ALIBABA CLOUD

Alibaba Cloud

游戏盾

快速入门

文档版本：20220601

阿里云

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。

2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。

3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。

4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。

5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。

6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	危险 重置操作将丢失用户配置数据。
警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	警告 重启操作将导致业务中断，恢复业务时间约十分钟。
注意	用于警示信息、补充说明等，是用户必须了解的内容。	注意 权重设置为0，该服务器不会再接受新请求。
说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击**设置**> **网络**> **设置网络类型**。
粗体	表示按键、菜单、页面名称等UI元素。	在**结果确认**页面，单击**确定**。
Courier字体	命令或代码。	执行 `cd /d C:/window` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *Instance_ID*
[] 或者 [a|b]	表示可选项，至多选择一个。	`ipconfig [-all|-t]`
{} 或者 {a|b}	表示必选项，至多选择一个。	`switch {active|stand}`

目录

1.步骤1：创建游戏实例

在接入游戏盾前，您必须在游戏盾控制台创建游戏实例（即为要接入的游戏创建游戏盾配置）。

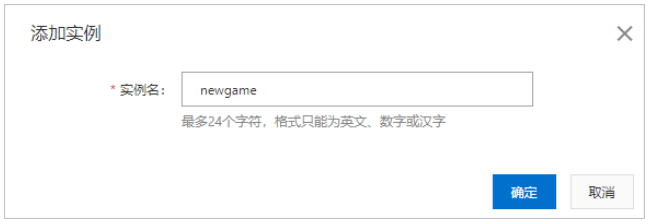
前提条件

已开通游戏盾且拥有剩余的实例授权数。更多信息，请参见计费方式。

操作步骤

1. 登录云盾游戏盾管理控制台。
2. 在**首页**，单击**创建实例**。
3. 在**添加实例**对话框，输入**实例名**，并单击**确定**。

   > **说明** 实例名最多包含24个字符，支持使用英文、数字或汉字。

   游戏盾将调用自动创建脚本，自动完成以下任务，为您生成默认游戏盾配置：**检查资源**、**创建游戏**、**创建业务**、**配置无限抗**。

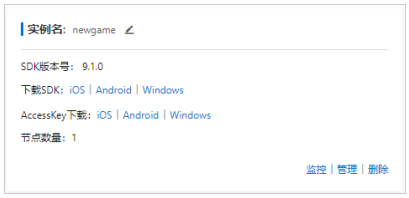
   默认游戏盾配置中包含已创建的游戏、业务，且业务下包含一个节点组和抗D节点。默认已开启无限抗。

4. 等待游戏实例自动创建和配置完成后，在**提示**对话框，单击**确认**。

执行结果

成功创建游戏实例后，您可以在**首页**查看新建的实例。



后续步骤

步骤2：配置防护目标

2.步骤2：配置防护目标

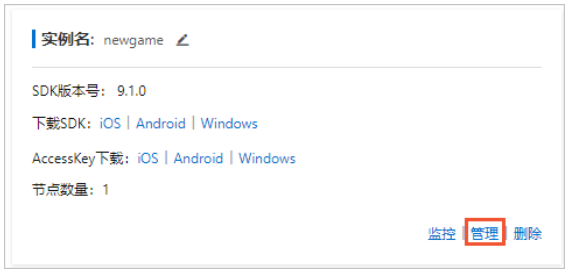
创建游戏实例后，您需要为实例配置防护目标。

前提条件

已创建游戏实例。相关操作，请参见步骤1：创建游戏实例。

操作步骤

1. 登录云盾游戏盾管理控制台。
2. 在**首页**，定位到要配置的实例，在操作选项中单击**管理**。



3. 在**防护目标配置**页签，单击**防护目标管理**，然后选择**添加防护目标**。

4. 在**新建防护目标**对话框，完成以下配置，并单击**确定**。

配置项	说明

配置项	说明
防护目标标识	为防护目标设置一个标识。 支持使用英文、数字、半角句号（.）、下划线（_）、短划线（-），且最多包含128个字符。 SDK接入时，该标识是本地服务转化接口的重要参数之一。
备注	设置备注信息。 支持使用英文、数字、半角句号（.）、下划线（_）、短划线（-），且最多包含128个字符。
协议支持	目前仅支持**TCP**，无需修改。
主线路IP	添加游戏的主线路IP地址。 最多可填写20个IP，多个IP间以半角逗号（,）分隔。 主线路IP始终在游戏盾网络的防护之下，不会被暴露给客户端；客户端获取的是游戏盾的防护节点，具备无限抗能力。
备线路IP	添加游戏的备线路IP地址。 最多可填写20个IP，多个IP间以半角逗号（,）分隔。 备线路IP会直接返回并暴露给客户端，没有任何安全防护能力。

执行结果

成功添加防护目标后，您可以在**防护目标配置**页签查看新建的防护目标。已创建的防护目标支持**编辑**和**删除**操作。

后续步骤

步骤3：完成业务接入

3.步骤3：完成业务接入

游戏盾支持SDK接入，该模式在DDoS攻击时可实现秒级切换，且具备CC攻击免疫能力。本文介绍了使用SDK接入的概述。

SDK接入概述

SDK接入后，实现秒级调度，且具备链路探测、智能化调度的能力。DDoS攻击时实现秒级切换，具备主动隔离恶意终端设备的能力。CC攻击协议级加密，零误杀、零漏过，免疫CC攻击。

