



Off The Grid 游戏分析报告

2024.10.16

Senna

DAMOCLES LABS

目录

- 概要
- 游戏背景
 - ◆ 游戏版本
 - ◆ 游戏类型&游戏引擎
 - ◆ 游戏玩法可能存在的问题
- 游戏安全分析
 - ◆ 游戏代码保护
 - ◆ 游戏基础反作弊
 - ◆ 游戏逻辑问题
 - ◆ 游戏协议
- Web3 安全分析
 - ◆ 代币合约安全
 - ◆ 游戏内经济系统安全
- 关于 Damocles

一、 概要

OTG (Off The Grid) 是一款 3A 级的大逃杀类 FPS GameFi, 该游戏采用了 Easy Anti-Cheat 反作弊方案, 因此在游戏运行时存在基础的反作弊方案, 但是由于 EAC 是一款免费的反作弊引擎, 因此绕过保护并不复杂; 同时由于只有基础的防护方案, 并没有针对外挂功能定制化的功能方案的加持, 导致玩家如果作弊的话其被检出率是很低的, 并且现在市面上已经有外挂开始售卖, 所以从整体游戏环境来看, 该游戏的安全性评分为 3 分。

安全性评分:



二、 游戏背景

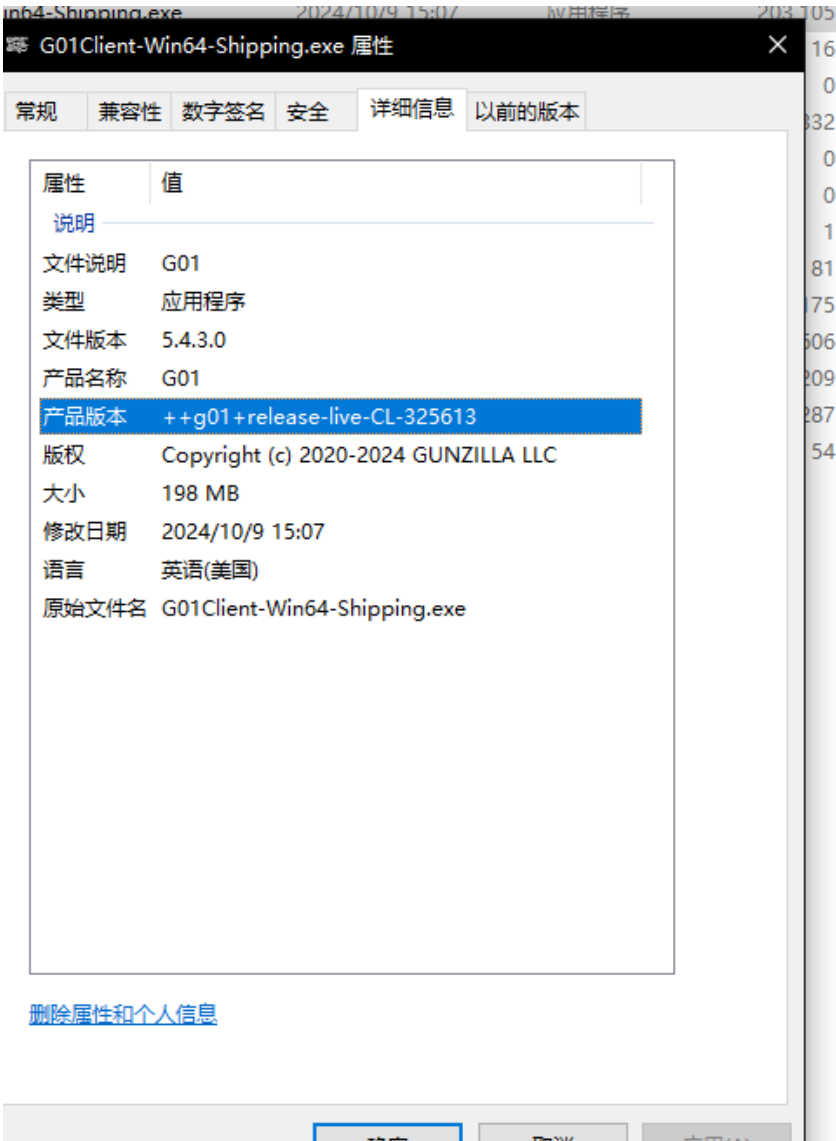
- 进行评估的游戏版本: ++g01+release-live-CL-325613
- 游戏类型&游戏引擎: FPS, UE 5.4.3
- 游戏玩法可能存在的问题:
 - 自瞄
 - 透视
 - 无后坐力
 - 子弹追踪
 - 加速
 - Fab 自定义脚本带来隐藏协议漏洞

