

接北斗星 9.0 后台 API 测试指引 .....	1
一.    北斗星 9.0 对接开发说明 .....	1
二. 开发注意事项 .....	2
1. 需要确认 API 程序包版本 .....	2
2. 9.0 行情 API 接口需要使用转换文件 .....	3
3. 对接交易接口时需要交易 API 初始完成才能操作其他操作 .....	4
4. 9.0 2FA 逻辑变化 .....	4
5. 请留意 API 授权到期日期 .....	7
6. 登录身份以及非单客户对接注意 .....	7
7. 委托及成交状态更新监听注意 .....	9
8. 港交所夜盘 (T+1 生效单) .....	10
9. 交易 API 密码到期类似极星“忽略”功能实现逻辑 .....	11
10. 隐含行情 .....	13
11. 开发手册见易盛开发文档 .....	15
三. 模拟环境 .....	15
3.1. 易盛北斗星 9.0 模拟环境 .....	15
3.2. 新永安北斗星 9.0 模拟环境 .....	17
四. 实盘 .....	19
4.1. 实盘 9.0 行情账号申请 .....	19
4.2. 9.0 交易接口授权申请 .....	20
4.3. 9.0 接口实盘绑定申请 .....	20
4.4. 9.0 行情账号以及交易授权续期 .....	20
4.5. 我司客服部联系方式 .....	20

## 接北斗星 9.0 后台 API 测试指引

### 一. 北斗星 9.0 对接开发说明

北斗星 9.0 交易后台是基于 Linux 运行的平台，易盛 3.0 后台是基于 Windows 运行的平台。北斗星 9.0 的系统性能稳定性优于易盛 3.0 系统。

注意：我司北斗星 9.0 系统只提供 9.0 行情接口。

- 9.0 行情平台：提供新行情 9.0 接口，但不兼容现有 3.0 接口，未来我司不再提供 3.0 行情。
- 9.0 交易平台：提供 9.0 交易接口，兼容现有 3.0 接口，客户可自行选择是否对接新的 9.0 交易接口。

9.0 交易接口中有部分交易所的商品代码与现有 3.0 接口不同，9.0 采用较为接近交易所商品代码。所以，对接存在三种情况，请根据业务需要自行选择对接方案

方案	交易	行情	转换品种代码	方案描述	说明及优势
----	----	----	--------	------	-------

编号	接口	接口			
1	9.0	9.0	无(都是 9.0 代码)	行情和交易接口都对接 9.0	推荐。 1) 行情和交易的品种代码一致, 无需转换, 需要《codemapping.csv》文件自动转换部分品种价格精度 2) 以后新功能支持性会比较好
2	3.0	9.0	根据提供的《codemapping.csv》自动转换行情代码为 3.0 的代码	行情对接 9.0, 交易接口不变, 还是 3.0	1) 适合过渡期来不及对接 9.0 交易接口的客户, 只需要对接 9.0 行情接口 2) 通过提供的《codemapping.csv》文件自动转换行情品种代码, 使用上达到行情和交易品种代码都是 3.0 品种代码的效果
3	3.0	9.0	自己做品种代码转换	行情对接 9.0, 交易接口不变, 还是 3.0	不推荐。 客户根据提供的 9.0 品种代码和 3.0 品种代码自行建立品种对应关系, 并转换, 以达到行情和交易品种代码一样的使用效果

## 二. 开发注意事项

### 1. 需要确认 API 程序包版本

因为易盛 9.0 支持 9.0 交易 API, 并且兼容原来 3.0 交易 API, 所以需要先确认您是以 9.0 交易 API 对接北斗星后台, 还是以 3.0 交易 API 对接北斗星后台。

#### 1.1. 3.0 交易 API 对接

- 1) 需要升级 3.0 交易 API 版本到 3.0.1.52 及以上
- 2) 3.0 交易 API 包下载地址:

<http://www.esunny.com.cn/uploads/software/API/%E5%A4%96%E7%9B%983.0tradeapi.zip>



### 如何获取 API 库



各个版本对应的API可从下表下载，另提供客户端下载用于API程序的测试监控。除8.0API外，所有开发包中均提供了易盛的API测试环境和测试授权码。用于易盛测试环境的API开发调试，无需再申请授权码，仅需要注册帐号或使用公共帐号。

市场	API类型	API开发包	更新日期	版本	测试账号	客户端下载
内盘	8.0API	点此下载			不提供测试环境	
	启明星API	点此下载	2018-4-18		点此注册	点此下载
外盘	3.0交易API	点此下载	2018-6-21	3.0.1.52	点此注册	
	3.0行情API	点此下载	2017-6-7		不提供测试环境	
	9.0行情API	点此下载	2017-12-11		用户名: ES 密码: 123456	

## 1.2. 9.0 交易 API 对接

- 1) 需要升级 9.0 交易 API 版本到 9.2.7.7 及以上
- 2) 9.0 交易 API 包下载地址:

<http://www.esunny.com.cn/uploads/software/API/%E5%A4%96%E7%9B%989.0tradeapi.zip>



### 如何获取 API 库



各个版本对应的API可从下表下载，另提供客户端下载用于API程序的测试监控。除8.0API外，所有开发包中均提供了易盛的API测试环境和测试授权码。用于易盛测试环境的API开发调试，无需再申请授权码，仅需要注册帐号或使用公共帐号。

市场	API类型	API开发包	更新日期	版本	测试账号	客户端下载
内盘	8.0API	点此下载			不提供测试环境	
	启明星API	点此下载	2018-4-18		点此注册	点此下载
外盘	3.0交易API	点此下载	2018-6-21	3.0.1.52	点此注册	
	3.0行情API	点此下载	2017-6-7		不提供测试环境	
	9.0行情API	点此下载	2017-12-11		用户名: ES 密码: 123456	
	9.0交易API	点此下载	2018-8-10	9.2.8.1	点此注册	点此下载

## 2. 9.0 行情 API 接口需要使用转换文件

因我司 3.0 系统与北斗星 9.0 系统采用部分不同的品种交易代码，所以如果您使用易盛 9.0 行情，可以使用转换文件《codemapping.csv》放在 9.0 行情 API 的 dll 同级根目录下，默认是运行根目录，API 会自动加载该文件并做好 9.0 行情代码与交易品种代码之间的品种代码转换和价格精度的转换，您就可以直接使用对应的接口交易品种代码来订阅 9.0 行情。如上面所说对接方案 1 和方案 2 将会使用到。