使用SDK接入时，您必须先获取SDK和AccessKey（具体请参见获取SDK和AccessKey），然后根据您的接入平台类型，完成相应的SDK接入配置，具体请参见以下不同操作系统的接入说明文档：

- Android接入
- iOS接入
- Windows C++接入
- Windows Python接入
- Unity接入

完成SDK接入后，您将可以通过SDK获得游戏盾转化的IP和端口。针对不同的业务类型，使用游戏盾转化IP和端口的方法也有区别，请分别参见以下文档：

- TCP（游戏登录、游戏服务器等）
- HTTP/S（API类、登录、数据获取等）
- HTTP/S（B/S-网页充值、客服、后台页面等）

4.SDK接入

4.1. 获取SDK和AccessKey

在游戏盾中创建游戏后，您就可以在游戏盾控制台直接获取正式版本的SDK包和AccessKey。游戏盾控制台提供适用于不同操作系统（例如iOS、Android、Windows）的SDK包和AccessKey。

前提条件

已创建游戏实例。相关操作，请参见步骤1：创建游戏实例。

操作步骤

1. 登录云盾游戏盾管理控制台。
2. 在**首页**，定位到已添加的游戏，根据操作系统类型（**iOS**、**Android**、**Windows**），单击**下载SDK**选项下的链接，直接下载对应操作系统的SDK包。

   例如单击**下载SDK**选项下的**iOS**，直接下载iOS SDK包。

> **说明**
>
> - 通过游戏盾控制台下载的版本即为正式版本。
> - 5.2.5版本以后的游戏盾SDK均通过SDK加密隧道方式接入，解决协议模拟型CC攻击。

3. 定位到已添加的游戏，根据操作系统类型（**iOS**、**Android**、**Windows**），单击**AccessKey下载**选项下的链接，直接下载对应操作系统的AccessKey。

   例如单击**AccessKey下载**选项下的**iOS**，直接下载iOS AccessKey。

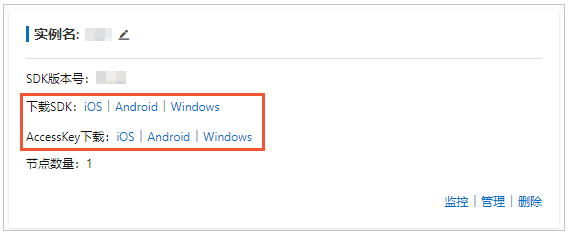
4.2. 平台接入配置

4.2.1. Android接入

本文介绍了使用Android Studio接入游戏盾SDK的操作方法。

前提条件

已通过游戏盾控制台获取Android版本的SDK包和AccessKey（即Appkey）。具体操作，请参见获取SDK和AccessKey。



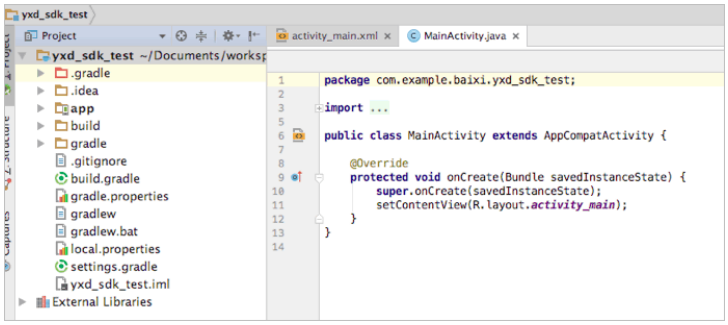
背景信息

核心接口函数包括：initEx、getProxyTcpDomain。更多介绍，请参见核心函数说明。

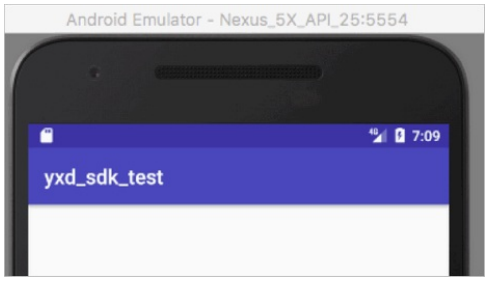
您可以联系游戏盾售后技术支持获取Demo程序。

操作步骤

1. 运行Android Studio。
2. 新建一个工程，并使用默认配置完成创建向导。示例中将工程命名为*yxd_sdk_test*。
   新建工程的目录结构如下图所示。

