TurboMeeting Integration Manual for Service Provider Partners

---

Version 3.5

This manual contains confidential information regarding RHUB products and services. Distribution of this manual must have the written permission from RHUB.

RHUB Communications, Inc.
2953 Bunker Hill Lane
Suite 400
Santa Clara, CA 95054

July, 2007

Index

TurboMeeting Integration Specification

1. Overview

1.1 Deployment Architect

This document describes the interfaces to integrate with RHUB TurboMeeting Web Conferencing system. The following figure shows the Internet based integration architect. The integration interfaces between TurboMeeting (TM) Server and Partner Server are based on URL calls.

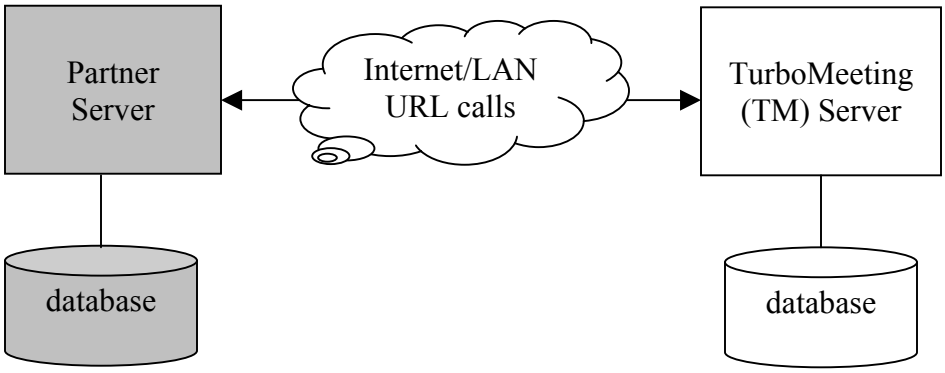

Figure 1. Integration Architect

This integration manual is for the partner that needs:

- Its own identity — private URL's for a user to start and join meetings
- Control of user authentication
- Control of scheduling meetings (optional)

The following figure shows the deployment architect for this type of integration.

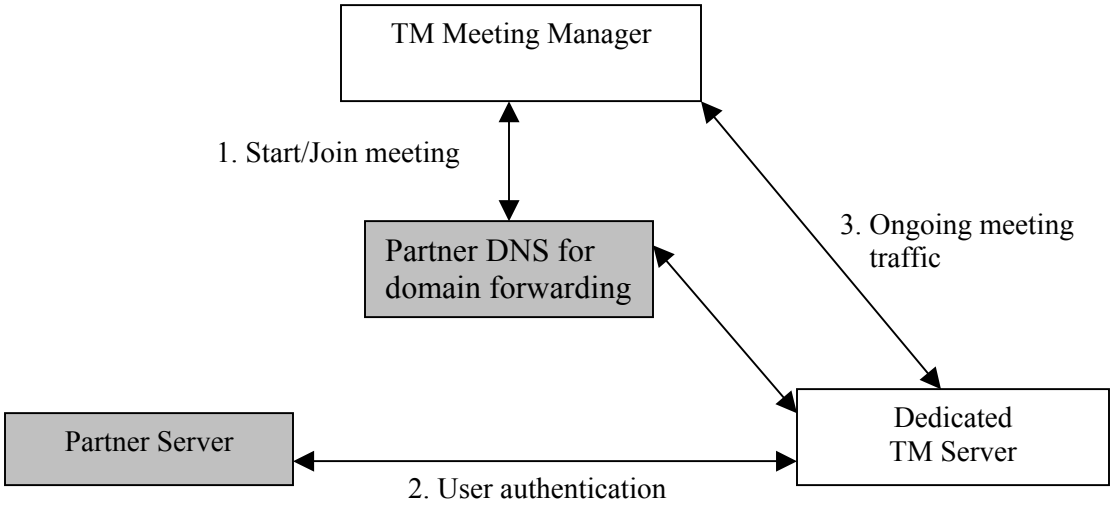

Figure 2. Deployment Architect

The table below shows the services provided by each party.

Services	RHUB	Partner
Dedicated TM Server to conduct meetings	√	
User authentication service		√
DNS service		√
Meeting scheduling services (optional)		√

1.2 Security on User Password and Meeting Password

To ensure the maximum security on user password and meeting passwords, here are the principles that this integration follows:

1. All passwords are transmitted over SSL between TM client and TM server, and between TM Server and Partner Server.
2. Meeting passwords for remote access are stored in SHA1 hash in the TM server database. Passwords for other types of meetings are stored in plain text in TM server database.

