Damocles

SynergyLand Security Analysis

2024.10.24

Senna

DAMOCLES LABS

Contents

Summary

SynergyLand is a top-down RPG game developed using the UE5 engine. It is currently in the testing phase, and due to the short testing period, Damocles was unable to analyze the full range of RPC protocols. Currently, analysis has been done only on Web3 aspects, as well as basic security and logic. The analysis results show that the overall security of the game is high, with robust code logic and synchronization mechanisms. However, the project team will need to strictly control the protocols in the future. Overall, it is a game with relatively high security. The security rating is 5 out of 5..

Security Rating： ★★★★★

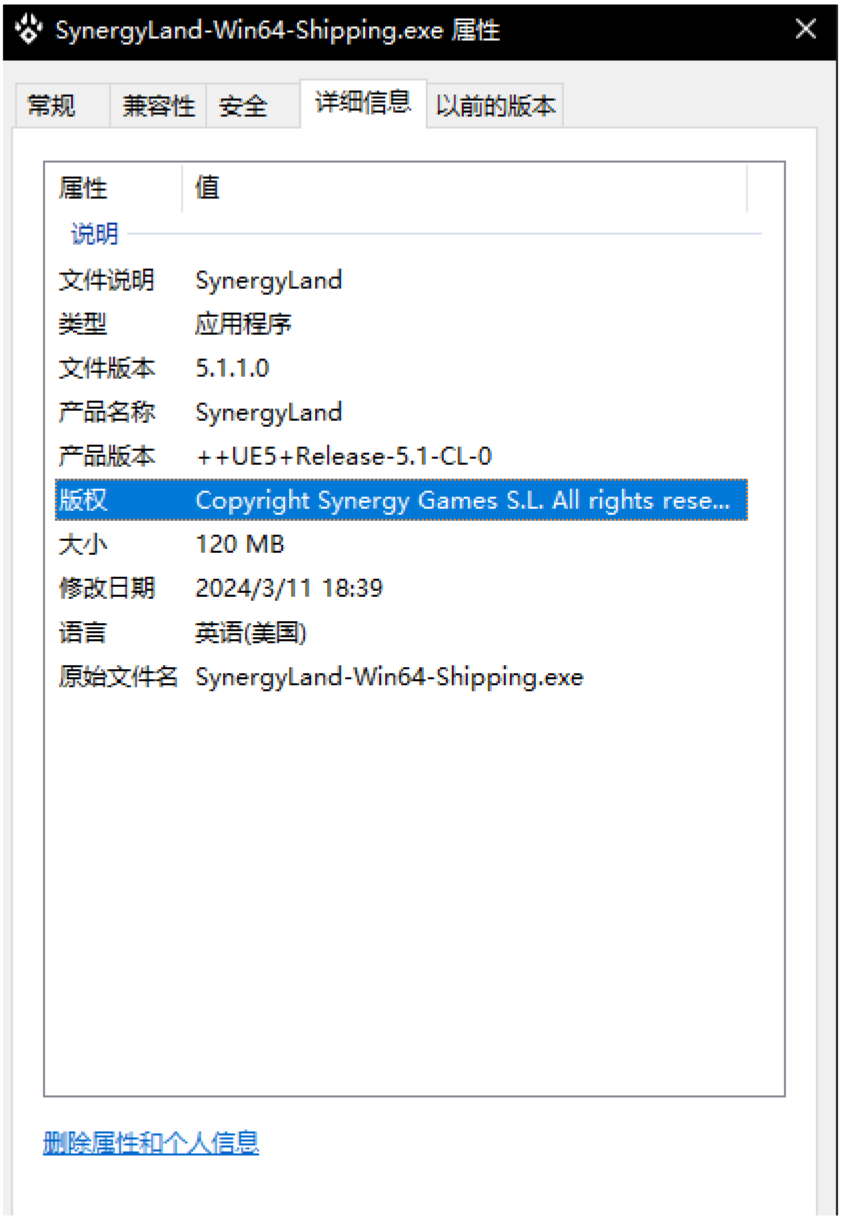
Game Background

- GGame Version Assessed: Game=27
- Game Type & Game Engine: RPG, UE 5.1.1.0
- Potential Gameplay Issues:
  - Multiple settlements
  - Ticker acceleration
  - Hidden protocol vulnerabilities brought by Fab custom scriptsGame Security Analysis

Game Code Protection：

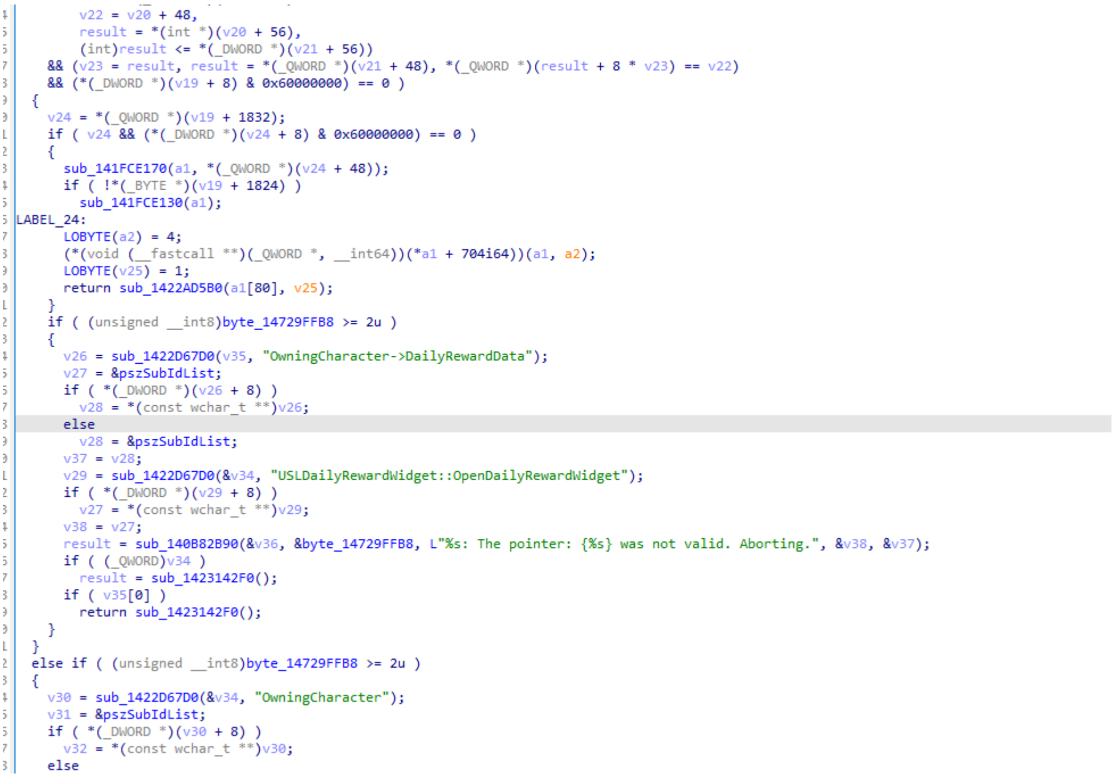
Analysis Process：

1. Determine the game engine by analyzing the game EXE since different engines have different analysis modes. Based on the identification of basic game information, we can confirm that Unity is used for game development.

