

第三方平台集成 TeamViewer 分析

小组成员：宋定杰 171250628，李辰辉 171250645，梁斌 171830506，陈维烨 171250599

1. 应用概述

TeamViewer 提供一种基于网络的 API，以帮助企业访问数据以及管理 TeamViewer 帐户的方方面面。企业可使用 API 开发用于将 TeamViewer 功能集成到企业环境中的应用程序，或者开发供所有人使用的应用程序。不仅节省时间，还可充分利用各类应用程序和服务，让企业的业务实现飞跃式发展。

TeamViewer 提供以下服务：

- 用户管理
与 TeamViewer 公司资料相关的方法（如创建新公司成员、修改成员信息）。
- 组管理
与 TeamViewer 计算机和联系人列表中的组相关的方法（如创建组、共享组）
- 会话管理
与 TeamViewer 远程控制会话相关的方法（如创建新会话、修改会话信息）。
- 连接报告
与 TeamViewer Management Console 中的连接报告相关的方法（如列出连接报告、导出报告）。
- 在线会议
与 TeamViewer 会议相关的方法（如安排会议、启动自发会议）
- 计算机和联系人
与 TeamViewer 计算机和联系人列表相关的方法（例如查看哪些人在线，与某联系人或设备开始远程控制会话，或从目标的地址簿批量导入新联系人）。

2. 集成逻辑

第三方平台集成 TeamViewer 包括功能集成与表示集成。

2.1 TeamViewer 实现功能集成

TeamViewer 提供对外的一组 API，其他软件系统可利用此 API 将 TeamViewer 集成到自己的系统和应用程序当中，实现可定制化的功能模块。

在 TeamViewer 提供的 API 中，有完善的身份验证流程和错误处理等，而这些非功能性 API 都是为了提供一个完善且安全的功能服务所必须的。在我们的应用程序中，通过调用 API，将其功能集成到自己的系统中，但是 API 的内部实现和数据库对外不透明，实现的是功能层级的集成，所以集成方只是对 TeamViewer 进行功能集成。

2.2 TeamViewer 实现表示集成

TeamViewer 的表示集成可分为使用 QT 开发的桌面客户端，提供 Android 设备使用的移动客户端与支持移动互联网的 WEB 客户端。以 TeamViewer 和 Android 的合作为例，TeamViewer 移动应用程序能够远程连接到 Android 设备上，进行设备的访问，其中的 TeamViewer Host 产品就能够永久访问和支持无人值守的 Android 移动设备或商业级设备，如销售终端设备、信息亭或数字标牌。

在这种情境下就相当于 TeamViewer Host 为用户提供了一个远程无人值守的 Android 移动设备或商业级设备的统一表示接口，以使用户对于这些远程设备来进行访问和操作，也就是将这些 Android 设备中不同的表示集成到 TeamViewer Host 中，用一个统一的表示方式来支持设备的访问，这就是其表示集成的体现点。

3. 实现机制

3.1 展示集成

3.1.1 概述

TeamViewer 在 13 版本之后桌面客户端界面使用 QT 进行开发，使用 QT 可以为流行的几乎所有平台提供近乎一样的展示界面，例如 Windows、Mac OS、Linux、iOS、Android、Windows Phone、Chrome OS，给予了一致的表示方式。TeamViewer 除了桌面端应用也为用户提供了网页端应用，便于用户在更多可支持 WEB 浏览器的设备终端中使用 TeamViewer 的功能。TeamViewer 的网页端提供了较多桌面端的功能，如用户管理、远程设备监控、备份、终端保护等。

3.1.2 范例

桌面客户端



网页端



3.2 功能集成

功能集成部分以账户关系、用户管理、小组管理、会话管理为例。

3.2.1 Account Management 账户关系

账户关系部分的主要功能有获取账户数据与更新账户数据, 其中获取账户数据功能具体为检索与访问令牌关联帐户的帐户信息, 更新账户数据功能具体为更改与访问令牌关联帐户的帐户信息。访问令牌需要提前通过专用的授权接口获取。以获取账户数据功能为例:

获取账户数据:

接入 url:

/api/v1/account

接入方式:

GET 方式

返回值举例:

返回:

HTTP/1.1 200 OK

Content Type: application/json

```
{
  "userid": "u1234567",
  "email": "jdoe@example.com",
  "name": "John Doe",
  "company_name": "John's Company",
  "email_validated": true
}
```

```
}
```

