Internet Engineering Task Force (IETF)
Request for Comments: 6364
Category: Standards Track
ISSN: 2070-1721

A. Begen
Cisco

October 2011

Session Description Protocol Elements for the Forward Error Correction (FEC) Framework

Abstract

This document specifies the use of the Session Description Protocol (SDP) to describe the parameters required to signal the Forward Error Correction (FEC) Framework Configuration Information between the sender(s) and receiver(s). This document also provides examples that show the semantics for grouping multiple source and repair flows together for the applications that simultaneously use multiple instances of the FEC Framework.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc6364.

Copyright Notice

Table of Contents

1. Introduction

The Forward Error Correction (FEC) Framework, described in [RFC6363], outlines a general framework for using FEC-based error recovery in packet flows carrying media content. While a continuous signaling between the sender(s) and receiver(s) is not required for a Content Delivery Protocol (CDP) that uses the FEC Framework, a set of parameters pertaining to the FEC Framework has to be initially communicated between the sender(s) and receiver(s). A signaling protocol (such as the one described in [FECFRAME-CFG-SIGNAL]) is required to enable such communication, and the parameters need to be appropriately encoded so that they can be carried by the signaling protocol.

One format to encode the parameters is the Session Description Protocol (SDP) [RFC4566]. SDP provides a simple text-based format for announcements and invitations to describe multimedia sessions. These SDP announcements and invitations include sufficient information for the sender(s) and receiver(s) to participate in the multimedia sessions. SDP also provides a framework for capability negotiation, which can be used to negotiate all, or a subset, of the parameters pertaining to the individual sessions.

The purpose of this document is to introduce the SDP elements that are used by the CDPs using the FEC Framework that choose SDP [RFC4566] for their multimedia sessions.

2. Requirements Notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Forward Error Correction (FEC) and FEC Framework

This section gives a brief overview of FEC and the FEC Framework.

3.1. Forward Error Correction (FEC)

Any application that needs reliable transmission over an unreliable packet network has to cope with packet losses. FEC is an effective approach that provides reliable transmission, particularly in multicast and broadcast applications where the feedback from the receiver(s) is either not available or quite limited.

In a nutshell, FEC groups source packets into blocks and applies protection to generate a desired number of repair packets. These repair packets can be sent on demand or independently of any receiver feedback. The choice depends on the FEC scheme or the Content Delivery Protocol used by the application, the packet loss characteristics of the underlying network, the transport scheme (e.g., unicast, multicast, and broadcast), and the application itself. At the receiver side, lost packets can be recovered by erasure decoding provided that a sufficient number of source and repair packets have been received.

3.2. FEC Framework

The FEC Framework [RFC6363] outlines a general framework for using FEC codes in multimedia applications that stream audio, video, or other types of multimedia content. It defines the common components and aspects of Content Delivery Protocols (CDPs). The FEC Framework also defines the requirements for the FEC schemes that need to be used within a CDP. However, the details of the FEC schemes are not specified within the FEC Framework. For example, the FEC Framework defines what configuration information has to be known at the sender and receiver(s) at a minimum, but the FEC Framework neither specifies how the FEC repair packets are generated and used to recover missing source packets, nor dictates how the configuration information is communicated between the sender and receiver(s). These are rather specified by the individual FEC schemes or CDPs.

3.3. FEC Framework Configuration Information

The FEC Framework [RFC6363] defines a minimum set of information that has to be communicated between the sender and receiver(s) for proper operation of a FEC scheme. This information is called the "FEC Framework Configuration Information". This information includes unique identifiers for the source and repair flows that carry the source and repair packets, respectively. It also specifies how the sender applies protection to the source flow(s) and how the repair flow(s) can be used to recover lost data.

Multiple instances of the FEC Framework can simultaneously exist at the sender and the receiver(s) for different source flows, for the same source flow, or for various combinations of the source flows. Each instance of the FEC Framework provides the following FEC Framework Configuration Information:

1. Identification of the repair flows.

2. For each source flow protected by the repair flow(s):

   A. Definition of the source flow.

   B. An integer identifier for this flow definition (i.e., tuple). This identifier MUST be unique among all source flows that are protected by the same FEC repair flow. Integer identifiers can be allocated starting from zero and increasing by one for each flow. However, any random (but still unique) allocation is also possible. A source flow identifier need not be carried in source packets, since source packets are directly associated with a flow by virtue of their packet headers.

