

## 第 XVI 部

### ENUM テストベッドの運用



## 第 16 部

### ENUM テストベッドの運用

#### 第 1 章 ENUM WG の概要と目的

ENUM WG は、ENUM 運用トライアルを行い、サーバ・クライアントの実装に関する議論を行うことを目的として 2002 年 11 月に設立された。

ENUM WG の 2003 年度の活動を、以下に挙げる。

- 2002 年 11 月 ENUM WG 設立
- 2002 年 12 月 WIDE 研究会にて BOF 開催
- 2003 年 3 月 WIDE 合宿にて BOF 開催
- 2003 年 4 月 ENUM/SIP 勉強会開催
- 2003 年 5 月 e164.wide.ad.jp ゾーン作成、実験開始
- 2003 年 5 月 WIDE 研究会にて BOF 開催
- 2003 年 8 月 SIP サーバのテスト開始
- 2003 年 9 月 e164.wide.ad.jp 登録システム作成
- 2003 年 9 月 WIDE 合宿にて BOF 開催・デモンストレーション
- 2003 年 9 月 ETJP 発足
- 2003 年 12 月 WIDE 研究会にて BOF 開催

本報告では、第 2 章で ENUM についての基本的な技術解説を行い、第 3 章では ETJP (ENUM Trial Japan) についてまとめる。第 4 章では、ENUM WG が構築した ENUM テストベッドについて述べ、第 5 章では、ENUM WG メンバーが自社の製品に加えた ENUM 機能も含めて、ENUM アプリケーションについて報告する。第 6 章では、2003 年 9 月の WIDE 合宿にて開催した、ENUM/SIP デモンストレーションについて述べる。最後に付録として、ENUM を利用するシーケンス例、および 5.2 節で紹介するデモにおけるすべてのプロトコル表示を掲載する。

#### 第 2 章 ENUM 技術概要

ENUM WG は今年度から報告を開始するため、本章では ENUM についての基本的な技術解説を行う。

##### 2.1 ENUM の概要

ENUM は、電話番号に対してインターネット上のサービスに対応させる機構である。電話番号として ITU-T により管理される E.164 電話番号を用い、それをドメイン名に変換し、URI を登録することができるリソースレコードを DNS に定義することで、電話番号に対応する URI を DNS に登録できるようにした。

ENUM の標準化が行われてきている背景として、E.164 電話番号 [64] は全世界でユニークな番号であること、数字のみであるため言語に依存しないこと、またインターネットで音声を運べる環境が整備され、VoIP (Voice over IP) システムが普及してきていることが挙げられる。

そこで、VoIP 端末を (電話) 番号で指定したい、VoIP 端末 (IP 電話) の番号はインターネットのデータベースで管理したい、さらには、インターネットの各種資源を (電話) 番号で指定したいという考えがあり、ENUM のアイデアが生まれていった。

ところで、10 年以上前に遠隔地への印刷プロトコルの標準化を行っていたときのアイデアとして、電話番号に対応する印刷機器を DNS に登録する方式が RFC1486、RFC1528、RFC1529、RFC1530 として提案されている。ドメイン名に対して MX レコードを指定し、email で文書を受け取るという方式であった。ENUM はそれを拡張した機構と考えられる [241]。

ENUM のプロトコルは IETF enum ワーキンググループで標準化され、2000 年 9 月に RFC2916 が Proposed Standard として発行された [242]。RFC2916 では電話番号からドメイン名への変換方式が定義さ

れ、専用のドメイン名として e164.arpa が用意された。また、ドメイン名へのサービス登録方法として、RFC2915 で定義されている NAPTR リソースレコードを用い、また E2U というサービスを用いることが決められた。その後、NAPTR リソースレコードが DDDS という仕組みの中で再定義されたことや曖昧さを減らすために、RFC2916 の改良としての rfc2916bis の議論が行われ、現在標準化手続き中である [59]。

ENUM と同じ方式を、ENUM の TLD である e164.arpa 以外のドメイン名を用いて実装した場合のことを、ENUM ライクという。ENUM ライク DNS を用いて組織内の内線番号の管理をしたり、電話網の経路制御のデータベースを実装するといったことが考えられる。

## 2.2 ENUM の詳細

### 2.2.1 電話番号からドメイン名への変換

ENUM は、ITU-T の決めた E.164 電話番号をドメイン名に対応付ける [242]。

以下、03-5297-2571 (JPRS の代表電話番号) を例として変換方法を示す。

- (1) E.164 番号は、国コードの後に各国の電話番号を付けた番号である。日本の国コードは 81 であり、0 から始まる市外局番を含む電話番号から先頭の 0 を取り、先頭に +81 を付けた番号が E.164 番号である。

```
+81-3-5297-2571
```

(E.164 番号表記、- は不要)

- (2) E.164 電話番号から、先頭の + を除く数字以外の文字 (- など) を削除する。これが ENUM 検索時に用いられる文字列 AUS となる。

```
+81352972571 (AUS)
```

- (3) 次に先頭の + を削除し、数字を逆順に並べ変える。

```
17527925318
```

- (4) 数字の間に "." を挿入する。

```
1.7.5.2.7.9.2.5.3.1.8
```

- (5) 最後に e164.arpa を追加すると ENUM ドメイン名となる。

```
1.7.5.2.7.9.2.5.3.1.8.e164.arpa
```

(ENUM ドメイン名)

### 2.2.2 NAPTR リソースレコード

2.2.1 節で示したドメイン名に対し、RFC3401、

3402、3403、3404 DDDS[245] で定義された NAPTR リソースレコードを用いて URI を登録する。ここで、NAPTR RR は以下のように記述する。

```
label IN NAPTR order preference flags service
regex replacement
```

- order と preference には符合なし整数を書き、数値の小さいものを優先する。preference よりも order を先に評価する。
- 有効なリソースレコードのうち、order 値が最小のエントリを採用する。エントリが複数存在した場合、preference 値が小さいものを用いる。
- flags には S A U P "などの文字を書き、NAPTR リソースレコードの評価方法を示す。ENUM では基本的には U を書き、NAPTR RR は URI を出力する。
- service には、この NAPTR RR エントリが適用されるプロトコル・サービスを指定する。ENUM の場合は E2U+ で始まり、IANA に登録された enumservice 文字列を指定する。
- regex replacement フィールドには AUS からの置換規則を書く。"!パターン!置換文字列!"のように指定する。パターンは POSIX の拡張正規表現で、"()" にマッチした文字列を、置換文字列中の \1 ~ \9 で参照できる。
- replacement フィールドには、ENUM の場合は "." を書く。

### 2.2.3 ENUM で取り扱うと想定されるサービス

現在は、表 2.1 に示される通信サービスについて、ENUM への登録方法が提案され、合意されつつある [57, 58, 60, 61]。

### 2.2.4 登録例

電話番号として、さきほどと同様に JPRS の代表番号を例としていくつかの登録例を示す。AUS は +81352972571 とする。

```
1.7.5.2.7.9.2.5.3.1.8.e164.arpa. IN NAPTR
```

```
100 10 "u" "E2U+sip"
```

```
"!\^+813(.*)$!sip:\1@tokyo.sipisp.jp!" .
```

という DNS 登録があった場合、+81352972571 に対する ENUM 検索結果は、sip:52972571@tokyo.sipisp.jp になる。

```
1.7.5.2.7.9.2.5.3.1.8.e164.arpa.
```

```
IN NAPTR 100 10 "u" "E2U+sip"
```

表 2.1. 各通信サービスと ENUM への登録方法

enum service	URI Scheme	
sip	sip:, sips:	SIP
h323	h323:	H.323
email:mailto	mailto:	email
fax:tel	tel:	PSTN 経由の FAX
sms:tel	tel:	Short Message Service
ems:tel	tel:	Enhanced Message Service
mms:tel	tel:	Multimedia Message Service
web:http	http:	web page
web:https	https:	web page ( TLS )
ft:ftp	ftp:	FTP
ifax:mailto	mailto:	Internet FAX

```
"!^.*$!sip:info@jprs.jp!" .
```

```
IN NAPTR 100 10 "u"
```

```
"E2U+email:mailto"
```

```
"!^.*$!mailto:info@jprs.jp!" .
```

という DNS 登録があった場合、+81352972571 に対する ENUM 検索結果は、sip:info@jprs.jp と mailto:info@jprs.jp となる。

同じ番号でも、SIP にのみ対応している ENUM クライアントから参照した場合、サービスフィールドが E2U+sip であるエントリを採用し、sip:info@jprs.jp を結果とする。

## 2.3 ENUM DNS の管理構造

### 2.3.1 ENUM DNS の管理 [66, 243, 244]

IETF enum ワーキンググループは、電話番号に対して URI を登録する方式を定義しているが、ITU-T と各国政府が電話番号を管理しているため、ENUM DNS の運用については ITU-T・IAB と各国政府が管理責任を持つ。又、ENUM DNS の TLD となる E164.ARPA は Tier0 と呼ばれ、ITU-T と IAB から委任を受けた RIPE NCC が運用している。現在は、トライアルの段階である。

