


CalDAV-4-Lox

Autor	Sven Thierfelder
Logo	
Status	STABLE
Version	1.0.2
Min. LB Version	LB 1.2
Download	https://github.com/svethi/LoxBerry-CalDAV4Lox/archive/caldav4lox-0.9.7.zip (letzte Version für LB 0.2.3) https://github.com/svethi/LoxBerry-CalDAV4Lox/archive/caldav4lox-1.0.2.zip (aktuelles Release)
Zweck	Das CalDAV Plugin bietet die Möglichkeit, Kalendereinträge in CalDAV oder Google Kalendern für Funktionen aller Art im Loxone MiniServer zu verwenden.
Languages	EN, DE
Diskussion	https://www.loxforum.com/forum/german/software-konfiguration-programm-und-visualisierung/60811-loxberry-caldav-4-lox-plugin-verf%C3%BCgbar

Version: 0.9.1

- kleine Fehlerbehebungen in den dargestellten Befehlserecgnungen

Version: 0.9.2

- weitere Fehlerkorrekturen in den dargestellten Hilfsangaben
- in der Befehlserecgnung zu verwendende Zeichenfolgen werden jetzt farblich hinterlegt
- Anpassungen an spezielle Kalenderausgaben
- es wird nur noch die URL überprüft, da bei bestimmten Googlekalenderlinks kein Username/Passwort benötigt wird

Version: 0.9.3

- Fehlerbehebung in der Googlekalenderverarbeitung

Version: 0.9.4

- komplette Überarbeitung und Austausch der RRULE-Engine für wiederkehrende Termine
- Support für RDATE (verschobene oder zusätzliche Termine in Serienterminen)
- Fehlermeldungen während des Verbindungsaufbaus werden nun angezeigt

Version: 0.9.5

- Bugfixing

Version: 0.9.6

- Bugfixing

Version: 0.9.7

- Performance Optimierung

Version: 1.0.0

- Update für LoxBerry 1.x
- Bugfixing für das Zeitonenhandling
- Nutzung des neuen Formvalidating URL mit Portangabe werden akzeptiert
- Autoupdatefeature hinzugefügt
- LoxBerry Logfeature hinzugefügt
- diese Version ist nicht auf einem LB 0.2.3 lauffähig

Version: 1.0.2

- optimierte RRULE Bearbeitung
- MQTT publishing bei installiertem MQTT-Gateway hinzugefügt

Inhalt

- Konfigurationsoptionen
- Einrichtungshilfe
- Funktion des Plugin
- MQTT-Support
 - MQTT ohne Miniserver
- Einrichtung in der Loxone Config Software
- Einrichtung eines iCloud Kalenders
- Einrichtung eines Nextcloud Kalenders
- Einrichten eines Google Kalenders
- Bekannte Probleme
- Fehler melden

Konfigurationsoptionen

- Depth: Als einzige Einstellung lässt sich das Plugin mit dieser Option an den verwendeten CalDAV-Server anpassen. Einige Server erwarten für die Abfrage eine Abfragetiefe von 0, andere eine 1. Da neuere Server meistens Depth 1 verwenden ist dies als Standard hier eingestellt. Sollte trotz korrekter URL zum Kalender kein Termin gefunden werden, kann die Einstellung 0 hier zum Erfolg führen.

Einrichtungshilfe

Mit der Einrichtungshilfe steht optional ein Tool zur Verfügung, mit dem man sich die URL für den virtuellen HTTP-Eingang zusammenbasteln lassen kann. Wer Schwierigkeiten damit hat sich die URL aus den einzelnen Komponenten zusammensetzen, gibt hier einfach die entsprechenden Daten ein und klickt auf "Angaben testen". Es erscheint daraufhin die benötigte URL. Des Weiteren wird der erzeugte Link auch gleich getestet und die Rückgabe unformatiert ausgegeben. Hier können Teile heraus kopiert und in der Befehlskennung verwendet werden. Noch weiter unten werden einige gängige Befehlskennungen zum direkten Verwenden für die angegebenen Suchbegriffe aufgelistet. Die in der Befehlskennung erforderlichen Zeichenfolgen sind farblich hinterlegt.