## 2.1.交易使用 9.0 交易 API 接口对接 9.0 后台

- 1) 请使用《/9.0 交易 API 客户使用映射文件/ codemapping.csv》
- 2) 9.0 交易品种代码见《9.0 品种信息.xls》

## 2.2.交易使用 3.0 交易 API 接口对接 9.0 后台

- 1) 请使用《/3.0 交易 API 客户使用映射文件/ codemapping.csv》
- 2) 3.0 交易品种代码见《3.0 品种信息.xls》

## 3. 对接交易接口时需要交易 API 初始完成才能操作其他操作

- 1) 9.0 交易接口初始化完成回调方法

```
/**
 * @brief 通知用户API准备就绪。
 * @details 只有用户回调收到此就绪通知时才能进行后续的各种行情数据查询操作。 \n
 * 此回调函数是API能否正常工作的标志。
 * @attention 就绪后才可以进行后续正常操作
 * @ingroup G_T_Login
 */
virtual void TAP_CDECL OnAPIReady() = 0;
```

- 2) 3.0 交易接口初始化完成回调方法

```
////////////////////////////////////
/// \fn OnInitFinished
/// \brief 发送Login成功后, 收到OnLogin应答成功后收到初始化操作完成
/// 所有的业务操作需要在本响应errCode为0 (成功) 后可进行
/// \param int errCode 错误码, 0-成功, 其他值-错误的原因代码
/// \return void __cdecl
////////////////////////////////////
virtual void __cdecl OnInitFinished(int errCode) = 0;
```

