Damocles

BigTime Analysis Report

2023.11.03

Senna

DAMOCLES LABS

Contents

一、 Summary

BigTime was launched with its token on October 10, 2023, sparking a GameFi frenzy. Our team has been following BigTime since September, but due to lack of eligibility, we were unable to conduct an analysis. After the recent reduction in registration requirements, we began a series of security analysis and testing on BigTime, including testing for game client attribute tampering, malicious GameRPC invocation, and token contract auditing. Through an overall evaluation of the game, we found that its security is poor, with low cheating costs for malicious players and low difficulty in analyzing the game. If the project team intends to sustain the game's operation, prioritizing the improvement of game security and fairness should be paramount.

二、 Game Background

- Game Version: v0.28-CL#78459
- Genres & Engine：MMORPG，UE4.27
- Possible Issues in GamePlay：
  - Illegal Movement (malicious packet manipulation through RPC for teleportation, speed hacks, etc.)
  - Speed Hacks (manipulation of in-game world time and UE framework time functions)
  - One-Button Combos/Looping Skills
  - NFT Forging Speedup

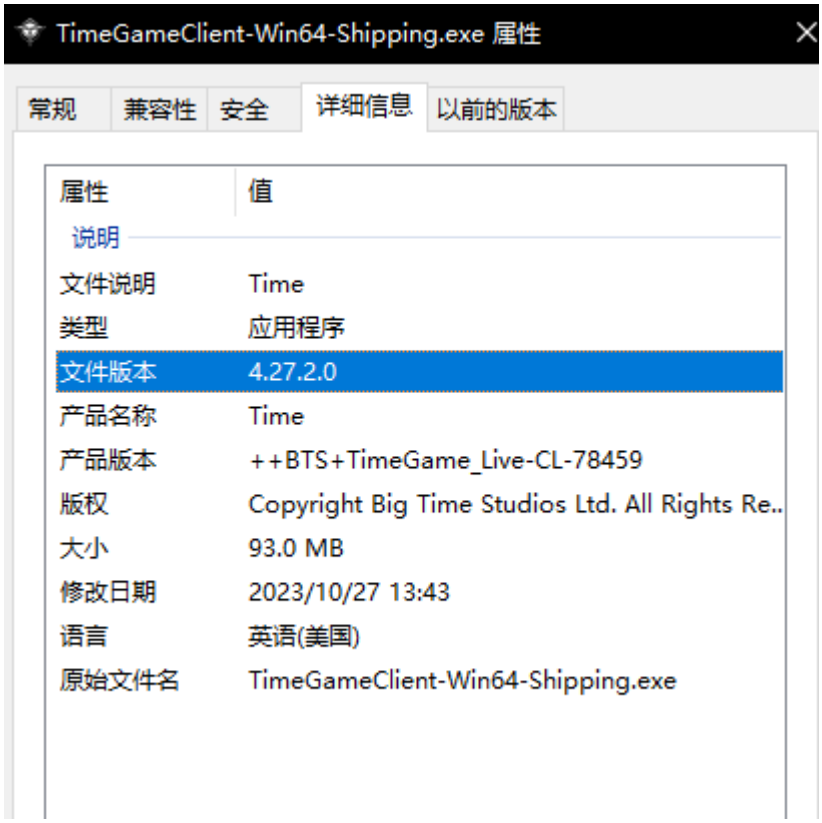
- NFT Random Number Manipulation
- Multiple Settlements after Dungeon Completion

三、 Game Security Analysis

Game Code Protection：

Process Analysis：

1. Since different game engines have different analysis modes, it is necessary to determine the engine used by the game after obtaining the game EXE. By identifying the basic information of the game, we can confirm that the game was developed using UE4.27.2.



2. After importing the game into IDA, it was observed that the game code was not obfuscated. Additionally, by utilizing feature code search specific to UE4.27, the GWorld variable could be quickly located.

```
ext:00000001407FD33D F3 [illegible]                    movss   xmm0, cs:dwo[illegible]
ext:00000001407FD345 0F C9                             xorps   xmm1, xmm1
ext:00000001407FD348 0F C1                             ucomiss xmm0, xmm1
ext:00000001407FD34B 7[illegible]                      jz      short loc_140[illegible]
ext:00000001407FD34D 4[illegible]                      mov     rbx, cs:Global_GWorld
ext:00000001407FD354 4[illegible]                      test    rbx, rbx
ext:00000001407FD357 7[illegible]                      jz      short lc[illegible]
ext:00000001407FD359 4[illegible]                      mov     r8b, 1
ext:00000001407FD35C [illegible]                       xor     edx, edx
ext:00000001407FD35E 48 8B[illegible]                  mov     rcx, rbx
```

```
164   }
165   LOBYTE(v11) = dword_145A0C768 != 0;
166   sub_141B47220(v12, v11);
167   if ( qword_145A0EF88 )
168     sub_143221300();
169   nullsub_2();
170   v17 = *(int *)(a1 + 3136);
171   v18 = 0;
172   LOBYTE(v3) = 0;
173   v68 = 0i64;
174   v19 = 0i64;
175   LODWORD(v78) = v3;
176   if ( (int)v17 > 0 )
177   {
178     v20 = *(_QWORD *)(a1 + 3128);
179     v21 = 0i64;
180     while ( *(_QWORD *)(*(_QWORD *)v20 + 640i64) != Global_GWorld )
181     {
182       v19 = (unsigned int)(v19 + 1);
183       ++v21;
184       v20 += 8i64;
185       if ( v21 >= v17 )
186         goto LABEL_37;
187     }
188     v68 = *(_QWORD *)(*(_QWORD *)(*(_QWORD *)(a1 + 3128) + 8i64 * (int)v19) + 176i64);
189 LABEL_37:
190     v22 = 0;
191     v23 = 0i64;
192     v76 = 0;
193     v73[0] = 0i64;
194     do
195     {
196       v24 = *(_QWORD *)(v23 + *(_QWORD *)(a1 + 3128));
197       v25 = *(_QWORD *)(v24 + 0x280);
198       if ( v25 && (*(_BYTE *)(v25 + 270) & 0x10) != 0 )
199       {
200         Global_GWorld = *(_QWORD *)(v24 + 0x280);
201         (*(void (__fastcall **)(__int64, __int64, __int64, __int64))(*(_QWORD *)a1 + 1112i64))(a1, v24, v21, v19);
202         if ( !v4 )
203           sub_143309290(*(_QWORD *)(v24 + 640), 2i64);
204         if ( !byte_145869BC1 && (unsigned __int8)sub_143679E50(*(_QWORD *)(v24 + 640)) )
205         {
206           sub_14318AFD0(*(_QWORD *)(v24 + 640));
207           sub_143162A50(*(_QWORD *)(v24 + 640), 0i64, 0i64, 0i64);
208         }
209         v26 = *(_QWORD *)(v24 + 640);
210         v27 = *(_BYTE *)(v26 + 267);
211         if ( (v27 & 1) != 0 )
212         {
213           *(_BYTE *)(v26 + 267) = v27 & 0xFE;
214           v28 = (_WORD *)sub_14365E540(v24 + 208, L"causeevent=", 0i64);
215           if ( *(_QWORD *)(v24 + 536) )
216             v29 = sub_14323AAA0();
217           else
218             v29 = 0i64;
219           v71 = v29;
220           if ( v28 && v29 )
221           {
222             v66 = 0i64;
223             v67 = 0i64;
224             sub_140800D4D0(&v66, 12i64);
225             v30 = HIDWORD(v67);
```