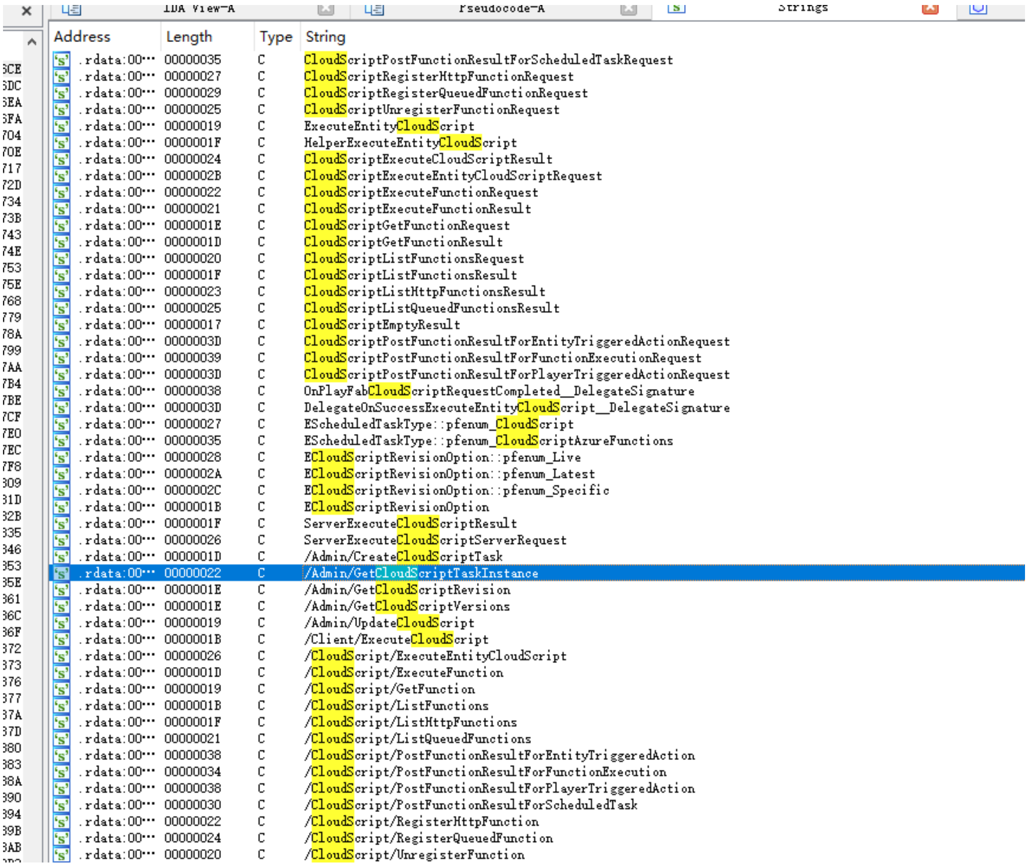


属性	值
说明	
文件说明	SynergyLand
类型	应用程序
文件版本	5.1.1.0
产品名称	SynergyLand
产品版本	++UE5+Release-5.1-CL-0
版权	Copyright Synergy Games S.L. All rights rese...
大小	120 MB
修改日期	2024/3/11 18:39
语言	英语(美国)
原始文件名	SynergyLand-Win64-Shipping.exe

2. Using IDA for decompilation, we found that the code was not encrypted and the strings were not encrypted.

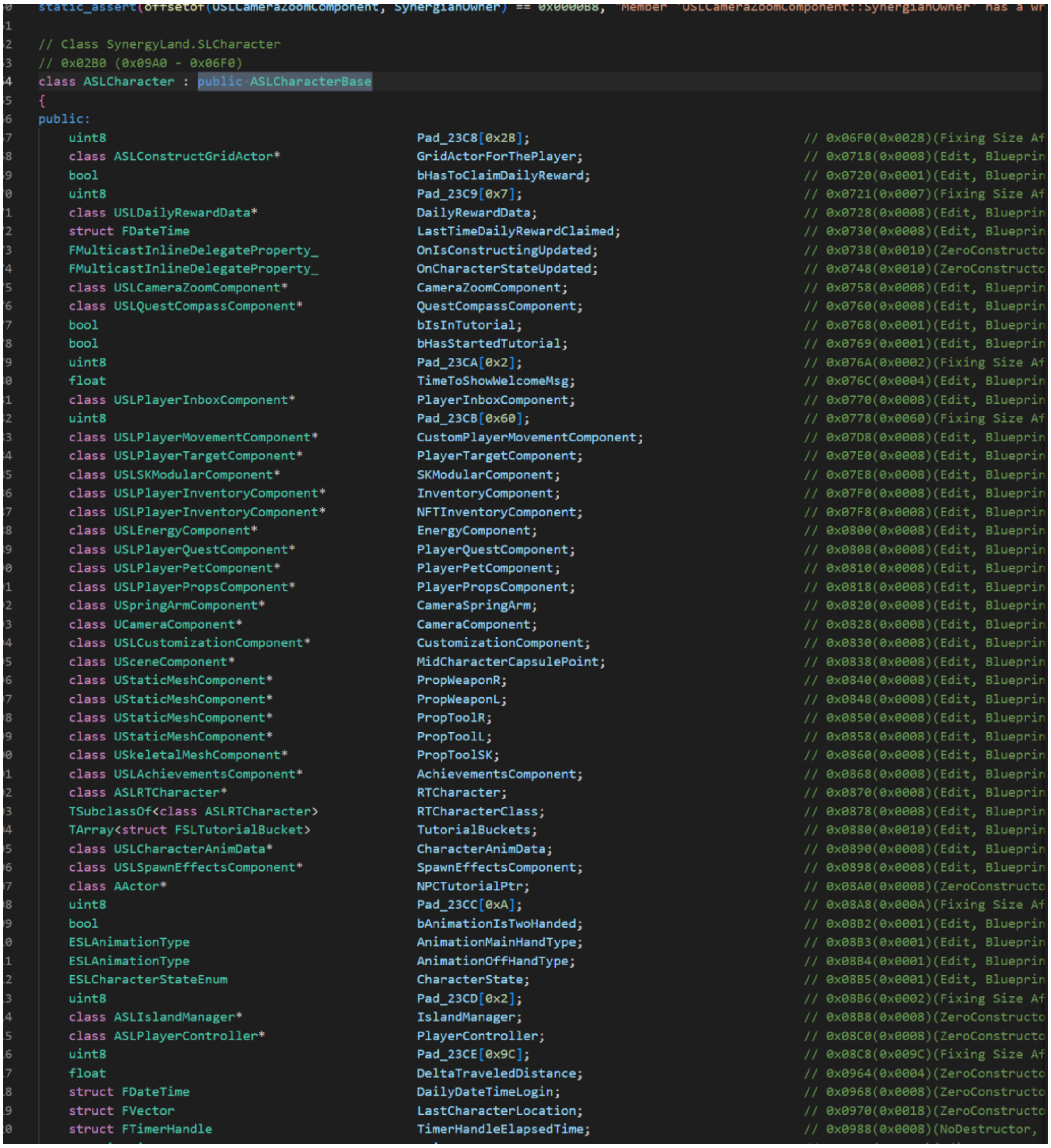


```
      v22 = v20 + 48,
      result = *(int *)(v20 + 56),
      (int)result <= *(_DWORD *)(v21 + 56))
  && (v23 = result, result = *(_QWORD *)(v21 + 48), *(_QWORD *)(result + 8 * v23) == v22)
  && (*(_DWORD *)(v19 + 8) & 0x60000000) == 0 )
 {
  v24 = *(_QWORD *)(v19 + 1832);
  if ( v24 && (*(_DWORD *)(v24 + 8) & 0x60000000) == 0 )
  {
    sub_141FCE170(a1, *(_QWORD *)(v24 + 48));
    if ( !*(_BYTE *)(v19 + 1824) )
      sub_141FCE130(a1);
LABEL_24:
    LOBYTE(a2) = 4;
    (*(void (__fastcall **)(_QWORD *, __int64))(*a1 + 704i64))(a1, a2);
    LOBYTE(v25) = 1;
    return sub_1422AD5B0(a1[80], v25);
  }
  if ( (unsigned __int8)byte_14729FFB8 >= 2u )
  {
    v26 = sub_1422D67D0(v35, "OwningCharacter->DailyRewardData");
    v27 = &pszSubIdList;
    if ( *(_DWORD *)(v26 + 8) )
      v28 = *(const wchar_t **)v26;
    else
      v28 = &pszSubIdList;
    v37 = v28;
    v29 = sub_1422D67D0(&v34, "USLDailyRewardWidget::OpenDailyRewardWidget");
    if ( *(_DWORD *)(v29 + 8) )
      v27 = *(const wchar_t **)v29;
    v38 = v27;
    result = sub_140B82B90(&v36, &byte_14729FFB8, L"%s: The pointer: {%s} was not valid. Aborting.", &v38, &v37);
    if ( (_QWORD)v34 )
      result = sub_1423142F0();
    if ( v35[0] )
      return sub_1423142F0();
  }
 }
 else if ( (unsigned __int8)byte_14729FFB8 >= 2u )
 {
  v30 = sub_1422D67D0(&v34, "OwningCharacter");
  v31 = &pszSubIdList;
  if ( *(_DWORD *)(v30 + 8) )
    v32 = *(const wchar_t **)v30;
  else
```



