CD 18XX/WD : 2018

Guidelines to thwart calendar abuse for calendaring and mail system operators

THE CALENDARING AND SCHEDULING CONSORTIUM

TC CALSPAM

CALCONNECT STANDARD

WORKING DRAFT

WARNING FOR DRAFTS

This document is not a CalConnect Standard. It is distributed for review and comment, and is subject to change without notice and may not be referred to as a Standard. Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

The Calendaring and Scheduling Consortium, Inc.
4390 Chaffin Lane
McKinleyville
California 95519
United States of America

copyright@calconnect.org
www.calconnect.org

CONTENTS

FOREWORD

The Calendaring and Scheduling Consortium ("CalConnect") is global non-profit organization with the aim of facilitating interoperability of technologies across user-centric systems and applications.

CalConnect works closely with liaison partners including international organizations such as ISO, OASIS and M3AAWG.

The procedures used to develop this document and those intended for its further maintenance are described in the CalConnect Directives and in this case also aligned with the procedures used at M3AAWG.

In particular the different approval criteria needed for the different types of CalConnect documents should be noted. This document was drafted in accordance with the editorial rules of the CalConnect Directives and M3AAWG.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CalConnect shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the CalConnect list of patent declarations received (see www.calconnect.com/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

This document was prepared by Technical Committee *CALSPAM*.

INTRODUCTION

Definition

"Calendar spam", unsolicited or otherwise unwanted calendar events and meeting invitations, is a recently exploited channel for abuse aimed at users of calendaring & scheduling systems. It is a new form of application-specific spam which takes advantage of the application layer across multiple technologies that spans scheduling, calendaring and messaging systems. As is the case with email spam, calendar spam is not only used to deliver unwanted information, but can also be used for malicious purposes such as phishing attempts and delivering dangerous payloads.

As calendar events and meeting invitations are often (but not exclusively) transported and delivered via email, combatting calendar spam requires awareness, intervention and integration with email systems and services.

Impact

Calendar spam is unique in a number of ways:

a. Calendar spam, unlike email, can be placed chronologically anywhere in calendars, in the past or the future, not just the present, making it difficult for the end-user to detect at the time of delivery.
b. Spam meeting invitations, may automatically see these unwanted invitations added to their calendar without their consent, with notifications sent to all their devices. These invitations are not only difficult to find, but in some cases there is no way for the user to remove these events short of deleting the entire calendar.
c. Calendar events and meeting invitations do not yet carry the rich provenance which today accompanies email (detailed header information), making it difficult to ascertain where and when events originated and were delivered.
d. Calendar events often contain notifications/alarms which are propagated across a user's desktop and mobile calendaring clients. It is common for users to have multiple calendaring clients which exacerbates the abuse.
e. Calendar events can include recurrence meaning that one event can show up in the user's calendar multiple times with multiple notifications/alarms being triggered over time.

1. SCOPE

This document specifies guidelines for calendaring and mail system operators to:

- detect the occurrence of calendar abuse;
- consider processes and procedures to mitigate calendar abuse; and
- suggest acceptable (non-abusive) practices with calendar usage.

2. NORMATIVE REFERENCES

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IETF RFC 6047, iCalendar Message-Based Interoperability Protocol (iMIP)

IETF RFC 5546, iCalendar Transport-Independent Interoperability Protocol (iTIP)

3. TERMS, DEFINITIONS, SYMBOLS AND ABBREVIATED TERMS

For the purposes of this document, the following terms and definitions apply.

3.1. Terms and definitions

3.1.1
calendar system

an information system that provides the calendar and scheduling functionality for end-user accounts. This includes creating, editing and deleting events as well as scheduling events between different user accounts including user accounts from other calendaring systems.

The *calendar system* should apply state-of-the-art methods to prevent spam being sent from and received by user accounts on their system.

> Note 1 to entry: The term *calendar system* in this standard specifically refers to calendaring systems that fulfill the requirements of CalConnect calendaring standards.

3.1.2
calendar spam

unsolicited calendar events and meeting invitations delivered through calendaring systems

3.1.3
calendar abuse

malicious usage of a *calendar system* (Clause 3.1.1), possibly leading to an *attack* (ISO/IEC 27000:2018, Clause 3.2) on the receiving user

3.1.4
attack

attempt to destroy, expose, alter, disable, steal or gain unauthorized access to or make unauthorized use of an asset

[SOURCE: ISO/IEC 27000:2018]

3.1.5
mail system

an information system that provides mail functionality. The most used method to send calendar invites between users is *iMIP* (RFC 6047), a way of exchanging iTIP (RFC 5546) messages using email. Therefor *mail systems* play a vital role in connecting users from different providers by creating events and inviting other users to join. Mail systems are used to transport the calendar relevant information from organizers to attendees of events.

