RFC 9074
"VALARM" Extensions for iCalendar

Abstract

This document defines a set of extensions to the iCalendar "VALARM" component to enhance the use of alarms and improve interoperability between clients and servers.

This document updates RFC 5545.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc9074.

Copyright Notice

Table of Contents

1. Introduction

The iCalendar specification [RFC5545] defines a set of components used to describe calendar data. One of those is the "VALARM" component, which appears as a subcomponent of the "VEVENT" and "VTODO" components. The "VALARM" component is used to specify a reminder for an event or task. Different alarm actions are possible, as are different ways to specify how the alarm is triggered.

As iCalendar has become more widely used and as client-server protocols, such as Calendaring Extensions to WebDAV (CalDAV) [RFC4791], have become more prevalent, several issues with "VALARM" components have arisen. Most of these relate to the need to extend the existing "VALARM" component with new properties and behaviors to allow clients and servers to accomplish specific tasks in an interoperable manner. For example, clients typically need a way to specify that an alarm has been dismissed by a calendar user or has been "snoozed" by a set amount of time. To date, this has been done through the use of custom "X-" properties specific to each client implementation, leading to poor interoperability.

This specification defines a set of extensions to "VALARM" components to cover common requirements for alarms not currently addressed in iCalendar. Each extension is defined in a separate section below. For the most part, each extension can be supported independently of the others; though, in some cases, one extension will require another. In addition, this specification describes mechanisms by which clients can interoperably implement common features, such as "snoozing".

2. Conventions Used in This Document

The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**NOT RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The notation used in this memo to (re-)define iCalendar elements is the ABNF notation of [RFC5234] as used by [RFC5545]. Any syntax elements shown below that are not explicitly defined in this specification come from iCalendar [RFC5545].

When XML element types in the namespaces "DAV:" and "urn:ietf:params:xml:ns:caldav" are referenced in this document outside of the context of an XML fragment, the string "DAV:" and "CALDAV:" will be prefixed to the element type names, respectively.

3. Extensible Syntax for VALARM

Section 3.6.6 of [RFC5545] defines the syntax for "VALARM" components and properties within them. However, as written, it is hard to extend this, e.g., by adding a new property common to all types of alarms. Since many of the extensions defined in this document need to extend the base

syntax, an alternative form for the base syntax is defined here, with the goal of simplifying specification of the extensions while augmenting the existing functionality defined in [RFC5545] to allow for nested subcomponents (as required by proximity alarm triggers (Section 8)).

A "VALARM" calendar component is redefined by the following notation:

```
alarmcext  = "BEGIN" ":" "VALARM" CRLF
               *alarmprop *alarm-subcomp
               "END" ":" "VALARM" CRLF

alarmprop  = (

               ; the following are REQUIRED
               ; but MUST NOT occur more than once

               action / trigger /

               ; one set of action properties MUST be
               ; present and MUST match the action specified
               ; in the ACTION property

               actionprops /

               ; the following are OPTIONAL
               ; and MAY occur more than once

               x-prop / iana-prop

               )

actionprops = *audiopropext / *disppropext / *emailpropext

audiopropext  = (

               ; 'duration' and 'repeat' are both OPTIONAL
               ; and MUST NOT occur more than once each,
               ; but if one occurs, so MUST the other

               duration / repeat /

               ; the following is OPTIONAL
               ; but MUST NOT occur more than once

               attach

               )

disppropext = (

               ; the following are REQUIRED
               ; but MUST NOT occur more than once

               description /

               ; 'duration' and 'repeat' are both OPTIONAL
               ; and MUST NOT occur more than once each,
               ; but if one occurs, so MUST the other

               duration / repeat

               )

emailpropext = (
```

```
                    ; the following are all REQUIRED
                    ; but MUST NOT occur more than once

                    description / summary /

                    ; the following is REQUIRED
                    ; and MAY occur more than once

                    attendee /

                    ; 'duration' and 'repeat' are both OPTIONAL
                    ; and MUST NOT occur more than once each,
                    ; but if one occurs, so MUST the other

                    duration / repeat

                    ; the following is OPTIONAL
                    ; and MAY occur more than once

                    attach

                    )

    alarm-subcomp = (

                    ; the following are OPTIONAL
                    ; and MAY occur more than once

                    x-comp / iana-comp

                    )
```

4. Alarm Unique Identifier

This extension adds a "UID" property to "VALARM" components to allow a unique identifier to be specified. The value of this property can then be used to refer uniquely to the "VALARM" component.

