

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 13, 2011

C. Daboo
Apple
July 12, 2010

**VALARM Extensions for iCalendar
draft-daboo-valarm-extensions-01**

Abstract

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

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 13, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	3
2.	Conventions Used in This Document	3
3.	Extensible syntax for VALARM	3
4.	VALARM Unique Identifier	5
5.	VALARM URI Action	6
6.	VALARM Agent Property	7
6.1.	Alarm Agent Property	8
6.2.	URI Property Parameter	9
7.	VALARM Status Property	10
7.1.	Last Triggered Property	11
8.	Snoozing Alarms	12
9.	Default Alarms in CalDAV	12
10.	Security Considerations	13
11.	IANA Considerations	13
12.	Acknowledgments	13
13.	References	14
13.1.	Normative References	14
13.2.	Informative References	14
Appendix A.	Change History (To be removed by RFC Editor before publication)	14
	Author's Address	15

1. Introduction

The iCalendar [[I-D.ietf-calsify-rfc2445bis](#)] specification defines a set of components used to describe calendar data. One of those is the "VALARM" component which appears as a sub-component of "VEVENT" and "VTODO" components. The "VALARM" component is used to specify a reminder for an event or to-do. 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 CalDAV [[RFC4791](#)] have become more popular, 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 [[RFC2119](#)].

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 [[I-D.ietf-calsify-rfc2445bis](#)] defines the syntax for "VALARM" components and properties within them. However, as written,

Daboo

Expires January 13, 2011

[Page 3]

it is hard to extend this by adding, e.g., a new property common to all types of alarm. 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.

A "VALARM" calendar component is re-defined by the following notation:

```
alarmcext  = "BEGIN" ":" "VALARM" CRLF
             alarmprop
             "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 is 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

    )
```

4. VALARM Unique Identifier

Several of the other extensions in this specification require identifying a specific instance of a "VALARM" component in an iCalendar stream. To aid that, this extension adds a "UID" property to "VALARM" components to allow a unique identifier to be specified. The value of this unique identifier 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 [[I-D.ietf-calsify-rfc2445bis](#)]. In particular it MUST be a globally unique identifier.

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. VALARM URI Action

Currently "VALARM" components have actions for audio, display and email. New types of action are of interest, e.g., SMS, instant messaging, etc. Rather than specify separate actions for these, an alternative is to define a "URI" action that allows any URI scheme to be used as an action, where it makes sense. Thus URI schemes for IM [[RFC3860](#)], SIP [[RFC3261](#)], TEL [[RFC3966](#)] etc could be used.

This extension defines a new "URI" property value for use with the "ACTION" property in "VALARM" components. A new set of action properties is defined for "VALARM" components based on this new action as defined by the syntax below.


```
actionvalue /= "URI"
            ; Adds a new action for a "VALARM"

actionprop  /= uriprop
            ; Re-defines the "VALARM" component to include
            ; "uriprop" as a new set of action properties

uriprop     = *(

            ; the following is REQUIRED,
            ; but MUST NOT occur more than once

            uri /

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

            duration / repeat

            )
```

6. VALARM Agent Property

With the advent of a standard client/server protocol for calendaring and scheduling data ([[RFC4791](#)]) there is a need to specify which client or server should handle the presentation of an alarm when it is triggered. For example, calendar users want to be able to receive alarms at all times, even when their desktop client might be unavailable. Since the server is "always on", a service running on the server could monitor alarm status and, when appropriate, trigger those alarms. In addition it may be important for only the server or the client to be set to handle an alarm - and in some cases only specific servers or clients.

To address this need, this specification defines an "ALARM-AGENT" iCalendar property that can be added to any "VALARM" component. This property specifies whether a client or server or both should be responsible for processing the alarm when it triggers. In addition, a "URI" property parameter can be added to uniquely identify the client or server that should process the alarm. This is defined by the syntax below.

