DAILY PREBUILT API TOOLS & TIPS

Manage call permissions with Daily's "knocking" feature

An overview of Daily's knocking feature for managing access to private video rooms.



Jess Mitchell
May 18, 2023 / 9 min read



When setting up video calls, we at Daily know different calls can require different levels of privacy. In some sessions, you want as many people as possible to join. In other, only a curated list of people should be allowed to enter. It's like having people over to your home; you might be

having an open house where anyone can drop in, or you might be having a dinner party where there are only six place settings with name cards at the table. In other words: it varies.

Daily video calls manage privacy settings in a similar way. There is a “public” room setting where anyone can come in, and a “private” setting where certain permissions are required.

In this post, we’ll compare variations in private room settings. Specifically, we’ll focus on the knocking feature that Daily offers, which allows video call participants to request to enter a private call.

Today’s agenda

As an overview, today we will:

- Learn how to update your Daily room privacy settings in the dashboard or with Daily’s REST API. These settings apply to Daily Prebuilt and custom UIs built with Daily’s client SDK for JavaScript.
- Learn how to turn on knocking for private rooms, as well as how the knocking setting affects different types of call participants.
- See how the knocking feature looks in Daily Prebuilt, where the video call UI is already built for you.
- Review which methods and events to use if you’re building your own custom UI with Daily’s Client SDK for JavaScript. In a future tutorial, we’ll show you how to build a custom knocking feature yourself.

So, let’s get started!

Public vs. private Daily rooms

As mentioned, Daily offers public and private rooms, which can be set using the privacy configuration.

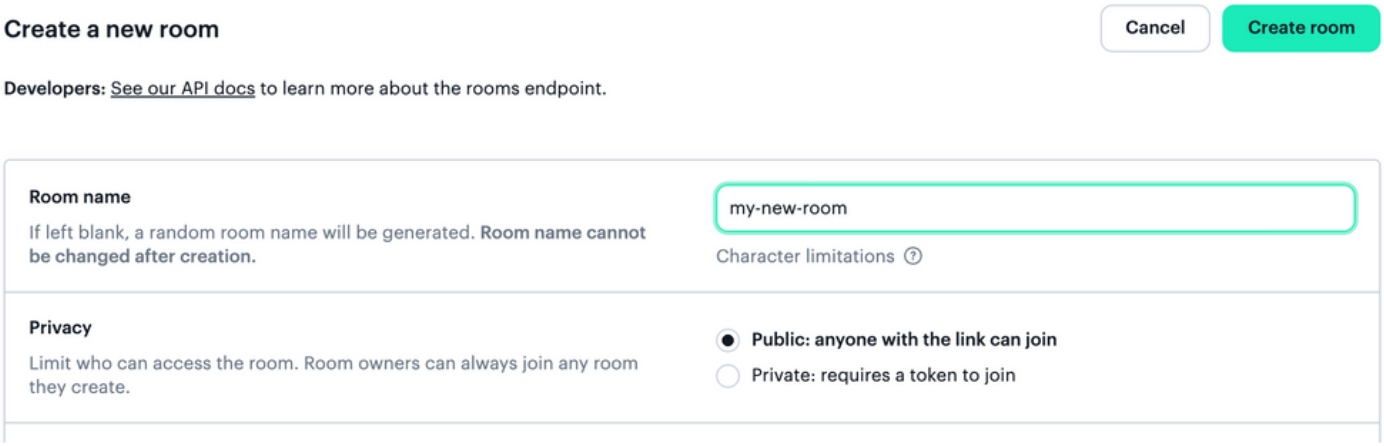
If a room is set to `"public"` – the default privacy setting – anyone can join a call in that room if they have the meeting link. It’s like leaving the door to your house unlocked – technically, anyone

could walk in.

Rooms can also be set to `"private"`, which is like locking your door. Once a room is private, only certain people will be able to join. (More on that below.)

You can change a room's privacy setting using the Daily dashboard or REST API.

If you're using the Daily dashboard to create or update a room, there's a Privacy setting in the room form.