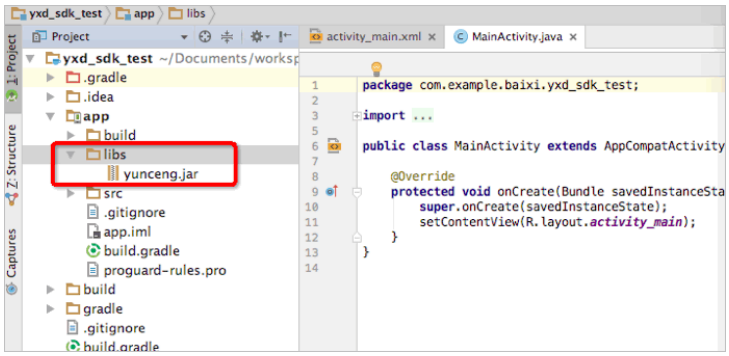


> **说明** 进行后续操作前，请确认新建工程可以正常运行。

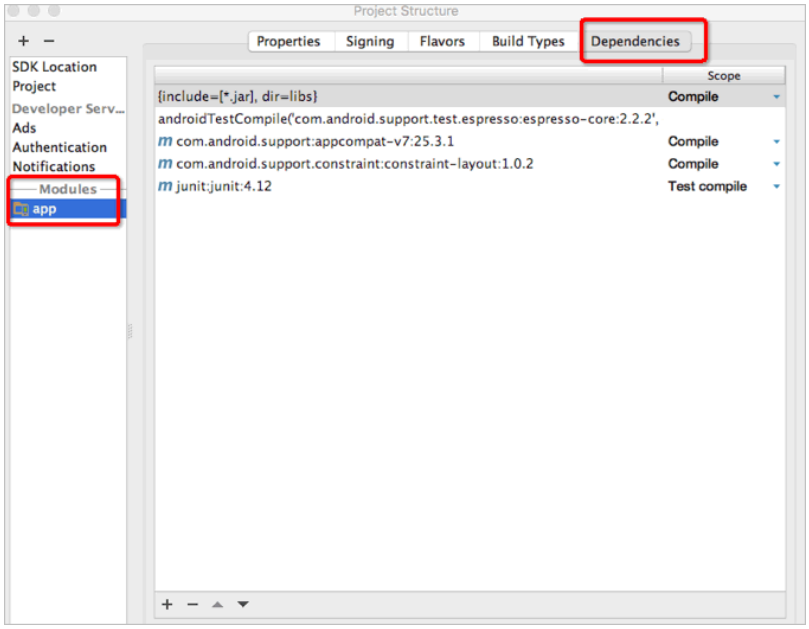


3. 添加JAR。

i. 将Android SDK包中的*yunceng.jar*文件复制到项目的**libs**目录下。

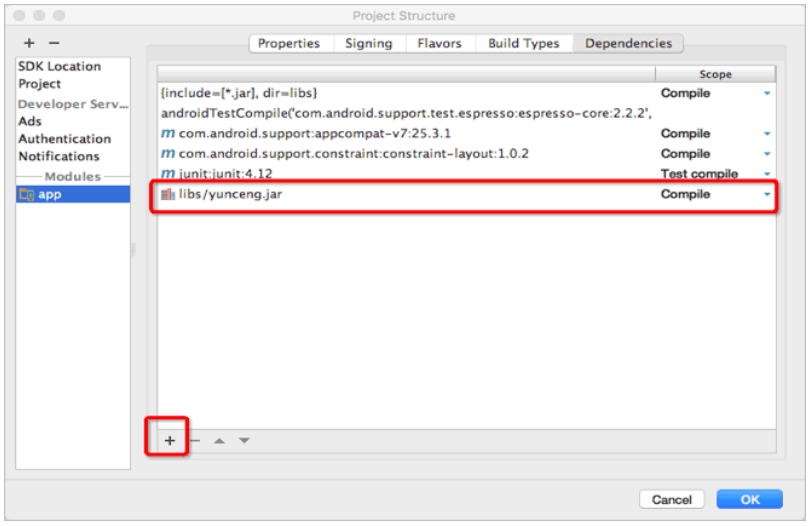


ii. 在Android Studio工程选项中，选择**File > Project Structure**，单击**app**组件，并打开**Dependencies**页签。



iii. 单击

图标，选择**jar dependency**，并添加**yunceng.jar**。



iv. 单击OK，完成添加。

4. 添加SO文件。进入src > main目录，在该目录下新建jniLibs目录，并将Android SDK包中的SO文件复制到jniLibs目录。

5. 设置访问权限。打开*AndroidManifest.xml*文件，参照下图在其中添加以下代码。

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
```

6. 初始化SDK。

> **说明** 只需成功初始化一次即可。

使用以下代码前，必须替换以下变量：

- 将 `<yourAppKey>` 替换成您通过游戏盾控制台获取的AccessKey。
- 将 `<token>` 替换成游戏内玩家的唯一ID。如果无法获取该ID，可以设置为一个默认值，但是不允许为空。

```
public int sdk_init() {
    String appkey = "<yourAppKey>";
    String token = "<token>";
    int ret = YunCeng.initEx(appkey, game);
    if (ret == 0) {
        Log.d("yuncengsdk", "sdk init success");
    } else {
        Log.e("yuncengsdk", "sdk init failed, ret " + ret);
    }
    return ret;
}
```

7. （可选）配置ProGuard。

如果您使用ProGuard进行混淆，则需要在ProGuard配置文件中添加以下内容：

