Use Cases for Session Initiation Protocol (SIP) Caller Preferences

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 documents 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

To view the list Internet-Draft Shadow Directories, see http://www.ietf.org/shadow.html.

Abstract

This document describes a set of use cases for the SIP Caller Preferences and Callee Capabilities extension. Each use case is presented as a desired routing decision, followed by the actual headers and processing logic which would result in that decision.
1 Introduction

The SIP extension for Caller Preferences and Callee Capabilities [1] describes mechanisms that allow a UA to register its capabilities in a REGISTER request. A caller can express preferences, either explicitly or implicitly, about how that request is to be handled. This is accomplished with the Accept-Contact and Reject-Contact header fields.

The purpose of this document is to demonstrate the ability of these basic primitives to meet the needs of many different routing problems. To do that, it presents a set of use cases. Each use case is described as a situation along with a desired routing goal. Then, it demonstrates how the various caller preferences headers and the proxy processing logic would result in the appropriate decision being made.

Demonstrating the ability of caller preferences to meet the needs of these routing problems serves two higher level goals. The first is to validate that the caller preferences specification is complete, and capable of solving real requirements. Since the caller preferences specification pre-dates the SIP change process [2], no requirements work was ever done for this extension. To some degree, this document "backfills" requirements. However, this is not an academic exercise only, since the use cases described here did result in changes in the specification in later versions.

The second goal of the usage cases is to motivate support for the draft. The caller preferences specification has received scant attention from the working group beyond the authors of this document. This is likely because people are not aware of the scope of problems which this extension can solve.

2 Routing of INVITE and MESSAGE to different UA

2.1 Desired Behavior

Address of Record (AOR) Y has two contacts Y1 and Y2. Y1 is a phone, and supports the standard operations INVITE, ACK, OPTIONS, BYE, and so on, while Y2 is a pager and supports only OPTIONS and MESSAGE. Caller X wants to send pages to Y. There is a lot of traffic in the network of both calls and pages, so there is a goal not to unnecessarily fork messages to devices that can’t support them. So, ensure that INVITEs of Y are delivered only to Y1, while MESSAGEs to Y are delivered only to Y2.

2.2 Solution
Y1 will create a registration which looks like, in part:

REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: <sip:Y1@pc.example.com>;methods="INVITE,ACK,OPTIONS,BYE,CANCEL"

Y2 will create a registration which looks like, in part:

REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: <sip:Y2@pc.example.com>;methods="OPTIONS,MESSAGE"

When a UAC sends an INVITE, it will arrive at the proxy for example.com. There are no caller preferences in the request. However, per Section 7.2.2 of [1], the proxy will construct an implicit require-tagged Accept-Contact preference that looks like:

(& (methods="INVITE"))

Applying the matching algorithm of RFC 2533 [3] to this feature set and those registered by Y1 and Y2, the feature set of Y1 alone matches. Thus, Y2 is discarded and the INVITE is routed to Y1.

If the request was MESSAGE, the proxy constructs an implicit require-tagged Accept-Contact preference that looks like:

(& (methods="MESSAGE"))

which matches the feature set of Y2, but not Y1. Thus, Y1 is discarded, and the request is routed to Y2.

3 Single Contact Not Matching Implicit Preferences

3.1 Desired Behavior

AOR Y has a single contact Y1. Its a phone, and therefore supports
the INVITE, BYE, OPTIONS, CANCEL and ACK methods, but not MESSAGE. A caller X sends a MESSAGE request. The desired behavior is that the request is still routed to the solitary contact so that it can generate a 405 response.

### 3.2 Solution

The single contact Y1 will generate a registration which looks like, in part:

```
REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: <sip:Y1@pc.example.com>;methods="INVITE,ACK,OPTIONS,BYE,CANCEL"
```

X sends a MESSAGE request. This results in an implicit require-tagged Accept-Contact preference:

```
(& (methods="MESSAGE"))
```

Since Y1 doesn’t match, it is discarded. However, according to the specifications, if there are no matching targets, the preference operation is re-run without implicit preferences. Since there were no explicit preferences, the request is routed without the caller preferences processing at all, and is sent to the single contact Y1 as desired.

### 4 Package-Based Routing

#### 4.1 Desired Behavior

AOR Y has a number of contacts, Y1, Y2, ..., Yn that can each support normal calls - INVITE, ACK, BYE, etc., and that can also support SUBSCRIBE for the "dialog" event package [4]. Y also has another contact YP that is a presence agent [5] - it can accept SUBSCRIBE requests for the "presence" event package. The goal is for subscribe requests for presence to be routed to YP while invites and subscribes for the dialog package are forked to Y1...Yn.

