



Event Notification in SIP

SUBSCRIBE and NOTIFY and an example service

Adam Roach
Ericsson Inc.
adam.roach@ericsson.com

Motivation

<draft-roach-sip-subscribe-notify-00.txt>

- SUBSCRIBE and NOTIFY methods have been mentioned in several contexts since at *least* late 1998
- If not defined in a central draft designed for extensibility, we risk losing the utility of these methods to a very narrow scope
- The intention of this draft is to begin discussion of what needs to be done to formalize these methods in earnest



Arguments for the utility of SUBSCRIBE and NOTIFY

“Why do we keep reinventing new specialized methods like PING and INCALL and PRESENCE and so on... all these functions can be handled by simple, reusable message primitives.

“The proposed SUBSCRIBE and NOTIFY methods... could be used to support presence, messaging, state change detection, etc. Very powerful idea.”

-Dean Willis, SIP WG Chair, Feb 1999

Details of the draft

- Adds two new methods: SUBSCRIBE and NOTIFY
- Adds a new "Event" header
- Act like BYE for call-member subscriptions
- Act like OPTIONS for third-party subscriptions

Call-related subscriptions

- May be requested by a party with whom a session is currently established or by a third party
- Example events include “call-terminated” and “dtmf”
- Correlated to correct call by Call-ID

Resource related subscriptions

- Requested by a third party
- Call-ID does not correspond to any INVITE-initiated session
- Example events include “terminal-free,” “user-login,” and “voicemail-waiting”

Notification

- Always share same remote-URI, local-URI, and Call-ID as SUBSCRIBE to which they correspond (same correlation as a session).
- Contain a single event in the Event header
- May contain other event-related parameters (as new headers or message body)



Subscription Duration

- Handling of subscription duration and expiration is identical to that of REGISTER, including cancellation of subscriptions
- Call-related events, of course, expire at the end of the call



Getting along with the neighbors

- Efforts taken to not disrupt the PINT-defined SUBSCRIBE and NOTIFY: if no “Event” header is present, clients can assume “all events” (as in PINT)
- Client can trivially support both drafts simultaneously
- May make sense to combine event notification efforts into a single, merged draft

Example Service: "Auto-Redial"

<draft-roach-sip-acb-00.txt>

- Terminal requesting service subscribes to "terminal-free" event
- When the called party's terminal has no ongoing calls, calling party receives notification
- New INVITE is then issued to start a session
- Chosen as example because existing SIP methods of providing this service are sub-optimal
- Contains very detailed callflows of SUBSCRIBE and NOTIFY in action

Open Questions

- Should the base SUBSCRIBE/NOTIFY draft go further in defining a base set of events?
- Is there sufficient interest in the types of services these methods provide to proceed with this work?
- Should the draft be expanded to allow subscriptions to events at proxies and other non-user-agent nodes?
- How should we proceed with coordination with PINT?
- How should expirations be handled for multiple events requested in a single SUBSCRIBE message?