各国の電話番号は階層は Tier1 と呼ばれ、Tier1 レジストリは各国政府の資格確認のもとで Tier0 に登録される。各国内の運用は、各国で決める。

NAPTR リソースレコードを保持する階層を Tier2 と呼ぶ。

日本の場合、総務省が電話番号を管理している。1.8.e164.arpa はまだ登録されていない。

### 2.3.2 ユーザ ENUM とオペレータ ENUM

JPNIC ENUM 研究グループでは、ENUM を、

利用方法により大きくユーザ ENUM とオペレータ ENUM の二つに分類した [63]。

事業者が網の実現のために用いるものをオペレータ ENUM と呼ぶ。IP 電話事業者が ENUM を用いて電話の経路制御を行う場合や、事業者間のナンバーポータビリティを実現するため共有データベースとして用いる場合などがある。

一方、ユーザ ENUM は、電話番号を割り当てられたユーザが希望する内容を ENUM DNS に登録する方式である。1 つの電話番号に対し、電話と email や web など、複数のサービスを登録し、ENUM クライアントによりサービスを選択する。IETF や UKEG, ENUM Forum など一般的に検討されている ENUM はユーザ ENUM である。

## 2.4 ENUM アプリケーション

ENUM アプリケーションは、電話番号を入力すると ENUM を検索し、その結果得られた URI に接続する。

評価手順を以下に示す。

- (1) ユーザが E.164 番号である AUS を入力する。
- (2) AUS からドメイン名へ変換し、DNS クエリを行う。
- (3) DNS 応答のうち、アプリケーションが対応しているサービスのエントリを選ぶ。
- (4) order 値が最も小さいものを選ぶ。
- (5) regexp を評価し、preference 値の小さい順に並べかえ、ユーザに選んでもらう。
- (6) アプリケーションを呼び出す。

現在の携帯電話は、電話と web ブラウザ、email の機能を含んでいる。また、FAX と電話の機能を持つ端末が多い。また PC は、IP 電話機能と email・web ブラウザの機能を持つことが可能である。そこ

で、複数の機能をもつ端末は容易に実現可能である。

そこで、SIP 電話と email 機能を持つ端末を考える。着信を受けたいユーザは、各自の電話番号の ENUM DNS に SIP URI と email アドレスを登録しておくとする。この場合、発信ユーザが SIP と email に対応した ENUM アプリケーションに電話番号を入力し、ENUM 検索を行うと、SIP と email の選択肢を出すことができる。音声での通話をしたい場合、まず SIP により電話をかけてみて、相手が不在の場合には email を送るとすることが可能となる。

---

### 第 3 章 ETJP について

---

日本国内における ENUM トライアル実施のため、JPNIC、JPRS、WIDE Project の三者が発起人となり、2003 年 9 月に ENUM トライアルジャパン (ENUM Trial Japan; ETJP) を設立した。2003 年 12 月末現在で、WIDE Project を含む 42 会員が参加している。

ETJP は、

- ENUM の基盤サービス、その上の通信アプリケーション、通信サービスまでを含めた、基本機能の技術的検証、実用性の技術的検証、サービス化に向けた技術的課題の整理と検討を行う場
- 諸外国の ENUM トライアルと連携し、国際レベルの技術的検証を行う場

を提供し、この場を用い、各会員が ENUM をベースとする通信アプリケーション、通信サービスの技術ノウハウを蓄積することを目的とする。

また、ETJP は全体として 1 つの成果を求めるとでなく、個々の会員が自由に技術実験を行う。すなわち、各会員もしくは複数会員からなるグループが、各自の技術を持ち寄り、実験を独自に企画し、実行する。以下に、主な活動項目を挙げる。

- ENUM トライアル用 DNS の運用
- ENUM を用いる通信機器、ソフトウェアの技術的検証
- ENUM を用いる通信サービスの技術的検証
- ENUM に関する情報の集積

ETJP は 1 年間 (2004 年 9 月末まで) の活動を予定しており、活動期間内のロードマップは以下の通

りである。

- 2003 年 9 月 発足
- 2003 年 9 月 ~ フェーズ 1 (ENUM を用いた通信機器、ソフトウェアの動作検証)
- 2003 年 12 月 ~ フェーズ 2 (ENUM を用いた通信サービスの動作検証)
- 2004 年 4 月 ~ フェーズ 3 (利用者による ENUM 操作まで含めた通信サービス全体の動作検証、他国の ENUM トライアルとの連携)

WIDE Project の ENUM WG メンバー (企業) の多くは ETJP 会員でもあり、ETJP フェーズ 1 での主な成果は 2003 年 9 月の WIDE 秋合宿における ENUM WG で発表されたものの発展形である。以下に ETJP フェーズ 1 で確認されたアプリケーション等を挙げる。

- ETJP 用 NAPTR 登録システム (JPRS、WIDE 用 NAPTR 登録システムを流用)
- ENUM を参照する SIP サーバ (SOFTFRONT、自社製品)
- ENUM を参照する SIP サーバ (JPRS、SIP Express Router (SER) にパッチを当てたもの)
- ENUM を参照する VoIP ルータ (CISCO、自社製品)
- ENUM を参照する VoIP ルータ (YAMAHA、自社製品)

---

### 第 4 章 ENUM テストベッド

---

#### 4.1 概要

ENUM WG では、e164.wide.ad.jp ドメインに、ENUM ライク DNS を運用している。WIDE メンバであれば、WIDE 証明書認証により、ENUM ライク DNS への NAPTR リソースレコードの登録が可能である。

また、ENUM アプリケーションのイメージをつかめるような、Web CGI による ENUM アプリケーションを作成した。さらに、ENUM 検索に対応した SIP サーバを運用している。

#### 4.2 使用する電話番号 (番号規則)

ENUM WG での議論の結果、電話番号のような

ものとして、8桁以上の番号を用いることが決まった。そして、WIDE 番号をもとに、1人当たり5つの番号を割り当てた。また、それ以外の番号についても登録することを可能とした。

### 4.3 ENUM 登録システム

<https://www.e164.wide.ad.jp/>

WIDE メンバ証明書を用いて番号管理を行う登録システムを作成した。番号規則に基づく番号と、任意の番号に対して、NAPTR RR、NS Delegation を登録することができる。

本システムは、メンバ証明書から WIDE 番号を得て、それをもとに認証を行っている。2003 年中には、24 人の利用があった。

#### 4.3.1 トップページ

トップページには登録可能番号へのリンクがあり、以下のように表示される。

You can edit 番号 1 番号 2 番号 3 番号 4  
番号 5

また、トップページから新しい E.164 番号を追加することができる。

New E.164 number be careful!!!

[入力ボックス] [enter]

入力ボックスに 81352972571 のような E.164 番号を入力することで、新しい番号を追加することができる。信用モデルを採用しているため、10桁以上の番号であればどんな番号でも登録することが可能である。また、追加するとすぐに反映される。

次に、登録可能番号を選ばるとリソースレコード編集画面に入る。

#### 4.3.2 リソースレコード編集画面

この画面の入力項目は、入力・編集後、[enter] を押すことで反映される。

- 新規・追加の場合は、空のエントリに記入して [enter] を押す。
- 複数行追加する場合は、毎回 [enter] を押して確定する。
- 削除の場合は [delete] 項目をチェックして [enter] を押す。

前項で追加した番号であれば、[ ] delete this number をチェックして [enter] を押すことで削除できる。また、ここで登録した内容のうち、NS リソースレ

コードと NAPTR リソースレコードのどちらを使用するかを選ぶ。

NAPTR リソースレコードの項目は、それぞれ delete チェックボックス、order、preference、ENUM-service、URI/regexp 情報で、order と preference には符合なし整数を、ENUMservice には E2U+を除いた ENUM service パラメータを、URI/regexp 情報には、URI または、!で始まる NAPTR リソースレコードの regexp フィールドを入力する。

たとえば、order、preference には 0 を、ENUMservice には email:mailto を、URI には mailto:fujiwara@jprs.co.jp のような mail URI を入力する。

NS resource record の入力項目には、ネームサーバのホスト名を入力する。

入力が完了すると、Zone file generation リンクをクリックすることで、zone ファイルを生成して reload を行う。また、View zonefile により、ゾーンファイルを見ることができる。

