



변조 앱 분석

91812452 서동훈

Contents

001. 개요

- › 변조앱이란?
- › 변조앱 선정
- › 변조앱 기능

002. 분석

- › 라이브러리
- › 후킹 함수

003. Q&A

변조 앱이란?

개요

분석

Q & A

- 변조 앱 or Mod(Modification) 앱 -

기존에 존재하는 앱을 제 3자가 수정한 앱

변조 앱 이란?

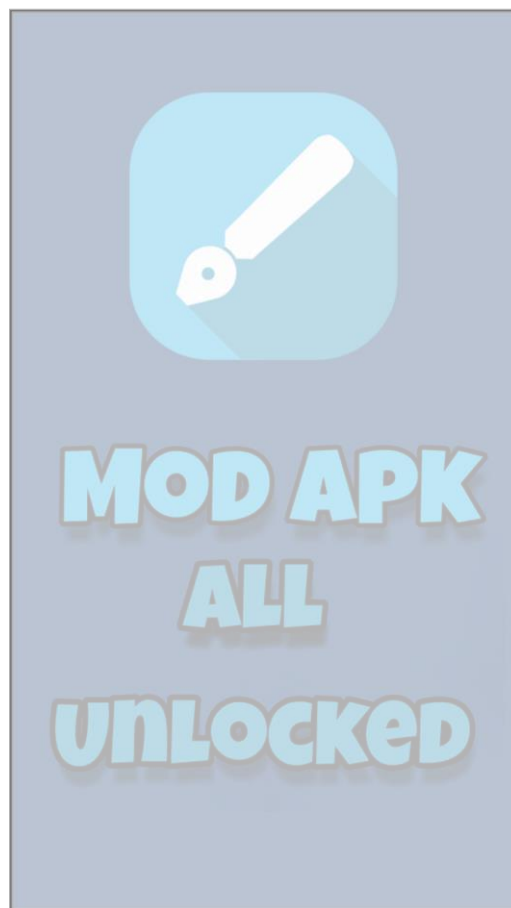
개요

분석

Q & A



〈기능 변경 및 추가〉



〈인증 우회〉



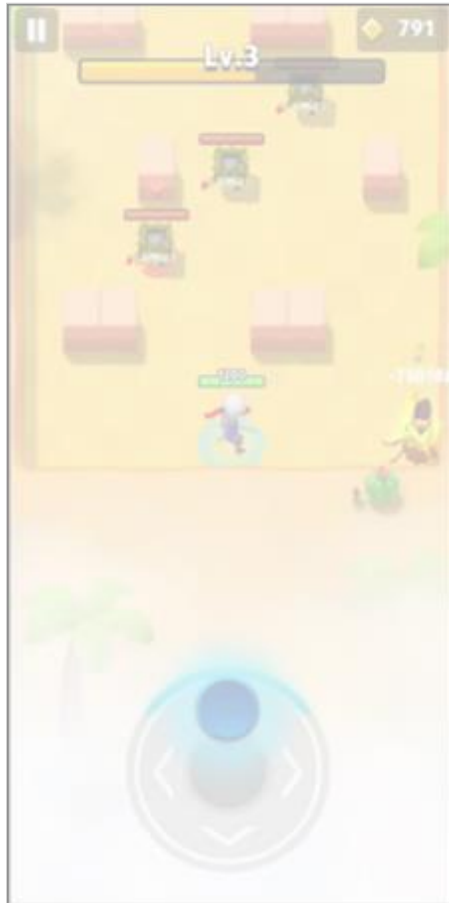
〈광고, 멀웨어 삽입〉

변조 앱 이란?

