

Ghost In The Game

Android MMO Game Security

束骏亮

GoSSIP-LoCCS

上海交通大学



introduction



为什么要研究MMO类游戏的安全？

- **频繁的在线互动**
 - 对于公平性有着很强的需求
- **庞大的用户群体**
 - 容易成为不法分子的攻击目标
- **强大的盈利能力**
 - 游戏开发者更加关心其安全性
- **运行环境多样且不可控**
 - 对防守方提出了更高的技术要求
- **代码架构复杂**
 - 在分析时面临更多的困难（也更加有趣）

Massively
Multiplayer
Online

1



Android平台游戏类应用概况

2



MMO类游戏安全分析技术

3



MMO类游戏常见威胁

4



MMO类游戏常见保护技术

JOSS



Android平台游戏类应用概况

GOSS

Android平台游戏类应用概况

1

387亿美元

2016年，全球移动平台游戏营收387亿美元，其中近1/4来自中国市场。

2

3亿美元

2017年第一季度，MIXI旗下移动游戏“怪物弹珠”营收3亿美元，坊间传闻腾讯旗下“王者荣耀”营收120亿人民币。

3

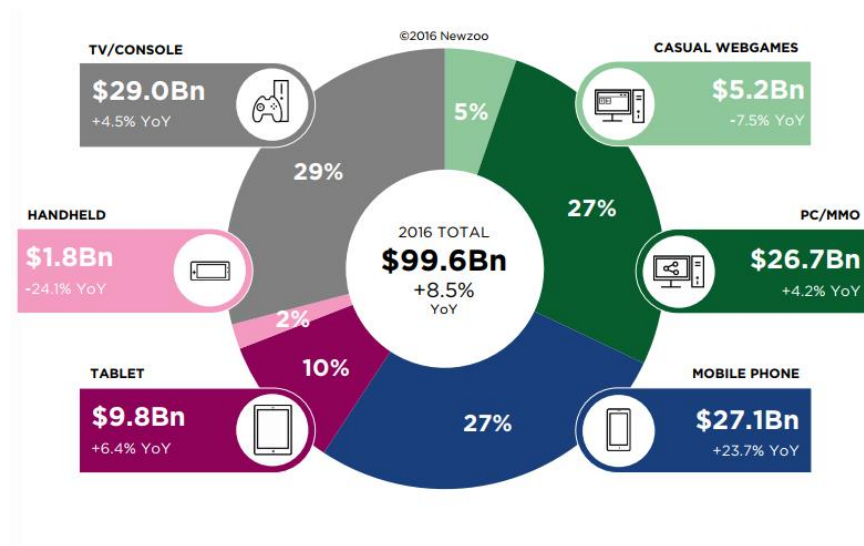
3.38万家

2016年底，国内的移动游戏内容提供商达到3.38万家。

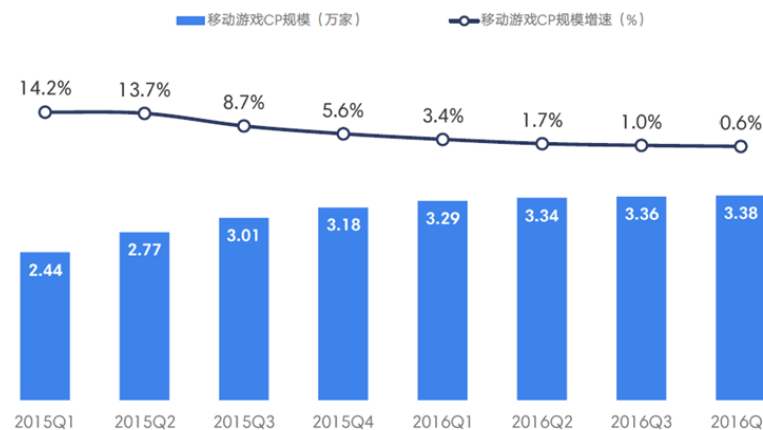
4

11.3亿台

2016年底，国内移动游戏活跃设备数达到了11.3亿台。



国内移动游戏CP规模和增速
(2015Q1-2016Q4)



数据来源: TalkingData 移动数据研究中心, 2015Q1-2016Q4

Android平台游戏类（MMO）应用概况

来源

国外以Google Play为主，国内以TAPTAP以及各大应用市场为主。

形式

形式上仍然是传统的Android应用程序，以APK的形式在网络上传播。

差异

和传统的Android应用程序相比，主要基于主流的第三方游戏引擎，以非JAVA的语言编写；应用体积巨大；在本地保存大量的图片、音乐、视频等素材。

特点

主要以Unity3D和Cocos2D两个主流的第三方游戏引擎为主，根据所使用引擎的不同，在代码和资源管理上呈现不同的结构特点。





MMO类游戏安全分析技术

GOSS

MMO类游戏安全分析技术

素材

文字、图片、音频、视频等构成游戏主要感官冲击的元素，以各自不同的格式保存在设备上。

逻辑

用于决定游戏行为的规则，用于决定游戏的走向、播放和管理各种素材，主要以代码的方式存在，具体格式因引擎不同而不同。

引擎

游戏的心脏，为上层的逻辑代码提供各种接口，决定了一个游戏在空间上的结构以及各个组件之间的关系。在Android平台上，一般以native lib的形式存在，游戏开发者使用其提供的各类接口完成游戏的开发。



```
56 void OnTriggerStay(Collider c)
57 {
58     var fire = c.GetComponent<Fire>();
59     if (fire && fire.alive)
60     {
61         float dist = 1-(((transform.position - fire.transform.position).magnitude));
62         NearHeat(dist);
63     }
64
65     var smoke = c.GetComponent<SmokeParticle>();
66 }
```

对于MMO类游戏的分析，我们一般围绕着引擎展开，以分析游戏的逻辑代码为主，辅以对素材的分析。

MMO类游戏安全分析技术-引擎

Unity3D

当前移动游戏引擎领域当之无愧的领跑者

Cocos2D

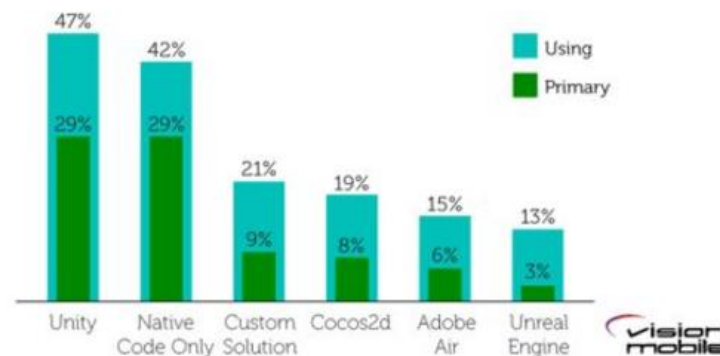
2D游戏引擎的翘楚，在国内的市场占有量要高于Unity3D

自定义引擎

部分移动游戏大厂会有自定义的游戏引擎，比如网易的neox。对于这些引擎的分析有助于我们分析同一个厂商开发的其他游戏。

UNITY BEATS ALL OTHER GAME DEVELOPMENT TOOLS, USED BY 47% OF GAME DEVELOPERS

% of game developers using and primarily using each tool (n=3,467)



Source: Developer Economics: State of the Developer Nation Q3 2014 | www.DeveloperEconomics.com/go
Licensed under CC BY ND | Copyright VisionMobile

MMO类游戏安全分析技术-程序分析

如何快速区分不同的游戏引擎

```
izhuer@xiaoZtongxuedeMacBook-Air ~/Desktop/report/dtcq cat AndroidManifest.xml | grep cocos
    <receiver android:name="org.cocos2dx.sdk.DTBCReceiver">
    <provider android:authorities="downloads.sh.lilith.dgame.lemon" android:name="com.igexin
.download.DownloadProvider" android:process=":pushservice"/>    <receiver android:name="org.coc
os2dx.sdk.PushMessageReciever">
izhuer@xiaoZtongxuedeMacBook-Air ~/Desktop/report/dtcq cat AndroidManifest.xml | grep unity
X izhuer@xiaoZtongxuedeMacBook-Air ~/Desktop/report/dtcq
```

- 查看AndroidManifest.xml文件
 - 一般来说，AndroidManifest.xml中会包含和游戏引擎相关的权限和字符串。也可以通过爆搜整个apk内的文件，来寻找“cocos2dx”和“unity”等关键字字符串
- 寻找特定的native lib文件
- 通过Google等搜索引擎搜索相关的新闻、信息

MMO类游戏安全分析技术-程序分析

如何快速区分不同的游戏引擎

- 查看AndroidManifest.xml文件
- 寻找特定的native lib文件
 - 不同的游戏引擎一般都会带有不同的native lib，在apk中的lib目录下能找到。基于cocos2d开发的戏一般会包含“cocos”和“lua”等字眼，基于unity3d开发的戏一般会有名为“libunity.so”的native lib。
- 通过Google等搜索引擎搜索相关的新闻、信息

libAudioCCReName.so	861 892
libAudioCore.so	1 444 104
libAudioEngine.so	1 503 300
libAudioEngineJni.so	5 296
libbdpush_V2_5.so	17 560
libbd_wsp_v1_0.so	55 708
libby-sdk-root-jni.so	13 432
libcocos2dlua.so	11 972 656
libcom_netease_androidcrashhandler_AndroidCrashHandler.so	308 680
libcom_netease_ps_codescanner.so	13 384
libcrypto.so	1 386 460
libentryexstd.so	107 784
libffmpeg.so	5 533 132
libfmodevent.so	386 436
libfmodex.so	1 148 596
libijkffmpeg.so	1 716 104
libijkplayer.so	347 192
libijkSDL.so	206 392
libijkutil.so	13 508
libngvideo.so	235 132
libunity.so	2 012 016

MMO类游戏安全分析技术-程序分析

如何快速区分不同的游戏引擎

- 查看AndroidManifest.xml文件
- 寻找特定的native lib文件
- 通过Google等搜索引擎搜索相关的新闻、信息

阴阳师是用Messiah引擎做的吗？

还是用的unity? Messiah有应用到哪些项目上? 战绩如何? 补充, 搜了下, 发现《天下》有用到 [图片]... 显示全部

添加评论 分享 邀请回答

查看全部 5 个回答

6 人赞同了该回答

不是, 应该是NeoX (简称nx...

Cocos2d-x引擎特点

- 开源
- 出色的2D支持，在国内被大规模使用
- 支持Lua和Javascript两种脚本语言，目前以Lua为主（得益于Lua优秀的C语言交互接口和更加小巧的解释器）



寻找逻辑脚本

- 目前主流的基于Cocos2d-x开发的游戏都使用了Lua作为脚本语言，和游戏主体相关的逻辑一般都以Lua为主体框架。
- 定位Lua脚本是所有分析工作的第一步。
- 定位Lua解释器是定位Lua脚本的第一步。



JOSS

MMO类游戏安全分析技术-程序分析-Cocos2d-x

定位Lua解释器

- 以native lib形式存在，一般和引擎在一个lib里
- 存放在lib/文件夹下
- 比较庞大
- 名字有特点

libAudioCCReName.so	861 892	513 628
libAudioCore.so	1 444 104	644 099
libAudioEngine.so	1 503 300	611 757
libAudioEngineJni.so	5 296	2 452
libbdpush_V2_5.so	17 560	8 129
libbd_wsp_v1_0.so	55 708	33 873
libby-sdk-root-jni.so	13 432	5 881
libcocos2dlua.so	11 972 656	5 028 733
libcom_netease_androidcrashhandler_AndroidCrashHandler.so	308 680	85 797
libcom_netease_ps_codescanner.so	13 384	5 609
libcrypto.so	1 386 460	448 238
libentryexstd.so	107 784	59 411
libffmpeg.so	5 533 132	2 449 571
libfmodevent.so	386 436	172 062
libfmodex.so	1 148 596	556 419
libijkffmpeg.so	1 716 104	887 528
libijkplayer.so	347 192	174 431
libijksdl.so	206 392	103 731
libijkutil.so	13 508	5 619
libngvideo.so	235 132	111 706

