Scheduled Live Events

Article • 05/08/2023

Overall Flow

You can schedule your Page or profile’s scheduled live event in advance, and then promote it on and off LinkedIn. The flow to create a scheduled live event is outlined below.

> ⓘ **Note**
>
> Members are currently limited to scheduling up to 10 scheduled live events per day (including both your profile and any Pages you manage).

1. If you would like to include an image with a scheduled live event announcement post, upload an announcement image `asset`. (optional)

```https
POST /assets?action=registerUpload
```

2. Create a scheduled `liveVideo` asset. Use the `asset` from step #1 if available.

```https
POST /liveVideos
```

3. Create an annoucement post. Use the `liveVideo` urn from step #2.

```https
POST /ugcPosts
```

4. When you are ready to go live and are nearing your scheduled live event time, register the live event `asset`.

> ⓘ **Note**
>
> The allowed go live time window is between 15 minutes before the scheduled time and 2 hours after the scheduled time.

https

```
POST /liveAssetActions?action=register
```

5. Begin RTMP(s) ingestion.

6. Update the scheduled `liveVideo` (step #2) with the live event `asset` from step #4. This step can only be completed once the `asset` status is AVAILABLE.

https

```
POST /liveVideos
```

7. End the scheduled live event.

https

```
POST https://api.linkedin.com/v2/liveAssetActions?action=end
```

1. Upload an Announcement Image (optional)

Upload an announcement image to increase engagement with your scheduled live event. This step is optional. If no announcement image is provided, a default announcement image will be used with the announcement post.

Supported Input Format: JPEG, PNG

Image Requirements:

- **preferred aspect ratio**: 16:9
- **max pixel limit (w*h)**: 36152320
- **max image size**: 209mb

The first request will register your announcement image asset. As with all Live Events APIs, the owner provided must be the author (person or organization URN) of the live event. Save the announcement image asset urn to be used in later steps.

Use the `uploadUrl` returned from this request to upload the announcement image.

Request

JSON

```
POST /assets?action=registerUpload

curl -X POST 'https://api.linkedin.com/v2/assets?action=registerUpload' -H 'X-
RestLi-Protocol-Version: 2.0.0' -H 'Content-Type: application/json' -H 'Authoriza-
tion: Bearer <redacted>' --data-raw '{
  "registerUploadRequest": {
    "recipes": [
      "urn:li:digitalmediaRecipe:video-liveannouncement-image"
    ],
    "owner": "urn:li:person:iKO_ntS9UQ",
    "serviceRelationships": [
      {
        "relationshipType": "OWNER",
        "identifier": "urn:li:userGeneratedContent"
      }
    ]
  }
}'
```

Response

JSON

```
{
    "value": {
        "uploadMechanism": {
            "com.linkedin.digitalmedia.uploading.MediaUploadHttpRequest": {
                "uploadUrl": "https://api.linkedin.com/mediaUpload/C5F22AQEU-
vl4IOBjE2g/feedshare-uploadedImage/0?ca=vector_feedshare&cn=uploads&m=AQJ2Wjp_Va-
Tk1wAAAXKfCVcFPtCwi6mquOfiHR356trDm59sEEg77qwSpA&app=1953784&sync=0&v=beta&ut=1hrN
_TF9ERQpg1",
                "headers": {
                    "media-type-family": "STILLIMAGE"
                }
            }
        },
        "asset": "urn:li:digitalmediaAsset:C5F22AQEUvl4IOBjE2g",
        "mediaArtifact": "urn:li:digitalmediaMediaArtifact:(urn:li:digitalmediaAs-
set:C5F22AQEUvl4IOBjE2g,urn:li:digitalmediaMediaArtifactClass:feedshare-uploaded-
Image)"
    }
}
```

Uploading the image using the uploadUrl in a curl request

curl

```
curl --upload-file ~/Downloads/liveAnnouncement.jpg -H 'Authorization: Bearer
<redacted>' 'https://api.linkedin.com/mediaUpload/C5F22AQEUvl4IOBjE2g/feedshare-
uploadedImage/0?
```

```
ca=vector_feedshare&cn=uploads&m=AQJ2Wjp_VaTk1wAAAXKfCVcFPtCwi6mquOfiHR356trDm59s-
EEg77qwSpA&app=1953784&sync=0&v=beta&ut=1hrN_TF9ERQpg1'
```

2. Create a Scheduled Live Event Asset

The scheduledAt parameter defines the epoch time in milliseconds of your scheduled live event.

