

Videokonferenz

Ein Meeting kann automatisch einen Videolink bekommen. Ein **RC-Admin** wählt den Standardanbieter unter **Einstellungen → Externe Dienste → Videokonferenz**; jedes neue Meeting erhält dann einen Link, der Link zieht beim Verschieben des Meetings mit, und nach dem Meeting lässt sich die Teilnahme zurücklesen.

Überblick

Anbieter	Link	Zieht mit dem Meeting um	Teilnahme	Was du brauchst
Jitsi	Raumname aus dem Meeting	nicht nötig (ein Raum hat keinen Termin)	nein	nichts - das öffentliche <code>meet.jit.si</code> oder dein eigener Server
Zoom	geplantes Meeting	ja	ja (bezahlter Tarif)	eine Server-to-Server-OAuth-App
Webex	geplantes Meeting	ja	ja	eine Service App und ihr Refresh-Token
Google Meet	ein Meet-Space (kein Kalendereintrag)	nicht nötig (ein Space hat keinen Termin)	ja (nur Namen)	ein OAuth-Client und sein Refresh-Token
Microsoft Teams	Online-Meeting eines Organisators	ja	ja	eine Entra-App, Administratorzustimmung und eine Application Access Policy

Ein Link entsteht beim Anlegen eines Meetings und sonst beim ersten Öffnen des Meetings, seiner Erinnerung oder seines Kalenderexports (ein Meeting, das vor der Einrichtung eines Anbieters angelegt wurde, bekommt ihn dann). Ein Meeting mit **niemandem außer dem Organisator** ist ein Blocker im Kalender dieser Person und bekommt keinen Link, bis jemand anderes eingeladen wird.

Das Erzeugen eines Links verhindert **nie**, dass ein Meeting angelegt wird: Lehnt der Anbieter ab, wird das Meeting ohne Link gespeichert und der Grund angezeigt. Dasselbe gilt für das Verschieben und das Zurücklesen der Teilnahme.

Die Zugangsdaten stehen in den Einstellungen der Instanz (wie die Mail-Einstellungen) und sind nur für RC-Admins lesbar. Alle Aufrufe an die Anbieter gehen vom Resourcecloud-Server aus, über den Proxy der Instanz, falls einer eingerichtet ist.

Teststand

Keiner der vier API-Anbieter (Zoom, Webex, Google Meet, Teams) wurde bisher gegen ein echtes Konto ausgeführt. Der Code ist vollständig (anlegen, verschieben, Teilnahme) und folgt der dokumentierten API des jeweiligen Anbieters, aber niemand hat ein echtes Konto angebunden. Nimm die Berechtigungsnamen unten als Ausgangspunkt aus dem Code und der Dokumentation der Anbieter und bestätige sie beim ersten Anbinden. Jitsi baut nur einen Link und ruft nichts auf, da gibt es nichts zu testen.

Anbieter	Link anlegen	Verschieben	Teilnahme	Gegen echtes Konto geprüft
Jitsi	konstruktionsbedingt	entfällt	entfällt	nicht nötig
Zoom	nicht getestet	nicht getestet	nicht getestet	offen
Webex	nicht getestet	nicht getestet	nicht getestet	offen
Google Meet	nicht getestet	entfällt	nicht getestet	offen

Microsoft Teams	nicht getestet	nicht getestet	nicht getestet	offen
-----------------	----------------	----------------	----------------	--------------

Wenn du einen davon anbindest, ist eine kurze Notiz, was funktioniert hat (und was du ändern musstest), sehr willkommen - sie kommt in diese Tabelle.

Jitsi

Wähle **Jitsi** und trage, wenn du einen eigenen Server betreibst, dessen Hostnamen ein (nur den Host, z. B. `meet.example.com` - ein eingefügtes `https://` und ein Pfad werden entfernt). Standard ist das öffentliche `meet.jit.si`. Der Raumname wird aus dem Meeting gebildet; wer den Link hat, kann beitreten - nimm für Sensibles einen eigenen Server mit Anmeldung.

Zoom

- 1 Wähle im **Zoom App Marketplace Develop → Build App → Server-to-Server OAuth App**.
- 2 Kopiere **Account ID**, **Client ID** und **Client Secret** in die drei Zoom-Felder.
- 3 Füge unter **Scopes** die Berechtigungen zum Anlegen und Ändern von Meetings hinzu (in Zooms aktueller Benennung `meeting:write:meeting:admin` und `meeting:update:meeting:admin`; ältere Apps nennen es `meeting:write:admin`) und für die Teilnahme `report:read:admin`. Zoom hat diese Namen mehrfach geändert - nimm die, die seine Scope-Liste für „Meeting anlegen / ändern“ und „Meeting-Berichte ansehen“ anbietet.
- 4 Aktiviere die App.