MMO类游戏安全分析技术-程序分析-Cocos2d-x

定位Lua解释器

- 在binary中寻找特定的接口字符串
- lua、cocos2d、cocos2dx等等

Function name

```
f lua_cocos2dx_sdk_SdkControllerDelegate_getPayChannel(  
f lua_cocos2dx_sdk_SdkControllerDelegate_checkOrderRec.  
f lua_cocos2dx_sdk_SdkControllerDelegate_switchAccount(l.  
f lua_cocos2dx_sdk_SdkControllerDelegate_relogin(lua_Stat.  
f lua_cocos2dx_sdk_SdkControllerDelegate_closeFlash(lua_S  
f lua_cocos2dx_sdk_SdkControllerDelegate_SetSDKScanner.  
f lua_cocos2dx_sdk_SdkControllerDelegate_sendCheckedO  
f lua_cocos2dx_sdk_SdkControllerDelegate_getCpid(lua_Sta  
f lua_cocos2dx_sdk_SdkControllerDelegate_videoInit(lua_Sta  
f lua_cocos2dx_sdk_SdkControllerDelegate_setPropString(lu  
f lua_cocos2dx_sdk_SdkControllerDelegate_isFlashClick(lua_  
f lua_cocos2dx_sdk_SdkControllerDelegate_isCCVideoReco  
f lua_cocos2dx_sdk_SdkControllerDelegate_isCCSupportSh.  
f lua_cocos2dx_sdk_AvatarLayer_isPosChanged(lua_State *)  
f lua_cocos2dx_sdk_SdkControllerDelegate_getSdkMode(lu  
f lua_cocos2dx_sdk_SdkControllerDelegate_getLoginUin(lua  
f lua_cocos2dx_sdk_SdkControllerDelegate_removeBuyCall.  
f lua_cocos2dx_sdk_SdkControllerDelegate_openQRScanne  
f lua_cocos2dx_sdk_SdkControllerDelegate_guestBind(lua_S  
f lua_cocos2dx_sdk_SdkControllerDelegate_stopCCService(l  
f lua_cocos2dx_sdk_SdkControllerDelegate_removeOrderRe  
f lua_cocos2dx_sdk_SdkControllerDelegate_adTraceEvent(lu  
f lua_cocos2dx_sdk_SdkControllerDelegate_videoPlayShow(  
f lua_cocos2dx_sdk_SdkControllerDelegate_getDeviceId(lua  
f lua_cocos2dx_sdk_SdkControllerDelegate_tryExit(lua_State  
f lua_cocos2dx_sdk_SdkControllerDelegate_getAppChannel  
f lua_cocos2dx_sdk_AvatarGroupLayer_setReorderDirty(lua_  
f lua_cocos2dx_sdk_AvatarSyncLayer_removeAllChildrenWit.  
f lua_cocos2dx_sdk_AvatarSyncLayer_enableSync(lua_State *  
f lua_cocos2dx_sdk_SdkControllerDelegate_ccVideoSetShar.  
f lua_cocos2dx_sdk_SdkControllerDelegate_getPayChannelE  
f lua_cocos2dx_sdk_SdkControllerDelegate_videoPlay(lua_St  
f lua_cocos2dx_sdk_SdkControllerDelegate_getSessionId(lu  
f lua_cocos2dx_sdk_SdkControllerDelegate_setPropInt(lua_?  
f lua_cocos2dx_sdk_SdkControllerDelegate_startCCService(  
f lua_cocos2dx_sdk_SdkControllerDelegate_netStateSetProp  
f lua_cocos2dx_sdk_SdkControllerDelegate_getProductCurr.  
f lua_cocos2dx_sdk_SdkControllerDelegate_openQRScanne  
f lua_cocos2dx_sdk_SdkControllerDelegate_adTraceEventWi
```


定位Lua脚本

- 由于Lua的开源特性，Lua解释器对Lua脚本的加载过程存在定制化差异
- 但是都会使用到一些固有的API，API列表可以在Lua源码的src/lapi.c中找到
- 寻找特定的接口调用，定位Lua脚本的加载过程
- lua_load

```




LABEL_14:
    if ( strncmp(v5, "L:grxx", 6u) )
    {
        if ( !strncmp(v5, "__sign_of_g18_enc__", 0x13u) )
        {
            memset((void *)v5, 0, 0x13u);
            lrc4::lrc4((lrc4 *)&v34);
            v32 = v5 + 19;
            v33 = v27 - 19;
            v24 = lua_load(v28, sub_312A88, &v32, a5);
        }
        else
        {
            v24 = luaL_loadbuffer(v28, v5, v27, a5);
        }
        v23 = v24;
        goto LABEL_33;
    }

```



逆向还原加载流程

- 定位Lua脚本的位置

 0B20D4
 0BAF71
 1E0D30
 2A9BF3
 2C3218
 2DC689
 2E3DA7
 3CE789
 3D46F7
 3DAAA5
 4A161D
 4B24BC

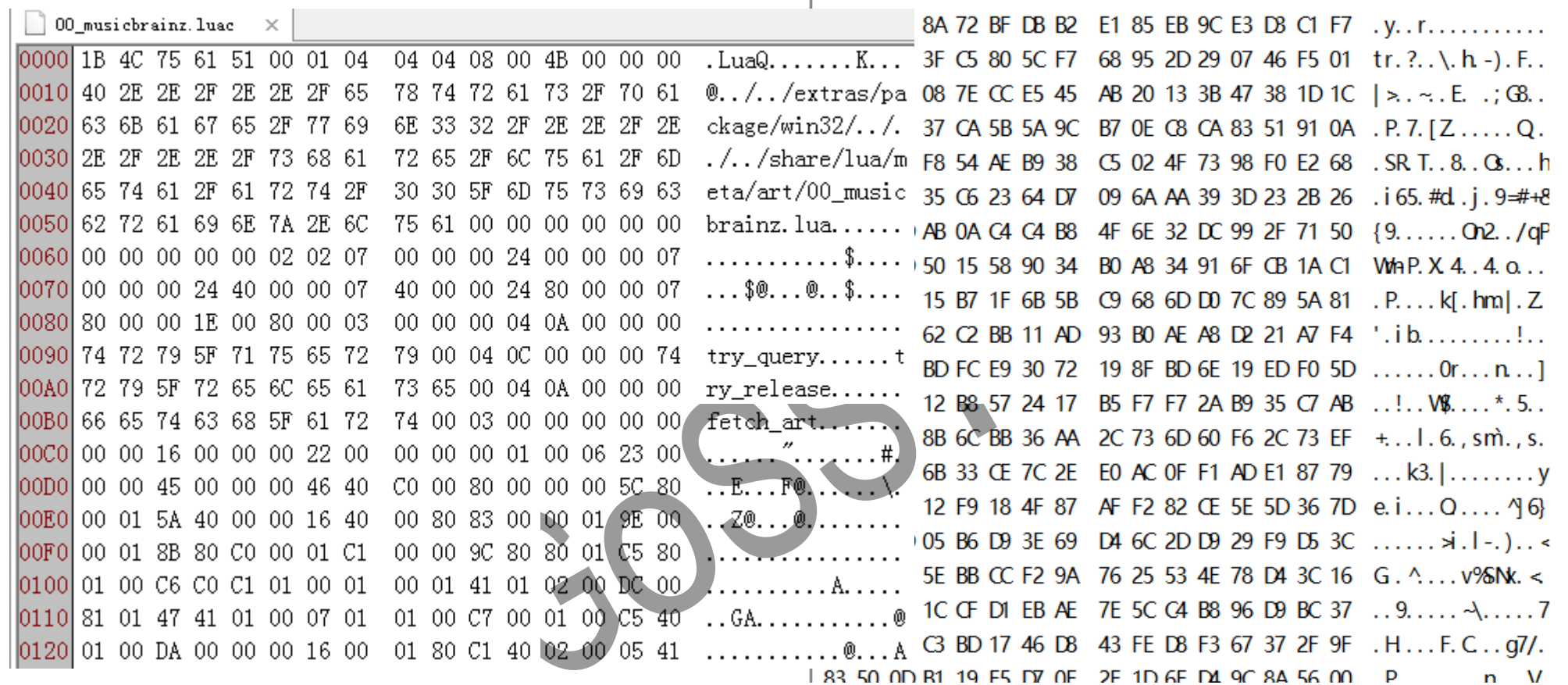
1 141	1 146
844	252
2 890	2 895
1 554	1 559
1 047	1 052
4 156	4 161
2 197	2 202
922	927
40 624	40 639
1 424	1 429
6 408	2 009
5 387	5 153

JOSS

MMO类游戏安全分析技术-程序分析-Cocos2d-x

逆向还原加载流程

- 判断是否存在对Lua脚本的处理



The screenshot displays a debugger window with two panes. The top pane shows a memory dump with hexadecimal values and their corresponding ASCII representations. The bottom pane shows the disassembly of the code, with instructions and their operands. The memory dump includes addresses from 0000 to 0120, and the disassembly shows instructions like .LuaQ, @../.., ckage/win32/.., ../share/lua/m, eta/art/00_music, brainz.lua,, \$@...@..\$, try_query, ry_release, fetch_art,, E...F@..., Z@...@...,, A...,, and @...A.

```
0000 1B 4C 75 61 51 00 01 04 04 04 08 00 4B 00 00 00 .LuaQ.....K... 3F C5 80 5C F7 68 95 2D 29 07 46 F5 01 tr.?.\..h-).F..
0010 40 2E 2E 2F 2E 2E 2F 65 78 74 72 61 73 2F 70 61 @../..../extras/pa 08 7E CC E5 45 AB 20 13 3B 47 38 1D 1C |>..~..E. ;G..
0020 63 6B 61 67 65 2F 77 69 6E 33 32 2F 2E 2E 2F 2E ckage/win32/../. 37 CA 5B 5A 9C B7 0E C8 CA 83 51 91 0A .P.7.[Z.....Q.
0030 2E 2F 2E 2E 2F 73 68 61 72 65 2F 6C 75 61 2F 6D ../share/lua/m F8 54 AE B9 38 C5 02 4F 73 98 F0 E2 68 .SR T..8..G...h
0040 65 74 61 2F 61 72 74 2F 30 30 5F 6D 75 73 69 63 eta/art/00_music 35 C6 23 64 D7 09 6A AA 39 3D 23 2B 26 .i 65. #d.j. 9=#+8
0050 62 72 61 69 6E 7A 2E 6C 75 61 00 00 00 00 00 00 brainz.lua..... AB 0A C4 C4 B8 4F 6E 32 DC 99 2F 71 50 {9.....On2../qP
0060 00 00 00 00 00 02 02 07 00 00 00 24 00 00 00 07 .....$.... 50 15 58 90 34 B0 A8 34 91 6F CB 1A C1 WhP.X 4..4. a..
0070 00 00 00 24 40 00 00 07 40 00 00 24 80 00 00 07 ...$@...@..$.... 15 B7 1F 6B 5B C9 68 6D D0 7C 89 5A 81 .P....k[.hm].Z
0080 80 00 00 1E 00 80 00 03 00 00 00 04 0A 00 00 00 ..... 62 C2 BB 11 AD 93 B0 AE A8 D2 21 A7 F4 '.i b.....!..
0090 74 72 79 5F 71 75 65 72 79 00 04 0C 00 00 00 74 try_query.....t BD FC E9 30 72 19 8F BD 6E 19 ED F0 5D .....Or...n..]
00A0 72 79 5F 72 65 6C 65 61 73 65 00 04 0A 00 00 00 ry_release..... 12 B8 57 24 17 B5 F7 F7 2A B9 35 C7 AB ...!..V$....*.5..
00B0 66 65 74 63 68 5F 61 72 74 00 03 00 00 00 00 00 fetch_art..... 8B 6C BB 36 AA 2C 73 6D 60 F6 2C 73 EF +...l.6,sm.,s.
00C0 00 00 16 00 00 00 22 00 00 00 00 01 00 06 23 00 .....#.. 6B 33 CE 7C 2E E0 AC 0F F1 AD E1 87 79 ...k3.|.....y
00D0 00 00 45 00 00 00 46 40 C0 00 80 00 00 00 5C 80 ..E...F@.....\.. 12 F9 18 4F 87 AF F2 82 CE 5E 5D 36 7D e.i...O....^6
00E0 00 01 5A 40 00 00 16 40 00 80 83 00 00 01 9E 00 ..Z@...@..... 05 B6 D9 3E 69 D4 6C 2D D9 29 F9 D5 3C .....>.l-.)..<
00F0 00 01 8B 80 C0 00 01 C1 00 00 9C 80 80 01 C5 80 ..... 5E BB CC F2 9A 76 25 53 4E 78 D4 3C 16 G.^...v%Nk.<
0100 01 00 C6 C0 C1 01 00 01 00 01 41 01 02 00 DC 00 .....A..... 1C CF D1 EB AE 7E 5C C4 B8 96 D9 BC 37 ..9.....\.....7
0110 81 01 47 41 01 00 07 01 01 00 C7 00 01 00 C5 40 ..GA.....@.. C3 BD 17 46 D8 43 FE D8 F3 67 37 2F 9F .H...F.C...g7/.
0120 01 00 DA 00 00 00 16 00 01 80 C1 40 02 00 05 41 .....@...A.. 18 3 50 0D B1 19 F5 17 0F 2F 1D 6F 1A 9C 8A 56 00 P n V
```