NS リソースレコードを使って Delegation した場合、正しくネームサーバを動かし、LAME Delegation にならないようにしなければならない。

### 4.4 簡易 ENUM クライアント

携帯電話は、電話、電子メール、ブラウザの機能を持つ。そこで、web CGI により、ENUM クライアント風のを試作し、将来の ENUM 端末を想像できるようなものを作成した。ENUM service としては、voice:tel、email:mailto、web:http のみを評価する。

voice:tel の場合、tel:+81352972571 のような URI を tel:0352972571 に変換してリンクをはることで、携帯電話から日本の電話番号へ発信できる。

http や mailto の場合は、そのままリンクをつけた結果を返す。そのリンクをクリックすることで、Web page を閲覧したり、email を送る画面に移動できる。

### 4.5 ENUM 対応 SIP サーバ

ENUM WG では、SOFTFRONT SIP サーバをベースとした ENUM 対応 SIP サーバのテストを行った。詳細は 5.2 節で報告する。

Windows Messenger 5.0 と YAMAHA RT55i/56v/57i、CISCO SIP Phone から接続できることを確認した。

#### 4.6 クライアントの設定について

YAMAHA RT での SIP 設定と、Windows Messenger での設定について示す。

##### 4.6.1 YAMAHA RT56v/57i

Web interface: 電話設定中の VoIP の設定

- VoIP の基本設定
  - VoIP 機能 使用する
  - TEL ポート 電話ユーザ名、着信指定
  - プリフィックス IP 電話サーバ
- IP 電話サーバの設定 登録
  - sip アドレス: sip:username@domainname (SIP URI)
  - サーバアドレス: SIP サーバのホスト名
  - ユーザ ID: SIP サーバに指定したユーザ名
  - パスワード: SIP サーバに指定したパスワード
  - 発信時のプリフィックス

コマンドライン

```

sip use on
sip server 1 sip-server.hostname register udp\
sip:username@domainname <sip-username>
<パスワード> (一行で)
analog extension dial prefix sip 1
    
```

相手に ENUM で SIP 通話するときは、電話機から電話番号を入れて電話をすることで発信する。指定したプリフィックスをつけることが必要である。

##### 4.6.2 Microsoft Windows Messenger 5.0

- Windows Messenger 4.6, 4.7 では接続できない
- .NET パスポートは不要である
- オプション アカウント SIP コミュニケーションサービスアカウント画面
  - 「SIP コミュニケーションサービスユーザを含める」にチェックを入れる

- サインイン名として SIP URI を入力 (sip:なし)
- 詳細設定画面へ
  - \* 手動設定
  - \* SIP サーバを指定
  - \* 接続に使用しているプロトコルとして UDP を指定

- トップ画面にて、SIP URI をオンラインに指定する
- ユーザ名、パスワードを入力

ENUM による SIP 通話を開始するときは、音声チャットの開始を選び、その他を指定し、次に相手の ENUM 番号を入力する。番号だけでよい。

#### 4.7 Privacy について

ENUM に登録した内容は DNS で公開される。また、電話番号体系とドメイン名がわかると、scan が可能である。

つまり、123456789 という番号を使っていることと、e164.wide.ad.jp というドメイン名がわかれば、誰でも 9.8.7.6.5.4.3.2.1.e164.wide.ad.jp を引くことができる。

逆に、通話したい相手にはこれらの情報を教えることになる。そのため、登録する内容は公開されるという前提で登録する必要がある。

#### 4.8 システム構成

ENUM テストベッドのシステム構成を図 4.1 に示す。

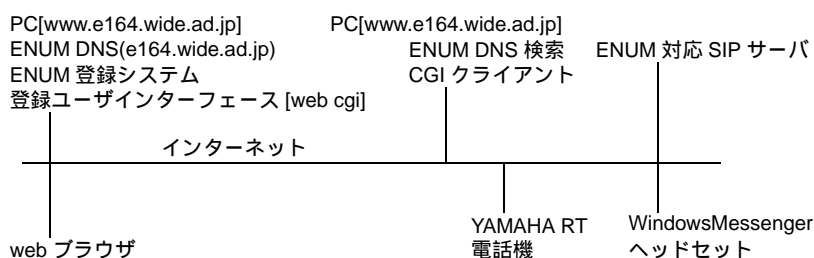


図 4.1. ENUM テストベッドのシステム構成



---

## 第 5 章 ENUM アプリケーション

---

### 5.1 CGI による ENUM クライアント

Web CGI により、ENUM クライアントのように見えるものを作成した。これについては、第 4 章で報告している。

### 5.2 SIP サーバの ENUM 機能

#### 5.2.1 はじめに

SOFTFRONT の SIP サーバと、その ENUM 機能拡張について述べる。まず、SIP サーバの概要と、その主な機能について紹介する。次に、このサーバをベースとして拡張として実現した ENUM 機能について、機能の概要と動作を説明する。さらに、ENUM WG の活動として本サーバを半年にわたって運用した際の運用状況と、その運用から得られた知見を紹介する。

#### 5.2.2 SOFTFRONT SIP サーバの特徴

SOFTFRONT SIP サーバは、RFC3261 に基づいた SIP サーバの実装である。プロキシやレジストラなどの基本サーバ機能を核とし、さらに拡張によってプレゼンスなどの SIP アプリケーションサーバを構築することを可能としている。

SIP プロキシの基本機能は、受信したリクエストを受け取り、ポリシーに基づき宛先を書き換えて、新たな宛先に送信することである。これを SIP メッセージのルーティングと呼んでいる。本 SIP サーバでは、独自のルーティングを行うための拡張インターフェースを用意している。

この拡張インターフェースを使用して、ENUM モジュールを追加することにより、以下に述べる ENUM 拡張を実装している。

#### 5.2.3 SIP プロキシの ENUM 拡張

SIP サーバ(プロキシ)が ENUM 機能を持つ事により、従来の ENUM に対応しないクライアントをそのまま使用しながら、ENUM を使用するシステムを構築することができる。しかしながら、SIP サーバによる ENUM 機能では、sip:以外の URI に対応

することが困難であり、クライアントが(SIP 以外の)プロトコルまたはアプリケーションを選ぶことができる ENUM の利点が失われることに注意する必要がある。

#### ENUM によるルーティング

ENUM 機能においては、SIP サーバ(プロキシ)がルーティングとして ENUM を利用する。ENUM とは、番号の入力に対して URI を出力することである。プロキシが電話番号を受け取った場合に、サーバが ENUM による解決を行って URI を取得し、さらに取得した URI を、SIP クライアントの登録アドレスへの解決など通常のプロキシ処理を行うことにより、SIP プロキシの ENUM 機能を実現することができる。

#### 電話番号の URI 表現

SIP サーバはクライアントからリクエストを受け取る。リクエストには、宛先が URI 形式で含まれている。電話番号を宛先とするためには、電話番号を URI 形式で表現する必要がある。

そのため、以下のような形式を使用する。電話番号にはハイフンを含めることができる。また E.164 形式の電話番号の場合には、+記号と国番号を付加する。また、sip:を使用する場合、ドメイン名は任意である。

```
tel:012345
tel:+81-3-3456-7890
sip:012345@domain.com
sip:+81-3-3456-7890@domain.com
```

#### 電話番号の処理

電話番号をドメイン名に変換する規則は次の通りである。電話番号からハイフンを取り除き、桁毎にピリオドで区切ってさらに逆順にして、TLD (Top Level Domain) を付加する。

- 1) tel:012-3456 電話番号を含む URI
- 2) 012-3456 電話番号を取り出す
- 3) 0123456 ハイフンを取り除く
- 4) 0123456 先頭に+記号があれば取り除く
- 5) 6.5.4.3.2.1.0 逆順にしてピリオドで区切る
- 6) 6.5.4.3.2.1.0.e164.wide.ad.jp.

TLD (Top Level Domain) を付加する  
使用する TLD は設定により与えることができる。

表 5.1. SIP サーバの利用状況

	2003 年 8 月	9 月	10 月	11 月	12 月	1 月
REGISTER 回数*	5376	14982	6924	5395	7047	6741
INVITE 回数*	13	936	45	0	2	0
ENUM 検索回数	6	556	4	0	1	0

\* REGISTER/INVITE 回数は、認証処理のため、少なくとも実際の数の倍の数がカウントされている。また、認証に失敗しているものも含まれている。

ここでは、e164.wide.ad.jp. を TLD としている。  
後に述べる正規表現による置換は、3) の形式に適用される。

**NAPTR レコードの解釈形式**

NAPTR {order} {pref} {flag} {service} {regex} {replace}

**例**

NAPTR 0 0 "u" "E2U+sip"  
"!^.\*\$!sip:vtr.alice@domain2.com!" .

order 16 bit 符号無し整数 小さいものを使用 (pref より優先)

pref 16 bit 符号無し整数 小さいものを優先

flags "u" URI を出力 (他に "a", "s", "p" など)

service このエントリが適用されるサービスを指定 (例: E2U+sip、E2U+web:http 等)

regex 正規表現による置換規則

replace 再度 NAPTR 検索する場合のドメイン名 (無視)