3. The FEC Encoding ID, identifying the FEC scheme.

4. The length of the Explicit Source FEC Payload ID (in octets).

5. Zero or more FEC-Scheme-Specific Information (FSSI) elements, each consisting of a name and a value where the valid element names and value ranges are defined by the FEC scheme.

FSSI includes the information that is specific to the FEC scheme used by the CDP. FSSI is used to communicate the information that cannot be adequately represented otherwise and is essential for proper FEC encoding and decoding operations. The motivation behind separating the FSSI required only by the sender (which is carried in a Sender-Side FEC-Scheme-Specific Information (SS-FSSI) container) from the rest of the FSSI is to provide the receiver or the third-party entities a means of controlling the FEC operations at the sender. Any FSSI other than the one solely required by the sender MUST be communicated via the FSSI container.

The variable-length SS-FSSI and FSSI containers transmit the information in textual representation and contain zero or more distinct elements, whose descriptions are provided by the fully specified FEC schemes.

4. SDP Elements

This section defines the SDP elements that MUST be used to describe the FEC Framework Configuration Information in multimedia sessions by the CDPs that choose SDP [RFC4566] for their multimedia sessions. Example SDP descriptions can be found in Section 6.

4.1. Transport Protocol Identifiers

This specification defines a new transport protocol identifier for the FEC schemes that take a UDP-formatted input stream and append an Explicit Source FEC Payload ID, as described in Section 5.3 of [RFC6363], to generate a source flow. This new protocol identifier is called 'FEC/UDP'. To use input streams that are formatted according to another <proto> (as listed in the table for the 'proto' field in the "Session Description Protocol (SDP) Parameters" registry), the corresponding 'FEC/<proto>' transport protocol identifier MUST be registered with IANA by following the instructions specified in [RFC4566].

Note that if a FEC scheme does not use the Explicit Source FEC Payload ID as described in Section 4.1 of [RFC6363], then the original transport protocol identifier MUST be used to support backward compatibility with the receivers that do not support FEC at all.

This specification also defines another transport protocol identifier, 'UDP/FEC', to indicate the FEC repair packet format defined in Section 5.4 of [RFC6363]. For detailed registration information, refer to Section 8.1.

4.2. Media Stream Grouping

In the FEC Framework, the 'group' attribute and the FEC grouping semantics defined in [RFC5888] and [RFC5956], respectively, are used to associate source and repair flows.

4.3. Source IP Addresses

The 'source-filter' attribute of SDP ("a=source-filter") as defined in [RFC4570] is used to express the source addresses or fully qualified domain names in the FEC Framework.

4.4. Source Flows

The FEC Framework allows that multiple source flows MAY be grouped and protected together by single or multiple FEC Framework instances. For this reason, as described in Section 3.3, individual source flows MUST be identified with unique identifiers. For this purpose, we introduce the attribute 'fec-source-flow'.

The syntax for the new attribute in ABNF [RFC5234] is as follows:

```
   fec-source-flow-line = "a=fec-source-flow:" SP source-id
        [";" SP tag-length] CRLF

   source-id = "id=" src-id
   src-id = 1*DIGIT ; Represented as 32-bit non-negative
                    ; integers, and leading zeros are ignored

   tag-length = "tag-len=" tlen
   tlen = %x31-39 *DIGIT
```

The REQUIRED parameter 'id' is used to identify the source flow. Parameter 'id' MUST be an integer.

The 'tag-len' parameter is used to specify the length of the Explicit Source FEC Payload ID field (in octets). In the case that an Explicit Source FEC Payload ID is used, the 'tag-len' parameter MUST exist and indicate its length. Otherwise, the 'tag-len' parameter MUST NOT exist.