MMO类游戏安全分析技术-程序分析-Cocos2d-x

逆向还原加载流程

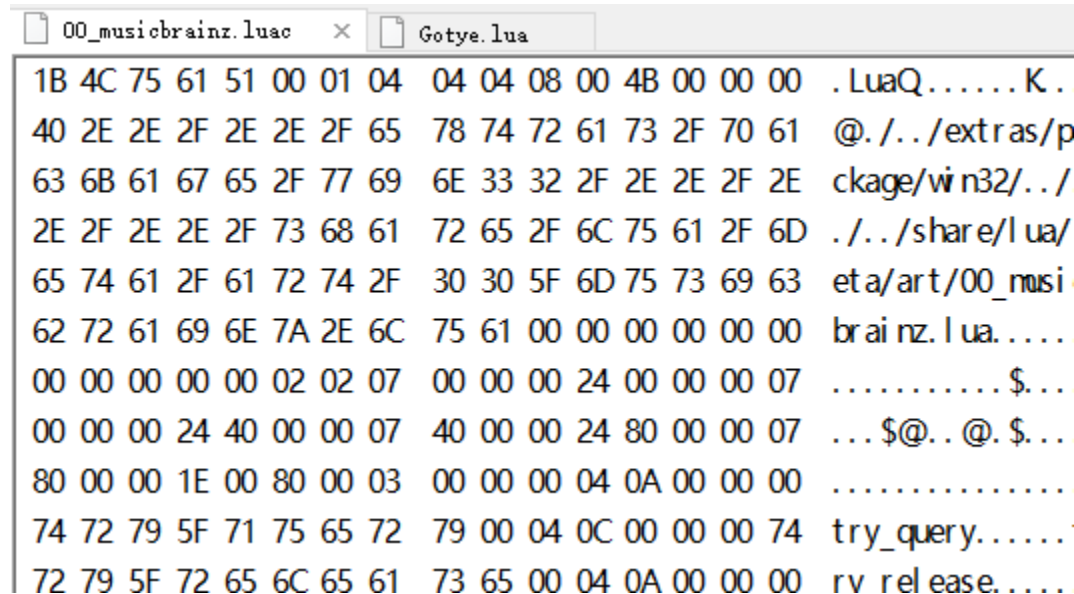
- 分析Lua脚本的加密/处理算法
- 还原Lua脚本

```
{
    v10 = *(_DWORD *)(v8 + 12);
    v9 = 0;
}
else
{
    v10 = *(_DWORD *)(v8 + 8);
    v9 = 1;
}
n = v8;
v8 = v10;
}
if ( v9 )
{
    if ( n == *(_DWORD *)(v6 + 36) )
        goto LABEL_35;
    v11 = sub_878D9C(n);
}
else
{
    v11 = n;
}
if ( sub_8B29CC(v11 + 16, &ptr) >= 0 )
{
    LABEL_13:
    sub_8B2DD0(&ptr);
    goto LABEL_14;
}
}
LABEL_35:
std::_Rb_tree<std::string,std::pair<std::string const,unsigned int>,std::
    v6 + 24,
    0,
    n,
    &ptr);
goto LABEL_13;
}
LABEL_14:
if ( strncmp(v5, "L:grxx", 6u) )
{
    if ( !strcmp(v5, "__sign_of_g18_enc__", 0x13u) )
    {
        memset(v5, 0, 0x13u);
        lrc4::lrc4((int)&v34);
        v32 = v5 + 19;
        v33 = v27 - 19;
        v24 = lua_load(v28, sub_312A88, &v32, a5);
    }
}
```

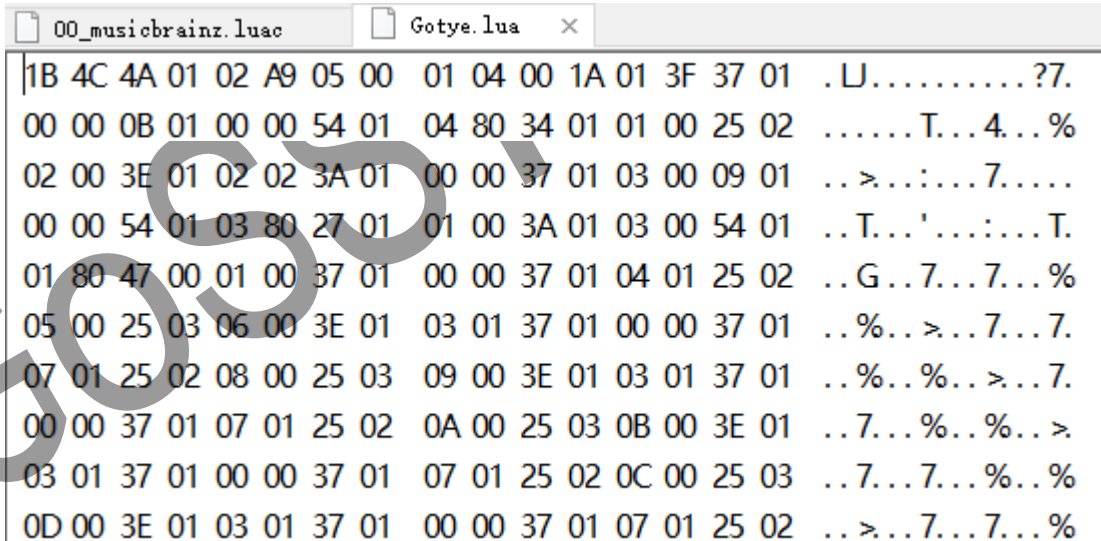
MMO类游戏安全分析技术-程序分析-Cocos2d-x

分析Lua脚本

- 判断Lua脚本的版本（不同版本的Lua脚本，magic number不同）
- 判断是否是luajit
- 寻找合适的反编译工具



```
00_musicbrainz.lua x Gotye.lua
1B 4C 75 61 51 00 01 04 04 04 08 00 4B 00 00 00 . LuaQ..... K.
40 2E 2E 2F 2E 2E 2F 65 78 74 72 61 73 2F 70 61 @./../extras/p
63 6B 61 67 65 2F 77 69 6E 33 32 2F 2E 2E 2F 2E ckage/win32/...
2E 2F 2E 2E 2F 73 68 61 72 65 2F 6C 75 61 2F 6D ./../share/lua/
65 74 61 2F 61 72 74 2F 30 30 5F 6D 75 73 69 63 eta/art/00_musi
62 72 61 69 6E 7A 2E 6C 75 61 00 00 00 00 00 00 brai nz. lua.....
00 00 00 00 00 02 02 07 00 00 00 24 00 00 00 07 ..... $...
00 00 00 24 40 00 00 07 40 00 00 24 80 00 00 07 ... $@.. @. $...
80 00 00 1E 00 80 00 03 00 00 00 04 0A 00 00 00 .....
74 72 79 5F 71 75 65 72 79 00 04 0C 00 00 00 74 try_query.....
72 79 5F 72 65 6C 65 61 73 65 00 04 0A 00 00 00 rv release.....
```



```
00_musicbrainz.lua x Gotye.lua x
1B 4C 4A 01 02 A9 05 00 01 04 00 1A 01 3F 37 01 .LJ.....?7.
00 00 0B 01 00 00 54 01 04 80 34 01 01 00 25 02 .... T...4...%
02 00 3E 01 02 02 3A 01 00 00 37 01 03 00 09 01 ..>...7....
00 00 54 01 03 80 27 01 01 00 3A 01 03 00 54 01 ..T...'...:....T.
01 80 47 00 01 00 37 01 00 00 37 01 04 01 25 02 ..G...7...7...%
05 00 25 03 06 00 3E 01 03 01 37 01 00 00 37 01 ..%..>...7...7.
07 01 25 02 08 00 25 03 09 00 3E 01 03 01 37 01 ..%..%..>...7.
00 00 37 01 07 01 25 02 0A 00 25 03 0B 00 3E 01 ..7...%..%..>
03 01 37 01 00 00 37 01 07 01 25 02 0C 00 25 03 ..7...7...%..%
0D 00 3E 01 03 01 37 01 00 00 37 01 07 01 25 02 ..>...7...7...%
```

MMO类游戏安全分析技术-程序分析-Cocos2d-x

分析Lua脚本

- luadec、unluac等等
- luajit目前没有很好用的反编译工具，可以使用luajit-decomp、ljd等工具协助分析

```
00_musicbrainz.luac  Gotye.lua  Gotye.asm  ×  localization.en.lua  Ilgnoth.lua
-- BYTECODE -- Gotye.lua:0-0↓
0001  TGETS   1  0  0  ; "gotyeapi"↓
0002  ISNEP   1  0↓
0003  JMP      1 => 0008↓
0004  GGET     1  1    ; "require"↓
0005  KSTR     2  2    ; "libgotyeapi"↓
0006  CALL     1  2  2↓
0007  TSETS    1  0  0  ; "gotyeapi"↓
0008 => TGETS    1  0  3  ; "islnited"↓
0009  ISNEN    1  0    ; 0↓
0010  JMP      1 => 0014↓
0011  KSHORT   1  1↓
0012  TSETS    1  0  2  ; "islnited"↓
```

```
00_musicbrainz.luac  Gotye.lua  Gotye.asm  localization.en.lua
elseif spellId == 210099 then--Ooze Fixate↓
    warnFixate:CombinedShow(1, args.destName)↓
    if args:IsPlayer() then↓
        specWarnFixate:Show(eyeName)↓
        voiceFixate:Play("targetyou")↓
    end↓
    if not addsTable[args.sourceGUID] then↓
        addsTable[args.sourceGUID] = true↓
        self.vb.lchorCount = self.vb.lchorCount + 1↓
    end↓
elseif spellId == 210094 then↓
    setRaidUnitId(args.destName)↓
end then↓
    = args.amount or 1↓
    2 then↓
```

分析Lua脚本

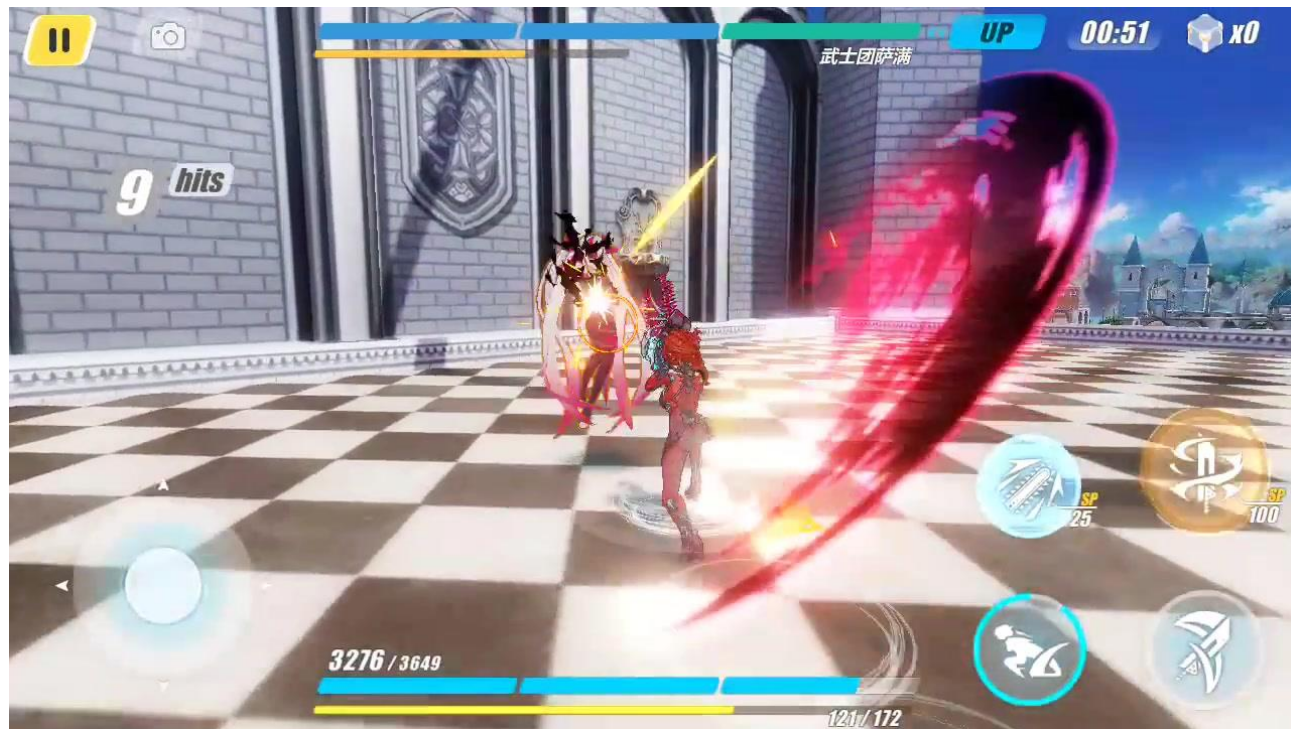
- 考验分析技术的时候到了
- 脚本语言相对容易分析
- 选择合适的字符串、资源文件帮助定位关键逻辑
- 理解逻辑、发现问题

