Pexip One-Touch Join

Deployment Guide

Software Version 24.3

Document Version 24.b

December 2020

]pexip[

Contents

About One-Touch Join **6**
Enabling One-Touch Join 6
Supported G Suite editions 6
Supported Exchange environments 7
Exchange servers 7
Outlook clients 7
Supported endpoints 7
Cisco OBTP 7
Poly OTD 7
Supported meeting types 8
Supported number of endpoints and Conferencing Nodes 8
Pexip Infinity server requirements 8
One-Touch Join process and deployment overview **9**
Process overview 9
Administrator configures OTJ 9
End user sends invitation 9
OTJ provides endpoint with meeting information 9
Frequency and limitations on calendar requests 10
Locations, Conferencing Nodes and redundancy 10
Conferencing Nodes 10
Management Node 11
Network architecture, firewalls and web proxy 11
Conferencing Nodes 11
Management Node 11
Port usage 11
Permitting the service account to access calendars 13
Exchange integrations 13
G Suite integrations 13
Using One-Touch Join with personal endpoints and calendars 13
Configuring Exchange on-premises for One-Touch Join **14**
Prerequisites 14
Creating a service account 14
Configuring Application Impersonation on the service account 16
Creating a new Distribution Group 16
Configuring application impersonation 16
Enabling basic authentication 17

Configuring calendar processing on room resource mailboxes 17
Recommended configuration 17
Optional configuration 18
Checking calendar processing settings 18
Adding a One-Touch Join Exchange integration on Pexip Infinity 20
Next steps 20

Configuring Office 365 for One-Touch Join 21

Prerequisites 21
Creating a service account 21
Configuring Application Impersonation on the service account 22
Creating a new Distribution Group 22
Configuring application impersonation 23
Configuring calendar processing on room resource mailboxes 24
Recommended configuration 24
Optional configuration 24
Checking calendar processing settings 25
Enabling OAuth authentication 26
Adding a One-Touch Join Exchange integration on Pexip Infinity 29
Configuring the Exchange integration 29
Signing in to the service account 31
Next steps 32

Configuring G Suite for One-Touch Join 33

Prerequisites 33
Creating a Service Account 33
Sharing calendars externally 36
Creating a room resource 37
Configuring the room resource 38
Sharing individual calendars with the service account 38
Auto-accepting invitations 39
Allowing users to book resources 39
Updating the per-user request quota 40
Requesting an increase to API limits 41
Adding a One-Touch Join G Suite integration on Pexip Infinity 42
Next steps 43

Configuring Pexip Infinity for One-Touch Join 44

Prerequisites 44
Adding a One-Touch Join profile 44
Adding One-Touch Join endpoint groups 46

Adding One-Touch Join endpoints 47
Adding endpoints individually 47
Adding OTJ endpoints in bulk 48
Adding One-Touch Join meeting processing rules 50
Testing the rule 52
Next steps 53

Configuring endpoints to support One-Touch Join 54

Prerequisites 54
Configuring Cisco OBTP endpoints for OTJ 54
Hiding or changing the meeting subject 54
Configuring Poly OTD endpoints for OTJ 55
DNS records 56
Poly authentication 56
Configuring Poly RealPresence Group series 56
Configuring Poly Trio series 58
Configuring Poly HDX series 61
Configuring Poly Studio X series and Poly G7500 series 63
Configuring Poly Debut series 64
Configuring Webex Edge endpoints for OTJ 65
Prerequisites 65

One-Touch Join meeting types and transforms 66

Fallback alias matching 66
Supported meeting types 66
Regex meeting type 68
Examples 69
Custom meeting type 69
Examples 70

Deploying a dedicated One-Touch Join platform 72

Minimum hardware requirements 72
Minimum Pexip Infinity platform configuration 72
One-Touch Join configuration 73

Scheduling and joining meetings using One-Touch Join 74

Viewing One-Touch Join status 75

Viewing One-Touch Join meetings 75
Viewing One-Touch Join endpoints 75

Configuring G Suite for domain user authorization 77

Prerequisites 77

Enabling authorization using OAuth 77
Creating a room resource 81
Configuring the room resource 82
Sharing individual calendars with the authorization user 82
Auto-accepting invitations 83
Allowing users to book resources 83
Updating the per-user request quota 84
Requesting an increase to API limits 85
Adding a One-Touch Join G Suite integration on Pexip Infinity 86
Configuring the G Suite integration 86
Authorizing calendar access 87
Next steps 89

Troubleshooting One-Touch Join 90

About One-Touch Join

Pexip Infinity's One-Touch Join (OTJ) feature integrates support for existing "click to join" videoconferencing endpoint workflows into your Pexip Infinity deployment. With One-Touch Join, when users schedule a meeting in Microsoft Outlook or Google Calendar and include in the meeting invitation a room that contains a supported Cisco or Poly endpoint, the endpoint will display a **Join** button just before the meeting is scheduled to begin. Participants can then simply walk into the room and select the button, and the endpoint will automatically dial in to the meeting.

One-Touch Join is available as an optional licensed feature within the Pexip Infinity platform.

In most cases, One-Touch Join will be implemented as a feature within a wider Pexip Infinity deployment, and run on Conferencing Nodes alongside other Pexip Infinity services. However, you can also set up separate OTJ locations within your deployment that contain Conferencing Nodes used solely for One-Touch Join. A third option appropriate in some situations is to implement a separate Pexip Infinity deployment purely for One-Touch Join, for example if you are a Pexip Service customer wishing to use One-Touch Join, or you are a large enterprise wishing to separate the resources used for your One-Touch Join deployment. For more information, see Deploying a dedicated One-Touch Join platform.

Enabling One-Touch Join

All Conferencing Nodes are capable of running One-Touch Join, although the service will only come into active operation on a node when the location the node is in is associated with a One-Touch Join Endpoint Group.

Enabling the Pexip One-Touch Join service within your Pexip Infinity deployment involves the following steps, each described in separate topics:

1. Depending on which calendar/email service is used in your environment, do one of the following:
   - Configuring G Suite for One-Touch Join
   - Configuring Exchange on-premises for One-Touch Join
   - Configuring Office 365 for One-Touch Join
2. Configuring Pexip Infinity for One-Touch Join
3. Configuring endpoints to support One-Touch Join
4. Viewing One-Touch Join status

For an overview of the process and general deployment and network considerations for One-Touch Join, see One-Touch Join process and deployment overview.

For a guide for end users, see Scheduling and joining meetings using One-Touch Join.

Supported G Suite editions

Pexip One-Touch Join is supported in the following G Suite environments:

- G Suite Basic
- G Suite Business
- G Suite Enterprise

Supported Exchange environments

Pexip One-Touch Join is supported in the following Microsoft Exchange environments:

Exchange servers

- Office 365
- Exchange 2013 (with the latest updates)
- Exchange 2016 (with the latest updates)
- Exchange 2019 (with the latest updates)

Outlook clients

Meetings scheduled in all Outlook clients are supported. Note that different third-party Outlook add-ins for different Outlook versions may format the join details for some meeting types slightly differently.

Supported endpoints

Endpoints used for One-Touch Join **must not** also be registered to the calendaring service on other systems such as the cloud-based Webex Hybrid Calendar Service, or Cisco TMS XE.

Cisco OBTP

Pexip Infinity One-Touch Join is supported on Cisco VTC endpoints that support Cisco One Button to Push (OBTP) and are running either TC or CE* software.

This includes:

- Cisco Webex Room series (Room, Room Kit)*
- Cisco C series (C20, C40, C60, C90)
- Cisco DX series (DX70, DX80)
- Cisco EX series (EX60, EX90)
- Cisco MX series (MX200, MX300, MX700, MX800)
- Cisco SX series VTC systems (SX10, SX20, SX80)

* Endpoints registered to Webex cloud must be using Webex Edge for Devices.

For information on how to configure these endpoints to support Pexip One-Touch Join, see Configuring Cisco OBTP endpoints for OTJ.

Poly OTD

Pexip Infinity One-Touch Join is supported on Poly VTC endpoints that support Poly One Touch Dial (OTD). This includes:

- Poly RealPresence Group series
- Poly Trio series
- Poly HDX series (unless Pexip Infinity has been deployed in a secure mode of operation - for more information, see Poly authentication); must be running a software version that supports NTLMv2 for calendaring, e.g. 3.1.11 or later
- Poly Studio X series
- Poly G7500 series
- Poly Debut series

For information on how to configure these endpoints to support Pexip One-Touch Join, see Configuring Poly OTD endpoints for OTJ.

Supported meeting types

This release of Pexip One-Touch Join can be used to join the following types of meetings:

- Pexip Infinity meetings (i.e. those scheduled using the VMR Scheduling for Exchange feature)
- Pexip Service meetings (i.e. those scheduled using the plugin available to Pexip Service users)
- Google Meet (for G Suite integrations only)
- Microsoft Teams
- Skype for Business
- Webex
- Zoom
- BlueJeans
- GoToMeeting

You can also create your own meeting processing rules for meeting types not listed above. For more information, see One-Touch Join meeting types and transforms.

Supported number of endpoints and Conferencing Nodes

The One-Touch Join feature will support up to 4,000 room resource calendars and associated endpoints. This applies to One-Touch Join both when integrated with a Pexip Infinity deployment (i.e. when running on Conferencing Nodes alongside other Pexip Infinity services), and as a deployment dedicated to One-Touch Join.

For **integrated One-Touch Join deployments** (i.e. where OTJ is being implemented as a feature within a wider Pexip Infinity deployment), a Pexip Infinity deployment with a single Conferencing Node per location should also support up to 170 OTJ room resource calendars and associated endpoints (although you may wish to include one or more additional Conferencing Nodes for redundancy). For large or busy deployments, you may need to add additional Conferencing Nodes per location to provide the additional capacity required when One-Touch Join is implemented — we recommend that you consult your Pexip authorized support representative for advice on your particular deployment.

These recommendations apply to Pexip Infinity deployments with one or two One-Touch Join Integrations. For deployments with multiple OTJ Integrations (for example, when implemented by service providers with multiple customers) we recommend a dedicated One-Touch Join deployment.

For **dedicated One-Touch Join deployments** of all sizes (i.e. up to the supported 4,000 room resource calendars and associated endpoints), we recommend one Conferencing Node for every 1,000 endpoints in a location (although you may wish to include one or more additional Conferencing Nodes for redundancy).

For information on the hardware requirements for OTJ, see Pexip Infinity server requirements.

Pexip Infinity server requirements

In most cases you will be enabling One-Touch Join within a new or existing Pexip Infinity deployment, and the One-Touch Join service can be run alongside other Pexip Infinity services on each Conferencing Node. Enabling One-Touch Join within most Pexip Infinity deployments (up to 170 endpoints — see Supported number of endpoints and Conferencing Nodes) will not significantly increase the processing requirements of the Management Node or Conferencing Nodes, therefore our standard server design guidelines still apply. However, if your deployment is expected to be particularly large or busy, we recommend that you consult your Pexip authorized support representative for advice.

For dedicated One-Touch Join deployments, see Minimum hardware requirements.

In both cases, we recommend that each Conferencing Node runs on a different VM host and uses different storage.

One-Touch Join process and deployment overview

This topic gives an overview of the process used by One-Touch Join to extract calendar information and provide it to endpoints, along with information on general deployment and network considerations.

Process overview

The general process from setting up One-Touch Join through to having the endpoint display a Join button at the start of a meeting is as follows:

Administrator configures OTJ

1. The administrator configures their G Suite, Exchange on-premises or Office 365 deployment to support Pexip Infinity One-Touch Join, and ensures that each physical meeting room that contains an endpoint to be used for One-Touch Join has an associated email address.
2. The administrator then configures One-Touch Join on the Pexip Infinity Management Node. This configuration is automatically replicated to the One-Touch Join service that runs on each Conferencing Node in the Pexip Infinity deployment.
3. Finally, the administrator configures their endpoints to support One-Touch Join.

End user sends invitation

When an end user wants to use a One-Touch Join room for a meeting, they create a meeting invitation in their usual way, using their usual client, ensuring that the room resource is added to the invitation.

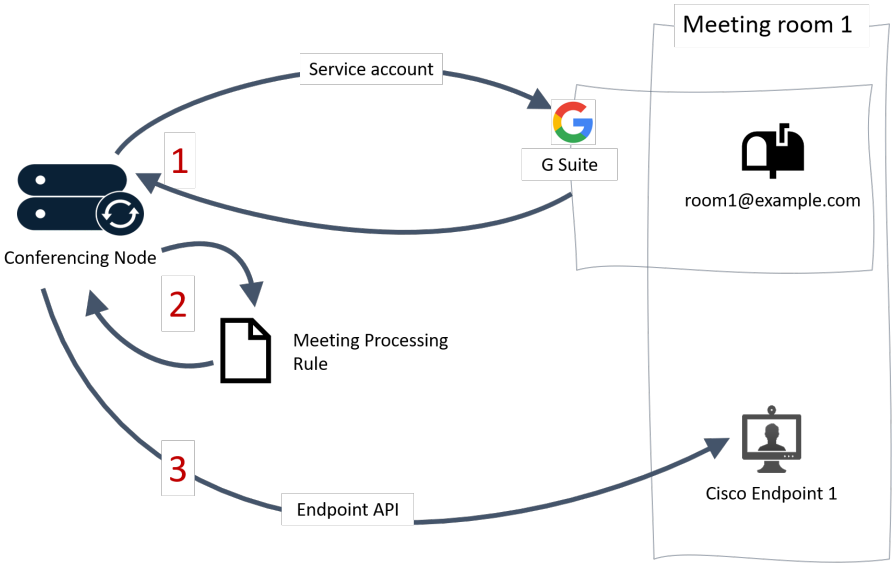
ⓘ Generally, rooms are added to a meeting invitation as a room resource, but One-Touch Join will also work if the room resource's email address is included in the list of invitees, or as a location.

OTJ provides endpoint with meeting information

1. Each meeting room resource has one Conferencing Node which will be its primary node. Periodically, One-Touch Join on the Conferencing Node connects to G Suite or Microsoft Exchange and uses the configured service account details to impersonate each room resource for which it is the primary node. For each room resource, One-Touch Join finds all meetings to which the room has been invited. By default, it does this for all meetings with a scheduled start time from one day in the past up to seven days in the future, but this range is configurable.
2. One-Touch Join parses the meeting invitation (in accordance with the relevant meeting processing rule) to obtain information about the meeting, which it uses to generate the alias that the endpoint will dial in order to join the meeting.
3. One-Touch Join then provides the meeting information to the endpoint that is associated with the room resource:
   - for Cisco endpoints, One-Touch Join pushes the meeting information to the endpoint
   - for Poly endpoints, the endpoint registers to the OTJ calendaring service on the Conferencing Node and periodically requests updated meeting information from the Conferencing Node.

   ⓘ More than one endpoint can be associated with a single room resource; in this case, all the endpoints will receive the same meeting information.
4. When the meeting is about to start, the endpoint will display a **Join** button; participants in the room simply click the button and the endpoint will dial in to the meeting.

The flow of information between the calendar/email service, One-Touch Join and the endpoint is shown in the following diagram (using G Suite and a Cisco endpoint as the example):



Frequency and limitations on calendar requests

The length of time taken for a meeting booked via Exchange or Google calendar to appear on the corresponding room endpoint depends on a number of factors, but is largely due to the number of endpoints in your deployment.

In general, for deployments of around 170 endpoints or fewer, the One-Touch Join service will poll room resource calendars with a maximum frequency of every 30 seconds. (It does not poll any more frequently than this to avoid impacting the performance of Conferencing Nodes.) Cisco endpoints will be updated after each poll; Poly endpoints will generally connect to the Conferencing Node to get updates every minute, but this will depend on the Poly configuration.

As you add more endpoints, One-Touch Join will reduce the frequency of requests correspondingly. For a deployment of 4,000 endpoints (the maximum supported number), endpoints will be updated around every 12 minutes. This is because both Microsoft Exchange and Google limit the number of API requests that can be made to their calendar services in a 24-hour period. It is possible to change the 24-hour quota to increase the frequency of endpoint updates in larger deployments, but note that doing so may impact the performance of the Conferencing Nodes, so you may need to consider deploying a dedicated One-Touch Join platform. We recommend you discuss larger deployments with your Pexip authorized support representative first.

- For G Suite deployments, you can change the 24-hour quota by Requesting an increase to API limits and then increasing the Maximum G Suite API requests, but this is a paid-for service.
- For Exchange deployments, you can change the 24-hour quota by increasing the Find Items Request Quota.

Locations, Conferencing Nodes and redundancy

Conferencing Nodes

All Conferencing Nodes in your deployment are capable of running One-Touch Join. However, the service will be in active operation on only those nodes that belong to a location that has been associated with a OTJ Endpoint Group (and when that Endpoint Group has been associated with an OTJ profile).

Within each such location, a maximum of five Conferencing Nodes will actively read room resource calendars and process meeting information. Responsibility for each room resource is spread across these nodes in order to balance the workload and provide redundancy. Should one node become unavailable (for example, if it is put into maintenance mode or loses connectivity), the other nodes will take over responsibility for its room resources.

However, if there are one or more Poly endpoints in the location, the One-Touch Join service on **all** nodes within the location will handle requests from Poly endpoints. Therefore round-robin DNS records are required for all nodes in a location that has Poly endpoints.

Note that if you put **all** Conferencing Nodes in a One-Touch Join location into maintenance mode, then none of the endpoints in the associated Endpoint Group will receive any updates (overflow locations are not used by One-Touch Join).

You can use existing system locations for One-Touch Join, in which case up to five Conferencing Nodes in that location will be actively operating One-Touch Join in addition to their core functions. Alternatively, you can set up system locations that will be used specifically for One-Touch Join. These can be in the same physical locations as your existing Conferencing Nodes, but their resources will be dedicated to One-Touch Join.

Management Node

As with other Pexip Infinity services, the One-Touch Join service will continue to function if the Management Node goes offline, although you will not be able to make any changes to the configuration of the service during this time.

For deployments using OAuth, the Management Node periodically refreshes OAuth tokens on behalf of Conferencing Nodes, so eventually (after some weeks) these nodes may become unable to authenticate with Exchange / G Suite.

Network architecture, firewalls and web proxy

Conferencing Nodes

Each Conferencing Node used for One-Touch Join requires a persistent connection to either G Suite or the Microsoft Exchange server (either directly or via a web proxy), and must be able to sign in to it as the service account.

If you are using OAuth for Exchange, or a G Suite integration, each Conferencing Node must be able to reach the OAuth token endpoint, either directly or via a web proxy.

Each Conferencing Node must be able to access the Cisco One-Touch Join endpoints within its location (using the endpoints' APIs), either directly or via a web proxy.

Poly endpoints must be able to connect directly to the Conferencing Nodes in their location.

Management Node

As with all Pexip Infinity deployments, the Management Node must be able to contact each Conferencing Node.

In addition, if your One-Touch Join deployment is using OAuth (within either an Exchange O365 integration, or a G Suite integration with domain user authorization), the Management Node will send requests to the OAuth token endpoint. These requests will be sent either directly or via the web proxy (if one has been configured for the Management Node).

Port usage

The following table lists the ports/protocols required for communication between the components of Pexip One-Touch Join:

Source address	Source port	Destination address	Dest. port	Protocol
Management Node	55000–65535	Web proxy (if configured for the Management Node)	8080 †	TCP
Management Node	55000–65535	OAuth token endpoint (for O365 or G Suite integrations) ◊ • for O365: **login.microsoftonline.com** • for G Suite domain user authorization: **oauth2.googleapis.com/token**	<any> ‡	TCP (HTTPS)
Conferencing Node	55000–65535	Web proxy (if configured for the system location to which the Conferencing Node belongs)	8080 †	TCP

Source address	Source port	Destination address	Dest. port	Protocol
Conferencing Node	55000–65535	G Suite (for G Suite Integrations) ◊	443 †	TCP (HTTPS)
Conferencing Node	55000–65535	Exchange Server (for Exchange on-premises or O365 integrations) ◊	443 †	TCP (HTTPS)
Conferencing Node	55000–65535	Exchange Server (only required if the O365 Autodiscover URL lookup has otherwise failed) ◊	80†	TCP (HTTP)
Conferencing Node	55000–65535	OAuth token endpoint (for O365 or G Suite integrations) ◊ • for O365: **login.microsoftonline.com** • for G Suite service account authorization: **googleapis.com/oauth2/v4/token** • for G Suite domain user authorization: **oauth2.googleapis.com/token**	<any> ‡	TCP (HTTPS)
Conferencing Node	55000–65535	Cisco endpoint API ◊	80/443 †	TCP (HTTP/HTTPS)
Poly endpoint	<any>	Conferencing Node	443	TCP (HTTPS)

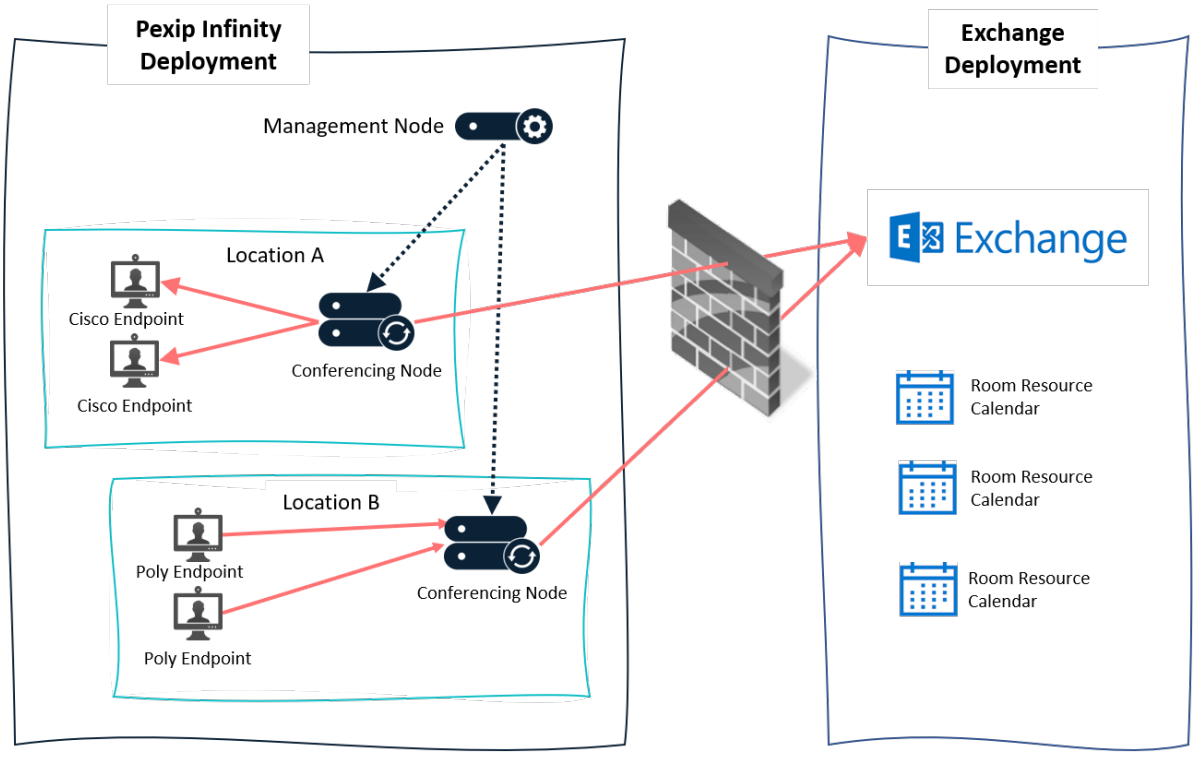
† Configurable by the administrator.

‡ Determined by Exchange / G Suite.

◊ Does not apply if a web proxy has been configured.

Note also that the ephemeral port range (55000–65535) is subject to change.

The diagram below summarizes the connectivity required between the components of Pexip One-Touch Join, using Microsoft Exchange as an example.



Note in most cases, and particularly for a dedicated One-Touch Join deployment, all Conferencing Nodes should remain within the internal network, and not in the DMZ.

Permitting the service account to access calendars

Exchange integrations

For Exchange integrations, the One-Touch Join service account must be able to impersonate the calendar of each OTJ room resource (or a user's personal calendar, if you wish to Use OTJ with personal endpoints and calendars). This is achieved by adding the email address to a specific OTJ Distribution Group, and giving the service account application impersonation rights to that group. For instructions on how to do this, see Configuring Application Impersonation on the service account (for Exchange on-premises) or Configuring Application Impersonation on the service account (for Office 365).

The use of Exchange impersonation is common in business applications that work with mail, when a single account needs to access many accounts.

The following information from Microsoft provides further background on the use of impersonation in Exchange:

- https://docs.microsoft.com/en-us/exchange/client-developer/exchange-web-services/impersonation-and-ews-in-exchange for guidelines on when to use impersonation in your Exchange service applications.
- https://blogs.msdn.microsoft.com/exchangedev/2009/06/15/exchange-impersonation-vs-delegate-access/ for information on the differences between impersonation and delegate access.

G Suite integrations

For G Suite integrations, the One-Touch Join service account (or the authentication user, if using 3-legged OAuth) must be able to access the calendar of each room resource. This is achieved by sharing the room resource's calendar (or the user's personal calendar, if you wish to Use OTJ with personal endpoints and calendars) with the service account. For instructions on how to do this, see Sharing individual calendars with the service account.

Note that the Google calendar API limits the number of calendars that can be shared within a 24 hour period to 750 (for more information, see https://support.google.com/a/answer/2905486?hl=en). This means that if you have more than 750 room resources that you wish to use for One-Touch Join, they will need to be set up over a period of days.

Using One-Touch Join with personal endpoints and calendars

Some users in your enterprise may have their own personal endpoints on their desk or in their office, which they want to integrate with their personal calendars so that they can simply use the "Join" button to connect to any video meetings that appear in their calendar.

To achieve this, you use the user's own email address as the room resource email address when configuring One-Touch Join. However, you must also ensure that the service account being used for One-Touch Join can access the user's calendar, as described in Permitting the service account to access calendars.

Configuring Exchange on-premises for One-Touch Join

This topic describes how to configure Microsoft Exchange in order to implement Pexip Infinity's One-Touch Join feature in a Microsoft Exchange on-premises environment.