Furthermore, it was also observed that the strings within the game were not encrypted.

```
rdata:00··· 0000000C    C    ComboNumber
rdata:00··· 00000011    C    bIsComboFinisher
rdata:00··· 00000016    C    PerComboHitDamageMult
rdata:00··· 00000017    C    PerComboHitDamageTypes
rdata:00··· 00000018    C    EndComboTriggeringState
rdata:00··· 0000001A    C    StartComboTriggeringState
rdata:00··· 00000024    C    EBTS_WeaponState::CAN_TRIGGER_COMBO
rdata:00··· 0000000E    C    ComboAnimTags
rdata:00··· 00000013    C    ComboHeavyAnimTags
rdata:00··· 0000000C    C    ComboUIData
rdata:00··· 0000000E    C    TGComboUIData
rdata:00··· 00000029    C    PerComboHitGASConfigSetToGrantMeleeOwner
rdata:00··· 00000025    C    PerComboAnimrateScaleModifierToOwner
rdata:00··· 00000012    C    FinisherComboIcon
rdata:00··· 0000002C    C    EFloatingTextDamageModifiers::ComboFinisher
rdata:00··· 00000018    C    GetNumPrimaryComboAnims
rdata:00··· 00000011    C    IncrementComboID
rdata:00··· 00000011    C    IsOnLastComboIdx
rdata:00··· 0000000D    C    ResetComboID
rdata:00··· 00000016    C    ToClient_ResetComboID
rdata:00··· 0000000B    C    ComboIndex
rdata:00··· 00000018    C    ComboIndex_ForDmgSweeps
rdata:00··· 00000009    C    ComboIdx
rdata:00··· 0000001D    C    GetComboFinisherIconOverride
rdata:00··· 00000012    C    Common/ComboArrow
rdata:00··· 0000000C    C    ComboButton
rdata:00··· 00000009    C    ComboBox
rdata:00··· 0000001D    C    MessageLogListingComboButton
rdata:00··· 00000015    C    EditableComboBox.Add
rdata:00··· 00000018    C    EditableComboBox.Delete
rdata:00··· 00000018    C    EditableComboBox.Rename
rdata:00··· 00000018    C    EditableComboBox.Accept
rdata:00··· 00000029    C    ToolBar.SToolBarComboButtonBlock.Padding
rdata:00··· 0000002E    C    ToolBar.SToolBarCheckComboButtonBlock.Padding
rdata:00··· 00000033    C    ToolBar.SToolBarComboButtonBlock.ComboButton.Color
rdata:00··· 00000026    C    Menu.SToolBarComboButtonBlock.Padding
rdata:00··· 0000002B    C    Menu.SToolBarCheckComboButtonBlock.Padding
rdata:00··· 00000030    C    Menu.SToolBarComboButtonBlock.ComboButton.Color
rdata:00··· 00000011    C    ComboButtonStyle
rdata:00··· 0000000E    C    ComboBoxStyle
rdata:00··· 00000017    C    MenuPlacement_ComboBox
rdata:00··· 0000001C    C    MenuPlacement_ComboBoxRight
rdata:00··· 0000000D    C    SComboButton
rdata:00··· 0000006B    C    E:/Jenkins/TimeGame_LiveBuild/workspace/Engine/Source/Runtime/Slat···
rdata:00··· 00000021    C    SComboBox< TSharedPtr<FString> >
rdata:00··· 00000065    C    E:\\Jenkins\\TimeGame_LiveBuild\\workspace\\Engine\\Source\\Runtim···
rdata:00··· 0000000F    C    SComboListType
rdata:00··· 00000016    C    SComboRow<OptionType>
rdata:00··· 00000024    C    EMultiBlockType::ToolBarComboButton
rdata:00··· 00000062    C    E:/Jenkins/TimeGame_LiveBuild/workspace/Engine/Source/Runtime/UMG/···
rdata:00··· 00000014    C    SComboBox<UObject*>
rdata:00··· 00000068    C    E:/Jenkins/TimeGame_LiveBuild/workspace/Engine/Source/Runtime/UMG/···
```