JOSS

MMO类游戏安全分析技术-程序分析-Unity3D

Unity3D引擎特点

- 受限开源
- 全面、强大、专业的综合性引擎，不止应用在游戏开发领域
- 目前世界范围内移动平台的主流游戏引擎
- 支持C#、Javascript、Lua等多种脚本语言，以C#为主



MMO类游戏安全分析技术-程序分析-Unity3D

分析C#脚本

- 由于Unity3D的受限开源特性，导致绝大部分基于Unity3D开发的游戏具有较为一致的代码结构和文件结构。
- 名为libunity.so的native lib是引擎的主体部分

名称	大小	压缩后大小	作
libunity.so	23 358 773	8 732 878	2
liblua.so	144 688	90 550	2
libmono.so	3 758 452	1 100 635	2
libmain.so	45 104	19 442	2
libhsoda.so	34 408	16 826	2
libBugly.so	140 808	71 100	2
libAVProLocal.so	13 712	6 501	2
libAkSoundEngine.so	2 350 216	984 313	2

MMO类游戏安全分析技术-程序分析-Unity3D

分析逻辑脚本

- 一般来说，基于Unity3D开发的游戏所使用的逻辑脚本会有三类呈现方式：
 - C#&JS
 - Lua
 - IL2CPP



IL2CPP

IL2CPP is a Unity-developed scripting back-end which you can use to build your project using IL2CPP, Unity converts IL code (sometimes called Intermediate Language code), before creating a native binary file (.exe, apk, .xap, for iOS), for security, and platform compatibility of your Unity projects.

MMO类游戏安全分析技术-程序分析-Unity3D

分析逻辑脚本-C#&JS

- 作为最常见的类型，使用C#或者JS作为脚本语言的Unity3D游戏，其C#相关代码位于assets/bin/Data/Managed/目录下

```
izhuer@xiaoZtongxuedeMacBook-Air ~/Desktop/report/jam-city/assets/bin/Data/Managed ls -lah
total 7704
drwxr-xr-x  19 izhuer  staff   646B Jul  9 13:06 .
drwxr-xr-x 148 izhuer  staff   4.9K Jul  9 13:06 ..
-rw-r--r--   1 izhuer  staff    33K Jul  9 13:06 ApexAI.dll
-rw-r--r--   1 izhuer  staff    16K Jul  9 13:06 ApexSerialization.dll
-rw-r--r--   1 izhuer  staff    37K Jul  9 13:06 ApexShared.dll
-rw-r--r--   1 izhuer  staff   117K Jul  9 13:06 Assembly-CSharp-firstpass.dll
-rw-r--r--   1 izhuer  staff   783K Jul  9 13:06 Assembly-CSharp.dll
-rw-r--r--   1 izhuer  staff    45K Jul  9 13:06 Assembly-UnityScript-firstpass.dll
-rw-r--r--   1 izhuer  staff    6.5K Jul  9 13:06 Assembly-UnityScript.dll
-rw-r--r--   1 izhuer  staff    20K Jul  9 13:06 Boo.Lang.dll
-rw-r--r--   1 izhuer  staff    40K Jul  9 13:06 P31RestKit.dll
-rw-r--r--   1 izhuer  staff    36K Jul  9 13:06 System.Core.dll
-rw-r--r--   1 izhuer  staff   301K Jul  9 13:06 System.Xml.dll
-rw-r--r--   1 izhuer  staff   102K Jul  9 13:06 System.dll
-rw-r--r--   1 izhuer  staff    38K Jul  9 13:06 UnityEngine.Analytics.dll
-rw-r--r--   1 izhuer  staff   192K Jul  9 13:06 UnityEngine.Networking.dll
-rw-r--r--   1 izhuer  staff   201K Jul  9 13:06 UnityEngine.UI.dll
-rw-r--r--   1 izhuer  staff   454K Jul  9 13:06 UnityEngine.dll
-rw-r--r--   1 izhuer  staff   1.4M Jul  9 13:06 mscorlib.dll
```

MMO类游戏安全分析技术-程序分析-Unity3D

分析逻辑脚本-C#&JS

- Assembly-CSharp.dll 是C#脚本编译以后生成的文件
- Assembly-UnityScript.dll 是Javascript 脚本编译以后生成的文件

```
izhuer@xiaoZtongxuedeMacBook-Air ~/Desktop/report/jam-city/assets/bin/Data/Managed ls -lah
total 7704
drwxr-xr-x  19 izhuer  staff   646B Jul  9 13:06 .
drwxr-xr-x 148 izhuer  staff   4.9K Jul  9 13:06 ..
-rw-r--r--   1 izhuer  staff    33K Jul  9 13:06 ApexAI.dll
-rw-r--r--   1 izhuer  staff    16K Jul  9 13:06 ApexSerialization.dll
-rw-r--r--   1 izhuer  staff    37K Jul  9 13:06 ApexShared.dll
-rw-r--r--   1 izhuer  staff   117K Jul  9 13:06 Assembly-CSharp-firstpass.dll
-rw-r--r--   1 izhuer  staff   783K Jul  9 13:06 Assembly-CSharp.dll
-rw-r--r--   1 izhuer  staff    45K Jul  9 13:06 Assembly-UnityScript-firstpass.dll
-rw-r--r--   1 izhuer  staff    6.5K Jul  9 13:06 Assembly-UnityScript.dll
-rw-r--r--   1 izhuer  staff    20K Jul  9 13:06 Boo.Lang.dll
-rw-r--r--   1 izhuer  staff    40K Jul  9 13:06 P31RestKit.dll
-rw-r--r--   1 izhuer  staff    36K Jul  9 13:06 System.Core.dll
-rw-r--r--   1 izhuer  staff   301K Jul  9 13:06 System.Xml.dll
-rw-r--r--   1 izhuer  staff   102K Jul  9 13:06 System.dll
-rw-r--r--   1 izhuer  staff    38K Jul  9 13:06 UnityEngine.Analytics.dll
-rw-r--r--   1 izhuer  staff   192K Jul  9 13:06 UnityEngine.Networking.dll
-rw-r--r--   1 izhuer  staff   201K Jul  9 13:06 UnityEngine.UI.dll
-rw-r--r--   1 izhuer  staff   454K Jul  9 13:06 UnityEngine.dll
-rw-r--r--   1 izhuer  staff   1.4M Jul  9 13:06 mscorlib.dll
```

分析逻辑脚本-C#&JS

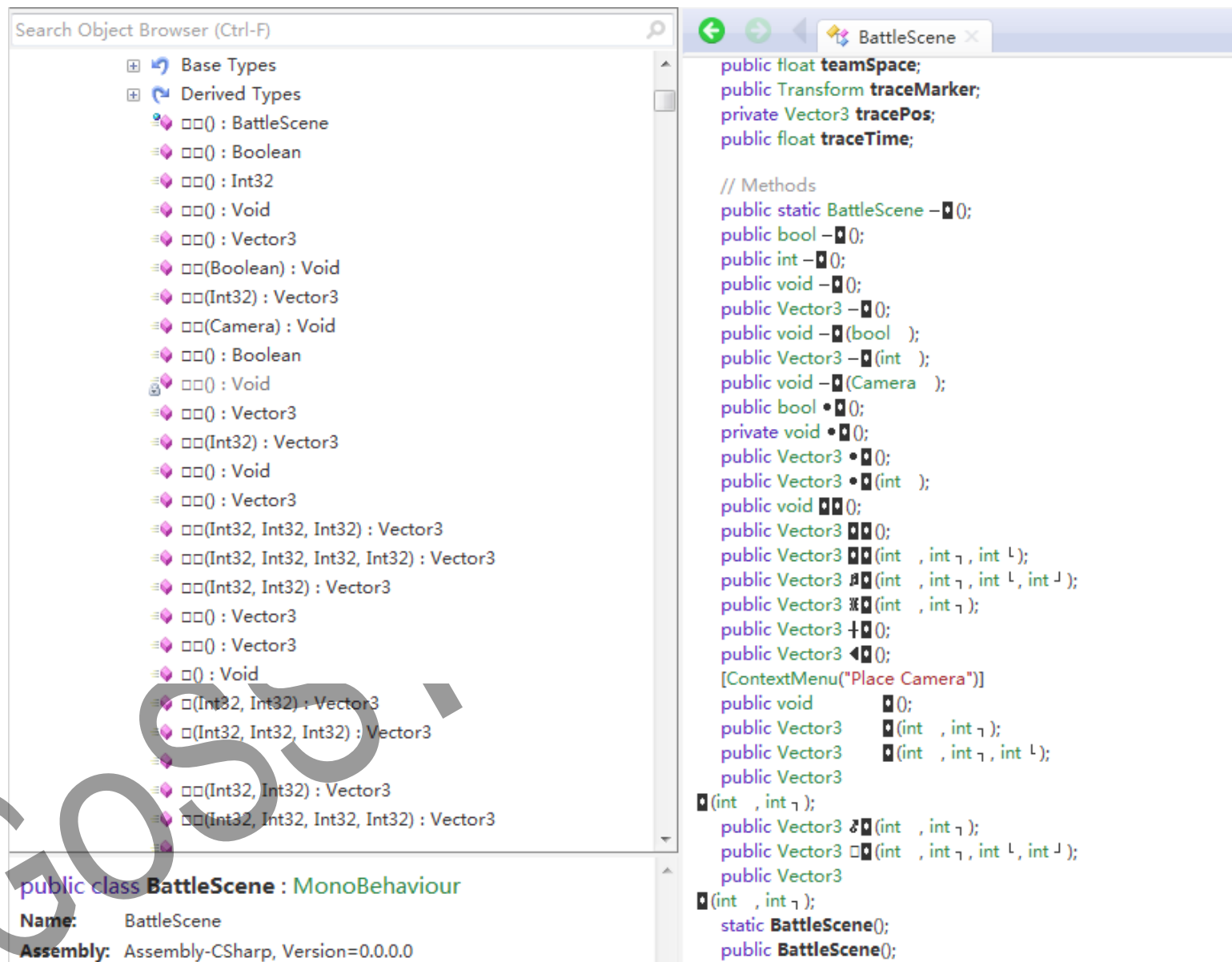
- 对于dll文件的分析，有很多成熟的工具可以使用
 - ILSpy（传统的C#反编译工具，提供可阅读的C#代码）
 - .NET Reflector（比ILSpy好用一些，能够部分修补被篡改过的文件头）
 - Reflexil（提供修改并重编译IL功能的插件）
 - ilasm & ildasm（il的逆向和重编译工具）