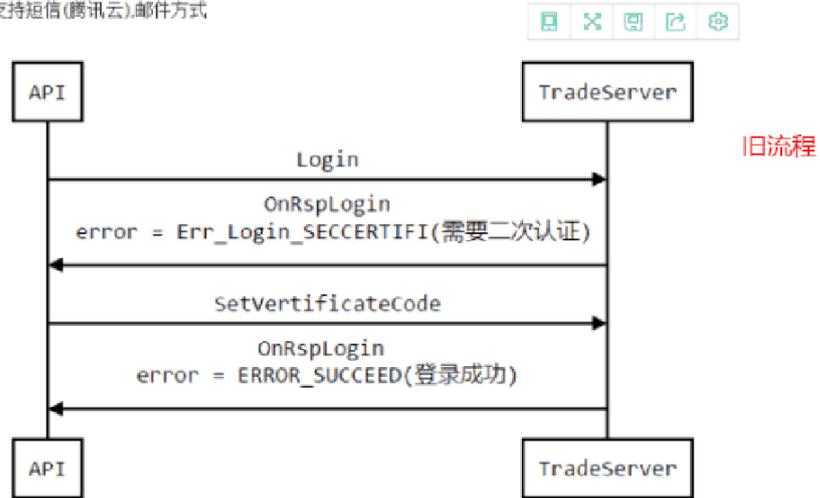
## 4. 9.0 2FA 逻辑变化

我司目前 9.0 后台版本是修订 7（后续如果后台升级会大于修订 7 版本），所以 2FA 在这个版本及以后跟之前的逻辑变化，请注意对接

## 4.1.修订 7 之前 2FA 逻辑

### 二次认证

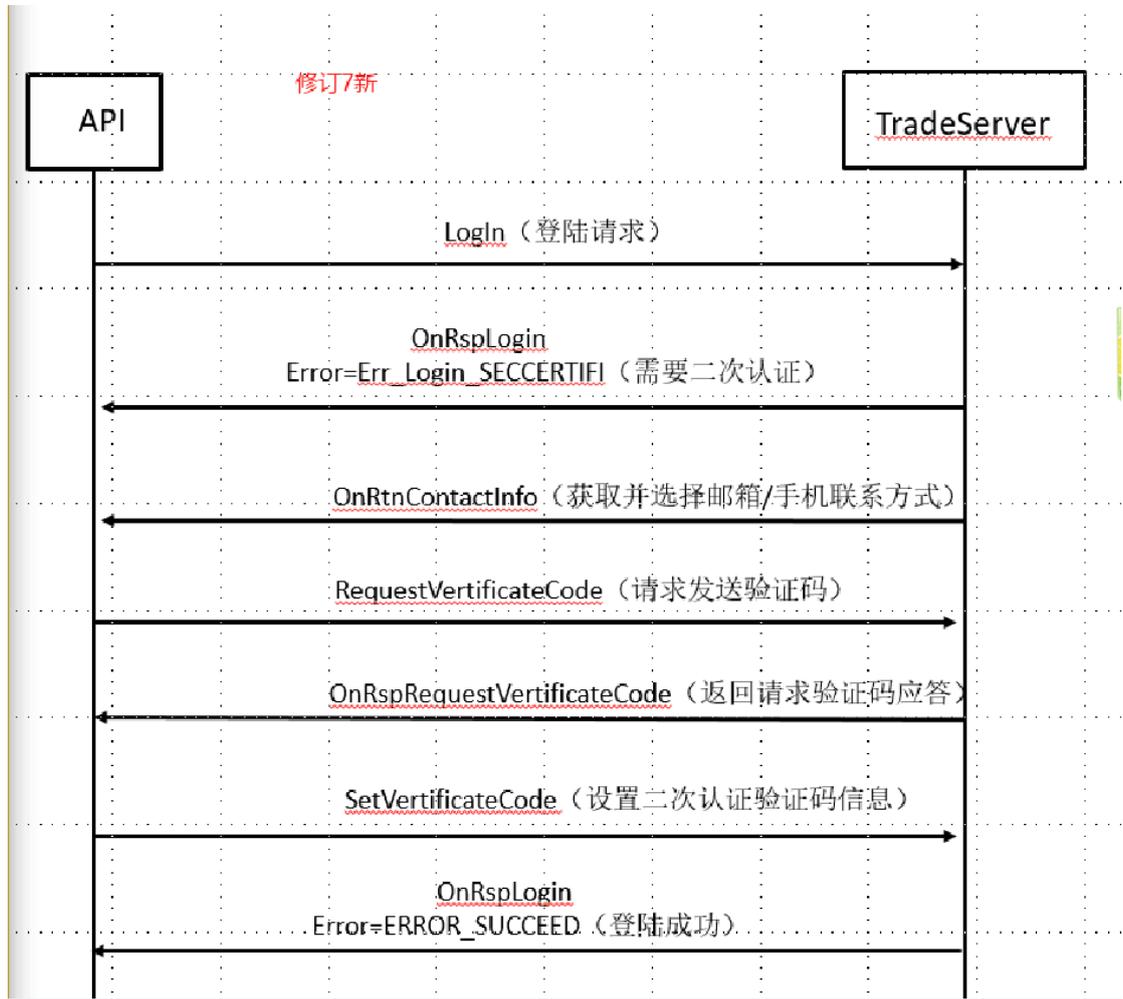
- 支持短信(腾讯云),邮件方式



#### 其他错误解释

- Err\_Certificate\_Logged 已完成登录密码验证、二次验证。再次发送二次验证码的情况。
- Err\_Login\_SecCertFailed 二次认证码错误
- Err\_Login\_SecCertTimeout 二次认证超时, 时限由后台控制。

## 4.2.修订 7 及之后版本 2FA 逻辑



## 4.3.注意 2FA 提示错误码的变化

1) 旧版本（修订 7 前）逻辑：

3.0 交易 API 登录北斗星 9.0 需要二次认证报错：27

//需要二次认证

```
const TErrorCodeType Err_Login_SECCERTIFI = 27;
```

9.0 交易 API 登录北斗星 9.0 后台需要二次认证报错：10015

//! 需要进行二次认证

```
const int TAPIERROR_LOGIN_SECCERTIFI = 10015;
```

2) 新版本（修订 7 及之后）逻辑：

3.0 交易 API 登录北斗星 9.0 需要二次认证报错：32

//需要进行动态认证

```
const TErrorCodeType Error_Login_DDA = 32;
```

9.0 交易 API 登录北斗星 9.0 后台需要二次认证报错：10003

///! 需要进行动态认证

```
const int TAPIERROR_LOGIN_DDA = 10003;
```

- 3) 9.0 操作员（非单客户）同样支持登录时二次认证的功能，处理逻辑跟单客户一样。

## 5. 请留意 API 授权到期日期

在登录，初始化完成后，可以调用查询 API 授权到期日查询方法查询您的使用的授权什么时候到期，可以做到提醒的功能，以免耽误您交易。

- 1) 9.0 交易接口自动回调授权到期日方法

```
/**
 * @brief API到期提醒回调
 * @details 此函数为Login()登录成功后，如果到期日与当前日期小于30天，则进行回调提醒。
 * @param[in] date 返回API授权到期日。
 * @param[in] days 返回还有几天授权到期。
 * @attention 该函数回调，则说明授权在一个月之内到期。否则不产生该回调。
 * @ingroup G_I_Login
 */
virtual void TAP_CDECL OnExpirationDate(ITapTrade::TAPIDATE date, int days) = 0;
```

- 2) 3.0 交易接口查询授权到期日方法

```
////////////////////////////////////
/// \fn GetCertCodeExpireDate
/// \brief 获取认证码过期日期
/// \param TDateType & expireDate
/// \return int __cdecl
////////////////////////////////////
virtual int __cdecl GetCertCodeExpireDate(TDateType& expireDate) = 0;
```

## 6. 登录身份以及非单客户对接注意

### 6.1. 登录身份

- 1) 9.0 目前登录时，没有身份字段区分
- 2) 3.0 登录时有身份字段区分

```

//登录身份类型
typedef char TEsLoginIdentityType;
//下单客户端
const TEsLoginIdentityType IDENTITY_CLIENT = 'c';
//下单人端
const TEsLoginIdentityType IDENTITY_TCLIENT = 'd';
//经纪人端
const TEsLoginIdentityType IDENTITY_BROKER = 'b';
//交易员端
const TEsLoginIdentityType IDENTITY_TRADER = 't';
//柜台管理端
const TEsLoginIdentityType IDENTITY_MANAGER = 'm';

```

2.1) 单客户：登入时账号填充适用 ClientNo 字段,Identity 字段填“c”

```

//登录请求结构
struct TEsLoginReqField
{
    TIsCaLoginType IsCaLogin; //是否CA认证
    TEsLoginIdentityType Identity; //登录身份类型,目前只支持单客户
    TIsForcePasswordType IsForcePwd; //是否强制修改密码
    // 使用不同账户登录时共用登录号

    union
    {
        TClientNoType ClientNo; //客户号,代理客户号
        TOperatorNoType OperatorNo; //操作员号,代理操作员号
    };

    /// 对应所使用的登录账号的密码
    TLoginPasswordType LoginPwd; //登录密码
    // 强制修改密码时的新密码
    TLoginPasswordType NewPwd; //强制修改密码登录时,新修改密码
    TOtpPassType OtpPass; //otp认证密码
    TEsSizeType CaLen; //CA信息长度, IsCaLogin为'Y'时, 本字段有效
    TCaInfoType CaInfo; //CA登录时填写,IsCaLogin为'Y'时, 本字段有效
};

```

2.2)其他类型账号(经纪人、下单人等): 登入时账号填充使用 OperatorNo 字段, Identity 字段请根据使用者的角色参考图 1 填写对应字段

```

//登录请求结构
struct TEsLoginReqField
{
    TIsCaLoginType IsCaLogin; //是否CA认证
    TEsLoginIdentityType Identity; //登录身份类型,目前只支持单客户
    TIsForcePasswordType IsForcePwd; //是否强制修改密码
    // 使用不同账户登录时共用登录号

    union
    {
        TClientNoType ClientNo; //客户号,代理客户号
        TOperatorNoType OperatorNo; //操作员号,代理操作员号
    };

    /// 对应所使用的登录账号的密码
    TLoginPasswordType LoginPwd; //登录密码
    // 强制修改密码时的新密码
    TLoginPasswordType NewPwd; //强制修改密码登录时,新修改密码
    TOtpPassType OtpPass; //otp认证密码
    TEsSizeType CaLen; //CA信息长度, IsCaLogin为'Y'时, 本字段有效
    TCaInfoType CaInfo; //CA登录时填写,IsCaLogin为'Y'时, 本字段有效
};