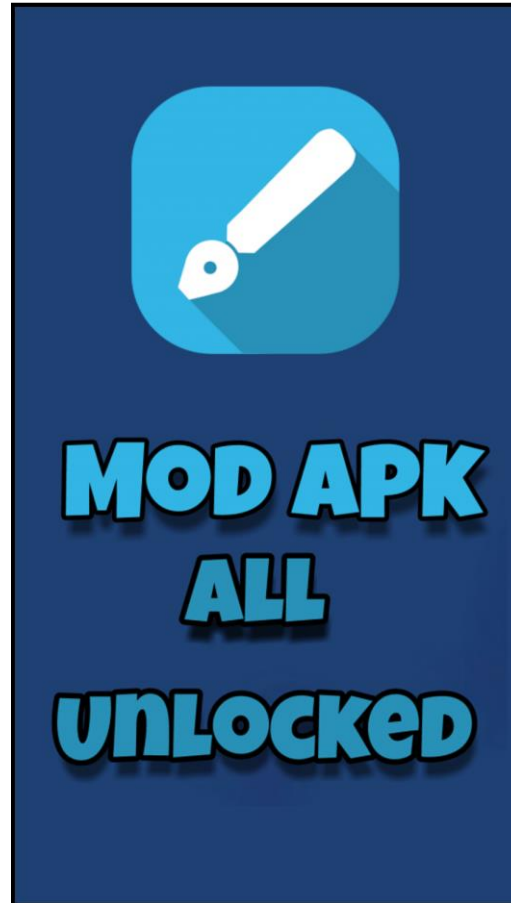
개요

분석

Q & A



〈기능 변경 및 추가〉



〈인증 우회〉



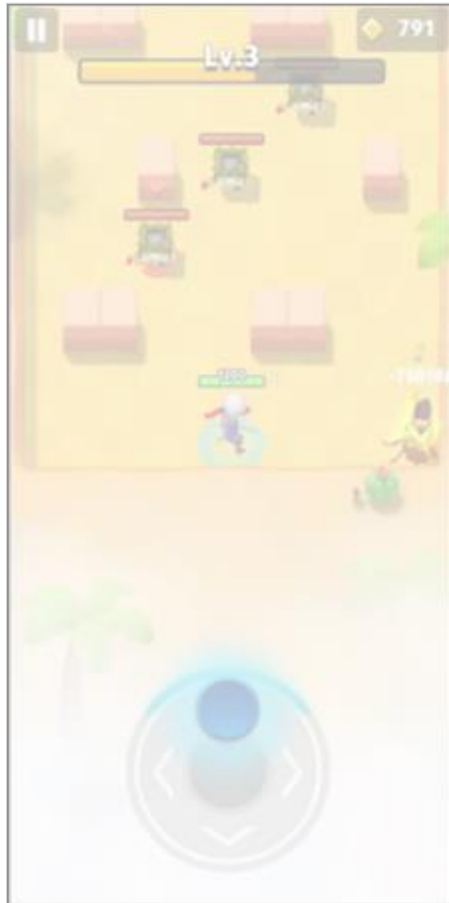
〈광고, 멀웨어 삽입〉

변조 앱 이란?

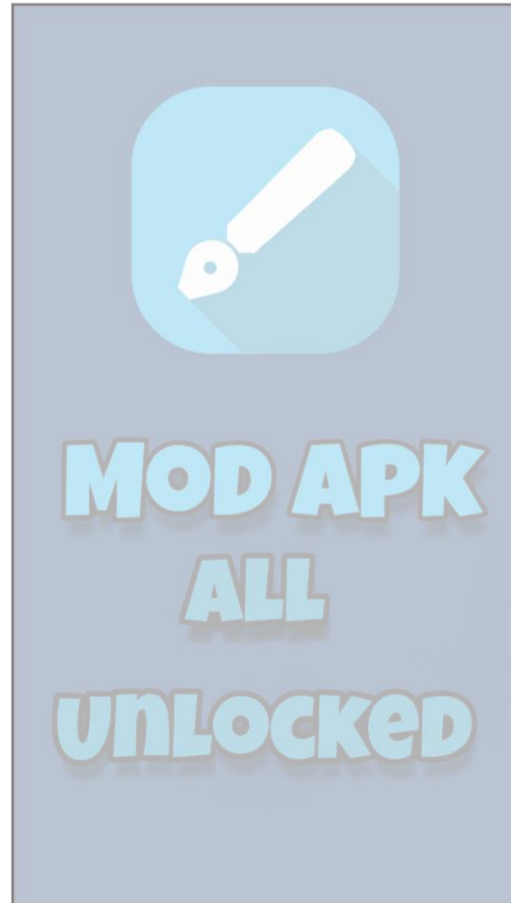
개요

분석

Q & A



〈기능 변경 및 추가〉



〈인증 우회〉



〈광고, 멀웨어 삽입〉

변조 앱 선정

- 앱 이름 : Combat [redacted] - [redacted]
- 출처 : BlackMod.net
- 선정 조건
 - 게임 플레이 -> 단순한 플레이 방식인가?
 - 분석 -> 패킹과 같은 안티 디버깅 유무
 - 네트워크 -> 데미지 계산에서의 네트워크 사용 여부