JOSS

MMO类游戏安全分析技术-程序分析-Unity3D

分析逻辑脚本-C#&JS

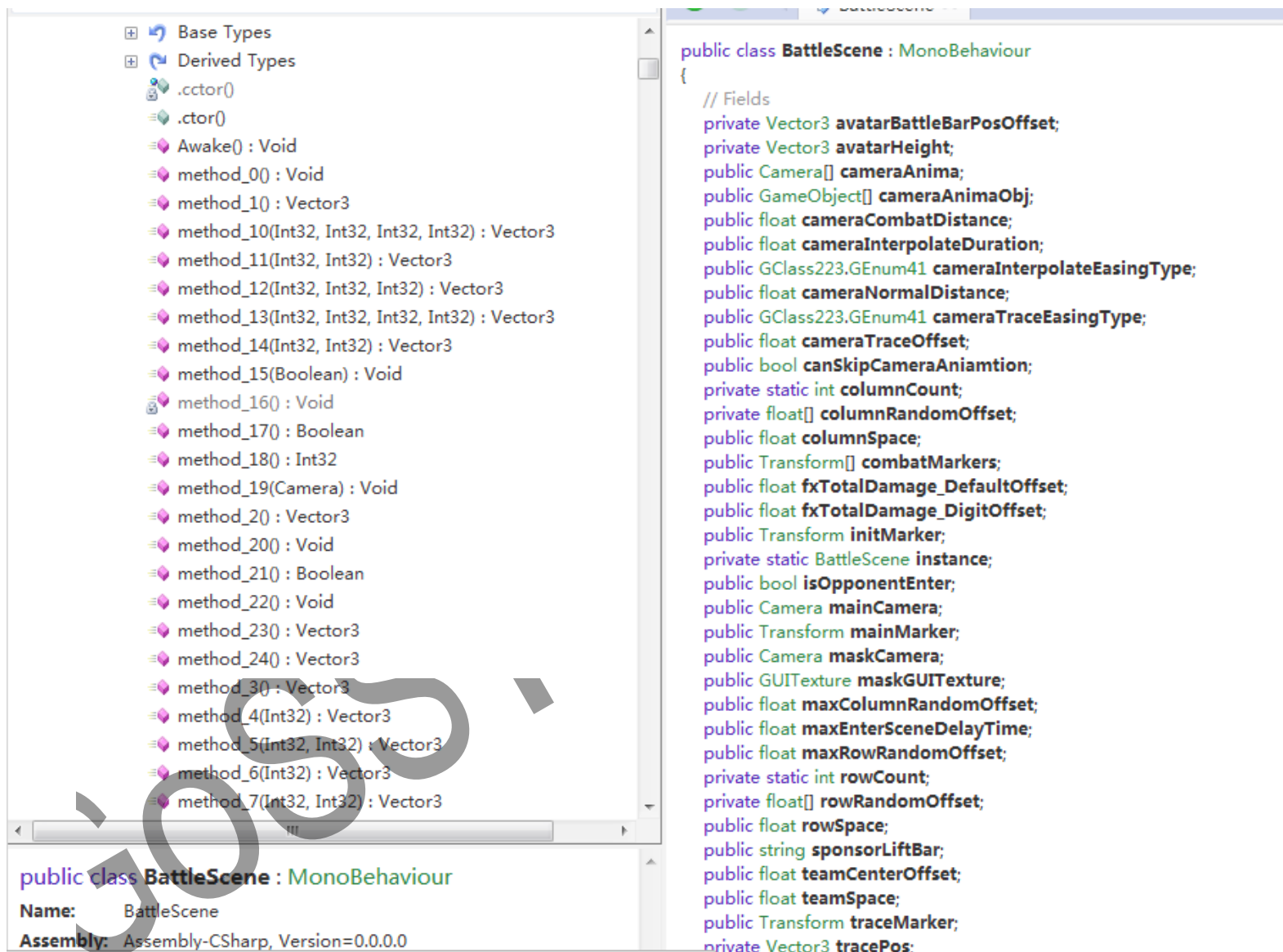
- 一般来说，不同于Cocos2D，Unity3D所使用的脚本文件是没有经过加壳的
- 但是，会有开发者对脚本进行一定程度的混淆



MMO类游戏安全分析技术-程序分析-Unity3D

分析逻辑脚本-C#&JS

- 需要先去混淆
 - de4dot





MMO类游戏安全分析技术-程序分析-Unity3D

分析逻辑脚本-Lua

- Unity3D中的Lua脚本一般用来实现热更新，常见的有slua，ulua,nlua等插件
- 分析方法和Cocos2D类似，解释器为liblua.so

1779537483067b945b3b43523142e18b	120 984	78 699
2347687963408b24f91c914d931708e4	4 548	403
2443045232380b54aba6e7a74190d6ae	4 352	324
3665490007517dc4fae79c88c23c5f43	4 548	382
04893642542345e45b6e91564da1ab4b	8 896	864
14925163514264d49abad95231376dc6	5 156	628
15763197700489a4da3ab1a696a5d463	8 600	841
32051127790583c4dbe85732ebf24a45	7 216	736
32975752089597b42a9f5bf172203d53	6 044	433
34010849856410a49b96c82fc4f496e1	135 288	18 053
0212939669387034ba0963f777f166f7	51 156	7 724
0426307913722594a81f1479c783af63	4 800	446
1404880760104684b879b021e6c10c2f	72 700	17 407
2454892691637744babaa4ebb1965ebe	4 372	218
3212775055298854a8f4e533d6d0152a	4 386	299
3467490907202064e83f176ecdef7c21	4 386	303
14187189481368348be677a72c4ee74a	4 908	528
054286444898676408fd7f1bac805b	17 760	1 234
204839182362847478ebb7f0a8b84dd9	17 916	3 624
38946799304117647875bf3b85b62aee	279 868	210 589



MMO类游戏安全分析技术-程序分析-Unity3D

分析逻辑脚本-IL2CPP

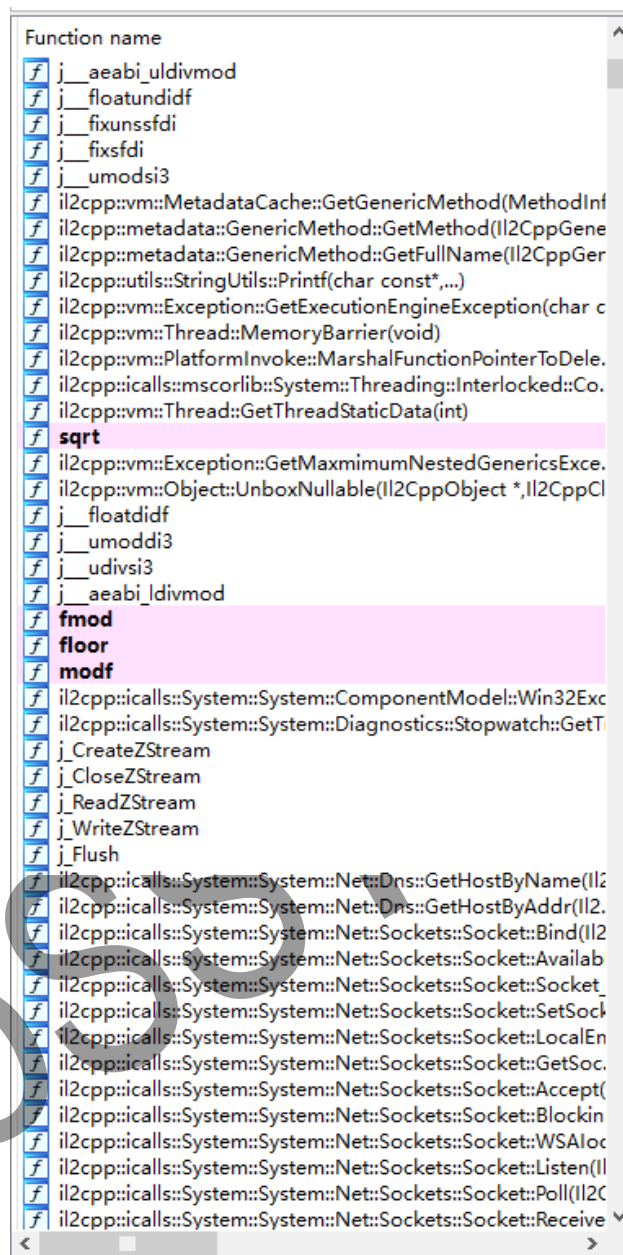
- 将C#编译为native code的技术
- C# -> IL -> CPP -> native code
- 编译后的代码为libil2cpp.so
- 一定程度上提升了安全性

libunity.so	19 090 081
liblua.so	263 664
libmain.so	19 804
libil2cpp.so	33 540 644
libhsoda.so	38 404
libAVProLocal.so	9 512
libAkSoundEngine.so	2 894 516