```

## 6.2. 非单客户身份对接注意

### 6.2.1) 获取下属单客户账号

## 1) 9.0 交易接口获取下属单客户方法

```
/**
 * @brief 查询用户下属的资金账号
 * @details TapAPIAccQryReq是包含资金账号结构的结构体，用户需要实例化此结构体，
 *          但是不需要填写，这个结构体主要是用来提醒用户保存资金账号，资金账号
 *          在后续的函数调用中很多函数都需要使用到。
 * @param[out] sessionID 返回请求的会话ID;
 * @param[in] qryReq 查询用户下属的资金账号的结构体指针
 * @retval 0 请求成功
 * @retval 非0 错误码
 * @operationtype 异步操作
 * @ingroup G_I_AccountInfo
 */
virtual ITapTrade::TAPIINT32 TAP_CDECL QryAccount (ITapTrade::TAPIUINT32 *sessionID, const ITapTrade::TapAPIAccQryReq *qryReq) = 0;
```

## 2) 3.0 交易接口获取下属单客户方法

```
////////////////////////////////////
/// \fn QryClients
/// \brief 查询交易员下属客户，暂时未用
/// \param const TOperatorClientQryReq & qryClients
/// \return int __cdecl
////////////////////////////////////
virtual int __cdecl QryClients(const TEsOperatorClientQryReqField& qryClients, int& iReqID) = 0;
```

### 6.2.2) 下单和查询交易数据需要使用对应单客户号

下单及查询交易数据时，不论 3.0 还是 9.0 接口，非单客户身份需要填写对接的下属客户账号去操作

```
//报单请求结构
struct TEsOrderInsertReqField
{
    TClientNoType ClientNo;
    TCommodityNoType CommodityNo;
    TContractNoType ContractNo;
    TOrderTypeType OrderType;
    TOrderWayType OrderWay;
    TOrderModeType OrderMode;
    TDateTimeType ValidDateTime;
    TIsRiskOrderType IsRiskOrder;
    TDirectType Direct;
    TOffsetType Offset;
    THedgeType Hedge;
    TTradePriceType OrderPrice;
    TTradePriceType TriggerPrice;
    TTradeVolType OrderVol;
    TTradeVolType MinMatchVol;

    TRandomise Randomise;
    TTradeVolType MinClipSize;
    TTradeVolType MaxClipSize;

    TOrderSaveIntType SaveInt;
    TOrderSaveDoubleType SaveDouble;
    TOrderSaveStringType SaveString;
};

public:
TEsOrderInsertReqField()
{
    memset(this, 0, sizeof(TEsOrderInsertReqField));
};
```

登录身份是非单客户时，报单及交易数据查询等涉及需要填写ClientNo的地方都需要使用操作员下属指定的客户号

例如：登录身份是下单人：P6666,报单时，ClientNo 需要填其下属客户号：6666。

## 7.委托及成交状态更新监听注意

### 1) 9.0 交易接口需要同时监听以下 2 个方法

```

* @brief 返回新委托。新下的或者其他地方下的推送过来的。
* @details 服务器接收到客户下的委托内容后就会保存起来等待触发，同时向用户回馈一个
*          新委托信息说明服务器正确处理了用户的请求，返回的信息中包含了全部的委托信息，
*          同时有一个用来标示此委托的委托号。
* @param[in] info 指向返回的信息结构体。当errorCode不为0时，info为空。
* @note 如果不关注此项内容，可以设定Login时的NoticeIgnoreFlag以屏蔽。
* @attention 不要修改和删除info所指示的数据；函数调用结束，参数不再有效。
* @ingroup G_T_TradeActions
*/
virtual void TAP_CDECL OnRtnOrder(const ITapTrade::TapAPIOrderInfoNotice *info) = 0;
/**
* @brief 推送来的成交信息
* @details 用户的委托成交后将向用户推送成交信息。
* @param[in] info 指向返回的信息结构体。当errorCode不为0时，info为空。
* @note 如果不关注此项内容，可以设定Login时的NoticeIgnoreFlag以屏蔽。
* @attention 不要修改和删除info所指示的数据；函数调用结束，参数不再有效。
* @ingroup G_T_TradeActions
*/
virtual void TAP_CDECL OnRtnFill(const ITapTrade::TapAPIFillInfo *info) = 0;

```

## 2) 3.0 交易接口

# 成交推送与委托状态推送

- 单腿成交信息推送

```
virtual void __cdecl OnRtnMatchState(const TEsMatchStateNoticeField& rsp) = 0
```

- 多腿成交信息推送

```
virtual void __cdecl OnRtnMatchInfo(const TEsMatchInfoNoticeField& rsp) = 0
```

成交信息的推送，具体再哪个函数返回取决于易盛的网关程序，多腿成交返回与OnRtnMatchInfo，单腿成交返回于OnRtnMatchState或者OnRtnMatchInfo，开发者在开发时需要处理以上两个函数接收成交。

## 8.港交所夜盘（T+1 生效单）

因为港交所对于 T+1 生效订单需要做特别标识，所以在下单港交所订单时，如果您想在 T+1 能正常生效，请按下面处理：

### 1) 9.0 交易接口

```

    /// 委托完整信息
    struct TapAPIOrderInfo
    {
        TAPIYNFLAG
        AddOneIsValid;    ///< 是否T+1有效
    }

```

## 2) 3.0 交易接口

```

///报单请求结构
struct TEsOrderInsertReqField
{
    TClientNoType      ClientNo;
    TCommodityNoType   CommodityNo;    //商品代码
    TContractNoType    ContractNo;     //合约代码
    TOrderTypeType     OrderType;      //委托类型
    TOrderWayType      OrderWay;       //委托方式,单客户不用关注本字段
    TOrderModeType     OrderMode;      //委托模式
    TDateTimeType      ValidDateTime;   //有效日期 (GTD情况下使用)
    TIsRiskOrderType   IsRiskOrder;    //风险报单
    TDirectType        Direct;         //买入卖出
    TOffsetType        Offset;         //开仓平仓
    THedgeType         Hedge;          //投机保值
    TTradePriceType    OrderPrice;     //委托价格
    TTradePriceType    TriggerPrice;   //触发价格
    TTradeVolType      OrderVol;       //委托数量
    TTradeVolType      MinMatchVol;    //最小成交量
}