1.3 Get Started

To get started, contact RHUB to create a dedicated TM server. From there, partner can define the URLs used in this integration. Figure 3 shows the customization page.

2. Create Customer Account

- **URL call from Partner Server to TM Server:**
  http://TMServerIP/as/wapi/service/ create_account

- **Parameters in the URL**:

Variable Name	Comment
`service_company_uid`	Partner ID
`company_name`	Customer organization name
`Email`	Email of the main contact
`number_of_meeting_room`	The number (integer) of meeting rooms
`number_of_participant`	The number (integer) of participants per meeting room

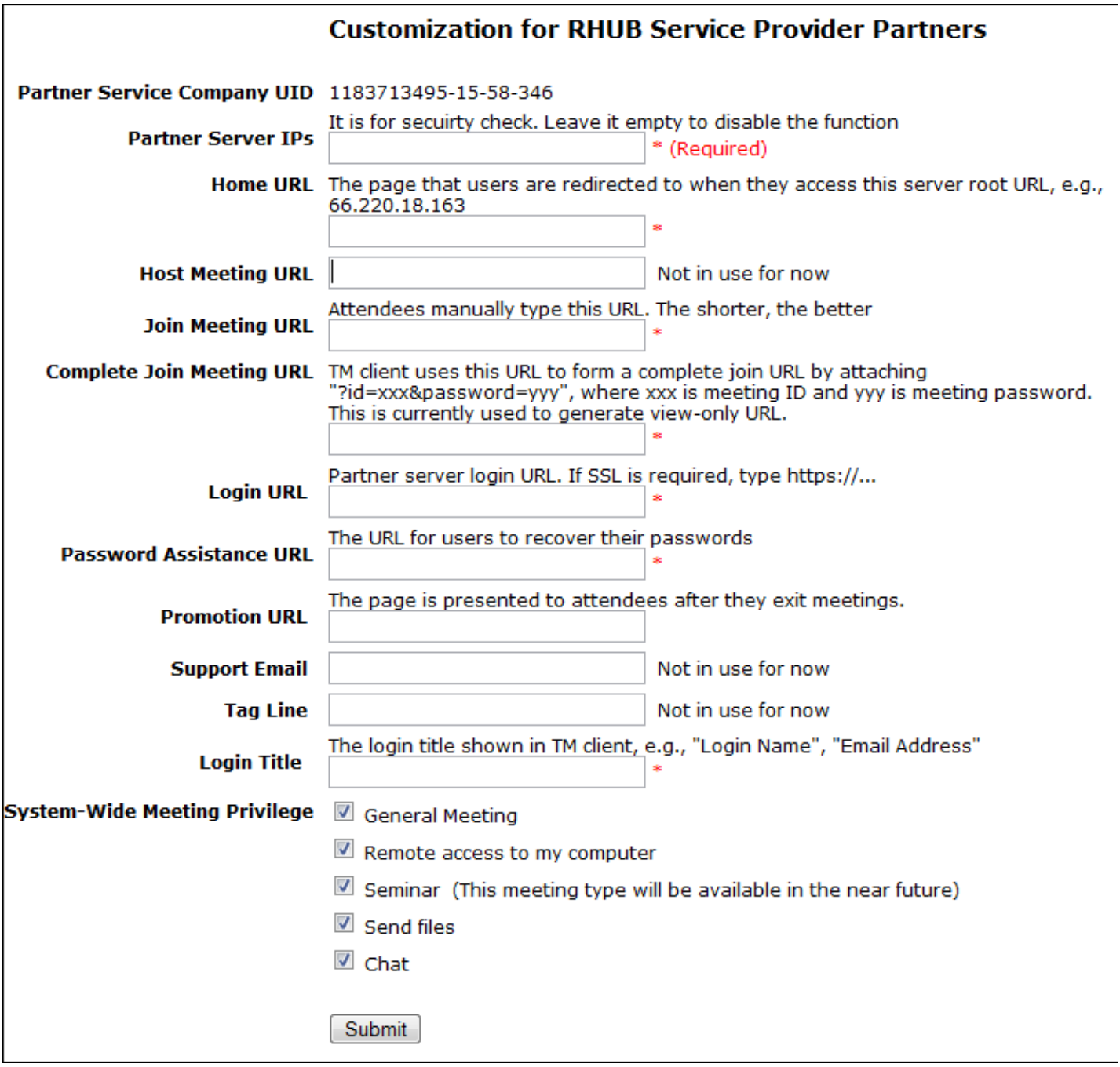

Figure 3. Customization for RHUB server provider partners

- **Return**

```
<__Return__>
      <__Status__>
            The status of execution
      </__Status__>
      <__Reason__>
            The reasons to explain failed execution
      </__Reason__>
      <__CompanyUid__>
            up to 36-character ID for the account created
      </__CompanyUid__>
</__Return__>
```

where “Status” values include:

- SUCCEED
- FAILED

If the Status is FAILED, “Reason” value will show the reasons of the failed execution and the “company_uid” value is empty.

3. Update Customer Account

- **URL call from Partner Server to TM Server**: http://TMServerIP/as/wapi/service/update_account

- **Parameters in the URL**:

Variable Name	Comment
service_company_uid	Partner ID
company_name	Customer organization name
email	Email of the main contact
number_of_meeting_room	The number (integer) of meeting rooms
number_of_participant	The number (integer) of participants per meeting room
company_uid	The user account ID returned from “Create Customer Account”

- **Return**

```
<__Return__>
      <__Status__>
            The status of execution
      </__Status__>
      <__Reason__>
            The reasons to explain failed execution
      </__Reason__>
```

```
</__Return__>
```

where “Status” values include:

- SUCCEED
- FAILED

If the Status is FAILED, “Reason” value will show the reasons of the failed execution.

4. Delete Customer Account

- **URL call from Partner Server to TM Server:**
  http://TMServerIP/as/wapi/service/delete_account

- **Parameters in the URL**:

Variable Name	Comment
service_company_uid	Partner ID
company_uid	Customer ID

- **Return**

```
<__Return__>
      <__Status__>
            The status of execution
      </__Status__>
      <__Reason__>
            The reasons to explain failed execution
      </__Reason__>
</__Return__>