三、 游戏安全性分析

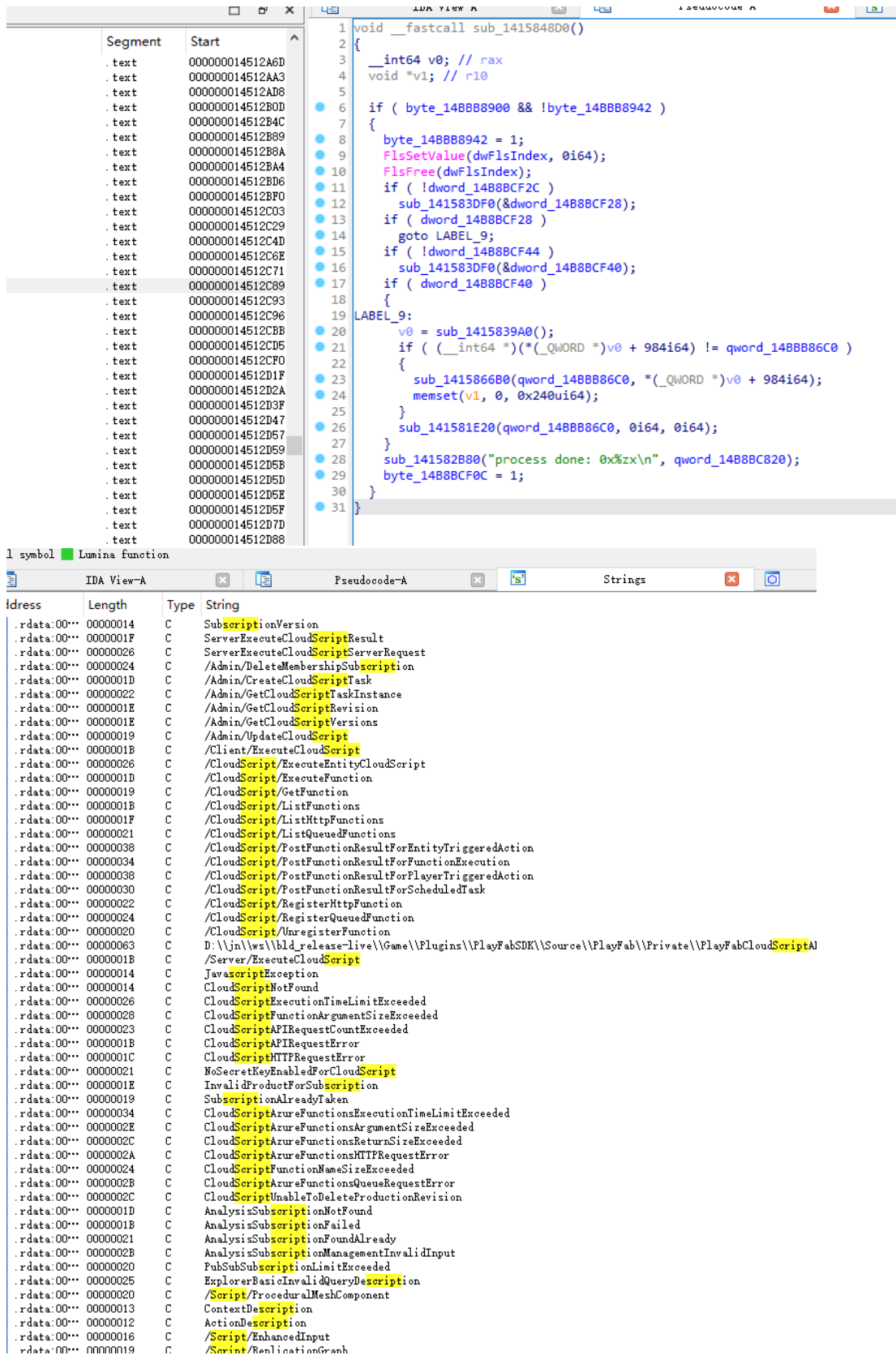
游戏代码保护：

分析过程：

1. 由于不同的引擎有不同的分析模式,所以在获取到游戏 EXE 后首先需要确定游戏使用的引擎,通过对游戏基础信息识别我们可以确定该游戏是使用 UE5 进行开发。



2. 通过 IDA 进行反编译，发现代码未加密、字符串未加密。



The screenshot displays the IDA Pro interface with the following components:

- Segment List:** A list of segments starting from 00000000 to 0000001A, all of type 'text'.
- Assembly View:** Shows the assembly code for the function `void __fastcall sub_1415848D0()`. The code includes instructions for setting up registers, conditional jumps, and function calls.
- Pseudocode View:** Shows the high-level logic of the function, including variable declarations, conditional checks, and function calls.
- Strings List:** A list of strings found in the binary, including various API names and error messages.

