分散システム特論レポート
RFC 2543:SIP(Session Initiation Protocol)

情報工学専攻　修士課程1年　荒木研究室　武井 健太郎

平成12年1月31日

1　機能の概要

SIP(Session Initiation Protocol) はマルチメディアセッション及びコールの確立、変更、切断を行うアプリケーション層の制御プロトコルである。SIP は SAP やメール、web などの方法によって宣伝されたセッションを開始したり、メンバーをそのセッションへ招待することができる。この際呼び出し側がそのセッションのメンバーである必要は無く、また、人だけでなくメディア記憶サービスのような「ロボット」をセッションへ招待することも可能である。

また SIP は透過的にネームマッピングなどのアドレス変換サービスをサポートしており、これを用いて要求をユーザの現在地にプロキシ及びリダイレクションすることによりユーザの個人移動性（personal mobility）を実現する。

SIP はマルチメディアコミュニケーションにおいて以下の5つの側面をサポートする

ユーザロケーション
：セッションでユーザが用いる端末の決定
ユーザ能力
：用いられるメディア及びメディアパラメータの決定
ユーザ有用性
：パーティのコミュニケーションへの参加意志決定
コールセットアップ
：コールパラメータの確立
コール処理
：コールの転送及び切断

SIP はマルチキャストや網目状に張られたユニキャストを用いて多地点パーティコールを開始することも可能である。しかし SIP はマルチキャストアドレスの割り当ては行わない。

SIP のその他の特徴としては以下のようなものが挙げられる。

- トランスポートプロトコルに依存しない。
- 会議制御・資源予約の機能は提供しない。
- 他のコールセットアップ、シグナリングプロトコルとの共用が可能。

2　定義・用語

この RFC で規定される定義・用語について説明する。

- User agent(UA)
  ser agent client(UAC) と User agent server(UAS) を含むアプリケーションプログラム。UAC(=calling user agent) はサーバに対して SIP 要求を作成し、それを送信する。UAS(=called user agent) は SIP 要求を受信した時にユーザに接触し、ユーザの代わりに応答を返す。応答は要求に対する受諾、拒否、リダイレクトなどを示す。

- Proxy Server
  プロキシはサーバ及び（他のクライアントの代わりに要求を作成する際には) クライアントとして動作する中継プログラムである。前の要求の結果（サーバからの応答）を待たずに、次の要求を発行する並列探索（parallel search）を行う場合がある。

- Redirect Server
  リダイレクトサーバは SIP 要求を受けると、そのアドレスを0以上の新しいアドレスに変換（マッピング）し、それをクライアントに返す。リダイレクトサーバはプロキシサーバとは異なりそれ自身が SIP 要求を開始することはない。また UAS のようにコールを受諾することもない。

- Registrar
  レジストラは REGISTER 要求を受諾しロケーションサービスを提供するサーバであり、多くの場合プロキシ・リダイレクトサーバと共存している。

- コール
  1 つのコールは同じソースから招待された会議の参加者全てを含む。SIP コールは Call-ID により識別される。Call-ID はそのホストにおいて重複しない識別子である local-id を用いて、「local-id@host」という形式で表現されるコール識別子である。

- セッション記述
  INVITE で招待されるセッションに関する記述。セッションで用いられるメディアタイプ・パラメータなどを含み、多くの場合 SDP により記述される。2 地点間コールの場合、要求メッセージのセッション記述には呼び出し側のパラメータや使用したいメディアが記述される。

3 動作の概要

3.1 SIP addressing

SIP においてアドレシングされる対象は「user at host」であり、これは SIP URL によって表現される。SIP URL は RFC2396(URI:Universal Resource Identifier) に従って実装されており、SIP メッセージ中においてその要求の発起者(From)、現時送信先(Request-URI)、最終受領者(To)を示す為に用いられる。また、オプションにより、トランスポートプロトコルやポート番号などを指示したり、電話番号をユーザ部分で用いることがある。多くの場合ユーザの SIP URL はそのユーザの E メールアドレスから推測することが可能である。
examples:
sip:j.doe@big.com
sip:alice@10.1.2.3;transport=tcp

3.2 SIP invitation

SIP の基本的な動作である SIP 招待について説明する。

呼び出し側 (以下 caller) のクライアント(UAC)が INVITE 要求を送信しようとする際、Request-URI は多くの場合 To ヘッダと同じ URL に設定される。そしてクライアントはローカルに配置されたプロキシサーバか、Request-URI に対応する IP アドレス・ポートにその要求を送信する。この時 Request-URI に明記されていなければ、トランスポートプロトコルは UDP、ポート番号は 5060 が用いられる。

呼ばれる側(以下 callee)のサーバ(UAS)は INVITE 要求を受け取るとその要求に対する受諾・拒否・リダイレクトを示す最終応答、または処理の進捗状況を示す仮応答を caller に送信する。場合によってはその caller-callee 間に介在するプロキシが callee の代わりにこれらの応答を発行する場合もある。リダイレクトサーバが INVITE 要求を受け取った場合、代わりの新たな接触先アドレスのリストを Contact ヘッダに示し、3xx クラス応答メッセージ(後述)を caller に返す。

caller が callee からの最終応答を受け取ると、その受領通知のため、ACK 要求を callee に送信する。または場合によっては BYE 要求を送信し、そのコールを切断することもある。

