SV.Cafe API Doc

联系 `hi@sv.cafe` 获取 app_id 和 app_secret，同时设置用于事件通知（如用户付款等）的回调地址（如需要）。

请务必保证 app_secret 私密，推荐 app_secret 只在服务端存储。

新建订单

GET https://sv.cafe/api/v1/orders/new

参数列表

参数	解释
app	必填，平台分配的 app_id
cny	订单金额，整数，范围 [1, 5000]，单位“元”
satoshi	按 BSV 数量下单，整数，范围 [100000, 400000000]，单位“聪”
address	必填，BSV 收款地址
comment	可选，用户自定义的信息
timestamp	必填，10 位 UNIX 时间戳，调试时为了关闭校验可以用 app_id 的值代替
signature	必填，将 app、cny 或 satoshi、address、timestamp、app_secret 参数的值**按顺序**拼接后 MD5

注意：参数 cny 和 satoshi 必须且只能有一个存在。如果按金额下单，则只传递 cny，如果按 BSV 数量下单，则只传递 satoshi。

返回值

调用成功，结果如下

```
{
    "r": "ok",
    "order": {
        "order_id": "20200503163204175334",
        "cny": 50,                                              # 订单金额
        "price": 1526.62,                                       # BSV 单价
        "receive_address": "17wgraDiNkQsvKpUnC65spkbxPx3tMZWAM", # 用户传递的收款地址
        "raw_address": "17wgraDiNkQsvKpUnC65spkbxPx3tMZWAM",     # 解码后的收款地址
        "satoshi": 3111448,                                     # 订单对应的 BSV 数量
        "status": "created",                                    # 订单状态，created、paid、closed
        "wechat_qr": "https://xxx.com/xxx.png",                 # 微信付款二维码
        "wechat_url": "https://xxx.com/xxx",                    # 微信付款链接，只有在微信里打开时才有效
        "alipay_qr": "https://xxx.com/xxx.png",                 # 支付宝付款二维码，在订单金额大于 1000 时该字段为 null
        "alipay_url": "https://xxx.com/xxx",                    # 支付宝付款链接，在订单金额大于 1000 时该字段为 null
        "comment": null,                                        # 用户自定义信息
        "txid": null                                            # 订单对应的 TXID
    }
}
```

可能的 status 值：

- created，订单已创建，用户未付款
- paid，用户已付款，订单正在处理（可能是正在发送 BSV，也可能是支付超时）
- closed，订单结束，用户成功付款，BSV 成功发出

注意：订单如果在下单 180 秒后才支付，会进入支付超时逻辑，需人工干预。

调用失败，结果如下：

```
{
    "r": "invalid_signature"
```

```
}
```

可能的 r 值：

- ok
- invalid_arguments
- invalid_app_id
- invalid_cny
- invalid_satoshi
- invalid_timestamp
- invalid_address
- invalid_signature
- unknown_exception

查询订单

GET https://sv.cafe/api/v1/orders/{order_id}

返回值

调用成功，结果如下

```
{
    "r": "ok",
    "order": {
        "order_id": "20200503163204175334",
        "cny": 50,                                              # 订单金额
        "price": 1526.62,                                       # BSV 单价
        "receive_address": "17wgraDiNkQsvKpUnC65spkbxPx3tMZWAM", # 用户传递的收款地址
        "raw_address": "17wgraDiNkQsvKpUnC65spkbxPx3tMZWAM",    # 解码后的收款地址
        "satoshi": 3111448,                                     # 订单对应的 BSV 数量
        "status": "closed",                                     # 订单状态，created、paid、closed
        "wechat_qr": "https://xxx.com/xxx.png",                 # 微信付款二维码
        "wechat_url": "https://xxx.com/xxx",                    # 微信付款链接（只能在微信里打开）
        "alipay_qr": "https://xxx.com/xxx.png",                 # 支付宝付款二维码，在订单金额大于 1000 时该字段为 null
        "alipay_url": "https://xxx.com/xxx",                    # 支付宝付款链接，在订单金额大于 1000 时该字段为 null
        "comment": null,                                        # 用户自定义信息
        "txid": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"              # 订单对应的 TXID
    }
}
```

调用失败，结果如下：

```
{
    "r": "invalid_order_id"
}
```

可能的 r 值：

- ok
- invalid_order_id
- unknown_exception

回调示例

如果设置了事件回调地址为 `https://example.com/callback`，当用户付款后，会触发一次 GET 回调，超时时间 2 秒，请求为

```
https://example.com/callback?event=paid_ok&order_id=20200503163204175334&cny=50&price=1526.62&receive_address=17wgraDiNkQsvKpUnC65spkbxPx3tMZWAM&raw_address=17wgraDiNkQsvKpUnC65spkbxPx3tMZWAM&satoshi=3111448&txid=449740bb71a1128029035acd95790d1ec6cfff66d451a5aa5f30b2e5505a85a5&timestamp=1598346462&signature=3783b0772f11c556cb6a623dd273d788
```

可能的 event 值：

- paid_ok，订单正常结束
- verify_failure，用户已付款，但订单信息不匹配，需要人工干预
- paid_timeout，用户已付款，但付款时已超时，需要人工干预
- unknown_exception

签名 `signature` 的值，由 app_id、event、order_id、timestamp、app_secret 的值**按顺序**拼接后 MD5 得到。

注意，回调时的 `txid` 参数只在 event 值为 `paid_ok` 才存在。

Python 签名代码示例

```python
import hashlib


def sign(*p):
    return hashlib.md5(u''.join(p).encode('utf8')).hexdigest().lower()


sign(
    'YOUR_APP_ID',
    '100',
    '17wgraDiNkQsvKpUnC65spkbxPx3tMZWAM',
    '1588495687',
    'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx',  # YOUR_APP_SECRET
)
```

付款交互示例

如果是桌面客户端集成，推荐直接展示二维码，让用户扫码支付。
演示视频

如果是手机端应用集成，用户无法扫码，推荐只使用支付宝付款链接，用户打开链接就可以直接跳转支付宝应用（微信付款链接无此效果）。
演示视频

参考

- Node.js 封装及示例代码 baryon/svcafe