개요

분석

Q & A



〈실행 시 인터넷 요구〉



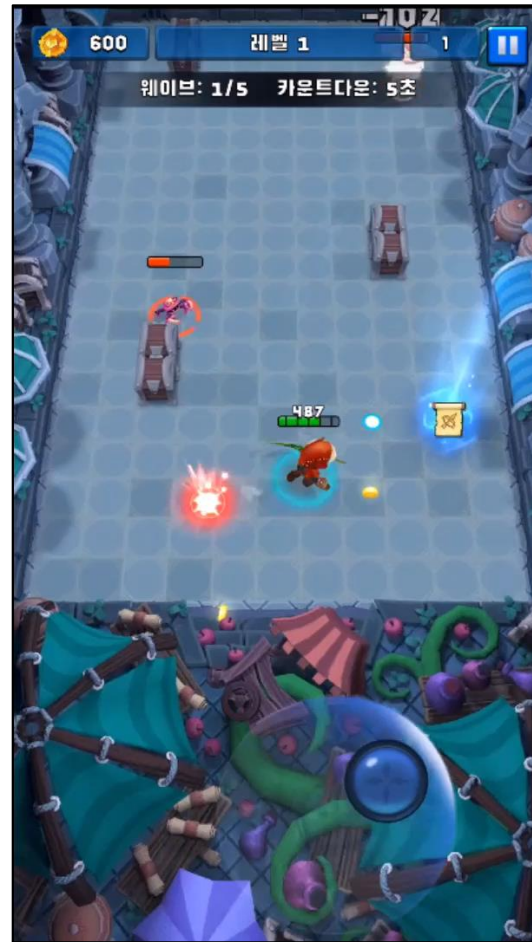
〈플레이 도중 요구 X〉

변조 앱 기능

개요

분석

Q & A



〈원본 앱〉



〈변조 앱〉

변조 앱 기능

개요

분석

Q & A



<원본 앱>



<변조 앱>



Mod Menu

- > il2cpp 및 기타 안드로이드 게임에 메뉴를 띄우는 모드
- > 오픈소스로 변조앱에서 자주 사용한다.

변조 앱 기능



HIGH HP

- > 유저 **HP** 대량 증가
- > 적들로 인한 **피해 X**



〈적용 전〉



〈적용 후〉

개요

분석

Q & A

변조 앱 기능



ONE SHOOT KILL

> 공격이 적 최대 HP 만큼 피해 입힘



<적용 전>



<적용 후>

개요

분석

Q & A

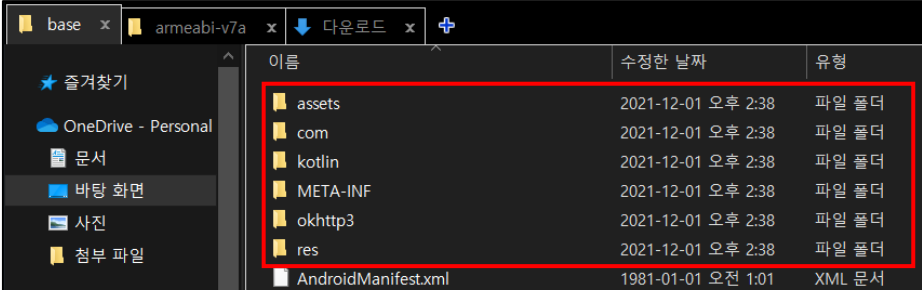
라이브러리

- 변조 부분을 알아내기 위한 라이브러리 획득
- 원본/변조 앱 라이브러리 추출 과정
 - Play Store/BlackMod.net을 통해 앱 설치 진행
 - > /data/app/[package]/base.apk 추출
 - > 추출한 APK 압축 해제
 - > library 획득
- 원본 앱의 경우 lib 폴더가 존재 하지 않음
 - > Play Store에서 App Bundle 형태로 앱 배포 시 네이티브 라이브러리를 압축하지 않은 형태로 배포
(enableUncompressedNativeLibs = False)

개요

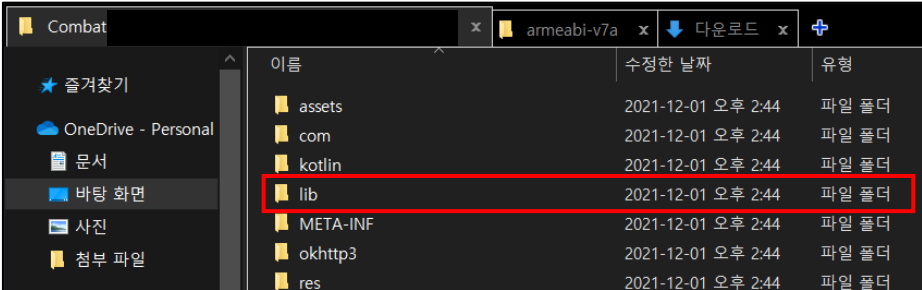
분석

Q & A



이름	수정된 날짜	유형
assets	2021-12-01 오후 2:38	파일 폴더
com	2021-12-01 오후 2:38	파일 폴더
kotlin	2021-12-01 오후 2:38	파일 폴더
META-INF	2021-12-01 오후 2:38	파일 폴더
okhttp3	2021-12-01 오후 2:38	파일 폴더
res	2021-12-01 오후 2:38	파일 폴더
AndroidManifest.xml	1981-01-01 오전 1:01	XML 문서

〈원본 앱〉



이름	수정된 날짜	유형
assets	2021-12-01 오후 2:44	파일 폴더
com	2021-12-01 오후 2:44	파일 폴더
kotlin	2021-12-01 오후 2:44	파일 폴더
lib	2021-12-01 오후 2:44	파일 폴더
META-INF	2021-12-01 오후 2:44	파일 폴더
okhttp3	2021-12-01 오후 2:44	파일 폴더
res	2021-12-01 오후 2:44	파일 폴더

〈변조 앱〉

라이브러리

APK DOWNLOADER
Free Download APK at Maximum Speed!