1 つのドメイン名に対して、複数の NAPTR レコードを定義することができる。また、サービス種別や優先度により適用するレコードやその順序を制御することができる。SIP サーバから ENUM を利用する場合には、検索された NAPTR レコードのうち、サービス種別が "E2U+sip" のレコードを選択する。

**5.2.4 WIDE における運用実験**

この ENUM 対応機能を持つ SIP サーバを、ENUM WG 内で運用実験を行った。ENUM WG の ML にて 2003/8/23 にアナウンスを行い、利用可能な SIP 端末を持っている有志に利用してもらった。

**SIP サーバの設定概要**

運用した SIP サーバは、IPv4/IPv6 の両方を利用可能とした。トランスポートは UDP のみを利用可

表 5.2. 日別 ENUM 検索回数 5 傑

2003-09-09	260 回
2003-09-27	62 回
2003-09-03	40 回
2003-09-11	32 回
2003-09-16	27 回

能とし、TCP/TLS による利用は disable した。

SIP サーバの管理する SIP URI のドメインは、特に限定せずに、いかなるドメインも収容可能な設定とした。これはどのような SIP クライアントが接続してくるか予想がつかないため、なるべく接続を可能とするようにしたためである。また、ドメインを限定しないことから、単一のサーバにすべての SIP クライアントを収容するモデルでの利用に限定される。

悪意のあるアクセスを防止するために、パスワードによる認証を行うこととした。REGISTER による登録の際と、INVITE による発信の際の両方について、SIP サーバによる認証を行うこととした。

ENUM による解決を有効に設定したが、ENUM 機能を利用せずに通常の SIP サーバとして利用することも可能である。

**SIP サーバの利用状況**

このサーバの半年にわたる利用状況を簡単にまとめた (表 5.1、表 5.2)。

- 運用期間: 2003/8/23-(2004/1月現在 運用継続中)
- 利用端末数 (一意な登録 SIP URI 数): 48 個
- 使用された発信先番号数: 46 個

アナウンス直後の 1 ヶ月ほどが主な利用期間であった。半年経過後も運用は継続しているものの、ほぼ利用停止状態である。

もっとも利用が多かったのが、WIDE 合宿の期間中である 9 月 9 日に 1 日あたり最大利用回数 260 回 (ENUM 検索回数) を記録している。そのほかの利

用としては、ETJP のデモのために利用されたことがあった。

REGISTER の回数の多さが際立つが、SIP 端末が動作中に電話をしなかったとしても、定期的に REGISTER を送信するためである。最近、利用はほぼ停止状態にもかかわらず REGISTER だけが継続されているのは、電源が入れたままになっている端末が存在するからであると思われる。

#### 運用から判明した知見

本サーバの運用では、単一のサーバにすべての端末を収容することを前提とし、サーバがカバーするドメインを定めることをしなかった。また、REGISTER と INVITE についてパスワードによる認証を行った。しかし、利用者側で独自に設置したサーバを経由して、本サーバへのアクセスを行った事例があり、この際に認証を要求してしまうという問題が生じた。本来想定していなかったケースではあるが、今後の複数のサーバによる運用に向けて参考になる知見であると考えられるので紹介する。

ドメイン内の UA がプロキシを使用する場合には、そのドメインで有効なパスワードを使用して認証することができる。これは REGISTER による登録や INVITE による発信のどちらにもあてはまる。ところが、ドメイン外の UA がプロキシを経由してドメイン内の UA に発信しようとした場合には、INVITE に対してプロキシは認証を要求してはならない。なぜなら、ドメイン外の UA は、プロキシのドメイン内で有効なパスワードを知らないはずだからである。このように、複数のドメインが連係するような運用では、単純に INVITE に対して認証を行うことができない。

単純にドメイン外からのリクエストについて認証を行わないとすると、悪意のあるアクセスを防ぐことができない。そのために、ドメイン外からのリクエストを受けとる際には、なんらかの方法で信用できる相手 (SIP サーバ) を識別し、信用できる相手からのみ、認証無しにリクエストを受けとることができなければならない。相手を識別するためには、簡易的な方法ではあるが IP アドレスを使用することが考えられる。その他には、TLS 接続を用いて証明書によるサーバ間の認証を行う方法も考えられる。これらの方法によって、SIP サーバ間の信頼関係を 1 つ 1 つ設定していくことが、安全な SIP サーバ網を構築するためには必要であると考えられる。

#### 5.2.5 おわりに

SOFTFRONT の SIP サーバとその ENUM 機能についての紹介をした。また、SIP サーバによる ENUM の処理を解説し、そのシーケンスを示した。さらに、ENUM WG において設置したサーバとその利用状況と得られた知見を紹介した。

### 5.3 インターネット電話 (H.323) とインターネット FAX (T.37) 端末を利用した ENUM 実験

#### 5.3.1 概要

本実験では、NAPTR レコードを複数記述した ENUM アドレスから、最適なプロトコルのアドレスを検索する ENUM クライアントを実装し、接続実験を行った。

#### 5.3.2 背景

ENUM では、NAPTR レコードへさまざまなアプリケーションの登録が可能である。1 つの ENUM アドレスに複数のプロトコルを記述して、最適なプロトコルを選択して接続する必要があり、NAPTR レコードの選択方法の検証が必要であると考えられる。

インターネット電話 (H.323[94]) とインターネット FAX (T.37[276]) の 2 つのプロトコルが利用可能な機器である DN-C100 に実装し実験を行った。

#### 5.3.3 実装

ENUM の NAPTR レコードは、下記の named フォーマットで記述される。

```
IN NAPTR order preference flags service
  regexp replacement
```

NAPTR では、機能識別のために service 名称を定義しており、機能名称として下記の名称を採用した。

- H.323 の service 名称 (ENUM Service Registration for H.323 URL[57]) E2U+h323
- T.37 のサービス名称 (IFAX service of ENUM[46]) E2U+ifax

#### ENUM クライアントの実装

機器 (DN-C100) における NAPTR レコードの検索について、1 つの ENUM 番号に h323 と ifax の URI を登録し、通信方法を自動選択する。

機器 (DN-C100) に、下記のアドレスを設定した。

```
機器 1 (ENUM 番号 10001)
H.323: tel@iptu1.enum.jp
```

T.37: fax@iptu1.enum.jp  
 機器 2 ( ENUM 番号 10009 )

H.323: tel@iptu2.enum.jp  
 T.37: fax@iptu2.enum.jp

機器 ( DN-C100 ) の T.37 と H.323 を切り替えるため、T.37 を使用する場合、ENUM 番号の前に 2 を付加した。

ENUM クライアントの ENUM アドレス検索方法を以下に示す。

- 1) order, preference によるソート
- 2) 機器が利用可能なプロトコルを service から検索
- 3) 電話番号を regexp に適用して URI 取得
- 4) URI を元に、相手に接続する

ENUM サーバの NAPTR レコードの登録

ENUM サーバの設定

```
$ORIGIN enum.jp.
1.0.0.0.1 NAPTR 100 10 "u" "E2U+h323"
"!^.*$!h323:tel@iptu1.enum.jp!" .
NAPTR 100 10 "u" "E2U+ifax"
"!^.*$!mailto:fax@iptu1.enum.jp!" .
NAPTR 99 10 "u" "E2U+mailto"
"!^.*$!mailto:honma@wide.ad.jp!" .
1.0.0.0.1.2 NAPTR 100 10 "u" "E2U+h323"
"!^.*$!h323:tel@iptu1.enum.jp!" .
NAPTR 100 10 "u" "E2U+ifax"
"!^.*$!mailto:fax@iptu1.enum.jp!" .
NAPTR 99 10 "u" "E2U+mailto"
"!^.*$!mailto:honma@wide.ad.jp!" .
9.0.0.0.1 NAPTR 100 10 "u" "E2U+h323"
"!^.*$!h323:tel@iptu2.enum.jp!" .
NAPTR 90 10 "u" "E2U+mailto"
"!^.*$!mailto:honma@wide.ad.jp!" .
NAPTR 100 10 "u" "E2U+ifax"
"!^.*$!mailto:fax@iptu2.enum.jp!" .
9.0.0.0.1.2 NAPTR 100 10 "u" "E2U+h323"
```

```
"!^.*$!h323:tel@iptu2.enum.jp!" .
NAPTR 90 10 "u" "E2U+mailto"
"!^.*$!mailto:honma@wide.ad.jp!" .
NAPTR 100 10 "u" "E2U+ifax"
"!^.*$!mailto:fax@iptu2.enum.jp!" .
5.0.0.0.1.2 NAPTR 100 10 "u" "E2U+mailto"
"!^.*$!mailto:honma@wide.ad.jp!" .
```

5.3.4 実験

本実験では、図 5.1 の構成でローカルネットワーク環境に、2 台の DN-C100 ( H.323+T.37 ) と DNS サーバ( ENUM )と SMTP サーバと PC を用意した。