The "UID" property defined here follows the definition in Section 3.8.4.7 of [RFC5545] with the security and privacy updates in Section 5.3 of [RFC7986]. In particular, it **MUST** be a globally unique identifier that does not contain any security- or privacy-sensitive information.

The "VALARM" component defined in Section 3 is extended here as:

```
  alarmprop  =/ (

                ; the following is OPTIONAL
                ; but MUST NOT occur more than once

                uid

                )
```

5. Alarm Related To

It is often convenient to relate one or more "VALARM" components to other "VALARM" components (e.g., see Section 7). This can be accomplished if the "VALARM" components each have their own "UID" property (as per Section 4).

This specification updates the usage of the "RELATED-TO" property defined in Section 3.8.4.5 of [RFC5545] to enable its use with "VALARM" components. Specific types of relationships between "VALARM" components can be identified by registering new values for the "RELTYPE" property parameter defined in Section 3.2.15 of [RFC5545].

The "VALARM" component defined in Section 3 is extended here as:

```
alarmprop  =/ (

             ; the following is OPTIONAL
             ; and MAY occur more than once

             related

             )
```

6. Alarm Acknowledgement

There is currently no way for a "VALARM" component to indicate whether it has been triggered and acknowledged. With the advent of a standard client/server protocol for calendaring and scheduling data ([RFC4791]), it is quite possible for an event with an alarm to exist on multiple clients in addition to the server. If each of those is responsible for performing the action when an alarm triggers, then multiple "alerts" are generated by different devices. In such a situation, a calendar user would like to be able to "dismiss" the alarm on one device and have it automatically dismissed on the others, too.

Also, with recurring events that have alarms, it is important to know when the last alarm in the recurring set was acknowledged so that the client can determine whether past alarms have been missed.

To address these needs, this specification adds an "ACKNOWLEDGED" property to "VALARM" components to indicate when the alarm was last acknowledged (or sent, if acknowledgement is not possible). This is defined by the syntax below.

```
alarmprop       =/ (

                    ; the following is OPTIONAL
                    ; but MUST NOT occur more than once

                    acknowledged

                    )
```

6.1. Acknowledged Property

Property Name: ACKNOWLEDGED

Purpose: This property specifies the UTC date and time at which the corresponding alarm was last sent or acknowledged.

Value Type: DATE-TIME

Property Parameters: IANA and nonstandard property parameters can be specified on this property.

Conformance: This property can be specified within "VALARM" calendar components.

Description: This property is used to specify when an alarm was last sent or acknowledged. This allows clients to determine when a pending alarm has been acknowledged by a calendar user so that any alerts can be dismissed across multiple devices. It also allows clients to track repeating alarms or alarms on recurring events or to-dos to ensure that the right number of missed alarms can be tracked.

Clients **SHOULD** set this property to the current date-time value in UTC when a calendar user acknowledges a pending alarm. Certain kinds of alarms, such as email-based alerts, might not provide feedback as to when the calendar user sees them. For those kinds of alarms, the client **SHOULD** set this property when the alarm is triggered and the action is successfully carried out.

When an alarm is triggered on a client, clients can check to see if an "ACKNOWLEDGED" property is present. If it is, and the value of that property is greater than or equal to the computed trigger time for the alarm, then the client **SHOULD NOT** trigger the alarm. Similarly, if an alarm has been triggered and an "alert" has been presented to a calendar user, clients can monitor the iCalendar data to determine whether an "ACKNOWLEDGED" property is added or changed in the alarm component. If the value of any "ACKNOWLEDGED" property in the alarm changes and is greater than or equal to the trigger time of the alarm, then clients **SHOULD** dismiss or cancel any "alert" presented to the calendar user.