If you're using the REST API to create or update a room, you can set the `privacy` option to either `"public"` or `"private"` like so:

```
curl -H "Content-Type: application/json" \
     -H "Authorization: Bearer $DAILY_API_KEY" \
     -XPOST -d \
     '{"name": "my-new-room",
       "privacy": "private"' \
     https://api.daily.co/v1/rooms/
```

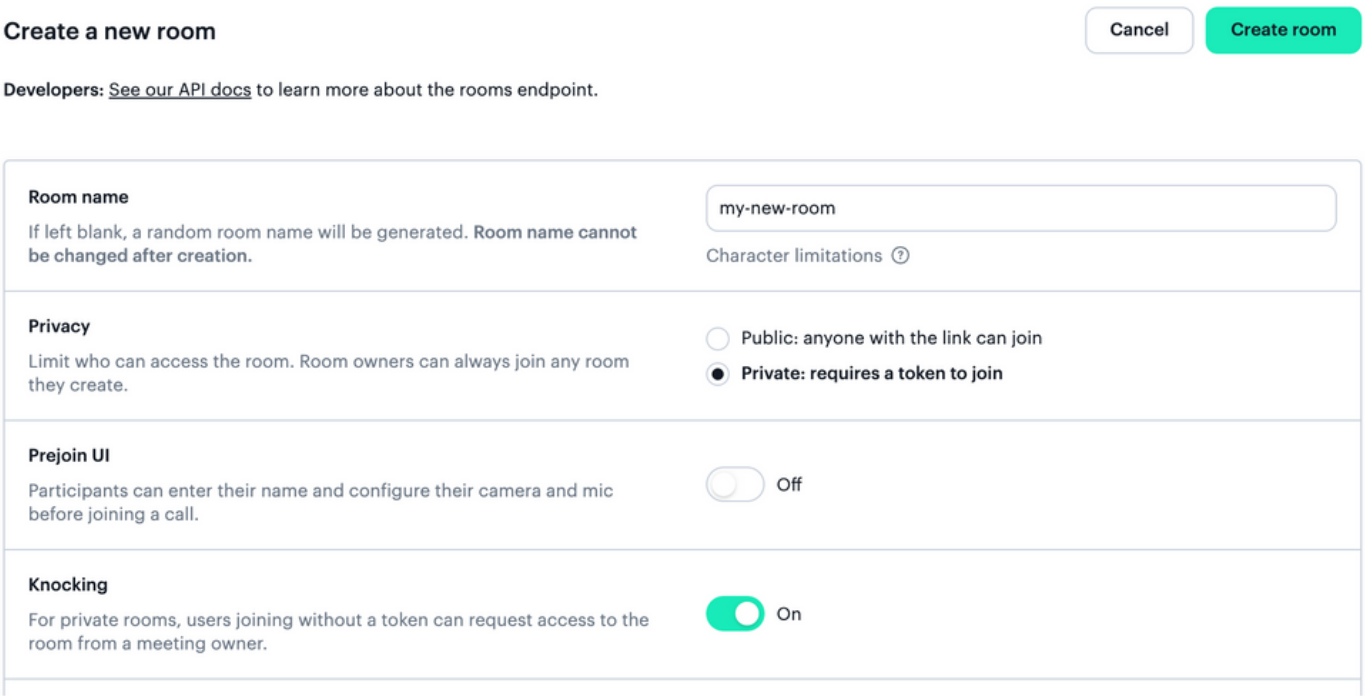
Note: Your Daily API key is available in the dashboard's developer page.

Daily's knocking feature and how to enable it

Knocking (a.k.a., the `enable_knocking` room setting) is a feature available for all private rooms. By default, it is not enabled, so it must be explicitly turned on. It has no impact on public rooms since anyone can join a public room.

Knocking can be enabled via the dashboard or REST API.

In the dashboard, look for the Knocking option when creating or updating a room.



With the REST API, use the Create a room or Modify a room endpoints with the `enable_knocking` option set to `true`:

```
curl -H "Content-Type: application/json" \
     -H "Authorization: Bearer $DAILY_API_KEY" \
     -XPOST -d \
     '{"name": "DAILY_ROOM_NAME",
       "privacy": "private",
       "properties" : {
          "enable_knocking":true}}' \
     https://api.daily.co/v1/rooms/
```

Note: `enable_knocking` is passed under `properties` and `privacy` is not, which is a detail this author has forgotten too many times to count.

Next, let's look at the differences between call participants with tokens and those without.

Types of call participants in private rooms

A private room can have different types of call participants:

1. Participants with tokens, including:

- Meeting owners (sometimes referred to as "hosts" or "admins"), who are call participants with an owner meeting token. (Read more about owner meeting tokens in our guide.)
- Guests with a non-owner meeting token for that room.