TikTok Twitter Facebook Lite

co [redacted]

Android 장치 구성

Device: 기본값 Architecture: 기본값 Android: 기본값

고급 옵션

Generate Download Link →

Combat [redacted]
0.26.0 · 2021년 12월 14일
ChillGaming

Device: 기본값 · Architecture: 기본값 · Android: 기본값

다운로드 위치: 주 서버

다운로드 설치

armeabi-v7a, arm64-v8a

게임 · 최신 업데이트

Car Stunt Races: Meg... Poppy Rope Game Dashero: 소드 & 매직 (...)

앱 · 최신 업데이트

NokoPrint - 모바일 인쇄 STARZPLAY HD Cinema Movies

〈apkcombo〉

개요

분석

Q & A

파일 홈 공유 보기

즐거찾기에 고정 복사 붙여넣기 경로 복사 바로 가기 붙여넣기 이동 위치 복사 위치 삭제 이를 바꾸기 새 폴더 빠른 연결 속성 히스토리

클립보드 구성 새로 만들기 열기

Combat Quest - Archero Action_0.25.0_apkcombo.

이름	수정한 날짜	유형
assets	2021-12-01 오후 2:44	파일 폴더
com	2021-12-01 오후 2:44	파일 폴더
kotlin	2021-12-01 오후 2:44	파일 폴더
lib	2021-12-01 오후 2:44	파일 폴더
META-INF	2021-12-01 오후 2:44	파일 폴더
okhttp3	2021-12-01 오후 2:44	파일 폴더
res	2021-12-01 오후 2:44	파일 폴더
AndroidManifest.xml	1981-01-01 오전 1:01	XML 문서
billing.properties	1981-01-01 오전 1:01	PROPERTIES 파일
classes.dex	1981-01-01 오전 1:01	DEX 파일
classes2.dex	1981-01-01 오전 1:01	DEX 파일
firebase-analytics.properties	1981-01-01 오전 1:01	PROPERTIES 파일

〈원본 앱〉

라이브러리

개요

분석

Q & A

- 원본 앱 네이티브 라이브러리

- libFirebaseCppAnalytics.so
- libFirebaseCppApp-7_2_0.so
- libil2cpp.so
- libmain.so
- libunity.so

- 변조 앱 네이티브 라이브러리

- libFirebaseCppAnalytics.so
- libFirebaseCppApp-7_2_0.so
- libbmt.so => BlackMod 라이브러리
- libil2cpp.so
- libmain.so
- libmod.so => 무적, 한방 등 주요 후킹 라이브러리
- libunity.so

libFirebaseCppAnalytics.so	1981-01-01 오전 1:01	SO 파일	44KB
libFirebaseCppApp-7_2_0.so	1981-01-01 오전 1:01	SO 파일	4,155KB
libil2cpp.so	1981-01-01 오전 1:01	SO 파일	61,300KB
libmain.so	1981-01-01 오전 1:01	SO 파일	19KB
libunity.so	1981-01-01 오전 1:01	SO 파일	13,121KB

<원본 앱>

libbmt.so	1980-12-31 오후 5:00	SO 파일	2,671KB
libFirebaseCppAnalytics.so	2021-11-24 오전 8:21	SO 파일	44KB
libFirebaseCppApp-7_2_0.so	2021-11-24 오전 8:21	SO 파일	4,155KB
libil2cpp.so	2021-11-24 오전 8:21	SO 파일	61,300KB
libmain.so	2021-11-24 오전 8:21	SO 파일	19KB
libmod.so	2021-11-24 오전 8:21	SO 파일	507KB
libunity.so	2021-11-24 오전 8:21	SO 파일	13,121KB

<변조 앱>

```
package com.unity3d.player;

import android.app.Activity;
import android.content.Intent;
import android.content.res.Configuration;
import android.os.Bundle;
import android.os.Process;
import android.view.KeyEvent;
import android.view.MotionEvent;

public class UnityPlayerActivity extends Activity implements IUnityPlayerLifecycleEvents {
    protected UnityPlayer mUnityPlayer;

    /* access modifiers changed from: protected */
    public String updateUnityCommandLineArguments(String str) {
        return str;
    }

    /* access modifiers changed from: protected */
    public void onCreate(Bundle bundle) {
        requestWindowFeature(1);
        super.onCreate(bundle);
        getIntent().putExtra("unity", updateUnityCommandLineArguments(getIntent().getStringExtra("unity")));
        this.mUnityPlayer = new UnityPlayer(this, this);
        setContentView(this.mUnityPlayer);
        this.mUnityPlayer.requestFocus();
    }
}
```