(1) H.323 の接続実験

機器 1 から ENUM で機器 2 のアドレスを検索し、H.323 アドレスを取得して接続する。

(2) T.37 の接続実験 ( 機器 1 ・ 機器 2 )

機器 1 から ENUM で機器 2 のアドレスを検索し、T.37 アドレスを取得して機器 2 へ送信する。

(3) T.37 の接続実験 ( 機器 1 ・ PC )

機器 1 から ENUM で PC のメールアドレスを検索し、T.37 のデータを PC のメールアドレスへ送信する。

本実験では、H.323 ゲートキーパを使用しない。

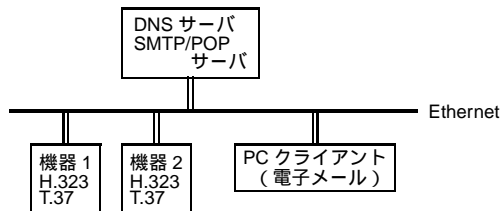


図 5.1. 構成図

5.3.5 実験結果

ENUM の登録確認

dig コマンドを使用して、NAPTR レコードを検索する。

```
elements117; <<>> DiG 8.3 <<>> 1.0.0.0.1.e164.enum.jp NAPTR
;; res options: init recurs defnam dnsrch
;; got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 13474
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2, ADDITIONAL: 0
;; QUERY SECTION:
;; 1.0.0.0.1.e164.enum.jp, type = NAPTR, class = IN
```

```

;; ANSWER SECTION:
1.0.0.0.1.e164.enum.jp. 1H IN NAPTR 100 10 "u" "E2U+h323" "!^.*$!h323:tel@iptu1.enum.jp!" .
1.0.0.0.1.e164.enum.jp. 1H IN NAPTR 100 10 "u" "E2U+ifax" "!^.*$!mailto:fax@iptu1.enum.jp!" .
1.0.0.0.1.e164.enum.jp. 1H IN NAPTR 99 10 "u" "E2U+mailto" "!^.*$!mailto:honma@wide.ad.jp!" .

;; AUTHORITY SECTION:
e164.enum.jp. 1H IN NS elements.enum.jp.

;; Total query time: 1 msec
;; FROM: elements.enum.jp to SERVER: 127.0.0.1
;; WHEN: Tue Oct 22 19:40:42 2003
;; MSG SIZE sent: 47 rcvd: 226

```

#### (1) H.323 の接続結果

- 1) 機器 1 から 10009 (機器 2) をダイヤルする。
- 2) 9.0.0.0.1.enum.jp の NAPTR レコードを DNS サーバに問い合わせる。
- 3) NAPTR レコードから、サービスを検索する。E2U+H323 の登録がない場合、エラーとなり、ビジー音を発生させる。
- 4) DNS サーバからのレスポンスを検索、regexp フィールドから URI を取り出す。
- 5) URI のホスト名から IP アドレスを DNS サーバへ問い合わせる。
- 6) H.323 により機器 2 へ接続する。
- 7) 機器 1 と機器 2 での通話が成功したことを確認した。

#### (2) T.37 の接続結果 (機器 1・機器 2)

- 1) 機器 1 から 210009 (機器 2) をダイヤルする。
- 2) 9.0.0.0.1.2.enum.jp の NAPTR レコードを DNS サーバに問い合わせる。
- 3) NAPTR レコードから、サービスを検索する。E2U+ifax を探した後、E2U+mailto を検索し、発見できない場合には、ビジー音を発生させる。
- 4) DNS サーバからのレスポンスを検索、regexp フィールドから URI を取り出す。
- 5) メールアドレスで T.37 のデータを SMTP サーバに送信する。
- 6) SMTP サーバから機器 2 へ T.37 のデータを送る。
- 7) 機器 1 から送った FAX データが機器 2 より印刷されたことを確認した。

#### (3) T.37 の接続結果 (機器 1・PC)

- 1) 機器 1 から 210005 (機器 2) をダイヤルする。
- 2) 5.0.0.0.1.2.enum.jp の NAPTR レコードを

DNS サーバに問い合わせる。

- 3) NAPTR レコードから、サービスを検索する。E2U+ifax を探した後、E2U+mailto を検索し、発見できない場合には、ビジー音を発生させる。
- 4) DNS サーバからのレスポンスを検索、regexp フィールドから URI を取り出す。
- 5) メールアドレスで T.37 のデータを SMTP サーバに送信する。
- 6) PC が SMTP/POP サーバからメール (T.37 のデータ) を取り出す。
- 7) 機器 1 から送った FAX データが PC のメールクライアントで表示されることを確認した。

#### 5.3.6 まとめ

ENUM クライアントによる、H.323、T.37 の相互通信テストと T.37 から PC のメールクライアントへの送信も問題なく利用できた。

regexp の実装は行ったが、送信先の振り分けやサーバの変更などの実験には至らなかった。今後、tier や regexp などを含めた実験を進めていく必要があると考える。

### 5.4 ENUM 対応 SIP UA による通話とプロトコル表示デモ

#### 5.4.1 概要

WIDE ENUM トライアル実験の一環として、2003 年 9 月の WIDE 合宿において、ENUM 対応 SIP User Agent (perl script によるソフトフォン) による通話とプロトコル表示解説のデモを行った。

ここではその目的ならびにデモにおけるプロトコル表示と解説を示す。

## 5.4.2 目的と特徴

ENUM は電話番号をインターネットアドレス (URI) へと対応させる機構である (2 章)。例えば、電話などのリアルタイムコミュニケーションの通信では、SIP URI が用いられる。この SIP アドレス以外にも、メールアドレスやウェブアドレスなどを自由に ENUM 機構で登録し、電話番号と対応づけて用いることができる。これらの SIP 以外の機能を用いて活用するには、ENUM 解決を SIP サーバに任せる (5.2 節) のではなく、UA 自身が ENUM 解決をすることが望ましい。

SIP UA は、指定された SIP アドレスの相手へ通信を行うとき、自分ではその SIP アドレスの解決をせずに、SIP 代理サーバに解決を任せることもできる。しかし、その場合はどの SIP 代理サーバに任せるとかという問題があり、あらかじめ UA に登録しておいたり、誰かに必ず教えてもらう必要がある。また、通信相手が通信可能な状況でも、それと直接関係ない SIP 代理サーバとの通信が不能であれば、影響を受けて通信ができなくなってしまう問題もある。

本 UA では、これらの問題に対応する目的と、ならびに、ENUM および SIP において DNS を用いてどのように通信が行われるかという各プロトコルの挙動理解を普及する目的から、以下の方針にて UA を設計作成してデモで用いた。

- UA 自身が ENUM 関係の DNS を自分で引いて処理を行う
- UA 自身が SIP 関係の DNS を自分で引いて処理を行う

この方針により、UA 自体は、全く特定のホストや IP アドレスなどの情報を設定として必要せずに、与えられた電話番号あるいは SIP URI の情報のみから、DNS を用いて通信先の解決を行うことを実現している。

なお、今回は WIDE ENUM トライアル実験として、WIDE の ENUM テストベッド (4 章) を用いたため、UA としては、e164.arpa の代わりに e164.wide.ad.jp を ENUM 用ドメインの情報としてあらかじめ保持している。

また、今回の特殊な事情として、実験で用いられた sip1.e164.wide.ad.jp ドメインを管轄する SIP サーバが、外から user@sip1.e164.wide.ad.jp 宛への INVITE リクエストでもパスワード認証を必要とする形で用いられたため、本来ならば必要としない、通

信相手側の管轄 SIP サーバのパスワード情報をあらかじめ持つ必要があった。

## 5.4.3 デモ環境

デモ環境としては、研究会合宿の一般ネットワークの上で行われ、ENUM や SIP などの DNS 設定は WIDE の ENUM テストベッド環境が用いられた。また、作成して用いた ENUM 対応 SIP UA とネットワーク接続の情報は表 5.3 の通りである。

表 5.3. デモ環境

使用機器:	ThinkPad X30 + ヘッドセット
使用 OS:	FreeBSD 4.8-RELEASE
記述言語:	Perl script で作成
ネットワーク:	研究会合宿の無線 LAN 利用

通信相手側は、SIP 対応で G.711 による音声通信ができれば対応することもできるが、今回のデモにおいては YAMAHA の RT に電話機を接続したものが用いられた。

## 5.4.4 デモ時のプロトコル画面表示と解説

今回のデモにおいては、ENUM 対応 SIP UA がまわりにはない状況において、電話番号指定にて通話がきちんとできることを示すとともに、まだ少数の人にしか知られていない ENUM と SIP のプロトコルについて実際どのようなしくみで行われるかを示すことも目的となっている。ここでは、実際にデモ画面に出されたプロトコルシーケンスとその解説について、以下に簡単に示す。