#### 4.2 Solution

Y1..Yn will generate REGISTER requests which look like, in part:
and Yp will generate a REGISTER request which looks like, in part:

REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: <sip:Yp@pc.example.com>;methods="SUBSCRIBE";events="presence"

A SUBSCRIBE request for presence will arrive at the proxy for example.com. It constructs an implicit require-tagged Accept-Contact preference from the request:

(& (methods="SUBSCRIBE") (events="presence"))

This feature set only matches the one registered by Yp. Therefore, Y1..Yn are eliminated, and the request is properly routed to the PA. Note that, had Y1..Yn not registered the event packages they support, the proxy would have preferentially routed the request to YP, and if that had failed, then forked to Y1..Yn. That is because of the greater score associated with YP.

INVITE requests result in an implicit require-tagged Accept-Contact preference:

(& (methods="INVITE"))

The implicit Accept-Contact feature set matches Y1..Yn, but not Yp. Thus, Yp is discarded and the call is delivered to Y1..Yn only.

A SUBSCRIBE for the dialog event package will result in an implicit require-tagged Accept-Contact preference:
This only matches Y1..Yn, so Yp is discarded, and the request is routed to the remaining contacts.

5 Audio/Video vs. Audio Only

5.1 Desired Behavior

X sends an invitation to Y to initiate an audio/video call, including both m=audio and m=video lines in the SDP. AOR Y has two contacts, Y1 and Y2. Y1 represents a normal audio phone which is preferred for normal audio calls. It will answer an audio/video call, refusing the video. Y2 represents an audio/video phone that should only used when needed. The caller really wants the called answered by a device that supports video.

5.2 Solution

Y1 will generate a registration which looks like, in part:

REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: <sip:Y1@pc.example.com>;audio;q=1.0

Y2 will generate a registration which looks like, in part:

REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: <sip:Y2@pc.example.com>;audio;video;q=0.2

Note the different q-values, allowing Y2 to be selected as a device of "last resort". Of course, both Y1 and Y2 need to be configured to express these two q-values.

To ensure that the call is preferentially routed to a device that supports video, the caller X sends an INVITE that looks like, in part:
The proxy will convert this to a feature predicate and also compute an implicit preference to support INVITE. Since neither contact indicated anything about methods, the implicit preference matches both and has no effect on the relative q-values.

The explicit feature set from the Accept-Contact matches Y2 and Y1. However, the score for Y2 is 1, and 0 for Y1. The result is that the audio/video phone (Y2) will receive the call first.

6 Audio/Video vs. Audio Only: A/V UA Fails

6.1 Desired Behavior

This case is identical to that of Section 5. However, for some reason the Audio/Video UA is not available (for example, its in use or offline). The desired behavior is for an audio-only call.

6.2 Solution

This will happen. Since Y1 didnt say anything about video, it still matched the caller preference, just not explicitly. The caller preference did not include the explicit tag, so Y1 remains as a viable contact, but with a lower q value than Y2. Once the request to Y2 fails, the proxy will try Y1.

7 Forcing Audio/Video

7.1 Desired Behavior

This case is similar to that of Section 5. However, X requires an audio/video call, and would like the call to fail if this is not possible rather than succeeding with only audio.

7.2 Solution

The solution is similar to that of Section 5, however the Accept-Contact header field now includes the explicit and require tags, guaranteeing that the call is never established to any UA that had not explicitly indicated support for video:

```
INVITE sip:Y@example.com SIP/2.0
Accept-Contact: *;video;require;explicit
```
This arrives at the example.com proxy. This explicit feature set matches the feature set for Y2 and Y1. However, the match for Y1 did not have a score of 1. Since the explicit and require tags are present, the contact is discarded.

8 Third Party Call Control - Forcing Media

8.1 Desired Behavior

Z is a third party call control controller [6] trying to establish an audio/video call from X to Y. (Both X and Y are configured the way Y is in 5) Z needs to send a offerless invite to X and use the resulting offer to send an invite to Y. When sending the offerless invite to X it must ensure that an audio/video contact (X2) is chosen over an audio only contact (X1).