The process involves the following steps, described in detail in the sections that follow:

1. Creating a service account for One-Touch Join. This service account will be used by One-Touch Join to read each room resource's calendar.
   - This should be a different service account to that used for VMR Scheduling for Exchange, because the configuration will be different.
2. Configuring Application Impersonation on the service account.
   - For more information and guidelines on the use of application impersonation in Exchange, see Permitting the service account to access calendars.
3. Enabling basic authentication for the service account.
4. Configuring calendar processing within Exchange.
5. Creating an associated Exchange integration on Pexip Infinity.

Prerequisites

Before you begin, ensure that the following configuration is complete:

1. Ensure **each physical room** that will have a One-Touch Join endpoint in it has an associated **room resource with an email address**.
2. **Enable auto calendar processing for each room resource**, so that the room will automatically accept meeting requests if it is available, and automatically decline an invitation if it is already booked.
3. Ensure you have access to your Exchange Admin Center (EAC) web interface, and access to Exchange Management PowerShell.
4. If your Exchange server does not use a globally trusted certificate, you must upload a custom CA certificate.

Creating a service account

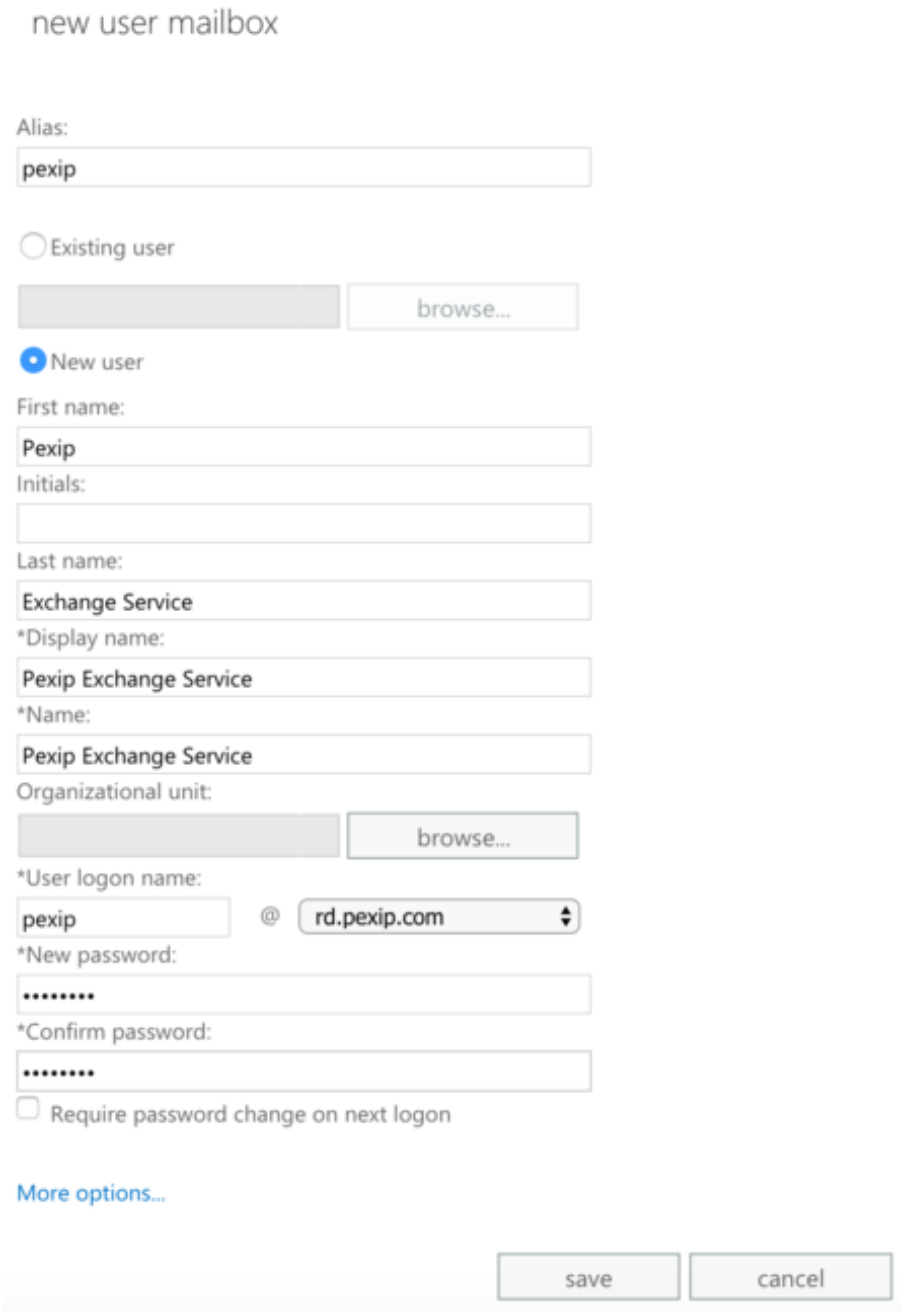
In this step, you create a service account that will be used to log in to Exchange to access the calendars of the room resources being used for One-Touch Join.

This service account should not be used for any other purpose other than One-Touch Join. You can however use the same service account for multiple One-Touch Join Exchange integrations.

You can create a new service account using **either** EAC or PowerShell, as follows:

EAC

1. Log in to your **Exchange Admin Center** as an administrator and go to **recipients > mailboxes**.
2. Add a new mailbox for the service account by selecting the + icon and then **User mailbox**.
3. Complete the fields as appropriate.
4. Uncheck the **Require password change on next logon** box.



5. Select Save.

PowerShell

The first command lets the administrator type in a password for the service account as a secure string. This password variable is then used in the second command to create a mailbox for the service account. The third command ensures the password of the service account will not expire.

```
$password = Read-Host "Enter password" -AsSecureString
```

```
New-Mailbox -Name "<Account Name>" -UserPrincipalName "<UPN>" -Password $password -Alias "<Account Alias>" -FirstName "<Account First Name>" -LastName "<Account Last Name>" -DisplayName "<Account Name>"
```

```
Set-ADUser -Identity "<UPN>" -PasswordNeverExpires $true
```

For example:

```
New-Mailbox -Name "Pexip OTJ Service Account" -UserPrincipalName pexip-otj-svc@example.com -Password $password -Alias pexip-otj-svc -FirstName "Pexip OTJ" -LastName "Service Account" -DisplayName "Pexip OTJ Service Account"
```

```
Set-ADUser -Identity pexip-otj-svc@example.com -PasswordNeverExpires $true
```

Configuring Application Impersonation on the service account

In this step, you create a new Distribution Group, and add the rooms to be used for One-Touch Join to the group. You then use PowerShell commands to make it so that the service account will only be able to impersonate members of that Group.

Configuring Application Impersonation in this way means that if rooms are added or removed from the group, this automatically updates whether or not the service account can impersonate them.

Creating a new Distribution Group

1. Log in to your **Exchange Admin Center** as an administrator and go to **recipients > groups**.
2. Select the **+** icon and select **add a new Distribution Group**.
3. Add the rooms you want to impersonate to the group.

   Note that the service account should **not** be added as a member of this distribution group. Instead, this step allows the service account to impersonate any member of this distribution group (i.e. any of the room resources).
4. Make sure to uncheck the option to make the group owner a group member. Otherwise the service account will be able to impersonate your account.
5. Also make sure to lock the group down so people cannot accidentally add themselves as group members. Do this by selecting ***Closed: Members can be added / removed only by the group owners***.

Configuring application impersonation

We recommend that you use combined PowerShell commands to configure application impersonation for the service account. This allows you to use variables, thus reducing possible copy and paste errors.

1. Configure the following variables with the values you actually want to use:
   - `otj_group_id`: the email of the distribution list whose members you want to be impersonated.
   - `otj_service_account`: the email of the service account you want to grant impersonation to.
   - `management_scope_to_create`: the name you want the newly created management scope to have.
   - `impersonation_role_name_to_create`: the name you want the newly created impersonation role to have.

   For example:

   ```
   $otj_group_id = "otjrooms@example.com"
   $otj_service_account = "pexip-otj-svc@example.com"
   $management_scope_to_create = "OTJ Management Scope"
   $impersonation_role_name_to_create = "OTJ Impersonation"
   ```
2. Create the management scope:

   ```
   $otj_group = Get-DistributionGroup -Identity $otj_group_id
   $otj_group_dn = $otj_group.DistinguishedName
   $restriction_filter = "MemberOfGroup -eq ""$otj_group_dn"""
   New-ManagementScope -Name $management_scope_to_create -RecipientRestrictionFilter $restriction_filter
   ```

   Example output:

   ```
   Name                  ScopeRestrictionType Exclusive RecipientRoot RecipientFilter
   ----                  -------------------- --------- ------------- ---------------
   OTJ Management Scope  RecipientScope       False                   MemberOfGroup -eq 'CN=OTJ Rooms20190430164340,OU...
   ```
3. Set up application impersonation using the previously created management scope:

   ```
   New-ManagementRoleAssignment -Name $impersonation_role_name_to_create -Role ApplicationImpersonation -User $otj_service_
   account -CustomRecipientWriteScope $management_scope_to_create
   ```

   Example output:

   ```
   Name               Role             RoleAssigneeName  RoleAssigneeType  AssignmentMethod  EffectiveUserName
   ----               ----             ----------------  ----------------  ----------------  -----------------
   OTJ Impersonation  ApplicationImp... pexip-otj-svc     User              Direct
   ```
4. Verify that the above commands worked as expected. In the following command, replace **<resource_email>** with the email of the room resource mailbox you want to test. If it is a room which is a member of the distribution list, it should show the OTJ Impersonation in the returned roles. If it is anything else outside of the distribution list, it should not have the OTJ Impersonation listed, which means the OTJ service account does not have permission to impersonate that user.

```
Get-ManagementRoleAssignment -Role ApplicationImpersonation -WritableRecipient "<resource_email>" | Format-List Name,
Role, RoleAssignee, CustomRecipientWriteScope
```

Expected output:

```
Name                : OTJ Impersonation
Role                : ApplicationImpersonation
RoleAssignee        : pexip-otj-svc
```

Enabling basic authentication

In this step, you enable basic authentication for the service account that One-Touch Join uses to log in to Exchange.

If you are using on-prem Exchange you need to ensure basic authentication is enabled for Exchange Web Services (EWS). When basic authentication is enabled, Pexip Infinity stores the credentials in encrypted form and all authentication is carried out over a secure TLS channel.

You can enable basic authentication using **either** Windows Service Manager or PowerShell, as follows:

Windows Service Manager	PowerShell

1. Go to the Windows server on which Exchange is installed and open the Service Manager.
2. Select the server on which Exchange is installed, and right-click to select **Computer Management**.
3. From the panel on the left, select **Services and Applications > Internet Information Services (IIS) Manager**.
4. Expand the options and select **Sites > Default Web Site > EWS**.
5. Select the **Authentication** button in the main pane.
6. Find **Basic Authentication** in the list and ensure it is ***Enabled***. (If not, right-click and select ***Enable***.)
7. Select **Save**.
	This command enables basic authentication on a specific server:
	`Set-WebServicesVirtualDirectory -Identity "<server>\EWS (Default Web Site)" -BasicAuthentication $true`
	For example, if your server name is PEXCHANGE then:
	`Set-WebServicesVirtualDirectory -Identity "PEXCHANGE\EWS (Default Web Site)" -BasicAuthentication $true`

Configuring calendar processing on room resource mailboxes

In this step, you change the calendar processing settings for room resources from the default to those required to support One-Touch Join.

Recommended configuration

In order to take full advantage of the functionality offered by Pexip Infinity One-Touch Join, we recommend that, for One-Touch Join room resources, you change three calendar processing options from the default.

- Firstly, when a meeting invite is received by a resource mailbox, by default the meeting subject is deleted and is replaced with the name of the organizer (for more information, see https://support.microsoft.com/en-gb/help/2842288/resource-mailbox-s-calendar-shows-the-organizer-s-name-instead-of-the).

  Because One-Touch Join accesses the meeting invites through the resource mailboxes, this default behavior means One-Touch Join won't have access to the original subject. You can choose to leave the default behavior for privacy reasons, or you can modify the calendar processing options for each mailbox so that the meeting subject is available to One-Touch Join and can be displayed on the meeting room endpoints.
- Secondly, by default the meeting invite body is deleted. If you wish One-Touch Join to parse meeting details from the body then you must set the **DeleteComments** property to ***False***. If you leave this set to ***True***, only those rules that process information in the calendar headers can be used (because the body will be deleted).
- Thirdly, by default the private flag is cleared. If you wish meetings that are marked as private by the organizer to remain marked as private in the room mailbox, you must set the **RemovePrivateProperty** flag to ***False***.

PowerShell command

To modify the calendar processing on a room from the default settings to those we recommend for One-Touch Join, use the following PowerShell command (replacing **resource_email** with the address of the room resource whose processing you wish to change):

```
Set-CalendarProcessing -Identity <resource_email> `
-DeleteSubject $False `
-AddOrganizerToSubject $False `
-DeleteComments $False `
-RemovePrivateProperty $False
```

Optional configuration

Hiding invitation details from other users

In order for One-Touch Join to function fully, the service account must be able to access the body of the invitation (which is why we recommend that you set the **DeleteComments** property to ***False***). However, this does mean that all other users in your deployment with access to the room resource calendar may also be able to view the body of the invitation (depending on your deployment's other policies). If you wish to prevent this, you can use the following PowerShell command to restrict what users can see by default, without restricting what the service account can access.

In the following command, replace **resource_name** with the name of the room resource, and replace **role** with one of the following roles:

- **AvailabilityOnly**: users will be able to view the room's availability, but nothing else
- **LimitedDetails**: users will be able to view the room's availability and the meeting subject and location, but not the body of the invitation.

```
Set-MailboxFolderPermission "resource_name:\Calendar" -User Default -AccessRights role
```

Allowing forwarding of external invitations

If you want to enable users to forward invitations from other organizations to your OTJ room resources, you must set the **ProcessExternalMeetingMessages** flag to ***True***. This also allows users external to your organization to invite the resource directly; you should therefore consult your Exchange administrator to determine whether this is appropriate in your environment.

If your Microsoft Exchange environment uses a security application (such as Office 365 ATP, or Mimecast) to re-write URLs, this may prevent OTJ from being used to join external Microsoft Teams meetings (for example, when a user inside your organization forwards an external Microsoft Teams meeting invitation to an OTJ room resource in order to join the meeting from that endpoint). To enable users to join these meetings using OTJ, you must ensure that the security application's URL re-write rules include an exception for any URL starting with the domain **https:\\teams.microsoft.com**

Checking calendar processing settings

The following PowerShell command can be used to check calendar processing settings on all of the rooms in the Distribution Group that was created for One-Touch Join.

We recommend copying and saving this as a file and running it from within PowerShell.

Before running, ensure that you edit `$otj_group_id = "otjrooms@example.com"` to use the email of the Distribution Group used in your own deployment.

```
$deleted_subjects = @()
$organizer_added = @()
$deleted_bodies = @()
$private_flag_reset = @()
$not_auto_accept = @()
$process_external = @()
```

```
$otj_group_id = "otjrooms@example.com"

Get-DistributionGroupMember -Identity $otj_group_id -ResultSize Unlimited | ForEach-Object {
Write-Host "Checking room '$($_.name)'"
$processing = Get-CalendarProcessing -Identity $_.name
$pass = $true
if ($processing.DeleteSubject) {
Write-Host "WARNING: The room '$($_.name)' is deleting the meeting subject" -ForegroundColor Red
$deleted_subjects += $_.name
$pass = $false
    }
if ($processing.AddOrganizerToSubject) {
Write-Host "WARNING: The room '$($_.name)' is adding the organizer to the meeting subject" -ForegroundColor Red
$organizer_added += $_.name
$pass = $false
    }
if ($processing.DeleteComments) {
Write-Host "WARNING: The room '$($_.name)' is deleting the meeting body" -ForegroundColor Red
$deleted_bodies += $_.name
$pass = $false
    }
if ($processing.RemovePrivateProperty) {
Write-Host "WARNING: The room '$($_.name)' is clearing the private flag on meetings" -ForegroundColor Red
$private_flag_reset += $_.name
$pass = $false
    }
if ($processing.AutomateProcessing -ne "AutoAccept") {
Write-Host "WARNING: The room '$($_.name)' is not configured to Auto Accept. Processing='$($processing.AutomateProcessing)'" -ForegroundColor Red
$not_auto_accept += $_.name
$pass = $false
    }
# Optional permission for allowing the external invites:
if ($processing.ProcessExternalMeetingMessages) {
Write-Host "The room '$($_.name)' is configured to process external (forwarded) meetings"
$process_external += $_.name
    }
if ($pass) {
Write-Host "INFO: All checks passed for room '$($_.name)'" -ForegroundColor Green
    }
}

Write-Host "Summary:"
Write-Host "There are $($deleted_subjects.count) rooms deleting the meeting subject"
if ($deleted_subjects) {
Write-Host $deleted_subjects -Separator ", "
Write-Host ""
}
Write-Host "There are $($organizer_added.count) rooms adding the organizer to the meeting subject"
if ($organizer_added) {
Write-Host $organizer_added -Separator ", "
Write-Host ""
}
Write-Host "There are $($deleted_bodies.count) rooms deleting the meeting body"
if ($deleted_bodies) {
Write-Host $deleted_bodies -Separator ", "
Write-Host ""
}
Write-Host "There are $($private_flag_reset.count) rooms clearing the private flag on meetings"
if ($private_flag_reset) {
Write-Host $private_flag_reset -Separator ", "
Write-Host ""
}
Write-Host "There are $($not_auto_accept.count) rooms not configured to Auto Accept"
if ($not_auto_accept) {
Write-Host $not_auto_accept -Separator ", "
Write-Host ""
}
Write-Host "There are $($process_external.count) rooms configured to process external (forwarded) meetings"
if ($process_external) {
Write-Host $process_external -Separator ", "
Write-Host ""
}
```

Adding a One-Touch Join Exchange integration on Pexip Infinity

In this step you log in to the Pexip Infinity Administrator interface and add details of the Exchange deployment you are integrating with, including details of the service account username and password (based on the configuration you have just set up in Exchange).

From the Pexip Infinity Administrator interface, go to **One-Touch Join > OTJ Exchange Integrations**.

Option	Description
Name	The name of this One-Touch Join Exchange integration.
Description	An optional description of this One-Touch Join Exchange integration.
Service account username	The username of the service account to be used by the One-Touch Join Exchange integration. This must be in the format **name@example.com**.
Enable OAuth	Leave this option disabled to continue using Basic Auth. (OAuth 2.0 is supported for Exchange in Office 365 only.)
Service account password	(Available if OAuth has not been enabled) The password of the service account to be used by the One-Touch Join Exchange integration.
Advanced options	
Find Items Request Quota	The number of Find Item requests that can be made by OTJ to your Exchange Server in a 24-hour period. The default of 1,000,000 should be sufficient for most deployments — for more information, see Frequency and limitations on calendar requests. We do not recommend increasing this quota unless you have deployed a dedicated One-Touch Join platform, because it will impact the performance of the Conferencing Nodes.
OTJ Exchange Autodiscover URLs	
ℹ This section is optional and will generally only be required if the Autodiscover URLs in your deployment do not use a standard location.	
Name	The name of this Exchange Autodiscover URL.
Description	An optional description of this Exchange Autodiscover URL.
Autodiscover URL	The URL used to connect to the Autodiscover service on the Exchange deployment. If you are using Office 365, you may need to enter your autodiscover URL manually, particularly if you are using a hybrid Exchange deployment. If your OTJ room resources and service account are hosted on O365, then you should enter **https://autodiscover-s.outlook.com/autodiscover/autodiscover.svc** as the **Autodiscover URL**. ℹ The URL must end in **.svc**; URLs ending in **.xml** are not supported.

Next steps

You must now configure the remainder of the One-Touch Join components on Pexip Infinity, as described in Configuring Pexip Infinity for One-Touch Join.

Configuring Office 365 for One-Touch Join

This topic describes how to configure Microsoft Exchange in order to implement Pexip Infinity's One-Touch Join feature in a Microsoft Office 365 environment.

The process involves the following steps, described in detail in the sections that follow:

1. Creating a service account for One-Touch Join. This service account will be used by One-Touch Join to read each room resource's calendar.
   - This should be a different service account to any used for VMR Scheduling for Exchange, because the configuration will be different.
2. Configuring Application Impersonation on the service account.
   - For more information and guidelines on the use of application impersonation in Exchange, see Permitting the service account to access calendars.
3. Configuring calendar processing within Exchange.
4. Enabling OAuth authentication for the service account.
5. Creating an associated Exchange integration on Pexip Infinity.

Prerequisites

Before you begin, ensure that the following configuration is complete:

1. Ensure **each physical room** that will have a One-Touch Join endpoint in it has an associated **room resource with an email address**.
2. **Enable auto calendar processing for each room resource**, so that the room will automatically accept meeting requests if it is available, and automatically decline an invitation if it is already booked.
3. Ensure that you have a license available for the service account; this is required for the service account to access Exchange.
4. Ensure you have access to your Office 365 web interface, and access to the Microsoft Azure Active Directory Module for Windows PowerShell. (If you are connecting from your Windows PC for the first time, you may need to run the `Install-Module MSOnline` PowerShell command; for more information, see https://docs.microsoft.com/en-us/office365/enterprise/powershell/connect-to-office-365-powershell.)

Creating a service account

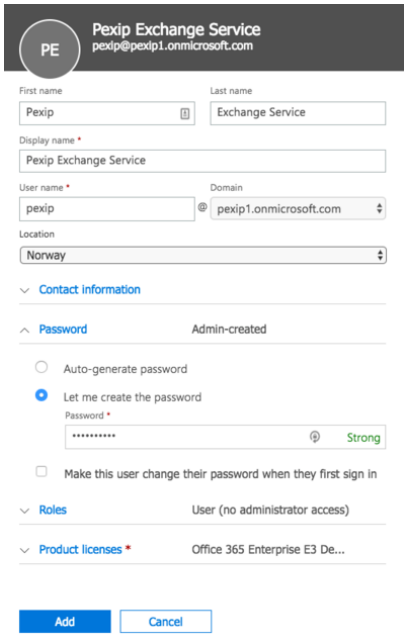
In this step, you create a service account that will be used to log in to Exchange to access the calendars of the room resources being used for One-Touch Join.

After creating the service account, you must assign it an appropriate Exchange license, such as Office 365 Enterprise E1, Office 365 Business Basic (formerly Essentials) or one of the Exchange Online plans.

This service account should not be used for any other purpose other than One-Touch Join. You can however use the same service account for multiple One-Touch Join Exchange integrations.

You can create a new service account using **either** the Office 365 admin portal or PowerShell, as follows:

O365	PowerShell

1. Go to portal.office.com and log in as the administrator.
2. Go to the admin portal by selecting the **Admin** tile (this will take you to https://portal.office.com/adminportal/home#/homepage).
3. From the **Users** section, select **Add a user** and complete the necessary fields:
a. In the **Password** section:
■ select **Let me create the password**
■ uncheck **Make this user change their password when they first sign in**.
b. In the **Product licenses** section, assign an appropriate product license from the available list.
 
 	
4. Select **Add** to create the user.
	Establishing a remote connection
	To use PowerShell for Office 365 you first need to connect remotely. Full instructions are given at https://technet.microsoft.com/en-gb/library/dn568015.aspx but the commands are:
	`Set-ExecutionPolicy RemoteSigned`
	`$UserCredential = Get-Credential`
	`$Session = New-PSSession -ConfigurationName Microsoft.Exchange -ConnectionUri https://outlook.office365.com/powershell-liveid/ -Credential $UserCredential -Authentication Basic -AllowRedirection`
	`Import-PSSession $Session`
	`Import-Module MsOnline`
	`Connect-MsolService -Credential $UserCredential`
	Creating the service account
	The first command lets the administrator type in a password for the service account as a secure string. This password variable is then used in the second command to create a mailbox for the service account. The third command ensures the password of the service account will not expire. The final command terminates the remote session.
	`$password = Read-Host "Enter password" -AsSecureString`
	`New-Mailbox -Name "<Account Name>" -MicrosoftOnlineServicesID "<UPN>" -Password $password -Alias "<Account Alias>" -FirstName "<Account First Name>" -LastName "<Account Last Name>" -DisplayName "<Account Name>"`
	`Set-MsolUser -UserPrincipalName "<UPN>" -PasswordNeverExpires $true`
	`Remove-PSSession $Session`
	For example:
	`New-Mailbox -Name "Pexip OTJ Service Account" -MicrosoftOnlineServicesID pexip-otj-svc@example.com -Password $password -Alias pexip-otj-svc -FirstName "Pexip OTJ" -LastName "Service Account" -DisplayName "Pexip OTJ Service Account"`
	`Set-MsolUser -UserPrincipalName pexip-otj-svc@example.com -PasswordNeverExpires $true`
	Assigning a license to the service account
	You must now assign an appropriate license to the service account. You can use the command:
	`Set-MsolUserLicense -UserPrincipalName "<UPN>" -AddLicenses "<license>"`

Configuring Application Impersonation on the service account

In this step, you create a new Distribution Group, and add the rooms to be used for One-Touch Join to the group. You then use PowerShell commands to make it so that the service account will only be able to impersonate members of that Group.

Configuring Application Impersonation in this way means that when a room is added to the group, the service account will automatically be able to impersonate it. Likewise, when a room is removed, the service account will no longer be able to impersonate it.

Creating a new Distribution Group

1. Go to admin.microsoft.com and log in as the administrator.
2. From the menu on the left hand side, select **Groups** > **Add a group**.

3. For the **Group Type**, select ***Distribution List***. Enter a name, email address and description and select **Add**.
4. Add as members of the Group the rooms to be used for One-Touch Join. These will be the rooms that the service account will impersonate.

   Note that the service account should **not** be added as a member of this distribution group. Instead, this step allows the service account to impersonate any member of this distribution group (i.e. any of the room resources).
5. Open up a remote PowerShell connection to Office 365 and import an Exchange session. For example see https://docs.microsoft.com/en-us/office365/enterprise/powershell/connect-to-all-office-365-services-in-a-single-windows-powershell-window

Configuring application impersonation

We recommend that you use combined PowerShell commands to configure application impersonation for the service account. This allows you to use variables, thus reducing possible copy and paste errors.

