

[AlarmKit](#) / [AlarmManager](#) / AlarmManager.AlarmConfiguration

Structure

AlarmManager.AlarmConfiguration

An object that contains all the properties necessary to schedule an alarm.

iOS 26.0+ | iPadOS 26.0+

```
struct AlarmConfiguration<Metadata> where Metadata : AlarmMetadata
```

Overview

Pass the schedule or countdown and any attributes you define to the `AlarmConfiguration` for the system to schedule. You can pass in an optional secondary intent that the system executes when a person taps a secondary button. This is only available after first unlock. You can also include custom sounds for your alarm.

The following example configures an alarm with a countdown duration.

```
let configuration = AlarmManager.AlarmConfiguration(countdownDuration: Alarm.CountdownDuration,
    schedule: nil,
    attributes: attributes,
    secondaryIntent: repeatIntent,
    alertConfiguration: AlertConfiguration(title: "Eggs are ready!", body: "Time to
```

Topics

Configuring a scheduled alarm

```
static func alarm(schedule: Alarm.Schedule?, attributes: Alarm
Attributes<Metadata>, stopIntent: (any LiveActivityIntent)?, secondary
Intent: (any LiveActivityIntent)?, sound: AlertConfiguration.AlertSound
) -> AlarmManager.AlarmConfiguration<Metadata>
```

Creates a configuration that behaves like a traditional alarm.

Configuring a countdown

```
init(countdownDuration: Alarm.CountdownDuration?, schedule: Alarm.
Schedule?, attributes: AlarmAttributes<Metadata>, stopIntent: (any Live
ActivityIntent)?, secondaryIntent: (any LiveActivityIntent)?, sound:
AlertConfiguration.AlertSound)
```

Creates a configuration that behaves like a countdown.

```
static func timer(duration: TimeInterval, attributes: AlarmAttributes<
Metadata>, stopIntent: (any LiveActivityIntent)?, secondaryIntent: (any
LiveActivityIntent)?, sound: AlertConfiguration.AlertSound) -> Alarm
Manager.AlarmConfiguration<Metadata>
```

Creates a configuration that behaves like a traditional timer.

See Also

Scheduling an alarm

```
func schedule<Metadata>(id: Alarm.ID, configuration: AlarmManager.Alarm
Configuration<Metadata>) async throws -> Alarm
```

Schedules a new alarm.