The assembly view shows the following code:

```

1 void __fastcall sub_1415848D0()
2 {
3     __int64 v0; // rax
4     void *v1; // r10
5
6     if ( byte_148BB8900 && !byte_148BB8942 )
7     {
8         byte_148BB8942 = 1;
9         FlsSetValue(dwFlsIndex, 0i64);
10        FlsFree(dwFlsIndex);
11        if ( !dword_1488BCF2C )
12            sub_141583DF0(&dword_1488BCF28);
13        if ( dword_1488BCF28 )
14            goto LABEL_9;
15        if ( !dword_1488BCF44 )
16            sub_141583DF0(&dword_1488BCF40);
17        if ( dword_1488BCF40 )
18        {
19            LABEL_9:
20            v0 = sub_1415839A0();
21            if ( (__int64 *)v0 + 984i64 != qword_148BB86C0 )
22            {
23                sub_1415866B0(qword_148BB86C0, *(__QWORD *)v0 + 984i64);
24                memset(v1, 0, 0x240ui64);
25            }
26            sub_141581E20(qword_148BB86C0, 0i64, 0i64);
27        }
28        sub_141582B80("process done: 0x%x\n", qword_148BB8C820);
29        byte_1488BCF0C = 1;
30    }
31 }

```

The strings list includes the following entries:

- SubscriptionVersion
- ServerExecuteCloudScriptResult
- ServerExecuteCloudScriptServerRequest
- /Admin/DeleteMembershipSubscription
- /Admin/CreateCloudScriptTask
- /Admin/GetCloudScriptTaskInstance
- /Admin/GetCloudScriptRevision
- /Admin/GetCloudScriptVersions
- /Admin/UpdateCloudScript
- /Client/ExecuteCloudScript
- /CloudScript/ExecuteEntityCloudScript
- /CloudScript/ExecuteFunction
- /CloudScript/GetFunction
- /CloudScript/ListFunctions
- /CloudScript/ListHttpFunctions
- /CloudScript/ListQueuedFunctions
- /CloudScript/PostFunctionResultForEntityTriggeredAction
- /CloudScript/PostFunctionResultForFunctionExecution
- /CloudScript/PostFunctionResultForPlayerTriggeredAction
- /CloudScript/PostFunctionResultForScheduledTask
- /CloudScript/RegisterHttpFunction
- /CloudScript/RegisterQueuedFunction
- /CloudScript/UnregisterFunction
- D:\\j\\ws\\bld_release-live\\Game\\Plugins\\PlayFabSDK\\Source\\PlayFab\\Private\\PlayFabCloudScriptAPI
- /Server/ExecuteCloudScript
- JavaScriptException
- CloudScriptNotFound
- CloudScriptExecutionTimeLimitExceeded
- CloudScriptFunctionArgumentSizeExceeded
- CloudScriptAPIRequestCountExceeded
- CloudScriptAPIRequestError
- CloudScriptHttpRequestError
- NoSecretKeyEnabledForCloudScript
- InvalidProductForSubscription
- SubscriptionAlreadyTaken
- CloudScriptAzureFunctionsExecutionTimeLimitExceeded
- CloudScriptAzureFunctionsArgumentSizeExceeded
- CloudScriptAzureFunctionsReturnSizeExceeded
- CloudScriptAzureFunctionsHttpRequestError
- CloudScriptFunctionNameSizeExceeded
- CloudScriptAzureFunctionsQueueRequestError
- CloudScriptUnableToDeleteProductionRevision
- AnalysisSubscriptionNotFound
- AnalysisSubscriptionFailed
- AnalysisSubscriptionFoundAlready
- AnalysisSubscriptionManagementInvalidInput
- PubSubSubscriptionLimitExceeded
- ExplorerBasicInvalidQueryDescription
- /Script/ProceduralMeshComponent
- ContextDescription
- ActionDescription
- /Script/EnhancedInput
- /Script/RelinquishGrab