Address	Length	Type	String
.rdata:00···	00000035	C	CloudScriptPostFunctionResultForScheduledTaskRequest
.rdata:00···	00000027	C	CloudScriptRegisterHttpFunctionRequest
.rdata:00···	00000029	C	CloudScriptRegisterQueuedFunctionRequest
.rdata:00···	00000025	C	CloudScriptUnregisterFunctionRequest
.rdata:00···	00000019	C	ExecuteEntityCloudScript
.rdata:00···	0000001F	C	HelperExecuteEntityCloudScript
.rdata:00···	00000024	C	CloudScriptExecuteCloudScriptResult
.rdata:00···	0000002B	C	CloudScriptExecuteEntityCloudScriptRequest
.rdata:00···	00000022	C	CloudScriptExecuteFunctionRequest
.rdata:00···	00000021	C	CloudScriptExecuteFunctionResult
.rdata:00···	0000001E	C	CloudScriptGetFunctionRequest
.rdata:00···	0000001D	C	CloudScriptGetFunctionResult
.rdata:00···	00000020	C	CloudScriptListFunctionsRequest
.rdata:00···	0000001F	C	CloudScriptListFunctionsResult
.rdata:00···	00000023	C	CloudScriptListHttpFunctionsResult
.rdata:00···	00000025	C	CloudScriptListQueuedFunctionsResult
.rdata:00···	00000017	C	CloudScriptEmptyResult
.rdata:00···	0000003D	C	CloudScriptPostFunctionResultForEntityTriggeredActionRequest
.rdata:00···	00000039	C	CloudScriptPostFunctionResultForFunctionExecutionRequest
.rdata:00···	0000003D	C	CloudScriptPostFunctionResultForPlayerTriggeredActionRequest
.rdata:00···	00000038	C	OnPlayFabCloudScriptRequestCompleted__DelegateSignature
.rdata:00···	0000003D	C	DelegateOnSuccessExecuteEntityCloudScript__DelegateSignature
.rdata:00···	00000027	C	EScheduledTaskType::pfenum_CloudScript
.rdata:00···	00000035	C	EScheduledTaskType::pfenum_CloudScriptAzureFunctions
.rdata:00···	00000028	C	ECloudScriptRevisionOption::pfenum_Live
.rdata:00···	0000002A	C	ECloudScriptRevisionOption::pfenum_Latest
.rdata:00···	0000002C	C	ECloudScriptRevisionOption::pfenum_Specific
.rdata:00···	0000001B	C	ECloudScriptRevisionOption
.rdata:00···	0000001F	C	ServerExecuteCloudScriptResult
.rdata:00···	00000026	C	ServerExecuteCloudScriptServerRequest
.rdata:00···	0000001D	C	/Admin/CreateCloudScriptTask
.rdata:00···	00000022	C	/Admin/GetCloudScriptTaskInstance
.rdata:00···	0000001E	C	/Admin/GetCloudScriptRevision
.rdata:00···	0000001E	C	/Admin/GetCloudScriptVersions
.rdata:00···	00000019	C	/Admin/UpdateCloudScript
.rdata:00···	0000001B	C	/Client/ExecuteCloudScript
.rdata:00···	00000026	C	/CloudScript/ExecuteEntityCloudScript
.rdata:00···	0000001D	C	/CloudScript/ExecuteFunction
.rdata:00···	00000019	C	/CloudScript/GetFunction
.rdata:00···	0000001B	C	/CloudScript/ListFunctions
.rdata:00···	0000001F	C	/CloudScript/ListHttpFunctions
.rdata:00···	00000021	C	/CloudScript/ListQueuedFunctions
.rdata:00···	00000038	C	/CloudScript/PostFunctionResultForEntityTriggeredAction
.rdata:00···	00000034	C	/CloudScript/PostFunctionResultForFunctionExecution
.rdata:00···	00000038	C	/CloudScript/PostFunctionResultForPlayerTriggeredAction
.rdata:00···	00000030	C	/CloudScript/PostFunctionResultForScheduledTask
.rdata:00···	00000022	C	/CloudScript/RegisterHttpFunction
.rdata:00···	00000024	C	/CloudScript/RegisterQueuedFunction
.rdata:00···	00000020	C	/CloudScript/UnregisterFunction