- (1) 指定された電話番号から対応する URI への解決

```
command> enum 9433-8351
WIDE の ENUM トライアル実験用の電話番号
9433-8351 へと発信指示
>>> NAPTR 1.5.3.8.3.3.4.9.e164.wide.ad.jp
『9433-8351』をひっくり返した『1.5.3.8.3.3.4.9.
e164.wide.ad.jp』で NAPTR レコードを問い合わせる (e164.wide.ad.jp は ENUM 用実験ドメ
イン)。
<<< NAPTR 10 20 "u"
    "E2U+email:mailto"
    "!"^.*$!mailto:fujiiwara@wide.ad.jp!" .
<<< NAPTR 10 30 "u" "E2U+web:http"
    "!"^.*$!http://www.wide.ad.jp/!" .
```

```
<<< NAPTR 10 10 "u" "E2U+sip"
"!^.*$!sip:fujiwara@sip1.e164.wide.ad.jp!" .
```

NAPTR レコード 3 つ ( メールとウェブと SIP アドレス ) を得る。

```
--- SIP URI =
sip:fujiwara@sip1.e164.wide.ad.jp
```

ここではそのうち SIP アドレス

『sip:fujiwara@sip1.e164.wide.ad.jp』を選択

(2) 得られた SIP URI から対応する SIP 通信先への解決

```
>>> NAPTR sip1.e164.wide.ad.jp
<<< NAPTR 0 0 "s" "SIP+D2U"
" _sip._udp.sip1.e164.wide.ad.jp.
```

```
OK: SIP NAPTR =
_sip._udp.sip1.e164.wide.ad.jp
(<= sip1.e164.wide.ad.jp)
```

ドメイン部 『sip1.e164.wide.ad.jp』により SIP 通信先の解決を図る。これにより、UDP で SIP 利用可能な SRV 用の『\_sip.\_udp.sip1.e164.wide.ad.jp』が返る

```
>>> SRV _sip._udp.sip1.e164.wide.ad.jp
<<< SRV 0 0 5060 sip1.e164.wide.ad.jp.
OK: SIP SRV = sip1.e164.wide.ad.jp/5060
(<= _sip._udp.sip1.e164.wide.ad.jp)
```

得られた 『\_sip.\_udp.sip1.e164.wide.ad.jp』で SRV を引き、サーバホスト名 『sip1.e164.wide.ad.jp』とポート番号 『5060』を得る。

```
>>> AAAA/A sip1.e164.wide.ad.jp
<<< AAAA/A 2001:240:2f:300::100
(<= sip1.e164.wide.ad.jp)
```

サーバ 『sip1.e164.wide.ad.jp』の IP アドレス 『2001:240:2f:300::100』を得る。

(3) 得られた SIP 通信先への INVITE リクエストとセッション確立

```
>>> INVITE
sip:fujiwara@sip1.e164.wide.ad.jp
(Cseq 30165) [to 2001:240:2f:300::100/5060]
```

『sip:fujiwara@sip1.e164.wide.ad.jp』の INVITE リクエストを、得られた SIP 通信先の IP アドレスとポート番号の元へ投げる。

```
<<< 407 Proxy Authorization Required
```

```
(CSeq 30165 INVITE)
[from 2001:240:2f:300::100/5060]
```

```
Proxy-Authenticate: Digest
realm="WIDE ENUM WG",
nonce="cfb897414240a0e91dd3edbb2a2d7985",
opaque="", qop="auth,auth-int",
stale=FALSE, algorithm=MD5
```

```
>>> ACK sip:fujiwara@sip1.e164.wide.ad.jp
(Cseq 30165) [to 2001:240:2f:300::100/5060]
```

SIP サーバから認証エラーとチャレンジ情報が返る。チャレンジ-レスポンス型なため、1 回目は必ず一旦エラーとなる (今回は、たまたま認証が必要な相手との通信例であることに注意)

```
>>> INVITE
sip:fujiwara@sip1.e164.wide.ad.jp
(Cseq 30166) [to 2001:240:2f:300::100/5060]
```

```
Proxy-Authorization: Digest
username="wide",
realm="WIDE ENUM WG",
nonce="cfb897414240a0e91dd3edbb2a2d7985",
opaque="",
uri="sip:fujiwara@sip1.e164.wide.ad.jp",
response="67feb09491f0a65fe7493f56109e5423"
```

パスワードにより計算した認証データ付で再び INVITE を投げる。

```
<<< 180 Ringing (CSeq 30166 INVITE)
[from 2001:240:2f:300::100/5060]
```

『180 Ringing』が返ってくる (通常は、180 の前に 『100 Trying』が入るが、SIP サーバが返さなかった)。この状況では相手側の電話の呼び出しがずっと鳴っている。

```
<<< 200 OK (CSeq 30166 INVITE) [from
2001:240:2f:300::100/5060]
```

相手側で電話の受話器をとると、相手から 『200 OK』がやって来る。

```
>>> ACK sip:fujiwara@sip1.e164.wide.ad.jp
(Cseq 30166)
[to 2001:200:0:ff20:2a0:deff:fe11:7373/5060]
```

相手へ ACK を返し、これで通話のためのやりとりが成立する。

```
=== RTP: 2001:200:0:ff20:205:3cff:fe06:2498/1139
```

<=> 2001:200:0:ff20:2a0:deff:fe11:7373/5004

実際の通話の音声は、双方向の RTP によって実現される。そして、この RTP のための情報は INVITE/200 OK/ACK において SDP で指定されている。どのような SDP がやりとりされたかは付録 B を参照されたい。

(4) 通話セッションの終了

```
<<< BYE
sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]:5060
(CSeq 30) [from 2001:240:2f:300::100/5060]
相手側にて受話器をおろして電話を切ると、BYE
リクエストがやってくる。
>>> 200 OK (CSeq 30)
[to 2001:240:2f:300::100/5060]
XXX RTP nothing
こちらから『200 OK』を返すことで、SIP のセ
ッション完了し、RTP (音声) も切られる。以上
で、電話における通話が完了する。
```

第 6 章 合宿での ENUM/SIP デモンストレーション 報告

ENUM WG では、2003 年 9 月の WIDE 合宿にて BOF を開催し、BOF 中と、夜の議論タイムに ENUM と SIP、さらにインターネット電話についてのデモンストレーションを行った。

議論タイムに自由に試す事ができる状態で設置したところ、のべ 29 人に利用していただき、好評であった。

6.1 ENUM/SIP デモンストレーション

ENUM ライク DNS サーバと登録システムは ENUM テストベッドで用意したものをを用いた(4章参照)。SIP サーバは SIP サーバの ENUM 機能で用意したものをを用いた(5.2 節参照)。

合宿ネットワークには、SIP 端末だけを置く。ENUM ライク DNS サーバ・登録システム・SIP サーバは、合宿ネットの外のインターネットに存在する。

SIP 端末として、WindowsXP の PC と、YAMAHA の VoIP 機能つきルータ RT57i、RT56v に電話機を接続したものをを用いた(図 6.1)。

RT1、RT2、RT3、PC1 にはそれぞれ SIP URI を与えてあり、SIP サーバに REGISTER しておいた。また、それぞれの機器の SIP URI を ENUM ライク DNS に登録し、番号で呼び出せるようにした。

この環境で、電話機 1、2、3 と PC1 の間で相互に番号で呼び出せることを示した。

また、ENUM 対応 SIP UA による通話とプロトコル表示デモ(5.4 節)で作成されたクライアントとも相互接続の実験を行った。

6.2 インターネット電話のデモンストレーション

フュージョンの 050 インターネット電話サービスのデモンストレーションを行った。以下のように、camp-net からフュージョンの SIP サーバに接続した。デモンストレーション当時は、フュージョンのインターネット電話へは公衆網からは着信できなかったため、発信できること、問題なく通話できることを確認した。

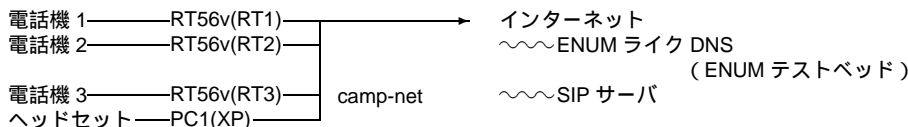


図 6.1. ENUM/SIP デモンストレーションに用いた機器

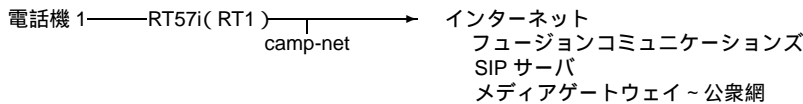


図 6.2. インターネット電話のデモンストレーションに用いた機器

W I D E P R O J E C T 2 0 0 3 a n n u a l r e p o r t



付録 A

ENUM を利用するシーケンス例として、alice が 012345 という bob の電話番号に向かって発信した例を示す。

F1

```
INVITE tel:012345 SIP/2.0
To: tel:012345
From: sip:alice@domain1.com;tag=456248
Call-ID: 843817637684230@998sdasdh09
Via: SIP/2.0/UDP
[2001:0DB8:1234::10]:5060;
```

```
branch=z9hG4bKnashds8
CSeq: 1 INVITE
Contact: sip:[2001:0DB8:1234::10]:5060
Content-Type: application/sdp
Content-Length: ...
```

