The Generic Gateway

– an ENUM driven PSTN-IP gateway

Reference Implementation

Specification and System Description

enum.at
net.communications

Version history:

Version	*Date/author*	*Changes*
0.2	15.09.2004/Klaus Darilion	initial version
0.3		adding details
0.4	12.10.2004/Klaus Darilion	detailed call flows, configuration of all services, local caching-only DNS server added
0.5	12.10.2004/Klaus Darilion	change the location of the missed Radiator accounting to /var/radiator/....
0.7		moved installation instructions into a separate document
0.8	07.12.2004/Klaus Darilion	remove sensitive information

Table of Contents

1 The Generic Gateway

The generic gateway (GG) supports various services (voice, fax ...), various protocols (SIP, H.323 ...) and arbitrary number-blocks (0780, 0720, +87810, 05 ...). The advantage of ENUM, the addressing of various services with a single ENUM domain introduces the problem of “how to find the proper service” in case of PSTN to IP calls. For plain IP-IP calls, typically the calling application knows the service (a softphone is aware if the user wants to send a fax or wants to call somebody). Therefore, the calling application can use ENUM to identify the address of the particular service.

In PSTN-IP calls, the ENUM lookup is done by the generic gateway or another intermediate node (e. g. a proxy or gatekeeper). In the call setup signaling on the PSTN side, there is no information available if the call is a standard phone call or a data call (modem, fax). If the ENUM lookup reveals a single NAPTR, this NAPTR will be chosen. If there are several NAPTR records, there exist several ways of choosing the call destination. Following are several possible routing scenarios.

- Single NAPTR record:
  Route the call to the URI defined in the NAPTR

- Multiple NAPTR records:
  From RFC 3761: *“A client implementing ENUM MUST adhere to the Order field but can simply take the Preference value "on advisement" as part of a client context specific selection method.”*

  Therefore, the generic gateway has to choose the NAPTR with the lowest order field (if supported). If there are several records with the same order, the generic gateway may choose the NAPTR with the lowest preference field or any other of them (if the generic gateway has other preferences (e. g. local policies).

Note: For some background information about ENUM please refer to RFC 3245.
http://ietf.org/rfc/rfc3245

2 The Generic Gateway Concept

Figure 1 shows the concept of the generic gateway. The generic gateway accepts incoming calls from the PSTN, and performs an ENUM lookup for the called number. If the ENUM lookup is successful, i. e. there are NAPTR records with supported service types, the generic gateway retrieves the URI and delivers the call to the IP destination.

Note: The term “generic gateway” refers to the whole gateway system whereas the term “gateway” refers to the Cisco gateway inside the generic gateway (refer to figure 4).

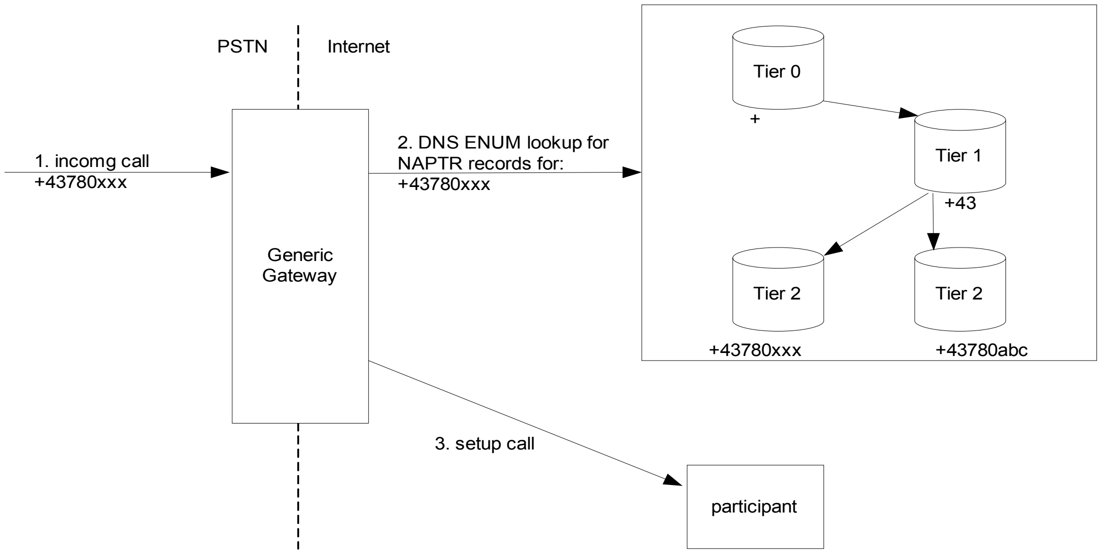


Figure 1: The generic gateway concept

Physically, the generic gateway offers one interface to the PSTN and one interface to the Internet as shown in figure 2.