We can also use UE Dumper to dump data structures for quick analysis understanding of the game logic.



```
static_assert(offsetof(USLCameraZoomComponent, SynergianOwner) == 0x0000B8, "Member 'USLCameraZoomComponent::SynergianOwner' has a wr

// Class SynergyLand.SLCharacter
// 0x02B0 (0x09A0 - 0x06F0)
class ASLCharacter : public ASLCharacterBase
{
public:
    uint8                                       Pad_23C8[0x28];                         // 0x06F0(0x0028)(Fixing Size Af
    class ASLConstructGridActor*                GridActorForThePlayer;                  // 0x0718(0x0008)(Edit, Blueprin
    bool                                        bHasToClaimDailyReward;                 // 0x0720(0x0001)(Edit, Blueprin
    uint8                                       Pad_23C9[0x7];                          // 0x0721(0x0007)(Fixing Size Af
    class USLDailyRewardData*                   DailyRewardData;                        // 0x0728(0x0008)(Edit, Blueprin
    struct FDateTime                            LastTimeDailyRewardClaimed;             // 0x0730(0x0008)(Edit, Blueprin
    FMulticastInlineDelegateProperty_           OnIsConstructingUpdated;                // 0x0738(0x0010)(ZeroConstructo
    FMulticastInlineDelegateProperty_           OnCharacterStateUpdated;                // 0x0748(0x0010)(ZeroConstructo
    class USLCameraZoomComponent*               CameraZoomComponent;                    // 0x0758(0x0008)(Edit, Blueprin
    class USLQuestCompassComponent*             QuestCompassComponent;                  // 0x0760(0x0008)(Edit, Blueprin
    bool                                        bIsInTutorial;                          // 0x0768(0x0001)(Edit, Blueprin
    bool                                        bHasStartedTutorial;                    // 0x0769(0x0001)(Edit, Blueprin
    uint8                                       Pad_23CA[0x2];                          // 0x076A(0x0002)(Fixing Size Af
    float                                       TimeToShowWelcomeMsg;                   // 0x076C(0x0004)(Edit, Blueprin
    class USLPlayerInboxComponent*              PlayerInboxComponent;                   // 0x0770(0x0008)(Edit, Blueprin
    uint8                                       Pad_23CB[0x60];                         // 0x0778(0x0060)(Fixing Size Af
    class USLPlayerMovementComponent*           CustomPlayerMovementComponent;          // 0x07D8(0x0008)(Edit, Blueprin
    class USLPlayerTargetComponent*             PlayerTargetComponent;                  // 0x07E0(0x0008)(Edit, Blueprin
    class USLSKModularComponent*                SKModularComponent;                     // 0x07E8(0x0008)(Edit, Blueprin
    class USLPlayerInventoryComponent*          InventoryComponent;                     // 0x07F0(0x0008)(Edit, Blueprin
    class USLPlayerInventoryComponent*          NFTInventoryComponent;                  // 0x07F8(0x0008)(Edit, Blueprin
    class USLEnergyComponent*                   EnergyComponent;                        // 0x0800(0x0008)(Edit, Blueprin
    class USLPlayerQuestComponent*              PlayerQuestComponent;                   // 0x0808(0x0008)(Edit, Blueprin
    class USLPlayerPetComponent*                PlayerPetComponent;                     // 0x0810(0x0008)(Edit, Blueprin
    class USLPlayerPropsComponent*              PlayerPropsComponent;                   // 0x0818(0x0008)(Edit, Blueprin
    class USpringArmComponent*                  CameraSpringArm;                        // 0x0820(0x0008)(Edit, Blueprin
    class UCameraComponent*                     CameraComponent;                        // 0x0828(0x0008)(Edit, Blueprin
    class USLCustomizationComponent*            CustomizationComponent;                 // 0x0830(0x0008)(Edit, Blueprin
    class USceneComponent*                      MidCharacterCapsulePoint;               // 0x0838(0x0008)(Edit, Blueprin
    class UStaticMeshComponent*                 PropWeaponR;                            // 0x0840(0x0008)(Edit, Blueprin
    class UStaticMeshComponent*                 PropWeaponL;                            // 0x0848(0x0008)(Edit, Blueprin
    class UStaticMeshComponent*                 PropToolR;                              // 0x0850(0x0008)(Edit, Blueprin
    class UStaticMeshComponent*                 PropToolL;                              // 0x0858(0x0008)(Edit, Blueprin
    class USkeletalMeshComponent*               PropToolSK;                             // 0x0860(0x0008)(Edit, Blueprin
    class USLAchievementsComponent*             AchievementsComponent;                  // 0x0868(0x0008)(Edit, Blueprin
    class ASLRTCharacter*                       RTCharacter;                            // 0x0870(0x0008)(Edit, Blueprin
    TSubclassOf<class ASLRTCharacter>           RTCharacterClass;                       // 0x0878(0x0008)(Edit, Blueprin
    TArray<struct FSLTutorialBucket>            TutorialBuckets;                        // 0x0880(0x0010)(Edit, Blueprin
    class USLCharacterAnimData*                 CharacterAnimData;                      // 0x0890(0x0008)(Edit, Blueprin
    class USLSpawnEffectsComponent*             SpawnEffectsComponent;                  // 0x0898(0x0008)(Edit, Blueprin
    class AActor*                               NPCTutorialPtr;                         // 0x08A0(0x0008)(ZeroConstructo
    uint8                                       Pad_23CC[0xA];                          // 0x08A8(0x000A)(Fixing Size Af
    bool                                        bAnimationIsTwoHanded;                  // 0x08B2(0x0001)(Edit, Blueprin
    ESLAnimationType                            AnimationMainHandType;                  // 0x08B3(0x0001)(Edit, Blueprin
    ESLAnimationType                            AnimationOffHandType;                   // 0x08B4(0x0001)(Edit, Blueprin
    ESLCharacterStateEnum                       CharacterState;                         // 0x08B5(0x0001)(Edit, Blueprin
    uint8                                       Pad_23CD[0x2];                          // 0x08B6(0x0002)(Fixing Size Af
    class ASLIslandManager*                     IslandManager;                          // 0x08B8(0x0008)(ZeroConstructo
    class ASLPlayerController*                  PlayerController;                       // 0x08C0(0x0008)(ZeroConstructo
    uint8                                       Pad_23CE[0x9C];                         // 0x08C8(0x009C)(Fixing Size Af
    float                                       DeltaTraveledDistance;                  // 0x0964(0x0004)(ZeroConstructo
    struct FDateTime                            DailyDateTimeLogin;                     // 0x0968(0x0008)(ZeroConstructo
    struct FVector                              LastCharacterLocation;                  // 0x0970(0x0018)(ZeroConstructo
    struct FTimerHandle                         TimerHandleElapsedTime;                 // 0x0988(0x0008)(NoDestructor,
```

Thus, understanding the game logic can be expedited through data structures and code analysis.

Analysis Conclusion：

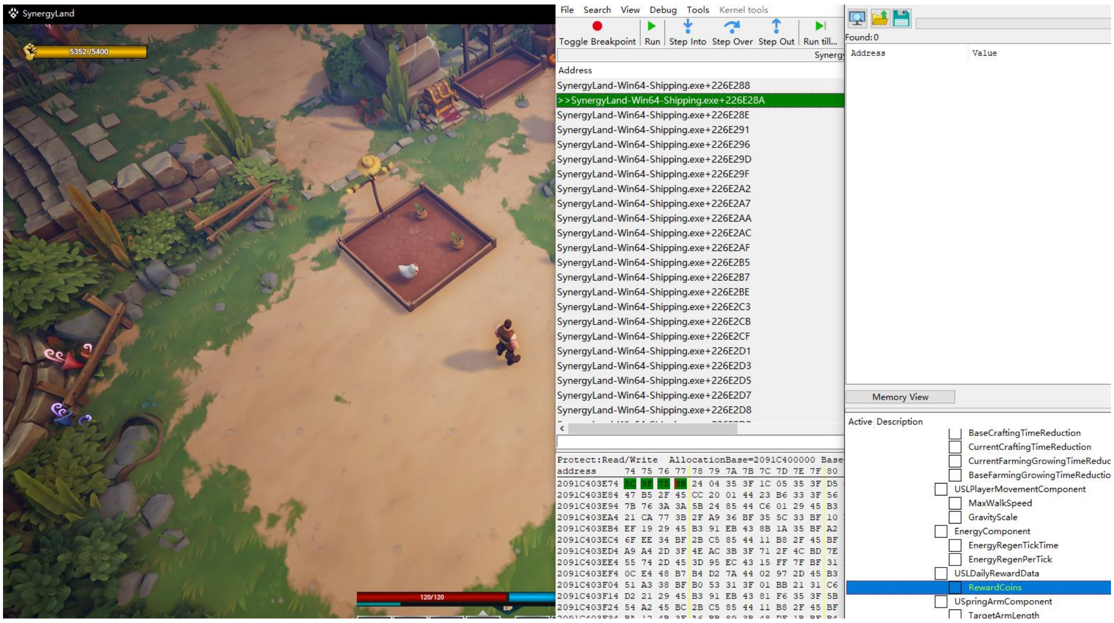
Conclusion: SynergyLand scored 0 points in terms of game code protection, as the client code and strings are not encrypted, making it easy for users to dump the game's data structures for quick analysis. However, many basic data are issued through synchronization framework, resulting in lower risk.

Fix Recommendations: Increase local encryption of code and local protection of strings.

Game Basic Anti-Cheat:

Analysis Process:

1. In the basic anti-cheat detection, we mainly tested two aspects: whether the game has anti-debugging and whether it has read/write protection.
2. When using CE for memory viewing in the game's open state, it was found that the memory could not be scanned, indicating that EAC was effective.