同时可以使用 UE Dumper 进行数据结构 dump，以便快速分析

```
// Class G01.GzBaseCharacter
// 0x0150 (0x07D0 - 0x0680)
class AGzBaseCharacter : public ACharacter
{
public:
    uint8 Pad_2D2A[0x38]; // 0x0680(0x0038)(
    class UGzDamageableComponent* DamageableComponent; // 0x0688(0x0008)(
    class UGzAbilitySystemComponent* AbilitySystemComponent; // 0x06C0(0x0008)(
    class UDataTable* DefaultAttributesDT; // 0x06C8(0x0008)(
    TArray<TSubclassOf<class UGameplayEffect>> StartupEffects; // 0x06D0(0x0010)(
    class FName CapsuleCollisionProfileName; // 0x06E0(0x0008)(
    class APlayerState* PersistentPlayerState; // 0x06E8(0x0008)(
    struct FGzNativeCharacterComponentSpec AkComponentSpec; // 0x06F0(0x0008)(
    class UGzCharacterAkComponent* AkComponent; // 0x0770(0x0008)(
    class UGzZoneTrackingComponent* ZoneTrackingComponent; // 0x0778(0x0008)(
    TArray<class UGzBPOOnlyCharacterComponentSpec*> BPOOnlyComponentSpecs; // 0x0780(0x0010)(
    TSubclassOf<class UGzAITokenComponent> AITokenComponentClass; // 0x0790(0x0008)(
    class UGzCombatComponent* CachedCombatComponent; // 0x0798(0x0008)(
    class UGzEnvironmentZoneManagerComponent* EnvironmentZoneManager; // 0x07A0(0x0008)(
    class UGzInvComponent* InvComponent; // 0x07A8(0x0008)(
    class UGzAITokenComponent* AITokenComponent; // 0x07B0(0x0008)(
    struct FLyraReplicatedAcceleration ReplicatedAcceleration; // 0x07B8(0x0003)(
    uint8 Pad_2D2B[0x5]; // 0x07BB(0x0005)(
    FMulticastInlineDelegateProperty_ OnRepPlayerState; // 0x07C0(0x0010)(

public:
    class USceneComponent* GetBPComponent(TSubclassOf<class USceneComponent> ComponentClass, const class FName CompName
    void OnRep_PersistentPlayerState();
    void OnRep_ReplicatedAcceleration();
}
```