Meetings werden für den eigenen Benutzer des Kontos angelegt (`/users/me/meetings`), in der unter Meetings eingestellten Zeitzone. Der Teilnahmebericht ist eine Zoom-Funktion bezahlter Tarife und erscheint einige Zeit nach dem Ende des Meetings.

Webex

- 1 Lege im **Webex Developer Portal** eine **Service App** mit dem Scope `meeting:schedules_write` (Meetings anlegen und verschieben) und für die Teilnahme `meeting:participants_read` an.
- 2 Bitte den Webex-Org-Admin, die Service App für die Organisation freizugeben.
- 3 Kopiere **Client ID**, **Client Secret** und das **Refresh Token** der freigegebenen App in die Webex-Felder.

Webex kann beim Einlösen des Refresh-Tokens ein neues ausgeben; Resourcecloud speichert das neue automatisch, damit es gültig bleibt. Legt lange niemand ein Meeting an, kann das Token ablaufen - trage dann ein neues ein.

Google Meet

- 1 Aktiviere in der **Google Cloud Console** die **Google Meet REST API** und lege einen **OAuth-Client** an.
- 2 Autorisiere ihn einmal für **Offline**-Zugriff mit dem Scope `https://www.googleapis.com/auth/meetings.space.created` (für die Teilnahme kann zusätzlich der Lese-Scope der Meet-API für Konferenzdatensätze nötig sein) und bewahre das **Refresh-Token** auf.
- 3 Kopiere **Client ID**, **Client Secret** und **Refresh Token** in die Google-Felder. Das Refresh-Token ist langlebig und wird nicht rotiert.

Ein Meet-Space hat weder Beginn noch Ende und legt keinen Kalendereintrag an, deshalb zieht beim Verschieben des Meetings nichts um. Die Meet-API nennt Teilnehmende mit Namen, nicht mit E-Mail-Adresse; der Abgleich der Teilnahme mit den Eingeladenen fällt deshalb meist auf den Namen zurück.

Microsoft Teams

- 1 Registriere in **Microsoft Entra ID** eine Anwendung und erzeuge ein **Client-Geheimnis**. Kopiere **Mandanten-ID**, **Client-ID** und **Geheimnis** in die Teams-Felder.
- 2 Füge unter **API-Berechtigungen** die *Anwendungs*-Berechtigung `OnlineMeetings.ReadWrite.All` (Microsoft Graph) hinzu und erteile die **Administratorzustimmung**. Für die Teilnahme füge ebenso `OnlineMeetingArtifact.Read.All` hinzu.
- 3 Lege eine **Application Access Policy** an, die der App erlaubt, für den Organisator zu handeln (Teams-PowerShell: `New-CsApplicationAccessPolicy` und `Grant-CsApplicationAccessPolicy`). Ohne sie antwortet Graph auch mit Zustimmung mit 403. Sie kann bis zu etwa 30 Minuten brauchen, bis sie wirkt.
- 4 Trage den **Organisator** ein - die Objekt-Id oder den Benutzerprinzipalnamen des Benutzers, für den die Meetings angelegt werden. Alle Meetings der Instanz gehören diesem einen Benutzer.

Ist die **Kalendersynchronisation** (Microsoft 365) eingeschaltet, legt der Outlook-Termin das Teams-Meeting selbst an und zieht mit um; die Einstellungen oben dienen dann nur noch der Teilnahme. Siehe die Kalender-Anleitung.

Wenn etwas schiefgeht

- Die Meldung nennt den Anbieter und bei einem abgelehnten Aufruf den HTTP-Code und den Anfang der Antwort des Anbieters - diese Antwort ist meist der schnellste Weg zur Ursache (*invalid_client*: falsche Id oder falsches Geheimnis; *403*: es fehlen eine Berechtigung, die Zustimmung oder die Teams-Policy; *400 invalid_grant*: das Refresh-Token ist falsch oder abgelaufen).
- Eine Meldung, dass die Zugangsdaten **nicht vollständig eingerichtet** sind, nennt die leeren Felder.
- Das Meeting selbst bleibt immer erhalten. Korrigiere die Einstellung und öffne das Meeting erneut: Ein Meeting ohne Link bekommt dann einen (es braucht mindestens zwei Personen).

- Es wird nie „zu früh“ etwas zurückgelesen: Ein leeres Teilnahme-Ergebnis heißt, dass der Anbieter noch keinen Bericht hat (Zoom und Teams brauchen nach dem Ende Minuten), nicht dass niemand kam.