Format Definition: This property is defined by the following notation:

```
acknowledged = "ACKNOWLEDGED" *acknowledgedparam ":" datetime CRLF

acknowledgedparam  = (

                    ; the following is OPTIONAL
                    ; and MAY occur more than once

                    (";" other-param)

                    )
```

Example: The following is an example of this property:

```
ACKNOWLEDGED:20090604T084500Z
```

7. Snoozing Alarms

Users often want to "snooze" an alarm, and this specification defines a standard approach to accomplish that.

To "snooze" an alarm that has been triggered, clients **MUST** do the following:

1. Set the "ACKNOWLEDGED" property (see Section 6.1) on the triggered alarm.
2. Create a new "VALARM" component (the "snooze" alarm) within the parent component of the triggered alarm (i.e., as a "sibling" component of the triggered alarm).
   a. The new "snooze" alarm **MUST** be set to trigger at the user's chosen "snooze" interval after the original alarm is triggered. Clients **SHOULD** use an absolute "TRIGGER" property with a "DATE-TIME" value specified in UTC.
   b. The new "snooze" alarm **MUST** have a "RELATED-TO" property (see Section 5) with a value set to the "UID" property value of the original "VALARM" component that was triggered. If the triggered "VALARM" component does not already have a "UID" property, the client **MUST** add one. The "RELATED-TO" property added to the new "snooze" alarm **MUST** include a "RELTYPE" property parameter with a value set to "SNOOZE" (see Section 7.1).

3. When the "snooze" alarm is triggered, the client **MUST** do the following:
   a. Update the "ACKNOWLEDGED" property on the original related alarm.
   b. If the "snooze" alarm is itself "snoozed", the client **MUST** remove the "snooze" alarm component and return to step 2.

      Otherwise, if the "snooze" alarm is dismissed, the client **MUST** do one of the following:
      ▪ Set the "ACKNOWLEDGED" property on the "snooze" alarm.
      ▪ Remove the "snooze" alarm component.

Note that regardless of the final disposition of the "snooze" alarm when triggered, the original "VALARM" component is left unchanged other than updating its "ACKNOWLEDGED" property.

7.1. Relationship Type Property Parameter

This specification adds the "SNOOZE" relationship type for use with the "RELTYPE" property defined in Section 3.2.15 of [RFC5545]. This is used when relating a "snoozed" "VALARM" component to the original alarm that the "snooze" was generated for.

7.2. Example

The following example shows the "snoozing", "re-snoozing", and dismissal of an alarm. Note that the encompassing "VCALENDAR" component has been omitted for brevity and that the line breaks surrounding the "VALARM" components are for clarity only and would not be present in the actual iCalendar data.

Assume that we have the following event with an alarm set to trigger 15 minutes before the meeting:

```
BEGIN:VEVENT
CREATED:20210302T151004Z
UID:AC67C078-CED3-4BF5-9726-832C3749F627
DTSTAMP:20210302T151004Z
DTSTART;TZID=America/New_York:20210302T103000
DTEND;TZID=America/New_York:20210302T113000
SUMMARY:Meeting

BEGIN:VALARM
UID:8297C37D-BA2D-4476-91AE-C1EAA364F8E1
TRIGGER:-PT15M
DESCRIPTION:Event reminder
ACTION:DISPLAY
END:VALARM

END:VEVENT
```