4.5. Repair Flows

A repair flow MUST contain only repair packets formatted as described in [RFC6363] for a single FEC Framework instance; i.e., packets belonging to source flows or other repair flows from a different FEC Framework instance cannot be sent within this flow. We introduce the attribute 'fec-repair-flow' to describe the repair flows.

The syntax for the new attribute in ABNF is as follows (CHAR and CTL are defined in [RFC5234]):

```
fec-repair-flow-line = "a=fec-repair-flow:" SP fec-encoding-id
     [";" SP flow-preference]
     [";" SP sender-side-scheme-specific]
     [";" SP scheme-specific] CRLF

fec-encoding-id = "encoding-id=" enc-id
enc-id = 1*DIGIT ; FEC Encoding ID

flow-preference = "preference-lvl=" preference-level-of-the-flow
preference-level-of-the-flow = 1*DIGIT
```

```
sender-side-scheme-specific = "ss-fssi=" sender-info
sender-info = element *( "," element )
element     = name ":" value
name        = token
token       = 1*<any CHAR except CTLs or separators>
value       = *<any CHAR except CTLs or separators>
separator   = "(" / ")" / "<" / ">" / "@"
               / "," / ";" / ":" / "\" / DQUOTE
               / "/" / "[" / "]" / "?" / "="
               / "{" / "}" / SP / HTAB

scheme-specific = "fssi=" scheme-info
scheme-info = element *( "," element )
```

The REQUIRED parameter 'encoding-id' is used to identify the FEC scheme used to generate this repair flow. These identifiers (in the range of [0 - 255]) are registered by the FEC schemes that use the FEC Framework and are maintained by IANA.

The OPTIONAL parameter 'preference-lvl' is used to indicate the preferred order for using the repair flows. The exact usage of the parameter 'preference-lvl' and the pertaining rules MAY be defined by the FEC scheme or the CDP. If the parameter 'preference-lvl' does not exist, it means that the receiver(s) MAY receive and use the repair flows in any order. However, if a preference level is assigned to the repair flow(s), the receivers are encouraged to follow the specified order in receiving and using the repair flow(s).

The OPTIONAL parameters 'ss-fssi' and 'fssi' are containers to convey the FEC-Scheme-Specific Information (FSSI) that includes the information that is specific to the FEC scheme used by the CDP and is necessary for proper FEC encoding and decoding operations. The FSSI required only by the sender (the Sender-Side FSSI) MUST be communicated in the container specified by the parameter 'ss-fssi'. Any other FSSI MUST be communicated in the container specified by the parameter 'fssi'. In both containers, FSSI is transmitted in the form of textual representation and MAY contain multiple distinct elements. If the FEC scheme does not require any specific information, the 'ss-fssi' and 'fssi' parameters MUST NOT exist.

4.6. Repair Window

The repair window is the time that spans a FEC block, which consists of the source block and the corresponding repair packets.

At the sender side, the FEC encoder processes a block of source packets and generates a number of repair packets. Then, both the source and repair packets are transmitted within a certain duration not larger than the value of the repair window. The value of the repair window impacts the maximum number of source packets that can be included in a FEC block.

At the receiver side, the FEC decoder should wait at least for the duration of the repair window after getting the first packet in a FEC block, to allow all the repair packets to arrive. (The waiting time can be adjusted if there are missing packets at the beginning of the FEC block.) The FEC decoder can start decoding the already received packets sooner; however, it SHOULD NOT register a FEC decoding failure until it waits at least for the duration of the repair window.

This document specifies a new attribute to describe the size of the repair window in milliseconds and microseconds.

The syntax for the attribute in ABNF is as follows:

```
repair-window-line = "a=repair-window:" window-size unit CRLF

window-size = %x31-39 *DIGIT ; Represented as
                             ; 32-bit non-negative integers

unit = "ms" / "us"
```

<unit> is the unit of time specified for the repair window size. Two units are defined here: 'ms', which stands for milliseconds; and 'us', which stands for microseconds.

The 'a=repair-window' attribute is a media-level attribute, since each repair flow MAY have a different repair window size.

Specifying the repair window size in an absolute time value does not necessarily correspond to an integer number of packets or exactly match with the clock rate used in RTP (in the case of RTP transport), causing mismatches among subsequent repair windows. However, in practice, this mismatch does not break anything in the FEC decoding process.