```

- 实盘：交易港交所夜盘时需要加入 港交所交易时段后有效标记，否则无法交易。报单请求结构中Hedge = HEDGE\_B即可。
- 模拟：模拟系统未模拟此交易规则，无论是否填写HEDGE\_B，均可在晚间交易

## 9.交易 API 密码到期类似极星“忽略”功能实现逻辑

非开户第一次登录交易，当您的交易 API 密码到期时，登录会返回报错信息：10006-需要强制修改密码，则可以根据您方需要通过下面的方法和逻辑实现极星客户端类似的“忽略”功能来保持交易密码不变。

```

///! 需要强制修改密码
const int      TAPIERROR_LOGIN_FORCE = 10006;

```

```

/**
 * @brief 发起登录请求。API将先连接服务，建立链路，发起登录认证。
 * @details 在使用函数前用户需要完成服务器的设置SetHostAddress()，并且创建TapAPITradeLoginAuth类型的用户信息，
 * 并且需要设置好回调接口。
 * 登录过程中建立连接的返回信息通过回调OnConnect返回给用户。
 * 连接建立后的用户验证回馈信息通过回调OnLogin()返回给用户。
 * 登录成功后API会自动进行API的初始化，API向服务器请求基础数据，查询完以后会通过回调OnAPIReady()
 * 指示用户API初始化完成，可以进行后续的操作了。
 * @param[in] loginAuth 登录验证信息结构指针。包含登录需要的验证信息。
 * @retval 0 登录成功，API开始准备后台数据
 * @retval !=0 错误码
 * @operationtype 异步操作
 * @note TapAPITradeLoginAuth中的NoticeIgnoreFlag用于标记是否忽略特定的通知回调。
 * 例如：不需要接收OnRtnFund和OnRtnPositionProfit,可以这么设定：
 * NoticeIgnoreFlag = TAPI_NOTICE_IGNORE_FUND | TAPI_NOTICE_IGNORE_POSITIONPROFIT;
 * @attention Login()函数调用成功只是代表于服务器建立了链路连接，只有回调OnLogin()的返回能指示用户是否成功登录了。
 * @ingroup G_T_Login
 */
virtual ITapTrade::TAPIINT32 TAP_CDECL Login(const ITapTrade::TapAPITradeLoginAuth *loginAuth) = 0;

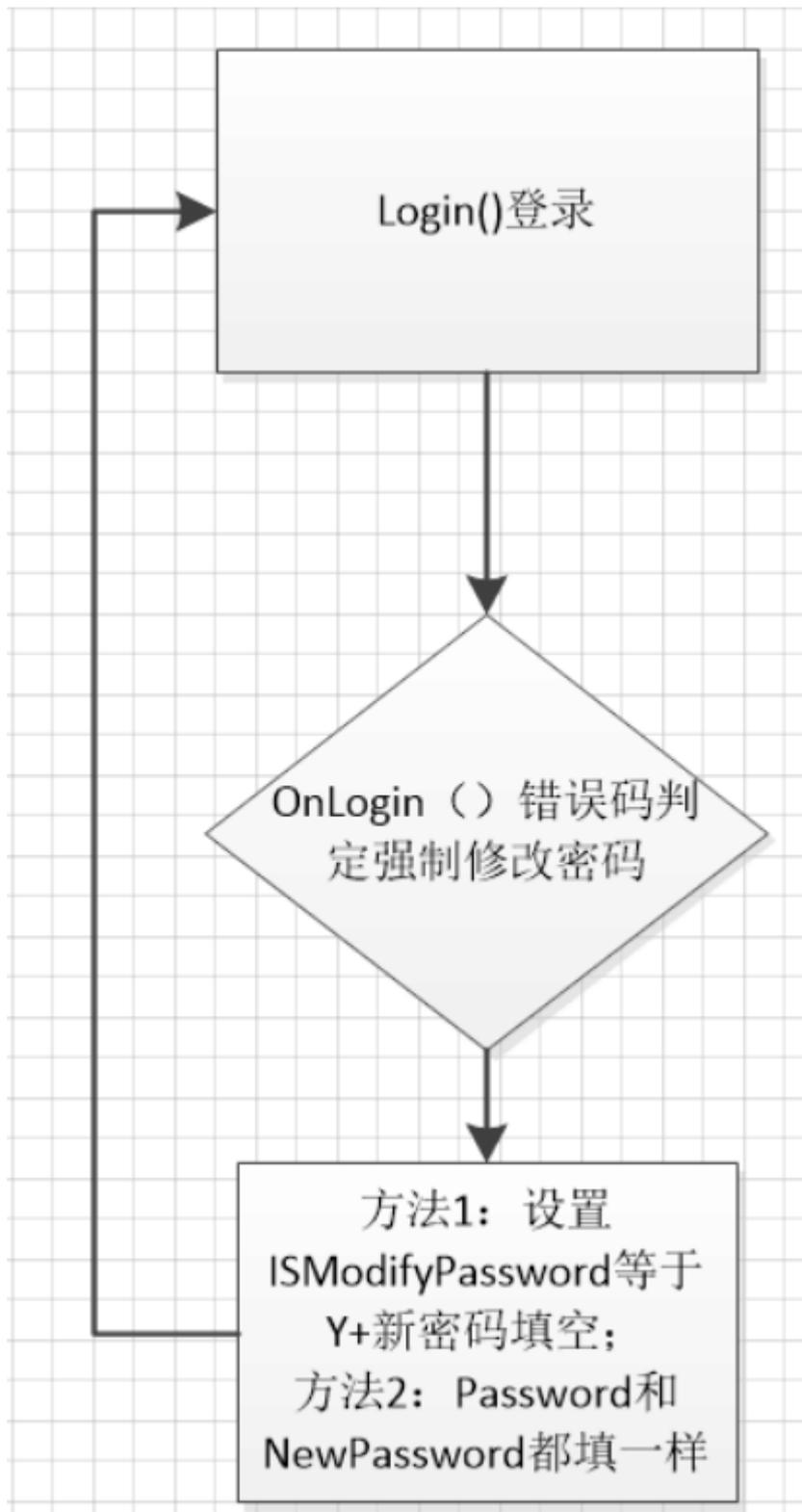
```