接入类型：

系统调用：直接获取账户信息是一种系统间的调用接口

3.2.2 User Management 用户管理

用户管理部分的主要功能包括检索用户列表、创建新的公司成员、获取一个用户的信息与修改用户个人信息。以检索用户列表为例：检索用户列表功能通过传入的 email 地址、用户名、权限筛选出一个公司内部的所有用户的信息；

检索用户列表**接入 url：**

/api/v1/users?full_list=true

接入方式：

GET 方式

返回值举例：

发送请求：GET /api/v1/users?full_list=true

返回：

HTTP/1.1 200 OK

Content-Type: application/json

```
{ "users" : [  
  { "id" : "u1234567",  
    "name" : "Mighty Administrator",  
    "permissions" : "ManageAdmins, ManageUsers, ShareOwnGroups,  
EditFullProfile, ViewAllConnections, ViewOwnConnections, EditConnections,  
DeleteConnections, ManagePolicies, AssignPolicies,  
AcknowledgeAllAlerts, AcknowledgeOwnAlerts, ViewAllAssets, ViewOwnAssets,  
EditAllCustomModuleConfigs, EditOwnCustomModuleConfigs",  
    "active": true,  
    "email": "admin@example.com"  
  },  
  { "id" : "u2345678",  
    "name" : "John Doe",  
    "permissions" : "EditFullProfile"  
    "email": "user@example.com"  
  }  
]
```

接入类型：

系统调用：检索用户列表是一种系统间的调用接口

3.2.3 Group management 小组管理

一个公司中存在来自不同部门的人的很多不同的 group，需要对这些 group 进行创建、访问、删除等等的管理。以获取所有的组与创建组外部服务接口为例：

1. 获取所有的组

接入 url:

/api/v1/users/<ulD>/groups, 可选参数: name、shared

接入方式:

GET

返回值举例:

发送请求: GET /api/v1/groups?name=Test

返回:

HTTP/1.1 200 OK

Content-Type: application/json

```
{ "groups" : [
  { "id" : "g53235",
    "name" : "Testing",
    "shared_with" : [
      { "userid" : "u631645",
        "name" : "Ted",
        "permissions" : "read"}],
    "permissions" : "owned"
  },
  { "id" : "g425123356",
    "name" : "Test",
    "owner" : {
      "userid" : "u814464403",
      "name" : "Tester" },
    "permissions" : "readwrite"
  }
]
```

2. 创建组

接入 url:

/api/v1/users/<ulD>/groups, 参数: name、pcpolicy_id

接入方式:

POST

返回值举例:

发送请求:

POST /api/v1/groups

Content-Type: application/json

```
{ "name" : "Test",  
  "policy_id": "5d93f008-3b1f-8472-8555-3549d5a68092" }
```

返回:

HTTP/1.1 200 OK

Content-Type: application/json

Location: https://webapi.teamviewer.com/groups/g425123356

```
{  
  "id" : "g425123356",  
  "name" : "Test",  
  "permissions" : "owned",  
  "policy_id": "5d93f008-3b1f-8472-8555-3549d5a68092" }
```

3.2.4 Session management 会话管理

1. 获取 session

接入 url:

/api/v1/sessions, 参数: groupid、full_list

接入方式:

GET

返回值举例:

发送请求:

GET /api/v1/sessions?groupid=g425123356&full_list=true

返回:

```
{  
  "sessions" : [  
    {  
      "code" : "s31 328 542",  
      "groupid" : "g425123356",  
      "description" : "Hello, I have an issue with my printer, can you  
      please assist?",  
      "end_customer" : { "name" : "Peter Niedhelp", "email" : "help  
      me@example.com"  
      },  
      "assigned_userid" : "u7254190",  
      "end_customer_link" : https://getpilot.teamviewer.com/  
      "supporter_link" : https://getpilot.teamviewer.com/  
      "custom_api" : "{ \"ticket_id\" : \"535824\"  
      \"support_session_type\" : \"Pilot\"}, ...]  
    }
```

2. 创建 session

接入 url:

/api/v1/sessions, 参数: groupid、groupname

接入方式:

POST

返回值举例:

HTTP/1.1 200 OK

Content-Type: application/json

Location: <https://webapi.teamviewer.com/api/v1/sessions/s12-345-678>

```
{"code" : "s12-345-678",  
"state" : "open",  
"groupid" : "g425123356",  
"end_customer" : { "name" : "Max" },  
"description" : "I have a problemwith myspace bar.",  
"assigned_userid" : "u7254190",  
"end_customer_link" : "https://getpilot.teamviewer.com/s12345678",  
"supporter_link" : "https://getpilot.teamviewer.com/s12345678-  
asfg1234asfg",  
"valid_until" : "2013-10-30T12:03:29Z"  
}
```

工作流程