For example, a scheduledAt time of `1588550400000` represents Monday, May 4, 2020 12:00:00 AM GMT.

The `name` parameter must be used to provide a title for your scheduled live event.

> **Note**
>
> The `name` has a max length limit of 75 chars.

If an `announcementImage` is available, include the asset urn from step #1, and provide an image title. If there is no announcement image, remove the announcementImage parameter from the request body.

(Member Profile) Request

JSON

```
POST /liveVideos

curl -i -X POST 'https://api.linkedin.com/v2/liveVideos' -H 'X-RestLi-Protocol-
Version: 2.0.0' -H "Content-Type: application/json" -H "Authorization: Bearer
<redacted>" --data-raw '{
  "author": {
    "member": "urn:li:person:iKO_ntS9UQ"
  },
  "scheduledAt": 1591892529000,
  "announcementImage": {
    "media": "urn:li:digitalmediaAsset:C5F22AQEUvl4IOBjE2g",
    "title": {
      "text": "Title of the image.",
      "inferredLocale": "en_US"
    }
  },
  "name": "Title of the scheduled live event"
}'
```

(Organization) Request

JSON

```
POST /liveVideos

curl -i -X POST 'https://api.linkedin.com/v2/liveVideos' -H 'X-RestLi-Protocol-
Version: 2.0.0' -H "Content-Type: application/json" -H "Authorization: Bearer
<redacted>" --data-raw '{
  "author": {
    "organization": "urn:li:organization:0000"
  },
  "scheduledAt": 1591892529000,
  "announcementImage": {
    "media": "urn:li:digitalmediaAsset:C5F22AQEUvl4IOBjE2g",
    "title": {
      "text": "Title of the image.",
      "inferredLocale": "en_US"
    }
  },
  "name": "Title of the scheduled live event"
}'
```

Response

Save the location header id returned within the response. Appending this id to `urn:li:liveVideo` will be used when creating the announcement post.

JSON

```
HTTP/2 201
x-li-responseorigin: RGW
location: /liveVideos/6676519335839784960
x-restli-id: 6676519335839784960
x-restli-protocol-version: 2.0.0
```

3. Create an Announcement Post

Creating the announcement post that will publish to your LinkedIn feed is similar to the existing live event ugc post, with a few key differences. Specifically, the `media`, `shareMediaCategory`, and `distribution` parameters will need to be updated for scheduled posts.

The `media` parameter must include the liveVideo urn (`urn:li:liveVideo:6676519335839784960`) saved from step #2.

The `shareMediaCategory` is now defined as `URN_REFERENCE`.

The `distribution` parameters are defined in the sample request below.

Request

JSON

```
POST /ugcPosts

curl -X POST 'https://api.linkedin.com/v2/ugcPosts' -H 'X-RestLi-Protocol-Version:
2.0.0' -H "Content-Type: application/json" -H "Authorization: Bearer <redacted>" -
-data-raw '{
  "author": "urn:li:person:iKO_ntS9UQ",
  "lifecycleState": "PUBLISHED",
  "specificContent": {
    "com.linkedin.ugc.ShareContent": {
      "media": [
        {
          "media": "urn:li:liveVideo:6676519335839784960",
          "status": "READY"
        }
      ],
      "shareCommentary": {
        "attributes": [],
        "text": "This is the text description displayed with the post."
      },
      "shareMediaCategory": "URN_REFERENCE"
    }
  },
  "visibility": {
    "com.linkedin.ugc.MemberNetworkVisibility": "PUBLIC"
  },
  "distribution": {
    "feedDistribution": "MAIN_FEED",
    "externalDistributionChannels": [],
    "distributedViaFollowFeed": true
  }
}'
```

Response

Appending the id returned to `https://www.linkedin.com/feed/update/urn:li:ugcPost:6676519700211556352` provides a reference for users to view the scheduled post on LinkedIn.

JSON

```
{
    "id": "urn:li:ugcPost:6676519700211556352"
}
```

4. Register the Live Event

When you are ready to go live to your scheduled live event, begin the process to register your asset and begin RTMP ingestion. This step remains unchanged from the existing live event asset registration. For reference, use the request below.