```

where “Status” values include:

- SUCCEED
- FAILED

If the Status is FAILED, “Reason” value will show the reasons of the failed execution.

5. Query Customer Account Status

- **URL call from Partner Server to TM Server:**
  http://TMServerIP/as/wapi/service/query_account

- **Parameters in the URL**:

Variable Name	Comment
service_company_uid	Partner ID
company_uid	Customer ID

- **Return**

```
<__Return__>
      <__Status__>
            The status of execution
      </__Status__>
      <__Reason__>
            The reasons to explain failed execution
      </__Reason__>
      <__CompanyName__>
            Company name
      </__CompanyName__>
      <__Email__>
            Email
      </__Email__>
      <__NumberOfMeetingRoom__>
            Number of meeting rooms
      </__NumberOfMeetingRoom__>
      <__NumberOfParticipant__>
            Number of participants
      </__NumberOfParticipant__>
</__Return__>
```

where “Status” values include:

- SUCCEED
- FAILED

If the Status is FAILED, “Reason” value will show the reasons of the failed execution. If the user account has been deleted, FAILED will be returned.

6. Query All Valid User Accounts

- **URL call from Partner Server to TM Server:**
  http://TMServerIP/as/wapi/service/query_all_account

- **Parameters in the URL**:

Variable Name	Comment
service_company_uid	Partner ID

- **Return**

```
<__Return__>
      <__Status__>
            The status of execution
```

```
        </__Status__>
        <__Reason__>
              The reasons to explain failed execution
        </__Reason__>
        <__CompanyList__>
              <__Company__>
                    <__CompanyName__>
                          Company name
                    </__CompanyName__>
                    <__CompanyUID__>
                          Company UID
                    </__CompanyUID__>
                    <__Email__>
                          Email
                    </__Email__>
                    <__NumberOfMeetingRoom__>
                          Number of meeting rooms
                    </__NumberOfMeetingRoom__>
                    <__NumberOfParticipant__>
                          Number of participants
                    </__NumberOfParticipant__>
              </__company__>
        </__CompanyList__>
</__Return__>
```

where “Status” values include:

- SUCCEED
- FAILED

If the Status is FAILED, “Reason” value will show the reasons of the failed execution. If the user account has been deleted, FAILED will be returned.