SIP 招待の詳しい実例については 6 章で述べる。

4 要求(Request)

要求メッセージのフォーマットは以下の通りである。

```
Request = Request-Line
          *( general-header
           | request-header
           | entity-header )
          CRLF
          [ message-body ]

Request-Line = Method SP Request-URI SP SIP-Version
CRLF
```

4.1 メソッド

```
Method = "INVITE" | "ACK" | "OPTIONS" | "BYE"| "CANCEL"
| "REGISTER"
```

INVITE
ユーザやサービスなどに対するセッションへの参加要求。メッセージボディには callee が招待されるセッションのセッション記述を含む。
ACK
INVITE 要求に対する最終応答の受領通知。メッセージボディに最終的なセッション記述を含むことがある。
OPTIONS
サーバの能力に関する問い合わせ要求。
BYE
コール解放要求。caller と callee の両方で発行されるこ

とができる。

CANCEL

自身が発行した要求のうち未決定のものに対する中止要求。処理が完了している要求には影響しない。

REGISTER

レジストラに対するアドレスの登録要求。To ヘッダに登録したいアドレスを示す。

5　応答 (Response)

要求メッセージを受けると受領者はそれを解釈・処理して応答メッセージを返さなければならない。応答メッセージのフォーマットは以下の通りである。

```
Response = Status-Line
          *( general-header
           | response-header
           | entity-header)
          CRLF
          [ message-body ]

Status-Line = SIP-version SP Status-Code SP
              Reason-Phrase CRLF
```

5.1　Status Code

ステータスコードは 3 桁の整数で表され、要求に対する処理結果または処理の進捗状況を示す。Reason-Phrase にはその簡単な説明が記述される。コードの 1 桁目は応答クラスを示し、以下の 6 種類のクラスが存在する。

1xx:Informational

要求の処理の進捗状況

–「 100 Trying 」「 180 Ringing 」など

2xx:Success

要求は正しく処理され受諾された

–「 200 OK 」

3xx:Redirection

要求を完了するためには追加動作が必要

–「 301 Moved Permanently 」「 302 Moved Temporarily 」など

4xx:Client Error

要求のシンタックスエラー、またはそのサーバでの処理不能

–「 400 Bad Request 」「 404 Not Found 」など

5xx:Server Error

有効に思われる要求をサーバが実行に失敗した

–「 500 Internal Server Error 」「 501 Not Implemented 」など

6xx:Gloval Failure

その要求はどのサーバにおいても実行不能

–「 600 Busy Elsewhere 」「 603 Decline 」など

これらのコードの多くは HTTP/1.1 の応答コードから採られており、HTTP で新たに定義されるコードとの重複を防ぐ為に、SIP で新たに付け加えられるコードは各クラスの x80 からになっている。

6　実例

SIP 招待の動作例を紹介する。

図 1 と図 2 はそれぞれプロキシ、リダイレクトサーバを介した SIP 招待の動作例をヘッダの内容とともに示している。ヘッダは簡略化されていて、メッセージボディや Via ヘッダ (要求メッセージが通ったパスを記述する。これにより応答が要求と同じ送信路を通ることが保証される) などが省かれている。

まず図 1 では caller(cz@cs.tu-berlin.edu) が callee(hennning@cs.columbia.edu) に対して送信した INVITE 要求をプロキシが受信する (1)。そしてプロキシはロケーションサービスに問い合わせ、より詳しい callee のアドレスを得る (2、3)。次にプロキシは得られたアドレスに対し、INVITE 要求を発行し (4)。それを受信した callee の UAS はユーザにそれを通知し (5)、元の caller に対し成功応答 (200 OK) を返す (6)。このとき応答メッセージには要求メッセージの To、From ヘッダがコピーされる。プロキシは元の caller にそれをフォワードし (7)、caller はその応答を受領した旨を通知するために ACK 要求を発行する (8)。プロキシはそのメッセージを caller にフォワードする (9)。最後の ACK はプロキシを経由せずに caller から直接 callee に送ることも可能である。

次に図 2 では、リダイレクトサーバ (tune) が caller から INVITE 要求を受け、それに対しリダイレクション応答 (302 Moved temporarily) を返している (4)。この時その応答メッセージには新たな接触先のアドレス (hgs@play.columbia.edu) を示したヘッダが付記されている。caller はその応答を受信すると、リダイレクトサーバに対し ACK メッセージを送信し (5)、ヘッダで示されたアドレスへ同じ Call-ID で再び INVITE 要求を発行する (6)。