1. You may need to enable customization, if this has not already been done within your organization:

   ```
   Enable-OrganizationCustomization
   ```
2. Configure the following variables with the values you actually want to use:
   - `otj_group_id`: the email of the distribution list whose members you want to be impersonated.
   - `otj_service_account`: the email of the service account you want to grant impersonation to.
   - `management_scope_to_create`: the name you want the newly created management scope to have.
   - `impersonation_role_name_to_create`: the name you want the newly created impersonation role to have.

   For example:

   ```
   $otj_group_id = "otjrooms@example.com"
   $otj_service_account = "pexip-otj-svc@example.com"
   $management_scope_to_create = "OTJ Management Scope"
   $impersonation_role_name_to_create = "OTJ Impersonation"
   ```
3. Create the management scope:

   ```
   $otj_group = Get-DistributionGroup -Identity $otj_group_id
   $otj_group_dn = $otj_group.DistinguishedName
   $restriction_filter = "MemberOfGroup -eq ""$otj_group_dn"""
   New-ManagementScope -Name $management_scope_to_create -RecipientRestrictionFilter $restriction_filter
   ```

   Example output:

   ```
   Name                 ScopeRestrictionType Exclusive RecipientRoot RecipientFilter
   ----                 -------------------- --------- ------------- ---------------
   OTJ Management Scope RecipientScope       False                   MemberOfGroup -eq 'CN=OTJ Rooms20190430164340,OU...
   ```
4. Set up application impersonation using the previously created management scope:

   ```
   New-ManagementRoleAssignment -Name $impersonation_role_name_to_create -Role ApplicationImpersonation -User $otj_service_
   account -CustomRecipientWriteScope $management_scope_to_create
   ```

   Example output:

   ```
   Name               Role             RoleAssigneeName  RoleAssigneeType  AssignmentMethod  EffectiveUserName
   ----               ----             ----------------  ----------------  ----------------  -----------------
   OTJ Impersonation  ApplicationImp... pexip-otj-svc     User              Direct
   ```
5. Verify that the above commands worked as expected. In the following command, replace **<resource_email>** with the email of the room resource mailbox you want to test. If it is a room which is a member of the distribution list, it should show the OTJ Impersonation in the returned roles. If it is anything else outside of the distribution list, it should not have the OTJ Impersonation listed, which means the OTJ service account does not have permission to impersonate that user.

   ```
   Get-ManagementRoleAssignment -Role ApplicationImpersonation -WritableRecipient "<resource_email>" | Format-List Name,
   Role, RoleAssignee, CustomRecipientWriteScope
   ```

   Expected output:

   ```
   Name                      : OTJ Impersonation
   Role                      : ApplicationImpersonation
   RoleAssignee              : pexip-otj-svc
   ```

Configuring calendar processing on room resource mailboxes

In this step, you change the calendar processing settings for room resources from the default to those required to support One-Touch Join.

Recommended configuration

In order to take full advantage of the functionality offered by Pexip Infinity One-Touch Join, we recommend that, for One-Touch Join room resources, you change three calendar processing options from the default.

- Firstly, when a meeting invite is received by a resource mailbox, by default the meeting subject is deleted and is replaced with the name of the organizer (for more information, see https://support.microsoft.com/en-gb/help/2842288/resource-mailbox-s-calendar-shows-the-organizer-s-name-instead-of-the).

  Because One-Touch Join accesses the meeting invites through the resource mailboxes, this default behavior means One-Touch Join won't have access to the original subject. You can choose to leave the default behavior for privacy reasons, or you can modify the calendar processing options for each mailbox so that the meeting subject is available to One-Touch Join and can be displayed on the meeting room endpoints.
- Secondly, by default the meeting invite body is deleted. If you wish One-Touch Join to parse meeting details from the body then you must set the **DeleteComments** property to ***False***. If you leave this set to ***True***, only those rules that process information in the calendar headers can be used (because the body will be deleted).
- Thirdly, by default the private flag is cleared. If you wish meetings that are marked as private by the organizer to remain marked as private in the room mailbox, you must set the **RemovePrivateProperty** flag to ***False***.

PowerShell command

To modify the calendar processing on a room from the default settings to those we recommend for One-Touch Join, use the following PowerShell command (replacing **resource_email** with the address of the room resource whose processing you wish to change):

```
Set-CalendarProcessing -Identity <resource_email> `
-DeleteSubject $False `
-AddOrganizerToSubject $False `
-DeleteComments $False `
-RemovePrivateProperty $False
```

Optional configuration

Hiding invitation details from other users

In order for One-Touch Join to function fully, the service account must be able to access the body of the invitation (which is why we recommend that you set the **DeleteComments** property to ***False***). However, this does mean that all other users in your deployment with access to the room resource calendar may also be able to view the body of the invitation (depending on your deployment's other policies). If you wish to prevent this, you can use the following PowerShell command to restrict what users can see by default, without restricting what the service account can access.

In the following command, replace **resource_name** with the name of the room resource, and replace **role** with one of the following roles:

- **AvailabilityOnly**: users will be able to view the room's availability, but nothing else
- **LimitedDetails**: users will be able to view the room's availability and the meeting subject and location, but not the body of the invitation.

```
Set-MailboxFolderPermission "resource_name:\Calendar" -User Default -AccessRights role
```

Allowing forwarding of external invitations

If you want to enable users to forward invitations from other organizations to your OTJ room resources, you must set the **ProcessExternalMeetingMessages** flag to ***True***. This also allows users external to your organization to invite the resource directly; you should therefore consult your Exchange administrator to determine whether this is appropriate in your environment.

If your Microsoft Exchange environment uses a security application (such as Office 365 ATP, or Mimecast) to re-write URLs, this may prevent OTJ from being used to join external Microsoft Teams meetings (for example, when a user inside your organization forwards an external Microsoft Teams meeting invitation to an OTJ room resource in order to join the meeting from that endpoint). To enable users to join these meetings using OTJ, you must ensure that the security application's URL re-write rules include an exception for any URL starting with the domain **https:\\teams.microsoft.com**

Checking calendar processing settings

The following PowerShell command can be used to check calendar processing settings on all of the rooms in the Distribution Group that was created for One-Touch Join.

We recommend copying and saving this as a file and running it from within PowerShell.

Before running, ensure that you edit `$otj_group_id = "otjrooms@example.com"` to use the email of the Distribution Group used in your own deployment.

```
$deleted_subjects = @()
$organizer_added = @()
$deleted_bodies = @()
$private_flag_reset = @()
$not_auto_accept = @()
$process_external = @()
$otj_group_id = "otjrooms@example.com"

Get-DistributionGroupMember -Identity $otj_group_id -ResultSize Unlimited | ForEach-Object {
Write-Host "Checking room '$($_.name)'"
$processing = Get-CalendarProcessing -Identity $_.name
$pass = $true
if ($processing.DeleteSubject) {
Write-Host "WARNING: The room '$($_.name)' is deleting the meeting subject" -ForegroundColor Red
$deleted_subjects += $_.name
$pass = $false
    }
if ($processing.AddOrganizerToSubject) {
Write-Host "WARNING: The room '$($_.name)' is adding the organizer to the meeting subject" -ForegroundColor Red
$organizer_added += $_.name
$pass = $false
    }
if ($processing.DeleteComments) {
Write-Host "WARNING: The room '$($_.name)' is deleting the meeting body" -ForegroundColor Red
$deleted_bodies += $_.name
$pass = $false
    }
if ($processing.RemovePrivateProperty) {
Write-Host "WARNING: The room '$($_.name)' is clearing the private flag on meetings" -ForegroundColor Red
$private_flag_reset += $_.name
$pass = $false
    }
if ($processing.AutomateProcessing -ne "AutoAccept") {
Write-Host "WARNING: The room '$($_.name)' is not configured to Auto Accept. Processing='$($processing.AutomateProcessing)'" -ForegroundColor Red
$not_auto_accept += $_.name
$pass = $false
    }
# Optional permission for allowing the external invites:
if ($processing.ProcessExternalMeetingMessages) {
Write-Host "The room '$($_.name)' is configured to process external (forwarded) meetings"
$process_external += $_.name
    }
if ($pass) {
Write-Host "INFO: All checks passed for room '$($_.name)'" -ForegroundColor Green
    }
}

Write-Host "Summary:"
Write-Host "There are $($deleted_subjects.count) rooms deleting the meeting subject"
if ($deleted_subjects) {
Write-Host $deleted_subjects -Separator ", "
Write-Host ""
```

```
}
Write-Host "There are $($organizer_added.count) rooms adding the organizer to the meeting subject"
if ($organizer_added) {
Write-Host $organizer_added -Separator ", "
Write-Host ""
}
Write-Host "There are $($deleted_bodies.count) rooms deleting the meeting body"
if ($deleted_bodies) {
Write-Host $deleted_bodies -Separator ", "
Write-Host ""
}
Write-Host "There are $($private_flag_reset.count) rooms clearing the private flag on meetings"
if ($private_flag_reset) {
Write-Host $private_flag_reset -Separator ", "
Write-Host ""
}
Write-Host "There are $($not_auto_accept.count) rooms not configured to Auto Accept"
if ($not_auto_accept) {
Write-Host $not_auto_accept -Separator ", "
Write-Host ""
}
Write-Host "There are $($process_external.count) rooms configured to process external (forwarded) meetings"
if ($process_external) {
Write-Host $process_external -Separator ", "
Write-Host ""
}
```

Enabling OAuth authentication

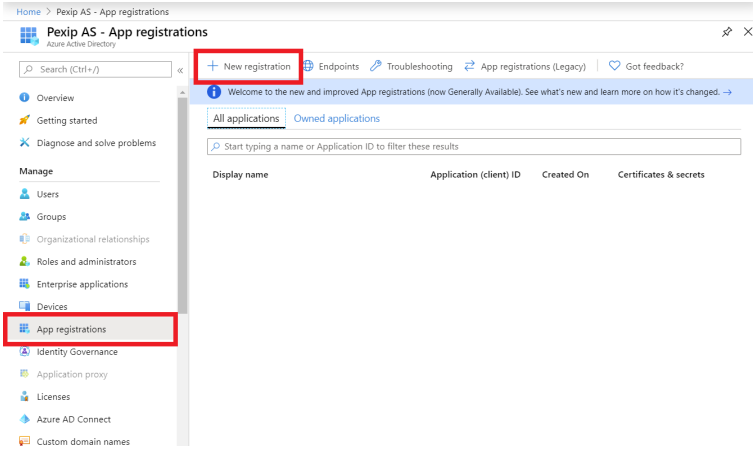
In this step, you enable OAuth authentication for the service account that One-Touch Join uses to log in to Exchange.

As of H2 2021, Microsoft will stop supporting and fully decommission Basic Authentication for EWS to access Exchange Online (for more information, see Microsoft's announcement). We therefore strongly recommend that for Office 365, all new deployments authenticate the service account using OAuth 2.0, and all existing deployments are updated to enable this option as soon as possible.

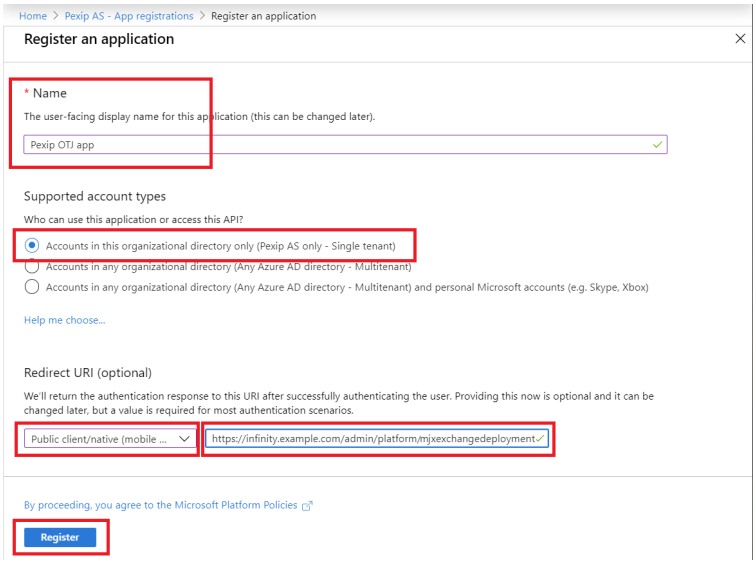
To use OAuth for the service account, you must create an app registration in Azure and then use the settings from this app registration when enabling and configuring the OAuth options within the One-Touch Join Exchange integration.

Create a new App Registration in Azure

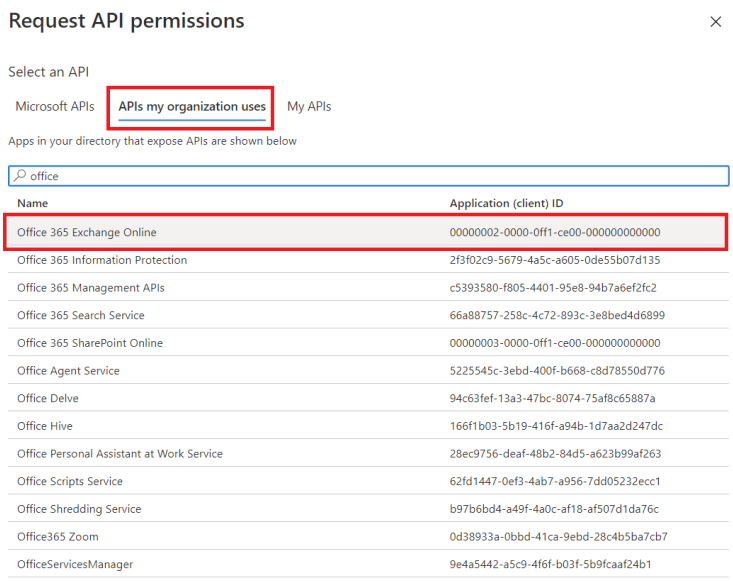
1. Log into the Azure portal at portal.azure.com.
2. From the main panel on the left, select **Azure Active Directory**.
3. Select **App Registrations** and then **New registration:**



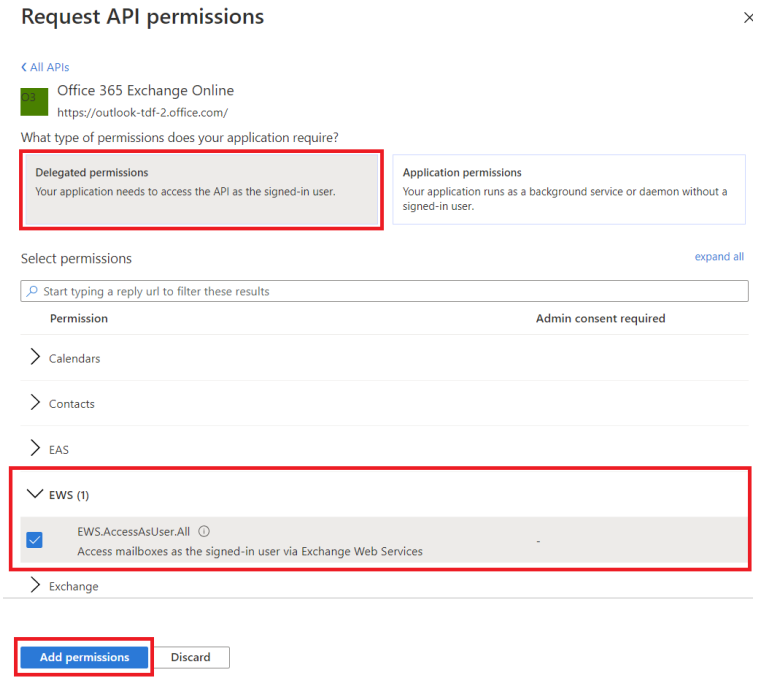
4. In the **Register an application** panel, enter the following options:
   a. **Name**: this can be anything you wish. In our example we have used ***Pexip OTJ App***.
   b. **Supported account types**: select ***Accounts in this organizational directory only***.
   c. **Redirect URI**: from the drop-down menu, select ***Public client/native (mobile and desktop)***. The URI should be the IP address or FQDN of the Management Node, in the format **https://<Management Node Address>/admin/platform/mjxexchangedeployment/oauth_redirect/**
   In our example we have used **https://infinity.example.com/admin/platform/mjxexchangedeployment/oauth_redirect/**
   You will need to enter this as the **OAuth redirect URI** when configuring a One-Touch Join Exchange integration.
   ℹ The **OAuth redirect URI** is the URI to which you will be returned after you have successfully signed in to the service account. It must be the same on Azure and Pexip Infinity in order for Azure to validate the sign-in request.



5. Select **Register**.
   A new panel will open where you can configure your application.
6. From the panel on the left, select **API permissions**.
7. Select **Add a permission**.
8. From the **Request API permissions** panel, select **APIs my organization uses**, search for **Office 365 Exchange Online** and select it:



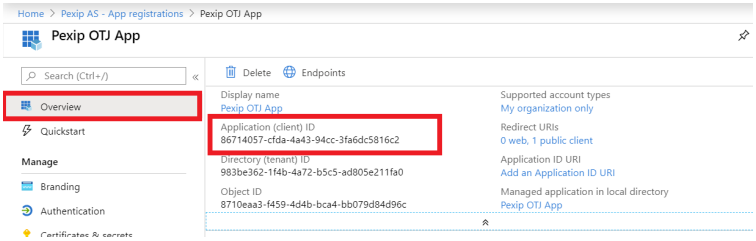
9. Select **Delegated permissions**, and from the **Select permissions** list, expand **EWS** and select **Access mailboxes as the signed-in user via Exchange Web Services**, and then select **Add permissions**:



Taking note of configuration

When you Configure the One-Touch Join Exchange integration and enable OAuth authentication for the service account, you'll need to provide the following information from Azure:

- **Application (client) ID**: this was generated for you by Azure when you saved the App Registration:



  You can find this again in Azure under **Azure Active Directory > App Registrations**, under the **Application (client) ID** column.

  You will need to enter this as the **OAuth client ID** when configuring the One-Touch Join Exchange integration.

- **Redirect URI**: this is the URI you entered when creating the App Registration.

  You can find this again in Azure under **Azure Active Directory > App Registrations**, clicking on the app registration, and then clicking **Redirect URIs**.

  You will need to enter this as the **OAuth redirect URI** when configuring the One-Touch Join Exchange integration.

You will also need to know the OAuth Endpoints to use. To find this information:

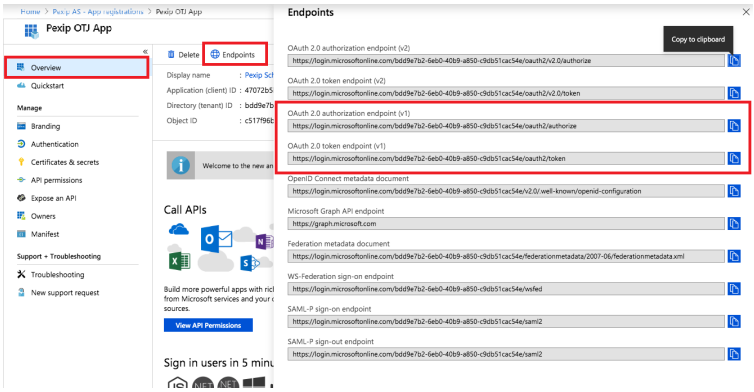
1. In the Azure Portal, select **Overview > Endpoints**.
2. Copy the URL of the **OAuth 2.0 authorization endpoint (v1)**.

   Ensure that you use the URL for **... endpoint (v1)**, not **... endpoint (v2)**.

   You will need to enter this as the **OAuth authorization endpoint** when configuring the One-Touch Join Exchange integration.
3. Copy the URL of the **OAuth 2.0 token endpoint (v1)**

   Ensure that you use the URL for **... endpoint (v1)**, not **... endpoint (v2)**.

   You will need to enter this as the **OAuth token endpoint** when configuring the One-Touch Join Exchange integration.



Adding a One-Touch Join Exchange integration on Pexip Infinity

In this step you log in to the Pexip Infinity Administrator interface and add details of the Exchange deployment you are integrating with, including details of the service account and OAuth access (based on the configuration you have just set up in Exchange). You must then sign in to Exchange using the service account.

Configuring the Exchange integration

From the Pexip Infinity Administrator interface, go to **One-Touch Join > OTJ Exchange Integrations**.

Option	Description
Name	The name of this One-Touch Join Exchange integration.
Description	An optional description of this One-Touch Join Exchange integration.
Service account username	The username of the service account to be used by the One-Touch Join Exchange integration. This must be in the format **name@example.com**.
Enable OAuth	Enable this option to authenticate the service account using OAuth 2.0. (This option is only supported for Exchange in Office 365.) As of H2 2021, Microsoft will stop supporting and fully decommission Basic Authentication for EWS to access Exchange Online (for more information, see Microsoft's announcement). We therefore strongly recommend that for Office 365, all new deployments authenticate the service account using OAuth 2.0, and all existing deployments are updated to enable this option as soon as possible.
OAuth client ID	(Available if OAuth has been enabled) The **Application (client) ID** which was generated by Azure when creating an App Registration in Azure Active Directory (see Taking note of configuration).
OAuth redirect URI	(Available if OAuth has been enabled) The redirect URI you entered when creating an App Registration in Azure Active Directory. This should be in the format **https://<Management Node Address>/admin/platform/mjxexchangedeployment/oauth_redirect/** The **OAuth redirect URI** is the URI to which you will be returned after you have successfully signed in to the service account. It must be the same on Azure and Pexip Infinity in order for Azure to validate the sign-in request.
OAuth authorization endpoint	(Available if OAuth has been enabled) The URL of the OAuth authorization endpoint (see Taking note of configuration). Ensure that you use the URL for **... endpoint (v1)**, not **... endpoint (v2)**.
OAuth token endpoint	(Available if OAuth has been enabled) The URL of the OAuth token endpoint (see Taking note of configuration). Ensure that you use the URL for **... endpoint (v1)**, not **... endpoint (v2)**.
Advanced options	
Find Items Request Quota	The number of Find Item requests that can be made by OTJ to your Exchange Server in a 24-hour period. The default of 1,000,000 should be sufficient for most deployments — for more information, see Frequency and limitations on calendar requests. We do not recommend increasing this quota unless you have deployed a dedicated One-Touch Join platform, because it will impact the performance of the Conferencing Nodes.
OTJ Exchange Autodiscover URLs	
This section is optional and will generally only be required if the Autodiscover URLs in your deployment do not use a standard location.	
Name	The name of this Exchange Autodiscover URL.
Description	An optional description of this Exchange Autodiscover URL.

Option	Description
Autodiscover URL	The URL used to connect to the Autodiscover service on the Exchange deployment. If you are using Office 365, you may need to enter your autodiscover URL manually, particularly if you are using a hybrid Exchange deployment. If your OTJ room resources and service account are hosted on O365, then you should enter **https://autodiscover-s.outlook.com/autodiscover/autodiscover.svc** as the **Autodiscover URL**. The URL must end in **.svc**; URLs ending in **.xml** are not supported.

When you have completed the above fields, select **Save**. You will be returned to the main OTJ Exchange Integration page. You must now sign in to the Exchange integration using the service account details you have just created.

Signing in to the service account

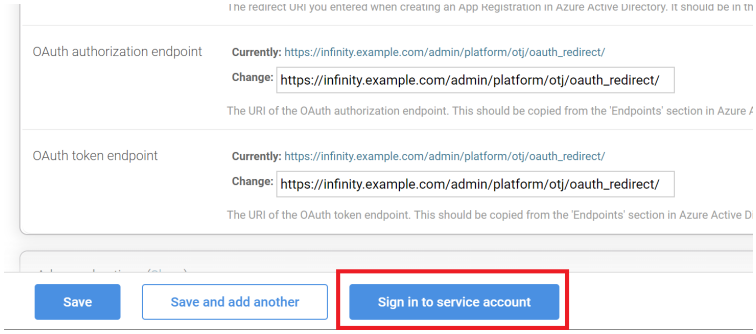
If you have enabled OAuth for the first time, you must sign in to the service account after saving the configuration of the One-Touch Join Exchange integration.

You may also need to re-sign in to the service account if:

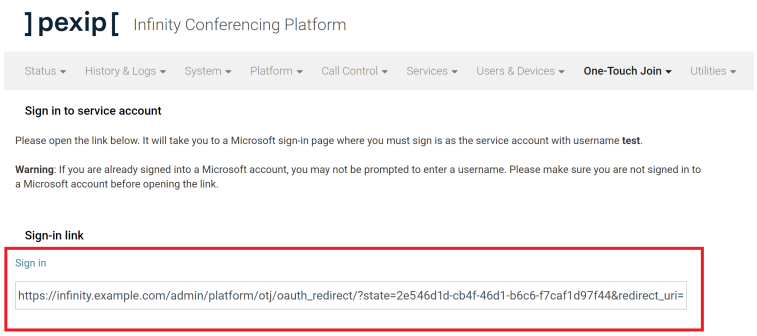
- you disable and then subsequently re-enable OAuth
- you update any of the following configuration for the One-Touch Join Exchange integration:
  - Service account username
  - OAuth client ID
  - OAuth token endpoint
- the Management Node has been offline for more than 90 days.

To sign in to the service account:

1. Ensure you have signed out of **all** Microsoft accounts on your device, including the Microsoft Azure portal.
2. From the Management Node, go to **One-touch Join > OTJ Exchange Integrations**, select the Exchange integration you have just created. At the bottom of the **Change OTJ Exchange Integration** page, select **Sign in to service account**:

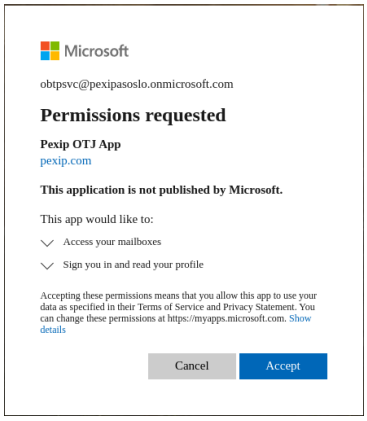


You will be taken to the **Sign in to service account** page.:



3. Copy the **Sign in** link and paste it into a new browser tab.
4. Sign in as the service account.

   You will be asked to permit the Application registration to access the service account:



5. Select **Accept**.

You should be returned to the **Sign in to service account** page and see the message **Successfully signed in**.