8.2 Solution

Z would include, in its INVITE, an Accept-Contact header field:

```
INVITE sip:X@example.com SIP/2.0
Accept-Contact: *;audio;video;require;explicit
```

This caller preference matches both X1 and X2. However, it matches X1 with a score of .5 and X2 with a score of 1. Because of the require and explicit tags, X1 is discarded despite the callee’s preference for it. Thus, the call is routed to X2.

9 Maximizing Media Overlaps

9.1 Desired Behavior

AOR Y has two contacts, Y1 that is a regular audio phone, and Y2 that is a PC capable of supporting both audio and session oriented IM [7]. X is a PC with capability to support audio, video and session oriented IM. X calls Y for the purpose of establishing a voice call. However, X wishes to connect to the device which has the maximal overlap with its media capabilities, in order to maximize the functionality available to the caller.

9.2 Solution

Y1 will generate a registration which looks like, in part:
Y2 will generate a registration which looks like, in part:

REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: <sip:Y2@pc.example.com>;audio;message

The solution requires the caller to support caller preferences. They would include, in their INVITE, an Accept-Contact that lists all the media types they support. In this case:

INVITE sip:Y@example.com SIP/2.0
Accept-Contact: *;audio;video;message

This would prefer a UA that supports all of the same types. Both Y1 and Y2 match. However, Y1 matches with a score of .33, and Y2 matches with a score of .66. This would result in the call being routed to Y2 first.

10 Multilingual Lines

10.1 Desired Behavior

AOR Y represents a shared line in an office. Several employees in the office have phones registered for Y. Some of the employees speak only English, some speak Spanish fluently and have some limited capability for English, and some speak both English and Spanish fluently. Calls from callers that speak only English should be parallel forded to all office workers that speak fluent English. If the call isn’t picked up, then the phones of workers that speak English marginally should be rung. Calls from callers that speak only Spanish should be forked only to workers that speak Spanish.

10.2 Solution

A user at phone Y1 that speaks English only would generate a REGISTER which looks like, in part:
A user at a phone Y2 that speaks Spanish and a little bit of English would generate a REGISTER that looks like, in part:

```
REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: sip:Y2@pc.example.com;languages="es"
Contact: <sip:Y2-en@pc.example.com>;languages="en";q=0.2
```

Effectively, Y2 has registered two contacts. Both of them route to the same device (pc.example.com), but they differ in their language support and relative q-values. Multiple contacts are needed whenever a UA wishes to express differing preferences for being reached for different feature collections.

A user at phone Y3 that speaks English and Spanish fluently would generate a REGISTER that looks like, in part:

```
REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: sip:Y3@pc.example.com;languages="es,en"
```

Notice that only a single contact is needed because the same q-value is applied across all feature collections.

For the language based routing to occur, the caller must indicate its language preferences explicitly:

```
INVITE sip:Y@example.com SIP/2.0
Accept-Contact: *;languages="en"
```

When a caller makes a call, and indicates that their language is English only, the result is an Accept-Contact feature set that looks like:
This matches all Y1 phones, the second contact registered by Y2 phones, and Y3 phones. However, the second contact registered by Y2 phones has a lower q-value, 0.2, and therefore, will be chosen with lower preference than Y1 and Y3. The call will be routed to Y1 or Y3 UAs first, as desired. If neither of those pick up, the call is routed to the English-language contact of Y2, which represents a user with moderate English skills. In fact, the user will know to answer with "Hello" instead of "Hola" because the request-URI will contain Y2-en instead of just Y2. A nice bonus feature.

When a caller makes a call, and indicates that their language is Spanish only, the result is an Accept-Contact feature set that looks like:

(& (languages="es"))

This matches the first contact of Y2 phones, and Y3 phones, all with equal preference. The result is that a call is routed to a user that speaks Spanish.

11 I Hate Voicemail!

11.1 Desired Behavior

AOR Y has two contacts, a phone Y1 and a voicemail service Y2. X wishes to call Y and talk in person. X does not want to be sent to voicemail under any circumstance.

11.2 Solution

The phone would register with a Contact that looks like, in part:

REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: sip:Y1@pc.example.com

and the voicemail server would register with a Contact that looks like, in part:
Note that the voicemail server need not actually register. There can be a configured contact and feature set defined for it instead.

A caller that wishes to avoid voicemail can include an explicit preference to avoid it. It would do this with the Reject-Contact header field:

INVITE sip:Y@example.com SIP/2.0
Reject-Contact: *;msgserver

Since this feature set contains a feature tag that is not contained in the registration for Y1, the feature set is discarded when examining Y1. However, the registration for Y2 contains all feature tags listed in the feature set, and so the rule is considered. There is a match, and therefore, Y2 is discarded.