2. Guests without a meeting token, assuming the `enable_knocking` room setting is enabled. (If it's not enabled, guests without tokens have no way to enter a call.)

Using our house analogy mentioned above, the meeting owner is like the owner of the house. They can go in whenever they want because it's their house and the owner meeting token acts as a key to unlock the door. When you have an owner meeting token, you also have special privileges, like being able to decide who else can enter the call.

Guests who enter with regular (i.e., non-owner) meeting tokens can enter right away since they also have a "key" to let themselves in. However, they don't have any special privileges. For example, if a guest without a token knocks to join the meeting, guests with non-owner tokens are not able to let them in.

If knocking is enabled, guests without tokens can request to be admitted into the call by a meeting owner by knocking.

If knocking is not enabled and a guest doesn't have a meeting token, they can't get in.



Now let's now look at how to create meeting tokens for private rooms.

How to create meeting tokens

To create an owner meeting token, use the REST API's meeting token endpoint and set the `is_owner` option to `true`, like so:

```
curl -H "Content-Type: application/json" \
     -H "Authorization: Bearer $DAILY_API_KEY" \
     -XPOST -d '{"properties":
                 {"is_owner":true, "room_name":"DAILY_ROOM_NAME"}}' \
     https://api.daily.co/v1/meeting-tokens
```

Note: You can currently only create meeting tokens via the REST API – not in the dashboard.

We'll test the knocking feature with Daily Prebuilt at the end of this post, so if you want to try it out yourself, create a token with the cURL command above.

To create a non-owner meeting token, create a token without the `is_owner` property, or explicitly set it to `false`:

```
curl -H "Content-Type: application/json" \
     -H "Authorization: Bearer $DAILY_API_KEY" \
     -XPOST -d '{"properties":
                 {"room_name":"DAILY_ROOM_NAME"}}' \
     https://api.daily.co/v1/meeting-tokens
```

Note: We recommend always including a `room name` and `expiry` option for meeting tokens since they cannot be invalidated after creating them. If a room name is not specified, the token will work for any room under your domain.

How to join a call with a meeting token

Daily calls can be joined in a few different ways, including:

- Navigating to a room's URL in your browser. The video call UI you see is Daily Prebuilt.

- Embedding Daily Prebuilt in your app's UI and programmatically joining the room using its URL.
- Building a custom UI with Daily's client SDK for JavaScript and programmatically joining the room using its URL..

To join with a token directly through a room URL in your browser, include a meeting token in the URL using a `?t=[token]` query param.

```
https://your-domain.daily.co/test-room-name?t=DAILY_MEETING_TOKEN
```

If your app has Daily Prebuilt embedded or a custom UI, it is recommended to pass the token in the `join()` method options instead of the room URL.

```
callObject.join({
  url: https://your-domain.daily.co/test-room-name,
  token: 'DAILY_MEETING_TOKEN'
});
```

Next, let's see how this feature looks in Daily Prebuilt. Afterwards, we'll give an overview of the methods and events to use if you're building a knocking feature yourself.

Joining a call with knocking enabled via Daily Prebuilt

Knocking is available in Daily video calls whether you're using Daily Prebuilt or you've built a custom UI with Daily's client SDK for JavaScript. The main difference is that Daily Prebuilt's UI is already built for you – just like the name suggests. As soon as you've updated your room's settings, Daily Prebuilt's UI will update accordingly so that guests can knock to enter the call.

(To contrast this, in a custom UI, you need to build out the functionality of the knocking feature yourself. This usually includes adding a button for guests to knock and a mechanism for meeting owners to review and then permit or reject the person who is knocking.)

When knocking is enabled and the call participants are joining via Daily Prebuilt, there are three possible user paths:

1. A participant with a valid meeting token joins and skips the knocking feature altogether. This could be someone with an owner or non-owner meeting token.
2. A guest tries to join the call without a token, knocks to enter, and the meeting owner accepts their request. The guest is let into the video call.
3. A guest tries to join the call without a token, knocks to enter, and the meeting owner rejects their request. The guest is shown a message that their request was denied. 🥲

When a meeting owner (i.e., a participant with a valid owner meeting token) is in the video call, they'll get an alert that the guest is trying to join anytime a guest knocks. Other guests in the call will not see this alert or be aware of the knocking. If no meeting owner is present, the knocking cannot be responded to, just like if you knock on someone's door and no one is home.