Therefore, with the confirmation that the GWorld variable can be located through feature code and the absence of encryption in the game, it is possible to extract the NamePool feature code and use SDK dumping tools for dumping.

```
C BP_ComBot_NewbCave_... 3
BlueprintGeneratedClass BP_Com...
BP_ComBot_NewbCave_W-01.BP...
struct UBP_ComBot_NewbCave_...
C BP_ComBot_NewbCave_... 3
BlueprintGeneratedClass BP_Com...
BP_ComBot_NewbCave_W-02.BP...
struct UBP_ComBot_NewbCave_...
C BP_ComBot_NewbCave_... 3
BlueprintGeneratedClass BP_Com...
BP_ComBot_NewbCave_W-03.BP...
struct UBP_ComBot_NewbCave_...
C BP_MeleeWeapon_classe... 14
struct TArray<struct FComboEven...
bool ShowCombo; // 0x588(0x01)
bool PreventComboProgression; ...
void DoComboWindowFX(); // Fu...
BP_MeleeWeapon.BP_MeleeWea...
void IsComboWindowOpen(bool...
BP_MeleeWeapon.BP_MeleeWea...
void PushLastComboEventAnalyti...
.BP_MeleeWeapon_C.PushLastCo...
void Set Show Combo(bool Show...
void Set Show Combo(bool Show...
BP_MeleeWeapon_C.Set Show Co...
void PossiblyAdjustComboIdx(str...
BP_MeleeWeapon_C.PossiblyAdju...
C BP_PlayerPawn_classes.h 4
void IsComboTutorialActive(bool...
BP_PlayerPawn.BP_PlayerPawn_C...
void ResetMeleeComboFromAbili...
BP_PlayerPawn_C.ResetMeleeCo...
C BTS_GameFramework_cla... 8
int32_t ComboNumber; // 0x1a8(...
bool bIsComboFinisher; // 0x230(...
```

```
    void UpdateMiniMapTexture(); // Function TimeGame.TG_LevelAnchor.UpdateMiniMapTexture // (Final|Native|Public|BlueprintCallable) // @ 
    bool CaptureMiniMapNavTexture(struct UWorld* World, struct UTexture2D*& out_Texture, struct FBox& out_LocalSpaceMapBounds); // Functio
};

// Class TimeGame.TG_LocalPlayer
// Size: 0x280 (Inherited: 0x258)
struct UTG_LocalPlayer : UBTS_LocalPlayer {
    char pad_258[0x28]; // 0x258(0x28)
};

// Class TimeGame.TG_MeleeWeaponBase
// Size: 0x560 (Inherited: 0x2a8)
struct ATG_MeleeWeaponBase : ABTS_MeleeWeaponBase {
    char pad_2A8[0x60]; // 0x2a8(0x60)
    struct UTG_SkillTreeComponent* MySkillTreeComp; // 0x308(0x08)
    struct FTGInventoryHandle MyInventoryData; // 0x310(0x1a8)
    struct UBTS_AbilitySystemComponent* AbilitySystemComponent; // 0x4b8(0x08)
    struct UTG_InventoryItemAttributeSet* AttributeSet; // 0x4c0(0x08)
    int32_t ComboIndex; // 0x4c8(0x04)
    int32_t ComboIndex_ForDmgSweeps; // 0x4cc(0x04)
    bool DamagedSomethingThisSwing; // 0x4d0(0x01)
    bool HitSomethingThisSwing; // 0x4d1(0x01)
    char pad_4D2[0x6]; // 0x4d2(0x06)
    struct USkeletalMeshComponent* SkelMeshComp; // 0x4d8(0x08)
    bool OnEquippedCalled; // 0x4e0(0x01)
    char pad_4E1[0x7]; // 0x4e1(0x07)
    struct FSweepParameters ActiveSweepParameters; // 0x4e8(0x68)
    struct TArray<struct UMeshComponent*> SecondaryMeshesToCleanUp; // 0x550(0x10)

    void ToClient_ResetComboID(bool AlsoResetDmgTypeIndex); // Function TimeGame.TG_MeleeWeaponBase.ToClient_ResetComboID // (Net|NetRelia
    void ResetComboID(bool AlsoResetDmgTypeIndex); // Function TimeGame.TG_MeleeWeaponBase.ResetComboID // (Final|Native|Public|Blueprint
    void OnRep_MyInventoryData(); // Function TimeGame.TG_MeleeWeaponBase.OnRep_MyInventoryData // (Native|Public) // @ game+0x1779200
    void K2_OnEquipped(struct ABTS_CharacterBase* Wearer); // Function TimeGame.TG_MeleeWeaponBase.K2_OnEquipped // (Event|Public|Blueprin
    bool IsSocketUnlocked(int32_t SocketIdx); // Function TimeGame.TG_MeleeWeaponBase.IsSocketUnlocked // (Final|Native|Public|Blueprint
    bool IsOnLastComboIdx(); // Function TimeGame.TG_MeleeWeaponBase.IsOnLastComboIdx // (Final|Native|Public|BlueprintCallable|Blueprint
    void IncrementComboID(); // Function TimeGame.TG_MeleeWeaponBase.IncrementComboID // (Final|Native|Public|BlueprintCallable) // @ game
    struct FTGWeaponData GetWeaponData(); // Function TimeGame.TG_MeleeWeaponBase.GetWeaponData // (Final|Native|Public|BlueprintCallable
    void GetTargetLockOnDistances(float& out_AcquireDist, float& out_DesiredDist); // Function TimeGame.TG_MeleeWeaponBase.GetTargetLockOn
    int32_t GetNumPrimaryComboAnims(); // Function TimeGame.TG_MeleeWeaponBase.GetNumPrimaryComboAnims // (Final|Native|Public|BlueprintCa
    struct FTGInventoryHandle GetMyInventoryHandle(); // Function TimeGame.TG_MeleeWeaponBase.GetMyInventoryHandle // (Native|Public|Bluep
```

Once you have obtained the game SDK, it can expedite the analysis process.