4.7. Bandwidth Specification

The bandwidth specification as defined in [RFC4566] denotes the proposed bandwidth to be used by the session or media. The specification of bandwidth is OPTIONAL.

In the context of the FEC Framework, the bandwidth specification can be used to express the bandwidth of the repair flows or the bandwidth of the session. If included in the SDP, it SHALL adhere to the following rules.

The session-level bandwidth for a FEC Framework instance or the media-level bandwidth for the individual repair flows MAY be specified. In this case, it is RECOMMENDED that the Transport Independent Application Specific (TIAS) bandwidth modifier [RFC3890] and the 'a=maxprate' attribute be used, unless the Application-Specific (AS) bandwidth modifier [RFC4566] is used. The use of the AS bandwidth modifier is NOT RECOMMENDED, since TIAS allows the calculation of the bitrate according to the IP version and transport protocol whereas AS does not. Thus, in TIAS-based bitrate calculations, the packet size SHALL include all headers and payload, excluding the IP and UDP headers. In AS-based bitrate calculations, the packet size SHALL include all headers and payload, plus the IP and UDP headers.

For the ABNF syntax information of the TIAS and AS, refer to [RFC3890] and [RFC4566], respectively.

5. Scenarios and Examples

This section discusses the considerations for Session Announcement and Offer/Answer Models.

5.1. Declarative Considerations

In multicast-based applications, the FEC Framework Configuration Information pertaining to all FEC protection options available at the sender MAY be advertised to the receivers as a part of a session announcement. This way, the sender can let the receivers know all available options for FEC protection. Based on their needs, the receivers can choose protection provided by one or more FEC Framework instances and subscribe to the respective multicast session(s) to receive the repair flow(s). Unless explicitly required by the CDP, the receivers SHOULD NOT send an answer back to the sender specifying their choices, since this can easily overwhelm the sender, particularly in large-scale multicast applications.

5.2. Offer/Answer Model Considerations

In unicast-based applications, a sender and receiver MAY adopt the Offer/Answer Model [RFC3264] to set the FEC Framework Configuration Information. In this case, the sender offers the options available to this particular receiver, and the receiver answers back to the sender with its choice(s).

Receivers supporting the SDP Capability Negotiation Framework [RFC5939] MAY also use this framework to negotiate all, or a subset, of the FEC Framework parameters.

The backward compatibility in the Offer/Answer Model is handled as specified in [RFC5956].

6. SDP Examples

This section provides SDP examples that can be used by the FEC Framework.

[RFC5888] defines the media stream identification attribute ('mid') as a token in ABNF. In contrast, the identifiers for the source flows are integers and can be allocated starting from zero and increasing by one for each flow. To avoid any ambiguity, using the same values for identifying the media streams and source flows is NOT RECOMMENDED, even when 'mid' values are integers.

In the examples below, random FEC Encoding IDs will be used for illustrative purposes. Artificial content for the SS-FSSI and FSSI will also be provided.

6.1. One Source Flow, One Repair Flow, and One FEC Scheme

```
SOURCE FLOWS             | INSTANCE #1
S1: Source Flow |--------| R1: Repair Flow
                         |
```

Figure 1: Scenario #1

In this example, we have one source video flow (mid:S1) and one FEC repair flow (mid:R1). We form one FEC group with the "a=group:FEC-FR S1 R1" line. The source and repair flows are sent to the same port on different multicast groups. The repair window is set to 150 ms.

```
v=0
o=ali 1122334455 1122334466 IN IP4 fec.example.com
s=FEC Framework Examples
t=0 0
a=group:FEC-FR S1 R1
m=video 30000 RTP/AVP 100
c=IN IP4 233.252.0.1/127
a=rtpmap:100 MP2T/90000
a=fec-source-flow: id=0
a=mid:S1
m=application 30000 UDP/FEC
c=IN IP4 233.252.0.2/127
a=fec-repair-flow: encoding-id=0; ss-fssi=n:7,k:5
a=repair-window:150ms
a=mid:R1
```