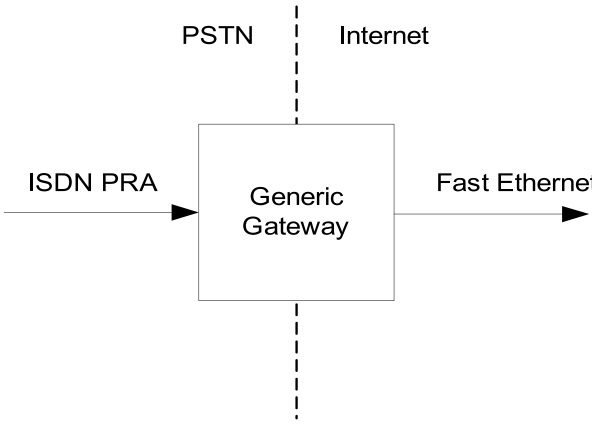


Figure 2: Physical interfaces

Figure 3 shows the logical interfaces of the generic gateway. To the PSTN there is still only ISDN whereas to the Internet there are several interfaces/protocols. The DNS and ENUM protocols are used to retrieve the service and the destination URI. Following, SIP, H.323 or SMTP is used to deliver the call.

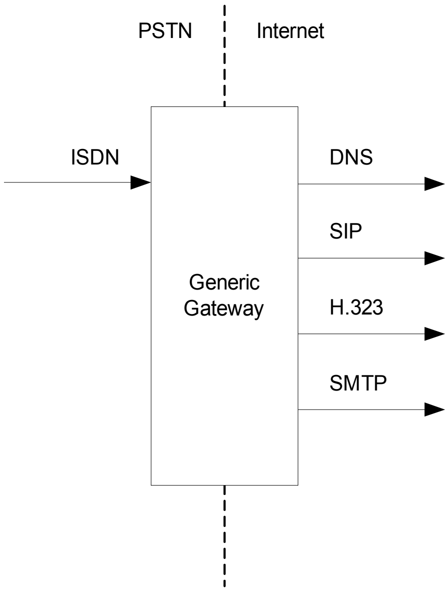


Figure 3: Logical interfaces

2.1 Features

All protocols used by the generic gateway are standardized as shown in table 1. The generic gateway supports voice (SIP and H.323) and fax (ifax – fax to email) sessions. The corresponding supported ENUM service types are shown in table 2. The supported VoIP codecs are shown in table 3.

The gateway supports ENUM routed basic calls (no supplementary services like call transfer, pickup ...) with SIP and H.323. Furthermore, reception of group 3 fax with email delivery is supported.

SIP domain resolving in the Cisco GW supports SRV records, but not NAPTR records (refer to RFC 3263).

The gateway supports relaying of DTMF signals by using named telephony events (NTEs) for both, SIP and H.323. This method is described in RFC 2833.

For the ENUM lookup the generic gateway queries the public DNS ENUM infrastructure (domain . e164.arpa).

Note: The generic gateway itself is independent from an ENUM registry system.

protocol	***standardized by***
ISDN	ITU
Group 3 Fax	ITU
H.323	ITU
DNS, NAPTR, ENUM	IETF: DNS RFC 1035 NAPTR RFC 3401-3404 and RFC 3761 (Cisco ENUM lookup supports RFC 2915) ENUM story in RFC 3245
SIP	IETF RFC 3261
SMTP	IETF RFC 2821

Table 1: Protocols/interfaces used by the generic gateway

service	***supported service types***
voice	E2U+sip sip+E2U E2U+voice:sip E2U+h323 h323+E2U E2U+voice:h323
fax	E2U+ifax:mailto

Table 2: Supported ENUM service types

codec	***description***
g711alaw	G.711 A Law 64000 bps
g711ulaw	G.711 u Law 64000 bps
g723ar53	G.723.1 ANNEX-A 5300 bps
g723ar63	G.723.1 ANNEX-A 6300 bps
g723r53	G.723.1 5300 bps
g723r63	G.723.1 6300 bps
g726r16	G.726 16000 bps
g726r24	G.726 24000 bps
g726r32	G.726 32000 bps
g728	G.728 16000 bps
g729br8	G.729 ANNEX-B 8000 bps
g729r8	G.729 8000 bps
gsmefr	GSMEFR 12200 bps
gsmfr	GSMFR 13200 bps

Table 3: Supported VoIP codecs

3 Detailed Architecture

The generic gateway implementation is shown in figure 4: It consists of a Cisco 5300 gateway and an additional Linux based “helper” PC which hosts several service which necessary to deal with the limitations of the Cisco gateway:

- Cisco's ENUM lookup is based on the obsolete RFC 2915, which uses different service types. Therefore, the ENUM service fields have to be converted for the Cisco gateway on the fly for each ENUM lookup by a DNS proxy.
- Cisco's fax “onramp” (fax to email) application does not support ENUM/ifax. Therefore, the ENUM lookup for ifax has to be done outside of the Cisco gateway.
- Cisco's email format is not supported by all email clients. Therefore, the email has to be “re-formated”.
- The ENUM lookup and service decision (voice or fax) has to be done before choosing the outgoing dial-peer. This is not possible with Cisco's IOS. Therefore, the ENUM lookup is done in the helper PC.