Die Bedeutung der einzelnen Parameter wird im Bereich "Funktion des Plugin" beschrieben

Funktion des Plugin

Das Plugin ist aus den bereits verfügbaren Scripts für die CalDAV Anbindung aus dem [Loxforum](#) entstanden. Erwähnt sei hier auch noch einmal [Christian Fenzl](#), der die Kalenderabfrage windowstauglich gemacht hat sowie das Caching des Google Kalenders beigesteuert hat.

Wie funktioniert nun das Ganze?

Das Plugin so konzipiert, dass im Plugin selbst keine Einstellungen vorgenommen werden müssen. Alle Daten, die zum Abfragen des Kalenders benötigt werden, werden in der URL des HTTP-Einganges als Parameter angegeben. So ist es auch möglich verschiedene Kalender auf unterschiedlichen Servern abzufragen.

Solch eine URL sieht in etwa so aus: `http://loxberry/plugins/caldav4lox/caldav.php?calURL=http%3A//localhost/remote.php/caldav/calendars/USER/defaultcalendar&user=USER&pass=PASSWORD&fwdays=5&events=abfuhr`

Die einzelnen Angaben bedeuten folgendes:

- `http://loxberry/plugins/caldav4lox/caldav.php?` - ist die Adresse des Plugin auf dem LoxBerry. Dieser ist unter Umständen anzupassen.

- calURL - ist die URL des entsprechenden Kalenders so wie sie z.B. im Thunderbird/Lightning anzugeben ist. Bei den meisten Kalendern kann man diese URL in den Eigenschaften des Kalenders in Erfahrung bringen. Bei OwnCloud z.B. ist neben dem Kalendernamen auch ein Linksymbol mit dem Hilfetext "CalDAV-Link". Klickt man dieses Symbol an, erscheint unter den Kalendern ein Textfeld mit dem benötigten Link. Es ist aber darauf zu achten, dass hier der Doppelpunkt hinter http(s) in %3A geändert werden muss.
- user - Euer Benutzername für den Kalender
- pass - Euer Passwort für den Kalender
- fwdays - da man keine Möglichkeit hat eine flexible Zeitspanne in die URL einzufügen, habe ich diese Variante gewählt. hier wird die Anzahl der Tage angegeben, für die im Voraus im Kalender nachgesehen werden soll. Vom aktuellen Datum der Anfrage an. Null für nur den aktuellen Tag. Wird der Parameter nicht angegeben, wird 0 angenommen.
- delay (im Beispiel nicht benutzt) - ist die Anzahl der Minuten, die vor und nach der aktuellen Zeit nachgesehen wird. Wird der Parameter nicht angegeben, wird 60 benutzt. (Beispiel: ein Termin 14:30 - 15:00 Uhr im Kalender, wird zwischen 13:30Uhr und 16:00Uhr angezeigt wenn delay nicht angegeben wurde.
- events - eine Liste der Terminnamen im Kalender, nach denen gesucht werden soll mit Pipe/senkrechter Strich getrennt (ich habe keine Leerzeichen/Sonderzeichen getestet, doch ich denke mal, man sollte darauf verzichten). Es wird auch nach Wortteilen gesucht. Ein Termin "Muellabfuhr" wird also auch durch Angabe von "llabf" gefunden.
- cache (im Beispiel nicht benutzt) - ist die Anzahl der Minuten, die eine Google Kalenderabfrage auf dem LoxBerry zwischengespeichert und benutzt wird bis der Google Kalender erneut "befragt" wird.