Daboo

Expires January 13, 2011

[Page 7]

```
alarmprop /= *(  
    ; the following is OPTIONAL,  
    ; and MAY occur more than once  
  
    alarm-agent  
  
    )
```

6.1. Alarm Agent Property

Property Name: ALARM-AGENT

Purpose: This property specifies whether a client, server, both or none gets to process an alarm when it is triggered.

Value Type: TEXT

Property Parameters: IANA, non-standard, and URI 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 who is responsible for processing an alarm when it is triggered. When the value "SERVER" is specified, only a server that has a copy of the event is responsible. When the value "CLIENT" is specified, only a client that has a copy of the event is responsible. When the value "BOTH" is specified, either a client or server is responsible. When the value "NONE" is specified, neither a client nor server is responsible (i.e., the alarm action is never carried out when it triggers).

If the "URI" property parameter is specified for "SERVER" or "CLIENT" values, then only the client or server identified by the "URI" value is responsible. Clients or servers that are not responsible for the alarm SHOULD NOT process the alarm action when it is triggered. If multiple clients or servers need to be specified, then the "ALARM-AGENT" property should be included multiple times in the "VALARM" component, with each one containing the appropriate "URI" property parameter value to identify each client or server.

In the absence of this property clients and servers can choose to process the alarm themselves as appropriate when it triggers.

This property MAY occur more than once to allow multiple

Daboo

Expires January 13, 2011

[Page 8]

processors of an alarm.

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

```
alarm-agent      = "ALARM-AGENT" alarmagentparam ":"
                  alarmagentvalue CRLF

alarmagentparam  = *(
                    ; the following is OPTIONAL,
                    ; but MUST NOT occur more than once

                    (";" uriparam) /

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

                    (";" other-param)

                    )

alarmagentvalue  = "SERVER" / "CLIENT" / "BOTH" / "NONE"
                  / iana-token / x-name
```

Example: The following is an example of this property:

```
ALARM-AGENT:SERVER
```

6.2. URI Property Parameter

Property Name: URI

Purpose: This property parameter specifies a URI that is related to the property it is applied to.

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

```
uriparam      = "URI" "=" DQUOTE uri DQUOTE
```

Description: This property parameter is used to specify a URI that is associated with the property it is applied to. Each property that allows this parameter to be specified MUST indicate what the value of the URI represents.

Example: The following is an example of this property:

```
ALARM-AGENT;URI="http://calendar.example.com":SERVER
```

7. VALARM Status Property

There is currently no way for a "VALARM" component to indicate the status of the alarm (e.g., whether it has been triggered or not). 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 dismissed on the others too.

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

To address these needs, this specification adds the following:

- o a "STATUS" property to "VALARM" components with values to indicate whether an alarm is active, inactive, or completed (dismissed).
- o a "LAST-TRIGGERED" property to "VALARM" components to indicate when the alarm was last triggered.

This is defined by the syntax below.


```
alarmprop      /= *(
                ; the following are OPTIONAL,
                ; but MUST NOT occur more than once

                status / last-triggered

                )

statvalue      /= statvalue-alarm

statvalue-alarm = "ACTIVE"      ; Indicates alarm is active and
                                ; waiting to be triggered.
                        / "CANCELLED" ; Indicates alarm is inactive
                                ; and won't be triggered.
                        / "COMPLETED" ; Indicates alarm has triggered
                                ; and been dismissed.
; Status values for a "VALARM"
```

[7.1.](#) Last Triggered Property

Property Name: LAST-TRIGGERED

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

Value Type: DATE-TIME

Property Parameters: IANA and non-standard 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 triggered. This 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.

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


```
last-triggered = "LAST-TRIGGERED" lasttrigparam ":" datetime CRLF

lasttrigparam = *(
    ; the following is OPTIONAL,
    ; and MAY occur more than once

    (";" other-param)
)
```

Example: The following is an example of this property:

```
LAST-TRIGGERED:20090604T084500Z
```

8. Snoozing Alarms

Users often want to "snooze" an alarm. That process effectively creates a new alarm to be triggered a set amount of time after the original being "snoozed". To accomplish this, clients SHOULD create a new "VALARM" component within the same component of the "VALARM" that was triggered and is being "snoozed". The new "VALARM" MUST be set to trigger by the user's chosen "snooze" interval after the original alarm triggered.

