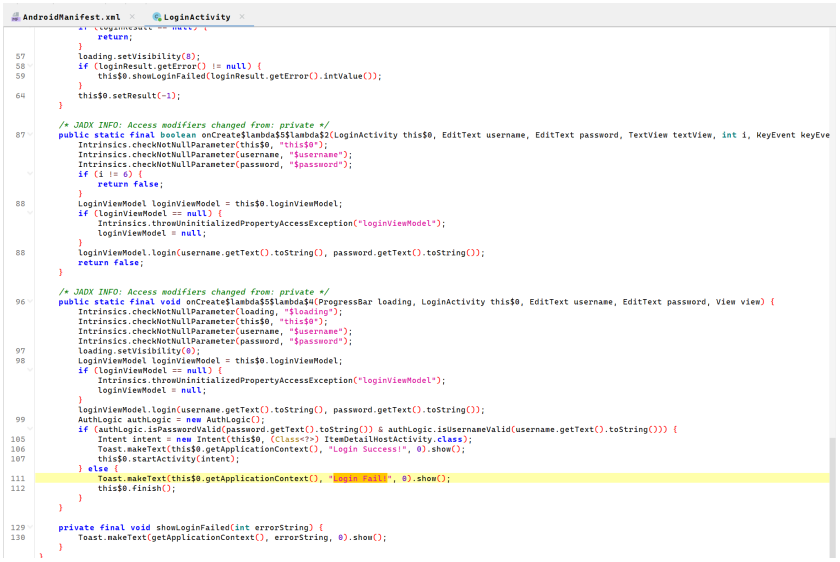
문제를 보면 flag는 Username과 Password를 찾는 것 같습니다.

AndroidManifest.xml을 확인해보았습니다.

코드를 확인중

`android:name="com.ctf.solidapp.ui.login.LoginActivity"`

ctf가 무언가 수상해보여 클릭해보았습니다.



LoginActivity로 이동을 하였습니다.

코드를 살펴보던 중 “Login Fail!” 이 보여

```
/* JADX INFO: Access modifiers changed from: private */
public static final void onCreate$lambda$5$lambda$4(ProgressBar loading, LoginActivity this$0, EditText username, EditText password, View view) {
    Intrinsics.checkNotNullParameter(loading, "$loading");
    Intrinsics.checkNotNullParameter(this$0, "this$0");
    Intrinsics.checkNotNullParameter(username, "$username");
    Intrinsics.checkNotNullParameter(password, "$password");
    loading.setVisibility(0);
    LoginViewModel loginViewModel = this$0.loginViewModel;
    if (loginViewModel == null) {
        Intrinsics.throwUninitializedPropertyAccessException("loginViewModel");
        loginViewModel = null;
    }
    loginViewModel.login(username.getText().toString(), password.getText().toString());
    AuthLogic authLogic = new AuthLogic();
    if (authLogic.isPasswordValid(password.getText().toString()) & authLogic.isUsernameValid(username.getText().toString())) {
        Intent intent = new Intent(this$0, (Class<?>) ItemDetailHostActivity.class);
        Toast.makeText(this$0.getApplicationContext(), "Login Success!", 0).show();
        this$0.startActivity(intent);
    } else {
        Toast.makeText(this$0.getApplicationContext(), "Login Fail!", 0).show();
        this$0.finish();
    }
}
```

AuthLogic을 클릭해 이동했습니다.

AndroidManifest.xml | LoginActivity | AuthLogic

```
package com.ctf.solidapp.ui.login;

import android.util.Base64;
import kotlin.Metadata;
import kotlin.jvm.internal.Intrinsics;
import kotlin.text.Charsets;

/* compiled from: Authenticator.kt */
@Metadata(d1 = {"\u0000\"\n\u0002\u0018\u0002\n\u0002\u0010\u0000\n\u0002\b\u0002\n\u0002\u0010\u000e\n\u0000\n\u0002\u0010\b\n\u0002\b\u0004\n\u0002\u0010\u000b\
/* loaded from: classes.dex */
public final class AuthLogic {
    private final String decodedUsername = decodeBase64("cmVkdGVhbQ==");
    private final String staticString = "gttcurct";
    private final int key = 6;

    private final String decodeBase64(String encoded) {
        byte[] decodedBytes = Base64.decode(encoded, 0);
        Intrinsics.checkNotNullExpressionValue(decodedBytes, "decodedBytes");
        return new String(decodedBytes, Charsets.UTF_8);
    }

    private final String xorString(String input, int key) {
        StringBuilder sb = new StringBuilder();
        int length = input.length();
        for (int i = 0; i < length; i++) {
            sb.append((char) (input.charAt(i) ^ key));
        }
        String sb2 = sb.toString();
        Intrinsics.checkNotNullExpressionValue(sb2, "xorResult.toString()");
        return sb2;
    }

    public final boolean isUsernameValid(String inputUsername) {
        Intrinsics.checkNotNullParameter(inputUsername, "inputUsername");
        return Intrinsics.areEqual(inputUsername, this.decodedUsername);
    }

    public final boolean isPasswordValid(String inputPassword) {
        Intrinsics.checkNotNullParameter(inputPassword, "inputPassword");
        return Intrinsics.areEqual(inputPassword, xorString(this.staticString, this.key));
    }
}
```

Username과 Password를 확인할 수 있습니다

복호화 되어있는 것 같아 key = 6을 통해 복호화를 진행하였고

flag{redteam_arrester}를 알아낼 수 있었습니다.

문제를 해결하며 처음에는 어떤 파일을 봐야 할지 감이 잡히지 않아 이것저것 눌러보며 차근차근 문제를 해결해 나갔습니다.