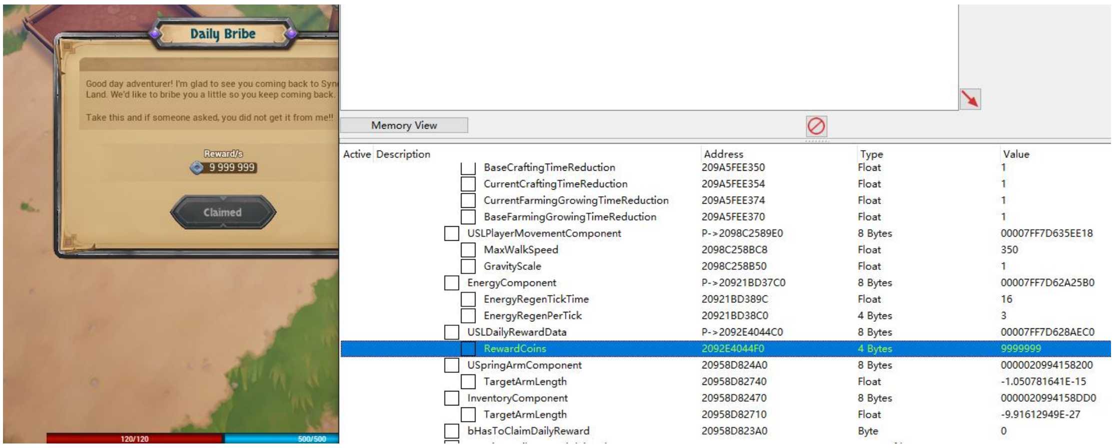
```
Address
SynergyLand-Win64-Shipping.exe+226E288
>>SynergyLand-Win64-Shipping.exe+226E28A
SynergyLand-Win64-Shipping.exe+226E28E
SynergyLand-Win64-Shipping.exe+226E291
SynergyLand-Win64-Shipping.exe+226E296
SynergyLand-Win64-Shipping.exe+226E29D
SynergyLand-Win64-Shipping.exe+226E29F
SynergyLand-Win64-Shipping.exe+226E2A2
SynergyLand-Win64-Shipping.exe+226E2A7
SynergyLand-Win64-Shipping.exe+226E2AA
SynergyLand-Win64-Shipping.exe+226E2AC
SynergyLand-Win64-Shipping.exe+226E2AF
SynergyLand-Win64-Shipping.exe+226E2B5
SynergyLand-Win64-Shipping.exe+226E2B7
SynergyLand-Win64-Shipping.exe+226E2BE
SynergyLand-Win64-Shipping.exe+226E2C3
SynergyLand-Win64-Shipping.exe+226E2CB
SynergyLand-Win64-Shipping.exe+226E2CF
SynergyLand-Win64-Shipping.exe+226E2D1
SynergyLand-Win64-Shipping.exe+226E2D3
SynergyLand-Win64-Shipping.exe+226E2D5
SynergyLand-Win64-Shipping.exe+226E2D7
SynergyLand-Win64-Shipping.exe+226E2D8
```

```
Protect:Read/Write  AllocationBase=2091C400000 Base
address     74 75 76 77 78 79 7A 7B 7C 7D 7E 7F 80
2091C403E74 5C 9F 75 BB 24 04 35 3F 1C 05 35 3F D5
2091C403E84 47 B5 2F 45 CC 20 01 44 23 B6 33 3F 56
2091C403E94 7B 76 3A 3A 5B 24 85 44 C6 01 29 45 B3
2091C403EA4 21 CA 77 3B 2F A9 36 BF 35 5C 33 BF 10
2091C403EB4 EF 19 29 45 B3 91 EB 43 8B 1A 35 BF A2
2091C403EC4 6F EE 34 BF 2B C5 85 44 11 B8 2F 45 BF
2091C403ED4 A9 A4 2D 3F 4E AC 3B 3F 71 2F 4C BD 7E
2091C403EE4 55 74 2D 45 3D 95 EC 43 15 FF 7F BF 31
2091C403EF4 0C E4 48 B7 B4 D2 7A 44 02 97 2D 45 B3
2091C403F04 51 A3 38 BF B0 53 31 3F 01 BB 21 31 C6
2091C403F14 D2 21 29 45 B3 91 EB 43 81 F6 35 3F 5B
2091C403F24 54 A2 45 BC 2B C5 85 44 11 B8 2F 45 BF
```

```
Found:0
Address                 Value
```

```
Active Description
        □ BaseCraftingTimeReduction
        □ CurrentCraftingTimeReduction
        □ CurrentFarmingGrowingTimeReduc
        □ BaseFarmingGrowingTimeReductio
      □ USLPlayerMovementComponent
        □ MaxWalkSpeed
        □ GravityScale
      □ EnergyComponent
        □ EnergyRegenTickTime
        □ EnergyRegenPerTick
      □ USLDailyRewardData
        □ RewardCoins
      □ USpringArmComponent
        □ TargetArmLength
```

After attaching using a custom CE, it was found that reading and writing to memory was possible. This, combined with dumped structures and analysis in IDA, was done.



Daily Bribe

Good day adventurer! I'm glad to see you coming back to Syn
Land. We'd like to bribe you a little so you keep coming back.

Take this and if someone asked, you did not get it from me!!

Reward/s

9 999 999

Active	Description	Address	Type	Value
	□ BaseCraftingTimeReduction	209A5FEE350	Float	1
	□ CurrentCraftingTimeReduction	209A5FEE354	Float	1
	□ CurrentFarmingGrowingTimeReduction	209A5FEE374	Float	1
	□ BaseFarmingGrowingTimeReduction	209A5FEE370	Float	1
	□ USLPlayerMovementComponent	P->2098C2589E0	8 Bytes	00007FF7D635EE18
	□ MaxWalkSpeed	2098C258BC8	Float	350
	□ GravityScale	2098C258B50	Float	1
	□ EnergyComponent	P->20921BD37C0	8 Bytes	00007FF7D62A25B0
	□ EnergyRegenTickTime	20921BD389C	Float	16
	□ EnergyRegenPerTick	20921BD38C0	4 Bytes	3
	□ USLDailyRewardData	P->2092E4044C0	8 Bytes	00007FF7D628AEC0
	□ RewardCoins	2092E4044F0	4 Bytes	9999999
	□ USpringArmComponent	20958D824A0	8 Bytes	0000020994158200
	□ TargetArmLength	20958D82740	Float	-1.050781641E-15
	□ InventoryComponent	20958D82470	8 Bytes	0000020994158DD0
	□ TargetArmLength	20958D82710	Float	-9.91612949E-27
	□ bHasToClaimDailyReward	20958D823A0	Byte	0

Analysis Conclusion：

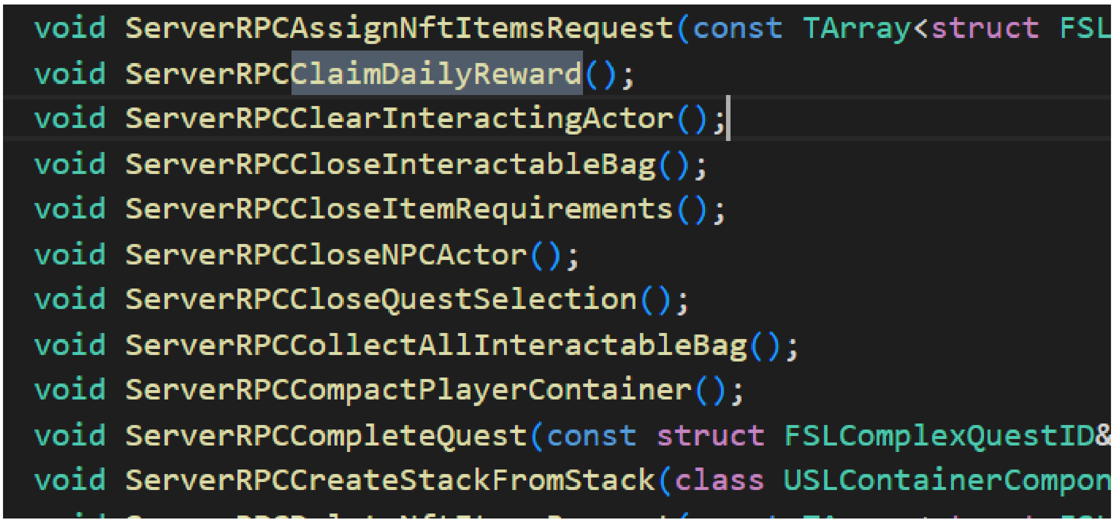
1. SynergyLand has a basic protection score of 0 in anti-cheat measures. Currently, the game lacks any anti-cheat measures, allowing players to freely read and manipulate in-game data. However, due to the game's robust synchronization and relatively simple logic, most data processing logic is handled on the server, resulting in a stable security level..
2. Testing focused on anti-debugging and read/write protection due to their critical role in cheat functionalities. Lack of these fundamental protections renders additional detections like injections and hooks ineffective.

