Internet Engineering Task Force
INTERNET DRAFT
draft-ietf-calsch-crisp-01.txt

J. Stracke
eCal

August 2001
Expires: February 2002

CAP Realtime iTIP-based Scheduling Profile (CRISP)

1. Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts. Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other docu¡ments at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Distribution of this document is unlimited. Please send comments to francis@ecal.com or to the ietf-calendar@imc.org discussion list (subscription address ietf-calendar-request@imc.org; "SUBSCRIBE" or "UNSUBSCRIBE" in the body).

2. Copyright Notice

3. Abstract

This document sets forth a restricted profile of [CAP], one which supports no operations beyond the scheduling functionality of [iTIP]. The motivation is to permit use of CAP's real-time iTIP functionality without exposing the calendar access functionality (which may require stricter security controls than iTIP).

4. Introduction

[iTIP] defines a scheduling protocol based on exchanging specially formatted [iCalendar] messages. iTIP is defined to be independent of transport protocol. At present, there is one standard binding of iTIP to a transport protocol, [iMIP], which carries iTIP messages in email. This is a useful base level capability (email can reach vir¡ tually any user on the Net), but can involve considerable latencies. A real-time binding for iTIP would be useful; it would permit appli¡ cation developers to give users better feedback on the progress of the iTIP operations.

Since CAP includes full iTIP functionality, one option would be to permit full access to CAP; to schedule an event with a remote user, one would then make a CAP connection to their CS. The problem is that such a connection may be considered a security risk in some organizations; even though the CS has ACLs to prevent the client from performing non-iTIP operations, it would be better if the client sim¡ ply could not attempt such operations. (It's as if mail administra¡ tors were told that an SMTP server outside the firewall had to include IMAP functionality as well.) Thus, this document defines CRISP, a profile of CAP, a subset which does not support non-iTIP operations.

This document does not specify the relationship between a CRISP server and a (full-powered) CAP server. They may be implemented together, with the CRISP server being nothing more than the CAP server responding in CRISP mode (e.g., based on source IP address); the CRISP server may act as a proxy for the CAP server (see Firewall Application, below); the two servers may feed into the same database, but not know about each other; or there may be no CAP server, only the CRISP server, used for interdomain scheduling, but not for calen¡ dar access. Or, of course, there may be other modes of operation. These are implementation details, which do not need to be included in a protocol spec.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

Note that CRISP is a replacement for the former proposal known as iRIP (no reference is available, because the Internet-Draft has long since expired), which was abandoned when it was realized that the functionality of CAP was a superset of the functionality of iRIP.

5. Profile Definition

A CRISP server is a CAP server with the following capabilities:

* ITIPVERSION=1.0
* CAPVERSION=1.0
* CAR=NONE
* QUERYLEVEL=NONE

In addition, various AUTH capabilities are expected. Anonymous authentication SHOULD be supported, since the point of CRISP is to permit iTIP communication across security domains. However, other authentication mechanisms may make sense in some cases; for example, if CRISP is being used for scheduling between cooperating companies (that is, in an extranet), then one company's CRISP server might be able to authenticate users from the other company.

Other capabilities which apply to iTIP operations MAY be specified; e.g., MAXDATE and MAXICALOBJECTSIZE.

Note that NONE is not a legal value for CAR or QUERYLEVEL in the cur¡ rent draft of CAP. This will have to be resolved.

A CRISP server MUST NOT accept any iCalendar component which is not a valid iTIP component.

In effect, the statement that a server is CRISP is a statement about the server's current advertised capabilities. It is conceivable that a CAP server might be CRISP under some conditions and not others. For example, the server might offer a CRISP capability set on initial connection, but upgrade to full CAP if the client uses STARTTLS and provides an appropriate certificate. It's not clear, though, whether there's any good way to advertise this fact. For the rest of this document, we will assume that a CRISP server is always CRISP.

6. Possible Firewall Application

This section is non-normative.

Clearly, it would be undesirable for an organization with a CAP server to have a CRISP server implemented completely separately, but having access to the same database. Such duplication would increase development costs, maintenance costs, and security exposure. On the other hand, it would be possible to build a CRISP server which han¡ dles all operations by proxying them to the CAP server. Such a proxy could be placed within the "no-man's-land" common in firewalls; the firewall would permit CAP connections from the outside to the proxy,

and from the proxy to the internal CAP server. The proxy would review all incoming iCalendar components and validate that they were legitimate iTIP operations; no non-iTIP components would be forwarded to the CAP server. Similarly, if necessary, the proxy might censor the iTIP replies coming from the CAP server.

Naturally, this is not the only approach possible; this section is merely illustrative. The CRISP client does not know or care how the CRISP server gets at the underlying calendar store.

7. Security Considerations

CRISP is a subset of [CAP], and accordingly inherits all of CAP's security analysis. However, new analysis does need to be done for the subset, especially since the whole point of the subset is to address security concerns.

8. Author's Address:

John Stracke
Chief Scientist
eCal Corp.
Email: francis@ecal.com

9. References

[iTIP] Silverberg, Mansour, Dawson, Hopson, "iCalendar Transport-Independent Interoperability Protocol (iTIP)", RFC 2446, November 1998

[iMIP] Dawson, Mansour, Silverberg, "iCalendar Message-Based Interop¡erability Protocol (iMIP)", RFC 2445, November 1998

[CAP] Mansour, Dawson, Royer, Taler, Hill, "Calendar Access Protocol (CAP)", draft-ietf-calsch-cap-05.txt, July 2001. Work in progress.

[iCAL] Dawson, Stenerson, "Internet Calendaring and Scheduling Core Object Specification (iCalendar)", RFC 2445, November 1998