```
-keep class com.aliyun.security.yunceng.** {*;}
```

后续步骤

完成SDK接入后，您将可以通过SDK获得游戏盾转化的IP和端口。针对不同的业务类型，使用游戏盾转化IP和端口的方法也有区别，请分别参见以下文档：

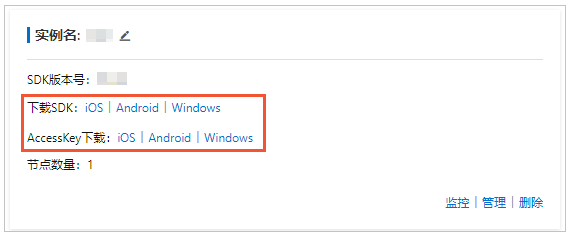
- TCP（游戏登录、游戏服务器等）
- HTTP/S（API类、登录、数据获取等）
- HTTP/S（B/S-网页充值、客服、后台页面等）

4.2.2. iOS接入

本文介绍了使用XCode接入游戏盾SDK的操作方法。

前提条件

已通过游戏盾控制台获取iOS版本的SDK包和AccessKey（即Appkey）。具体操作，请参见获取SDK和AccessKey。

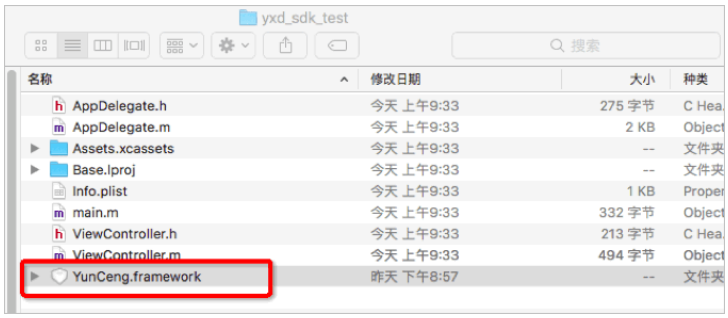


操作步骤

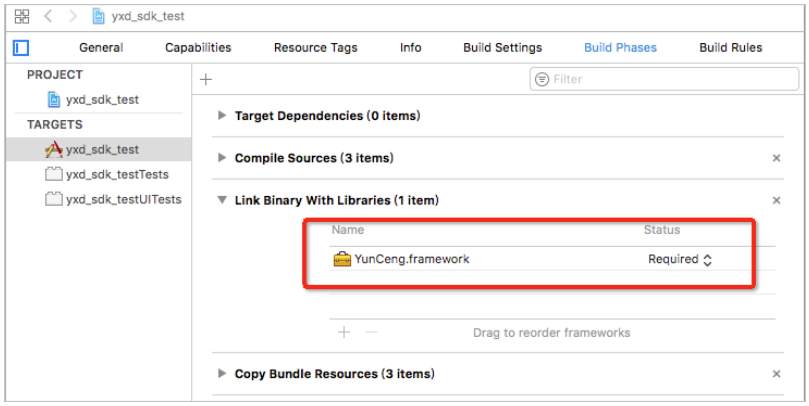
1. 运行XCode。
2. 新建一个工程，选择Single View Application，并按照默认配置完成创建向导。示例中将工程命名为*yxd_sdk_test*。

   > **说明** 进行后续操作前，请确认新建工程可以正常运行。

3. 添加库依赖。将iOS SDK包中的*YunCeng.framework*复制到新建工程（示例中是*yxd_sdk_test*）的目录下。

4. 修改以下工程配置：
   - 将Build Phases修改为Link Binary With Libraries。
   - 添加*YunCeng.framework*及其他依赖framework。



5. 初始化SDK。

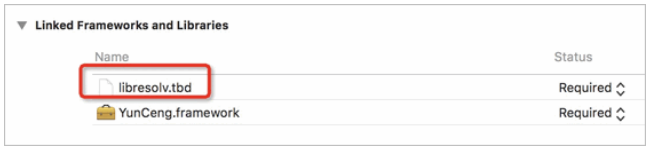
   使用以下代码前，必须替换以下变量：

   - 将 `<yourAppKey>` 替换成您通过游戏盾控制台获取的AccessKey。
   - 将 `<token>` 替换成游戏内玩家的唯一ID。如果无法获取该ID，可以设置为一个默认值，但是不允许为空。