因此通过数据结构和代码可以快速的对游戏逻辑进行理解。

分析结论：

结论：OTG 在游戏代码保护方面得分为 0 分，其 client 代码未加密，字符串未加密，

用户可以很轻松的 dump 游戏的数据结构，从而进行快速分析。

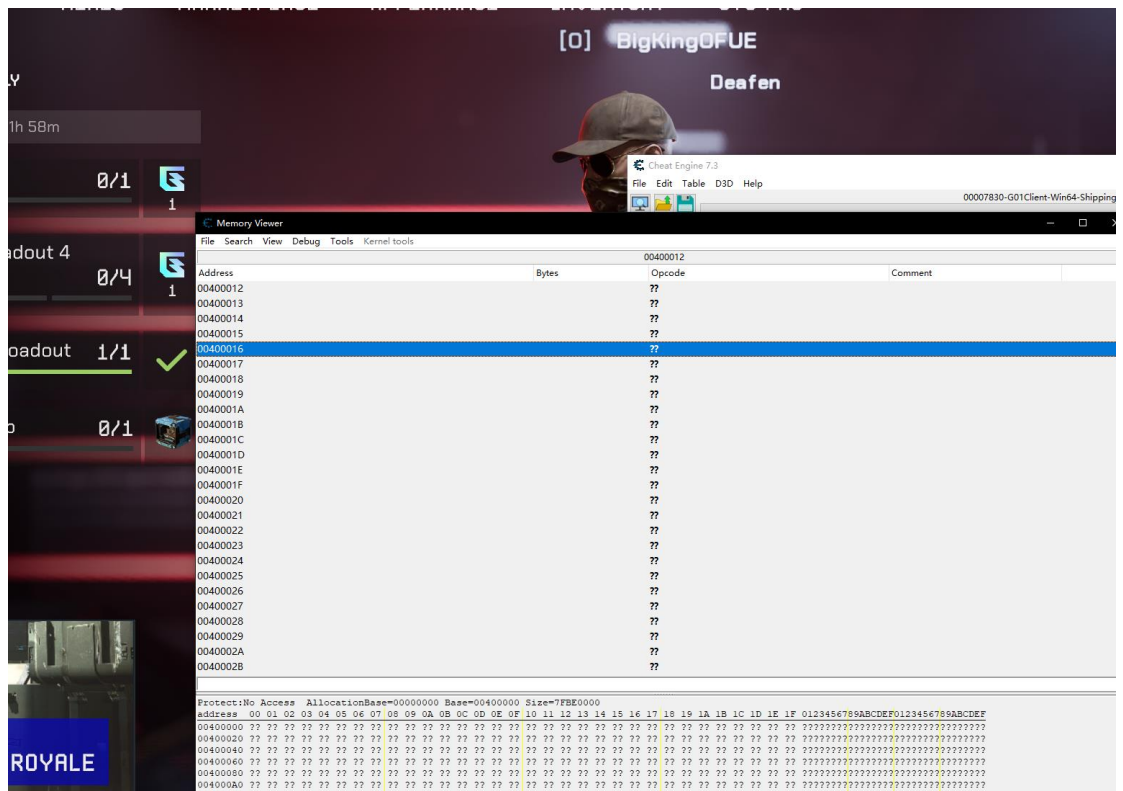
修复建议：增加对代码的本地加密，字符串的本地保护。

游戏基础反作弊：

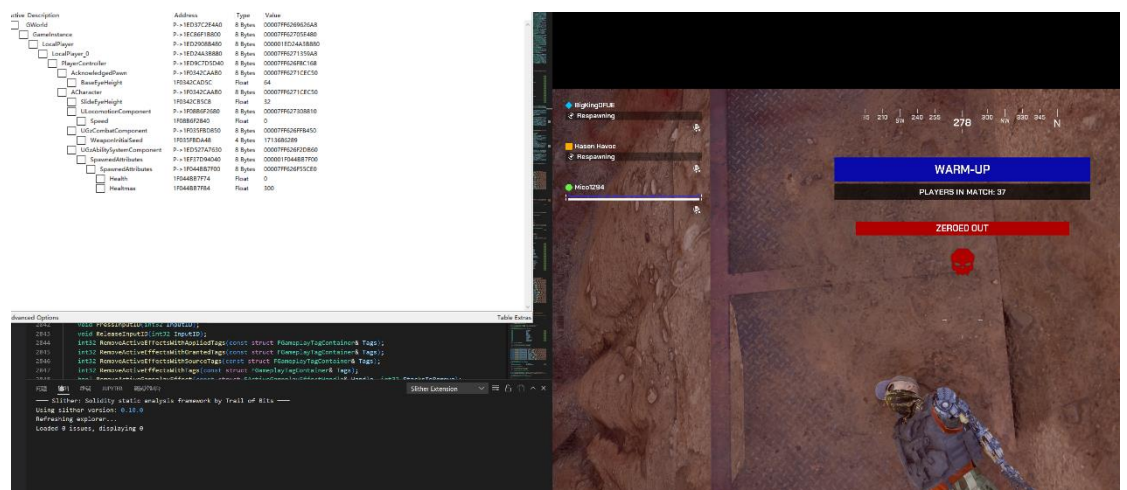
分析过程：

1. 在基础反作弊检测方面，我们主要从两个方面进行测试，一个是游戏是否存在反调试，另一个是游戏是否存在读写保护。

2. 在游戏打开状态下使用 CE 进行内存查看，发现无法扫描到内存，即 EAC 生效。



3. 使用特制 CE 进行附加以后发现, 可以对内存的读取与写入, 因此结合 dump 以后的结构体与 IDA 进行代码分析。



分析结论：

1. OTG 在反作弊对抗上基本保护为 3 分，游戏通过引入 EAC 对游戏进行保护，在一定程度上来说提升了一定的安全性，能够提升分析或者攻击游戏的技术门槛，但是对于有经验的攻击者无法进行有效的防御，同时由于缺少功能方案，对于攻击者来说只要能够绕过 EAC 的进程保护就可以对游戏数据进行修改，且无法被检测到。
2. 只测试反调试和读写保护两个方面的原因是对于一块外挂来说，找数据与实现功能只需要通过调试和读写就可以实现。如果最基础的两个保护能力都缺失的话，那么一些注入、hook 等检测也毫无意义。

修复建议：增加功能方案，同时将敏感数据加入到同步框架种。

游戏协议&逻辑安全性分析

分析过程：

1. 通过对游戏结构体与代码逻辑的分析发现，在针对武器的属性同步时，并未将某些针对性数据同步到服务器，因此可以在本地修改属性值，配合自瞄功能来进行作弊。即：

```
class AGzWeaponActor : public AAActor
{
public:
    uint8                                Pad_30B2[0x10];
    class AAActor*                        MagazineProp;
    struct FGzWeaponConstructionInfo      ConstructionInfo;
    class USkeletalMeshComponent*         SkeletalMeshComponent;
    class UGzWeaponComponent*             WeaponComponent;
    class UGzWeaponItemData*              WeaponItemData;
    class UGzWeaponSkinItemData*          WeaponSkinItemData;
    class UGzWeaponAttachmentComponent*   AttachmentComponents[0x8];
    class UAkComponent*                   AkComponent;
    TArray<struct FGzInventoryItemAttachmentContent> PendingAttachments;
    TArray<class UObject*>                AttachmentModifierResources;
    TArray<class UGzWeaponBehaviorAttachment*> BehaviorAttachments;
    uint8                                Pad_30B3[0x20];

public:
    void OnInitBehaviorAttachments();
}
```