When the alarm is triggered, the user decides to "snooze" it for 5 minutes. The client acknowledges the original alarm and creates a new "snooze" alarm as a sibling of, and relates it to, the original alarm (note that both occurrences of "VALARM" reside within the same "parent" VEVENT):

```
BEGIN:VEVENT
CREATED:20210302T151004Z
UID:AC67C078-CED3-4BF5-9726-832C3749F627
DTSTAMP:20210302T151516Z
DTSTART;TZID=America/New_York:20210302T103000
DTEND;TZID=America/New_York:20210302T113000
SUMMARY:Meeting

BEGIN:VALARM
UID:8297C37D-BA2D-4476-91AE-C1EAA364F8E1
TRIGGER:-PT15M
DESCRIPTION:Event reminder
ACTION:DISPLAY
ACKNOWLEDGED:20210302T151514Z
END:VALARM

BEGIN:VALARM
UID:DE7B5C34-83FF-47FE-BE9E-FF41AE6DD097
TRIGGER;VALUE=DATE-TIME:20210302T152000Z
RELATED-TO;RELTYPE=SNOOZE:8297C37D-BA2D-4476-91AE-C1EAA364F8E1
DESCRIPTION:Event reminder
ACTION:DISPLAY
END:VALARM

END:VEVENT
```

When the "snooze" alarm is triggered, the user decides to "snooze" it again for an additional 5 minutes. The client once again acknowledges the original alarm, removes the triggered "snooze" alarm, and creates another new "snooze" alarm as a sibling of, and relates it to, the original alarm (note the different UID for the new "snooze" alarm):

```
BEGIN:VEVENT
CREATED:20210302T151004Z
UID:AC67C078-CED3-4BF5-9726-832C3749F627
DTSTAMP:20210302T152026Z
DTSTART;TZID=America/New_York:20210302T103000
DTEND;TZID=America/New_York:20210302T113000
SUMMARY:Meeting

BEGIN:VALARM
UID:8297C37D-BA2D-4476-91AE-C1EAA364F8E1
TRIGGER:-PT15M
DESCRIPTION:Event reminder
ACTION:DISPLAY
ACKNOWLEDGED:20210302T152024Z
END:VALARM

BEGIN:VALARM
UID:87D690A7-B5E8-4EB4-8500-491F50AFE394
TRIGGER;VALUE=DATE-TIME:20210302T152500Z
RELATED-TO;RELTYPE=SNOOZE:8297C37D-BA2D-4476-91AE-C1EAA364F8E1
DESCRIPTION:Event reminder
ACTION:DISPLAY
END:VALARM

END:VEVENT
```

When the second "snooze" alarm is triggered, the user decides to dismiss it. The client acknowledges both the original alarm and the second "snooze" alarm:

```
BEGIN:VEVENT
CREATED:20210302T151004Z
UID:AC67C078-CED3-4BF5-9726-832C3749F627
DTSTAMP:20210302T152508Z
DTSTART;TZID=America/New_York:20210302T103000
DTEND;TZID=America/New_York:20210302T113000
SUMMARY:Meeting

BEGIN:VALARM
UID:8297C37D-BA2D-4476-91AE-C1EAA364F8E1
TRIGGER:-PT15M
DESCRIPTION:Event reminder
ACTION:DISPLAY
ACKNOWLEDGED:20210302T152507Z
END:VALARM

BEGIN:VALARM
UID:87D690A7-B5E8-4EB4-8500-491F50AFE394
TRIGGER;VALUE=DATE-TIME:20210302T152500Z
RELATED-TO;RELTYPE=SNOOZE:8297C37D-BA2D-4476-91AE-C1EAA364F8E1
DESCRIPTION:Event reminder
ACTION:DISPLAY
ACKNOWLEDGED:20210302T152507Z
END:VALARM

END:VEVENT
```

8. Alarm Proximity Trigger

Currently, a "VALARM" is triggered when a specific date-time value is reached. It is also desirable to be able to trigger alarms based on location, e.g., when arriving at or departing from a particular location.

This specification adds the following elements to "VALARM" components to indicate when an alarm can be triggered based on location.