•説明

alice が電話番号 012345 を宛先として発信する。宛先は tel:形式の URI を使用している。宛先として使用するののは先頭行にある URI(リクエスト URI) であり、To ヘッダの URI ではないことに注意する。

F2

```
DNS Query
5.4.3.2.1.0.e164.wide.ad.jp. NAPTR?
```

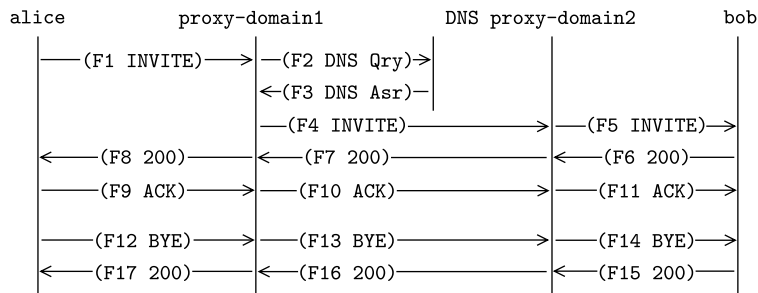
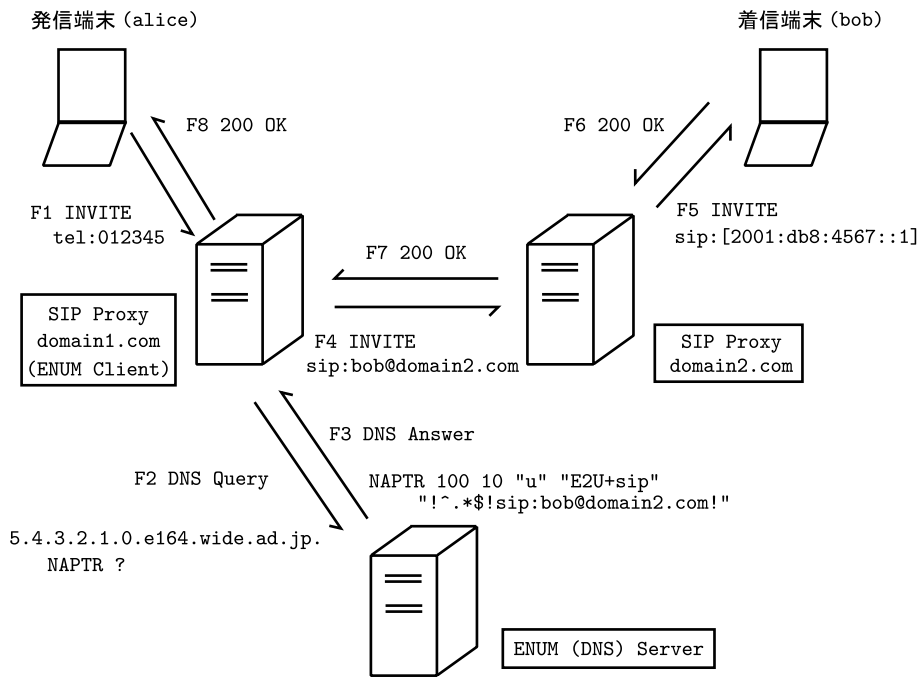


図 A.1. ENUM を利用するシーケンス例

- 説明  
電話番号を宛先としたリクエストを受け取った SIP サーバは、ENUM の規則に従いドメイン名を組み立てて、DNS へ問い合わせを送る。tel:012345 から電話番号を取りだし、ENUM のドメイン名に変換（桁毎にピリオドで区切り逆順にする）し、その NAPTR レコードを問い合わせる。

**F3**

```
DNS Answer
5.4.3.2.1.0.e164.wide.ad.jp.
NAPTR 0 0 "u" "E2U+sip"
"!^.*$!sip:bob@domain2.com!"
5.4.3.2.1.0.e164.wide.ad.jp.
NAPTR 0 0 "u" "E2U+email"
"!^.*$!mailto:bob@domain2.com!"
```

- 説明  
DNS から応答を受け取る。受け取った複数のレコードから、サービスとして SIP プロキシが対応可能な "E2U+sip" を選択する。NAPTR レコードから得られた正規表現の書換規則 "!^.\*\$!sip:bob@domain2.com!" を、電話番号文字列 "012345" に適用する。この例では全置換となり、URI として、sip:bob@domain2.com を得る

**F4**

```
INVITE sip:bob@domain2.com SIP/2.0
To: tel:012345
From: sip:alice@domain1.com;tag=456248
Call-ID: 843817637684230@998sdasdh09
Via: SIP/2.0/TLS
[2001:0DB8:2345::5060]:5060;
branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP
[2001:0DB8:1234::10]:5060;
branch=z9hG4bKnashds8
CSeq: 1 INVITE
Contact: sip:[2001:0DB8:1234::10]:5060
Record-Route:
<sip:[2001:0DB8:2345::5060]:5060;lr>
Content-Type: application/sdp
Content-Length: ...
```

- 説明  
ENUM により取得した URI が、自ドメインで

はなく他のドメインなので、リクエスト URI を置き換えたとえば、domain2.com のプロキシに送る。

**F5**

```
INVITE sip:[2001:0DB8:4567::10]:5060 SIP/2.0
To: tel:012345
From: sip:alice@domain1.com;tag=456248
Call-ID: 843817637684230@998sdasdh09
Via: SIP/2.0/UDP
[2001:0DB8:3456::5060]:5060;
branch=z9hG4bK8924b.1
Via: SIP/2.0/UDP
[2001:0DB8:2345::5060]:5060;
branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP
[2001:0DB8:1234::10]:5060;
branch=z9hG4bKnashds8
CSeq: 1 INVITE
Contact: sip:[2001:0DB8:1234::10]:5060
Record-Route:
<sip:[2001:0DB8:3456::5060]:5060;lr>
Record-Route:
<sip:[2001:0DB8:2345::5060]:5060;lr>
Content-Type: application/sdp
Content-Length: ...
```

- 説明  
REGISTER されている sip:bob@domain2.com のアドレスを取得し、リクエスト URI を書きかえて着信 UA ( bob ) に送る。

**F6**

```
SIP/2.0 200 OK
To: tel:012345;tag=893219
From: sip:alice@domain1.com;tag=456248
Call-ID: 843817637684230@998sdasdh09
Via: SIP/2.0/UDP
[2001:0DB8:3456::5060]:5060;
branch=z9hG4bK8924b.1
Via: SIP/2.0/UDP
[2001:0DB8:2345::5060]:5060;
branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP
[2001:0DB8:1234::10]:5060;
branch=z9hG4bKnashds8
CSeq: 1 INVITE
```

Contact: sip:[2001:0DB8:1234::10]:5060

Record-Route:

<sip:[2001:0DB8:3456::5060]:5060;lr>

Record-Route:

<sip:[2001:0DB8:2345::5060]:5060;lr>

Content-Type: application/sdp

Content-Length: ...

•説明

INVITE を受諾し、200 OK 応答を返す。Via ヘッダに従って発信 UA (alice) に転送 (F7、F8) される。

**F7**

200 OK

**F8**

200 OK

**F9**

ACK sip:[2001:0DB8:4567::10]:5060 SIP/2.0

To: tel:012345;tag=893219

From: sip:alice@domain1.com;tag=456248

Call-ID: 843817637684230@998sdasdh09

Via: SIP/2.0/UDP

[2001:0DB8:1234::10]:5060;

branch=z9hG4bKnashds8

CSeq: 1 ACK

Contact: sip:[2001:0DB8:1234::10]:5060

Route: <sip:[2001:0DB8:2345::5060]:5060;lr>

Route: <sip:[2001:0DB8:3456::5060]:5060;lr>

Content-Length: 0

•説明

200 OK を受け取ったことを通知するために ACK を送る。そして、Route ヘッダに従って着信 UA に配送 (F10、F11) される。このとき、ENUM による解決は不要であり、行われない。

**F10**

ACK

**F11**

ACK

**F12**

BYE sip:[2001:0DB8:4567::10]:5060 SIP/2.0

To: tel:012345;tag=893219

From: sip:alice@domain1.com;tag=456248

Call-ID: 843817637684230@998sdasdh09

Via: SIP/2.0/UDP

[2001:0DB8:1234::10]:5060;

branch=z9hG4bKnashds8

CSeq: 2 BYE

Contact: sip:[2001:0DB8:1234::10]:5060

Route: <sip:[2001:0DB8:2345::5060]:5060;lr>

Route: <sip:[2001:0DB8:3456::5060]:5060;lr>

Content-Length: 0

•説明

通話を終了するために BYE を送る。Route ヘッダに従って着信 UA に配送 (F13、F14) される。このとき、ENUM による解決は不要であり行われない。