比如 SpreadData、SwayData、RecoilData 这部分参与到射击落点判定，与击中判定逻辑种的数据。

2. 该游戏采用的 Azure Play Fab Game Server 解决方案来作为游戏服务器，链上操作包括 Decode Hex 功能是通过 GS 中的 Cloud Script 来进行上链。同时由于 PlayFab 依托于 RestAPI 因此项目方需要严格保存好 APP:title 权限的账号，避免改权限账号的滥用或者 Secret Key 泄露导致的游戏数据删除等风险。

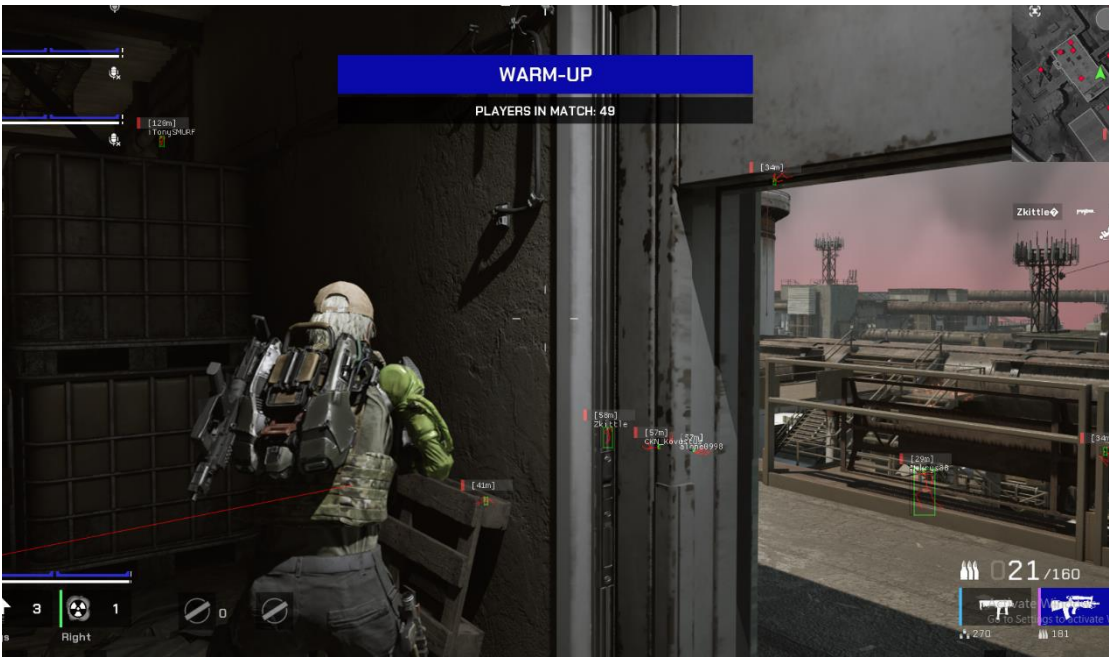
```
POST https://1E5FA.playfabapi.com/CloudScript/ExecuteFunction?sdk=UE4M3FL-1.120.230707 HTTP/1.1
accept: */*
accept-encoding: deflate, gzip
content-length: 141
content-type: application/json; charset=utf-8
host: 1E5FA.playfabapi.com
user-agent: GDI/++g0l+release-live-CL-325613 (http-legacy) Windows/10.0.19045.1.256.64bit
x-entitytoken:
HbZeHFWdjs3RFl1VURHsdhGw9hWnJYQ1Vqanlkdl0o0ElG81Y5K2x4MzRjPYx7Im10iIyMDIOLTEwLTE2VDE0Qj930jE3W1IzIm1k-cI6Tl9wZw5JZEN+lw5LY3QilLCJLIjoIMjAyaWCOwMCOwN1QzNDw0WzozNioiLCJmSI6IjIwMjQ
QwNDw0MTdaliwi dGllIjoIY3YzV1NBVzFFanMiLCJpZGkiOiJodHwzovL2FwS51oG1jZ2FtZUMuZGV2L2VwawMw+b2F1dGvdjF0MzB1MDdaNGIyNDY4NGM5MGEzZGJkYWNEMjU4NTM0NGIiLCJ0IjoIaW50ZkZjaWVwIiLCJlYyI6ImRp
llcl9hY2N+dw50IUVBQzY3NjhhMUNwOEERBTUwM0U1RkE+Q0EzRjNEENTY3RETYoTREOC81MDMyOEZBOTFBMOZGNDk4LyIsImVpIjoIMTNUMjhhGQTKxQTNWRjQ5OCiSIzImV0IjoI dG10bGVFeGxhaWVyaXZFY291bnQiEQ==