```
- (int) SDKInit {
    const char * appkey = "<yourAppKey>";
    const char * token = "<token>";
    int ret = YunCeng_InitEx(appkey, token);
    if (ret == 0) {
        NSLog(@"sdk init success");
    } else {
        NSLog(@"sdk init failed, ret %d", ret);
    }
    return ret;
}
```

   故障排查：

   - 如果编译时出错并提示 "_OBJC_CLASS_$_CTTelephonyNetworkInfo", referenced from，请参见步骤4添加*CoreTelephony.framework*库。
   - 如果编译时出错并提示 "_res_9_getservers", referenced from，请参见步骤4添加*libresolv.tbd*库。



后续步骤

完成SDK接入后，您将可以通过SDK获得游戏盾转化的IP和端口。针对不同的业务类型，使用游戏盾转化IP和端口的方法也有区别，请分别参见以下文档：

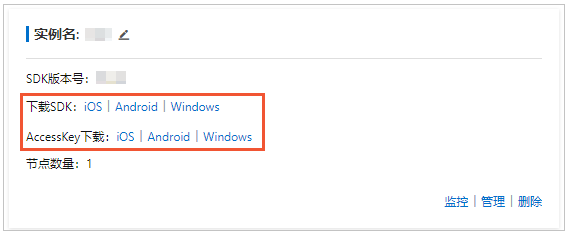
- TCP（游戏登录、游戏服务器等）
- HTTP/S（API类、登录、数据获取等）
- HTTP/S（B/S-网页充值、客服、后台页面等）

4.2.3. Windows C++接入

本文介绍了通过Windows C++接入游戏盾SDK的操作方法。

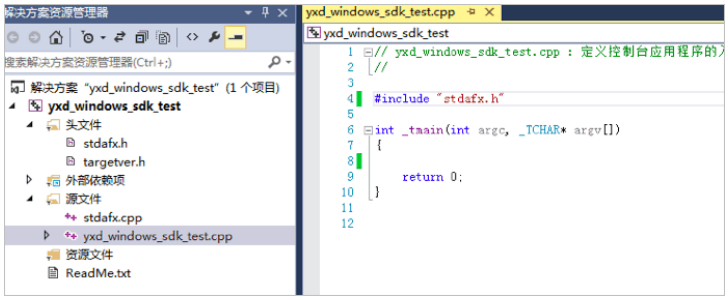
前提条件

已通过游戏盾控制台获取Windows版本的SDK包和AccessKey（即Appkey）。具体操作，请参见获取SDK和AccessKey。



操作步骤

1. 在Windows上运行C++编译器。
2. 新建一个console项目。示例中将项目命名为*yxd_windows_sdk_test*。



3. 配置Lib依赖。
   i. 在新建项目的目录下，新建*libs*目录。
   ii. 将游戏盾Windows SDK中的*YunCeng-WINDOWS.lib*文件复制并粘贴到*libs*目录下。
   iii. 打开项目的属性页，在**链接器 > 常规**中，添加**./libs**作为**附加库目录**。
   iv. 在**链接器 > 输入**中，添加**YunCeng.WINDOWS.lib**作为**附加依赖项**。
4. 加入头文件，编写测试代码。

```
char appkey[] = "testgroupid";
eAlSdkRet ret = YunCeng_InitEx(appkey, "token");
if (ret != cAlSdkOK) {
        printf("init sdk failed.\n");
        return -1;
    }
```

5. 将*YunCeng-WINDOWS.dll*移动到项目可执行文件（示例中是*yxd_windows_sdk_test.exe*）所在目录。

6. 测试是否可以获取IP。

```
/*
 ret = YunCeng_GetProxyTcpByIp("token","groupId", "192.168.0.1","80", ip,ip_len,  port,
port_len);  */
ret= YunCeng_GetProxyTcpByDomain("token","groupId", "aliyundoc.com","80", ip, ip_len, p
ort,port_len);
if (ret != cAlSdkOK) {
    printf("get next ip failed. \n");
   } else {
    printf("get next ip success. %s %s\n", ip, port);
   }
```

4.2.4. Windows Python接入

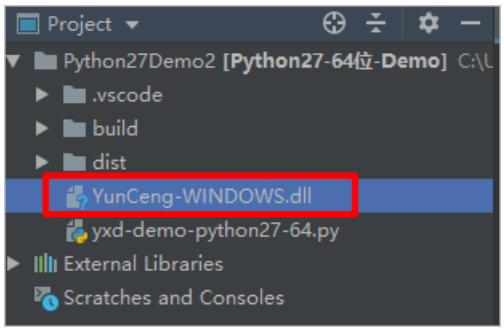
本文以PyCharm为例介绍在Windows操作系统中使用Python接入游戏盾SDK的操作方法。

前提条件

- 从游戏盾控制台获取Windows版本的SDK包和AccessKey（即Appkey）。更多信息，请参见获取SDK和AccessKey。
-

操作步骤

1. 运行Python编辑器。示例中运行PyCharm。
2. 新建一个Python 64位环境工程。
3. 将游戏盾Windows SDK包中的*YunCeng-WINDOWS.dll*文件复制到程序目录下。



4. 在代码中添加以下内容，载入*YunCeng-WINDOWS.dll*。

```
g_Dll = cdll.LoadLibrary("YunCeng-WINDOWS.dll")
```

5. 测试是否可以获取IP。

```
Init_result = g_Dll.YunCeng_InitEx(access_key,token)//初始化
#返回值
ip_len=18
ip = create_string_buffer('/0'*ip_len)
port_len=18
port = create_string_buffer('/0'*port_len)
ret=g_Dll.YunCeng_GetProxyTcpByDomain("玩家ID","GroupName","防护目标标识","防护目标端口", i
p, ip_len, port,port_len);
if  ret == 0 ://获取成功
```

4.2.5. Unity接入

游戏盾提供Unity C#接入头文件，请联系售后技术支持获取。

4.3. 业务接入配置

4.3.1. TCP（游戏登录、游戏服务器等）

TCP业务接入在整个协议接入中最容易实现，无需特殊处理，直接使用游戏盾转化后的IP和端口进行通信即可。

具体请参见以下示例。