Next steps

You must now configure the remainder of the One-Touch Join components on Pexip Infinity, as described in Configuring Pexip Infinity for One-Touch Join.

Configuring G Suite for One-Touch Join

This topic describes how to configure G Suite in order to implement Pexip Infinity's One-Touch Join feature in a G Suite environment.

The process involves the following steps, described in more detail in the sections that follow:

1. Creating a Service Account to use for One-Touch Join.
2. Creating a room resource for each physical room that will have a One-Touch Join endpoint in it.
3. Configuring the room resource with the necessary permissions and settings to support One-Touch Join.
4. Updating the quota for the number of user requests per 100 seconds.
5. For larger deployments, Requesting an increase to API limits.
6. Adding a One-Touch Join G Suite integration on Pexip Infinity.

If you have already set up a One-Touch Join G Suite integration and simply wish to add an existing room to it, you need only configure the room resource in G Suite and then add the endpoint to the G Suite integration in Pexip Infinity.

We recommend that you authorize One-Touch Join to access calendar information using a service account, as described in the following steps. This method (sometimes referred to as two-legged OAuth) offers the easiest setup for One-Touch Join, and is recommend by Google because it is designed for server-to-server applications (for more information, see https://developers.google.com/identity/protocols/oauth2/service-account). Alternatively, you may need to use a G Suite domain user for authorization (sometimes refered to as three-legged OAuth); for instructions on how to do this, see Configuring G Suite for domain user authorization.

Prerequisites

In the deployment model described below, the service account will require access to the endpoints' calendars. G Suite service accounts always use the **iam.gserviceaccount.com** domain rather than your own domain, so you will need to configure G Suite to allow endpoint calendars to be shared externally. Some enterprises will require approval for this configuration, so you should confirm that it will be permitted within your deployment. If not, you can consider Configuring G Suite for domain user authorization as an alternative.

Creating a Service Account

In this step, you create a project to use for One-Touch Join. You then create the Service Account that One-Touch Join will use to access the room resources' calendars, and generate a private key that One-Touch Join will use to authenticate when signing in to G Suite as the Service Account.

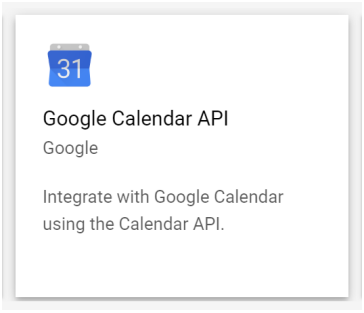
The service account belongs to the project you have created for OTJ. It can be used for multiple One-Touch JoinG Suite integrations.

1. Creating a new project:
   a. Go to https://console.developers.google.com (logged in as a G Suite administrator).
   b. From the top left of the page, select the down arrow:

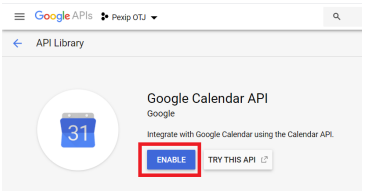
   

   c. Select **New Project**.
   d. Enter a **Project name** (e.g. One-Touch Join) and select **Create**.
2. Enabling the Calendar API for the project:
   a. Go to https://console.developers.google.com
   b. From the top left of the page, select the down arrow, select your newly-created project, and select **Open**. Your new project should now be showing at the top left of the page:

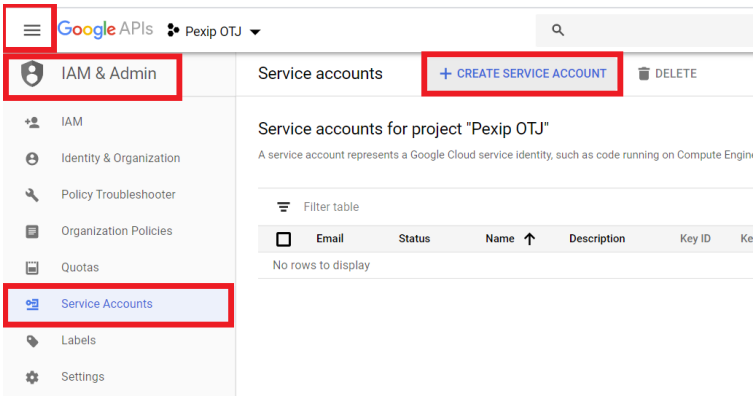
c. From the navigation menu on the left of the screen, select **APIs & Services > Library**, then scroll down and select the **Google Calendar API** tile:



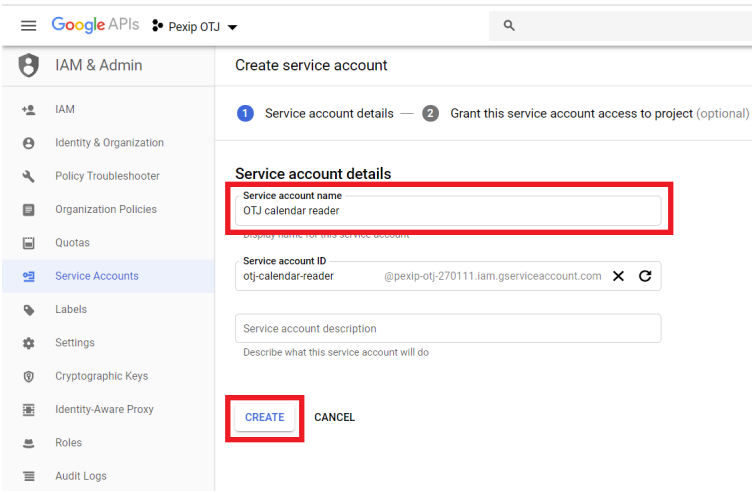
d. Select **Enable**:



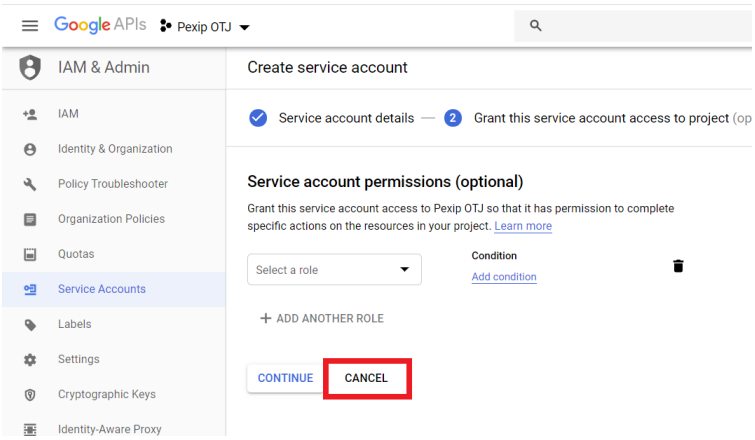
3. Creating the Service Account:
   a. Go to https://console.developers.google.com
   b. From the navigation menu on the left of the screen, select **IAM & Admin > Service Accounts**.
   c. Select **Create Service Account**:



   d. Enter a name (e.g. One-Touch Join Calendar Reader) and select **Create**:

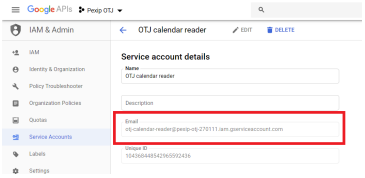


e. On the next page, which asks about permissions, select **Cancel** (the account does not need any of these permissions):

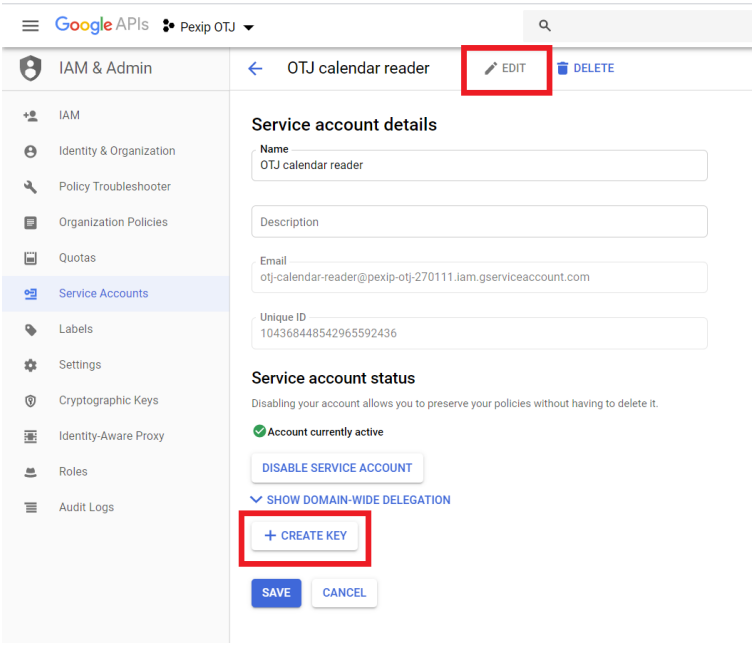


4. Generating a key file:
   a. From the **Service accounts** page, select the Service Account.

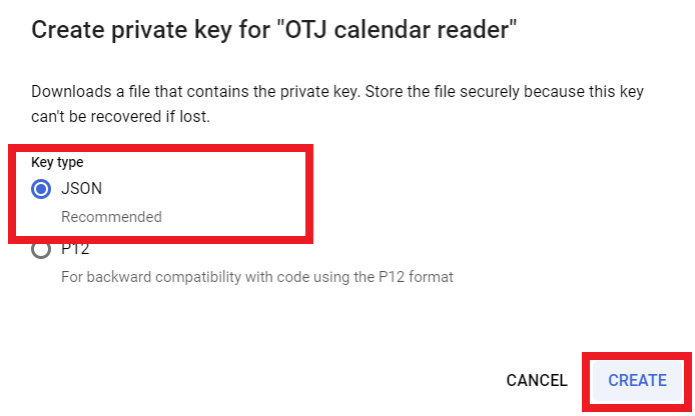
      Take note of the service account's **Email** address here - you will need it in later steps:



   b. From the **Service account details** page, select **Edit**, then **Create Key**:



c. Select a **Key type** of ***JSON*** and select **Create**:



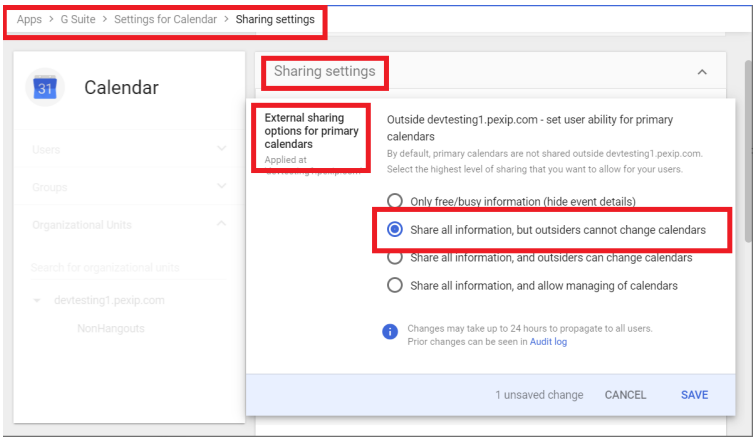
This will download a JSON file containing the private key. This key will be required when Adding a One-Touch JoinG Suite integration.

For more information on using OAuth 2.0 to authenticate the Service Account, see https://developers.google.com/identity/protocols/OAuth2ServiceAccount.

Sharing calendars externally

In this step, you configure G Suite to permit endpoint calendars within your domain to be shared with the service account (which uses the external **iam.gserviceaccount.com** domain). For more information, see Prerequisites.

1. Go to https://admin.google.com (logged in as a G Suite administrator).
2. Select **Apps > G Suite > Calendar**.
3. From the **Sharing settings** section, ensure that **External sharing options for primary calendars** is set to ***Share all information, but outsiders cannot change calendars***:

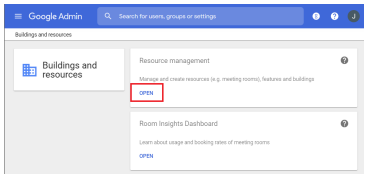


Creating a room resource

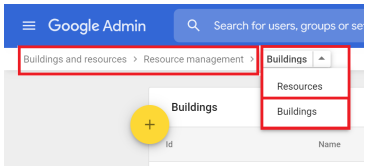
(Required only if your room resources do not already exist - otherwise you can skip this step.)

In this step, you create a room resource in G Suite for each physical room that is to be used for One-Touch Join. G Suite will automatically assign an email address to the room.

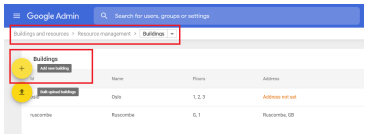
1. If a **building** for the room resource does not already exist, create one as follows:
   a. Go to https://admin.google.com (logged in as a G Suite administrator).
   b. Select the **Buildings and resources** tile, and then from the **Resource management** section select **Open**:

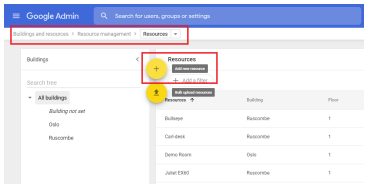
   From the drop-down along the top left of the screen, select **Buildings**:

   c. Select **+** to **Add new building**:

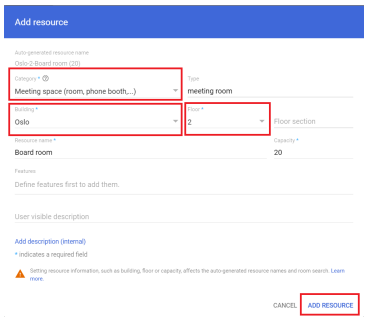
   

   d. Enter a **Name** and the list of **Floors**, and select **Add Building**.
2. Create the room resource:
   a. Go back to the **Resources** page and Select **+** to **Add new resource**:

b. For the **Category**, select ***Meeting space (room, phone booth,...)***.

c. Select the **Building** and **Floor** in which the room is located, enter a **Name** and the room's **Capacity**, then select **Add Resource**:



The resource will be created and added to the list. You can click on the new resource to view information about it, such as the email address it was automatically assigned.

For more information on setting up buildings and other resources in G Suite, including how to add buildings and resource in bulk and using CSV imports, see https://support.google.com/a/answer/1033925.

Configuring the room resource

In these steps, you allow the One-Touch Join Service Account to access the calendar of each room resource that you want to use for One-Touch Join, and set the calendar to auto-accept invitations. We also recommend that you make the calendar available to all users in your domain in such a way that allows them to book meetings using the resource, without being able to view the details of any other meetings in the resource's calendar.

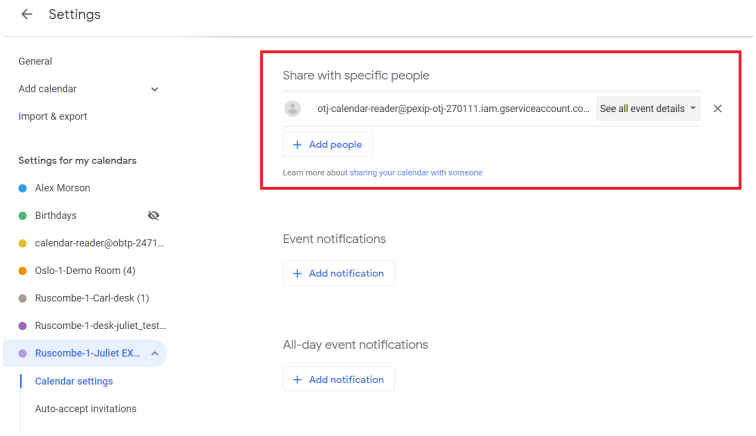
Sharing individual calendars with the service account

Note that the Google calendar API limits the number of calendars that can be shared within a 24 hour period to 750 (for more information, see https://support.google.com/a/answer/2905486?hl=en). This means that if you have more than 750 room resources that you wish to use for One-Touch Join, they will need to be set up over a period of days.

For deployments with more than around 50 rooms, we have developed a Python script that can be used to share your room resource calendars with the service account, and create a CSV that can be used to import endpoint configuration to One-Touch Join. You must be familiar with Python in order to use this script; contact your Pexip authorized support representative for more information.

1. Go to https://calendar.google.com (logged in as a G Suite administrator so that you have permission to share the calendars).
2. From the left-hand panel, select the **+** next to **Other calendars** and then select **Browse resources**.
3. Expand the sections if necessary, and tick the boxes of all the room resources whose calendars you want to share with the service account.

   This will add the room resources to the **Settings for other calendars** section in the left-hand panel.
4. For each of the rooms:
   a. From the **Settings for my calendars** section, select the room resource and then select **Share with specific people**.
   b. Select **Add people**.
   c. In the **Share with specific people** dialog, enter the email address of the One-Touch Join service account. Ensure the **Permissions** are set to ***See all event details***.
   d. Select **Send**:



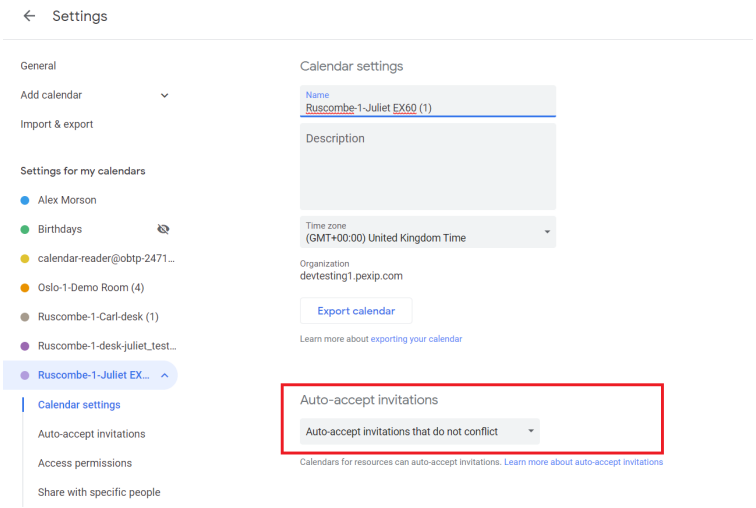
For more information on sharing room and resource calendars in G Suite, see https://support.google.com/a/answer/1034381.

Auto-accepting invitations

By default, when creating room resources in G Suite, calendar processing is set to **Auto-accept invitations that do not conflict**. You must ensure you keep this setting for all room resources, so that the room will automatically accept meeting requests if it is available, and automatically decline an invitation if it is already booked.

To check this setting:

1. Go to https://calendar.google.com (logged in as a G Suite administrator so that you have permission to share the calendars).
2. From the left-hand panel, select the room resource and select **Settings and sharing**.
3. In the **Auto-accept invitations** section, ensure that ***Auto-accept invitations that do not conflict*** is selected:

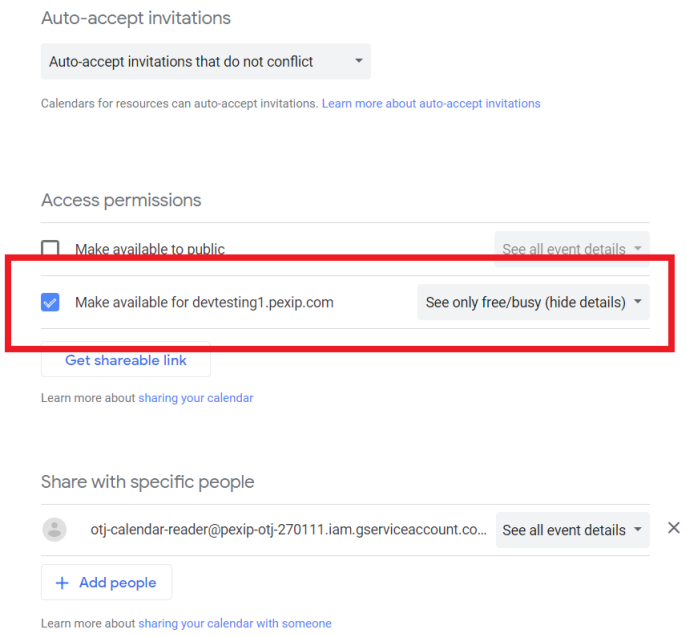


Allowing users to book resources

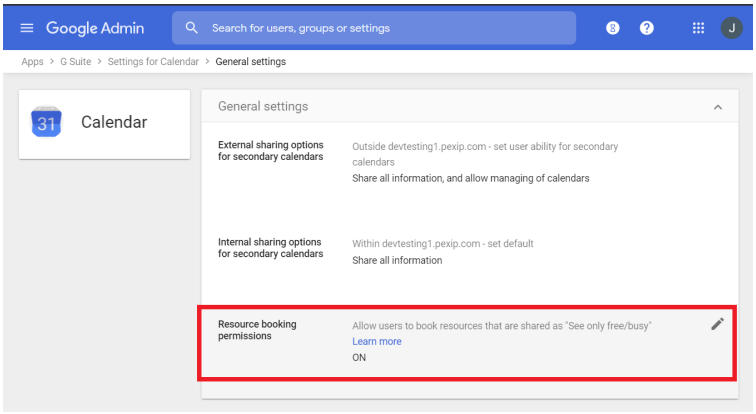
We recommend that you configure your G Suite calendar settings to allow end users to book a room resource without seeing details of the room's other bookings. To do this, you configure the room resource's calendar so that all users in your domain have permission to see its free/busy status, without being able to see the invitation details. You then on a global basis permit users to book resources to which they have free/busy access.

To do this:

1. Go to https://calendar.google.com (logged in as a G Suite administrator so that you have permission to share the calendars).
2. From the left-hand panel, select the room resource and select **Settings and sharing**.
3. In the **Access permissions** section, select **Make available for <your domain>**, and ensure that ***See only free/busy (hide details)*** is selected:



4. Go to admin.google.com (logged in as a G Suite administrator).
5. From the left-hand menu, select **Apps > G Suite > Calendar**.
6. Scroll down to **General Settings** and select **Resource Booking Permissions**.
7. Ensure that **Allow users to book resources that are shared as See only free/busy** is set to ***ON***:



Updating the per-user request quota

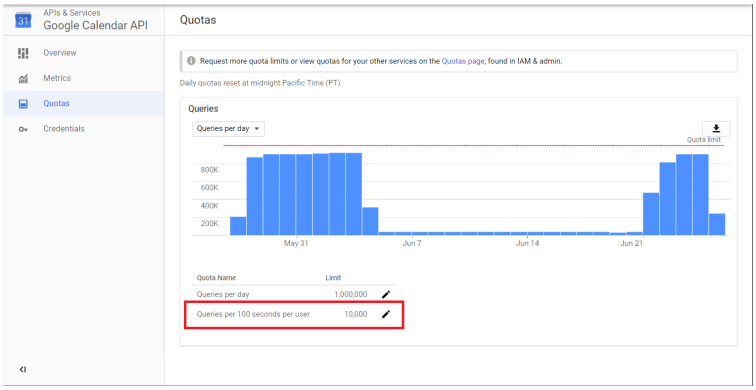
In this step you increase the limit on the number of queries per 100 seconds per user to the Google Calendar API.

The default number of queries per 100 seconds per user is 500. We recommend that you increase this to 10,000, as follows:

1. Go to https://console.developers.google.com (logged in as a G Suite administrator).
2. From the top left of the page, select the project you created for One-Touch Join:

3. From the navigation menu at the top left of the page, select **IAM & Admin > Quotas**.
4. From the Quotas page, select **Edit Quotas** and then select **Google Calendar API - Queries per 100 seconds per user**.

   You will be taken to the **Google Calendar API > Quotas** page.
5. Change **Queries per 100 seconds per user** to **10,000**:



> You may also need to request an increase to the number of **Queries per day** for larger deployments - for more information, see Requesting an increase to API limits.

Requesting an increase to API limits

This optional step applies to larger deployments only (more than around 170 room resources), and should be performed if you wish to reduce the amount of time taken for endpoints to be updated with additions or changes to their corresponding room resource calendar.

The maximum frequency with which an endpoint will be updated with meeting information is every 30 seconds. For deployments with more than around 170 endpoints, this frequency will decrease in line with the number of endpoints (up to around 20 minutes for deployments with around 6,000 endpoints). This is due to a limit on the number of Calendar API requests permitted by Google in a 24-hour period — for more information, see https://developers.google.com/calendar/pricing.

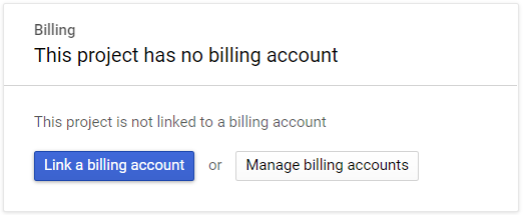
To reduce the time taken to update endpoints in these larger deployments, you can request an increase to the number of Calendar API requests One-Touch Join can make.

> When your request has been implemented by Google, you must then increase the Maximum G Suite API requests on Pexip Infinity in order to take advantage of the increase.

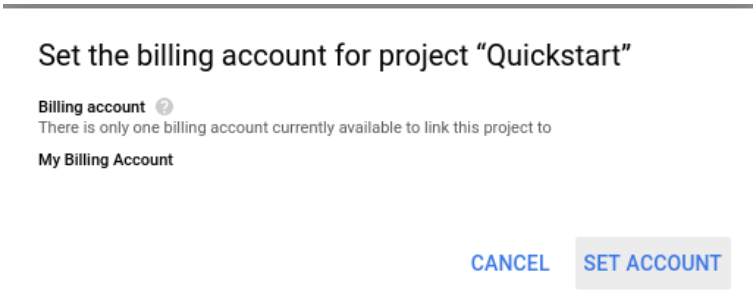
To request an increase to the API limits:

1. If you do not already have one, create a Cloud Billing Account (note that this is different from a G Suite billing account).