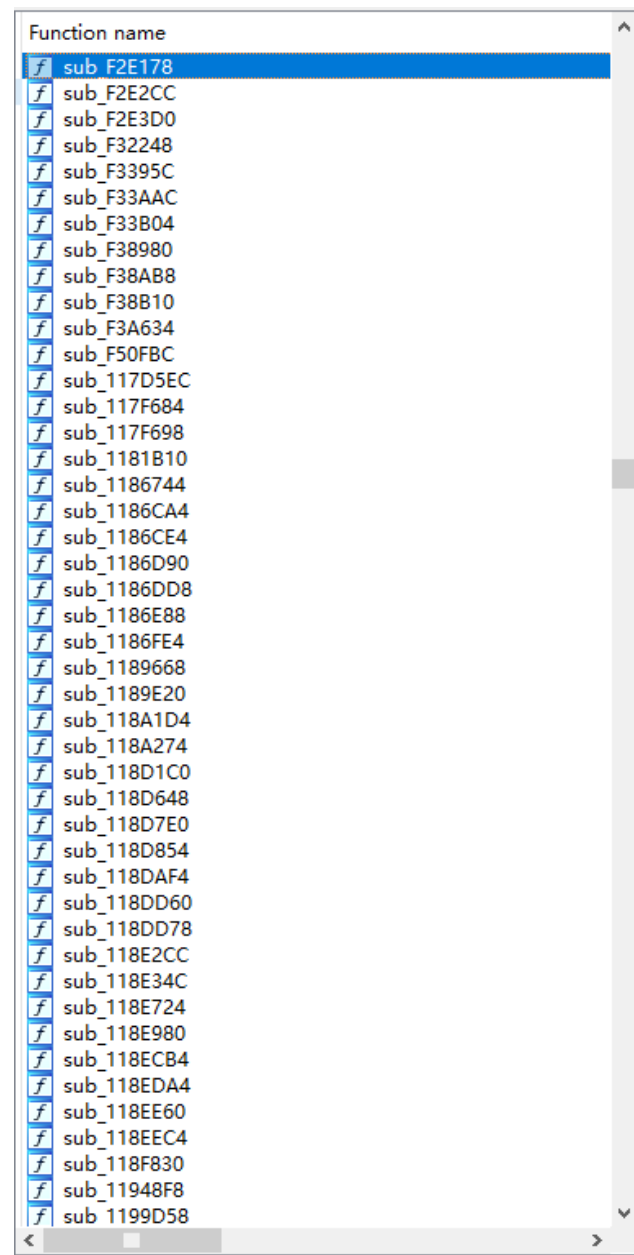
MMO类游戏安全分析技术-程序分析-Unity3D

分析逻辑脚本-IL2CPP

- 大部分的符号信息被隐藏了
- Il2cpp的代码在使用的时候存在一定的模式
 - 符号信息保存在
assets\bin\Data\Managed\Metadata\global-metadata.dat中
 - 使用il2cppdumper工具帮助还原符号信息



```
Function name
j__aeabi_uldivmod
j__floatundidf
j__fixunsfidi
j__fixsfdi
j__umodsi3
il2cpp::vm::MetadataCache::GetGenericMethod(MethodInfo)
il2cpp::metadata::GenericMethod::GetMethod(Il2CppType*)
il2cpp::metadata::GenericMethod::GetFullName(Il2CppType*)
il2cpp::utils::StringUtils::Printf(char const*,...)
il2cpp::vm::Exception::GetExecutionEngineException(char const*)
il2cpp::vm::Thread::MemoryBarrier(void)
il2cpp::vm::PlatformInvoke::MarshalFunctionPointerToDelegate
il2cpp::icalls::mscorlib::System::Threading::Interlocked::CompareExchange
il2cpp::vm::Thread::GetThreadStaticData(int)
sqrt
il2cpp::vm::Exception::GetMaximumNestedGenericsException
il2cpp::vm::Object::UnboxNullable(Il2CppObject*, Il2CppType*)
j__floatdidf
j__umoddi3
j__udivsi3
j__aeabi_ldivmod
fmod
floor
modf
il2cpp::icalls::System::System::ComponentModel::Win32Exception
il2cpp::icalls::System::System::Diagnostics::Stopwatch::GetTime
j__CreateZStream
j__CloseZStream
j__ReadZStream
j__WriteZStream
j__Flush
il2cpp::icalls::System::System::Net::Dns::GetHostByName(Il2CppObject*)
il2cpp::icalls::System::System::Net::Dns::GetHostByAddress(Il2CppObject*)
il2cpp::icalls::System::System::Net::Sockets::Socket::Bind(Il2CppObject*)
il2cpp::icalls::System::System::Net::Sockets::Socket::Available
il2cpp::icalls::System::System::Net::Sockets::Socket::Socket
il2cpp::icalls::System::System::Net::Sockets::Socket::SetSocketOptions
il2cpp::icalls::System::System::Net::Sockets::Socket::LocalEndPoint
il2cpp::icalls::System::System::Net::Sockets::Socket::GetSocketOptions
il2cpp::icalls::System::System::Net::Sockets::Socket::Accept
il2cpp::icalls::System::System::Net::Sockets::Socket::BlockIn
il2cpp::icalls::System::System::Net::Sockets::Socket::WSALoc
il2cpp::icalls::System::System::Net::Sockets::Socket::Listen(Il2CppObject*)
il2cpp::icalls::System::System::Net::Sockets::Socket::Poll(Il2CppObject*)
il2cpp::icalls::System::System::Net::Sockets::Socket::Receive
```



```
Function name
sub_F2E178
sub_F2E2CC
sub_F2E3D0
sub_F32248
sub_F3395C
sub_F33AAC
sub_F33B04
sub_F38980
sub_F38A88
sub_F38B10
sub_F3A634
sub_F50FBC
sub_117D5EC
sub_117F684
sub_117F698
sub_1181B10
sub_1186744
sub_1186CA4
sub_1186CE4
sub_1186D90
sub_1186DD8
sub_1186E88
sub_1186FE4
sub_1189668
sub_1189E20
sub_118A1D4
sub_118A274
sub_118D1C0
sub_118D648
sub_118D7E0
sub_118D854
sub_118DAF4
sub_118DD60
sub_118DD78
sub_118E2CC
sub_118E34C
sub_118E724
sub_118E980
sub_118ECB4
sub_118EDA4
sub_118EE60
sub_118EEC4
sub_118F830
sub_11948F8
sub_1199D58
```

分析逻辑脚本-IL2CPP

- 使用il2cppdumper工具帮助还原符号信息

- 定位注册函数

- 提取参数r0和r1
- 在使用il2cppdumper时输入r0和r1
- 完成修复
- 分析

```
il2cpp::vm::MetadataCache::Register(
```

MMO类游戏安全分析技术-程序分析-Unity3D

分析逻辑脚本-IL2CPP

- 使用il2cppdumper工具帮助还

原符号信息

- 定位注册函数
- 提取参数r0和r1
- 在使用il2cppdumper时输入r0和r1
- 完成修复
- 分析

```
; il2cpp::vm::MetadataCache::Register(IL2CppCodeRegistration const*, IL2CppMetadataRegistration const*, IL2CppCodeGenOptions const*)  
j__ZN6il2cpp2vm13MetadataCache8RegisterEPK22IL2CppCodeRegistrationPK26IL2CppMetadataRegistrationPK20IL2CppCodeGenOptions  
; CODE XREF: .text:018A7A18↓j
```

```
ADRL    R12, 0x1EC53F4  
LDR     PC, [R12, #(_ZN6il2cpp2vm13MetadataCache8RegisterEPK22IL2CppCodeRegistrationPK26IL2CppMetadataRegistrationPK20IL2CppCodeGenOptions)]
```

```
LDR     R1, [R1, R2] ; unk_1DD4190  
ADD     R0, R12, R2 ; unk_1DAB758  
ADD     R2, R3, R2 ; unk_1BC9A9C  
B       j__ZN6il2cpp2vm13MetadataCache8RegisterEPK22IL2CppCodeRegistrationPK26IL2CppMetadataRegistrationPK20IL2CppCodeGenOptions
```

分析逻辑脚本-IL2CPP

- 使用il2cppdumper工具帮助还原符号信息
 - 定位注册函数
 - 提取参数r0和r1
 - 在使用il2cppdumper时输入r0或r1
 - 完成修复
 - 分析

```
PS C:\Users\EverMars\Desktop> .\I12CppDumper.exe
Select Mode: 1. Manual 2.Auto
CodeRegistration : 1dab758
MetadataRegistration : 1dd4190
Dumping...
Done !
Press any key to exit...
PS C:\Users\EverMars\Desktop> _
```

MMO类游戏安全分析技术-程序分析-Unity3D

分析逻辑脚本-IL2CPP

- 使用il2cppdumper工具帮助还原符号信息
 - 定位注册函数
 - 提取参数r0和r1
 - 在使用il2cppdumper时输入r0或r1
 - 完成修复
 - 分析

```
private virtual ushort System.IConvertible.ToInt16(IFormatProvider provider); // 15165f8↓
private virtual uint System.IConvertible.ToInt32(IFormatProvider provider); // 15166a8↓
private virtual ulong System.IConvertible.ToInt64(IFormatProvider provider); // 1516758↓
public virtual int CompareTo(object value); // 151687c↓
public virtual bool Equals(object obj); // 1516938↓
public virtual int GetHashCode(); // 1516948↓
public virtual int CompareTo(int value); // 151696c↓
public virtual bool Equals(int obj); // 151699c↓
static bool ProcessTrailingWhitespace(bool tryParse, string s, int position, Exception exc); // 15157dc↓
static bool Parse(string s, bool tryParse, out int result, out Exception exc); // 15169b0↓
public static int Parse(string s, IFormatProvider provider); // 1516cf4↓
public static int Parse(string s, NumberStyles style); // 1516d18↓
static bool CheckStyle(NumberStyles style, bool tryParse, Exception exc); // 1516d34↓
static bool JumpOverWhite(int pos, string s, bool reportError, bool tryParse, Exception exc); // 1516e64↓
static void FindSign(int pos, string s, NumberFormatInfo nfi, bool foundSign, bool negative); // 1516fbc↓
static void FindCurrency(int pos, string s, NumberFormatInfo nfi, bool foundCurrency); // 1517140↓
static bool FindExponent(int pos, string s, int exponent, bool tryParse, Exception exc); // 15172a4↓
static bool FindOther(int pos, string s, string other); // 1517744↓
static bool ValidDigit(char e, bool allowHex); // 1517894↓
static Exception GetFormatException(); // 151573c↓
static bool Parse(string s, NumberStyles style, IFormatProvider fp, bool tryParse, out int result, out Exception exc); // 15179b0↓
public static int Parse(string s); // 1518ac4↓
public static int Parse(string s, NumberStyles style, IFormatProvider provider); // 1515a14↓
public static bool TryParse(string s, out int result); // 1518b10↓
public static bool TryParse(string s, NumberStyles style, IFormatProvider provider, out int result); // 1518b58↓
public virtual string ToString(); // 1518b9c↓
public virtual string ToString(IFormatProvider provider); // 1518c54↓
public string ToString(string format); // 1518d28↓
```

MMO类游戏安全分析技术-程序分析-Unity3D

分析逻辑脚本-IL2CPP

- 使用il2cppdumper工具帮助还原符号信息
 - 定位注册函数
 - 提取参数r0和r1
 - 在使用il2cppdumper时输入r0或r1
 - 完成修复
 - 分析

```
public void SetupLevelDamageStastics(); // 5b67e0↓
```

loc_5B67E0

```
STMFD
ADD
LDR
LDR
ADD
ADD
LDRB
CMP
BNE
LDR
LDR
ADD
LDR
LDR
BL
ADD
MOV
STRB
```

```
; DATA XREF: .data.rel.ro:01DFAB64↓
SP!, {R4,R5,R11,LR}
R11, SP, #8
R0, =(_GLOBAL_OFFSET_TABLE_ - 0x5B67F8)
R4, =0x3E38
R0, PC, R0 ; _GLOBAL_OFFSET_TABLE_
R0, R4, R0
R0, [R0,#(byte_1EC90B6 - 0x1EC9020)]
R0, #0
loc_5B6828
R0, =(_GLOBAL_OFFSET_TABLE_ - 0x5B6814)
R1, =0xFFFFCDB80
R5, PC, R0 ; _GLOBAL_OFFSET_TABLE_
R0, [R1,R5]
R0, [R0]
j__ZN6il2cpp2vm13MetadataCache24InitializeMethodMetadataEj ;
R0, R4, R5
R1, #1
R1, [R0,#(byte_1EC90B6 - 0x1EC9020)]
```

MMO类游戏安全分析技术-程序分析-自定义引擎案例

判断引擎类型

- 历史经验
- 搜索引擎
- 查看一些配置文件、符号信息
- 逆向Java层逻辑

```
public void onCreate(Bundle arg7) {
    int v5 = 128;
    int v3 = 8;
    super.onCreate(arg7);
    if(Channel.getInstance() != null) {
        Channel.getInstance().on_newIntent(this.getIntent());
    }

    if((this.getIntent().getFlags() & 4194304) != 0) {
        this.finish();
    }
    else {
        this.m_progress_dlg = new ProgressDialog(((Context)this));
        this.m_progress_dlg.setIndeterminate(false);
        this.m_progress_dlg.setCanceledOnTouchOutside(false);
        this.m_progress_dlg.setCancelable(false);
        this.m_progress_dlg.setProgressStyle(1);
        this.m_progress_dlg.setMax(100);
        this.m_progress_dlg.setTitle(this.getStringId("launcher_copy_data"));
        this.m_progress_dlg.setIcon(this.getDrawableId("ic_launcher"));
        this.m_progress_dlg.getWindow().setFlags(v3, v3);
        this.m_progress_dlg.getWindow().addFlags(131200);
        int v0 = Launcher.getCoreNumber();
        this.m_platform_config = new PlatformConfigParser(((Context)this));
        this.m_platform_config.addVariable("SDK_INT", Build.VERSION.SDK_INT);
        this.m_platform_config.addVariable("CORE_NUM", v0);
        this.m_platform_config.addVariable("MODEL", Build.MODEL);
        this.m_platform_config.addVariable("MANUFACTURER", Build.MANUFACTURER);
        Log.i(" ", "SDK_INT is " + Build.VERSION.SDK_INT);
        Log.i(" ", "CORE_NUM is " + v0);
        Log.i(" ", "MODEL is " + Build.MODEL);
        Log.i(" ", "MANUFACTURER is " + Build.MANUFACTURER);
        this.m_is_gl_loaded = false;
        this.m_view = new GLSurfaceView(((Context)this));
        if(Build.VERSION.SDK_INT >= 14) {
            if(Build.VERSION.SDK_INT >= 19) {
                this.m_view.setSystemUiVisibility(3846);
            }
            else {
                this.m_view.setSystemUiVisibility(1285);
            }
        }
    }
}
```


MMO类游戏安全分析技术-程序分析-自定义引擎

判断引擎类型

- 查看/proc/<pid>/maps
- 查看/proc/<pid>/fd/*
- 寻找关键文件

```
acez@debian-armel:~$ cat /proc/self/maps
00008000-00012000 r-xp 00000000 08:01 380      /bin/cat
00019000-0001a000 r-xp 00009000 08:01 380      /bin/cat
0001a000-0001b000 rwxp 0000a000 08:01 380      /bin/cat
01c86000-01ca7000 rwxp 00000000 00:00 0        [heap]
b6d07000-b6e7e000 r-xp 00000000 08:01 7366     /usr/lib/locale/locale-archive
b6e7e000-b6fa8000 r-xp 00000000 08:01 761      /lib/arm-linux-gnueabi/libc-2.13.so
b6fa8000-b6fb0000 --p 0012a000 08:01 761      /lib/arm-linux-gnueabi/libc-2.13.so
b6fb0000-b6fb2000 r-xp 0012a000 08:01 761      /lib/arm-linux-gnueabi/libc-2.13.so
b6fb2000-b6fb3000 rwxp 0012c000 08:01 761      /lib/arm-linux-gnueabi/libc-2.13.so
b6fb3000-b6fb6000 rwxp 00000000 00:00 0
b6fb6000-b6fd3000 r-xp 00000000 08:01 1381     /lib/arm-linux-gnueabi/ld-2.13.so
b6fd3000-b6fd5000 rwxp 00000000 00:00 0
b6fd9000-b6fda000 rwxp 00000000 00:00 0
b6fda000-b6fdb000 r-xp 0001c000 08:01 1381     /lib/arm-linux-gnueabi/ld-2.13.so
b6fdb000-b6fdc000 rwxp 0001d000 08:01 1381     /lib/arm-linux-gnueabi/ld-2.13.so
bed47000-bed68000 rw-p 00000000 00:00 0        [stack]
fffff000-fffff1000 r-xp 00000000 00:00 0        [vectors]
acez@debian-armel:~$
```

```
lr-x----- root    root    2015-11-19 11:26 11 -> pipe:[2618350]
l-wx----- root    root    2015-11-19 11:26 12 -> pipe:[2618350]
l-wx----- root    root    2015-11-19 11:26 13 -> /dev/cpuctl/tasks
l-wx----- root    root    2015-11-19 11:26 14 -> /dev/cpuctl/bg_non_interactive/tasks
lrwx----- root    root    2015-11-19 11:26 15 -> anon_inode:[eventpoll]
lr-x----- root    root    2015-11-19 11:26 16 -> /data/app/ma. .... .appinjection-1/base.apk
lr-x----- root    root    2015-11-19 11:26 17 -> pipe:[2619617]
lrwx----- root    root    2015-11-19 11:26 18 -> socket:[2619615]
l-wx----- root    root    2015-11-19 11:26 19 -> pipe:[2619617]
lrwx----- root    root    2015-11-19 11:26 2 -> /dev/null
```