Analysis of conclusion：

Based on the analysis, it is evident that BigTime has scored 0 in terms of game code protection. The game lacks any form of protection, such as custom encryption or code obfuscation, traditionally employed in games to safeguard their source code. The absence of robust code protection in BigTime lowers the barrier and cost for malicious players to analyze the game's code. This increases the risk of unfair advantages for players using cheats or hacks, which can significantly impact the game's economy.

The lack of code protection not only compromises the fairness of the game but also poses potential risks to the game's economic model. Without proper safeguards, malicious players can easily exploit vulnerabilities, manipulate game mechanics, or disrupt the balance of in-game resources. This can undermine the overall gameplay experience for legitimate players and damage the integrity of the game's economy.

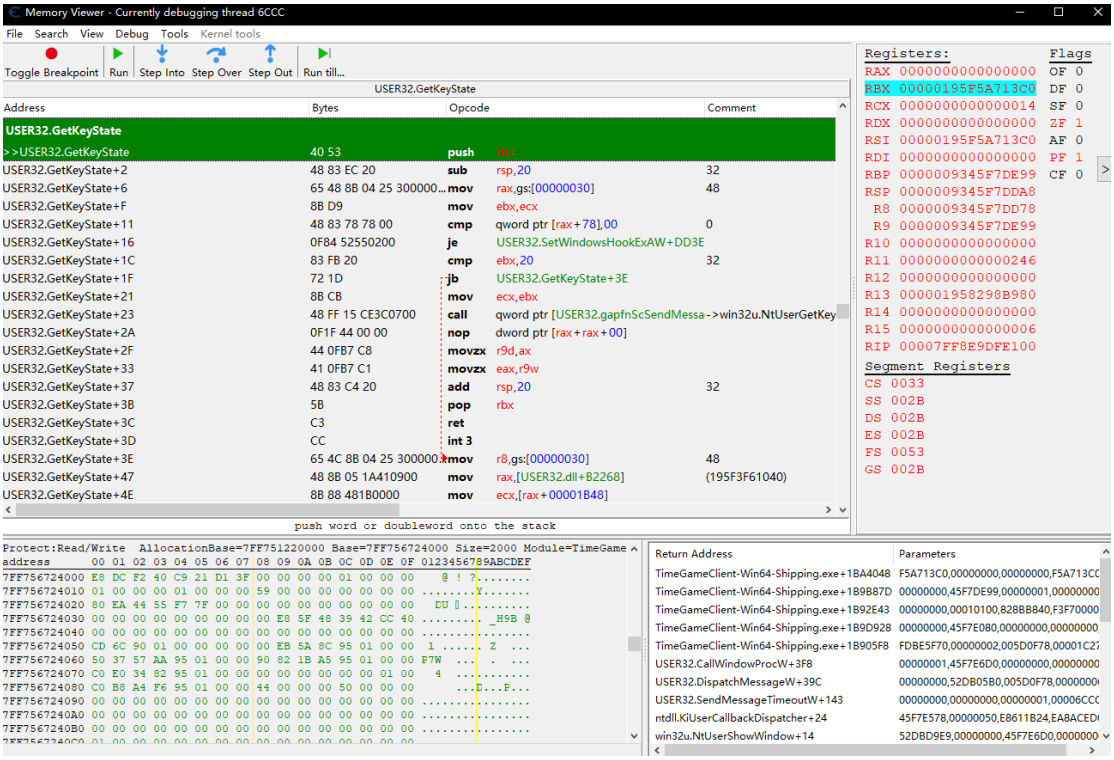
To address these issues, it is crucial for the development team behind BigTime to implement robust code protection mechanisms. These measures can include code obfuscation, encryption, anti-tampering techniques, and other security measures to make it more challenging for malicious players to analyze and manipulate the game's code. By enhancing code protection, the game can provide a fairer and more secure environment for all players and mitigate potential risks to its economy.

Damocles

Game Basic Anti-Cheat:

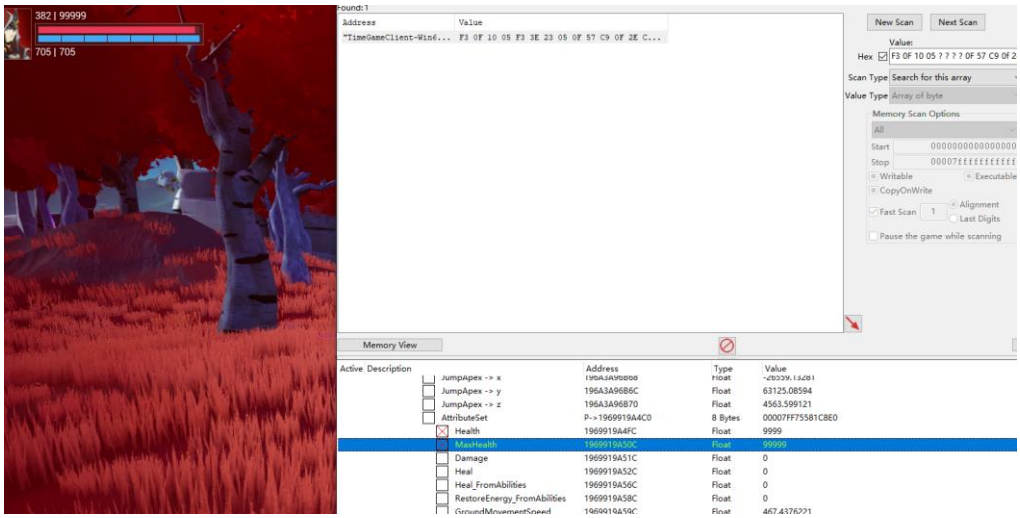
Process Analysis:

1. In terms of basic anti-cheat detection, we primarily conducted tests in two areas: anti-debugging and read/write protection.
2. While the game was running, we attempted to attach Cheat Engine (CE) and set breakpoints on common functions. However, we observed that the game did not terminate or display any error messages, indicating a lack of anti-debugging measures.