7. User Authentication

- **URL call from TM Server to Partner Server:**
  http://partner-defined-url

- **Parameters in the URL**:

Variable Name	Mandatory	Comment
`Email`	Yes	User email or user name
`password`	Yes	User password
`service_company_uid`	Yes	Partner ID
`pass_through`	Optional < 256 chars	A value came from Partner Server when a user starts a scheduled or unscheduled meeting at Patner Server site.

The above parameters will be sent to the partner server via POST, not via GET. This is the only API using POST to ensure the best security practice.

- **Return**

```
<__Return__>
      <__Status__>
            The status of execution
      </__Status__>
      <__Reason__>
            The reasons to explain failed execution
      </__Reason__>
      <__CompanyUid__>
            This must be a valid user account ID generated by TM
            Server (see Section 2)
      </__CompanyUid__>
      <__FirstName__>
            First name of the user
      </__FirstName__>
      <__LastName__>
            Last name of the user
      </__LastName__>
      <__UserID__>
            The user ID in partner server
      </__UserID__>
      <__QueryScheduledMeetingURL__>
            The URL to query the list of scheduled meetings. The
            URL should be invalid when this login user-session is
            expired. Keep the session alive for at least 20
            minutes. If this field is empty, TM client & server
            will manage the scheduled meetings.
      </__QueryScheduledMeetingURL__>
</__Return__>
```

where “Status” values include:

- SUCCEED
- FAILED

If the Status is FAILED, “Reason” value will show the reasons of the failed execution.

8. Schedule a Meeting

- **URL call from Partner Server to TM Server:**
  http://TMServerIP/as/wapi/service/remote_schedule

- **Parameters in the URL**:

Variable Name	Mandatory	Comment
service_company_uid	Yes	Partner ID

company_uid	Yes	Customer ID
user_id	Yes	User ID at partner server
meeting_type	Yes	0: GENERAL_MEETING 2: REMOTE ACCESS 3: SEMINAR
password	Yes	Meeting password, un-encrypted

Note that user_id has to agree with the __UserID__ returned during the user authentication step.

- **Return**

```
<__Return__>
      <__Status__>
            The status of execution
      </__Status__>
      <__Reason__>
            The reasons to explain failed execution
      </__Reason__>
      <__MeetingID__>
            The meeting ID for the scheduled meeting
      </__MeetingID__>
</__Return__>
```

where “Status” values include:

- SUCCEED
- FAILED

If the Status is FAILED, “Reason” value will show the reasons of the failed execution.

9. Update a Scheduled Meeting

- **URL call from Partner Server to TM Server:**
  http://TMServerIP/as/wapi/service/update_scheduled_meeting

  At the present, the API is used to update the meeting password that is changed in partner server.

- **Parameters in the URL**:

Variable Name	Mandatory	Comment
service_company_uid	Yes	Partner ID
company_uid	Yes	Customer ID
user_id	Yes	User ID at partner server

meeting_type	Yes	0: GENERAL_MEETING 2: REMOTE ACCESS 3: SEMINAR
password	Yes	Meeting password, un-encrypted
meeting_id	Yes	Meeting ID

Note that user_id has to agree with the __UserID__ returned during the user authentication step.

- **Return**

```
<__Return__>
      <__Status__>
            The status of execution
      </__Status__>
      <__Reason__>
            The reasons to explain failed execution
      </__Reason__>
</__Return__>
```

where "Status" values include:

- SUCCEED
- FAILED

If the Status is FAILED, "Reason" value will show the reasons of the failed execution.

10. Start an Unscheduled Meeting

For the users who have downloaded TM Meeting Manager, they would most likely start a meeting by clicking the TM Meeting Manager icon on their desktop.

For the users who have not downloaded the meeting manager, they should be asked to go to the host-meeting page hosted at the partner server. After users click some button or link on the host-meeting page to start a meeting, the partner server should redirect users to the following URL.

- **URL**: http://TMServerIP/as/wapi/goto_downloader

- **Parameters in the URL**:

Variable Name	Mandatory	Comment
role	Yes	Value: host
email	Yes	User login name
user_password	Yes	User login password
pass_through	Optional	This will be passed back to

	< 256 chars	the Partner Server during the user authentication. This value typically is the user session ID at the Partner Server website. It can serve for additional security protection. For example, Patnter Server can use this parameter together with email and password to authenticate the user and make sure that the user has logined to Partner Server before start a meeting using this one-click URL.