MMO类游戏安全分析技术-程序分析-自定义引擎

分析逻辑脚本-变种python

- 定位脚本文件
 - assets/script.npk
- 分析加载脚本逻辑
- 解密脚本

```
IDA View-A Pseudocode-E Pseudocode-D Pseudocode-C Pseudocode-B Pseudocode-A
78 if ( v34 - 3 != &dword_2064214 )
79 {
80     v19 = (unsigned int *)(v34 - 1);
81     __dmb(0xFu);
82     do
83     {
84         v20 = __ldrex(v19);
85         while ( __strex(v20 - 1, v19) );
86         __dmb(0xFu);
87         if ( v20 <= 0 )
88             operator delete(v8);
89     }
90     sprintf((char *)(v1 + 1232), "%s/%s", v32, v1 + 972);
91     sprintf((char *)(v1 + 972), (const char *)(v1 + 1232));
92     *(_WORD *)(v1 + 1232 + strlen((const char *)(v1 + 1232))) = *(_WORD *)"/";
93     v9 = (const char *)((*int (__fastcall **)(int))(*(_DWORD *)v1 + 168))(v1);
94     strncpy(&dest, v9, 0x104u);
95     strncat(&dest, (const char *)(v1 + 972), 0x104u);
96     v10 = strlen(&dest);
97     *(_DWORD *)(&dest + v10) = 'kpn.';
98     *(&dest + v10 + 4) = a_npk[4];
99     if ( access(&dest, 0) != -1 )
100     {
101         v34 = &dword_2064220;
102         LODWORD(v11) = sub_928630((int)&v35, &dest, 0, 0, -1LL);
103         if ( v11 )
104             sub_92AAD4(&v34, &v35);
105         sub_90BBB8(0, "script.npk MD5:%s", v34);
106         v12 = v34 - 3;
107         if ( v34 - 3 != &dword_2064214 )
108         {
109             v27 = (unsigned int *)(v34 - 1);
110             __dmb(0xFu);
111             do
112             {
113                 v28 = __ldrex(v27);
114                 while ( __strex(v28 - 1, v27) );
115                 __dmb(0xFu);
116                 if ( v28 <= 0 )
117                     operator delete(v12);
118             }
119             v13 = *(_DWORD *)(v1 + 192);
120             v14 = (*int (__fastcall **)(int, int, int))dword_1840BE8;
121             v15 = dword_1840BE8;
122             *(_DWORD *)(&dest + v10 + 4) = dword_1840BE8;
123             v16 = (*v14)(v15, v13, v1 + 972);
124             v17 = (void *)(v32 - 12);
```

MMO类游戏安全分析技术-程序分析-自定义引擎

分析逻辑脚本-变种python

- 分析脚本
 - 使用了指令集置换技术修改了python引擎
 - 逆向python引擎
 - PyEval_EvalFrameEx, 找到修改过的opcode表
 - 对pyc进行bytecode分析, 找到opcode对应关系
- 重新编译开源的python反编译器
- 反编译pyc
- 分析

```
1134     v23 += 3;
1135     v39 = v38 | (v26 << 8);
1136 }
1137 LABEL_61:
1138     v17 = 16;
1139     v41 = 1;
1140     switch ( v37 )
1141     {
1142     case 3u:
1143         v146 = 0;
1144         v147 = (DWORD *) (v868 - 2);
1145         v870 = (int) (v868 - 2);
1146         v148 = *(int **) (v870 + 4);
1147         if ( v148 )
1148         {
1149             if ( v148 != &Py_NoneStruct )
1150             {
1151                 v146 = 0;
1152                 v866 = v23;
1153                 v149 = *(DWORD **) (v870 + 4);
1154                 ++v148;
1155                 goto LABEL_887;
1156             }
1157         }
1158     else
1159     {
1160         v146 = 1;
1161     }
1162     v866 = v23;
1163 LABEL_885:
1164     v472 = (DWORD *) PySys_GetObject("stdout");
1165     v149 = v472;
1166     if ( !v472 )
1167     {
1168         PyErr_SetString(PyExc_RuntimeError[0], "lo:
1169         v473 = 0;
1170         v20 = -1;
1171         goto LABEL_1272;
1172     }
1173     ++v472;
1174 LABEL_887:
1175     v473 = 0;
1176     if ( PyFile_SoftSpace(v149, 0) )
1177         v20 = PyFile_WriteString(" ", v149);
1178     if ( v20 )
1179         goto LABEL_1260;
```

MMO类游戏安全分析技术-程序分析-动态调试

动态调试

- 和调试普通的Android应用程序类似
- 以调试native lib为主
- Lua等脚本的调试较为困难
- 有些游戏会有反调试的措施

```
u33 = *(_BYTE *)(u234 - 36) | (*( _BYTE *)(u234 - 34) << 16) | (*( _BYTE *)(u234 - 35) << 8);
u35 = __ROR4__((u22 + u30) & (u22 ^ u17) ^ u17) + u24 - 176418897 + u6, 25);
u34 = u29 + u35;
*(_DWORD *)(u239 + 116) = u33 | (u32 << 24);
u36 = u33 | (u32 << 24);
u38 = __ROR4__((u29 + u35) & (u29 ^ u22) ^ u22) + u17 + u27 + 1200080426, 20);
u37 = u34 + u38;
u39 = (*( _BYTE *)(u234 - 30) << 16) | (*( _BYTE *)(u234 - 31) << 8) | (*( _BYTE *)(u234 - 32) | (*( _BYTE *)(u234 - 29) << 24);
*(_DWORD *)(u239 + 120) = u39;
u41 = __ROR4__((u34 ^ u29) & (u34 + u38) ^ u29) + u22 + u31 - 1473231341, 15);
u40 = u37 + u41;
u42 = u39;
u43 = u234;
u44 = (*( _BYTE *)(u234 - 26) << 16) | (*( _BYTE *)(u234 - 27) << 8) | (*( _BYTE *)(u234 - 28) | (*( _BYTE *)(u234 - 25) << 24);
u45 = (u37 + u41) ^ u37;
*(_DWORD *)(u239 + 124) = u44;
u47 = __ROR4__(u36 - 45705983 + u29 + ((u37 + u41) & (u37 ^ u34) ^ u34), 10);
u46 = u40 + u47;
u48 = u44;
u49 = (*( _BYTE *)(u234 - 22) << 16) | (*( _BYTE *)(u43 - 23) << 8) | (*( _BYTE *)(u43 - 24) | (*( _BYTE *)(u43 - 21) << 24);
*(_DWORD *)(u239 + 128) = u49;
u51 = __ROR4__(u34 + u39 + 1770035416 + ((u40 + u47) & u45 ^ u37), 25);
u50 = u46 + u51;
u52 = (u46 + u51) ^ u46;
u54 = __ROR4__(u37 + u48 - 1958414417 + ((u46 ^ u40) & (u46 + u51) ^ u40), 20);
u53 = u50 + u54;
u55 = (*( _BYTE *)(u43 - 18) << 16) | (*( _BYTE *)(u43 - 19) << 8) | (*( _BYTE *)(u234 - 20) | (*( _BYTE *)(u234 - 17) << 24);
u56 = ((u50 + u54) & u52 ^ u46) + u40 + u49 - 42063;
*(_DWORD *)(u239 + 132) = u55;
u57 = u55;
u58 = (u50 + u54) ^ u50;
u60 = __ROR4__(u56, 15);
```

00000052 md5:307

MMO类游戏安全分析技术-网络流量分析

网络流量分析

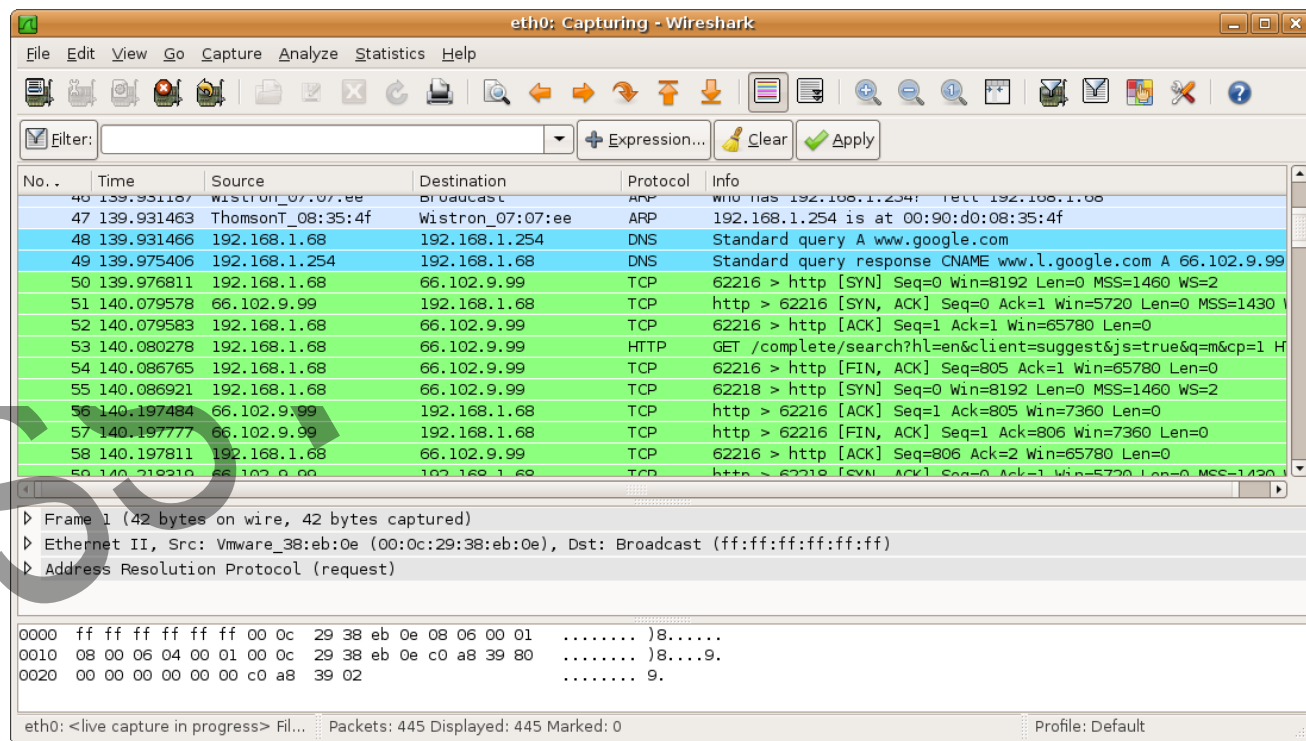
- 一般来说，MMO游戏中存在多种类型的网络通信
 - 第三方SDK、公告信息等，以HTTP/HTTPS为主，使用Burpsuite分析
 - 与游戏内容相关的流量，以socket为主



MMO类游戏安全分析技术-网络流量分析

网络流量分析

- 一般来说，MMO游戏中存在多种类型的网络通信
 - 第三方SDK、公告信息等，以HTTP/HTTPS为主，使用Burpsuite分析
 - 与游戏内容相关的流量，以socket为主，使用wireshark、风声等工具