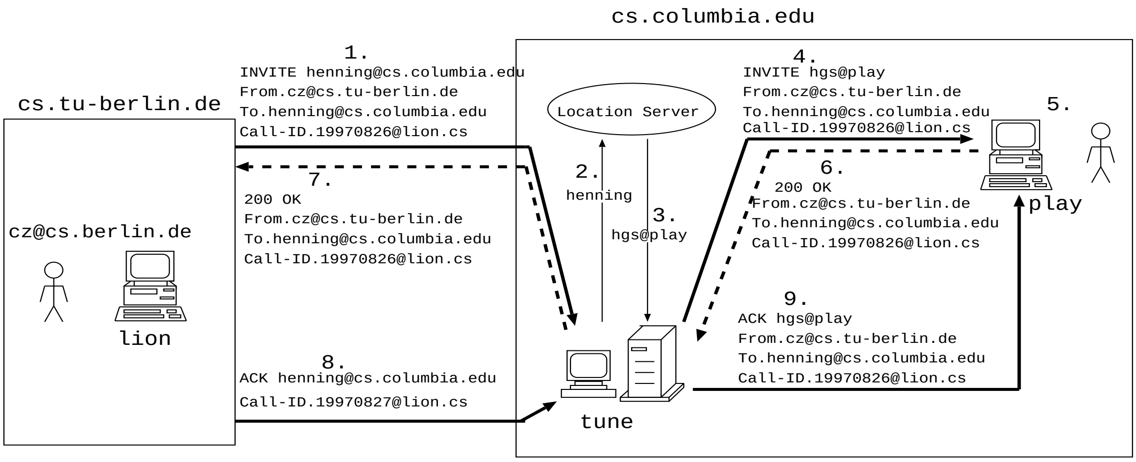

図 1: プロキシを介した SIP 招待の動作例

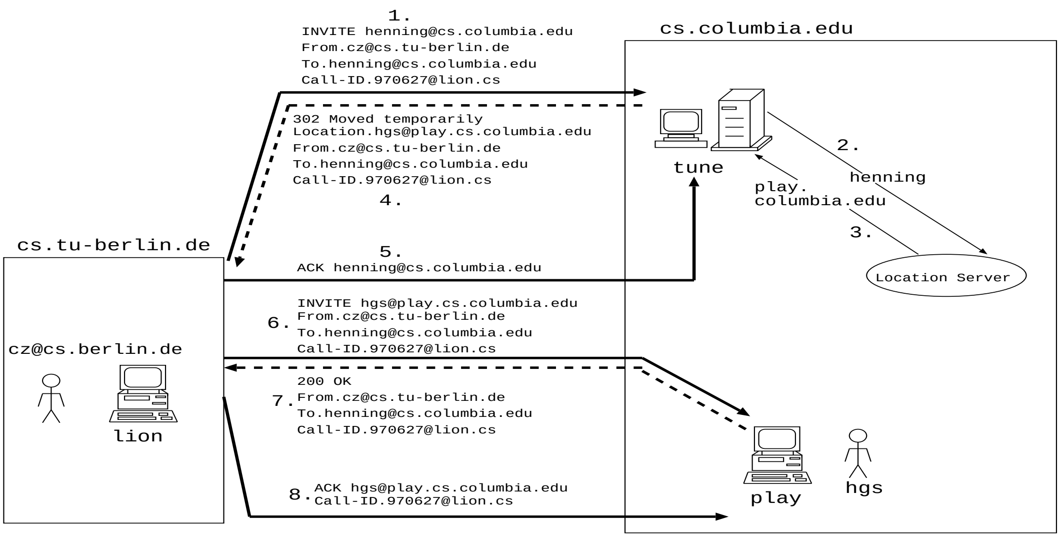

図 2: リダイレクトサーバを介した SIP 招待の動作例

7 実装

発表では SIP の実装として SDR(Session Dieectory) を紹介した。tcpdump を用いて caller(silvia.csce.kyushu-u.ac.jp) から callee(iris.csce.kyushu-u.ac.jp) に呼びかけた時のトラフィックの解析を callee 側で行った。その結果の冒頭部分を以下に抜粋する。

```
iris # tcpdump dst port 5060
02:56:49.601944 silvia.csce.kyushu-u.ac.jp.ms-sna-server
                    > iris.csce.kyushu-u.ac.jp.5060: udp 523
02:56:49.912608 iris.csce.kyushu-u.ac.jp.1640
                    > silvia.csce.kyushu-u.ac.jp.5060: udp 249
```

8 おわりに

SIP は IETF(Internet Engineering Task Force) のマルチメディアデータ・制御アーキテクチャの一部として設計されている。このアーキテクチャにはネットワーク資源予約のための RSVP(Resource Reservation Protocol)、実時間データ転送及び QoS フィードバックのための RTP(Real-time Transport Protocol)、ストリームメディア配送のための RTSP(Real Time Streaming Protocol)、マルチメディアセッションの宣伝のための SAP(Session Announcement Protocol)、マルチメディアセッションの記述のための SDP(Session Description Protocol) などのプロトコルが採り入れられている。しかし SIP はこれらのプロトコルからは独立して動作する。

また IESG(Internet Engineering Steering Group) ではメールアドレスや電話番号などによる「name lookup」

の標準を作るためのいくつかのワーキンググループを設立する予定である。そして「name lookup」の機能が必要な場合には、SIP の持つ lookup 機能よりも、それらのワーキンググループによるプロトコルを利用することが推奨されている。そのため将来的には SIP もそれらのプロトコルを利用するようになると考えられる。