3. We attempted to modify the in-game Health value using Cheat Engine (CE) and found that the modifications took effect without triggering any pop-up windows or notifications from the game. It's important to note that modifying the Health value in this manner is only for visual

purposes, as the actual Health value is typically stored on the server. Therefore, any local modifications would not have any actual impact on the gameplay.



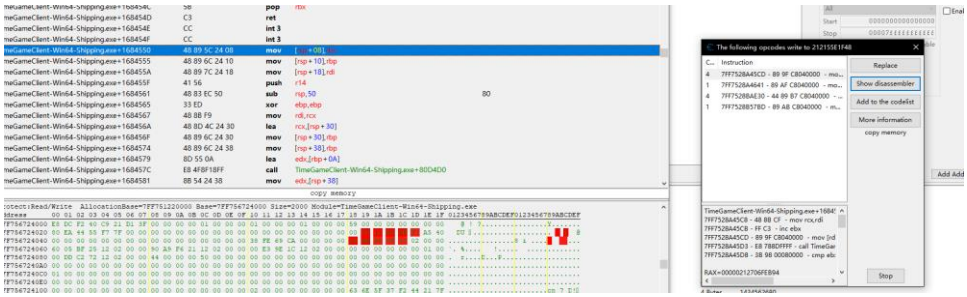
Analysis of conclusion：

1. Based on the assessment, BigTime scores 0 in terms of anti-cheat capabilities, indicating that it lacks effective measures to prevent cheating. This means that malicious users can exploit the game and engage in cheating activities without significant obstacles.
2. The reason for focusing on testing anti-debugging and read/write protection is that these two aspects are fundamental to cheat detection. For most cheat programs, data manipulation and functionality implementation can be achieved through debugging and memory read/write operations. If the basic protections in these two areas are absent, other detection methods such as injection and hooking become less meaningful or effective in preventing cheats.

Game Logic Issues

Process Analysis:

For MMO games developed using the Unreal Engine (UE), the potential benefits of tampering with local data are typically low due to the presence of robust synchronization mechanisms. These mechanisms are designed to synchronize various Actors and their attributes, as well as perform server-side validation. However, upon analyzing the source code of BigTime, it is evident that the game has not effectively implemented attribute synchronization mechanisms. Some data, such as the Combo Index feature, appears to be stored locally, allowing for potential manipulation. By setting breakpoints on the write functions related to the Combo Index and analyzing the code, it is possible to debug and manipulate the combo functionality. (Due to the specific operation will affect the fairness, so we do not demonstrate)



```
__int64 __fastcall sub_141684550(__int64 a1)
{
  int v2; // ebx
  __int64 result; // rax
  __int64 v4; // rbx
  __int64 v5; // [rsp+30h] [rbp-28h] BYREF
  __int64 v6; // [rsp+38h] [rbp-20h]
  __int64 v7; // [rsp+40h] [rbp-18h] BYREF
  __int64 v8; // [rsp+48h] [rbp-10h]

  v5 = 0i64;
  v6 = 0i64;
  sub_14080D4D0(&v5, 10i64);
  LODWORD(v6) = v6 + 10;
  if ( (int)v6 > SHIDWORD(v6) )
    sub_14080CFE0(&v5);
  sub_140866820(v5, 10, (unsigned int)"Blueprint", 10, 63);
  v2 = *(_DWORD *)(a1 + 1224) + 1;
  *(_DWORD *)(a1 + 1224) = v2;
  result = sub_14167D350(a1);
  if ( v2 >= *(_DWORD *)(result + 2048) )
  {
    v7 = 0i64;
    v8 = 0i64;
    sub_14080D4D0(&v7, 9i64);
    LODWORD(v8) = v8 + 9;
    if ( (int)v8 > SHIDWORD(v8) )
      sub_14080CFE0(&v7);
    v4 = v7;
    sub_140866820(v7, 9, (unsigned int)"Rollover", 9, 63);
    *(_DWORD *)(a1 + 1224) = 0;
    result = (*(__int64 (__fastcall **)(__int64))(*(_QWORD
    if ( (_BYTE)result )
      result = sub_1417BB9E0(a1, 0i64);
    if ( v4 )
      result = sub_141855700(v4);
  }
  if ( v5 )
    result = sub_141855700(v5);
  return result;
}
```

xrefs to sub_141684550

Directi	Ty	Address	Text
D···	j	sub_1417BD530+15	jmp sub_141684550
D···	o	.pdata:0000000145BA···	RUNTIME_FUNCTION <rva sub_141684550, rva algn_141684692, \