The *mail system* should apply state-of-the-art methods to prevent spam being sent by and received from user accounts on their system.

3.1.6
user system

an information system that provides authentication and autorization functionality. The *user system* should prevent fake, bot or spam registrations in order to limit the number of user accounts on their system, that can later be used for creating spam (either mail or calendar).

The *user system* should also prevent real user accounts being compromised by malicious actors by providing state-of-the-art authentication methods e.g. two-factor-authentication.

3.1.7
spam

unsolicited or unwanted information

3.2. Symbols and abbreviated terms

For the purposes of this document, the following abbreviations apply.

3.2.1
iMIP

iCalendar Message-Based Interoperability Protocol (iMIP)

[SOURCE: RFC 6047]

3.2.2
iTIP

iCalendar Transport-Independent Interoperability Protocol (iTIP)

[SOURCE: RFC 5546]

3.2.3
SMTP

Simple Mail Transfer Protocol

[SOURCE: RFC 2821]

4. BEST COMMON PRACTICES FOR MITIGATING CALENDAR SPAM

4.1. Detecting and mitigating spam on source system

User accounts could be compromised by malicious actors or free hosting providers could be abused with bots signing up for free accounts. These accounts are then used to create calendar spam events. The calendar system uses templating to send an email invitation with the calendar event attached and the event content will also be inserted into body of the email. The "source" hosting provider should take steps to detect and mitigate this internal abuse on the calendar system and the email system.

4.1.1. using calendar systems

There are different best common practices that can be applied here:

a. abuse detection on front-end usage using input as network/IPs, user agents, click rate / path
b. checking the event content (subject, description, recurrence, number of attendees, links, …) for typical spam patterns before creating the event and sending the email invitations

There are many potential actions that could be invoked if potential spam is detected (e.g. not sending, display frontend error or feedback, alert user account, apply rate limiting, demand solving captcha before sending, block the account altogether and more.)

4.1.2. using smtp

There are different best common practices that can be applied here:

a. abuse detection for SMTP access using input as network patterns, DNSBL checks against the client IP, …
b. check email for spam content patterns (malicious content, blacklisted/known phishing URLs, …) using standard email anti-spam scanning applications

There are many potential actions that could be invoked if potential spam is detected (e.g. bounce the message, discard the message, InfoSec data sharing with receiving email / calendar providers, and more.)

4.2. detecting and mitigating spam on receiving system

Spam events are typically received by recipients in two ways:

a. via email from an external system, or
b. directly from another account (bot or compromised) within the *calendar system*

Events from internal accounts may propagate natively within the *calendar system* or they may propagate over email, depending on implementation. The "receiving" hosting provider can take steps to detect and mitigate the "external" abuse on the *calendar system* and the *mail system*.

4.2.1. mail system

There are different best common practices that can be applied here:

a. abuse detection for receiving email including input as network, mail header/ structure, …
b. check email for spam content patterns using standard email anti-spam scanning applications, DNSBLs, URIBLs, etc.
c. check sender From address reputation using internal and external sources e.g. subscribe to InfoSec feeds of known malicious addresses, organiser on whitelist, …

There are many potential actions that could be invoked if potential spam is detected (e.g. bounce the message, discard the message, put the message in quarantaine or spam folder.) Interaction (e.g. adding the event to the end-user’s calendar) with the *calendar system* should not be initiated in these instances.

As some of these actions do not deliver the email to the user and no interaction with the *calendar system* occurs, the recipient has no way to handle false positives. Therefore these actions can only be taken if the *mail system* is very certain about this being abuse or spam.

For some of the milder actions (e.g. putting in spam folder) the user should be offered options. For example, allow these emails to be marked as false positives and offer the client option to manually insert the events into the user’s calendar.

4.2.2. interaction between *mail system* and *calendar system*

Interaction between the *mail system* and the *calendar system* should follow these best common practices:

a. the interaction should only be triggered for emails not already identified as spam during applying the above mentioned best common practices for mail systems
b. the events should be parsed by the *calendar system* due to the domain knowledge regarding calendar structure not present or mature in most types of *mail system*
c. the event content should be checked for spam patterns (subject, description, recurrence, links, …) to determine the likeliness being *spam*
d. depending on the likeliness being *spam*, spam handling options should be offered in the users settings for insert (e.g. only automatic insert for organizers on a whitelist / personal address book, state of this event in availability of calendar (e.g. free, conditional or blocked))

There are many potential actions that could be invoked if potential spam detected)e.g. not automatically inserting, deactivated notifications, ...)