   Full instructions are available via https://cloud.google.com/billing/docs/how-to/manage-billing-account#create_a_new_billing_account.
2. Link the Cloud Billing Account to the project you created when Creating a service account:
   a. Go to https://console.developers.google.com (logged in as a G Suite administrator).
   b. Ensure that the project shown in the top left corner is the one you created for One-Touch Join when Creating a service account.
   c. Select the burger menu from the top left of the page and select **Billing**. When the following message appears, select **Link a billing account**:

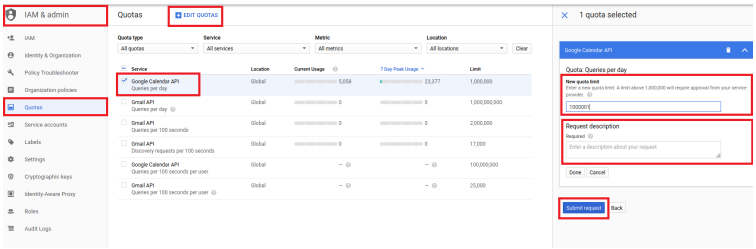


d. Select the account to link to:



3. Request an increase to your quota:
   a. From the navigation menu at the top left of the page, select **IAM & admin > Quotas**.
   b. From the Quotas page, select **Edit Quotas** and then select **Google Calendar API**.

      In the panel that appears on the right, enter the **New quota limit** that you wish to request, and in the **Request description** field, enter the reason for requesting the increase:



   c. Select **Submit request**.

Quota increase requests typically take two business days to process.

Adding a One-Touch Join G Suite integration on Pexip Infinity

In this step you configure Pexip Infinity with details of the G Suite deployment configured above, including details of the service account used to access calendars.

From the Pexip Infinity Administrator interface, go to **One-Touch Join > OTJ G Suite Integrations**.

Option	Description
Name	The name of this One-Touch JoinG Suite integration.
Description	An optional description of this One-Touch JoinG Suite integration.
Account email	If you are authorizing using a service account, enter the email address of the service account that One-Touch Join will use to log in to G Suite. If you are authorizing using a G Suite domain user, enter the email address of the user.

Option	Description
Enable user authorization	If you are authorizing using a service account — the recommended method — this should be left blank. Select this option only if you will be authorizing using a G Suite domain user.
Private key	(Available when authorizing using a service account, i.e. user consent authorization has not been enabled) The private key used by One-Touch Join to authenticate the service account when logging in to G Suite. For instructions on how to obtain this, see Generating a key file. This must include all the text in the file between (and including) `-----BEGIN PRIVATE KEY-----` and `-----END PRIVATE KEY-----`
Client ID	(Available when user consent authorization has been enabled) The client ID of the application you created in the Google API Console, for use by OTJ.
Client secret	(Available when user consent authorization has been enabled) The client secret of the application you created in the Google API Console, for use by OTJ.
Redirect URI	(Available when user consent authorization has been enabled) The redirect URI you configured in the Google API Console. It must be in the format: **https://<Management Node FQDN>/admin/platform/mjxgoogledeployment/oauth_redirect/** ℹ This must use the Management Node's FQDN; it cannot use its IP address. You must therefore ensure you have appropriate internal DNS records set up for the Management Node.
Advanced options	
Maximum G Suite API requests	The maximum number of API requests that can be made by One-Touch Join to your G Suite Domain in a 24-hour period. We recommend you set this value to 90% of your total permitted requests. Google's default is 1,000,000 so by default this is set to 900,000 on Pexip Infinity. If you increase the number of API requests, you should also increase this setting to 90% of that number. For more information, see Frequency and limitations on calendar requests.
Google OAuth 2.0 endpoint	The URI of the Google OAuth 2.0 endpoint.
Google authorization server	The URI of the Google authorization server.

Next steps

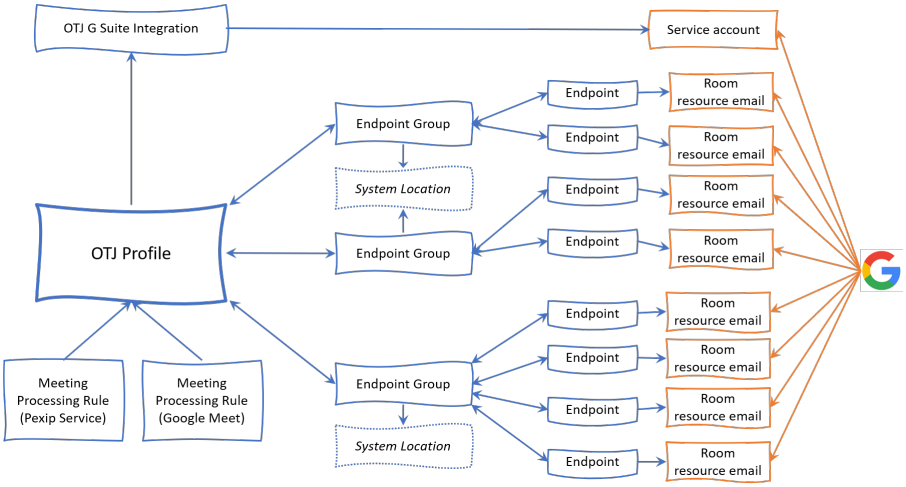
You must now configure the remainder of the One-Touch Join components on Pexip Infinity, as described in Configuring Pexip Infinity for One-Touch Join.

Configuring Pexip Infinity for One-Touch Join

This topic describes how to configure Pexip Infinity when enabling the One-Touch Join feature. It covers configuration of the various Pexip Infinity components, each described in detail in the sections that follow:

1. Adding a One-Touch Join profile
2. Adding One-Touch Join endpoint groups
3. Adding One-Touch Join endpoints
4. Adding One-Touch Join meeting processing rules

The diagram below shows (in blue) the components that are configured on Pexip Infinity and how they are related to each other. It also shows (in orange) how the Pexip Infinity components are associated with your calendar/email service — in this example we have used G Suite, with support for Google Meet and Pexip Service meeting types:



Prerequisites

Before you start configuring Pexip Infinity, you must first do one of the following, depending on your calendar/email service:

- Configure G Suite for One-Touch Join, including Adding a One-Touch Join G Suite integration on Pexip Infinity, or
- Configure Exchange on-premises for One-Touch Join, including Adding a One-Touch Join Exchange integration on Pexip Infinity
- Configure Office 365 for One-Touch Join, including Adding a One-Touch Join Exchange integration on Pexip Infinity

Adding a One-Touch Join profile

In this step you create a profile that you will use to link together all the components for this particular deployment: the Exchange or G Suite integration, the endpoint groups (and therefore endpoints), and the rules to be used to process meeting invitations.

A single Pexip Infinity One-Touch Join profile can apply to **either** an Exchange integration or a G Suite integration, but not both. However, a One-Touch Join profile can contain a mixture of Cisco and Poly endpoints.

An endpoint group, and therefore an endpoint (and its room resource), can belong to only one One-Touch Join profile. If you do not assign an endpoint group to a One-Touch Join profile, the endpoints in that group will not be used for One-Touch Join.

To add a One-Touch Join profile, from the Pexip Infinity Administrator interface, go to **One-Touch Join > OTJ Profiles**.

Option	Description
Name	The name of this One-Touch Join profile.
Description	An optional description of this One-Touch Join profile.
No. of upcoming days	The number of days of upcoming One-Touch Join meetings to be shown on endpoints. This will also be the number of days of future meetings shown on the One-Touch Join Status page.
Enable non-video meetings	***Enabled***: If One-Touch Join has not been able to obtain a video address from the meeting, then the meeting will still appear on the endpoint as a scheduled meeting, showing the information that was able to be parsed, but the **Join** button will not appear. ***Disabled***: If there is no video address, the available meeting information will not appear on the endpoint. Note that the meeting will still exist in the room resource's calendar, so conflicting meetings cannot occur.
Enable private meetings	Determines whether or not meetings flagged as private are processed by the One-Touch Join service. ***Enabled***: Private meetings will be processed in the same way as any other meeting. **Disabled**: Private meetings are not processed by One-Touch Join, and therefore the meeting information will not appear on the endpoint. Note that the meeting will still exist in the room resource's calendar, so conflicting meetings cannot occur. Note that if this is set to ***Enabled***, you can still prevent private meeting details from being displayed on individual Poly endpoints by disabling the endpoint's **Show Private Meeting Information** setting.
Process alias for private meetings	(Applies if **Enable private meetings** has been selected) ***Enabled***: For private meetings, the meeting alias will be extracted from the invitation in the usual way. ***Disabled***: For private meetings, the available meeting information — apart from the meeting alias — will appear on the endpoint and therefore the **Join** button will be disabled.
Replace subject for private meetings	(Applies if **Enable private meetings** has been selected) ***Enabled***: For private meetings, the endpoint will display the organizer's name in place of the meeting subject. ***Disabled***: For private meetings, the endpoint will display the meeting subject in the usual way.
Replace empty subject	***Enabled***: For meetings that do not have a subject, the endpoint will display the organizer's name in place of the subject. ***Disabled***: For meetings that do not have a subject, the endpoint will display a blank field in place of the subject.
Exchange integration	(Applies if this OTJ profile is for an Exchange or Office 365 integration) The Exchange integration used by this One-Touch Join profile. You should already have created this as part of either Configuring Exchange on-premises for One-Touch Join or Configuring Office 365 for One-Touch Join, but you can configure it now by selecting the green plus symbol + to the right of the field.
G Suite integration	(Applies if this OTJ profile is for a G Suite integration) The G Suite integration used by this One-Touch Join profile. You should have already created this as part of Configuring G Suite for One-Touch Join, but you can configure it now by selecting the green plus symbol + to the right of the field.
Endpoint Groups	The Endpoint Groups used by this One-Touch Join profile.

Option	Description
Cisco endpoint configuration options	
Start buffer	The number of minutes before a meeting's scheduled start time that the "Join" button on the endpoint will become enabled for that meeting. An endpoint can offer more than one "Join" button if there is an overlap between different meetings' start and end buffers.
End buffer	The number of minutes after a meeting's scheduled end time that the "Join" button on the endpoint will become disabled for that meeting.
Default API username Default API password	The user name and password used by One-Touch Join to access a Cisco OBTP endpoint's API. The API is used by the Conferencing Node to configure the endpoint with meeting information. The account being used must have a role of either ***User*** or ***Admin***. The **Default API username** and **password** is only used if the configuration for the Cisco OBTP endpoint in within One-Touch Join does not include an API username and password. A default is offered because some deployments will have the same username and password for all endpoints.
Verify endpoint certificates by default	Whether or not to verify the TLS certificate of a Cisco OBTP endpoint by default when accessing its API. Can be overridden per endpoint using the endpoint's Verify endpoint API TLS certificate setting. For more information, see Managing trusted CA certificates.
Use HTTPS for endpoint API	Whether or not to use HTTPS by default when accessing a Cisco OBTP endpoint's API. Can be overridden per endpoint using the endpoint's Use HTTPS setting. ***Enabled***: Use HTTPS to access an endpoint's API. ***Disabled***: Use HTTP to access an endpoint's API.
Webex endpoint configuration options	
Enable Webex Cloud Sign In	Enable webex cloud sign in to connect to webex edge registered endpoints
Client ID	The Client ID created when registering your Webex Integration. Maximum length: 100 characters.
Redirect URI	The redirect URI you entered when creating an Integration with Webex It should be in the format 'https://<Management Node Address>/admin/platform/mjxintegration/oauth_redirect/
Client secret	The Webex Integration Client Secret.

Adding One-Touch Join endpoint groups

In this step you create endpoint groups, and optionally add endpoints to each group. Each endpoint can belong to only one endpoint group; an endpoint group can contain a mix of Cisco OBTP and Poly OTD endpoints. In general, we recommend that all endpoints in the same physical location are assigned to one endpoint group.

Each endpoint group is associated with a system location; if there are more than 5 Conferencing Node in one location, only 5 will be actively running One-Touch Join. This is because each Conferencing Node will be connecting to Exchange, and the messaging overhead needs to be limited.

From the Pexip Infinity Administrator interface, go to **One-Touch Join > OTJ Endpoint Groups**.

Option	Description
Name	The name of this One-Touch Join endpoint group.

Option	Description
Description	An optional description of this One-Touch Join endpoint group.
System location	The system location of the Conferencing Nodes which will provide One-Touch Join services for this endpoint group.
OTJ profile	The One-Touch Join profile to which this endpoint group belongs.
Endpoints	The endpoints that belong to this One-Touch Join endpoint group.

Adding One-Touch Join endpoints

In this step you add details of the endpoints that will be used for One-Touch Join, and the room resource that each endpoint is associated with. You can add endpoints individually, or in bulk using a CSV import.

After you have added details of your One-Touch Join endpoints to Pexip Infinity, you will also need to configure the settings on each endpoint to support One-Touch Join. We recommend that you do this after you have completed the following configuration.

If there are multiple endpoints in a single room, you should associate each endpoint with the same room resource, so that each endpoint will receive the same meeting details.

Adding endpoints individually

From the Pexip Infinity Administrator interface, go to **One-Touch Join > OTJ Endpoints**.

Option	Description
Name	The name of this One-Touch Join endpoint.
Description	An optional description of this One-Touch Join endpoint.
Endpoint type	The type of "click to join" feature supported by this endpoint. ***Cisco OBTP***: an endpoint that supports Cisco's One Button to Push (OBTP). You should ensure that this endpoint has already been set up in accordance with Configuring Cisco OBTP endpoints for OTJ. ***Poly OTD***: an endpoint that supports Poly's One Touch Dial (OTD). You must complete the steps in this Adding One-Touch Join Endpoints section **before** you set up your Poly endpoints in accordance with Configuring Poly OTD endpoints for OTJ. ***Webex Edge***:
Configuration options for Cisco OBTP endpoints	
Endpoint API address	The IP address or FQDN of the endpoint's API.
Endpoint API port	The port of the endpoint's API. Default: 443 if HTTPS is used, otherwise 80 for HTTP.
Endpoint API username Endpoint API password	The user name and password used by One-Touch Join to access a Cisco OBTP endpoint's API. The API is used by the Conferencing Node to configure the endpoint with meeting information. The account being used must have a role of either ***User*** or ***Admin***. Either both these fields must be configured, or both these fields must be left blank. If both these fields are left blank, the One-Touch Join profile's Default API username and password will be used.

Option	Description
Verify endpoint API TLS certificate	Whether to enable TLS verification when accessing this endpoint's API. Only applicable if using HTTPS to access this endpoint's API. ***Use OTJ profile default***: Use the Verify endpoint certificates by default setting configured for the One-Touch Join profile that this endpoint is associated with. ***On***: Enable TLS verification. ***Off***: Do not use TLS verification. For more information, see Managing trusted CA certificates.
Use HTTPS	Whether to use HTTPS to access this endpoint's API. ***Use OTJ profile default***: Use the Use HTTPS for endpoint API setting configured for the One-Touch Join profile that this endpoint is associated with. ***On***: Use HTTPS to access this endpoint's API. ***Off***: Use HTTP to access this endpoint's API.
Configuration options for Poly OTD endpoints	
Poly Calendaring Username	The username the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information. This must be the same as the **User Name** or **User** (the field name will vary) configured on the Poly endpoint.
Poly Calendaring password	The password the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information. This must be the same as the **Password** configured on the Poly endpoint.
Raise alarms	When enabled, an alarm will be raised if this endpoint has not contacted the calendaring service on the Conferencing Node in the last 10 minutes.
Configuration options for Webex Edge endpoints	
Webex Device ID	
Configuration options for all endpoints	
Raise alarms	When enabled, an alarm will be raised if this endpoint has not contacted the calendaring service on the Conferencing Node in the last 10 minutes.
Room resource email	The email address of the room resource associated with this endpoint. This must match an email address that has been configured in Exchange or G Suite. For Poly endpoints, this must be the same as the **Email** or **Mailbox** (where this setting is available) configured on the Poly endpoint.
Endpoint Group	The Endpoint Group to which this endpoint belongs.

Adding OTJ endpoints in bulk

You can add multiple One-Touch Join endpoints by importing a CSV file.

When formatting your import file:

- A header row in the CSV file is optional. If included, it must use the same field names as shown in the following sections, but you may change the order of the fields. If a header row is not used, fields must be in the same order as shown.
- All non-blank fields must contain valid data.
- If non-ASCII characters are used, the file must be encoded as UTF-8 text.

- All fields are case-sensitive.
- Values may optionally be enclosed in double quotation marks; any strings containing commas must be enclosed in double quotation marks e.g. "description for x, y and z".

Note that you can perform an export of existing data to produce an example file in the correct format.

To add multiple endpoints by importing a CSV file:

1. Create the CSV file, using the following format:

```
name,description,endpoint_type,api_address,api_port,api_username,api_password,poly_username,poly_password,verify_
cert,use_https,room_resource_email,mjx_endpoint_group_name
```

where

Field name	Content	Required field for...
name	This field cannot be blank. The name of this One-Touch Join endpoint. You should ensure there are no duplicate names, either within the CSV file, or between the CSV file and the existing endpoints (unless you wish the existing configuration to be overwritten).	Cisco Poly
description	An optional description of this One-Touch Join endpoint.	
endpoint_type	The type of "click to join" feature supported by this endpoint. Valid values are: ○ CISCO ○ POLY	Cisco Poly
api_address	The IP address or FQDN of the Cisco OBTP endpoint's API.	Cisco
api_port	The port of the Cisco OBTP endpoint's API. If this is left blank, the defaults (443 if HTTPS is used, otherwise 80 for HTTP) will be used.	
api_username	The username used by OTJ to access the Cisco OBTP endpoint's API.	
api_password	The password used by OTJ to access the Cisco OBTP endpoint's API.	
poly_username	The username the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	Poly
poly_password	The password the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	Poly
verify_cert	Whether to enable TLS verification when accessing this Cisco OBTP endpoint's API. Only applicable if using HTTPS to access this endpoint's API. Valid values are: ***GLOBAL***: Use the Verify endpoint certificates by default setting configured for the One-Touch Join profile that this endpoint is associated with. ***YES***: Enable TLS verification. ***NO***: Do not use TLS verification.	

Field name	Content	Required field for...
use_https	Whether to use HTTPS to access this Cisco OBTP endpoint's API. ***GLOBAL***: Use the Use HTTPS for endpoint API setting configured for the One-Touch Join profile that this endpoint is associated with. ***YES***: Use HTTPS to access this endpoint's API. ***NO***: Use HTTP to access this endpoint's API.	
room_resource_email	This field cannot be blank. The email address of the room resource associated with this endpoint. This must match the email address that has been configured in Exchange or G Suite.	Cisco Poly
mjx_endpoint_group_name	The endpoint group to which this endpoint belongs. If this field is set, it must contain the name of an existing endpoint group.	

2. From the Pexip Infinity Administrator interface, go to **One-Touch Join > OTJ Endpoints** and from the bottom right of the screen, select **Import**.
3. From the **Import OTJ Endpoint Configuration** page, select **Choose file** and then navigate to the CSV file you have created.
4. Select **Save**.

The imported endpoints will be added to your One-Touch Join configuration.

Duplicates

If any records in the CSV file have the same **name** field (regardless of whether or not any of the other fields are different), only one endpoint with that name will be created. This endpoint will use the last record that was imported.

If any records in the CSV file have the same **name** as an existing endpoint, the existing configuration will be overwritten by the imported endpoint's configuration.

Adding One-Touch Join meeting processing rules

In this step you create a prioritized set of rules that specifies each of the meeting types you expect users in your deployment to encounter, and how the invitations for these meetings should be processed in order to obtain the alias that the endpoint must dial in order to join the meeting.

One-Touch Join supports meetings from a number of different providers. For each of these supported meeting types, One-Touch Join knows what information to look for in the meeting invitation, and how to use what it finds to derive an alias that the endpoint can dial in order to join that meeting. In most cases, you can simply use the default processing for each supported meeting type. However, you also have the option to override the default processing with your own transform pattern to change how the alias is constructed. You can also write your own regex and custom rules if you wish to enable One-Touch Join for other meeting types or conferencing providers not currently supported.

A single One-Touch Join profile will normally have multiple meeting processing rules associated with it — we recommend that you create one rule for each Meeting type you expect users in your environment to encounter, including any invitations received from external contacts where users may wish to use an internal meeting room to join the meeting. The **Priority** option should be used to ensure that all rules for supported meeting types are processed before any ***Domain***, ***Regex*** or ***Custom*** rules. (Note that the order in which the supported meeting types are prioritized between themselves is not important.)

When One-Touch Join processes a meeting invitation, it goes through each meeting rule in order of priority to find a match.

- If a match is found, it uses the information in the invitation, processed in accordance with the rule's settings, to derive an alias to use to join the meeting.
- If none of the meeting processing rules match (or there are no meeting processing rules configured or enabled), One-Touch Join will search the invitation for a URI or address with a **sip:**, **sips:** or **h323:** prefix, and use that as the alias.

One-Touch Join then provides the endpoint with the alias, along with other meeting information such as the start time, end time, subject, and organizer's name.

If no alias has been obtained, One-Touch Join may still provide the meeting information to the endpoint, depending on the Enable non-video meetings and Enable private meetings settings for the profile being used.

Each meeting processing rule is associated with a single One-Touch Join profile, and therefore will apply to either an Exchange integration or a G Suite integration, but not both.

To view, edit and create meeting processing rules, from the Pexip Infinity Administrator interface, go to **One-Touch Join > OTJ Meeting Processing Rules**.

Option	Description
Name	The name of this One-Touch Join meeting processing rule.
Description	An optional description of this meeting processing rule.
OTJ profile	The One-Touch Join profile associated with this meeting processing rule.
Priority	The priority of this rule. Rules are checked in ascending priority order (starting at 1) until the first matching rule is found, and it is then applied. We recommend that meeting types other than ***Domain***, ***Regex*** or ***Custom*** are given highest priority. You can then use lower **Priority** options to determine the order in which any ***Domain***, ***Regex*** and ***Custom*** rules are applied, particularly if you are using more than one of these meeting types.
Meeting type	The type of meeting invitation to which this rule applies. You can select one of the supported meeting types from the drop-down list, or select ***Regex*** or ***Custom*** if you wish to define your own meeting processing rule. For a full list of available meeting types, and guidance on which to use in your deployment, particularly when joining Teams or Skype for Business meetings, see Supported meeting types.

Default processing enabled	(Does not apply to ***Custom*** meeting types)
	• For meeting types other than ***Regex***:
	  ○ check this box to use the default transform pattern for the selected meeting type (for a list of the default transform patterns for each meeting type, see Supported meeting types), or
	  ○ clear this box to write your own **Transform pattern** for this meeting type.
	• For ***Regex*** meeting type:
	  ○ check this box to use the matched string, unchanged, as the alias that the endpoint will dial to join the meeting, or
	  ○ clear this box to use a regex **Replace string** to transform the matched string into the alias to dial.
	For more information, see Regex meeting type.

Transform pattern	(Available and required when **Default processing** is disabled and any **Meeting type** option other than ***Custom*** or ***Regex*** has been selected.) A Jinja2 snippet that is used to process the meeting information from calendar events of the selected **Meeting type** in order to derive the meeting alias. If you disable **Default processing** after creating and saving the rule, this field will show the default transform pattern, which you can then edit. For a list of the valid variables for each meeting type, see Supported meeting types.
Match string	(Available and required when a **Meeting type** of ***Regex*** has been selected.) The regular expression that defines the string to search for in the invitation.

Option	Description
Replace string	(Available and required when **Default processing** is disabled and a **Meeting type** of ***Regex*** has been selected.) A regular expression that defines how to transform the matched string into the alias to dial.
Domain	(Available and required when a **Meeting type** of either ***Domain*** or ***Microsoft Teams Meeting Properties*** has been selected.) The domain from which the meeting invitation was sent. • For a **Meeting type** of ***Domain***, this is the domain that OTJ will search for in the meeting body, in order to match this rule. • For a **Meeting type** of ***Microsoft Teams Meeting Properties***, this is the domain that OTJ will append to the meeting ID after the rule has been matched, in order to create the alias that the endpoint will dial to join the meeting.
Custom template	(Available and required when a **Meeting type** of ***Custom*** has been selected.) A Jinja2 script which is used to process the meeting information from calendar events in order to extract the meeting alias. For more information, see Custom meeting type.
Enabled	Determines whether or not the rule is enabled. Any disabled rules still appear in the rules list but are ignored. Use this setting to test configuration changes, or to temporarily disable specific rules.

Testing the rule

When you have created and saved a meeting processing rule, a **Test OTJ Meeting Processing Rule** button will appear at the bottom of the page. This will take you to the **Test Meeting Processing** page, which allows you to test that the rule works as expected for the selected deployment and meeting type, and also allows you to edit the configuration for that rule until you get the desired results.

When searching a meeting invitation for the text to transform into an alias, OTJ will search either the invitation's properties, or the invitation's body (depending on the selected **Meeting type**) — and so when testing a rule, you will see either a **Calendar event properties** field or a **Calendar event body** field as appropriate. These fields will in most cases contain some example text in the format expected by OTJ, but you can enter other text here to help you test the rule, for example if you know that the format will be different in your deployment. However, since these two fields are there purely to assist you when testing the rule, and do not make up part of the rule itself, any changes to these fields will not be saved.

To test the rule:

1. Review and complete the following fields:

Option	Description
Read-only fields	
Integration type	This read-only field shows whether the rule will be applied to a G Suite or Exchange integration. This is based on the integration option selected in the OTJ profile associated with the rule.
Meeting type	This read-only field shows the meeting type associated with this rule.
Configuration that can be edited and saved	
The available fields will depend on the selected meeting type. You can edit these fields and re-test the rule until you get the desired results.	
Domain	The **Domain** currently configured for this rule.
Match string Replace string	The **Match string** (and **Replace string**, where applicable) currently configured for this rule.
Transform pattern	The pattern that will be used to transform specific text in the meeting invitation into an alias to dial. ○ If you selected **Default processing enabled**, this will be the default transform pattern for this meeting type. ○ If you did not select **Default processing enabled**, this will be the **Transform pattern** you entered.
Custom template	The **Custom template** currently configured for this rule.
Example text used when testing the rule	
Calendar event properties	(Available for some meeting types) A JSON field representing the event properties that OTJ expects to find for the selected **Meeting type** (for G Suite integrations, this will contain a subset of the Google Event Properties; for Exchange integrations, this will be the Exchange MAPI Properties). This data will be used to generate the meeting alias. In most cases this field will be populated automatically, but you can edit it if you know that the format used in your deployment will be different.
Calendar event body	(Available for some meeting types) An example of the text that OTJ expects to find in the body of the invitation for the selected **Meeting type**, and which will be used to generate the meeting alias. In most cases this will be populated automatically, but you can paste in the full text from an actual meeting invitation used in your deployment and test the rule against this.