〈원본 앱〉

```
package com.unity3d.player;

import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.content.res.Configuration;
import android.os.Bundle;
import android.os.Process;
import android.view.KeyEvent;
import android.view.MotionEvent;
import net.thdl.THAdsManager;
import uk.lgl.modmenu.StaticActivity;

public class UnityPlayerActivity extends Activity implements IUnityPlayerLifecycleEvents {
    protected UnityPlayer mUnityPlayer;

    public void onCreate(Bundle bundle) {
        onCreateTH(bundle);
        THAdsManager.a((Context) this);
    }

    public void onStart() {
        super.onStart();
        THAdsManager.b("reward");
    }


    /* access modifiers changed from: protected */
    public String updateUnityCommandLineArguments(String str) {
        return str;
    }

    public void onCreateTH(Bundle bundle) {
        StaticActivity.Start(this);
        requestWindowFeature(1);
        super.onCreate(bundle);
        getIntent().putExtra("unity", updateUnityCommandLineArguments(getIntent().getStringExtra("unity")));
        this.mUnityPlayer = new UnityPlayer(this, this);
        setContentView(this.mUnityPlayer);
        this.mUnityPlayer.requestFocus();
    }
}
```

〈변조 앱〉


```
public class StaticActivity {
    public static String cacheDir;

    public static void Start(final Context context) {
        if (Build.VERSION.SDK_INT < 23 || Settings.canDrawOverlays(context)) {
            new Handler().postDelayed(new Runnable() {
                public void run() {
                    context.startService(new Intent(context, FloatingModMenuService.class));
                }
            }, DefaultRenderersFactory.DEFAULT_ALLOWED_VIDEO_JOINING_TIME_MS);
        } else {
            context.startActivity(new Intent("android.settings.action.MANAGE_OVERLAY_PERMISSION", Uri.parse("package:" + context.getPackageName())));
            Process.killProcess(Process.myPid());
        }
    }
}
```



- 변조 앱에서 다음 순서를 통해 'mod' 라이브러리를 불러온다.
com.unity3d.player.UnityPlayer.onCreate() -> onCreateTH()
-> uk.lgl.modmenu.StaticActivity.Start()
-> uk.lgl.modmenu.FloatingModMenuService.onCreate()

```
public void onCreate() {
    super.onCreate();
    System.loadLibrary("mod");
    initFloating();
    initAlertDiag();
    final Handler handler = new Handler();
    handler.post(new Runnable() {
        public void run() {
            FloatingModMenuService.this.Thread();
            handler.postDelayed(this, 1000);
        }
    });
}
```