Hier noch ein paar Infos für die Google Jünger: Eine richtige CalDAV Abfrage mit Username und Passwort ist gegen den Google Kalender nicht so einfach. Die Alternative dazu ist die Verwendung der Privatadresse des Kalenders - siehe [CalDAV-4-Lox#Einrichten eines Google Kalenders](#)

Als Rückgabe erhält der virtuelle HTTP-Eingang Daten wie folgende:

```
{
  "Muellabfuhr": {
    "Start": 194832000,
    "End": 194918400,
    "Description": "gelbe Tonne=1",
    "fwDay": 1,
    "wkDay": 5
  }
  "MSTest": {
    "Start": 194802300,
    "End": 194804100,
    "Description": "",
    "fwDay": 0,
    "wkDay": 4
  }
  "now": 194801634
}
```

Das Ergebnis ist dann im JSON Format. Für jeden Suchbegriff gibt es ein Objekt. Gibt es zu dem Suchbegriff keine aktuellen Termin, so sind die Werte -1 und Description "". Die zurückgegebenen Zeitwerte sind Loxone Zeitwerte und können über <v.u> humanreadable angezeigt werden. Zum Vergleich der Zeit mit der aktuellen Loxone-Zeit muss erst aus den Zeitvariablen des Systems die Loxone-Zeit errechnet werden. Das geht über einen Formelbaustein mit Eingängen AI1 = Tage seit 2009, AI2 = Stunde, AI3 = Minuten und AI3 = Sekunden und der Formel ((I1*24+I2)*60+I3)*60+I4

Description sind die Notizen zum Termin ... hier kann man weitere Infos eingeben. Hier unterscheidet ich z.B. welche Tonne abgeholt wird. fwDay ist der Tag in der Zukunft, an dem der Termin stattfindet. Rufst Du Montags ab und hast fwdays=5 angegeben und der Termin, der gefunden wurde ist am Mittwoch, so ist der Wert 2. wkDay ist der Wochentag des Termins 1=Mo - 7=So. Im vorigen Beispiel wäre das also 3. now ist die aktuelle Zeit zur Antwortzeit. Diese kann für Vergleiche/Berechnungen im Programm verwendet werden.

MQTT-Support

Mit Version 1.0.2 wurde der MQTT-Support eingefügt. Die MQTT Funktionalität bezieht sich in diesem Fall auf das Publizieren von Nachrichten. Für diese Funktion ist das installierte MQTT-Gateway Plugin von [Christian Fenzl](#) notwendig und wird dann auch automatisch verwendet. Der Vorteil dieser Variante liegt darin, dass es auf diesem Weg z.B. einfach möglich wird, Texte auf den MiniServer zu übertragen.

Die verwendeten Topics folgen einem Schema. Das Topic beginnt mit caldav4lox/events/. Es folgt der übergebene Suchbegriff und daran anschließend dann sämtliche Werte, die auch im JSON übergeben werden. Wenn also als gesuchtes Event „Test“ angegeben wird, dann wäre das Topic c aldav4lox/events/Test/. Wenn man die Description dann im Miniserver erhalten will, würde man caldav4lox/events/Test/description verwenden.

Einrichtung:

1. MQTT Gateway

- a. MQTT Gateway installieren
- b. Im MQTT Gateway eine Subscription anlegen: `caldav4lox/events/#`
- 2. CalDAV-Plugin
 - a. Im CalDAV-Plugin die URL(s) zusammenstellen (lassen)
 - b. Mit dem Webbrowser die URL testen
- 3. Loxone Config
 - a. Für den Abruf des Kalenders beim CalDAV-Plugin einen "Virtuellen Ausgang" erstellen
 - b. Virtuelle Ausgangs-Befehle erzeugen, die die URL bzw. bei mehreren Suchen die URLs abrufen
 - c. Die Virtuellen Ausgangsbefehle auf eine Seite ziehen und einen Trigger anschließen (z.B. Stundenimpuls)
 - d. Im MQTT Gateway die "Incoming Overview" öffnen. Dort werden die Daten angezeigt.
 - e. Entsprechend [MQTT - Schritt für Schritt: MQTT -> Loxone](#) virtuelle Eingänge für die Daten von CalDAV anlegen

Bitte beachten, dass das MQTT Gateway identische Daten nicht mehrfach an den Miniserver sendet (Cache).