Let's see how each of these user paths look in Daily Prebuilt. The meeting owner's experience will be shown in both videos so they can be in the call to receive the knock.

First, let's start with the guest who knocks and is let in:

0:00 / 0:28 1×

Guest participant knocking and being admitted into a Daily video room

On the left side of the screen, we see that the owner is let right in when they join with a meeting token. On the right side, the guest knocks. We then see an alert pop up in the owner's view. They click "Allow" on the alert and the guest is let into the call.

Now, let's see what it looks like when a guest is denied access to the call.

0:00 / 0:08 1×

Guest participant knocking and not being admitted in to a Daily video room

The guest knocks and the alert is shown to the owner. The owner clicks "Deny" and the guest is shown a message that their request was denied. We explicitly tell the guest this because we always want users to understand what happened so they don't think it was an error or a bug. The guest can revisit the original room URL to knock again if they think they were rejected accidentally, though.

Building a custom knocking feature: An overview

Daily Prebuilt is built with Daily's client SDK for JavaScript, which means anything in Daily Prebuilt can be built custom, too. Custom UIs are more complex than Daily Prebuilt, so we'll cover this topic more thoroughly in a future post.

For now, let's review the important methods and events you'll need to use to build this feature yourself.

Note: When we talk about a guest without a token "joining" a private call, they may be in the "lobby" or be in the actual call with other participants. If they are in the lobby, they have requested to join a call but have not been been granted access yet.

Knocking-related methods in our Client SDK for JavaScript

`preAuth()` : Pre-authenticates a participant before joining a call. This allows you to access certain call information, such as the participant's access level before they try to join the call.

`accessState()` : Specifically returns the access state level of a call participant. Access state levels can be any of the following:

- `"full"` : The participant is authorized to join.
- `"lobby"` : The participant must knock to enter.
- `"none"` : The participant is not authorized to join and cannot knock.
- `"unknown"` : Their access state has not been determined yet. The participant has not attempted to join or preauthenticated joining.

`requestAccess()` : Requesting access is the equivalent to knocking. If a guest calls this method, they are asking to update their access state level from `"lobby"` to `"full"` .

`waitingParticipants()` : This allows meeting owners to retrieve the list of guests (or "participants") waiting in the lobby. Anyone in the list has knocked and is waiting to be let in. ☝️

`updateWaitingParticipants()` : Allows the meeting owner to respond to a request to join (i.e., knocking).

Knocking-related events in our Client SDK for JavaScript:

`"waiting-participant-added"` : Emitted when a participant has been added to the `waitingParticipants()` list.

`"waiting-participant-updated"` : Emitted when a participant in the `waitingParticipants()` list updates their name.

`"waiting-participant-removed"` : Emitted when a participant has been removed from the `waitingParticipants()` list.

`"error"` : Emitted when an unrecoverable error occurs in a call. In this case, the error event with an `errorMsg` of "Join request rejected" tells us a meeting owner denied the knocking request.

`"access-state-updated"` : Emitted when the access state of the local participant has changed. This will be emitted when a meeting owner responds to a knocking request.

This list should help get you started, but stay tuned for our upcoming tutorial on building this feature with Daily's client SDK for JavaScript.

Conclusion

In this post, we reviewed call permissions and how to enable knocking for private Daily rooms. In the next post, we'll look at how to build a custom UI that offers a knocking feature similar to that of Daily Prebuilt.

In the meantime, learn more about knocking and call permissions in our room privacy guide and our tutorial on building a prejoin UI, which is where the knocking feature is typically added in a custom UI.

Never miss a story

Get the latest direct to your inbox.

Email*

Country*

Join peerConnection - our video/WebRTC developer community.

By submitting, you consent to allow Daily.co to store and process the personal data from this form to fulfill your request.

Subscribe



daily

Interactive Live Streaming for Dummies: Daily.co Special Edition

Get your free copy

Startup program

Use Cases

Telehealth

Live Commerce

EdTech

Event Tech

Creator Technology

Virtual Collaboration

Social and Gaming

Read case studies

Dev Resources

Blog

Guides

Demos

Support

API status

Changelog

Contact us

Contact Us

Live chat

Support

Japan

Resources

Blog

Resource center

Events

Twillio alternative

Vonage alternative

Agora alternative

Videosaurus

Site index

Security & Privacy

Security center

Privacy policy

Terms of service

Cookie policy