2. Select **Test OTJ Meeting Processing Rule**.

   The **Result** field shows the meeting alias that would be extracted based on the rule's current configuration and the example calendar event properties or body.
   - If this is blank, the example calendar event properties / body did not contain any text that could be matched and transformed according to the rule as currently configured.
   - If the result is not as expected, edit the fields above as appropriate.
3. When the configuration is producing the desired result, to save the changes you have made, select **Save changes and return**.

Next steps

You should now complete the steps in Configuring endpoints to support One-Touch Join for each endpoint.

Configuring endpoints to support One-Touch Join

This topic describes how to configure each of the supported endpoint types — Cisco OBTP, Poly OTD and Webex Edge — so they can be used with Pexip Infinity One-Touch Join.

Prerequisites

We recommend that you have already completed the steps in Configuring Pexip Infinity for One-Touch Join. In particular, you will need some of the information that you previously entered when Adding One-Touch Join endpoints to Pexip Infinity, in order to complete the configuration on each endpoint.

Configuring Cisco OBTP endpoints for OTJ

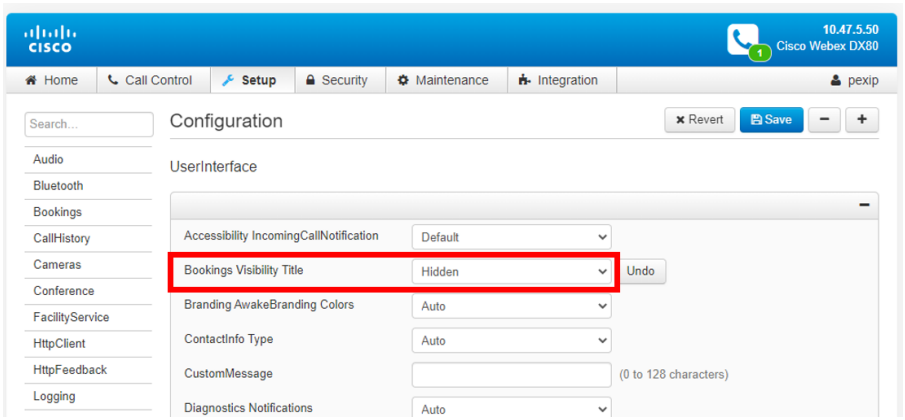
In order for Cisco OBTP endpoints to obtain One-Touch Join meeting information, the Conferencing Node associated with the endpoint uses the endpoint's API to push the information out to the endpoint.

The endpoint must have an account set up with a role of ***Admin*** that can be used by One-Touch Join to access the endpoint's API.

Hiding or changing the meeting subject

The next major release of Pexip Infinity, version 25, will include options to hide or change the meeting subjects that will appear on OTJ endpoints, according to rules that can be applied across your One-Touch Join deployment. In the meantime, you can hide meeting subjects from appearing on the screen of individual Cisco endpoints (running CE 9.12.3 or later) by configuring the endpoint as follows:

1. On the Cisco endpoint web UI, go to **Setup > Configuration > User Interface**.
2. For **Bookings Visibility Title**, select ***Hidden***:

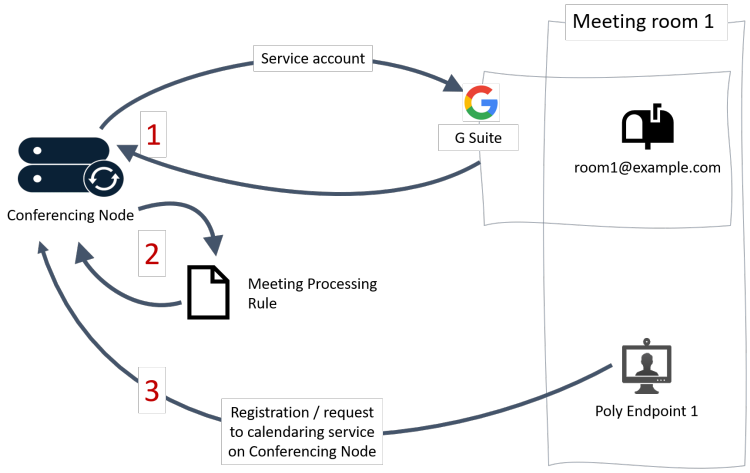


Configuring Poly OTD endpoints for OTJ

In order for Poly OTD endpoints to obtain One-Touch Join meeting information, each One-Touch Join Conferencing Node emulates a Microsoft Exchange server. The Poly endpoint then connects to the Conferencing Node and registers to the calendaring service on the node in order to pull meeting information, as shown in the diagram below.

Note that this emulation of an Exchange calendaring service on the Conferencing Node is purely to provide the Poly endpoint with its meeting information. It is completely separate to the process by which the Conferencing Node initially obtains the meeting information from the calendar/email service being used for One-Touch Join — which can be either Exchange or G Suite.

It is important that you do not set up your Poly endpoints until after you have completed the steps to add the endpoint details to Pexip Infinity.



DNS records

If you have a One-Touch Join deployment that includes Poly endpoints in a location with more than one Conferencing Node, you should spread the Poly endpoint registrations across all nodes in the location to maximize performance and provide redundancy. To achieve this, we recommend that all Poly endpoints in a location register to a single FQDN which uses round-robin DNS to resolve to each Conferencing Node in turn. This will require you to set up appropriate DNS records for all Conferencing Nodes in the location, and ensure that your DNS server is configured to round-robin between these records.

For more information and examples, see Enabling Poly endpoints to register to One-Touch Join.

Poly authentication

In normal Pexip Infinity usage Poly endpoints authenticate to One-Touch Join using digest authentication, with the exception of HDX endpoints which require NTLMv2.

When Pexip Infinity has been deployed in a secure mode of operation (and therefore FIPS compliance has been enabled), NTLMv2 and digest authentication are disabled and basic authentication is used. As a result, when in this mode:

- HDX endpoints are not supported
- Trio endpoints must be configured to allow basic authentication.

Configuring Poly RealPresence Group series

To configure a Poly RealPresence Group Series for One-Touch Join, use the following settings:

Field	Poly configuration	Matching Infinity configuration	Additional info
Email	The email address of the room resource configured in Exchange or G Suite that is associated with this endpoint.	This must be the same as the Room resource email configured on Pexip Infinity for this endpoint.	
Domain	Leave blank.		This is the Exchange domain, and is not required for One-Touch Join.
User Name	The username the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring Username configured on Pexip Infinity for this endpoint.	
Password	The username the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring password configured on Pexip Infinity for this endpoint.	
Auto Discover Using	Do not select this button.		Auto Discovery is not supported. Instead, you should manually configure the **Microsoft Exchange Server** settings.
Microsoft Exchange Server	• If you have a single Conferencing Node in this location, enter the IP Address or FQDN of the node (in the format **192.168.0.0** or **host.example.com**). • If you have multiple Conferencing Nodes in this location, you should use DNS round robin; therefore this will be the FQDN of the DNS record for this location (in the format **host.example.com**). In both cases, the location is the Pexip Infinity location associated with the Endpoint Group to which this endpoint belongs.		
Secure Connection Protocol	Select ***Automatic***.		
Meeting Reminder Time in Minutes	Optional		Can still be used in conjunction with One-Touch Join.
Play Reminder Tone When Not in a Call	Optional		Can still be used in conjunction with One-Touch Join.
Show Information for Meetings Set to Private	Optional		**Enable private meetings** must be enabled on the One-Touch Join Profile associated with this endpoint in order for this setting to apply. If **Enable private meetings** has been disabled on the One-Touch Join Profile, this setting will have no effect.

To confirm that the Poly RealPresence Group Series endpoint has registered successfully with the calendaring service:

1. On the endpoint, go to **Admin Settings > Servers > Calendaring Service**.
2. Confirm that the **Registration Status** is showing as ***Registered***.

Configuring Poly Trio series

When configuring Poly Trio series endpoints for One-Touch Join, you should use a Generic base profile unless your deployment specifically requires you to use a Skype for Business base profile. Configuration instructions for each are given below.

Configuring Poly Trio using a generic base profile

1. Open the endpoint's web configuration utility at **https://<ipaddress>**, select ***Admin***, and log in using the admin password.
2. From **Simple Setup > Base Profile** select ***Generic*** and then select **Save**.
3. Edit the config file as follows (this can be done via **Utilities > Import & Export Configuration**):
   - add `feature.contactPhotoIntegration.enabled="0"`
   - if the Trio is running software version 5.9.1.11135 or later and Pexip Infinity has been deployed in a secure mode of operation (and therefore FIPS compliance has been enabled), you must allow the Trio to use basic authentication: add `feature.exchange.allowBasicAuth="1"`
4. From **Settings > Applications**, configure the Poly trio as follows:

Field	Poly configuration	Matching Infinity configuration	Additional info
Exchange Applications			
Exchange Calendar	Select **Enable**.		
Auto Discover Using	Select **Disable**.		Auto Discover is not supported. Instead, you should manually configure the **Exchange Server URL** settings.
Exchange Server URL	Enter **https://<address>/EWS/Exchange.asmx** For **<address>**: ○ If you have a single Conferencing Node in this location, enter the IP Address or FQDN of the node. ○ If you have multiple Conferencing Nodes in this location, you should use DNS round robin; therefore this will be the FQDN of the DNS record for this location. In both cases, the location is the Pexip Infinity location associated with the Endpoint Group to which this endpoint belongs.		
Exchange Sign-In *			
Exchange Email	Leave blank.		
Domain	Leave blank.		This is the Exchange domain, and is not required for One-Touch Join.

Field	Poly configuration	Matching Infinity configuration	Additional info
User	The username the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring Username configured on Pexip Infinity for this endpoint.	
Password	The password the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring password configured on Pexip Infinity for this endpoint.	

* Available for endpoints running version 5.9.2.7727 or later. For earlier versions, you must have physical access to the endpoint's touch screen; use this to log in using the **User** and **Password** credentials as described above.

To confirm that the Poly Trio endpoint has registered successfully with the calendaring service:

1. On the endpoint, go to **Diagnostics > Exchange Status**.
2. Confirm that **Exchange Calendar** is showing as ***Synchronized***.

Configuring Poly Trio using Skype for Business base profile

You should only use the Skype for Business base profile if specifically required in your deployment (for example, if you wish to place PSTN calls via Skype for Business server); otherwise use the generic base profile.

When the Trio is configured as described below, it will still register with the calendaring service on the Conferencing Node to obtain meeting information, but it will also register with Skype for Business and use that to place outbound calls.

1. Open the endpoint's web configuration utility at **https://<ipaddress>**, select ***Admin***, and log in using the admin password.
2. From **Simple Setup > Base Profile** select ***Skype for Business*** and then select **Save**.
3. Edit the config file as follows (this can be done via **Utilities > Import & Export Configuration**):
   - add `feature.exchangeVoiceMail.enabled="0"`
   - add `exchange.showSeparateAuth="1"`
   - add `feature.exchangeContacts.enabled="0"`

4. From **Settings > Skype For Business SignIn**, configure the Poly trio as follows:

Field	Poly configuration	Matching Infinity configuration	Additional info
Skype for Business			
Use User Credentials	Select **Enable**.		
Authentication Type	Select **User Credentials**.		
Sign-in Address	The endpoint's Skype for Business address.		
Domain	The endpoint's Skype for Business domain.		
User	The name the endpoint will use to authenticate with Skype for Business.		
Password	The password the endpoint will use to authenticate with Skype for Business.		
Microsoft Exchange Server Configuration			
Exchange Email	The email address of the room resource that is associated with this endpoint.	This must be the same as the Exchange target mailbox configured on the endpoint, and Room resource email configured on Pexip Infinity for this endpoint.	
Exchange Domain	Leave blank.		
Exchange User	The username the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring Username configured on Pexip Infinity for this endpoint.	
Exchange Password	The password the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring password configured on Pexip Infinity for this endpoint.	
Exchange Target Mailbox	The email address of the room resource that is associated with this endpoint.	This must be the same as the Exchange Email configured on the endpoint, and Room resource email configured on Pexip Infinity for this endpoint.	

5. From **Settings > Applications**, configure the Poly trio as follows:

Field	Poly configuration	Matching Infinity configuration	Additional info
Exchange Applications			
Exchange Calendar	Select **Enable**.		
Auto Discover	Select **Disable**.		Auto Discover is not supported. Instead, you should manually configure the **Exchange Server URL** settings.
Exchange Server URL	Enter **https://<address>/EWS/Exchange.asmx** For **<address>**: ○ If you have a single Conferencing Node in this location, enter the IP Address or FQDN of the node. ○ If you have multiple Conferencing Nodes in this location, you should use DNS round robin; therefore this will be the FQDN of the DNS record for this location. In both cases, the location is the Pexip Infinity location associated with the Endpoint Group to which this endpoint belongs.		

To confirm that the Poly Trio endpoint has registered successfully with the calendaring service:

a. On the endpoint, go to **Diagnostics > Exchange Status**.

b. Confirm that **Exchange Calendar** is showing as ***Synchronized***.

Configuring Poly HDX series

To configure the Poly HDX for One-Touch Join, go to the endpoint's IP address.

From **Admin Settings > Global Services > Calendaring Service**, enter the following:

Field	Poly configuration	Matching Infinity configuration	Additional info
Enable Calendaring Service	Select this option.		

Field	Poly configuration	Matching Infinity configuration	Additional info
Microsoft Exchange Server Address	• If you have a single Conferencing Node in this location, enter the IP Address or FQDN of the node (in the format **192.168.0.0** or **host.example.com**). • If you have multiple Conferencing Nodes in this location, you should use DNS round robin; therefore this will be the FQDN of the DNS record for this location (in the format **host.example.com**). In both cases, the location is the Pexip Infinity location associated with the Endpoint Group to which this endpoint belongs.		
Domain	Leave blank.		This is the Exchange domain, and is not required for One-Touch Join.
User Name	The username the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring Username configured on Pexip Infinity for this endpoint.	
Password	Select this option. The following two fields will appear:		
New Password Confirm Password	The password the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring password configured on Pexip Infinity for this endpoint.	
Mailbox	For Exchange integrations: the email address of the room resource configured in Exchange that is associated with this endpoint. For G Suite integrations: the User Name entered above.	For Exchange integrations, this must be the same as the Room resource email configured on Pexip Infinity for this endpoint.	
Reminder Time in Minutes	Optional		Can still be used in conjunction with One-Touch Join.
Play Reminder Tone	Optional		Can still be used in conjunction with One-Touch Join.
Show Private Meeting Information	Optional		**Enable private meetings** must be enabled on the One-Touch Join Profile associated with this endpoint in order for this setting to apply. If **Enable private meetings** has been disabled on the One-Touch Join Profile, this setting will have no effect.

To confirm that the Poly HDX endpoint has registered successfully with the calendaring service:

1. On the endpoint, go to **Admin Settings > Global Services > Calendaring Service**.
2. Confirm that there is a green tick next to **Enable Calendaring Service**.

Configuring Poly Studio X series and Poly G7500 series

To configure the Poly Studio or Poly G7500 for One-Touch Join, go to the endpoint's IP address and sign in to the endpoint if required.

From **Servers > Calendaring Service**, enter the following:

Field	Poly configuration	Matching Infinity configuration	Additional info
Enable Calendaring Service	Select this option.		
Email	The email address of the room resource configured in Exchange or G Suite that is associated with this endpoint.	This must be the same as the Room resource email configured on Pexip Infinity for this endpoint.	
Domain	Leave blank.		This is the Exchange domain, and is not required for One-Touch Join.
User Name	The username the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring Username configured on Pexip Infinity for this endpoint.	
Password	The password the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring password configured on Pexip Infinity for this endpoint.	
Microsoft Exchange Server	• If you have a single Conferencing Node in this location, enter the IP Address or FQDN of the node (in the format **192.168.0.0** or **host.example.com**). • If you have multiple Conferencing Nodes in this location, you should use DNS round robin; therefore this will be the FQDN of the DNS record for this location (in the format **host.example.com**). In both cases, the location is the Pexip Infinity location associated with the Endpoint Group to which this endpoint belongs.		
Meeting Reminder Time in Minutes	Optional		Can still be used in conjunction with One-Touch Join.
Play Reminder Tone When Not in a Call	Optional		Can still be used in conjunction with One-Touch Join.

Field	Poly configuration	Matching Infinity configuration	Additional info
Show Information for Meetings set to Private	Optional		**Enable private meetings** must be enabled on the One-Touch Join Profile associated with this endpoint in order for this setting to apply. If **Enable private meetings** has been disabled on the One-Touch Join Profile, this setting will have no effect.

To confirm that the Poly Studio / Poly G7500 endpoint has registered successfully with the calendaring service:

1. On the endpoint, go to **Servers > Calendaring Service**.
2. Confirm that the **Registration Status** is showing as ***Registered***.

Configuring Poly Debut series

To configure the Poly Debut for One-Touch Join, from **Server Settings > Calendar**, enter the following:

Field	Poly configuration	Matching Infinity configuration	Additional info
Enable Calendar	Select ***Enable***.		
Microsoft Exchange Server	• If you have a single Conferencing Node in this location, enter the IP Address or FQDN of the node (in the format **192.168.0.0** or **host.example.com**). • If you have multiple Conferencing Nodes in this location, you should use DNS round robin; therefore this will be the FQDN of the DNS record for this location (in the format **host.example.com**). In both cases, the location is the Pexip Infinity location associated with the Endpoint Group to which this endpoint belongs.		
Domain	Leave blank.		This is the Exchange domain, and is not required for One-Touch Join.
User Name	The username the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring Username configured on Pexip Infinity for this endpoint.	
Password	The password the endpoint will use when connecting and authenticating to the calendaring service on the Conferencing Node, to obtain meeting information.	This must be the same as the Poly Calendaring password configured on Pexip Infinity for this endpoint.	

To confirm that the Poly Debut endpoint has registered successfully with the calendaring service:

1. On the endpoint, go to the **Device Status** page.
2. In the **Calendar** row of the table, check that the **Status** is showing as ***Registered***.

Configuring Webex Edge endpoints for OTJ

In order for Cisco Webex Edge endpoints to obtain One-Touch Join meeting information, the Conferencing Node connects to the Webex Cloud, which then uses the endpoint's API to push the meeting information to the endpoint.

Prerequisites

Webex endpoints must be:

- registered to Webex Edge for devices
- running CE 9.14 software or later

Creating a Webex Integration

In this step, you create a webex integration that will allow

One-Touch Join meeting types and transforms

This topic details the meeting types, transform patterns and variables that are supported when Adding One-Touch Join meeting processing rules.

You must configure One-Touch Join with information about all the different types of meeting invitations you expect to encounter in your deployment, and rules for how the information in each of these invitations should be used to derive the alias that the endpoint will dial to join the meeting.

You can select from the currently supported meeting types (which you can edit if necessary), or create your own regex or custom rules if you wish to enable One-Touch Join for other meeting types or conferencing providers not already supported. There are also some non-configurable fallback settings that are used when no other rules match.

You must also ensure that your deployment has appropriate Call Routing Rules to enable the One-Touch Join endpoint to dial the meeting aliases that are derived for each meeting type.

Fallback alias matching

If One-Touch Join cannot find a valid meeting alias because none of the meeting processing rules match, or because there are no meeting processing rules configured or enabled, as a fallback it will **always** search the body and the location of the invitation for one of the following patterns to use as the alias to dial:

- sip:<uri>
- sips:<uri>
- h323:<address>

Supported meeting types

The table below lists the currently supported configurable **Meeting types**. For each type, the **Default transform pattern** shows how, when default processing is enabled, One-Touch Join uses the information it finds in the meeting invitation to derive the alias that the endpoint will dial to join the meeting. The table also lists the **Valid variables** that can be used when creating a custom transform pattern for this meeting type.

Meeting type	Usage and notes	Default transform pattern	Valid variables
Pexip Infinity	For meetings scheduled using Pexip's VMR Scheduling for Exchange feature, and which use the default Joining instructions template. These meetings typically include a join link in the format **pexip://<meeting_id>@<domain>**. If your VMR Scheduling for Exchange deployment does not use the default template, or uses an alias in a different format, you should select a **Meeting type** of Domain or Custom instead.	{{meeting_id}}@{{domain}}	• meeting_id • domain
Pexip Service	For meetings held in Pexip Service VMRs.	{{meeting_id}}@pexip.me	• meeting_id • domain

Meeting type	Usage and notes	Default transform pattern	Valid variables
Microsoft Teams Meeting Properties	(Not currently supported for G Suite integrations) For meetings hosted in Microsoft Teams. This rule should be sufficient if all your Teams meetings are internal; otherwise we recommend that you also add any relevant ***Microsoft Teams Meeting Body for ...*** rules. You must provide the Domain that will be used when deriving the alias — this should be the domain from which the meeting invitation was sent.	{{meeting_id}}@{{domain}}	• meeting_id • domain
Microsoft Teams Meeting Body for Poly	If you expect users in your deployment to receive invitations to Microsoft Teams meetings sent from domains other than your own, where the meeting organizer is using a Poly — Teams integration.	{{tenant_id}}.{{meeting_id}}@t.plcm.vc	• meeting_id • domain • tenant_id
Microsoft Teams Meeting Body for BlueJeans	If you expect users in your deployment to receive invitations to Microsoft Teams meetings sent from domains other than your own, where the meeting organizer is using a BlueJeans — Teams integration.	{{tenant_id}}.{{meeting_id}}@teams.bjn.vc	• meeting_id • domain • tenant_id
Microsoft Teams Meeting Body for Pexip Service	If you expect users in your deployment to receive invitations to Microsoft Teams meetings sent from domains other than your own, where the meeting organizer is using a Pexip Service — Teams integration.	{{meeting_id}}@{{domain}}	• meeting_id • domain
Microsoft Teams Meeting Body for Pexip Infinity	If you expect users in your deployment to receive invitations to Microsoft Teams meetings sent from domains other than your own, where the meeting organizer is using a Pexip Infinity — Teams integration.	{{prefix}}{{meeting_id}}@{{domain}}	• meeting_id • domain • prefix
Google Meet	(Not currently supported for Exchange (on-premises or O365) integrations) For meetings scheduled using Google Meet.	{{meeting_id}}@{{domain}}	• meeting_id • domain
Skype for Business	For Skype for Business meetings. The **domain** used is the domain of the organizer's email address. You must also ensure you have a Call Routing Rule configured that includes the following settings (replacing `example\.com` in the example below with the domain of the organizer's email address): • **Destination alias regex match**: `__sfb__([a-z0-9]+)\.([a-z\.\-]+)@(example\.com)` • **Regex replace string**: `sip:\2@\3;gruu;opaque=app:conf:focus:id:\1` • **Call target**: ***Lync / Skype for Business clients, or meetings via a Virtual Reception***	__sfb__{{focus_id}}.{{user}}@{{domain}}	• focus_id • domain

Meeting type	Usage and notes	Default transform pattern	Valid variables
Skype for Business Meeting Body for Poly	For Skype for Business meetings, where the meeting organizer is using a SfB — Poly integration. The domain is hard-coded to **v.plcm.vc**.	{{tenant_id}}.{{meeting_id}}@v.plcm.vc	• meeting_id • domain • tenant_id
Webex	For Webex meetings.	{{meeting_id}}@{{domain}}	• meeting_id • domain
Zoom	For Zoom meetings. In all cases the domain is hard coded to **zoomcrc.com**.	{{meeting_id}}@zoomcrc.com	• meeting_id • domain
BlueJeans	For BlueJeans meetings. In all cases the domain is hard coded to **bjn.vc**.	{{meeting_id}}@bjn.vc	• meeting_id • domain
GoToMeeting	For GoToMeeting meetings.	{{meeting_id}}@{{domain}}	• meeting_id • domain
Domain	If you expect users in your deployment to receive invitations for meetings that do not fall into any of the above categories, you can use this rule to enable meetings where the alias is from a known domain. We recommend that ***Domain*** rules are given a lower priority than any of the other rules. You must provide the Domain that will be searched for in order to match this rule. This rule will search the body and the location for a match. The search will result in a match even if the URI includes one or more subdomains of the domain being searched for. The domain can also include subdomains. When there is a match, the full URI will be used as the meeting alias. For example, if the **domain** is **sales.example.com**, that will match **alice@sales.example.com** and **alice@us.sales.example.com** but not **alice@example.com**.	{{meeting_id}}@{{domain}}	• meeting_id • domain
Regex	See Regex meeting type		
Custom	See Custom meeting type		

Regex meeting type

A **Meeting type** of ***Regex*** enables you to use a regular expression to search for a particular **Match string** in the body and location of the invitation. You can then either:

- select **Default processing enabled** to use the matched string as the alias that the endpoint will dial to join the meeting, or
- disable **Default processing enabled** to use a regex **Replace string** to transform the matched string into the alias to dial.

For more information on using regular expressions with Pexip Infinity, see Regular expression reference.

Examples

Matching without a transform

This example searches the invitation for any alias in the format of **<name>.vmr@example.com**, and uses that as the alias to dial:

Meeting type	Regex
Default processing enabled	Yes
Match string	[\w+].vmr@example.com

In this example, if the meeting body contains the following text:

```
From a video system (SIP/H.323): alice.vmr@example.com
```

then the alias that will be dialed to join the meeting will be **alice.vmr@example.com**

Transforming a URL into an alias

This example searches the invitation for a URL in the format **https::///<domain>/meet/<name>**and transforms that into an alias in the format **<name>@<domain>**:

Meeting type	Regex
Default processing enabled	No
Match string	https:\/\/([^\/]+)/meet\/(\d+)
Replace string	\2@\1

In this example, if the meeting body contains the following text:

```
From web browser & other ways to join:
https://pexip.me/meet/123456
```

then the alias that will be dialed to join the meeting will be **123456@pexip.me**

Custom meeting type

A **Meeting type** of ***Custom*** enables more advanced processing by allowing you to use a Jinja2 template with access to all **calendar_event** information, which you can then use to generate the alias that the endpoint will dial to join the meeting. For more information on using Jinja2 with Pexip Infinity, see Jinja2 templates and filters.

A custom meeting type can be used to enable meeting types or conferencing providers not listed above, or to provide a workaround if any supported providers change their current implementations.