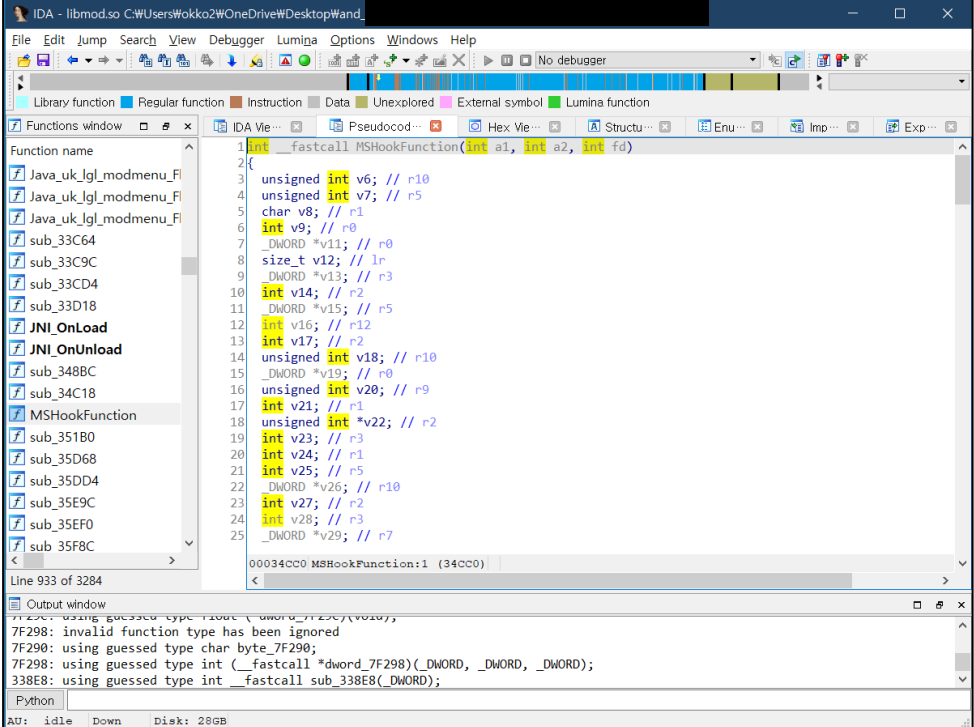

후킹 함수

개요

분석

Q & A

- MSHookFunction(void *symbol, void *hook, void **old)
=> Java나 Object-C와 같은 높은 수준의 런타임 동작을 수정할 수 있는 함수로 다음과 같이 지원되는 기능(예: 사용자 지정 클래스 로더, 런타임 API)을 사용하여 동작한다.
- ida의 기능 중 함수가 호출되는 부분을 확인할 수 있는 xrefs 기능을 사용하여 후킹 코드가 있는 부분을 탐색



```
1 int __fastcall MSHookFunction(int a1, int a2, int fd)
2 {
3     unsigned int v6; // r10
4     unsigned int v7; // r5
5     char v8; // r1
6     int v9; // r0
7     _DWORD *v11; // r0
8     size_t v12; // lr
9     _DWORD *v13; // r3
10    int v14; // r2
11    _DWORD *v15; // r5
12    int v16; // r12
13    int v17; // r2
14    unsigned int v18; // r10
15    _DWORD *v19; // r0
16    unsigned int v20; // r9
17    int v21; // r1
18    unsigned int *v22; // r2
19    int v23; // r3
20    int v24; // r1
21    int v25; // r5
22    _DWORD *v26; // r10
23    int v27; // r2
24    int v28; // r3
25    _DWORD *v29; // r7
26
27    000340C0 MSHookFunction:1 (340C0)
28
29    Line 933 of 3284
30
31    Output window
32    7F298: using guessed type 'DWORD' *v10;
33    7F298: invalid function type has been ignored
34    7F290: using guessed type char byte_7F290;
35    7F298: using guessed type int (__fastcall *dword_7F298)(_DWORD, _DWORD, _DWORD);
36    338E8: using guessed type int __fastcall sub_338E8(_DWORD);
37
38    Python
39
40    AU: idle Down Disk: 28GB
```

〈MSHookFunction〉

후킹 함수

[개요](#)[분석](#)[Q & A](#)

xrefs to MSHookFunction			
Direction	Type	Address	Text
Up	o	LOAD:00001400	Elf32_Sym <aMshookfunction - byte_8460, MSHookFunction, 0x4EC, 0x12, W; "MSHookFunction"
Up	j	j_MSHookFunction+8	LDR PC, [R12, # (MSHookFunction_ptr - 0x7DD04)]; MSHookFunction
Do...	o	.got:MSHookFunction_ptr	DCD MSHookFunction

Line 2 of 3

OK Cancel Search Help

〈xrefs to MSHookFunction〉



```
1 // attributes: thunk
2 int __fastcall j_MSHookFunction(int a1, int a2, int fd)
3 {
4     return MSHookFunction(a1, a2, fd);
5 }
```

〈j_MSHookFunction〉



xrefs to j_MSHookFunction			
Direction	Type	Address	Text
Do...	p	sub_33D18+180	BL j_MSHookFunction
Do...	p	sub_33D18+1B0	BL j_MSHookFunction
Do...	p	sub_33D18+1E0	BL j_MSHookFunction

Line 2 of 3

OK Cancel Search Help

〈xrefs to j_MSHookFunction〉

```
v6 = sub_337F4(off_7F008);
if ( v6 )
    v6 += 0x2D24AA8;
j_MSHookFunction(v6, sub_33C64, &dword_7F294);
v7 = sub_337F4(off_7F008);
if ( v7 )
    v7 += 0x2D0D414;
j_MSHookFunction(v7, sub_33C9C, &dword_7F298);
v8 = sub_337F4(off_7F008);
if ( v8 )
    v8 += 0x2D22E10;
j_MSHookFunction(v8, sub_33CD4, &dword_7F29C);
if ( (v0 & 1) != 0 )
```

〈part of sub_33D18〉

후킹 함수

[개요](#)[분석](#)[Q & A](#)

```
v6 = sub_337F4(off_7F008);  
if ( v6 )  
    v6 += 0x2D24AA8;  
j_MSHookFunction(v6, sub_33C64, &dwor_7F294);  
v7 = sub_337F4(off_7F008);  
if ( v7 )  
    v7 += 0x2D0D414;  
j_MSHookFunction(v7, sub_33C9C, &dwor_7F298);  
v8 = sub_337F4(off_7F008);  
if ( v8 )  
    v8 += 0x2D22E10;  
j_MSHookFunction(v8, sub_33CD4, &dwor_7F29C);  
if ( (v0 & 1) != 0 )  
    operator delete(v10);  
if ( (v13 & 1) != 0 )  
    operator delete(v11);  
if ( (v14 & 1) != 0 )  
    operator delete(p);  
return 0;
```

<sub_33D18>

.data:0007F008	off_7F008	DCD	aLibil2cppSo	; DATA XREF: sub_33D18:loc_33D88↑o
.data:0007F008				; sub_33D18+78↑r ...
.data:0007F008				; "libil2cpp.so"

<off_7F008>



.rodata:0007197B	aLibil2cppSo	DCB	"libil2cpp.so",0	; DATA XREF: sub_33D18+78↑o
------------------	--------------	-----	------------------	-----------------------------

<aLibil2cppSo>



라이브러리를 불러오는 함수임을 추측할 수 있다.