The result is that the user is never routed to voicemail.

12 I Hate People!

12.1 Desired Behavior

The situation is similar to Section 11, except the caller wishes to only leave a message, not actually speak to the person. Therefore, they would send an INVITE which looks like, in part:

INVITE sip:Y@example.com SIP/2.0
Accept-Contact: *;msgserver;require;explicit

This caller preference matches both Y1 and Y2. Y1 matches, but with a score of zero. Y2 matches with a score of 1. Since both the require and explicit flags are set, Y1 is discarded. Therefore, the call is routed to Y2, the voicemail server, as desired.

13 Prefer Voicemail
13.1 Desired Behavior

The situation is similar to that of Section 11. However, the caller prefers to leave a message. If voicemail is not available, they are willing to talk to a person.

13.2 Solution

To accomplish this behavior, the caller would generate an INVITE that looks like, in part:

```
INVITE sip:Y@example.com SIP/2.0
Accept-Contact: *;voicemail
```

This matches Y1 with a score of zero and Y2 with a score of 1. Therefore, Y2 (the voicemail server) is tried first, and if that fails, the lower priority Y1 (the phone) is tried next.

14 Routing to an Executive

14.1 Desired Behavior

Y is the AOR of an executive. It has three contacts. Y1 is the phone on the executive’s desk. Y2 is the phone on the desk of the executive’s assistant. Y3 is the address of an auto-attendant system that can answer general questions, route calls to other parties, etc. By default, calls to Y should be directed to Y2, and if that fails, to Y3. If Y3 doesn’t answer then Y1 should ring.

14.2 Solution

This is primarily a called party feature, and is best accomplished with a CPL script [8]. However, it can be accomplished with caller preferences alone by properly setting the q-values across the three devices. Assuming this coordination is possible, here are the settings that would be made:

Y1 would generate a REGISTER that looks like, in part:

```
REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: sip:Y1@pc.example.com;q=0.1
```
Y2 would generate a REGISTER that looks like, in part:

```
REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: sip:Y2@pc2.example.com;attendant;q=1.0
```

Y3 would generate a REGISTER that looks like, in part:

```
REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: sip:Y3@pc3.example.com;attendant;automata;q=0.5
```

Note that, in reality, the automated attendant would probably not use REGISTER. Since the attendant would be used for every employee in the company, a static contact would probably be added administratively for each user in the enterprise. However, the information in that static contact would be identical to the information in the registration above.

When X makes a call to the executive, Y, and expresses no preferences, the call is first routed to Y2, then Y3, then Y1, all as a result of the proper setting of the q-values.

15 Speak to the Executive

15.1 Desired Behavior

This case is similar to that of Section 14, but this time the caller, X, has a preference. X calls Y, but wants to speak directly to the executive. X doesn’t want the call to ring either the assistant or the auto attendant (automata).

15.2 Solution

X’s INVITE would look like, in part:

```
INVITE sip:Y@example.com SIP/2.0
Reject-Contact: *;attendant
Reject-Contact: *;automata
```
Note that the caller uses two separate Reject-Contact header field values, rather than a single one with two separate feature parameters. The distinction is important. If X had use a single value with two parameters, a matching UA would need to declare that it was BOTH an attendant and an automata. If it only declared that it was one of these, based on the matching rules in the caller preferences specification, it would not be rejected.

The above request would result in the elimination of both Y2 and Y3 as contacts. The call would then be routed to Y1, as desired.

16 Mobile Phone Only

16.1 Desired Behavior

The situation is similar to that in Section 14. However, the executive also has a mobile phone which they have registered. Caller X knows that the owner of Y is traveling, and that an assistant is covering the office phone. X wants to call Y and ring only the mobile phone.

16.2 Solution

The mobile phone would generate a registration which looks like, in part:

REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: sip:Y4@mobile.example.com;mobility="mobile";q=0.5

The caller would express their preference by generating an INVITE which looks like, in part:

INVITE sip:Y@example.com SIP/2.0
Accept-Contact: *;mobility="mobile";require;explicit

All three contacts match. However, Y1 through Y3 match with a score of zero. Y4 matches with a score of 1. Because of the require and explicit tags, Y1 through Y3 are discarded, and only Y4 is used, as desired.