///**登录认证信息**

```

struct TapAPITradeLoginAuth
{
    TAPISTR_20 UserNo;          ///< 用户名
    TAPIYNFLAG ISModifyPassword; ///< 是否修改密码
    TAPISTR_20 Password;       ///< 密码
    TAPISTR_20 NewPassword;    ///< 新密码
};

```



## 10. 隐含行情

隐含价格：行情价格上有星星，表示隐含价格，和对应的挂单量。现在正常行情是 10 档报价，隐含价格就是 10 档之外额外的报价，来源可能是场外交易或组合派生价格额外的一档报价。这个价格可能优于十档里的第一档报价，这时候盘上就显示出带\*号的

价格，表示当前显示的是隐含价格。隐含价格是一档行情，可以显示到 10 档里相应的位置，这时候的十档和原始的十档就已经不一样了。

## 10.1) 是否有隐含行情需要看交易所是否提供，交易所提供才有

## 10.2) API 获取隐含行情字段

### 10.2.1) 9.0 行情接口隐含报价字段

易盛模拟环境的 9.0 行情隐含报价不准确，请以实盘 9.0 行情为准。模拟行情是处理后的行情。

```
/// 行情全文
struct TapAPIQuoteWhole
{
    TapAPIContract      Contract;          ///< 合约
    TAPISIR_10          CurrencyNo;       ///< 币种编号
    TAPICHAR            TradingState;     ///< 交易状态。1,集合竞价;2,集合竞价撮合;3,连续交易;4,交易暂停;5,闭市
    TAPIDTSTAMP         DateTimeStamp;    ///< 时间戳
    TAPIQPRICE          QPreClosingPrice; ///< 昨收盘价
    TAPIQPRICE          QPreSettlePrice;  ///< 昨结算价
    TAPIQVOLUME         QPrePositionQty;  ///< 昨持仓量
    TAPIQPRICE          QOpeningPrice;    ///< 开盘价
    TAPIQPRICE          QLastPrice;       ///< 最新价
    TAPIQPRICE          QHighPrice;       ///< 最高价
    TAPIQPRICE          QLowPrice;        ///< 最低价
    TAPIQPRICE          QHisHighPrice;    ///< 历史最高价
    TAPIQPRICE          QHisLowPrice;     ///< 历史最低价
    TAPIQPRICE          QLimitUpPrice;    ///< 涨停价
    TAPIQPRICE          QLimitDownPrice;  ///< 跌停价
    TAPIQVOLUME         QTotalQty;        ///< 当日总成交量
    TAPIQPRICE          QTotalTurnover;   ///< 当日成交金额
    TAPIQVOLUME         QPositionQty;     ///< 持仓量
    TAPIQPRICE          QAveragePrice;    ///< 均价
    TAPIQPRICE          QClosingPrice;    ///< 收盘价
    TAPIQPRICE          QSettlePrice;     ///< 结算价
    TAPIQVOLUME         QLastQty;         ///< 最新成交量
    TAPIQPRICE          QBidPrice[20];    ///< 买价1-20档
    TAPIQVOLUME         QBidQty[20];     ///< 买量1-20档
    TAPIQPRICE          QAskPrice[20];   ///< 卖价1-20档
    TAPIQVOLUME         QAskQty[20];     ///< 卖量1-20档
    TAPIQPRICE          QImpliedBidPrice; ///< 隐含买价
    TAPIQVOLUME         QImpliedBidQty;  ///< 隐含买量
    TAPIQPRICE          QImpliedAskPrice; ///< 隐含卖价
    TAPIQVOLUME         QImpliedAskQty;  ///< 隐含卖量
    TAPIQPRICE          QPreDelta;        ///< 昨虚实度
    TAPIQPRICE          QCurrDelta;       ///< 今虚实度
    TAPIQVOLUME         QInsideQty;       ///< 内盘量
    TAPIQVOLUME         QOutsideQty;     ///< 外盘量
    TAPIQPRICE          QTurnoverRate;    ///< 换手率
    TAPIQVOLUME         QSDAvgQty;       ///< 五日均量
    TAPIQPRICE          QPERatio;        ///< 市盈率
    TAPIQPRICE          QTotalValue;     ///< 总市值
    TAPIQPRICE          QNegotiableValue; ///< 流通市值
    TAPIQDIFF           QPositionTrend;   ///< 持仓走势
    TAPIQPRICE          QChangeSpeed;    ///< 涨幅
    TAPIQPRICE          QChangeRate;     ///< 涨幅
    TAPIQPRICE          QChangeValue;    ///< 涨跌值
    TAPIQPRICE          QSwing;          ///< 振幅
    TAPIQVOLUME         QTotalBidQty;    ///< 委买总量
    TAPIQVOLUME         QTotalAskQty;    ///< 委卖总量
};
```

### 10.2.2) 3.0 行情接口隐含报价字段

```

struct STKDATA //即时数据结构
{
    char    Market[40];    //市场中文名
    char    Code[66];    //合约代码

    float   YClose;      //昨收盘
    float   YSettle;     //昨结算
    float   Open;       //开盘价
    float   High;       //最高价
    float   Low;        //最低价
    float   New;        //最新价
    float   NetChg;     //涨跌
    float   Markup;    //涨跌幅
    float   Swing;     //振幅
    float   Settle;    //结算价
    float   Volume;   //成交量
    float   Amount;   //持仓量

    float   Ask[10];    //申卖价
    float   AskVol[10]; //申卖量
    float   Bid[10];   //申买价
    float   BidVol[10]; //申买量

    float   AvgPrice;  //平均价

    float   LimitUp;   //涨停板
    float   LimitDown; //跌停板
    float   HistoryHigh; //合约最高
    float   HistoryLow; //合约最低

    float   YOPI;     //昨持仓
    float   ZXSD;    //昨虚实度
    float   JXSD;    //今虚实度
    float   CJJE;    //成交金额

    //新增外盘
    float   IClose;   //收盘价
    float   Lastvol;  //最新成交量
    float   status;   //合约交易状态 -1:未知 0:开市 1:无红利 2:竞价 3:挂起 4:闭市 5:开市前 6:闭市前 7:快市
    float   updatetime; //更新时间,形如: 235959表示23: 59: 59

    float   BestBPrice; //最优买价
    float   BestBVol;   //最优买量
    float   BestSPrice; //最优卖价
    float   BestSVol;   //最优卖量
};