후킹 함수

```
v6 = sub_337F4(off_7F008);  
if ( v6 )  
    v6 += 0x2D24AA8;  
j_MSHookFunction(v6, sub_33C64, &dwor_7F294);  
v7 = sub_337F4(off_7F008);  
if ( v7 )  
    v7 += 0x2D0D414;  
j_MSHookFunction(v7, sub_33C9C, &dwor_7F298);  
v8 = sub_337F4(off_7F008);  
if ( v8 )  
    v8 += 0x2D22E10;  
j_MSHookFunction(v8, sub_33CD4, &dwor_7F29C);  
if ( (v0 & 1) != 0 )  
    operator delete(v10);  
if ( (v13 & 1) != 0 )  
    operator delete(v11);  
if ( (v14 & 1) != 0 )  
    operator delete(p);  
return 0;
```

<sub_33D18>

```
C:\Users\okko2\OneDrive\Desktop\Tools\IL2CppDumper-v6.6.5>IL2CppDumper.exe "C:\Users\okko2\OneDrive\Desktop\Assets\bin\Data\Managed\Metadata\global-metadata.dat" C:\Users\okko2\OneDrive\Desktop\Assets\bin\Data\Managed\Metadata\global-metadata.dat C:\Users\okko2\OneDrive\Desktop\Assets\bin\Data\Managed\Metadata\global-metadata.dat  
Initializing metadata...  
Metadata Version: 24.2  
Initializing il2cpp file...  
Applying relocations...  
IL2Cpp Version: 24.2  
Searching...  
Change il2cpp version to: 24.4  
CodeRegistration : 393fc14  
MetadataRegistration : 3940000  
Dumping...  
Done!  
Generate struct...  
Done!  
Generate dummy dll...  
Done!  
Press any key to exit...
```

<il2CppDumper>

```
614791 // RVA: 0x2D249B0 Offset: 0x2D249B0 VA: 0x2D249B0 Slot: 5  
614792 public int GetNextLevelDamage(IHeroSettings heroSettings) { }  
614793  
614794 // RVA: 0x2D24AA8 Offset: 0x2D24AA8 VA: 0x2D24AA8 Slot: 6  
614795 public int GetCurrentHp(IHeroSettings heroSettings) { }  
614796  
614797 // RVA: 0x2D24CF8 Offset: 0x2D24CF8 VA: 0x2D24CF8 Slot: 7  
614798 public int GetNextLevelHp(IHeroSettings heroSettings) { }  
614799
```

<dump.cs>

후킹 함수

[개요](#)[분석](#)[Q & A](#)

```
v6 = sub_337F4(off_7F008);  
if ( v6 )  
    v6 += 0x2D24AA8;  
j_MSHookFunction(v6, sub_33C64, &dw_7F294);  
v7 = sub_337F4(off_7F008);  
if ( v7 )  
    v7 += 0x2D0D414;  
j_MSHookFunction(v7, sub_33C9C, &dw_7F298);  
v8 = sub_337F4(off_7F008);  
if ( v8 )  
    v8 += 0x2D22E10;  
j_MSHookFunction(v8, sub_33CD4, &dw_7F29C);  
if ( (v0 & 1) != 0 )  
    operator delete(v10);  
if ( (v13 & 1) != 0 )  
    operator delete(v11);  
if ( (v14 & 1) != 0 )  
    operator delete(p);  
return 0;
```

〈sub_33D18〉

int GetCurrentHp(IHeroSettings heroSettings) { }

**void DealDamage(IPlayerParameters targetPlayer,
float adjustDamage) { }**

float get_Damage() { }

후킹 함수

[개요](#)[분석](#)[Q & A](#)

```
int sub_33C64()
{
    int v0; // r4
    int result; // r0
    v0 = byte_7F290;
    result = dword_7F294();
    if ( v0 )
        result *= "utIcE21__widen_and_group_intEPcS2_S2_S2_RS2_S3_RKNS_6localeE";
    return result;
}
```

<sub_33C64(GetCurrentHP)>

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xAAB7F290)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
aab7f290  00 00 00 00 00 90 42 e5 00 40 2a e5 00 70 26 e5  ....B..@*..p&.
aab7f2a0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```