Line 1 of 2

OK Cancel Search Help

Analysis of conclusion：

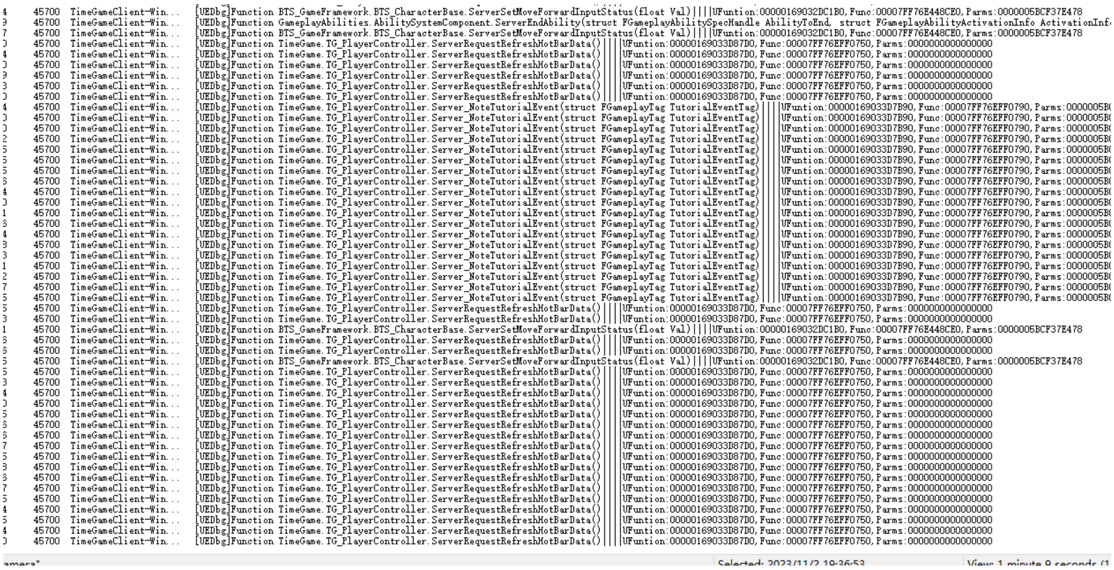
1. In terms of overall game logic security, BigTime doesn't have significant vulnerabilities. However, there are still some security risks present, leading to a logic security score of 4 out of 10.。
2. It is advisable to focus on implementing server-side encryption for sensitive attributes that lack synchronization mechanisms. By shifting the responsibility of encryption to the server, the game can enhance security and mitigate the risk of data tampering or unauthorized modifications.

Game RPC Analysis

Due to the sensitivity of RPC, the analysis is not carried out for lack of project authorization. The BigTime RPC security protection is currently 0, and for some RPC packages tested and found to be accepted by the server, the security rating

is 0. A detailed audit of RPC overall security is recommended for the project.

The following figure shows part of the RPC information。



```
4    45700  TimeGameClient-Win...   [UEDbg]Function BTS_GameFramework.BTS_CharacterBase.ServerSetMoveForwardInputStatus(float Val)||||UFuntion:00000169032DC1B0,Func:00007FF76E448CE0,Parms:0000005BCF37E478
9    45700  TimeGameClient-Win...   [UEDbg]Function GameplayAbilities.AbilitySystemComponent.ServerEndAbility(struct FGameplayAbilitySpecHandle AbilityToEnd, struct FGameplayAbilityActivationInfo ActivationInf
7    45700  TimeGameClient-Win...   [UEDbg]Function BTS_GameFramework.BTS_CharacterBase.ServerSetMoveForwardInputStatus(float Val)||||UFuntion:00000169032DC1B0,Func:00007FF76E448CE0,Parms:0000005BCF37E478
0    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
4    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
0    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
9    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
3    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
0    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
4    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
0    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
0    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
2    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
5    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
5    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
5    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
6    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
4    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
0    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
1    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
6    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
4    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
3    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
3    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
1    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
2    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
7    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
5    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.Server_NoteTutorialEvent(struct FGameplayTag TutorialEventTag)||||UFuntion:00000169033D7B90,Func:00007FF76EFF0790,Parms:0000005B
5    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
3    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
1    45700  TimeGameClient-Win...   [UEDbg]Function BTS_GameFramework.BTS_CharacterBase.ServerSetMoveForwardInputStatus(float Val)||||UFuntion:00000169032DC1B0,Func:00007FF76E448CE0,Parms:0000005BCF37E478
6    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
6    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
5    45700  TimeGameClient-Win...   [UEDbg]Function BTS_GameFramework.BTS_CharacterBase.ServerSetMoveForwardInputStatus(float Val)||||UFuntion:00000169032DC1B0,Func:00007FF76E448CE0,Parms:0000005BCF37E478
5    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
3    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
4    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
0    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
5    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
6    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
6    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
7    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
5    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
3    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
6    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
7    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
5    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
4    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
5    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
4    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
0    45700  TimeGameClient-Win...   [UEDbg]Function TimeGame.TG_PlayerController.ServerRequestRefreshHotBarData()||||UFuntion:00000169033D87D0,Func:00007FF76EFF0750,Parms:0000000000000000
amera"                                              Selected: 2023/11/2 19:36:53            View: 1 minute 9 seconds (1
```

Damocles

WEB3 Security Analysis:

Summary:

BigTime, as a blockchain game, can be divided into two parts in terms of Web3 design: the foundational BigTime token component and the in-game Web3 economic system component. This design is relatively separate compared to other games. The in-game component is responsible for token generation and NFT forging, while a fixed-circulation token contract is deployed on the Ethereum network.

Token Contract Security Analysis:

Token base information as below:

Address	0x64Bc2cA1Be492bE7185FAA2c8835d9b824c8a194
Symbol	BIGTIME
Owner	0xc3322716475fba83bfc057112247a43f1a1f2c4c**(GnosisSafe)**
TotalSupply	5,000,000,000

File 5 of 5 : BigTimeToken.sol

```
//SPDX-License-Identifier: Unlicense
pragma solidity 0.8.4;

import "@openzeppelin/contracts/token/ERC20/ERC20.sol";

contract BigTimeToken is ERC20 {
    constructor(
        string memory _name,
        string memory _symbol,
        uint _totalSupply,
        address _owner
    ) ERC20(_name, _symbol) {
        _mint(_owner, _totalSupply * 10 ** decimals());
    }
}
```

The BigTime token contract utilizes a multi-signature wallet to mint tokens and then deploys them with a fixed supply. Due to the simplicity of the current token

contract's functionality, the basic security of the contract is considered sufficient. By observing the transaction information of the Owner wallet, it can be seen that after acquiring the tokens, the Owner wallet transferred a portion of the tokens to several other wallets.