Fix Recommendations: Implement additional features and include sensitive data in the synchronization framework.

Game Protocol & Logic Security Analysis

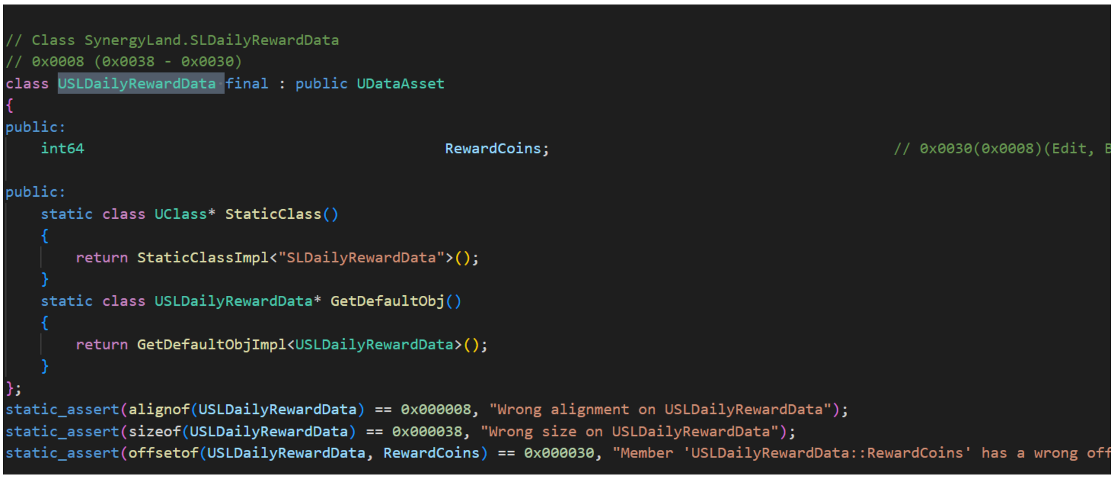
Analysis Process:

1. Through analysis of the game's structures and code logic, it was found that some logic uses ServerRPC method and does not store attributes or undergo synchronization. Taking ClaimDailyReward as an example, this function is called when claiming rewards.



```
void ServerRPCAssignNftItemsRequest(const TArray<struct FSL
void ServerRPCClaimDailyReward();
void ServerRPCClearInteractingActor();
void ServerRPCCloseInteractableBag();
void ServerRPCCloseItemRequirements();
void ServerRPCCloseNPCActor();
void ServerRPCCloseQuestSelection();
void ServerRPCCollectAllInteractableBag();
void ServerRPCCompactPlayerContainer();
void ServerRPCCompleteQuest(const struct FSLComplexQuestID&
void ServerRPCCreateStackFromStack(class USLContainerCompon
```

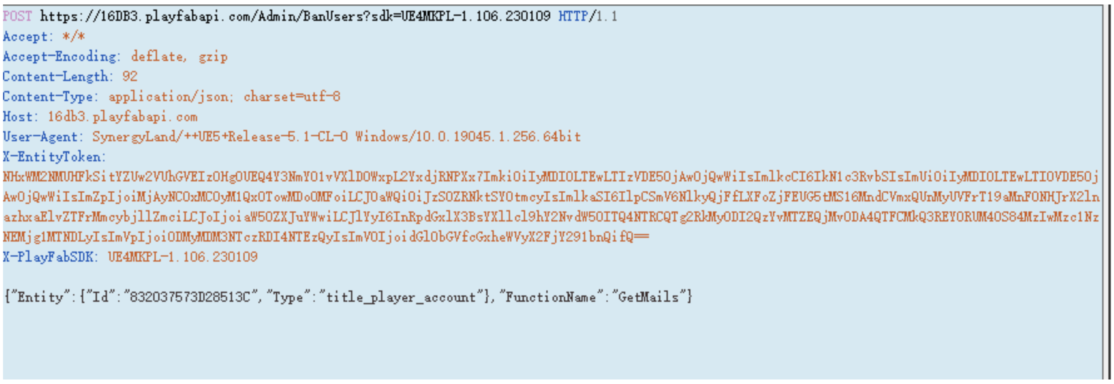
The number of coins is mainly controlled by this field.



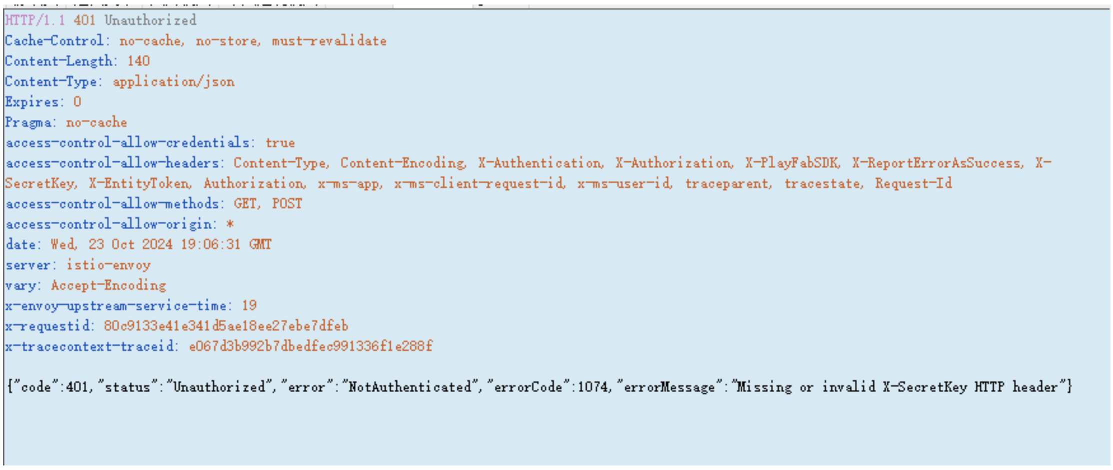
```
// Class SynergyLand.SLDailyRewardData
// 0x0008 (0x0038 - 0x0030)
class USLDailyRewardData final : public UDataAsset
{
public:
    int64                                        RewardCoins;                                       // 0x0030(0x0008)(Edit, E

public:
    static class UClass* StaticClass()
    {
        return StaticClassImpl<"SLDailyRewardData">();
    }
    static class USLDailyRewardData* GetDefaultObj()
    {
        return GetDefaultObjImpl<USLDailyRewardData>();
    }
};
static_assert(alignof(USLDailyRewardData) == 0x000008, "Wrong alignment on USLDailyRewardData");
static_assert(sizeof(USLDailyRewardData) == 0x000038, "Wrong size on USLDailyRewardData");
static_assert(offsetof(USLDailyRewardData, RewardCoins) == 0x000030, "Member 'USLDailyRewardData::RewardCoins' has a wrong off
```

This field appears to not be synchronized by the server. There seems to be a risk of tampering, but due to time constraints, further analysis was not possible. It is recommended that the project team strengthen judgment on similar attributes