<Turn off HIGHHP>

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xAAB7F290)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
aab7f290  01 00 00 00 00 90 42 e5 00 40 2a e5 00 70 26 e5  ....B..@*..p&.
aab7f2a0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```

<Turn on HIGHHP>

후킹 함수

[개요](#)[분석](#)[Q & A](#)

```
int sub_33C64()
{
    int v0; // r4
    int result; // r0

    v0 = byte_7F290;
    result = dword_7F294();
    if ( v0 )
        result *= "utIcE21_widen_and_group_intEPcS2_S2_S2_RS2_S3_RKNS_6localeE";
    return result;
}
```

<sub_33C64(GetCurrentHP)>

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xAAB7F294)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
aab7f294  00 90 42 e5 00 40 2a e5 00 70 26 e5 00 00 00 00  ..B..@*...p&....
aab7f2a4  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xe5429000)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
e5429000  70 4c 2d e9 10 b0 8d e2 04 f0 1f e5 b0 4a ea b7  pL-.....J..
e5429010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```

push {r4, r5, r6, sl, fp, lr}
add fp, sp, #0x10
ldr pc, [pc, #-4]
.BYTE 0xb0, 0x4a, 0xea, 0xb7

후킹 함수

[개요](#)[분석](#)[Q & A](#)

```
int __fastcall sub_33C9C(int a1, int a2, int a3)
{
    int v3; // s0

    v3 = 0;
    if ( !byte 7F290 )
        v3 = a3;
    return dword_7F298(a1, a2, v3);
}
```

〈sub_33C9C(DealDamage)〉

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xAAB7F290)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
aab7f290  00 00 00 00 00 90 42 e5 00 40 2a e5 00 70 26 e5  ....B..@*..p&.
aab7f2a0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```

〈Turn off HIGHHP〉

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xAAB7F290)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
aab7f290  01 00 00 00 00 90 42 e5 00 40 2a e5 00 70 26 e5  ....B..@*..p&.
aab7f2a0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```

〈Turn on HIGHHP〉

후킹 함수

[개요](#)[분석](#)[Q & A](#)


```
int __fastcall sub_33C9C(int a1, int a2, int a3)
{
    int v3; // s0

    v3 = 0;
    if ( !byte_7F290 )
        v3 = a3;
    return dword_7F298(a1, a2, v3);
}
```

<sub_33C9C(DealDamage)>

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xAAB7F298)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
aab7f298  00 40 2a e5 00 70 26 e5 00 00 00 00 00 00 00 00  .@*..p&.....
aab7f2a8  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xe52a4000)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
e52a4000  f0 4f 2d e9 1c b0 8d e2 04 f0 1f e5 1c d4 e8 b7  .0-.....
e52a4010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```



```
push {r4, r5, r6, r7, r8, sb, sl, fp, lr}
add fp, sp, #0x1c
ldr pc, [pc, #-4]
.BYTE 0x1c, 0xd4, 0xe8, 0xb7
```

후킹 함수

[개요](#)[분석](#)[Q & A](#)

```
int sub_33CD4()
{
    int v0; // r4
    float v1; // r0
    float v2; // s0

    v0 = byte_7F291;
    v1 = dword_7F29C();
    v2 = v1 * 1000.0;
    if ( !v0 )
        v2 = v1;
    return LODWORD(v2);
}
```

<sub_33CD4(getDamage)>

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xAAB7F291)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
aab7f291  00 00 00 00 90 42 e5 00 40 2a e5 00 70 26 e5 00  ....B..@*..p&..
aab7f2a1  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```

<Turn off ONE SHOOT KILL>

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xAAB7F291)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
aab7f291  01 00 00 00 90 42 e5 00 40 2a e5 00 70 26 e5 00  ....B..@*..p&..
aab7f2a1  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```

<Turn on ONE SHOOT KILL>

후킹 함수

[개요](#)[분석](#)[Q & A](#)

```
int sub_33CD4()
{
    int v0; // r4
    float v1; // r0
    float v2; // s0

    v0 = byte_7F291;
    v1 = dword_7F29C();
    v2 = v1 * 1000.0;
    if ( !v0 )
        v2 = v1;
    return LODWORD(v2);
}
```

<sub_33CD4(getDamage)>

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xAAB7F29c)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
aab7f29c  00 70 26 e5 00 00 00 00 00 00 00 00 00 00 00 00  .p&.....
aab7f2ac  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```

```
[SM G930S::com.chillgaming.combatquest]-> console.log(hexdump(ptr(0xe5267000)));
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
e5267000  0c 00 90 e5 1e ff 2f e1 04 f0 1f e5 18 2e ea b7  ....//.....
e5267010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  .....
```



```
ldr r0, [r0, #0xc]
bx lr
ldr pc, [pc, #-4]
.BYTE 0x18, 0x2e, 0xea, 0xb7
```

Q & A