6.2. Two Source Flows, One Repair Flow, and One FEC Scheme

```
SOURCE FLOWS
S2: Source Flow |          | INSTANCE #1
                |---------| R2: Repair Flow
S3: Source Flow |
```

Figure 2: Scenario #2

In this example, we have two source video flows (mid:S2 and mid:S3) and one FEC repair flow (mid:R2) protecting both source flows. We form one FEC group with the "a=group:FEC-FR S2 S3 R2" line. The source and repair flows are sent to the same port on different multicast groups. The repair window is set to 150500 us.

```
v=0
o=ali 1122334455 1122334466 IN IP4 fec.example.com
s=FEC Framework Examples
t=0 0
a=group:FEC-FR S2 S3 R2
m=video 30000 RTP/AVP 100
c=IN IP4 233.252.0.1/127
a=rtpmap:100 MP2T/90000
a=fec-source-flow: id=0
a=mid:S2
m=video 30000 RTP/AVP 101
c=IN IP4 233.252.0.2/127
a=rtpmap:101 MP2T/90000
a=fec-source-flow: id=1
a=mid:S3
m=application 30000 UDP/FEC
c=IN IP4 233.252.0.3/127
a=fec-repair-flow: encoding-id=0; ss-fssi=n:7,k:5
a=repair-window:150500us
a=mid:R2
```

6.3. Two Source Flows, Two Repair Flows, and Two FEC Schemes

```
SOURCE FLOWS              | INSTANCE #1
S4: Source Flow |--------| R3: Repair Flow

S5: Source Flow |--------| INSTANCE #2
                          | R4: Repair Flow
```

Figure 3: Scenario #3

In this example, we have two source video flows (mid:S4 and mid:S5) and two FEC repair flows (mid:R3 and mid:R4). The source flows mid:S4 and mid:S5 are protected by the repair flows mid:R3 and mid:R4, respectively. We form two FEC groups with the "a=group:FEC-FR S4 R3" and "a=group:FEC-FR S5 R4" lines. The source and repair flows are sent to the same port on different multicast groups. The repair window is set to 200 ms and 400 ms for the first and second FEC group, respectively.

```
v=0
o=ali 1122334455 1122334466 IN IP4 fec.example.com
s=FEC Framework Examples
t=0 0
a=group:FEC-FR S4 R3
a=group:FEC-FR S5 R4
m=video 30000 RTP/AVP 100
c=IN IP4 233.252.0.1/127
a=rtpmap:100 MP2T/90000
a=fec-source-flow: id=0
a=mid:S4
m=video 30000 RTP/AVP 101
c=IN IP4 233.252.0.2/127
a=rtpmap:101 MP2T/90000
a=fec-source-flow: id=1
a=mid:S5
m=application 30000 UDP/FEC
c=IN IP4 233.252.0.3/127
a=fec-repair-flow: encoding-id=0; ss-fssi=n:7,k:5
a=repair-window:200ms
a=mid:R3
m=application 30000 UDP/FEC
c=IN IP4 233.252.0.4/127
a=fec-repair-flow: encoding-id=0; ss-fssi=n:14,k:10
a=repair-window:400ms
a=mid:R4
```

6.4. One Source Flow, Two Repair Flows, and Two FEC Schemes

```
SOURCE FLOWS                | INSTANCE #1
S6: Source Flow |--------| R5: Repair Flow
                |
                |--------| INSTANCE #2
                         | R6: Repair Flow
```