"PROXIMITY" property: indicates that a location-based trigger is to be used and which action is used for the trigger

"VLOCATION" component(s) [RFC9073]:   used to indicate the actual location(s) to trigger off of, specified with a URL property containing a 'geo' URI [RFC5870], which allows for two or three coordinate values with an optional uncertainty

```
alarmprop       =/ (

                    ; the following is OPTIONAL
                    ; but MUST NOT occur more than once

                    proximity /

                    )
alarm-subcomp   =/ (

                    ; the following is OPTIONAL
                    ; and MAY occur more than once but only
                    ; when a PROXIMITY property is also present

                    locationc

                    )
```

Typically, when a "PROXIMITY" property is used, there is no need to specify a time-based trigger using the "TRIGGER" property. However, since "TRIGGER" is defined as a required property for a "VALARM" component, for backwards compatibility, it has to be present but ignored. To indicate a "TRIGGER" that is to be ignored, clients **SHOULD** use a value a long time in the past. A value of "19760401T005545Z" has been commonly used for this purpose.

8.1. Proximity Property

Property Name: PROXIMITY

Purpose: This property indicates that a location-based trigger is applied to an alarm.

Value Type: TEXT

Property Parameters: IANA and nonstandard property parameters can be specified on this property.

Conformance: This property can be specified within "VALARM" calendar components.

Description: This property is used to indicate that an alarm has a location-based trigger. Its value identifies the action that will trigger the alarm.

When the property value is set to "ARRIVE", the alarm is triggered when the calendar user agent arrives in the vicinity of one or more locations. When set to "DEPART", the alarm is triggered when the calendar user agent departs from the vicinity of one or more locations. Each location **MUST** be specified with a "VLOCATION" component. Note that the meaning of "vicinity" in this context is implementation defined.

When the property value is set to "CONNECT", the alarm is triggered when the calendar user agent connects to an automobile to which it has been paired via Bluetooth [BTcore]. When set to "DISCONNECT", the alarm is triggered when the calendar user agent disconnects from an

automobile to which it has been paired via Bluetooth. Note that neither current implementations of proximity alarms nor this document have a mechanism to target a particular automobile. Such a mechanism may be specified in a future extension.

Format Definition: This property is defined by the following notation:

```
proximity = "PROXIMITY" *proximityparam ":" proximityvalue CRLF

proximityparam  = (

                 ; the following is OPTIONAL
                 ; and MAY occur more than once

                 (";" other-param)

                 )
proximityvalue  = "ARRIVE" / "DEPART" /
                  "CONNECT" / "DISCONNECT" / iana-token / x-name
```

8.2. Example

The following example shows a "VALARM" component with a proximity trigger set to trigger when the device running the calendar user agent leaves the vicinity defined by the URL property in the "VLOCATION" component. Note use of the "u=" parameter with the 'geo' URI to define the uncertainty of the location determination.

```
BEGIN:VALARM
UID:77D80D14-906B-4257-963F-85B1E734DBB6
ACTION:DISPLAY
TRIGGER;VALUE=DATE-TIME:19760401T005545Z
DESCRIPTION:Remember to buy milk
PROXIMITY:DEPART
BEGIN:VLOCATION
UID:123456-abcdef-98765432
NAME:Office
URL:geo:40.443,-79.945;u=10
END:VLOCATION
END:VALARM
```

9. Security Considerations

In addition to the security properties of iCalendar (see Section 7 of [RFC5545]), a "VALARM", if not monitored properly, can be used to disturb users and/or leak personal information. For instance, an undesirable audio alert could cause embarrassment; an unwanted display alert could be considered an annoyance; or an email alert could be used to leak a user's location to a third party or to send unsolicited email to multiple users. Therefore, CalDAV clients and servers that accept

iCalendar data from a third party (e.g., via iCalendar Transport-Independent Interoperability Protocol (iTIP) [RFC5546], a subscription feed, or a shared calendar) **SHOULD** remove each "VALARM" from the data prior to storing in their calendar system.

Security considerations related to unique identifiers for "VALARM" are discussed in Section 4.

10. Privacy Considerations

A proximity "VALARM", if not used carefully, can leak a user's past, present, or future location. For instance, storing an iCalendar resource containing proximity "VALARM"s to a shared calendar on CalDAV server can expose to anyone that has access to that calendar the user's intent to leave from or arrive at a particular location at some future time. Furthermore, if a CalDAV client updates the shared iCalendar resource with an "ACKNOWLEDGED" property when the alarm is triggered, this will leak the exact date and time that the user left from or arrived at the location. Therefore, CalDAV clients that implement proximity alarms **SHOULD** give users the option of storing and/or acknowledging the alarms on the local device only and not storing the alarm and/or acknowledgement on a remote server.