x-playfab-sdk: UE4M3FL-1.120.230707

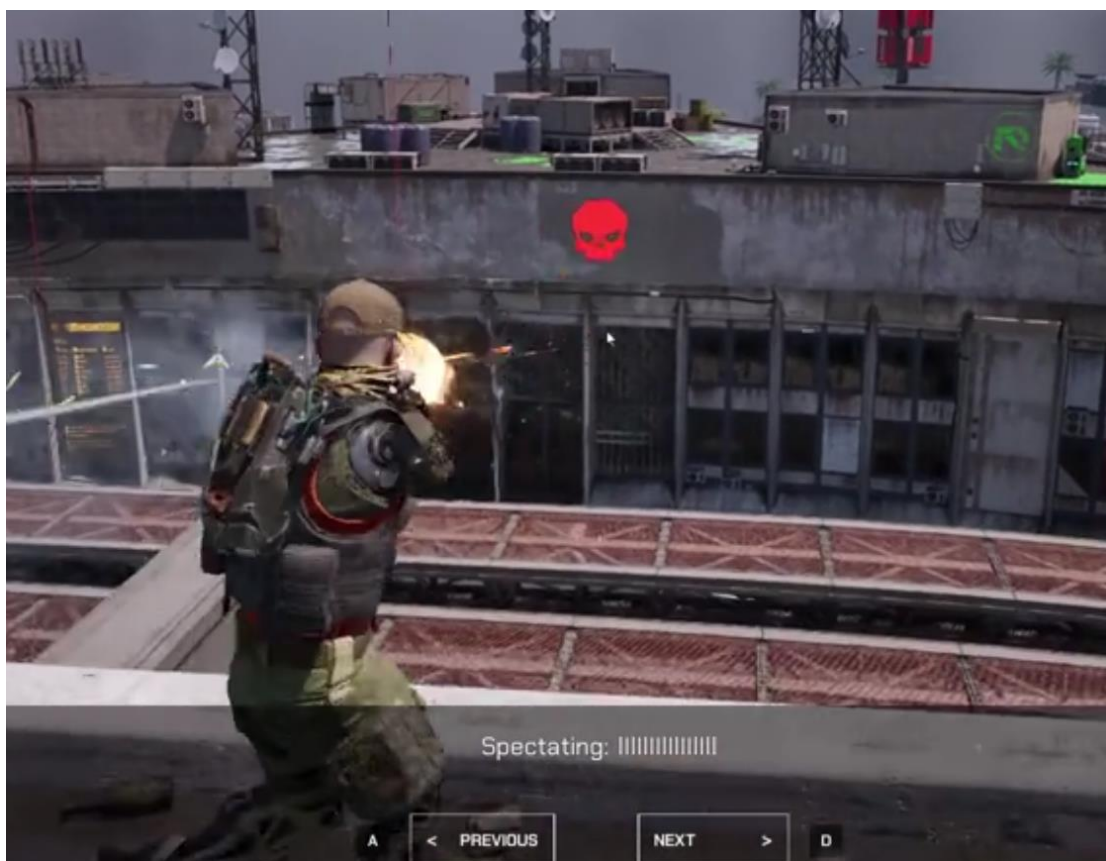
{"FunctionName": "GetPrimaryBalance", "FunctionParameter": {"sessionId": "f0273993-57b4-4e9c-bc67-7502b1368a88", "v": "325613", "featureSwitches": {}}}
```

```
HTTP/1.1 200 OK
Cache-Control: no-cache, no-store, must-revalidate
Content-Length: 179
Content-Type: application/json
Expires: 0
Pragma: no-cache
access-control-allow-credentials: true
access-control-allow-headers: Content-Type, Content-Encoding, X-Authentication, X-Authorization, X-PlayFabSDK, X-ReportErrorAsSuccess, X-SecretKey, X-EntityToken, Authorization, x-ms-app,
x-ms-client-request-id, x-ms-user-id, traceparent, tracestate, Request-Id
access-control-allow-methods: GET, POST
access-control-allow-origin: *
date: Wed, 16 Oct 2024 14:47:31 GMT
server: istio-envoy
vary: Accept-Encoding
x-envoy-upstream-service-time: 56
x-requestid: 585ce7782af49b6c8a9191f35ff856
x-tracecontext-traceid: 6ff4caae506051ef7499bbeb95e4a69a

{"code":200,"status":"OK","data":{"ExecutionTimeMilliseconds":8,"FunctionName":"GetPrimaryBalance","FunctionResult":{"Code":1000,"BackendVersion":250797,"BalanceString":"10.26"}}
```