Figure 4: Scenario #4

In this example, we have one source video flow (mid:S6) and two FEC repair flows (mid:R5 and mid:R6) with different preference levels. The source flow mid:S6 is protected by both of the repair flows. We form two FEC groups with the "a=group:FEC-FR S6 R5" and "a=group:FEC-FR S6 R6" lines. The source and repair flows are sent to the same port on different multicast groups. The repair window is set to 200 ms for both FEC groups.

```
v=0
o=ali 1122334455 1122334466 IN IP4 fec.example.com
s=FEC Framework Examples
t=0 0
a=group:FEC-FR S6 R5
a=group:FEC-FR S6 R6
m=video 30000 RTP/AVP 100
c=IN IP4 233.252.0.1/127
a=rtpmap:100 MP2T/90000
a=fec-source-flow: id=0
a=mid:S6
m=application 30000 UDP/FEC
c=IN IP4 233.252.0.3/127
a=fec-repair-flow: encoding-id=0; preference-lvl=0; ss-fssi=n:7,k:5
a=repair-window:200ms
a=mid:R5
m=application 30000 UDP/FEC
c=IN IP4 233.252.0.4/127
a=fec-repair-flow: encoding-id=1; preference-lvl=1; ss-fssi=t:3
a=repair-window:200ms
a=mid:R6
```

7. Security Considerations

There is a weak threat if the SDP is modified in a way that it shows an incorrect association and/or grouping of the source and repair flows. Such attacks can result in failure of FEC protection and/or mishandling of other media streams. It is RECOMMENDED that the receiver perform an integrity check on SDP to only trust SDP from trusted sources. The receiver MUST also follow the security considerations of SDP [RFC4566]. For other general security considerations related to SDP, refer to [RFC4566]. For the security considerations related to the use of source address filters in SDP, refer to [RFC4570].

The security considerations for the FEC Framework also apply. Refer to [RFC6363] for details.

8. IANA Considerations

8.1. Registration of Transport Protocols

This specification updates the "Session Description Protocol (SDP) Parameters" registry as defined in Section 8.2.2 of [RFC4566]. Specifically, it adds the following values to the table for the ’proto’ field.

Type	SDP Name	Reference
proto	FEC/UDP	[RFC6364]
proto	UDP/FEC	[RFC6364]

8.2. Registration of SDP Attributes

This document registers new attribute names in SDP.

```
SDP Attribute ("att-field"):
     Attribute name:     fec-source-flow
     Long form:          Pointer to FEC Source Flow
     Type of name:       att-field
     Type of attribute:  Media level
     Subject to charset: No
     Purpose:            Provide parameters for a FEC source flow
     Reference:          [RFC6364]
     Values:             See [RFC6364]

SDP Attribute ("att-field"):
     Attribute name:     fec-repair-flow
     Long form:          Pointer to FEC Repair Flow
     Type of name:       att-field
     Type of attribute:  Media level
     Subject to charset: No
     Purpose:            Provide parameters for a FEC repair flow
     Reference:          [RFC6364]
     Values:             See [RFC6364]

SDP Attribute ("att-field"):
     Attribute name:     repair-window
     Long form:          Pointer to FEC Repair Window
     Type of name:       att-field
     Type of attribute:  Media level
     Subject to charset: No
     Purpose:            Indicate the size of the repair window
     Reference:          [RFC6364]
     Values:             See [RFC6364]
```

9. Acknowledgments

The author would like to thank the FEC Framework Design Team for their inputs, suggestions, and contributions.

10. References

10.1. Normative References

[RFC6363] Watson, M., Begen, A., and V. Roca, "Forward Error Correction (FEC) Framework", RFC 6363, October 2011.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.

[RFC4570] Quinn, B. and R. Finlayson, "Session Description Protocol (SDP) Source Filters", RFC 4570, July 2006.

[RFC5888] Camarillo, G. and H. Schulzrinne, "The Session Description Protocol (SDP) Grouping Framework", RFC 5888, June 2010.

[RFC5956] Begen, A., "Forward Error Correction Grouping Semantics in the Session Description Protocol", RFC 5956, September 2010.

[RFC3890] Westerlund, M., "A Transport Independent Bandwidth Modifier for the Session Description Protocol (SDP)", RFC 3890, September 2004.

[RFC5234] Crocker, D., Ed., and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

10.2. Informative References

[FECFRAME-CFG-SIGNAL] Asati, R., "Methods to convey FEC Framework Configuration Information", Work in Progress, September 2011.

[RFC5939] Andreasen, F., "Session Description Protocol (SDP) Capability Negotiation", RFC 5939, September 2010.

Author's Address

Ali Begen
Cisco
181 Bay Street
Toronto, ON M5J 2T3
Canada

EMail: abegen@cisco.com