Privacy considerations related to unique identifiers for "VALARM" are discussed in Section 4.

11. IANA Considerations

11.1. Property Registrations

This document defines the following new iCalendar properties that have been added to the "Properties" registry defined in Section 8.2.3 of [RFC5545] and located here: <https://www.iana.org/assignments/icalendar>.

Property	Status	Reference
ACKNOWLEDGED	Current	RFC 9074, Section 6.1
PROXIMITY	Current	RFC 9074, Section 8.1

Table 1: Additions to the Properties Registry

11.2. Relationship Types Registry

This document defines the following new iCalendar relationship type that has been added to the "Relationship Types" registry defined in Section 8.3.8 of [RFC5545] and located here: <https://www.iana.org/assignments/icalendar>.

Relationship Type	Status	Reference
SNOOZE	Current	RFC 9074, Section 7.1

Table 2: Addition to the Relationship Types Registry

11.3. Proximity Values Registry

A new iCalendar registry for values of the "PROXIMITY" property has been created and is located here: <https://www.iana.org/assignments/icalendar>.

Additional values **MAY** be used, provided the process described in Section 8.2.1 of [RFC5545] is used to register them, using the template in Section 8.2.6 of [RFC5545].

The following table has been used to initialize the Proximity Value Registry.

Value	Status	Reference
ARRIVE	Current	RFC 9074, Section 8.1
DEPART	Current	RFC 9074, Section 8.1
CONNECT	Current	RFC 9074, Section 8.1
DISCONNECT	Current	RFC 9074, Section 8.1

Table 3: Initial Contents of the Proximity Values Registry

12. References

12.1. Normative References

[RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5234]   Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <https://www.rfc-editor.org/info/rfc5234>.

[RFC5545]   Desruisseaux, B., Ed., "Internet Calendaring and Scheduling Core Object Specification (iCalendar)", RFC 5545, DOI 10.17487/RFC5545, September 2009, <https://www.rfc-editor.org/info/rfc5545>.

[RFC5870]   Mayrhofer, A. and C. Spanring, "A Uniform Resource Identifier for Geographic Locations ('geo' URI)", RFC 5870, DOI 10.17487/RFC5870, June 2010, <https://www.rfc-editor.org/info/rfc5870>.

[RFC7986]   Daboo, C., "New Properties for iCalendar", RFC 7986, DOI 10.17487/RFC7986, October 2016, <https://www.rfc-editor.org/info/rfc7986>.

[RFC8174]   Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC9073]   Douglass, M., "Event Publishing Extensions to iCalendar", RFC 9073, DOI 10.17487/RFC9073, August 2021, <https://www.rfc-editor.org/info/rfc9073>.

12.2. Informative References

[BTcore]   Bluetooth Special Interest Group, "Bluetooth Core Specification Version 5.0 Feature Overview", December 2016, <https://www.bluetooth.com/bluetooth-resources/bluetooth-5- go-faster-go-further/>.

[RFC4791]   Daboo, C., Desruisseaux, B., and L. Dusseault, "Calendaring Extensions to WebDAV (CalDAV)", RFC 4791, DOI 10.17487/RFC4791, March 2007, <https://www.rfc-editor.org/info/rfc4791>.

[RFC5546]   Daboo, C., Ed., "iCalendar Transport-Independent Interoperability Protocol (iTIP)", RFC 5546, DOI 10.17487/RFC5546, December 2009, <https://www.rfc-editor.org/info/rfc5546>.

Acknowledgements

Authors' Addresses

Cyrus Daboo
Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
United States of America
Email: cyrus@daboo.name
URI: http://www.apple.com/

Kenneth Murchison (EDITOR)
Fastmail US LLC
Suite 1201
1429 Walnut St
Philadelphia, PA 19102
United States of America
Email: murch@fastmailteam.com
URI: http://www.fastmail.com/