2. The game uses Azure PlayFab Game Server solution as the game configuration server. Since PlayFab relies on RestAPI, the project team must securely store the APP:title permission account to avoid misuse or risks such as game data deletion due to leaked Secret Key



```
POST https://16DB3.playfabapi.com/Admin/BanUsers?sdk=UE4MKPL-1.106.230109 HTTP/1.1
Accept: */*
Accept-Encoding: deflate, gzip
Content-Length: 92
Content-Type: application/json; charset=utf-8
Host: 16db3.playfabapi.com
User-Agent: SynergyLand/++UE5+Release-5.1-CL-0 Windows/10.0.19045.1.256.64bit
X-EntityToken:
NHxWM2NMUHFkSitYZUw2VUhGVEIzOHg0UEQ4Y3NmY01vVXlDOWxpL2YxdjRNPXx7ImkiOiIyMDI0LTEwLTIzVDE5OjAwOjQwWiIsImlkcCI6IkN1c3RvbSIsImUiOiIyMDI0LTEwLTI0VDE5Oj
AwOjQwWiIsImZpIjoiMjAyNC0xMC0yM1QxOTowMDowMFoiLCJ0aWQiOiJzS0ZRNktSY0tmcyIsImlkaSI6IlpCSmV6NlkyQjFfLXFoZjFEUG5tMS16MndCVmxQUnMyUVFrT19aMnF0NHJrX2ln
azhxaElvZTFrMmcybjllZmciLCJoIjoiaW50ZXJuYWwiLCJlYyI6InRpdGxlX3BsYXllcl9hY2NvdW50ITQ4NTRCQTg2RkMyODI2QzYvMTZEQjMvODA4QTFCMkQ3REY0RUM4OS84MzIwMzc1Nz
NEMjg1MTNDLyIsImVpIjoiODMyMDM3NTczRDI4NTEzQyIsImV0IjoidGl0bGVfcGxheWVyX2FjY291bnQifQ==
X-PlayFabSDK: UE4MKPL-1.106.230109

{"Entity":{"Id":"832037573D28513C","Type":"title_player_account"},"FunctionName":"GetMails"}
```



```
HTTP/1.1 401 Unauthorized
Cache-Control: no-cache, no-store, must-revalidate
Content-Length: 140
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, 23 Oct 2024 19:06:31 GMT
server: istio-envoy
vary: Accept-Encoding
x-envoy-upstream-service-time: 19
x-requestid: 80c9133e41e341d5ae18ee27ebe7dfeb
x-tracecontext-traceid: e067d3b992b7dbedfec991336f1e288f

{"code":401,"status":"Unauthorized","error":"NotAuthenticated","errorCode":1074,"errorMessage":"Missing or invalid X-SecretKey HTTP header"}
```

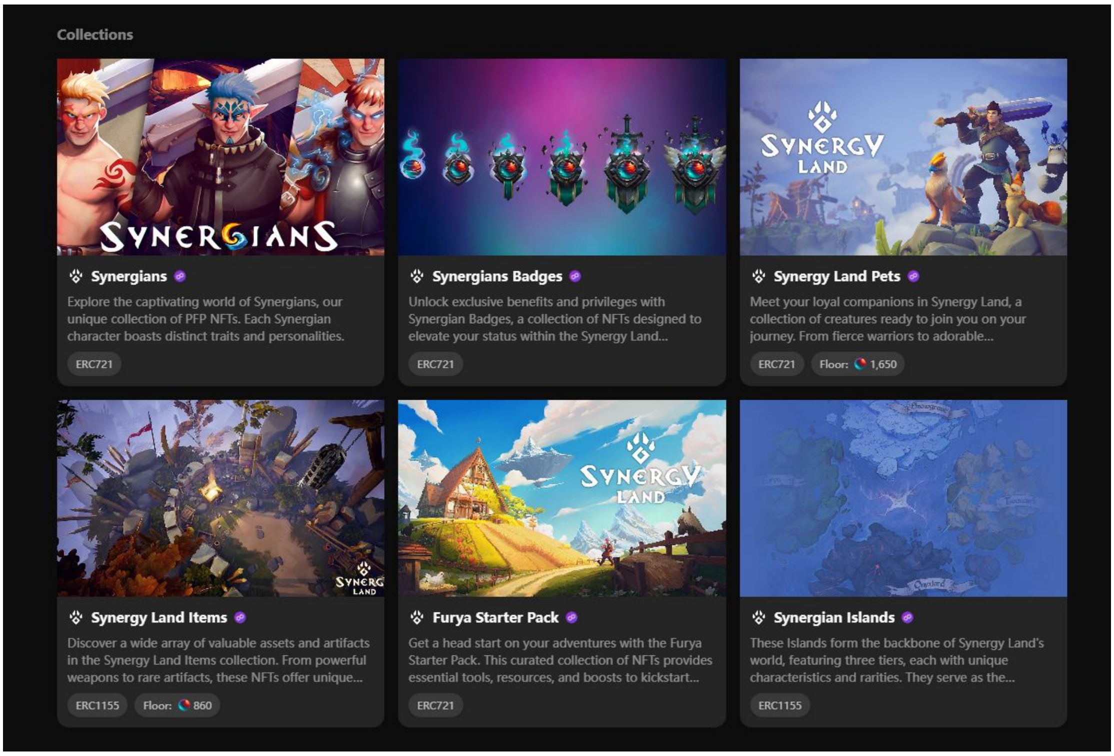
Analysis Conclusion：

1. Due to the short testing period, a detailed analysis of protocols and blueprint functions was not possible. The above conclusions are mainly based on logical deduction. It is recommended that the project team conduct thorough testing and control of the protocol part when the game goes live.

Fix Recommendations: Increase synchronization of sensitive data, encrypt script interactions, and rigorously test the protocol section.

WEB3 Security Analysis：

SynergyLand has released six NFTs including characters and land. The six contract codes are similar, using a proxy contract + royalty NFT contract structure..



Collections

Synergians

Explore the captivating world of Synergians, our unique collection of PFP NFTs. Each Synergian character boasts distinct traits and personalities.

ERC721

Synergians Badges

Unlock exclusive benefits and privileges with Synergian Badges, a collection of NFTs designed to elevate your status within the Synergy Land...

ERC721

Synergy Land Pets

Meet your loyal companions in Synergy Land, a collection of creatures ready to join you on your journey. From fierce warriors to adorable...

ERC721 Floor: 1,650

Synergy Land Items

Discover a wide array of valuable assets and artifacts in the Synergy Land Items collection. From powerful weapons to rare artifacts, these NFTs offer unique...

ERC1155 Floor: 860

Furya Starter Pack

Get a head start on your adventures with the Furya Starter Pack. This curated collection of NFTs provides essential tools, resources, and boosts to kickstart...

ERC721

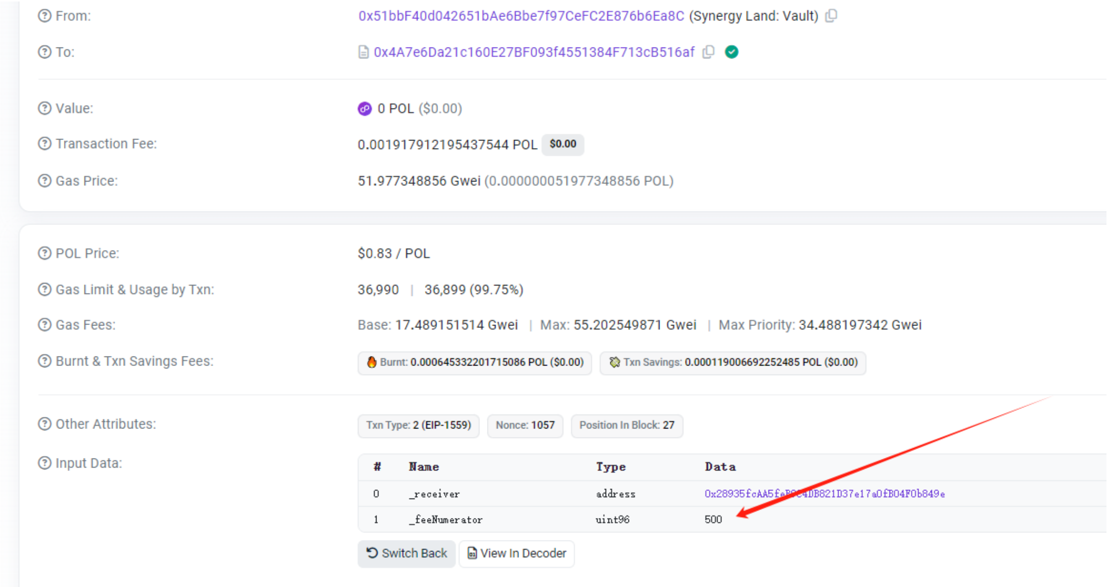
Synergian Islands

These Islands form the backbone of Synergy Land's world, featuring three tiers, each with unique characteristics and rarities. They serve as the...