```
Socket socket = new Socket("127.0.0.1", target_port.toString());
//tcp socket长连接
//target_port是游戏盾返回的本地随机端口
```

4.3.2. HTTP/S（API类、登录、数据获取等）

本文介绍了使用游戏盾SDK接入HTTP/S API类、登录、数据获取等业务的相关内容。

运维端配置防护目标

运维人员在游戏盾控制台添加防护目标时，需要将HTTP/S类业务的**防护目标标识**设置为一个（新分配的）标准域名，且该域名必须解析到127.0.0.1。参见下图示例。

关于添加防护目标的操作，请参见步骤2：配置防护目标。

客户端调用SDK

游戏盾转化后的127.0.0.1和8901（随机端口）可以拼接成HTTP格式，提供给HTTP业务使用，例如 `http://127.0.0.1:8910` 。

HTTPS协议下，则必须将127.0.0.1替换成一个有证书的标准域名，且域名必须解析到127.0.0.1，例如 `http://login-for-yxd.aliyundoc.com:8910/login-for-yxd.aliyundoc.com` 。使用这种方式访问能够解决主机头匹配、证书校验等问题。

示例代码

```
String url = 'https://'+"login-for-yxd.aliyundoc.com"+':'+target_port.toString();//http(s) url拼接请求短连接
```

其中， `target_port` 返回本地随机端口。

更多信息

参见HTTPS业务接入最佳实践，进一步了解在游戏盾中接入HTTPS业务时需要执行的特殊处理。

4.3.3. HTTP/S（B/S-网页充值、客服、后台页面等）

关于使用游戏盾SDK接入HTTP/S B/S-网页充值、客服、后台页面登录业务的配置说明，请参见HTTP/S APl类业务文档。本文介绍了接入后的使用方法。

使用方法	特性	说明
直接使用游戏盾转化后的接口唤起浏览器访问	成本低、兼容性差	存在以下问题： 游戏盾的IP+Port在IOS环境下，一旦App被切换到后台，端口可能会快速不可用，充值类弱交互的可以勉强使用，但是后台类业务体验会非常差。 使用App内的WebView替换本地浏览器，可以解决此问题。具体请参见下一个方法。
集成腾讯的WebView框架，直接在游戏内调用，不跳转到系统的浏览器内	成本高、兼容性好	建议您自行检索接入方法。
（推荐）在WebView的基础上，给WebView设置本地代理，通过游戏盾将远端代理映射到本地，使全部WebView流量经过游戏盾	成本较高、效果最佳	此方式DNS会暴露原站，所以原站必须在阿里云-杭州区域，这个区域的原站即使暴露，游戏盾转发依然能够通畅。 建议您自行检索接入方法。

4.4. 获取客户端真实IP

4.4.1. 原理说明

本文介绍了将业务接入游戏盾防护后，如何获取客户端的真实IP。

背景介绍

游戏盾是FullNat代理模式，经过游戏盾的请求，其客户端的IP地址会变成游戏盾的IP地址。针对有获取客户端真实IP需求的游戏盾用户，本文提供了获取客户端IP的解决方案。

实现原理

游戏盾通过TCP协议的Option字段来携带和传递客户端的IP信息，俗称TOA。由于游戏盾的TOA协议格式属于游戏盾专有，服务器需要集成游戏盾提供的TOA模块才能获取客户端IP信息。游戏盾TOA模块支持通过应用层Hook的方式（无需修改代码）进行集成。

- Linux

  Linux服务器环境需要安装应用层Hook-Toa模块。更多信息，请参见Linux。

- Windows

  Windows服务提供了部分程序的应用层Hook-Toa模块。更多信息，请参见Windows。

部署架构支持情况

场景	支持的架构	不支持的架构
四层TCP协议获取客户端真实IP	• 游戏盾->阿里云服务器、非阿里云服务器 • 游戏盾->阿里云SLB-4层转发->阿里云服务器	游戏盾->非阿里云SLB-4层转发->服务器
七层HTTP、HTTPS协议获取客户端真实IP	• 游戏盾->阿里云服务器、非阿里云服务器 • 游戏盾->阿里云SLB-4层转发->阿里云服务器	• 游戏盾->阿里云SLB-7层转发（包含WAF、高防IP的7层接入）->阿里云服务器 • 游戏盾->非阿里云SLB-4、7层转发->服务器

说明 游戏盾是四层转发，不托管HTTPS证书，无法查看您HTTPS数据流中的数据信息。所以，七层协议获取客户端真实IP不是通过XFF字段获取的，是通过在服务端的TOA模块适配直接获取客户端真实IP。

4.4.2. Linux

本文介绍了在Linux服务器环境下获取客户端真实IP的操作方法。

集成应用层Hook-TOA模块

1. 执行*install.sh*，安装toa_server相关服务。
2. 携带*preload.so*启动服务器。假设服务器名为nginx，则启动命令如下。

   ```
   LD_PRELOAD=./preload.so ./nginx
   ```

   说明 您需要根据实际情况找到您程序的启动地方，增加如上命令参数进行服务的启动。

   nginx service接入流程

   i. 运行*install.sh*。

   ii. 确认 */usr/lib/systemd/system/nginx.service*文件是否存在。

iii. 更新*mynginx.sh*。

```
cat > /root/mynginx.sh
```

iv. 将以下内容保存到*mynginx.sh*文件中，其中 `path-to-preload.so` 替换成*preload.so*的全路径。