17 Simultaneous Languages
17.1 Desired Behavior

AOR Y is as in Section 10. Caller X, fluent in both English and Spanish, has discovered that company’s Spanish language documentation is inconsistent with the English language documentation, and wants to discuss the differences between the two. So X wants to speak with one of the workers that is fluent in both English and Spanish.

17.2 Solution

The caller would generate an INVITE which looks like, in part:

INVITE sip:Y@example.com SIP/2.0
Accept-Contact: *;language="en";require
Accept-Contact: *;language="es";require

This will require a Contact URI to match both constraints. That means it needs to support English and Spanish. This will achieve the desired property.

Note that there are two separate Accept-Contact header fields. If the caller had instead used this INVITE:

INVITE sip:Y@example.com SIP/2.0
Accept-Contact: *;language="en,es";require

It would have connected them to a UA that speaks either English or Spanish, which is not what is desired here.

18 UA Routing

18.1 Desired Behavior

AOR Y has contacts Y1 and Y2. The addresses Y1 and Y2 are behind a firewall and are not addressable by all callers. Caller X makes a call to Y and is connected to Y1. The call fails for some reason, and X wants to reestablish it, reaching only Y1.

18.2 Solution

There is currently no generally workable solution to this problem. The best solution that exists does make use of caller preferences.
Lets say that the Contact URI for Y1 was sip:Y1@host.example.com. The caller would generate an INVITE which looks like, in part:

INVITE sip:Y@example.com SIP/2.0
Accept-Contact: sip:Y1@host.example.com;require

or, equivalently:

INVITE sip:Y@example.com SIP/2.0
Accept-Contact: *;uri-user="<Y1>";uri-domain="host.example.com";require

When this request arrives at the proxy for example.com, it attempts to apply the caller preferences. Following the guidelines in the caller preferences extension, Y1 would have included a uri-user and uri-domain feature tag in its registration:

REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: sip:Y1@host.example.com;uri-user="<Y1>"
;uri-domain="host.example.com"

as would have Y2:

REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: sip:Y2@pc.example.com;uri-user="<Y2>"
;uri-domain="pc.example.com"

The proxy then applies the caller preferences. Only Y1 is a match. So, Y2 is discarded, and the request is routed to Y1 as desired.

The difficulty is that this solution won’t always work. In a multi-proxy scenario, it is possible that the routing logic changes, and therefore the request is never even routed to the proxy where Y1 has registered.
19 The Number you Have Called..

19.1 Desired Behavior

Consider once more the case of the executive, where the caller wishes to reach only their mobile phone (Section 16). However, there is a twist. The callee Y has moved to new address YY, and all the configuration described for the callee now applies to YY. The old address Y remains with a pair of statically assigned contacts. One contact is YY. The other is M referencing an automaton that generates a voice message reporting that the number has been changed. The caller is unaware of the move and calls Y, requesting to reach the mobile phone in exactly the same way they did in Section 16. The call should connect to the mobile.

19.2 Solution

There would be four registrations against YY:

YY1 would generate a REGISTER that looks like, in part:

REGISTER sip:example.com SIP/2.0
To: sip:YY@example.com
Contact: sip:YY1@pc.example.com;q=0.1

YY2 would generate a REGISTER that looks like, in part:

REGISTER sip:example.com SIP/2.0
To: sip:YY@example.com
Contact: sip:YY2@pc2.example.com;attendant;q=1.0

YY3 would generate a REGISTER that looks like, in part:

REGISTER sip:example.com SIP/2.0
To: sip:YY@example.com
Contact: sip:YY3@pc3.example.com;attendant;automata;q=0.5

YY4 would generate a REGISTER that looks like, in part:
Although it would be configured administratively, there are two registered contacts for Y. The first is for the forwarding:

```
REGISTER sip:example.com SIP/2.0
To: sip:YY@example.com
Contact: sip:YY4@mobile.example.com;mobility="mobile";q=0.5
```

and the second for the automated answering service:

```
REGISTER sip:example.com SIP/2.0
To: sip:Y@example.com
Contact: sip:machine@example.com;automata;q=0.5
```

The caller, not knowing that Y has moved, calls Y and asks for their mobile phone:

```
INVITE sip:Y@example.com SIP/2.0
Accept-Contact: *;mobility="mobile";require;explicit
```

This reaches the example.com proxy, which finds two registrations. Only one of these is associated with feature parameters (the automata). Therefore, the caller preferences are applied to it alone. The feature sets match, but not explicitly. Since the require and explicit tags are present, the contact for the automata is dropped. The proxy therefore sends the call to sip:Y@example.com, which remains because it was unfiltered by the caller preferences processing. There, there are four registrations, all of which are associated with feature parameters. The caller preferences are applied. Only YY4 matches explicitly, however. As such, the call is forwarded there, and the mobile phone rings.