```

隐含行情字段

## 11. 开发手册见易盛开发文档

- 1) 9.0API 请参考《易盛外盘 9.0API 培训.pdf》
- 2) 3.0API 请参考《易盛国际金融衍生品交易分析系统 3.0API 使用说明.pdf》

## 三. 模拟环境

我司跟易盛都有北斗星 9.0 的模拟环境，可以先在易盛北斗星 9.0 模拟开发测试，拿到 9.0 交易授权，然后再来我司 9.0 环境业务测试。也可直接申请 9.0 交易授权后，到我司 9.0 模拟环境开发测试。

### 3.1. 易盛北斗星 9.0 模拟环境

易盛北斗星 9.0 模拟环境，有 9.0 模拟行情以及 9.0 模拟交易环境

### 3.1.1 易盛模拟账号申请

1) 登录网页:

<http://61.163.243.173:8081/index.php?m=content&c=index&a=lists&catid=50>

: | 61.163.243.173:8081/index.php?m=content&c=index&a=lists&catid=50



2) 填写完信息后, 根据提示做激活, 激活后等待易盛发账号信息到您邮箱

### 3.1.2 下载对应最新的 API 包和对应的客户端

地址: <http://www.esunny.com.cn/index.php?a=lists&catid=49>

#### 3 如何获取 API 库

各个版本对应的API可从下表下载, 另提供客户端下载用于API程序的测试监控。除8.0API外, 所有开发包中均提供了易盛的API测试环境和测试授权码。用于易盛测试环境的API开发调试, 无需再申请授权码, 仅需要注册帐号或使用公共帐号。

市场	API类型	API开发包	更新日期	版本	测试账号	客户端下载
内盘	8.0API	点此下载			不提供测试环境	
	启明星API	点此下载	2018-4-18		点此注册	点此下载
外盘	3.0交易API	点此下载	2018-6-21	3.0.1.52	点此注册	
	3.0行情API	点此下载	2017-6-7		不提供测试环境	
	9.0行情API	点此下载	2017-12-11		用户名: ES 密码: 123456	
	9.0交易API	点此下载	2018-8-10	9.2.8.1	点此注册	点此下载

### 3.1.3 留意易盛模拟环境地址变化

地址变化通知:

<http://www.esunny.com.cn/index.php?a=show&catid=25&id=83>

	变更前	变更后
内盘API模拟交易联通	123.15.58.21:6160	61.163.243.173:6160
内盘API模拟交易电信	222.88.40.170:6160	123.161.206.213:6160
内盘API模拟行情联通	123.15.58.21:6161	61.163.243.173:6161
内盘API模拟行情电信	222.88.40.170:6161	123.161.206.213:6161
外盘API模拟交易联通	123.15.58.28:8383	61.163.243.173:8383
外盘API模拟交易电信	222.88.40.171:8383	123.161.206.213:8383
外盘API模拟行情联通	123.15.58.21:7171	61.163.243.173:7171
外盘API模拟行情电信	222.88.40.170:7171	123.161.206.213:7171

上述变更于2018年5月17日18:00之后生效，请API开发测试人员及时注意调整，由此给您带来的不便，请谅解。

### 3.1.4 对接开发测试

拿到易盛模拟账号后，可以开始对接易盛 9.0 环境进行测试。

请注意：在易盛测试环境测试后，上我司实盘前，最好也先在我司 9.0 模拟环境测试一下。

## 3.2.新永安北斗星 9.0 模拟环境

我司只有北斗星 9.0 的交易模拟环境,9.0 模拟行情请使用易盛 9.0 模拟行情,支持模拟撮合。

### 3.2.1. 申请交易授权

不论客户是使用 9.0 交易接口还是使用 3.0 交易接口，都需要申请交易授权，才可以对接交易 API。

3.2.1.1) 自行向易盛申请授权

进入该网页：<http://www.esunny.com.cn/index.php?a=lists&catid=49>

根据您使用的是 9.0 交易接口还是 3.0 交易接口填写不同的表格，跟易盛发邮件到“处理人”申请。一般为一个工作日。

## 4 如何接入易盛实盘系统 ▶

若API开发者需要接入期货公司的实盘服务器进行API交易或行情,需要向易盛公司申请实盘接入的API授权。**注:若您只是连接易盛提供的模拟环境做测试,无需申请实盘授权。**

**关于交易API:** 易盛的交易API授权均采用后台服务端+API客户端程序双重认证。在您得到易盛实盘API授权文件后,需要将授权文件提供给您实盘接入的期货公司,由期货公司参照授权为您的帐号配置开通API权限,同时您需要替换API程序中的授权码为正式的授权码。交易API的申请书和申请方式详见下方表格。

**关于行情API:**  
内盘的行情API授权随交易API一同发放,无需特别申请。

外盘3.0行情API授权,需要由期货公司先联系易盛公司获取,由期货公司将3.0行情API授权发放给客户。外盘9.0行情API,也由期货公司联系易盛公司开通对应的API帐号,用户拿到对应帐号后,从下方表格下载公共授权后可以直登录行情API获取行情。即外盘的行情API由期货公司来申请。

各个API对应的授权申请如下,请下载申请表发送至对应处理人的邮件即可:

市场	API类型	处理人	申请表	备注
内盘	8.0API	zd@esunny.cc	点此下载	
	启明星API	zyx@esunny.cc	点此下载	
外盘	3.0交易API	wjm@esunny.cc	点此下载	
	3.0行情api	用户所在期货公司		期货公司为客户发放API授权码
	9.0行情API	用户所在期货公司	点此下载授权	期货公司为客户申请行情账号,用此授权登陆API
	9.0行情API账号		点此下载	暂停申请
	9.0交易API	wjm@esunny.cc	点此下载	