MQTT ohne Miniserver

Wenn der Trigger nicht vom Miniserver ausgelöst werden soll oder kann, kann auch ein Cronjob oder ein anderes Tool die URL des Plugins abrufen, und somit den Kalender über CalDAV4Lox „befragen“. Das Plugin übergibt die Informationen an MQTT und das MQTT-Gateway pushed die Daten (wie oben beschrieben) an den Miniserver. Außerdem können die Kalenderdaten auch von anderen MQTT-fähigen Geräten genutzt werden, ohne dass überhaupt ein Miniserver verwendet wird.

Einrichtung in der Loxone Config Software

Das Plugin wurde für die Verwendung mit dem virtuellen HTTP-Eingang entwickelt. Detaillierte Informationen hierzu gibt es in der [Loxone Dokumentation](#).

Einrichtung eines iCloud Kalenders

Zuerst müsst ihr diese Anleitung abarbeiten: <https://community.openhab.org/t/solved-apple-icloud-caldav-connection/32510>
(Arbeitszeit ca. 15 Minuten)

Apple arbeitet auch mit **Depth:1**.

Dann könnt ihr euren Link zu dem Kalender **URL des Kalenders**: eintragen. Im Anschluss gebt ihr noch euren Benutzernamen und als Passwort das bei Apple generierte Passwort ein - **nicht euer Apple-ID Passwort!**

Depth: 

Einstellungshilfe

Die folgenden Einstellungen erstellen die im virtuellen HTTP-Eingang nötige URL. Des Weiteren kann von hier aus die Kalenderabfrage und Rückgabe getestet werden.

URL des Kalenders:	<input type="text" value="https://p42-caldav.icloud.com:443"/>
Kalender-User:	<input type="text" value=""/> @ <input type="text" value="com"/>
Passwort des Kalender-Users:	<input type="password"/>
Tage in der Zukunft:	<input type="text" value="7"/>
Minuten vor und nach:	<input type="text"/>
Google Cachedauer:	<input type="text" value="1400"/>
Suchbegriffe	<input type="text" value="Wertstoff Biotonne grau Papiertonne"/>
<input type="button" value="Angaben testen"/>	
<input type="button" value="Log anzeigen"/>	

Einrichtung eines Nextcloud Kalenders

Meldet euch in der Nextcloud an und öffnet euren Kalender. In der Kalender APP könnt ihr den Kalender über "..." einen Link erzeugen.

Eigenschaften (Virtueller HTTP Eingang)	
Eigenschaft	Wert
☐ Allgemein	
Bezeichnung	NextcloudAbfall
Beschreibung	ASP im Nextcloud KAlender
Objekttyp	Virtueller HTTP Eingang
☐ Einstellungen	
URL	http:// _____ /plugins/caldav4lox/caldav.php?calURL=https%3A/ _____ /remote.php/dav/calendars. _____
Abfragezyklus [s]	10
Timeout [ms]	4000
Anzahl erlaubte Timeouts	10

Den Abfragezyklus könnt ihr auf 86400 (ein Tag) einstellen.

Einrichten eines Google Kalenders

Depth: 1

Auf <https://calendar.google.com/> zu den Einstellungen des Kalenders gehen (oben Zahnrad / "Einstellungen" / "Einstellungen für meine Kalender" / Kalender auswählen), dort dann unter "Kalender integrieren" die "Privatadresse im iCal Format" kopieren und im Feld URL des Kalenders eingeben

User: und Passwort: leer lassen

Der Standard Timeout von 4000ms kann bei Google Kalender u.U. nicht reichen. Es ist besser ihn auf den Maximalwert von 8000 zu setzen.

Bekannte Probleme

Aktuelle Kalendertools benutzen beim Verschieben eines Termins aus einer Terminreihe gern mal die Variante einen neuen Termin dafür anzulegen und dazu anzugeben welcher Termin aus welcher Reihe damit ersetzt wird (RECURRENCE-ID, UID, SEQUENCE). Diese Variante wird nicht unterstützt. Beim Verschieben eines Termins sollte dieser Einzeltermin gelöscht und dafür ein neuer Einzeltermin angelegt werden.

PHP führt manchmal zu einem SSL Context Creation Failure Neustart des Loxberrys hilft

Fehler melden



Fehlerberichte bitte direkt im [Loxforum](#) melden.