You can use the following calendar event dictionary items, in conjunction with any other literal values if required (e.g. if the domain is always a known quantity), to create the Jinja script:

Item	Type
subject	string
organizer_full_name	string
organizer_first_name	string
organizer_last_name	string

Item	Type	
organizer_email	string	
start_time end_time	dictionary	Properties: • year • month • day • hour • minute • second
is_private	boolean	
body	string	
location	string	
properties	dictionary	**G Suite** A G Suite **calendar_event** will contain a Google Calendar Event resource. For more information, see https://developers.google.com/calendar/v3/reference/events. **Exchange** An Exchange **calendar_event** may contain any EWS MAPI properties from the following list: • **item_class** (string): for options, see https://docs.microsoft.com/en-gb/office/vba/outlook/Concepts/Forms/item-types-and-message-classes • **sensitivity** (string): for options, see https://docs.microsoft.com/en-us/dotnet/api/microsoft.exchange.webservices.data.sensitivity?view=exchange-ews-api • **is_recurring** (boolean): ***True*** if the meeting is part of a recurring series, otherwise ***False***. • **calendar_item_type** (string): for options, see https://docs.microsoft.com/en-us/exchange/client-developer/web-service-reference/calendaritemtype#text-value • **teams_vtc_conference_id**: available for Teams meetings only. • **online_meeting_conf_link**: available for Skype for Business meetings only. • **uc_capabilities**: available for WebEx meetings only.

Examples

The following examples show basic jinja templates that can be used in the Custom template field.

Searching by partial alias

This first example searches the **calendar_event.body** (i.e. the text in the body of the meeting invitation) for an alias that includes **.vmr@example.com**. It then uses the full alias as the meeting alias to dial:

```
{% set matches = pex_regex_search("([\w.-]+\.vmr@example\.com)", calendar_event.body) %}
{% if matches %}
 {{matches[0]}}
{% endif %}
```

In the above example, if the meeting body contains **alice.vmr@example.com**, this will be used as the alias for the meeting.

Searching by top-level domain

This next example searches the **calendar_event.body** (i.e. the text in the body of the meeting invitation) for an alias that includes a domain ending in **.com**. It then uses the full alias as the meeting alias to dial:

```
{% set groups = pex_regex_search("([a-z0-9.-]+)@([a-z0-9.-]+.com)", calendar_event.body) %}
{% if groups %}
 {{ groups[0] }}@{{ groups[1] }}
{% endif %}
```

In the above example, if the meeting body contains **alice.vmr@example.com**, this will be used as the alias for the meeting.

Searching the location for a partial alias

This example searches the **calendar_event.location** (i.e. the text in the location field of the meeting invitation) for an alias that includes **.vmr@example.com**. It then uses the full alias as the meeting alias to dial:

```
{% set matches = pex_regex_search("([\w.-]+\.vmr@example\.com)", calendar_event.location) %}
{% if matches %}
 {{matches[0]}}
{% endif %}
```

In the above example, if the meeting location contains **alice.vmr@example.com**, this will be used as the alias for the meeting.

Lifesize Cloud example

This example searches a standard Lifesize Cloud meeting invitation and converts the URL into a meeting alias:

```
{% set matches = pex_regex_search("https://call.lifesizecloud.com/([0-9.-]+)", calendar_event.body) %}
{% if matches %}
 {{matches[0]}}@lifesizecloud.com
{% endif %}
```

In the above example, if the meeting body contains **https://call.lifesizecloud.com/123456**, the alias that will be used to join the meeting will be **123456@lifesizecloud.com**.

Deploying a dedicated One-Touch Join platform

In most cases, One-Touch Join will be implemented as a feature within a wider Pexip Infinity deployment, and run on Conferencing Nodes alongside other Pexip Infinity services. However, you can also set up separate OTJ locations within your deployment that contain Conferencing Nodes used solely for One-Touch Join. A third option appropriate in some situations is to implement a separate Pexip Infinity deployment purely for One-Touch Join, for example if you are a Pexip Service customer wishing to use One-Touch Join, or you are a large enterprise wishing to separate the resources used for your One-Touch Join deployment.

If you are implementing a dedicated One-Touch Join deployment alongside but separate from a Pexip Infinity deployment, they do not need to be running the same software version, as there is no interaction between the two deployments. This means that existing Pexip Infinity environments can implement a dedicated One-Touch Join deployment without having to upgrade their existing software.

Minimum hardware requirements

A dedicated One-Touch Join deployment consists of one Management Node and at least one Conferencing Node. Further Conferencing Nodes can be deployed for redundancy.

For dedicated One-Touch Join-only deployments, the resource requirements are minimal, therefore you may use the **minimum** server specifications outlined below. However, if you expect to broaden your deployment to implement some of the wider Pexip Infinity features in the future, you will need to increase the specifications of your hardware, as detailed in the server design guidelines.

- Management Node:
  - 4 cores
  - 4 GB RAM
  - AVX or later processor
  - 100 GB SSD storage
  - The Pexip Infinity VMs are delivered as VM images (.ova etc.) to be run directly on the hypervisor. No OS should be installed.
- Conferencing Nodes:
  - 4 cores
  - 4 GB RAM
  - AVX or later processor
  - 50 GB SSD storage per Conferencing Node, 500 GB total per server (to allow for snapshots etc.)
  - The Pexip Infinity VMs are delivered as VM images (.ova etc.) to be run directly on the hypervisor. No OS should be installed.

Minimum Pexip Infinity platform configuration

You must ensure the following components of the Pexip Infinity platform are configured and working appropriately:

- DNS servers
- NTP servers
- Locations (note that you do not need to configure any overflow locations, as this concept is not used by One-Touch Join).
- Licenses: you will need an **OTJ** license for each endpoint that will use the One-Touch Join feature.
- Custom CA certificates: only required if you are using One-Touch Join with Exchange on-premises, and your Exchange server does not use a globally trusted certificate.

Call Routing Rules are not required on the dedicated One-Touch Join deployment, because these deployments do not handle any calls. However, you must ensure that your call control system is configured so that calls being placed by the endpoints to each of the supported meeting types can be routed appropriately.

One-Touch Join configuration

The process of configuring One-Touch Join in a dedicated environment is the same as when configuring it as part of a wider Pexip Infinity deployment, namely:

1. Configuring your calendar/email service:
   - Configure G Suite for One-Touch Join, including Adding a One-Touch Join G Suite integration on Pexip Infinity, or
   - Configure Exchange on-premises for One-Touch Join, including Adding a One-Touch Join Exchange integration on Pexip Infinity, or
   - Configure Office 365 for One-Touch Join, including Adding a One-Touch Join Exchange integration on Pexip Infinity
2. Adding a One-Touch Join profile
3. Adding One-Touch Join endpoint groups
4. Adding One-Touch Join endpoints
5. Adding One-Touch Join meeting processing rules

For more information, see Configuring Pexip Infinity for One-Touch Join

Scheduling and joining meetings using One-Touch Join

When Pexip Infinity's One-Touch Join feature has been enabled for meeting rooms in your environment, you don't need to do anything special in order to use it — everything will happen automatically:

1. You or the meeting organizer create a meeting invitation in Outlook, Google calendar, or via the Teams client in your usual way. This includes any invitations that are created by using add-in buttons, for example for Pexip scheduled meetings or for Webex. Just ensure you have added the meeting room to the invitation as a room resource.
2. Each endpoint in each meeting room will display a list of upcoming meetings for that room. When a meeting is due to start, the endpoint in the meeting room will show a **Join** button.
3. When you are ready to join the meeting, just press the **Join** button. The endpoint will dial in to the meeting.

Viewing One-Touch Join status

You can check the status of your One-Touch Join deployment by viewing a list of all currently scheduled One-Touch Join meetings, and by viewing a list of all endpoints enabled for One-Touch Join.

Viewing One-Touch Join meetings

To view a list of all currently scheduled meetings that use Pexip Infinity's One-Touch Join feature in your deployment, go to **Status > One-touch Join Meetings**.

This page lists all One-Touch Join meetings with a start time from one day in the past up to the number of days in the future specified by the associated One-Touch Join profile's No. of upcoming days setting.

This information is updated each time the OTJ process runs. The OTJ process obtains meeting information by reading the room resources' calendars, and then processing the information based on the currently configured OTJ profile settings and meeting processing rules. This means that any changes to room resources' calendars (e.g. adding meetings, canceling meetings, or changing the meeting information), or any changes to the way the meeting information is processed (e.g. changes to the OTJ profile settings, or to meeting processing rules) will be reflected in the status after the OTJ process next runs. This could be between 30 seconds and many minutes, depending on the number of OTJ rooms in your deployment.

To view full details about a meeting, click on the meeting subject. The following information is available for each meeting:

Field	Description
Meeting subject	The text that appears in the subject line of the meeting invitation. This field will show the organizer's name instead of the meeting subject if either: • Replace subject for private meetings has been enabled and the meeting was flagged as private, or • Replace empty subject has been enabled and there was no subject.
Organizer name *	The name of the person who created the meeting invitation.
Organizer email	The email address of the person who created the meeting invitation.
Start time	The scheduled start time of the meeting. This does not include the Start buffer.
End time	The scheduled end time of the meeting. This does not include the End buffer.
Endpoint name	The name of the endpoint, as configured in Pexip Infinity.
OTJ Profile name	The name of the OTJ profile used when processing this meeting.
Meeting alias	The alias that the endpoint will use to dial in to the meeting. This will be blank if either: • Process alias for private meetings has been disabled and the meeting was flagged as private, or • Enable non-video meetings has been enabled, but OTJ was not able to obtain a valid alias for the meeting.
Meeting room email *	The email address of the room resource in whose calendar the meeting has been scheduled.
Matched meeting processing rule *	The name of the meeting processing rule that was matched and used to process this meeting. This will be blank if the meeting information did not match any meeting processing rules, and Enable non-video meetings has been enabled.

* Only displayed when you have selected an individual OTJ meeting to view.

Viewing One-Touch Join endpoints

To view a list of all endpoints in your deployment that are actively available for use by Pexip Infinity's One-Touch Join feature, go to **Status > One-touch Join Endpoints**.

This page lists all Cisco endpoints that One-Touch Join has successfully contacted, and all Poly endpoints that have successfully contacted One-Touch Join. For both, it lists the date and time of the most recent contact.

To view full details about an endpoint, click on the endpoint name. The following information is available for each endpoint:

Field	Description
Endpoint name	The name of the endpoint, as configured in Pexip Infinity.
Endpoint type	The type of "click to join" feature supported by this endpoint.
Endpoint address	The IP address of the endpoint.
Meeting room email	The email address of the room resource associated with this endpoint.
OTJ Profile name	The name of the OTJ profile used when processing this meeting.
Conferencing Node *	The IP address and name of the Conferencing Node that last had contact with the endpoint.
Last contact time	The date and time that contact was last made with the endpoint.
Number of meetings *	The number of currently scheduled One-Touch Join meetings that will use this endpoint.

* Only displayed when you have selected an individual OTJ meeting to view.

Configuring G Suite for domain user authorization

This topic describes an alternative method to configuring G Suite for One-Touch Join in environments where the recommended method of using a service account for authorization is not desirable. This alternative method uses a domain user for authorization (referred to as the "authorization user"), which authenticates to G Suite using 3-legged OAuth.

The process involves the following steps, described in more detail in the sections that follow:

1. Setting up OAuth authentication for One-Touch Join.
2. Creating a room resource for each physical room that will have a One-Touch Join endpoint in it.
3. Configuring the room resource with the necessary permissions and settings to support One-Touch Join.
4. Updating the quota for the number of user requests per 100 seconds.
5. For larger deployments, Requesting an increase to API limits.
6. Adding a One-Touch Join G Suite integration on Pexip Infinity.

If you have already set up a One-Touch Join G Suite integration and simply wish to add an existing room to it, you need only configure the room resource in G Suite and then add the endpoint to the G Suite integration in Pexip Infinity.

Prerequisites

You must have already created a user account specifically to be used as the G Suite authorization user. This user account does not need to have any special privileges; as part of the configuration described below you will grant this user access to all the One-Touch Join room resource calendars.

Enabling authorization using OAuth

In this step you create a project to use for One-Touch Join. You then enable the Calendar API for this project, and create the OAuth credentials to be used when One-Touch Join accesses the API as the authorization user.

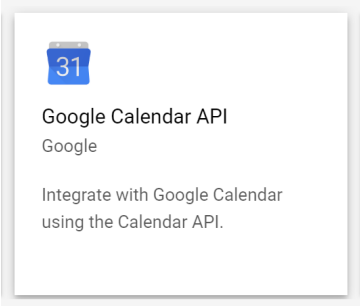
1. Creating a new project:
   a. Go to https://console.developers.google.com (logged in as a G Suite administrator).
   b. From the top left of the page, select the down arrow:

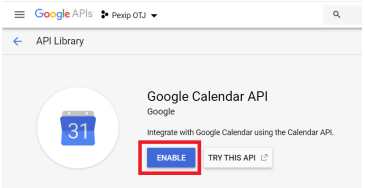
   c. Select **New Project**.
   d. Enter a **Project name** (e.g. One-Touch Join) and select **Create**.
2. Enabling the Calendar API for the project:
   a. Go to https://console.developers.google.com
   b. From the top left of the page, select the down arrow, select your newly-created project, and select **Open**. Your new project should now be showing at the top left of the page:

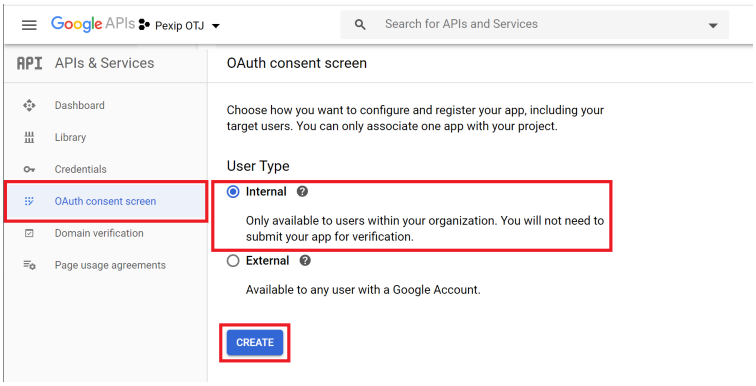
   c. From the navigation menu on the left of the screen, select **APIs & Services > Library**, then scroll down and select the **Google Calendar API** tile:



d. Select **Enable**:

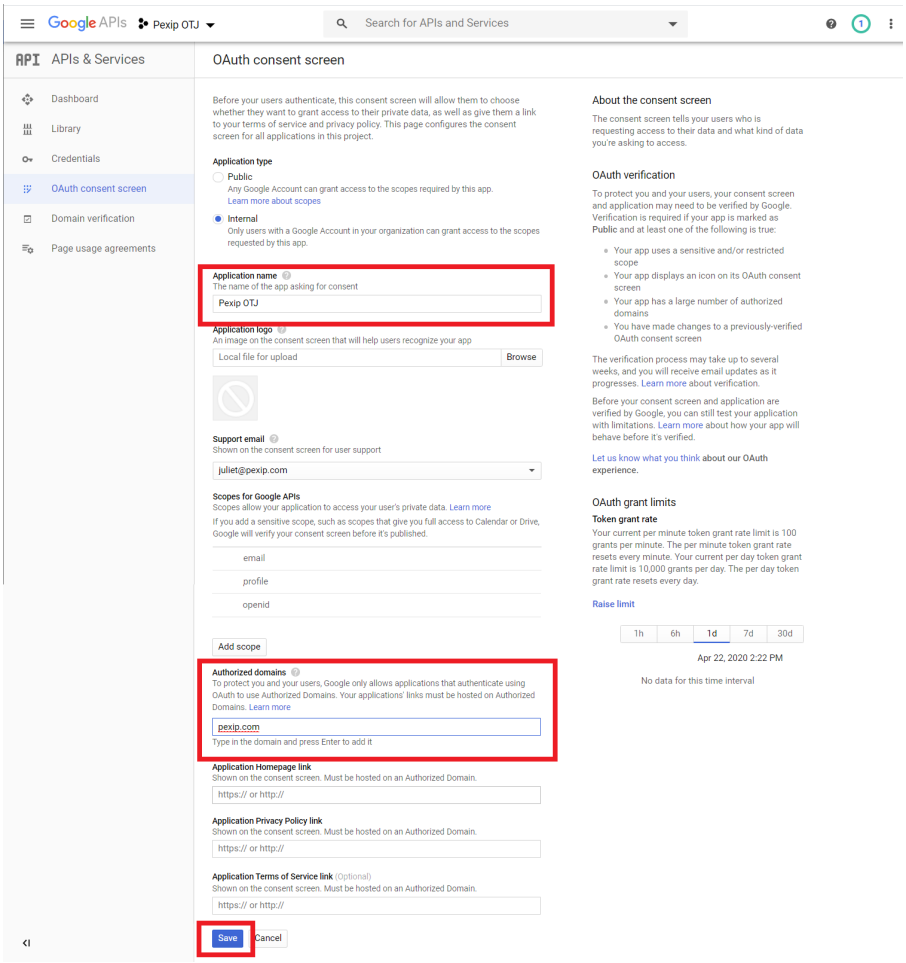


3. Creating an OAuth consent screen:
   a. From https://console.developers.google.com, from the left-hand panel select **OAuth consent screen**. Select a **User Type** of ***Internal*** and then select **Create**:

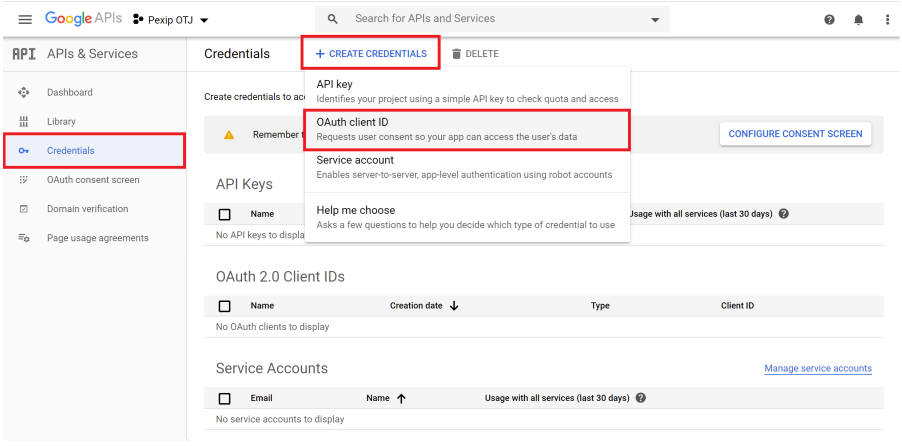


   b. From the OAuth consent screen page:
      - under **Application name**, enter a name for your OTJ application
      - under **Authorized domains**, enter the domain of the Management Node.

Select **Save**:



4. Creating the OAuth credentials:
   a. From https://console.developers.google.com, from the left-hand panel select **Credentials** and then select **Create Credentials > OAuth client ID**:

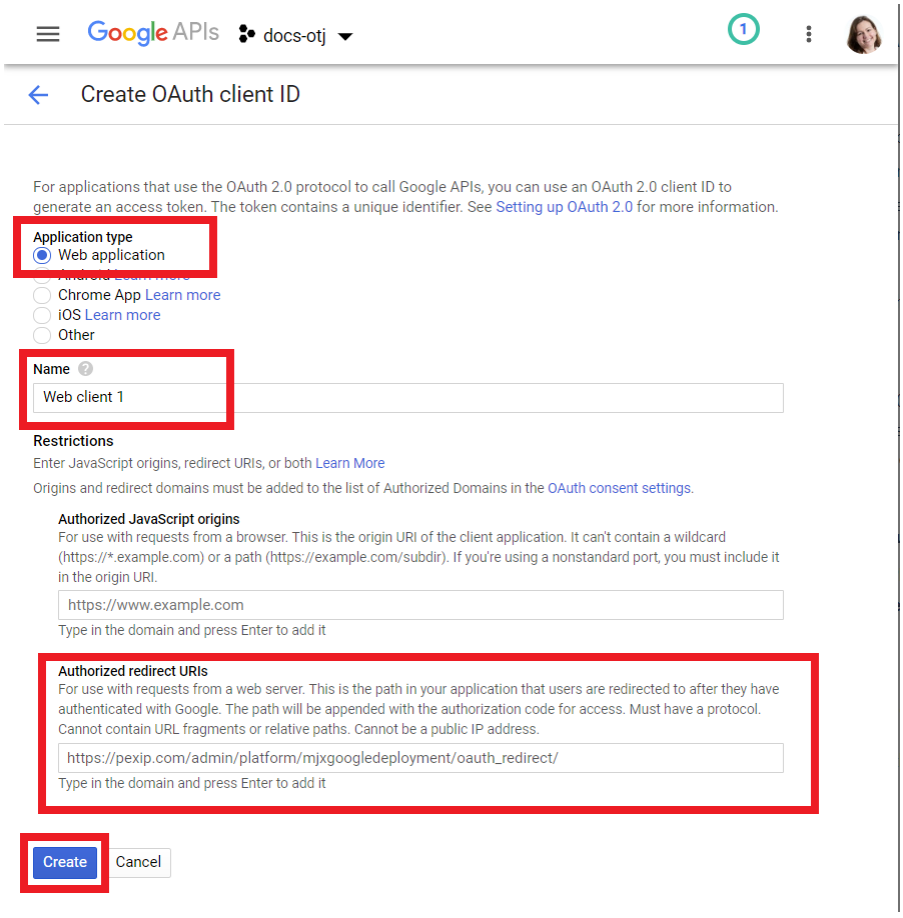


   b. From the **Create OAuth client ID** page:, select an **Application type** of ***Web application***.

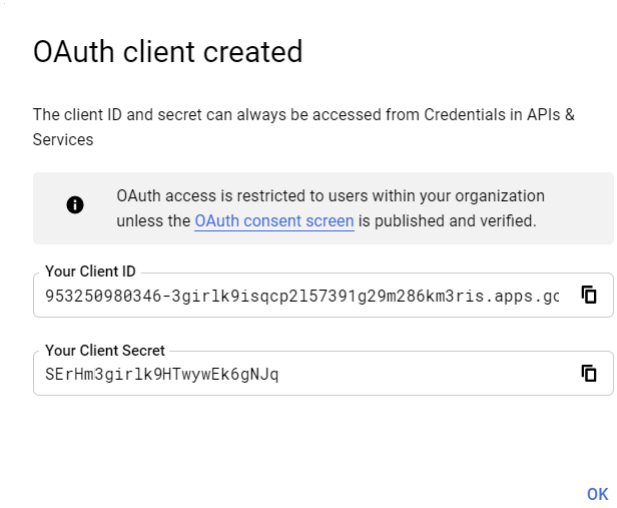
- Enter a **Name** for the application
- under Authorized redirect URIs, enter **https://<Management Node FQDN>/admin/platform/mjxgoogledeployment/oauth_redirect/**

  This must use the Management Node's FQDN; it cannot use its IP address. You must therefore ensure you have appropriate internal DNS records set up for the Management Node.

Select **Create**:



c. The following **OAuth client created** screen will appear. Take note of the **Your Client ID** and **Your Client secret**; you will need these when Adding a One-Touch Join G Suite integration on Pexip Infinity on the Management Node:

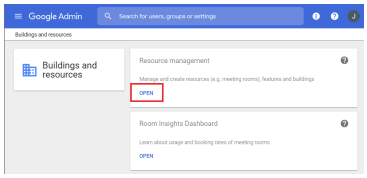


Creating a room resource

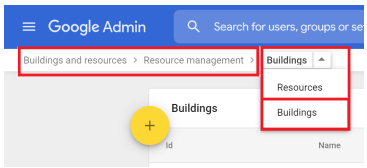
(Required only if your room resources do not already exist - otherwise you can skip this step.)

In this step, you create a room resource in G Suite for each physical room that is to be used for One-Touch Join. G Suite will automatically assign an email address to the room.

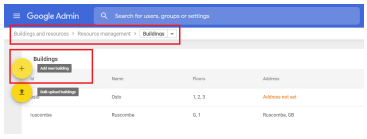
1. If a **building** for the room resource does not already exist, create one as follows:
   a. Go to https://admin.google.com (logged in as a G Suite administrator).
   b. Select the **Buildings and resources** tile, and then from the **Resource management** section select **Open**:



   From the drop-down along the top left of the screen, select **Buildings**:

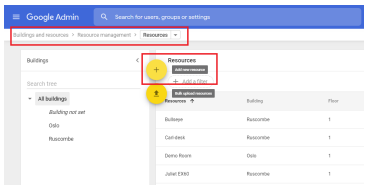


   c. Select **+** to **Add new building**:

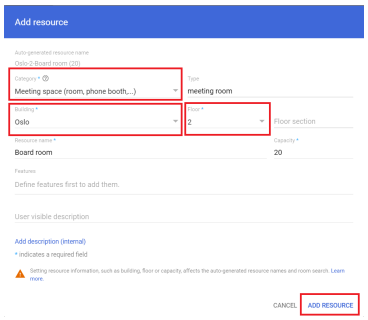


   d. Enter a **Name** and the list of **Floors**, and select **Add Building**.

2. Create the room resource:
   a. Go back to the **Resources** page and Select **+** to **Add new resource**:



   b. For the **Category**, select ***Meeting space (room, phone booth,...)***.
   c. Select the **Building** and **Floor** in which the room is located, enter a **Name** and the room's **Capacity**, then select **Add Resource**:



The resource will be created and added to the list. You can click on the new resource to view information about it, such as the email address it was automatically assigned.

For more information on setting up buildings and other resources in G Suite, including how to add buildings and resource in bulk and using CSV imports, see https://support.google.com/a/answer/1033925.