```
#!/bin/bash
LD_PRELOAD=path-to-preload.so /usr/sbin/nginx
```

v. 更新权限。

```
chmod +x /root/mynginx.sh
```

vi. 编辑*nginx.service*。

```
vi /usr/lib/systemd/system/nginx.service
```

vii. 将其中的 `ExecStart=/usr/sbin/nginx` 行替换为 `ExecStart=/root/mynginx.sh` 。

viii. 重启服务。

```
service nginx restart
```

ix. 设置开机自动启动。

```
systemctl enable nginx.service
```

> **说明** 也可以使用脚本启动*nginx_reload.sh*，命令如下。
>
> ```
> killall nginx
> LD_PRELOAD=path-to-preload.so /usr/local/nginx/sbin/nginx
> ```
>
> 然后把*nginx_reload.sh*路径加到*rc.local*开机启动配置文件中。

3. 配置toa服务开机自动启动。

```
chmod +x /etc/rc.d/rc.local
```

4. 确定服务是否成功加载*preload.so*。

   示例命令如下。

   - 查端口是否开启：`netstat -ntulp |grep 48888`
   - nginx是否正确运行，查PID：`ps -ef | grep nginx`
   - 查是否加载preload：`cat /proc/PID/maps | grep preload.so`

集成代码层TOA

游戏盾TOA模块在本机安装一个进程，监听UDP:48888端口，业务进程将获取到的非真实IP和Port按照固定格式传入此端口进行查询，查询后会返回客户的真实IP和端口。

注意事项

- 确保本机防火墙没有禁用UDP协议上的127.0.0.1:48888端口。
- 使用接入方法（使用UDP协议，127.0.0.1:48888端口访问API服务）时，需要设置超时时间，以免未知问题造成的服务block。

- 由于是旁路获取，所以理论上有极小概率获取失败，主服务本身需要考虑获取失败时的对应方案。
- Server/Clinet任何一端主动close socket时都会删除源IP/Port数据，请重新连接建立数据。

具体请参见TOA压缩包内的指导文档进行操作。详细情况可以咨询游戏盾服务团队。

4.4.3. Windows

本文介绍了在Windows服务器环境下获取客户端真实IP的操作方法。

安装方法

1. 使用Visual Studio 2013以上版本打开*toaservice.sln*并编译。
2. 运行编译打包成的*toaservice.exe*。此为绑定网卡监听并提供UDP 48888端口访问API服务的后台进程。

集成应用层Hook-TOA模块

> **说明** 适用C++，不适用C#等其它语言。

接入方法如下。

1. 目标程序启动时尽快加载*GetSourceName.DLL*，加载方法为：`LoadLibraryA("GetSourceName.dll")`。

   > **说明** 详情见*toaservice.sln*中的TestGetSourceName项目。

2. *GetSourceName.DLL*加载后会HOOK getpeername及accept并返回真实IP。

集成代码层TOA

与Linux相同，通过代码的方式集成。游戏盾TOA模块在本机安装一个进程，监听UDP:48888端口，业务进程将获取到的非真实IP和Port按照固定格式传入此端口进行查询，查询后会返回客户的真实IP和端口。

注意事项

- 该源码仅支持Windows Server 2008。
- 测试时确保本机防火墙关闭，否则WSASocket SOCK_RAW无法sniffer到接收的数据包，导致无法获取真实IP。
- 由于是旁路获取，所以理论上有极小概率获取失败，主服务本身需要考虑获取失败时的对应方案。
- LoadLibrary加载*GetSourceName.dll*后，不能FreeLibrary释放*GetSourceName.dll*，否则有崩溃风险。

  > **说明** 自行购买微软的Detours替换mhook可解决该问题。

- *GetSourceName.dll*、*toaservice.exe*和加载*GetSourceName.dll*的*EXE*程序必须在同一目录。
- 加载*GetSourceName.dll*的EXE程序必须使用管理员权限启动。
- 出于性能优化sniffer时需要绑定网卡，若服务器有多个网卡，请修改源码（*toa_service/win/toaservice/toaservice/Sniffer.cpp*131-139行）绑定正确网卡。

具体请参见TOA压缩包内的指导文档进行操作。详细情况可以咨询游戏盾服务团队。

4.5. 核心函数说明

游戏盾SDK中包含两个核心函数，分别是initEx和GetProxyTcpByDomain。本文介绍了核心函数的作用和参数描述。

initEx

initEx用于初始化SDK，在首次使用SDK时调用。如果返回值不是0，建议循环调用一直到成功。

initEx的参数描述如下。

名称	说明
access_key	游戏盾接入身份密钥。从游戏盾管理控制台下载。具体操作，请参见获取SDK和AccessKey。