> **Note**
>
> Newly registered Live Events will be discarded if ingestion has not started within 4 hours. After ingestion has started, a timeout will occur if the ingest URL has not received any data within 90 seconds.

Request

JSON

```
POST https://api.linkedin.com/v2/liveAssetActions?action=register

curl -X POST 'https://api.linkedin.com/v2/liveAssetActions?action=register' \
-H 'X-RestLi-Protocol-Version: 2.0.0' \
-H 'Content-Type: application/json' \
-H 'Authorization: Bearer <redacted>' \
--data-raw '{
    "registerLiveEventRequest": {
        "owner": "urn:li:person:iKO_ntS9UQ",
        "recipes": ["urn:li:digitalmediaRecipe:feedshare-live-video"],
        "region": "WEST_US"
    }
}'
```

5. Begin RTMP(s) Ingestion

RTMP(s) ingestion may only begin 15 minutes prior to your `scheduledAt`, or 2 hours after your `scheduledAt` time.

6. Update the Scheduled Live Event

Using the registered live event asset from step #4, we can update the scheduled live event (step #2) and effectively transition our announcement post to a scheduled live event. You will need the live event id saved from step #2 to update the `liveVideoAsset` parameter. Ensure the live event asset status is AVAILABLE before proceeding with this step.

A successful response returns a 204 that your asset has been updated.

Request

JSON

```
POST /liveVideos

curl -X POST 'https://api.linkedin.com/v2/liveVideos/6676519335839784960' -H 'X-RestLi-Protocol-Version: 2.0.0' -H "Content-Type: application/json" -H "Authorization: Bearer <redacted>" --data-raw '{
  "patch": {
    "$set": {
      "liveVideoAsset": {
        "media": "urn:li:digitalmediaAsset:C5524AQFazrOYw4_q2Q"
      }
    }
  }
}'
```

7. End Live Event Asset

For any live Event asset created, once ingestion has completed always be sure to conclude your event by sending the action to end live event. This step remains unchanged. For reference, use the following request:

Request

JSON

```
POST https://api.linkedin.com/v2/liveAssetActions?action=end

curl -X POST 'https://api.linkedin.com/v2/liveAssetActions?action=end' -H "Authorization: Bearer <redacted>" --data-raw '{
  {
      "asset": "urn:li:digitalmediaAsset:C5624AQEUbk4_xZgHJQ"
  }
}'
```

Retrieving Information about Created Scheduled Live Events

To retrieve information about all scheduled live events by the author, send a GET request to the live events API. You must include the encoded author urn along with your request. Possible values for the author urn include: person urn or organization urn.

Request

JSON

```
curl -X GET 'https://api.linkedin.com/v2/liveVideos?
q=author&author=urn%3Ali%3Aperson%3AiKO_ntS9UQ' -H 'X-RestLi-Protocol-Version:
2.0.0' -H "Content-Type: application/json" -H "Authorization: Bearer <redacted>"
```

Response

JSON

```
{
    "paging": {
        "start": 0,
        "count": 15,
        "links": [],
        "total": 4
    },
    "elements": [
        {
            "scheduledAt": 1591892529000,
            "name": "Title Goes West",
            "liveVideoAsset": {
                "media": "urn:li:digitalmediaAsset:C5524AQFazrOYw4_q2Q"
            },
            "id": 6676519335839784960
        },
        {
            "scheduledAt": 1591744521000,
            "id": 6676245139628879872
        },
        {
            "scheduledAt": 1591731035000,
            "id": 6676183242384732160
        },
        {
            "scheduledAt": 1592004112000,
            "id": 6676174265252966400
        }
    ]
}
```

To retrieve information about a single scheduled live event, you may send a GET request to the liveVideos service along with the live events id.

Request

JSON

```
curl -X GET 'https://api.linkedin.com/v2/liveVideos/6676519335839784960' -H 'X-RestLi-Protocol-Version: 2.0.0' -H "Content-Type: application/json" -H "Authorization: Bearer <redacted>"
```

Response

JSON

```
{
    "scheduledAt": 1591892529000,
    "name": "Title Goes West",
    "liveVideoAsset": {
        "media": "urn:li:digitalmediaAsset:C5524AQFazrOYw4_q2Q"
    },
    "id": 6676519335839784960
}
```