Configuring the room resource

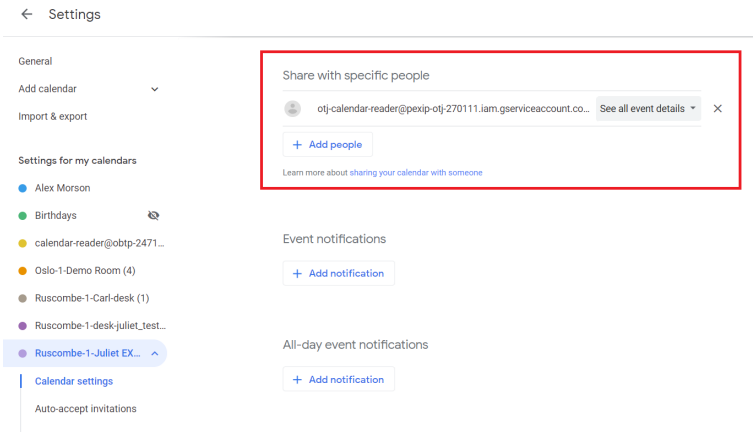
In these steps, you allow the authorization user to access each calendar of each room resource that you want to use for One-Touch Join, and set the calendar to auto-accept invitations. We also recommend that you make the calendar available to all users in your domain in such a way that allows them to book meetings using the resource, without being able to view the details of any other meetings in the resource's calendar.

Sharing individual calendars with the authorization user

Note that the Google calendar API limits the number of calendars that can be shared within a 24 hour period to 750 (for more information, see https://support.google.com/a/answer/2905486?hl=en). This means that if you have more than 750 room resources that you wish to use for One-Touch Join, they will need to be set up over a period of days.

1. Go to https://calendar.google.com (logged in as a G Suite administrator so that you have permission to share the calendars).
2. From the left-hand panel, select the **+** next to **Other calendars** and then select **Browse resources**.
3. Expand the sections if necessary, and tick the boxes of all the room resources whose calendars you want to share with the authorization user.

   This will add the room resources to the **Settings for other calendars** section in the left-hand panel.
4. For each of the rooms:
   a. From the **Settings for my calendars** section, select the room resource and then select **Share with specific people**.
   b. Select **Add people**.
   c. In the **Share with specific people** dialog, enter the email address of the One-Touch Join authorization user. Ensure the **Permissions** are set to ***See all event details***.
   d. Select **Send**:



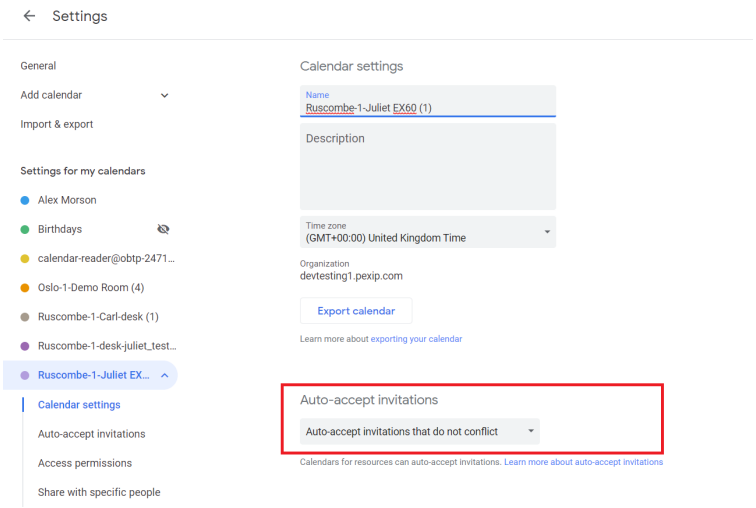
For more information on sharing room and resource calendars in G Suite, see https://support.google.com/a/answer/1034381.

Auto-accepting invitations

By default, when creating room resources in G Suite, calendar processing is set to **Auto-accept invitations that do not conflict**. You must ensure you keep this setting for all room resources, so that the room will automatically accept meeting requests if it is available, and automatically decline an invitation if it is already booked.

To check this setting:

1. Go to https://calendar.google.com (logged in as a G Suite administrator so that you have permission to share the calendars).
2. From the left-hand panel, select the room resource and select **Settings and sharing**.
3. In the **Auto-accept invitations** section, ensure that ***Auto-accept invitations that do not conflict*** is selected:

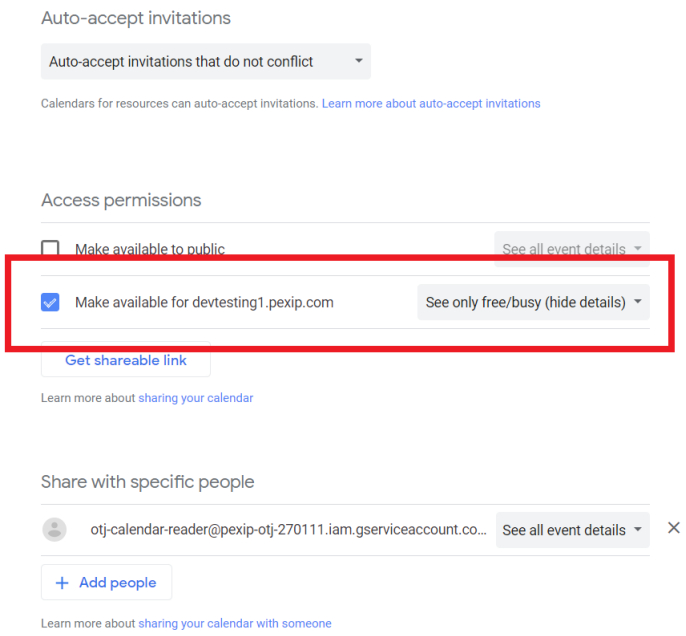


Allowing users to book resources

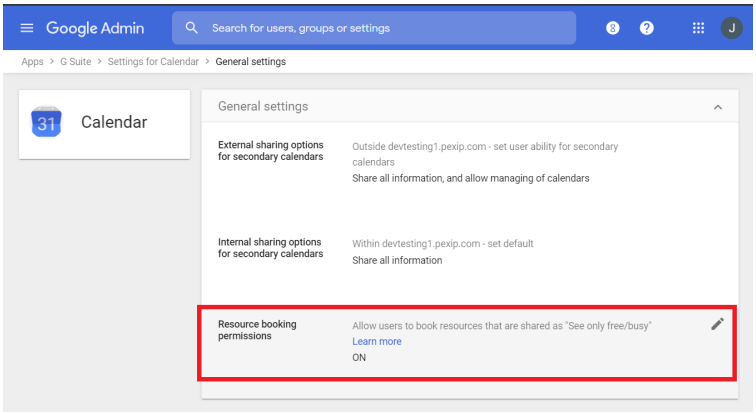
We recommend that you configure your G Suite calendar settings to allow end users to book a room resource without seeing details of the room's other bookings. To do this, you configure the room resource's calendar so that all users in your domain have permission to see its free/busy status, without being able to see the invitation details. You then on a global basis permit users to book resources to which they have free/busy access.

To do this:

1. Go to https://calendar.google.com (logged in as a G Suite administrator so that you have permission to share the calendars).
2. From the left-hand panel, select the room resource and select **Settings and sharing**.
3. In the **Access permissions** section, select **Make available for <your domain>**, and ensure that ***See only free/busy (hide details)*** is selected:



4. Go to admin.google.com (logged in as a G Suite administrator).
5. From the left-hand menu, select **Apps > G Suite > Calendar**.
6. Scroll down to **General Settings** and select **Resource Booking Permissions**.
7. Ensure that **Allow users to book resources that are shared as See only free/busy** is set to ***ON***:



Updating the per-user request quota

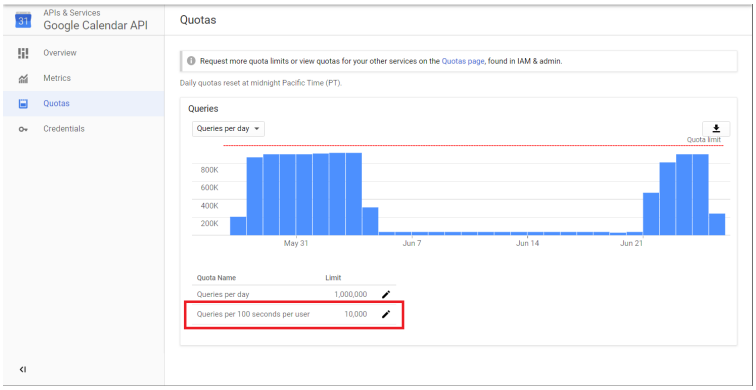
In this step you increase the limit on the number of queries per 100 seconds per user to the Google Calendar API.

The default number of queries per 100 seconds per user is 500. We recommend that you increase this to 10,000, as follows:

1. Go to https://console.developers.google.com (logged in as a G Suite administrator).
2. From the top left of the page, select the project you created for One-Touch Join:

3. From the navigation menu at the top left of the page, select **IAM & Admin > Quotas**.
4. From the Quotas page, select **Edit Quotas** and then select **Google Calendar API - Queries per 100 seconds per user**.

   You will be taken to the **Google Calendar API > Quotas** page.
5. Change **Queries per 100 seconds per user** to **10,000**:



> You may also need to request an increase to the number of **Queries per day** for larger deployments - for more information, see Requesting an increase to API limits.

Requesting an increase to API limits

This optional step applies to larger deployments only (more than around 170 room resources), and should be performed if you wish to reduce the amount of time taken for endpoints to be updated with additions or changes to their corresponding room resource calendar.

The maximum frequency with which an endpoint will be updated with meeting information is every 30 seconds. For deployments with more than around 170 endpoints, this frequency will decrease in line with the number of endpoints (up to around 20 minutes for deployments with around 6,000 endpoints). This is due to a limit on the number of Calendar API requests permitted by Google in a 24-hour period — for more information, see https://developers.google.com/calendar/pricing.

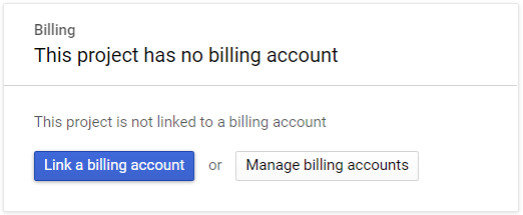
To reduce the time taken to update endpoints in these larger deployments, you can request an increase to the number of Calendar API requests One-Touch Join can make.

> When your request has been implemented by Google, you must then increase the Maximum G Suite API requests on Pexip Infinity in order to take advantage of the increase.

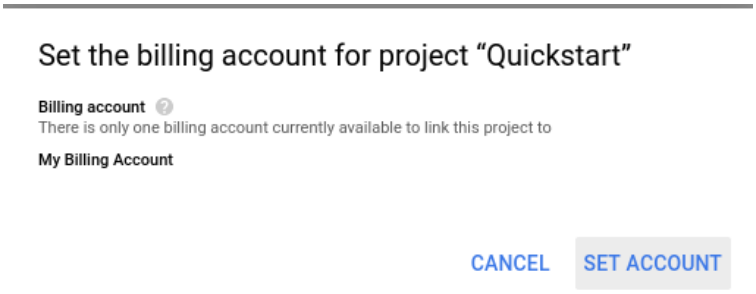
To request an increase to the API limits:

1. If you do not already have one, create a Cloud Billing Account (note that this is different from a G Suite billing account).

   Full instructions are available via https://cloud.google.com/billing/docs/how-to/manage-billing-account#create_a_new_billing_account.
2. Link the Cloud Billing Account to the project you created when Creating a service account:
   a. Go to https://console.developers.google.com (logged in as a G Suite administrator).
   b. Ensure that the project shown in the top left corner is the one you created for One-Touch Join when Creating a service account.
   c. Select the burger menu from the top left of the page and select **Billing**. When the following message appears, select **Link a billing account**:

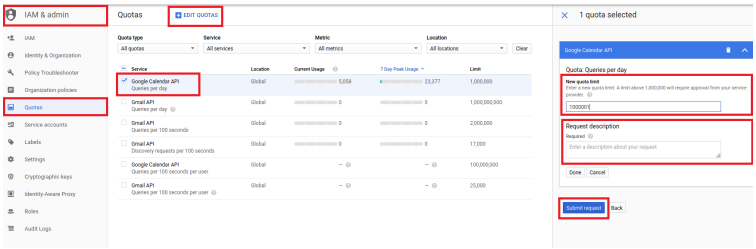


d. Select the account to link to:



3. Request an increase to your quota:
   a. From the navigation menu at the top left of the page, select **IAM & admin > Quotas**.
   b. From the Quotas page, select **Edit Quotas** and then select **Google Calendar API**.

      In the panel that appears on the right, enter the **New quota limit** that you wish to request, and in the **Request description** field, enter the reason for requesting the increase:



   c. Select **Submit request**.

Quota increase requests typically take two business days to process.

Adding a One-Touch Join G Suite integration on Pexip Infinity

In this step you configure Pexip Infinity with details of the G Suite deployment configured above. You must then log in to G Suite as the authorization user and grant the One-Touch Join app access to the room resource calendars.

Configuring the G Suite integration

From the Pexip Infinity Administrator interface, go to **One-Touch Join > OTJ G Suite Integrations**.

Option	Description
Name	The name of this One-Touch JoinG Suite integration.
Description	An optional description of this One-Touch JoinG Suite integration.

Option	Description
Account email	If you are authorizing using a service account, enter the email address of the service account that One-Touch Join will use to log in to G Suite. If you are authorizing using a G Suite domain user, enter the email address of the user.
Enable user authorization	If you are authorizing using a service account — the recommended method — this should be left blank. Select this option only if you will be authorizing using a G Suite domain user.
Private key	(Available when authorizing using a service account, i.e. user consent authorization has not been enabled) The private key used by One-Touch Join to authenticate the service account when logging in to G Suite. For instructions on how to obtain this, see Generating a key file. This must include all the text in the file between (and including) `-----BEGIN PRIVATE KEY-----` and `-----END PRIVATE KEY-----`
Client ID	(Available when user consent authorization has been enabled) The client ID of the application you created in the Google API Console, for use by OTJ.
Client secret	(Available when user consent authorization has been enabled) The client secret of the application you created in the Google API Console, for use by OTJ.
Redirect URI	(Available when user consent authorization has been enabled) The redirect URI you configured in the Google API Console. It must be in the format: **https://<Management Node FQDN>/admin/platform/mjxgoogledeployment/oauth_redirect/** ℹ This must use the Management Node's FQDN; it cannot use its IP address. You must therefore ensure you have appropriate internal DNS records set up for the Management Node.
Advanced options	
Maximum G Suite API requests	The maximum number of API requests that can be made by One-Touch Join to your G Suite Domain in a 24-hour period. We recommend you set this value to 90% of your total permitted requests. Google's default is 1,000,000 so by default this is set to 900,000 on Pexip Infinity. If you increase the number of API requests, you should also increase this setting to 90% of that number. For more information, see Frequency and limitations on calendar requests.
Google OAuth 2.0 endpoint	The URI of the Google OAuth 2.0 endpoint.
Google authorization server	The URI of the Google authorization server.

When you have completed the above fields, select **Save**. You will be returned to the main OTJ G Suite Integration page. You must now authorize calendar API access to the G Suite Integration using the account details you have just created, using the following steps.

Authorizing calendar access

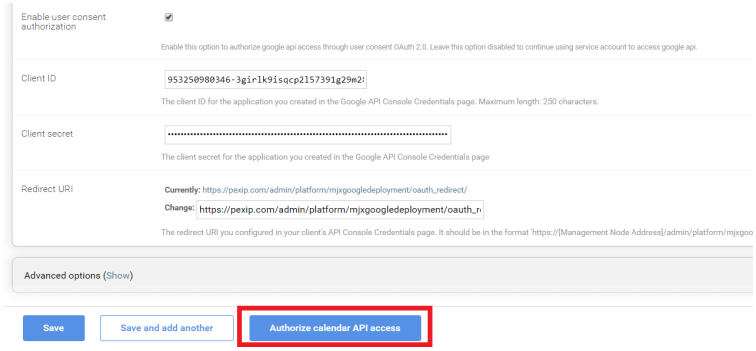
If you have enabled OAuth for the first time, after saving the configuration of the One-Touch Join G Suite integration you must sign in to G Suite as the authorization user.

You may also need to re-sign in to the authorization user account if:

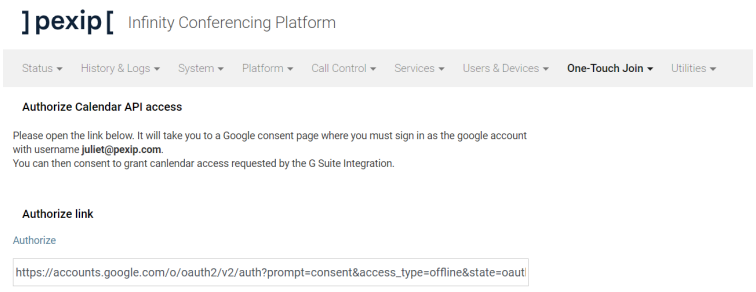
- you disable and then subsequently re-enable OAuth
- you update any of the following configuration for the One-Touch Join G Suite integration:
  - Account email
  - Client ID
  - Client secret
  - Google OAuth 2.0 endpoint
  - Google authorization server
- the refresh token has expired (for more information about when this might happen, see https://developers.google.com/identity/protocols/oauth2#expiration).

To sign in to G Suite as the authorization user:

1. Ensure you have signed out of all Google accounts on your device.
2. From the Management Node, go to **One-touch Join > OTJ G Suite Integrations** and select the G Suite integration you have just created. At the bottom of the **Change OTJ G Suite Integration** page, select **Authorize calendar API access**:

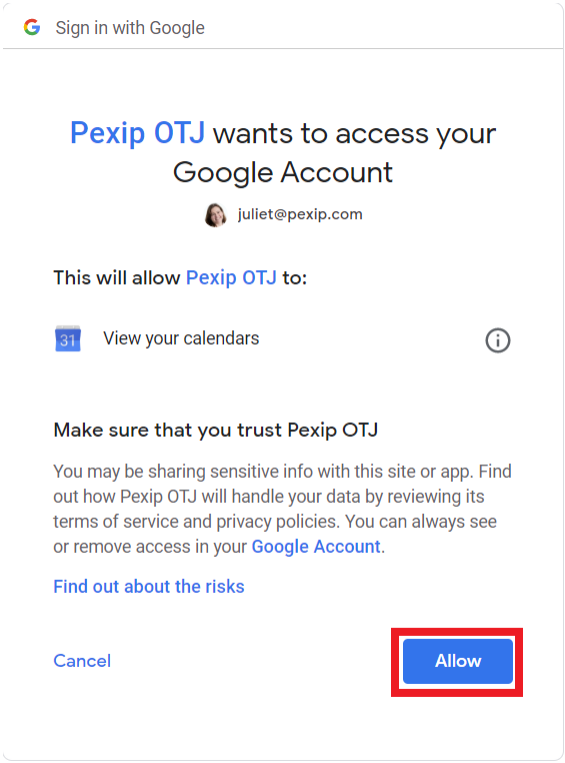


3. You will be taken to the **Authorize Calendar API access** page. Select **Authorize**:



4. Enter the email address of the authorization user (which you previously entered as the **Account email**) and sign in.

5. At the consent screen, **Allow** the Pexip OTJ app to **View your calendars**:



You should be returned to the **Authorize Calendar API access** page and see the message **Successfully authorized**.

Next steps

You must now configure the remainder of the One-Touch Join components on Pexip Infinity, as described in Configuring Pexip Infinity for One-Touch Join.

Troubleshooting One-Touch Join

This section provides guidance on troubleshooting issues with Pexip Infinity's One-Touch Join feature, including issues specific to Cisco or Poly endpoints.

For guidance on the troubleshooting of general issues, see Troubleshooting the Pexip Infinity platform.

Symptom	Possible cause	Resolution
One-Touch Join issues		
A meeting has been scheduled and is showing on the room endpoint, but there is no **Join** button.	One-Touch Join has not been able to obtain a meeting room alias from the invitation because it does not match a meeting processing rule and does not contain a URI or address prefixed with **sip:**, **sips:** or **h323:**.	Review the meeting processing rules.
	The meeting processing rule that you expected to match is associated with a different OTJ profile than the endpoint. For example, the endpoint has an Exchange email address and is associated with an Exchange integration, but the rule that the meeting matches is associated with a G Suite integration, or vice versa.	1. Check that the **OTJ Endpoint** has been associated with an **OTJ Endpoint Group**. 2. Check that the **OTJ Endpoint Group** is associated with the same **OTJ Profile** as the **Meeting Processing Rule** that you expected to match.
	The meeting is not a video meeting.	If you do not want non-video meetings to appear on the room endpoint, you can disable the Enable non-video meetings option.
A meeting has been scheduled and is showing on the room endpoint, but either there is no **Join** button, or the **Join** button appeared and then disappeared.	The endpoint is being managed by Webex Cloud Calendar or TMS XE, and these systems are overriding the meeting information from One-Touch Join.	Ensure that any endpoints used for One-Touch Join are not also registered to the calendaring service on other systems such as the cloud-based Webex Hybrid Calendar Service, or Cisco TMS XE.
Meetings are being deleted from an endpoint that is managed by TMS, without TMS XE.	There is a known bug (CSCvv93408) with TMS version 15.9 and later whereby TMS will erroneously replace meetings that have been pushed to the endpoint using the endpoint's API.	Ensure that the following configuration for the endpoint has been made in TMS: • Disable **Allow booking** for the endpoint • Change **Meeting Type** to ***Reservation***. If the problem persists, we recommend removing the endpoint from TMS until this bug is fixed by Cisco.
A meeting has been scheduled and is showing on the room endpoint, but there is no **Join** button. The support log shows the message: `Could not find an alias for this meeting which had no body. This could be a meeting room configuration issue.`	One-Touch Join has not been able to obtain a meeting room alias from the invitation because the meeting information supplied in the body ("description") of the invitation has been stripped by Exchange prior to One-Touch Join processing the meeting.	Change the calendar processing rules for the room to ensure that the meeting body is not deleted. For instructions on how to do this, see either Configuring calendar processing (for Exchange on-premises) or Configuring calendar processing (for O365).

Symptom	Possible cause	Resolution
An external Microsoft Teams meeting has been scheduled but there is no **Join** button.	Your Microsoft Exchange environment uses a security application (such as Office 365 ATP, or Mimecast) to re-write URLs, meaning that One-Touch Join has not been able to obtain the join URL. For more information, see Allowing forwarding of external invitations (for Exchange on-premises) or Allowing forwarding of external invitations (for O365).	Ensure that the security application's URL re-write rules include an exception for any URL starting with the domain **https:\\teams.microsoft.com**
There is a delay between a meeting invitation being sent and it appearing on the room endpoint.	A short delay is expected due to internal processing, and the actual time taken will depend on the number of endpoints in your One-Touch Join deployment, and the number of daily API requests you are allowed to make to your calendar service. Limits are also imposed so that Conferencing Nodes do not become overloaded with One-Touch Join requests. For more information, see Frequency and limitations on calendar requests.	For larger G Suite integrations you can ask for an increase to the number of calendar API requests you can make in a 24-hour period, thus allowing you to update endpoints more frequently. For more information, see Requesting an increase to API limits. You could also consider Deploying a dedicated One-Touch Join platform.
On the status page and logs, the **Alias** field is blank.	Process alias for private meetings has been disabled and the meeting was flagged as private.	Review whether these settings are appropriate for your deployment.
	Enable non-video meetings has been enabled, but OTJ was not able to obtain a valid alias for the meeting. This may be because Exchange is using default calendar processing, which removes the header and body of the invitation, and replaces the subject with the organizer's name.	Ensure that Exchange calendar processing properties are changed from the default, as per the instructions in Configuring calendar processing on room resource mailboxes.
On the status page and logs, the **Subject** field is showing the organizer's name.	Replace subject for private meetings has been enabled and the meeting was flagged as private, or	Review whether these settings are appropriate for your deployment.
	Replace empty subject has been enabled and there was no subject. This may be because Exchange is using default calendar processing, which removes the header and body of the invitation, and replaces the subject with the organizer's name.	Ensure that Exchange calendar processing properties are changed from the default, as per the instructions in Configuring calendar processing on room resource mailboxes.
An endpoint has been deleted from the Pexip Infinity configuration but its details are still appearing on the **OTJ Endpoints** status page.	The status page is refreshed once an hour.	Wait up to one hour for the endpoint's details to be removed.
A meeting that has been canceled is still appearing on the **OTJ Meetings** status page.	The status page is refreshed once an hour.	Wait up to one hour for the meeting's details to be removed.

Symptom	Possible cause	Resolution

When configuring Exchange you are getting the following errors or warnings:
`ErrorCode="InvalidUser"`
`ErrorMessage="Invalid user"`
	The service account being used for One-Touch Join does not exist, or does not have a valid license.	• Ensure that the service account has been added correctly, with the correct username and password/authentication information.
		• Ensure that the service account has an appropriate Exchange license, such as Office 365 Enterprise E1, Office 365 Business Basic (formerly Essentials) or one of the Exchange Online plans.

Cisco endpoint issues		
One-Touch Join cannot contact an endpoint via its API. The following appears in the alarms and logs: `Non-200 status code returned when trying to upload OBTP bookings to endpoint` and `StatusCode="307"`	One-Touch Join is configured to communicate with the endpoint via HTTP and the endpoint redirects to HTTPS.	Configure One-Touch Join to use HTTPS to communicate with the endpoint.
A Cisco SX series endpoint running TC software may display the "Meeting will automatically connect" message if there is no URI in the meeting invitation.	This is a known issue with the Cisco endpoint when running this software.	
Poly endpoint issues		
Meetings are not appearing on the Poly endpoint.	The configuration for the endpoint on Pexip Infinity or on the endpoint itself is incorrect.	Ensure that the configuration for endpoint on Pexip Infinity and on the endpoint itself is correct, in particular that the username and password configured on both match. Ensure that the endpoint is showing as registered to the calendaring service.
	The Poly endpoint is registered to the calendaring service but One-Touch Join hasn't found any meetings.	View the Meeting status page to see if any meetings have been found for this endpoint. Check for any Google Gatherer/Exchange Gatherer alarms, which would indicate issues with reading specific calendars.
	The Poly endpoint has lost connection with the OTJ calendaring service and has become unregistered, meaning it is no longer receiving updated meeting information. To check if there is still contact with the endpoint: • If Raise alarms is enabled for this endpoint, an OTJ Poly Endpoint Error alarm will appear on the Pexip Infinity Administrator interface if it has been more than 10 minutes since there was contact with the endpoint. • If this option is not enabled, view the Endpoint status and check the last contact time. If this is more than 10 minutes ago the endpoint may have lost connection.	On the Poly endpoint, disable and re-enable the calendaring service.