3. 同时发现目前市面已有多款外挂，还是需要项目方增加重视





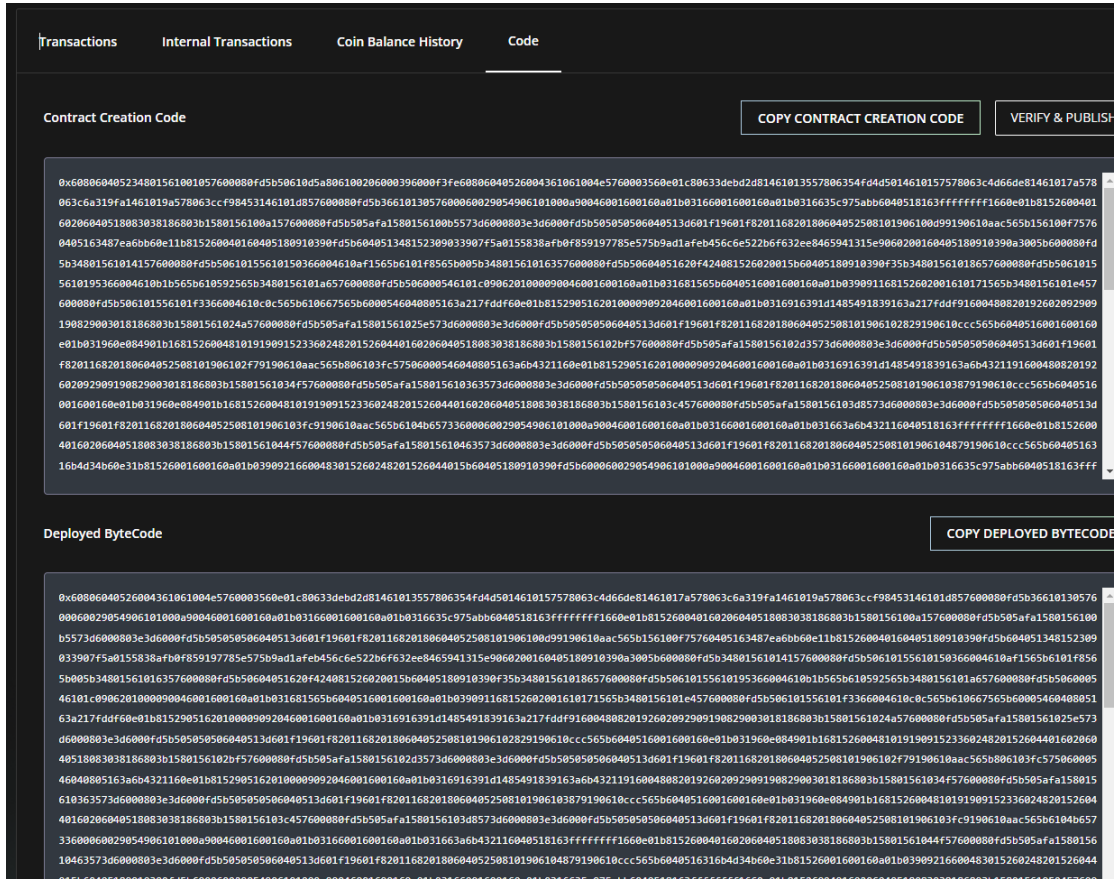
分析结论：

当前逻辑存在风险该风险的主要成因是由于同步不同步导致客户端容易出现恶性外挂功能，同时由于链上交互脚本不透明，无法判断风险状况，建议项目方对功能脚本进行严格审计，同时对权限账户进行严格控制。

修复建议：增加敏感数据同步，脚本交互加密，同时增加功能方案。

WEB3 安全分析：

OTG 目前代币合约代码并未开源，采用代理合约的方式



Decoder 合约开源，每次调用均由 GameServer 上的 Cloud Script 进行代理调用，因此风

险较小，主要的资产风险还是要集中在代理函数本身的安全性上。

关于 Damocles

Damocles labs 是成立于 2023 年的安全团队, 专注于 Web3 行业的安全, 业务内容包括:

GameFi 安全顾问、合约代码审计, 业务代码审计, 渗透测试, GameFi 漏洞挖掘, GameFi



外挂分析, GameFi 反作弊。

我们会在 Web3 安全行业持续发力, 并且尽可能多的输出分析报告, 提升项目方和用户
对 GameFi 安全的感知度, 以及促进行业的安全发展。

官网: <http://damocleslabs.com/>

Twitter: <https://twitter.com/DamoclesLabs>