**F13**

BYE

**F14**

BYE

**F15**

SIP/2.0 200 OK

To: tel:012345;tag=893219

From: sip:alice@domain1.com;tag=456248

Call-ID: 843817637684230@998sdasdh09

Via: SIP/2.0/UDP

[2001:0DB8:3456::5060]:5060;

branch=z9hG4bK8924b.1

Via: SIP/2.0/UDP

[2001:0DB8:2345::5060]:5060;

branch=z9hG4bK2d4790.1

Via: SIP/2.0/UDP

[2001:0DB8:1234::10]:5060;

branch=z9hG4bKnashds8

CSeq: 2 BYE

Contact: sip:[2001:0DB8:1234::10]:5060

Content-Length: 0

•説明

着信 UA から 200 OK 応答が返され Via ヘッダに従って発信 UA に向けて配送される (F16–F17)。

**F16**

200 OK

**F17**

200 OK

---

**付録 B デモで行われた SIP 部分のプロトコル表示**

---

5.2 節で紹介したデモにおけるすべてのプロトコル表示を掲載する。

```
>>> to 2001:240:2f:300::100/5060
INVITE sip:fujiwara@sip1.e164.wide.ad.jp SIP/2.0
Via: SIP/2.0/UDP [2001:200:0:ff20:205:3cff:fe06:2498]:5060
From: <sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]>;tag=1033192065
To: <sip:fujiwara@sip1.e164.wide.ad.jp>
Call-ID: 6396cf26@[2001:200:0:ff20:205:3cff:fe06:2498]
CSeq: 18848 INVITE
Contact: <sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]:5060>
Content-Type: application/sdp
Content-Length: 177

v=0
o=- 1063085649 1063085649 IN IP6 2001:200:0:ff20:205:3cff:fe06:2498
s=-
c=IN IP6 2001:200:0:ff20:205:3cff:fe06:2498
t=0 0
m=audio 1190 RTP/AVP 0
a=rtpmap:0 PCMU/8000

<<< from 2001:240:2f:300::100/5060
SIP/2.0 407 Proxy Authorization Required
Call-ID: 6396cf26@[2001:200:0:ff20:205:3cff:fe06:2498]
Content-Length: 0
CSeq: 18848 INVITE
From: <sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]>;tag=1033192065
Proxy-Authenticate: Digest realm="WIDE ENUM WG", nonce="cfb897414240a0e91dd3edbb2a2d7985",
    opaque="", qop="auth,auth-int", stale=FALSE, algorithm=MD5
To: <sip:fujiwara@sip1.e164.wide.ad.jp>;tag=1063085753467
Via: SIP/2.0/UDP [2001:200:0:ff20:205:3cff:fe06:2498]:5060

>>> to 2001:240:2f:300::100/5060
ACK sip:fujiwara@sip1.e164.wide.ad.jp SIP/2.0
Via: SIP/2.0/UDP [2001:200:0:ff20:205:3cff:fe06:2498]:5060
From: <sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]>;tag=1033192065
To: <sip:fujiwara@sip1.e164.wide.ad.jp>
Call-ID: 6396cf26@[2001:200:0:ff20:205:3cff:fe06:2498]
CSeq: 18848 ACK
Content-Length: 0
```

```
>>> to 2001:240:2f:300::100/5060
INVITE sip:fujiwara@sip1.e164.wide.ad.jp SIP/2.0
Via: SIP/2.0/UDP [2001:200:0:ff20:205:3cff:fe06:2498]:5060
From: <sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]>;tag=934803303
To: <sip:fujiwara@sip1.e164.wide.ad.jp>
Call-ID: 475fe914@[2001:200:0:ff20:205:3cff:fe06:2498]
CSeq: 18849 INVITE
Contact: <sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]:5060>
Content-Type: application/sdp
Content-Length: 177
Proxy-Authorization: Digest username="wide", realm="WIDE ENUM WG",
    nonce="cfb897414240a0e91dd3edbb2a2d7985", opaque="", uri="sip:fujiwara@sip1.e164.wide.ad.jp",
    response="67feb09491f0a65fe7493f56109e5423"
```

```
v=0
o=- 1063085649 1063085649 IN IP6 2001:200:0:ff20:205:3cff:fe06:2498
s=-
c=IN IP6 2001:200:0:ff20:205:3cff:fe06:2498
t=0 0
m=audio 1194 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

```
<<< from 2001:240:2f:300::100/5060
SIP/2.0 180 Ringing
Call-ID: 475fe914@[2001:200:0:ff20:205:3cff:fe06:2498]
Content-Length: 0
CSeq: 18849 INVITE
From: <sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]>;tag=934803303
Server: SOFTFRONT-SIP-Proxy ($Revision: 1.100.2.7 $)
To: <sip:fujiwara@sip1.e164.wide.ad.jp>;tag=164723426
Via: SIP/2.0/UDP [2001:200:0:ff20:205:3cff:fe06:2498]:5060
```

```
<<< from 2001:240:2f:300::100/5060
SIP/2.0 200 OK
Call-ID: 475fe914@[2001:200:0:ff20:205:3cff:fe06:2498]
Contact: <sip:fujiwara@[2001:200:0:ff20:2a0:deff:fe11:7373]:5060>
Content-Length: 189
Content-Type: application/sdp
CSeq: 18849 INVITE
From: <sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]>;tag=934803303
Record-Route: <sip:fujiwara@sip1.e164.wide.ad.jp:5060;maddr=[2001:240:2f:300:0:0:100]>
Server: SOFTFRONT-SIP-Proxy ($Revision: 1.100.2.7 $)
To: <sip:fujiwara@sip1.e164.wide.ad.jp>;tag=164723426
Via: SIP/2.0/UDP [2001:200:0:ff20:205:3cff:fe06:2498]:5060
```

## 第 16 部 ENUM テストベッドの運用

W I D E R E P O R T  
2 0 0 3  
a n n u a l r e p o r t

```
v=0
o=- 3272074472 3272074472 IN IP6 2001:200:0:ff20:2a0:deff:fe11:7373
s=-
c=IN IP6 2001:200:0:ff20:2a0:deff:fe11:7373
t=0 0
m=audio 5004 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=ptime:20

>>> to 2001:200:0:ff20:2a0:deff:fe11:7373/5060
ACK sip:fujiwara@sip1.e164.wide.ad.jp SIP/2.0
Via: SIP/2.0/UDP [2001:200:0:ff20:205:3cff:fe06:2498]:5060
From: <sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]>;tag=934803303
To: <sip:fujiwara@sip1.e164.wide.ad.jp>
Call-ID: 475fe914@[2001:200:0:ff20:205:3cff:fe06:2498]
CSeq: 18849 ACK
Content-Length: 0

RTP: 2001:200:0:ff20:205:3cff:fe06:2498/1194 <=> 2001:200:0:ff20:2a0:deff:fe11:7373/5004

<<< from 2001:240:2f:300::100/5060
BYE sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]:5060 SIP/2.0
Call-ID: 475fe914@[2001:200:0:ff20:205:3cff:fe06:2498]
Content-Length: 0
CSeq: 30 BYE
From: <sip:fujiwara@sip1.e164.wide.ad.jp>
Max-Forwards: 69
Record-Route: <sip:fujiwara.feb48@sip1.e164.wide.ad.jp:5060;maddr=[2001:240:2f:300:0:0:0:100]>
Server: SOFTFRONT-SIP-Proxy ($Revision: 1.100.2.7 $)
To: <sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]>;tag=934803303
User-Agent: YAMAHA RT56v
Via: SIP/2.0/UDP [2001:240:2f:300:0:0:0:100];branch=z9hG4bK16ff67cf.1
Via: SIP/2.0/UDP [2001:200:0:ff20:2a0:deff:fe11:7373]:5060;branch=z9hG4bK4215059335

>>> to 2001:240:2f:300::100/5060
SIP/2.0 200 OK
Via: SIP/2.0/UDP [2001:240:2f:300:0:0:0:100];branch=z9hG4bK16ff67cf.1
Via: SIP/2.0/UDP [2001:200:0:ff20:2a0:deff:fe11:7373]:5060;branch=z9hG4bK4215059335
To: <sip:kick@[2001:200:0:ff20:205:3cff:fe06:2498]>;tag=934803303
From: <sip:fujiwara@sip1.e164.wide.ad.jp>
Call-ID: 475fe914@[2001:200:0:ff20:205:3cff:fe06:2498]
CSeq: 30 BYE
Content-Length: 0
```