4.2.3. *calendar system*

Besides inserting or not inserting the received events into the user calendar during the interaction between *mail system* and *calendar system*, the *calendarsystem* should offer these best common practices:

a. offer the end-user a delete option for unwanted events (e.g. mark as spam in the client) in order to give the user the option for deleting the unwanted events without notifying the organizer.
b. consider sending ARF reports for calendar abuse reporting
c. store information about how an event was inserted into the users calendar (e.g. Message-ID) in order to be able to inform the user about this contextual information and to provide additional information to the sending system about the abuse

There are many potential actions that could be invoked if spam is detected by the user e.g. sending Feedback loop if MailID and original email is still available in the *mail system*.

4.3. other ways calendar *spam* can occur

4.3.1. subscribing to shared calendars containing malicious events

Another way how malicious events can end up in users calendar are shared calendars being manipulated on origin side. Popular calendars e.g. official vacation/bank holidays in countries or states or schedules of popular sports clubs could be target for phishing / taken over by spammers.

Single malicious events within these subscribed calendars can not be deleted if shared read-only. More robust controls may be needed for calendar subscribers, but unsubscribing the specific calendar can solve the problem on an all or nothing approach (also the wanted events are then unsubscribed and deleted from users calendar).

4.3.2. *iTIP*

Calendar systems using *iTIP* for direct communication between each other e.g. within the same *calendar system* also need to consider and implement anti-abuse options as mentioned above.

5. CONCLUSION

Spam is a well know problem regarding to mail. As calendaring uses mail as main communication protocol, calendar spam is one of the logical next steps from spammers trying to extend their impact and attack vectors. Knowledge of both domains is needed to understand the interaction and new possibilities coming with this thread.

Service providers working together in the field of mail and calendaring security exchange their general views, but will always have specific systems and workflows handling the thread depending on their architecture and infrastructure. Therefor the aim of this document is to provide a general overview about what is needed to fight calendar spam, but does not suggest concrete implementations.

There is no doubt, that this topic will grow over time and preparing the systems and workflows of doing so is needed. It’s all about understanding and preventing the thread and if we are successful in doing so, nobody will ever recognize.

6. ACKNOWLEDGEMENTS

The editor of this document wishes to thank all people engaged at CalConnect - the Calendaring and Scheduling Consortium and attendees of the M3AAWG conference sessions about the topic, as well as the following individuals who have participated in the drafting, review, and discussion of this document:

Arne Allisat, Bron Gondwana, Andrew Laurence, Andrey Maevsky, Dave Thewlis, Jesse Thompson and Ronald Tse.

Author

Thomas Schäfer (editor)

1&1 Mail&Media Development and Technology GmbH

APPENDIX A (INFORMATIVE) TECHNICAL INFORMATION

A.1. Structure of a best practice iMIP message containing an event

Email messages may have more than a single iCalendar file attached, but the best practice is to only attach a single iMIP (RFC 6047) file to each email.

The recommended structure the email is as follows:

```
multipart/mixed
   multipart/alternative
       text/plain
       text/html
   text/calendar; method=REQUEST
   application/ics (with a content-disposition:attachment) BASE64
```

This structure is based on interoperability testing with various existing implementations. Some clients will only see the part with the standard `text/calendar` content-type and the method header. Other clients are only able to attached parts with `application/ics` (which is non-standard).

It is also recommended that the filename of the `application/ics` part ends with the `.ics` file extension.

Some vendors add links within the HTML part which can be used from non-calendaring-aware email clients to accept or decline a request without having to process the calendar parts at all. The server just updates the ORGANIZER's copy of the event based on the link clicked.

When using standard conform *calendar systems* the structure of the email will be like above and the `text/plain` and `text/html` part of the message in the body will also include information of the event e.g. subject, description, … . This does not prevent spammers from not including this potential malicious content besides the attached files, so all parts need to be parsed to detect malicious content in events.

BIBLIOGRAPHY

[1] ISO/IEC 27000:2018, *Information technology -- Security techniques -- Information security management systems -- Overview and vocabulary*

[2] IETF RFC 2821, Simple Mail Transfer Protocol