9. Default Alarms in CalDAV

Users often want to have a default alarm applied to new events that they create or to new invitations that arrive on a CalDAV [[RFC4791](#)] server. Since this behavior is expected to occur no matter which client a user is using, it is beneficial if the server itself is responsible for managing the creation of the default alarm.

This specification defines four new WebDAV properties that can be used to specify different sets of default alarms. Clients can store "VALARM" components in these properties to setup the defaults. When a new event or todo is created on the server, the server will automatically add the default, as appropriate. When a new event or todo invitation is delivered to the calendar user, the server will automatically add the default alarm, as appropriate.

The four new WebDAV properties are:

CALDAV:default-alarm-vevent-datetime A default alarm applied to "VEVENT" components whose "DTSTART" property value is "DATE-TIME"

Daboo

Expires January 13, 2011

[Page 12]

CALDAV:default-alarm-vevent-date A default alarm applied to "VEVENT" components whose "DTSTART" property value is "DATE"

CALDAV:default-alarm-vtodo-datetime A default alarm applied to "VTODO" components whose "DUE" or "DTSTART" property value is "DATE-TIME"

CALDAV:default-alarm-vtodo-date A default alarm applied to "VTODO" components whose "DUE" or "DTSTART" property value is "DATE", or when neither of those properties is present

The WebDAV properties are defined on a calendar user's "calendar home" collection, or on individual calendar collections. When events or tasks are created in a calendar, the server will first inspect the WebDAV properties on the calendar collection to see if the appropriate property is present. If it is, the server will use the value of that property to set the default in the new component. If the property is not present on the calendar collection, the server will determine whether the property is present on the calendar home collection that contains the calendar collection. If present on the calendar home, the property value there is used as the default.

The WebDAV property value MUST be one or more "VALARM" components or the empty string. If an empty string is used, no default alarms are applied. TODO: more detail here

10. Security Considerations

TODO: talk about importance of stripping VALARMS from incoming iTIP. Talk about VALARMS being used to "spam" - particularly nasty if the server handles it.

11. IANA Considerations

TODO: tables registering new properties, parameters and values.

12. Acknowledgments

This specification came about via discussions at the Calendaring and Scheduling Consortium. Also, thanks to the following for providing feedback: Bernard Desruisseaux, Mike Douglass, Ciny Joy.

13. References

Daboo

Expires January 13, 2011

[Page 13]

13.1. Normative References

- [I-D.ietf-calsify-rfc2445bis]
Desruisseaux, B., "Internet Calendaring and Scheduling Core Object Specification (iCalendar)",
[draft-ietf-calsify-rfc2445bis-10](#) (work in progress),
April 2009.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC4791] Daboo, C., Desruisseaux, B., and L. Dusseault,
"Calendaring Extensions to WebDAV (CalDAV)", [RFC 4791](#),
March 2007.

13.2. Informative References

- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", [RFC 3261](#), June 2002.
- [RFC3860] Peterson, J., "Common Profile for Instant Messaging (CPIM)", [RFC 3860](#), August 2004.
- [RFC3966] Schulzrinne, H., "The tel URI for Telephone Numbers", [RFC 3966](#), December 2004.

Appendix A. Change History (To be removed by RFC Editor before publication)

Changes in -01:

1. Removed DESCRIPTION as an allowed property in the URI alarm.
2. Added statement about what to do when ALARM-AGENT is not present.
3. Allow multiple ALARM-AGENT properties to be present.
4. Removed SNOOZE-UNTIL - snoozing now accomplished by creating a new VALARM.
5. Remove VALARM by reference section.
6. Added more detail to CalDAV default alarms.

Author's Address

Cyrus Daboo
Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
USA

Email: cyrus@daboo.name

URI: <http://www.apple.com/>