token	游戏内的玩家标识。发生DDoS攻击时定位恶意玩家或黑客使用，如果未定义可以设置为*Default*。

GetProxyTcpByDomain

GetProxyTcpByDomain函数用于获取回传IP和随机端口，使用同步阻塞的方法。

GetProxyTcpByDomain函数的特性说明如下。

- 在单次App启动后的生命周期内，同样的参数返回的IP和端口结果相同。
- 单次调用转化后的IP和端口在本次App生命周期内不会变化，IP和端口会自动检查是否可用，自动进行节点切换。
- App重新启动后，获取的IP和端口会发生变化。
- 函数返回的Target_ip是固定的127.0.0.1，您也可以动态获取此函数的返回结果。

GetProxyTcpByDomain的参数描述如下。

名称	说明
Token	游戏内的玩家标识。发生DDoS攻击时定位恶意玩家或黑客使用，如果未定义可以设置为Default。
GroupName	游戏业务对应的节点组标识，例如：access.v812vCOE21.aliyundoc.com。在游戏盾控制台添加游戏和业务时需要配置节点组，根据游戏同时在线用户规模分配独享节点数量。一个游戏对应多个节点组。
Dip	防护目标标识，需要转化的远端服务器的IP，在游戏盾无限抗防护目标处配置获取。
Dport	服务器的业务端口，按实际需要传入，无需在游戏盾中配置。
target_ip	返回的IP，固定为127.0.0.1。
target_port	返回的随机端口。

4.6. SDK错误码

本文介绍了游戏盾SDK返回值的定义以及常见错误码的说明。

返回值定义

返回值	定义
0	成功接入游戏盾SDK。
1000~1999	接入游戏盾SDK时网络通信错误。
2000~2999	接入游戏盾SDK时appkey错误或SDK初始化错误。
3000~3999	接入游戏盾SDK时游戏盾控制中心错误。
4000~4999	SDK与游戏盾控制中心的数据交换错误。

常见错误码

返回值	说明	解决方法
-1	游戏盾SDK的groupname或者其他参数的值为空	请输入groupname或者其他参数的值。
0	成功接入游戏盾SDK	不涉及。
2000	游戏盾SDK的appkey值为空	请获取appkey的值。
2001	游戏盾SDK的appkey值的格式错误	请验证appkey值的格式是否完整。
2002	游戏盾SDK的appkey值的长度过长	请检查appkey值的长度是否正确。
2005	没有调用游戏盾SDK的初始化接口	请调用游戏盾SDK的初始化接口。
3201	游戏盾SDK功能未开启	请开启游戏盾SDK功能。
3305	游戏盾SDK请求参数错误	请检查请求参数是否正确，如果问题仍未解决，请联系阿里云售后技术支持。
3306	游戏盾SDK请求类型错误	请检查调用接口是否正确，如果问题仍未解决，请联系阿里云售后技术支持。
3307	游戏盾SDK请求参数错误	请检查请求参数是否正确，如果问题仍未解决，请联系阿里云售后技术支持。
3500	游戏盾的节点组内没有配置IP	请在游戏盾控制台配置IP。
3600	游戏盾的节点组内没有可用IP	请在游戏盾控制台重新配置IP或开启无限抗功能。
3700	游戏盾SDK的分组名字（groupname）错误	请检查参数是否正确传入了groupname，如果问题仍未解决，请联系阿里云售后技术支持。
3702	游戏盾的防护目标为空	请在游戏盾控制台进行无限抗防护目标配置。

返回值	说明	解决方法
3703	游戏盾的转发规则为空	请在游戏盾控制台进行无限抗端口配置。
3800	游戏盾SDK数据通过HTTP网络传输被劫持	请联系阿里云售后技术支持。
3999	游戏盾SDK接口参数错误	请检查参数是否正确，如果问题仍未解决，请联系阿里云售后技术支持。
4000	游戏盾SDK数据通过HTTP网络传输被劫持	请联系阿里云售后技术支持。
9100	游戏盾SDK接口被多个线程同时调用	请在一个线程调用完成后再发起新的调用。

说明 如果您的问题未能解决，请联系阿里云售后技术支持。

4.7. SDK排错

本文介绍了未正确接入游戏盾SDK时的排错方法。

背景信息

App调用GetProxyTcpByDomain通过安全网络服务管理调度可用IP并返回客户端，客户端根据不同的请求线路随机访问不同的安全网络IP池，最终到达回源服务器。

调试建议

Debug模式一定要详细打印传入参数和返回参数，方便问题排查。如果遇到棘手问题，建议通过Wireshark抓包，提供异常报文给游戏盾团队协助您分析传输问题。

操作步骤

1. 版本问题自查。

   根据SDK错误码完成自查，根据错误提示修改游戏盾控制台的配置或者调整调用参数。

2. 确认原站状态。

   确认原站的状态，对应真实服务器的IP和端口是否能够正常访问。

   您可以联系游戏盾服务人员协助您判断游戏盾到您原站的传输链路状态，以防止原站有防火墙策略，拦截了游戏盾的回源IP地址。这种方式下获得的结果会比您在公网测试更准确。

3. 再次确认传入参数问题。

   打印两个信息，第一个是通过SDK获取到的IP和Port信息，另外一个是您访问的URL。再次确认游戏盾传入的参数是否正常，参数大小写敏感，一定要注意。

4. 确认协议类型。

   - TCP协议：无需特殊处理，返回IP和端口正常都可用。
   - HTTP/HTTPS协议：可能需要处理HOST匹配问题，需要服务端工作人员配合。
   - WS/WSS协议：可能需要处理HOST匹配问题，需要服务端工作人员配合。