ERC1155

The ERC1155 contract for Land is primarily analyzed.

- **Circulation: 500**
- **Royalty: 5%**



From: 0x51bbF40d042651bAe6Bbe7f97CeFC2E876b6Ea8C (Synergy Land: Vault)

To: 0x4A7e6Da21c160E27BF093f4551384F713cB516af

Value: 0 POL ($0.00)

Transaction Fee: 0.001917912195437544 POL $0.00

Gas Price: 51.977348856 Gwei (0.000000051977348856 POL)

POL Price: $0.83 / POL

Gas Limit & Usage by Txn: 36,990 | 36,899 (99.75%)

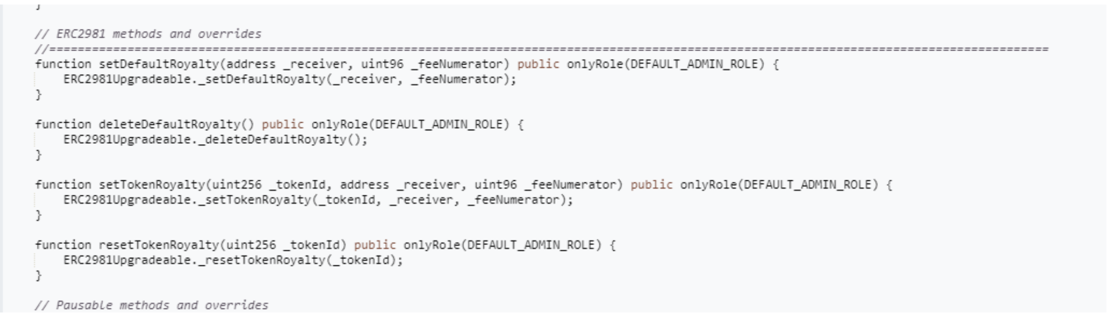
Gas Fees: Base: 17.489151514 Gwei | Max: 55.202549871 Gwei | Max Priority: 34.488197342 Gwei

Burnt & Txn Savings Fees: Burnt: 0.000645332201715086 POL ($0.00) Txn Savings: 0.000119006692252485 POL ($0.00)

Other Attributes: Txn Type: 2 (EIP-1559) Nonce: 1057 Position In Block: 27

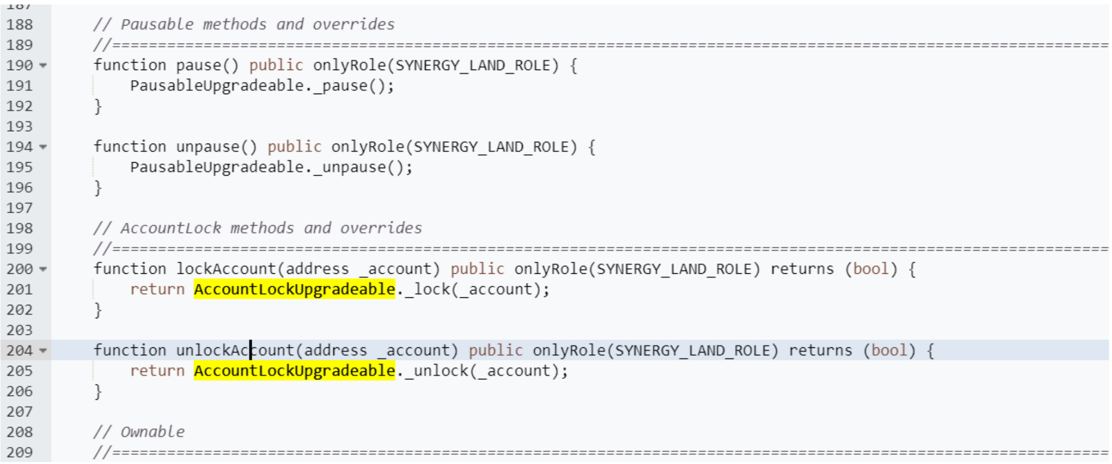
Input Data:

#	Name	Type	Data
0	_receiver	address	0x28935fcAA5feB0C4DB821D37e17a0fB04F0b849e
1	_feeNumerator	uint96	500



```
}

// ERC2981 methods and overrides
//==============================================================================================================================
function setDefaultRoyalty(address _receiver, uint96 _feeNumerator) public onlyRole(DEFAULT_ADMIN_ROLE) {
    ERC2981Upgradeable._setDefaultRoyalty(_receiver, _feeNumerator);
}

function deleteDefaultRoyalty() public onlyRole(DEFAULT_ADMIN_ROLE) {
    ERC2981Upgradeable._deleteDefaultRoyalty();
}

function setTokenRoyalty(uint256 _tokenId, address _receiver, uint96 _feeNumerator) public onlyRole(DEFAULT_ADMIN_ROLE) {
    ERC2981Upgradeable._setTokenRoyalty(_tokenId, _receiver, _feeNumerator);
}

function resetTokenRoyalty(uint256 _tokenId) public onlyRole(DEFAULT_ADMIN_ROLE) {
    ERC2981Upgradeable._resetTokenRoyalty(_tokenId);
}

// Pausable methods and overrides
```

Addresses with SYNERGY_LAND_ROLE permission can perform the Lock operation on any account. Wallets that are locked cannot transfer or destroy NFTs.



```
    // Pausable methods and overrides
    //=====================================================================================================
    function pause() public onlyRole(SYNERGY_LAND_ROLE) {
        PausableUpgradeable._pause();
    }

    function unpause() public onlyRole(SYNERGY_LAND_ROLE) {
        PausableUpgradeable._unpause();
    }

    // AccountLock methods and overrides
    //=====================================================================================================
    function lockAccount(address _account) public onlyRole(SYNERGY_LAND_ROLE) returns (bool) {
        return AccountLockUpgradeable._lock(_account);
    }

    function unlockAccount(address _account) public onlyRole(SYNERGY_LAND_ROLE) returns (bool) {
        return AccountLockUpgradeable._unlock(_account);
    }

    // Ownable
    //=====================================================================================================
```

This operation may be related to the project's market-making and prevention of user misconduct in the future.

Currently, there are no obvious issues with the contracts. It is recommended that the project team upgrade the permission wallet addresses to multisig and enhance permission control when officially operating in the future.

Damocles

About Damocles

Damocles Labs is a security team established in 2023, specializing in security for the Web3 industry. Their services include contract code auditing, business code auditing, penetration testing, GameFi code auditing, GameFi vulnerability discovery, GameFi cheat analysis, and GameFi anti-cheat measures. They are committed to making continuous efforts in the Web3 security industry, producing as many analysis reports as possible, raising awareness among project owners and users about GameFi security, and promoting the overall security development of the industry.。

Twitter: https://twitter.com/DamoclesLabs

WebSite：http://damocleslabs.com/

Analysis Report repo：https://github.com/DamoclesLabs/GameFi-Analysis-Report/