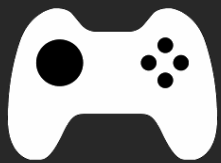


MMO类游戏安全分析技术-网络流量分析

网络流量分析

- 一般来说，MMO游戏中存在多种类型的网络通信
 - 第三方SDK、公告信息等，以HTTP/HTTPS为主，使用Burpsuite分析
 - 与游戏内容相关的流量，以socket为主，使用wireshark、风声等工具
 - Socket流量往往是经过加密/编码/变换的，对这类流量的分析依赖于对逻辑的分析

```
CDqjElkCAAAA9@@@3Q$@F4|\B      7进入蒲家 K4G当前278999<@
A3gNCDmjElkEAAAA'图%!)
19BIQYa<@
A3gNCDmjElkFAAAA'图%!)
19BIQY@
BHgNCDmjElkGAAAA'图%!)
1;fffff>@C@@K"@S6@[<ep?c(@k<@s<ep?{      桃木剑@      @@@      !)19A      @
BCANCCmIGVkhAAAA'图%'图%!)
1;@@C@@KS[cks{
      z布带@      @mz@      !)19A      @
BCANCC6IGVkhAAAA'图%'图%!)
1;@@C@@Kt1F@SS'}%6@[(Lc^/$@k
@s(L{      桃木剑@      @TkqAÇ% Dc{?UI5dn@      !)19A      @
BCANCC+IGVkBAAAA'图%'图%!)
1;@@C@@KS[cks{h?      棉护手@      @@Q}@@      !)19A      @
BCANCDmIGVKEAAAA'图%'图%!)
1;@@C@@KS[cks{@M9n      @头巾@      @3DvfM9tZ@      !)19A      @
BCANCDmIGVKGAAAA'图%'图%!)_س
1;@@C@@KS[cks{@@皮帽@      @zÑIF?@      !)19A      @
BCANCDmIGVKEAAAA'图%'图%!)
?@@C@@K!)19A{&@@1@<@烽牌@      @_=W=?F@
BCANCDmIGVkBAAAA'图%'图%!)
1;@C@K8!9@S8~M@[p={cM<@k""""""T@sp={ 轻罗扇@      @ `?h?~I@      !)19A)
```



MMO类游戏常见威胁

GOSS

MMO类游戏常见威胁

客户端侧

- 素材窃取
- 逻辑篡改
- 功能辅助



服务器侧

- 协议劫持



MMO类游戏常见威胁-素材窃取

攻击目标：窃取APK内/保存在磁盘上的图片、动画、音乐等素材。

影响：部分素材经过专业人士设计，价值不菲同时存在版权归属，被窃取后会给开发者带来经济上的损失；同时，可能对游戏未来的走向产生剧透。



MMO类游戏常见威胁-素材窃取

常见的技术

- 资源解包
- 对于处理过的资源文件，基于对游戏逻辑的逆向，定位并还原素材文件
- 动态调试，从内存中提取素材数据

JOSS

MMO类游戏常见威胁-逻辑篡改

攻击目标： 游戏运行时的本地运算逻辑

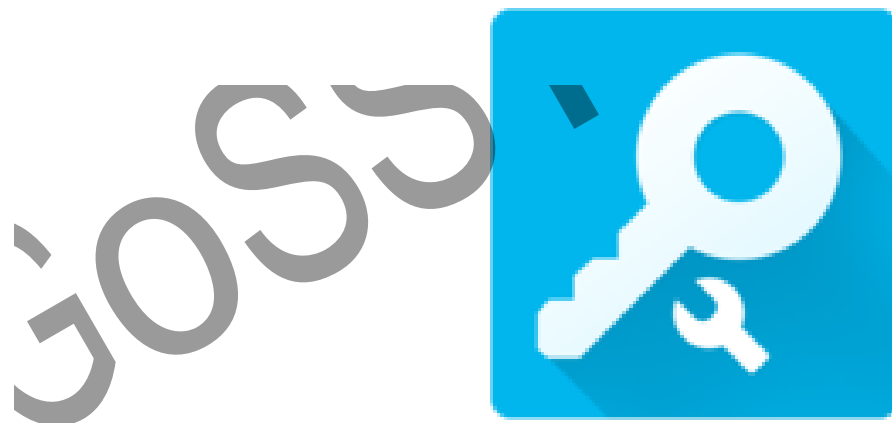
影响： 获取原本无法获得的道具、金钱、属性、功能，达到传统意义上的作弊，影响游戏平衡。



MMO类游戏常见威胁-逻辑篡改

常见技术

- 内存修改（八门神器、GameGuardian等）
 - 对目标进程进行ptrace，然后直接操作内存（需root，容易检测）
 - 操作/proc/<pid>/mem文件（需root，较难检测）
 - 定制运行环境，修改虚拟机、系统库或内核（很难检测）



MMO类游戏常见威胁-逻辑篡改

常见技术

- 代码修改（各类游戏的修改版）
 - 对游戏的脚本进行逆向分析，修改特定位置代码后对原文件进行替换/重打包（能够大规模的传播）
- 动态hook（需root，难以工具化）

NO.1	NO.2	NO.3
皇途霸业无限金币钻石内购破解版  71M / 中文 / v1.2.0 安卓版 ★★★★★ 皇途霸业无限金币钻石破解版是一款很有意思的非常酷炫的角色扮演类游戏，游戏的不仅画面制作的很精美，战斗场面也很刺激过瘾，剧 点击下载	皇途内购破解版  278.0M / 中文 / 1.1 安卓版 ★★★★★ 皇途内购破解版是一款战国题材的角色扮演类手游，玩家在皇途内购破解版中扮演的是王族后裔，为了完成一统中原的千秋大业，必须要 点击下载	地下城突击者内购破解版  276M / 中文 / v1.6.2 安卓版 ★★★★★ 地下城突击者内购破解版是一款Q版卡通风格的动作类手机游戏，西西本页为大家带来的是地下城突击者的内购破解版，玩家即刻下载登 点击下载
西游释厄传OL内购破解版  88.3M / 中文 / 1.1.2安卓版 ★★★★★ 西游释厄传OL内购破解版是一款西游题材的角色扮演类手游，玩家在西游释厄传OL内购破解版中要扮演一位盖世英雄，降妖除魔拯救三 点击下载	宠物小精灵变态破解版  46.2M / 中文 / 2017 最新变态版 ★★★★★ 宠物小精灵变态破解版是一款能让玩家在游戏中宠物小精灵的世界中成就训练师伟大梦想的角色扮演类手游，在宠物小精灵变态破解版中， 点击下载	传奇1.76手机版无限金币版  244M / 中文 / 7.0.118安卓版 ★★★★★ 传奇1.76手机版无限金币版是一款传奇风格的角色扮演类手游，相信老玩家一定还记得端游传奇1.76吧，这款游戏延续了端游的核心玩 点击下载

MMO类游戏常见威胁-逻辑篡改

常见技术

- 本地文件修改
 - 类似代码修改，对一些数值/配置文件的位置、格式进行逆向，然后篡改

JOSS

MMO类游戏常见威胁-功能辅助

攻击目标：替代玩家完成某些不可能完成的操作（从人类的角度）、重复性的操作。

影响：极大的增强玩家某些方面的游戏实力或降低玩家某些方面的游戏难度，给游戏平衡性带来影响。



MMO类游戏常见威胁-功能辅助

常见技术

- 能力增强（自动瞄准、透视等功能）
 - 对内存中某些特定数据结构进行读取，转化为可读信息后反馈给游戏操作者或直接操作游戏
 - 对屏幕图像、声音等数据进行捕捉，从中提取额外的信息（较难实现）



MMO类游戏常见威胁-功能辅助

常见技术

- 挂机脚本（替代玩家进行游戏操作）
 - 基于对游戏逻辑的逆向分析，从内存、网络流量、屏幕图像中读取当前的游戏状态
 - 通过系统接口替代玩家做出操作
 - adb shell input
 - Monkey
 - Accessibility辅助功能
 - 调用内存中的函数
 - 发送网络包



MMO类游戏常见威胁-协议劫持

攻击目标：劫持游戏客户端和服务端之间的通信信道

影响：客户端和服务端之间可能会传递和玩家相关的敏感数据（账号信息、聊天信息等），被窃取后会危害玩家账号安全和隐私安全；伪造通信数据可以达到欺骗服务器的目的，从而对服务器的数据和运算逻辑产生影响，达到作弊的目的。

JOSS

MMO类游戏常见威胁-协议劫持

常见技术

- Burpsuite和wireshark抓包分析（部分游戏直接明文传输数据）
- 直接重放
- 逆向游戏代码中和网络通信相关的部分，分析网络流量格式以及功能。对截获的网络流量进行解密/篡改或者脱离游戏直接和服务器通信。
- 动态调试/hook游戏代码，使用游戏内接口伪造通信包

JOSS



MMO类游戏常见保护技术

GOSS

MMO类游戏常见保护技术

攻击方

- 身份为设备的拥有者
- 拥有所有设备资源，可以进行任意操作
 - 所有的权限
 - 查看、操作其他进程
 - 虚拟机、内核、外接设备等
- 可以有复杂的部署、使用方式
- 不计成本

防守方

- 身份为应用程序的开发者
- 拥有有限的资源，操作受限
 - 仅能够操作自身的进程
 - 获取其他进程的部分信息，无法操作
- 有限的部署、使用方式
- 需要考虑对游戏本身的影响



MMO类游戏常见保护技术-资源加密

目标：保护游戏相关的本地资源不被窃取、逆向、篡改

常见技术

- 对需要保护的文件进行加密，同时修改引擎中相关加载逻辑，实现强度合理的加密
 - Cocos2d以及自定义引擎可以直接修改
 - Unity3D可以通过patch libunity.so的方式实现（patchelf等工具）
 - 基于白盒的高强度加密算法
- 混淆代码、修改文件头以对抗常见的逆向工具
- 对引擎的相关逻辑进行混淆，增加通过动态调试直接获取明文数据的难度
- 减少明文数据在内存中存在的时间

MMO类游戏常见保护技术-代码完整性保护

目标：保护游戏本地运算逻辑不被篡改

常见技术

- 在文件加密的基础上，对关键的代码文件和数值文件进行完整性校验
- 对内存中的关键代码块、函数入口进行完整性校验
- 进行重打包检测
 - apk完整性
 - 签名
 - 特征值
- 调试、hook检测

MMO类游戏常见保护技术-抗内存修改

目标：保护游戏运行时内存数据不被窃取、篡改

常见技术

- ptrace检测
- 使用inotify等工具监控/proc/<pid>/mem
- 对内存中的关键数据采用加密存放和运算

JOSS

MMO类游戏常见保护技术-运行环境检测

目标：确保游戏运行时没有修改器等外挂在运行

常见技术

- 受限于权限，一般采用黑名单机制
- 常见外挂的特征（包名、进程名、签名、代码等）
- 常见的hook工具特征（xposed、adbi等）

JOSS

MMO类游戏常见保护技术-服务器同步运算

目标：在服务器端发现已经发生的作弊行为

常见技术

- 将尽可能多的运算放在服务器进行
- 在游戏运行时收集用户的操作数据和游戏的运行数据，上传至服务器进行校验

JOSS

MMO类游戏常见保护技术-安全网络通信

目标：保证客户端与服务器之间的网络通信数据不被窃取、篡改

常见技术

- 使用安全的网络协议并正确部署（https）
- 对基于socket的通信流量进行加密/校验处理
 - 使用安全的加密算法（AES-CBC等）
 - 使用安全的哈希算法（SHA256等）
 - 使用安全的密钥交换协议来协商密钥（RSA、ECDH等）

MMO类游戏常见保护技术

Last But Not Least

- 分析待保护游戏的整体架构（代码、服务器、游戏玩法）
 - 定位需要保护的元素
 - 制定保护方案
- 多种技术协同
- 即时更新各类保护方案（尤其是存在被破解可能性的）
- 跟进新出现的外挂、作弊技术
- 尽可能收集用户游戏时的各项数据

MMO类游戏常见保护技术

Last But Not Least

- 分析待保护游戏的整体架构（代码、服务器、游戏玩法）
 - 定位需要保护的元素
 - 制定保护方案
- 多种技术协同
- 即时更新各类保护方案（尤其是存在被破解可能性的）
- 跟进新出现的外挂、作弊技术
- 尽可能收集用户游戏时的各项数据

特别鸣谢



张奇



张倬



简鲲鹏



刘穆清

JOSS

THANK YOU

JOSS