For example,

http://192.168.1.100/as/wapi/goto_downloader?role=host&email=jdoe&user_password=password&pass_through=8218921n12f1212891z1m21n2

Note that in order for a user to start a meeting, TM Server will call Partner Server to authenticate the user. Therefore, the URL should provide user login name and password. Since the TM server does not provide SSL service with a publically authenticated SSL certificate, the URL should be based on HTTP rather than HTTPS.

To protect the user password shown in the HTTP URL, partner can use an encrypted password. TM server will pass the encrypted password to the partner server during the user authentication.

11. Start a Scheduled Meeting

For the users who have downloaded TM Meeting Manager, they would most likely start a meeting by clicking the TM Meeting Manager icon on their desktop and then clicking one of the scheduled meetings shown in the TM client to start a scheduled meeting. Partner Server does not need to anything except responding the call of `__QueryScheduledMeetingURL__` by TM Server. The URL is returned during User Authentication.

For the users who have not downloaded the meeting manager, they should be asked to go to the host-meeting page hosted at the partner server. After users click some button or link on the host-meeting page to start a scheduled meeting, the partner server should redirect users to the following URL.

- **URL**: http://TMServerIP/as/wapi/goto_downloader

- **Parameters in the URL**:

Variable Name	Mandatory	Comment
`role`	Yes	Value: host
`Email`	Yes	User login name
`password`	Yes	Meeting password
`user_password`	Yes	User login password
`pass_through`	Optional < 256 chars	This will be passed back to the Partner Server during the user authentication. This value typically is the user session ID at the Partner Server website. It can serve for additional security protection. For example, Patnter Server can use this parameter together with email and password to authenticate the user and make sure that the user has logined to Partner Server before start a meeting using this one-click URL.

For example,

http://192.168.1.100/as/wapi/goto_downloader?role=host&email=jdoe&meeting_id=4334-8490&password=www&user_password=password&pass_through=81829129hqwy9218ha5qwj127ekqw

Note that in order for a user to start a meeting, TM Server will call Partner Server to authenticate the user. Therefore, the URL should provide user login name and password. Since the TM server does not provide SSL service with a public ally authenticated SSL certificate, the URL should be based on HTTP rather than HTTPS.

To protect the user password shown in the HTTP URL, partner can use an encrypted password. TM server will pass the encrypted password to the partner server during the user authentication.

12. Join A Meeting

For the users who have downloaded TM Meeting Manager, they would most likely join a meeting by clicking the TM Meeting Manager icon on their desktop.

For the users who have not downloaded the meeting manager, they should be asked to go to the join-meeting page hosted at the partner server. After users fill a form with the meeting info such as meeting ID and password, and submit it to join a meeting, the partner server should redirect users to the following URL.

- **URL**: http://TMServerIP/as/wapi/goto_downloader

- **Parameters in the URL**:

Variable Name	Mandatory	Comment
`role`	Yes	Value: attendee
`name`	Yes	Attendee name
`password`	Yes	Meeting password
`submit`	Yes	Value: submit

For example,

http://192.168.1.100/as/wapi/goto_downloader?role=attendee&name=john_doe&meeting_id=4334-8490&password=password&sumbit=submit

13. Query Scheduled Meetings

- **URL call from TM Server to Partner Server:**
  http://partner-defined-url

  This URL is user-specific, which is the value of `<__QueryScheduledMeetingURL__>` returned from User Authentication.

- **Return**

```
<__Return__>
      <__Status__>
            The status of execution
      </__Status__>
      <__Reason__>
            The reasons to explain failed execution
      </__Reason__>
      <__ScheduledMeeting__>
         <__Meeting__>
            <__MeetingId__>
               The scheduled meeting ID, which is acquired in
               Section 6.
            </__MeetingId__>
            <__ScheduledStartTime__>
               The scheduled start time (YYYY/MM/DD HH:MM)
            </__ScheduledStartTime__>
            <__MeetingTopic__>
```

```
            The meeting subject
         </__MeetingTopic__>
         <__MeetingPassword__>
            Meeting password (un-encrypted)
         </__MeetingPassword__>
      </__Meeting__>
   </__ScheduledMeeting__>
</__Return__>
```

where “Status” values include:

- SUCCEED
- FAILED

If the Status is FAILED, “Reason” value will show the reasons of the failed execution.