20 The Number you Have Called, Take Two
20.1 Desired Behavior

This use case is nearly identical to that of Section 19. However, this time, the caller wishes to contact the personal phone of Y. They don’t feel strongly about it, and will accept other devices.

20.2 Solution

The INVITE generated by the caller in this case will look like:

```
INVITE sip:Y@example.com SIP/2.0
Accept-Contact: *;class="personal"
```

This reaches the example.com proxy. Once more, the first registration (which forwards to the address-of-record for YY) is unaffected by the caller preferences computation. The other contact, for the automata, is a match, but its score is zero. Therefore, the first contact is preferred. This causes the call to be routed to sip:YY@example.com. There, all four contacts match, but none explicitly. Indeed, all four have a score of zero against the explicit preference. However, they each match the implicit preference for the INVITE method. The result is that the the q-values are unaffected, and the call is routed to YY2 first.

21 Security Considerations

There are no security considerations associated with this specification.

22 IANA Considerations

There are no IANA considerations associated with this specification.

23 Author’s Addresses

Jonathan Rosenberg
dynamicsoft
72 Eagle Rock Avenue
First Floor
East Hanover, NJ 07936
e-mail: jdrosen@dynamicsoft.com

Paul Kyzivat
Cisco Systems
24 Normative References

25 Informative References


and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
# Table of Contents

1. Introduction ........................................ 4
2. Routing of INVITE and MESSAGE to different UA ....... 4
   2.1 Desired Behavior .................................. 4
   2.2 Solution ........................................ 4
3. Single Contact Not Matching Implicit Preferences ... 5
   3.1 Desired Behavior .................................. 5
   3.2 Solution ........................................ 6
4. Package-Based Routing ................................ 6
   4.1 Desired Behavior .................................. 6
   4.2 Solution ........................................ 6
5. Audio/Video vs. Audio Only ............................ 8
   5.1 Desired Behavior .................................. 8
   5.2 Solution ........................................ 8
6. Audio/Video vs. Audio Only: A/V UA Fails ............ 9
   6.1 Desired Behavior .................................. 9
   6.2 Solution ........................................ 9
7. Forcing Audio/Video ................................ 9
   7.1 Desired Behavior .................................. 9
   7.2 Solution ........................................ 9
8. Third Party Call Control - Forcing Media ............ 10
   8.1 Desired Behavior .................................. 10
   8.2 Solution ........................................ 10
9. Maximizing Media Overlaps ............................ 10
   9.1 Desired Behavior .................................. 10
   9.2 Solution ........................................ 10
10. Multilingual Lines .................................. 11
   10.1 Desired Behavior .................................. 11
   10.2 Solution ........................................ 11
11. I Hate Voicemail! .................................. 13
   11.1 Desired Behavior .................................. 13
   11.2 Solution ........................................ 13
12. I Hate People! ..................................... 14
   12.1 Desired Behavior .................................. 14
13. Prefer Voicemail .................................... 14
   13.1 Desired Behavior .................................. 14
   13.2 Solution ........................................ 14
14. Routing to an Executive ............................. 15
   14.1 Desired Behavior .................................. 15
   14.2 Solution ........................................ 15
15. Speak to the Executive ............................... 16
   15.1 Desired Behavior .................................. 16

J. Rosenberg et. al. [Page 1]
15.2 Solution ............................................ 16
16 Mobile Phone Only ..................................... 17
16.1 Desired Behavior ..................................... 17
16.2 Solution ............................................ 17
17 Simultaneous Languages ............................... 17
17.1 Desired Behavior ..................................... 18
17.2 Solution ............................................ 18
18 UA Routing ........................................... 18
18.1 Desired Behavior ..................................... 18
18.2 Solution ............................................ 18
19 The Number you Have Called.. .......................... 20
19.1 Desired Behavior ..................................... 20
19.2 Solution ............................................ 20
20 The Number you Have Called, Take Two .............. 21
20.1 Desired Behavior ..................................... 22
20.2 Solution ............................................ 22
21 Security Considerations ............................... 22
22 IANA Considerations ................................... 22
23 Author’s Addresses ..................................... 22
24 Normative References .................................. 23
25 Informative References ................................ 23