### 2.2.1.2) 让我司代为申请交易授权

#### 1) 9.0API 交易授权申请

请填写《API 交易授权申请表/易盛外盘 9.0 交易系统 API 申请表.doc》,然后发到我司后台技术组邮箱,并说明申请 9.0 交易授权: [cxyhkit@yafco.com](mailto:cxyhkit@yafco.com)

#### 2) 3.0API 交易授权申请

请填写《API 交易授权申请表/易盛外盘 3.0 交易系统 API 申请表.doc》,然后发到我司后台技术组邮箱,并说明申请 3.0 交易授权: [cxyhkit@yafco.com](mailto:cxyhkit@yafco.com)

## 3.2.2. 申请我司 9.0 模拟账号和 API 绑定

### 3.2.2.1) 申请我司 9.0 模拟账号

填写表格《UAT-客服部文件/F-CSD-10 中國新永安期貨模擬交易端申請書.pdf》,并发邮件到我司后台技术组邮箱: [cxyhkit@yafco.com](mailto:cxyhkit@yafco.com) 说明申请 9.0 模拟账号,或者直接在我司官网联系我司客服 QQ 说明。

### 3.2.2.2) 申请 9.0 模拟账号与 API 交易授权绑定

需要把交易授权 PDF 文件随着模拟账号申请表一并提供给我司,做 API 绑定,且开发时,请在您程序中使用该份授权文件的授权码。

## 3.2.3. 根据易盛 API 接口进行开发测试

### 3.2.3.1) 程序包下载

1) 使用我司包里的 API

2) 去易盛官网下载最新的 API 包

地址: <http://www.esunny.com.cn/index.php?a=lists&catid=49>

根据自身对接接口下载对应最新 API 包

### 3 如何获取 API 库

各个版本对应的API可从下表下载, 另提供客户端下载用于API程序的测试监控。除8.0API外, 所有开发包中均提供了易盛的API测试环境和测试授权码。用于易盛测试环境的API开发调试, 无需再申请授权码, 仅需要注册帐号或使用公共帐号。

市场	API类型	API开发包	更新日期	版本	测试账号	客户端下载
内盘	8.0API	<a href="#">点此下载</a>			不提供测试环境	
	启明星API	<a href="#">点此下载</a>	2018-4-18		<a href="#">点此注册</a>	<a href="#">点此下载</a>
外盘	3.0交易API	<a href="#">点此下载</a>	2018-6-21	3.0.1.52	<a href="#">点此注册</a>	
	3.0行情API	<a href="#">点此下载</a>	2017-6-7		不提供测试环境	
	9.0行情API	<a href="#">点此下载</a>	2017-12-11		用户名: ES 密码: 123456	
	9.0交易API	<a href="#">点此下载</a>	2018-8-10	9.2.8.1	<a href="#">点此注册</a>	<a href="#">点此下载</a>

3.2.3.2) 根据接口进行开发和测试

## 3.2.4. 我司北斗星 9.0 模拟客户端下载

地址: [http://www.yafco.com.hk/upfile/admin/file/20190305/20190305114714\\_75305.zip](http://www.yafco.com.hk/upfile/admin/file/20190305/20190305114714_75305.zip)

客户服务  
CUSTOMER SERVICE

开户指南

线上开户

模拟交易申请

出入金流程

银行账户 +

**软件下载 -**

期货软件

证券软件

常用表格

常见问题 +

期货软件

标准交易软件 其他工具 程序化交易 **模拟交易**

北斗星9.0模拟极星客户端 **NEW**  
版本: 9.3.30 更新时间: 2019-01-22

该软件9.3客户端对接的是北斗星9.0后台模拟系统

[软件下载](#)

[说明书](#)

## 四. 实盘

### 4.1. 实盘 9.0 行情账号申请

9.0 实盘行情是需要另外申请行情账号的。申请需要提供下面 2 个信息:

4.1.1) 填写申请表格

《API 客户信息登记表.docx》

4.1.2) 注意是否需要 ICE 行情并确认付费事宜

需要说明是否需要 ICE 行情，如果需要 ICE 行情需要列明要哪几个交易所；ICE 交易所行情订阅费用为 118 美元/交易所/月，如需要 ICE 行情也请注明同意付费，新永安将每月从期货账户中进行扣取。

4.1.3) 联系我司官网客服 QQ，并提交上面行情申请资料

## 4.2.9.0 交易接口授权申请

如果还未申请交易授权，见[\[申请交易授权\]](#)，按需申请交易授权

## 4.3. 9.0 接口实盘绑定申请

4.3.1) 填写表格

《Pro-客服部文件/F-CSD-15 易盛外盘交易系统应用编程接口使用申请书.pdf》

《Pro-客服部文件/F-CSD-19 易盛外盘交易系统应用编程接口使用承诺书.pdf》

4.3.2) 随申请表格以及交易授权 PDF 文件一起发送给我司客服部进行绑定处理

## 4.4.9.0 行情账号以及交易授权续期

见[\[申请交易授权\]](#)，注意，填写表格时，需要填写原应用程序 ID 或者告知我司哪个授权需要延期申请

## 4.5. 我司客服部联系方式

1) 官网 QQ

地址：<http://www.yafco.com.hk/>



安全 | www.yafco.com.hk

期货 FUTURES | **证券 SECURITIES** | 资产管理 ASSET MANAGEMENT | 开户指南 A/C OPENING GUIDE | 线上开户 ONLINE A/C OPENING | 款项提存 DEPOSIT | 软件下载 SOFTWARE | 常见问题 FAQ

**研究中心 RESEARCH CENTER**

市场快讯 | 期货策略 | 股市速递 | 证券研究 MORE+

**最新公告 NOTICE**

期货公告 | 证券公告 MORE+

**股市速递 STOCK MARKET DAILY NOTES** MORE+

**合约细则及保证金要求 CONTRACT DETAILS AND MARGIN REQUIREMENTS** MORE+

期货 | 证券

客服热线 4008-418-508  
 下载热线 (852)2529 7066  
 预约开户  
 回到顶部

2) 客服热线电话：4008-418-508