Latest 7 ERC-20 Token Transfer Events

Advanced Filter

Txn Hash	Method	Age	From		To	Value	Token
0xcf751dc512ae532e1...	Exact Input V...	22 days 47 mins ago	Uniswap V2: BABYBigT...	IN	0xC33227...1A1F2c4C	96,952,022.7685365	ERC-20: BAB.....ime
0xb020bc506679a69f5...	Exec Transact...	24 days 7 hrs ago	0xC33227...1A1F2c4C	OUT	0x46950B...E1fd7D37	500,000,000	Big Time (BIGTIM...)
0x143212da4904b110...	Exec Transact...	27 days 12 hrs ago	0x549AD7...eF5AC1F5	IN	0xC33227...1A1F2c4C	100	Big Time (BIGTIM...)
0x2ea518f26b17d3f9a...	Exec Transact...	27 days 21 hrs ago	0xC33227...1A1F2c4C	OUT	0x46950B...E1fd7D37	100	Big Time (BIGTIM...)
0x328624a5be5f0f736...	Exec Transact...	41 days 15 hrs ago	0xC33227...1A1F2c4C	OUT	0x549AD7...eF5AC1F5	500,000,000	Big Time (BIGTIM...)
0xbe427253ccfb7c04a...	Exec Transact...	41 days 15 hrs ago	0xC33227...1A1F2c4C	OUT	0x772f5e...4c7f998A	65,000,000	Big Time (BIGTIM...)
0x8115989a2546c700...	0x60806040	48 days 5 hrs ago	Null: 0x000...000	IN	0xC33227...1A1F2c4C	5,000,000,000	Big Time (BIGTIM...)

Most of these wallets are using Safe multi-signature wallets. Based on this, it can be observed that the overall security risks associated with the token mainly come from private key leaks and whether the project team has privileged accounts. Although multi-signature wallets are used, there is still a certain risk of funds being stolen if there is a leakage of the private key of a privileged account.

Game Economy System Security Analysis:

In BigTime, players can enter the space of the Time Guardians to forge time hourglasses, charge them, and perform other actions that directly affect market balance. Some of this functionality is executed locally, although it is unclear how the game server (GS) is designed. However, this behavior is considered high-risk.

```
nistrator\Downloads\UnrealDumper-4.25-main\bin\Debug\Games\TimeGameClient-Win64-Shipping\DUMP\BP_StoryUIComponent_classes.h
oid SpawnTestEncounter(struct FDataTableRowHandle Data, int32_t Enemy Level); // Function BP_StoryUIComponent.BP_StoryUIComponent_C.Spawn
oid SpawnTestEnemy(struct FDataTableRowHandle Data, int32_t EnemyLevel, int32_t EnemyAmount); // Function BP_StoryUIComponent.BP_StoryUIC
oid Server_CompleteObjective(); // Function BP_StoryUIComponent.BP_StoryUIComponent_C.Server_CompleteObjective // (Net|NetReliableNetServ
oid Server_DoOtherNamedAction(struct FString ActionString); // Function BP_StoryUIComponent.BP_StoryUIComponent_C.Server_DoOtherNamedActi
oid Server_DialogueOptionChosen(struct AActor* DialogSource, int32_t Index); // Function BP_StoryUIComponent.BP_StoryUIComponent_C.Server
oid GiveReward(float XPAmount, struct TArray<struct FTGItemGrantDatum>& Rewards); // Function BP_StoryUIComponent.BP_StoryUIComponent_C.G
oid ShowVendorInventory(); // Function BP_StoryUIComponent.BP_StoryUIComponent_C.ShowVendorInventory // (Net|NetReliableNetServer|Bluepri
oid TriggerQuestEvent(struct APawn* Instigator, struct FGameplayTag EventTag); // Function BP_StoryUIComponent.BP_StoryUIComponent_C.Trig
oid BeginQuest(struct UBTS_QuestDataBase* QuestData, struct ABTS_PlayerControllerBase* Player); // Function BP_StoryUIComponent.BP_StoryU
```

```
void ToServer_FinishCraft(struct ATG_CraftingTable_NFTWorkshopBase* Workshop, struct FString idOfActiveCraftToFinish); // Function TimeGame.TG_PlayerController_InvBase.ToServer_FinishCraft // (Net|NetReliableNative|Eve
void StartDeferredOLCurrencyBackendUpdate(); // Function TimeGame.TG_PlayerController_InvBase.StartDeferredOLCurrencyBackendUpdate // (Final|Native|Public) // @ game+0x17d6ac0
void StartDeferredNFTBackendUpdate(); // Function TimeGame.TG_PlayerController_InvBase.StartDeferredNFTBackendUpdate // (Final|Native|Public) // @ game+0x17d6aa0
void SetupEquipmentReplicators(struct ATG_PlayerPawn* ForPawn, bool InHasStoredNFTData); // Function TimeGame.TG_PlayerController_InvBase.SetupEquipmentReplicators // (Final|Native|Public) // @ game+0x17d69d0
void OnTimePieceChanged(struct UTG_InventorySetComponent* InventoryComp, struct TArray<Int32_t>& ModifiedSlotIndexes); // Function TimeGame.TG_PlayerController_InvBase.OnTimePieceChanged // (Final|Native|Public|HasOutPa
void OnServer_NotifyReceivedOpenLootCurrency(struct FTGNFTOpenLootCurrencyBalanceDatum& CurrencyDatum, struct FTGInventoryItemStorageDatum& InventoryDatum); // Function TimeGame.TG_PlayerController_InvBase.OnServer_Not
void OnServer_NotifyReceivedNFT(struct FTGNFTItemDatum& NFTDatum, struct FTGInventoryItemStorageDatum& InventoryDatum); // Function TimeGame.TG_PlayerController_InvBase.OnServer_NotifyReceivedNFT // (Native|Event|Public
void OnRep_ShopInventoryComp(); // Function TimeGame.TG_PlayerController_InvBase.OnRep_ShopInventoryComp // (Final|Native|Public) // @ game+0x17d6320
void OnRep_ItemChildInventories(); // Function TimeGame.TG_PlayerController_InvBase.OnRep_ItemChildInventories // (Final|Native|Public) // @ game+0x17d6300
void OnRep_HeldSPACEInventoryComp(); // Function TimeGame.TG_PlayerController_InvBase.OnRep_HeldSPACEInventoryComp // (Final|Native|Public) // @ game+0x17d62e0
void OnRep_HeldDECOInventoryComp(); // Function TimeGame.TG_PlayerController_InvBase.OnRep_HeldDECOInventoryComp // (Final|Native|Public) // @ game+0x17d62c0
void OnRep_HeldCosmeticsInventoryComp(); // Function TimeGame.TG_PlayerController_InvBase.OnRep_HeldCosmeticsInventoryComp // (Final|Native|Public) // @ game+0x17d62a0
void OnNFTNoLongerOwnedByPlayer(struct FGuid ItemGUID); // Function TimeGame.TG_PlayerController_InvBase.OnNFTNoLongerOwnedByPlayer // (Final|Native|Protected|HasDefaults) // @ game+0x17d6200
bool OnClient_UpgradeWorkshop(struct ATG_CraftingTable_NFTWorkshopBase* Workshop); // Function TimeGame.TG_PlayerController_InvBase.OnClient_UpgradeWorkshop // (Final|Native|Public|BlueprintCallable) // @ game+0x17d607
bool OnClient_UnLockRecipeRoll(struct ATG_CraftingTable_NFTWorkshopBase* Workshop, struct FString RecipeSlug, int32_t RollIdx); // Function TimeGame.TG_PlayerController_InvBase.OnClient_UnLockRecipeRoll // (Final|Native
bool OnClient_StartCraft_WithGameCost(struct ATG_CraftingTable_NFTWorkshopBase* Workshop, struct FString RecipeSlug, struct TArray<struct FTGInventorySlotHandle>& InputItemsToFeedTheRecipe, struct TArray<struct FTGInver
bool OnClient_StartCraft(struct ATG_CraftingTable_NFTWorkshopBase* Workshop, struct FString RecipeSlug, struct TArray<struct FTGInventorySlotHandle>& InputItemsToFeedTheRecipe); // Function TimeGame.TG_PlayerController
bool OnClient_SpeedUp(struct ATG_CraftingTable_NFTWorkshopBase* Workshop, struct FString idOfActiveCraftToSpeedUp); // Function TimeGame.TG_PlayerController_InvBase.OnClient_SpeedUp // (Final|Native|Public|BlueprintCall
bool OnClient_ResetRolls(struct ATG_CraftingTable_NFTWorkshopBase* Workshop, struct FString RecipeSlug); // Function TimeGame.TG_PlayerController_InvBase.OnClient_ResetRolls // (Final|Native|Public|BlueprintCallable) //
void OnClient_NotifyReceivedOpenLootCurrency(struct FTGNFTOpenLootCurrencyBalanceDatum CurrencyDatum, struct FTGInventoryItemStorageDatum InventoryDatum); // Function TimeGame.TG_PlayerController_InvBase.OnClient_Notify
void OnClient_NotifyReceivedNFT(struct FTGNFTItemDatum NFTDatum, struct FTGInventoryItemStorageDatum InventoryDatum); // Function TimeGame.TG_PlayerController_InvBase.OnClient_NotifyReceivedNFT // (Net|NetReliableNative
bool OnClient_LockRecipeRoll(struct ATG_CraftingTable_NFTWorkshopBase* Workshop, struct FString RecipeSlug, int32_t RollIdx); // Function TimeGame.TG_PlayerController_InvBase.OnClient_LockRecipeRoll // (Final|Native|Pu
bool OnClient_GetNewRecipeRolls(struct ATG_CraftingTable_NFTWorkshopBase* Workshop, struct FString RecipeSlug); // Function TimeGame.TG_PlayerController_InvBase.OnClient_GetNewRecipeRolls // (Final|Native|Public|Bluepri
bool OnClient_FinishCraft(struct ATG_CraftingTable_NFTWorkshopBase* Workshop, struct FString idOfActiveCraftToFinish); // Function TimeGame.TG_PlayerController_InvBase.OnClient_FinishCraft // (Final|Native|Public|Bluep
void OnChildInventoriesReplicated(); // Function TimeGame.TG_PlayerController_InvBase.OnChildInventoriesReplicated // (Event|Public|BlueprintEvent) // @ game+0x1b3dc70
bool IsItemEquippable(struct FTGInventoryItemStorageDatum& StorageDatum); // Function TimeGame.TG_PlayerController_InvBase.IsItemEquippable // (Final|Native|Public|HasOutParms|BlueprintCallable|BlueprintPure|Const) // @
bool IsAllowedToChangePocketWatches(); // Function TimeGame.TG_PlayerController_InvBase.IsAllowedToChangePocketWatches // (Final|Native|Public|BlueprintCallable|BlueprintPure|Const) // @ game+0x17d5110
```

There are many RPC functions similar to this, and considering the high cost of testing, we currently do not perform any security testing. We hope that the project team can exercise strict judgment on this part of the content on the server.

About Damocles

Damocles Labs is a security team established in 2023, focusing on security in the Web3 industry. Their core services include contract code audits, business code audits, penetration testing, GameFi code audits, GameFi vulnerability discovery, GameFi cheat analysis, and GameFi anti-cheat solutions.

Their goal is to make continuous efforts in the Web3 security industry and generate as many analysis reports as possible. They aim to enhance the awareness of GameFi security among project teams and users, as well as promote the overall security development within the industry.