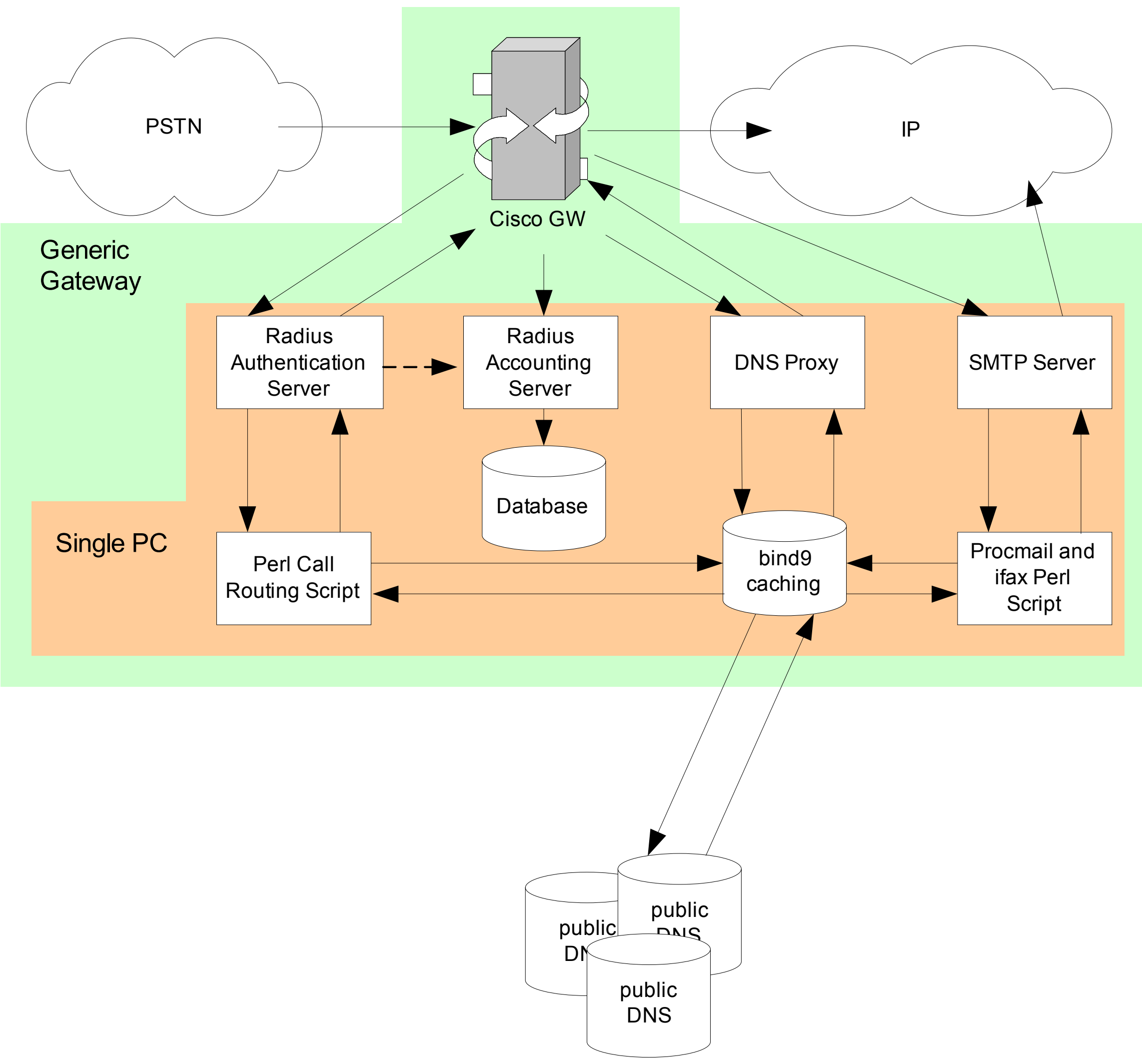


Figure 4: Generic gateway architecture

The GG consists of the following components:

- Cisco AS5300 Gateway: PSTN – IP gateway
- Radius Authentication Server: makes no authentication but performs an ENUM lookup and signals the destination service (sip, h323 or fax) back to the GW. It can be possible to add a blacklist for calling and called number.
- DNS proxy: As the Cisco GW does not support the NAPTR service fields (E2U+...), the DNS requests will be sent to a proxy which rewrites the NAPTR records into the Cisco format (...+E2U).
- Radius Accounting Server: All calls will be accounted for call statistics analysis and troubleshooting. The radius server writes the accounting records into a database system for easier debugging and start/stop records matching.
- SMTP Server: The SMTP server receives the emails with fax attachment and delivers it to the local “ifax” user.
- ifax sendmail application: The ifax application receives the fax email from the GW, reformat the email, performs an ENUM ifax lookup and sends the email to the resolved mailto: URL.
- Database System: The database system is used to store the accounting data.

3.1 Call Flow

Following is a short description how incoming calls are handled by the GG.

3.1.1 SIP

Figure 5 shows the messages flow during a SIP call excluding the RTP streams. Figure 6 shows the corresponding packet trace. During a SIP call, the following steps are performed by the generic gateway:

Note: This packet traces does not include the DNS messages between the DNS proxy and the local, caching-only DNS server (refer to figure 4).

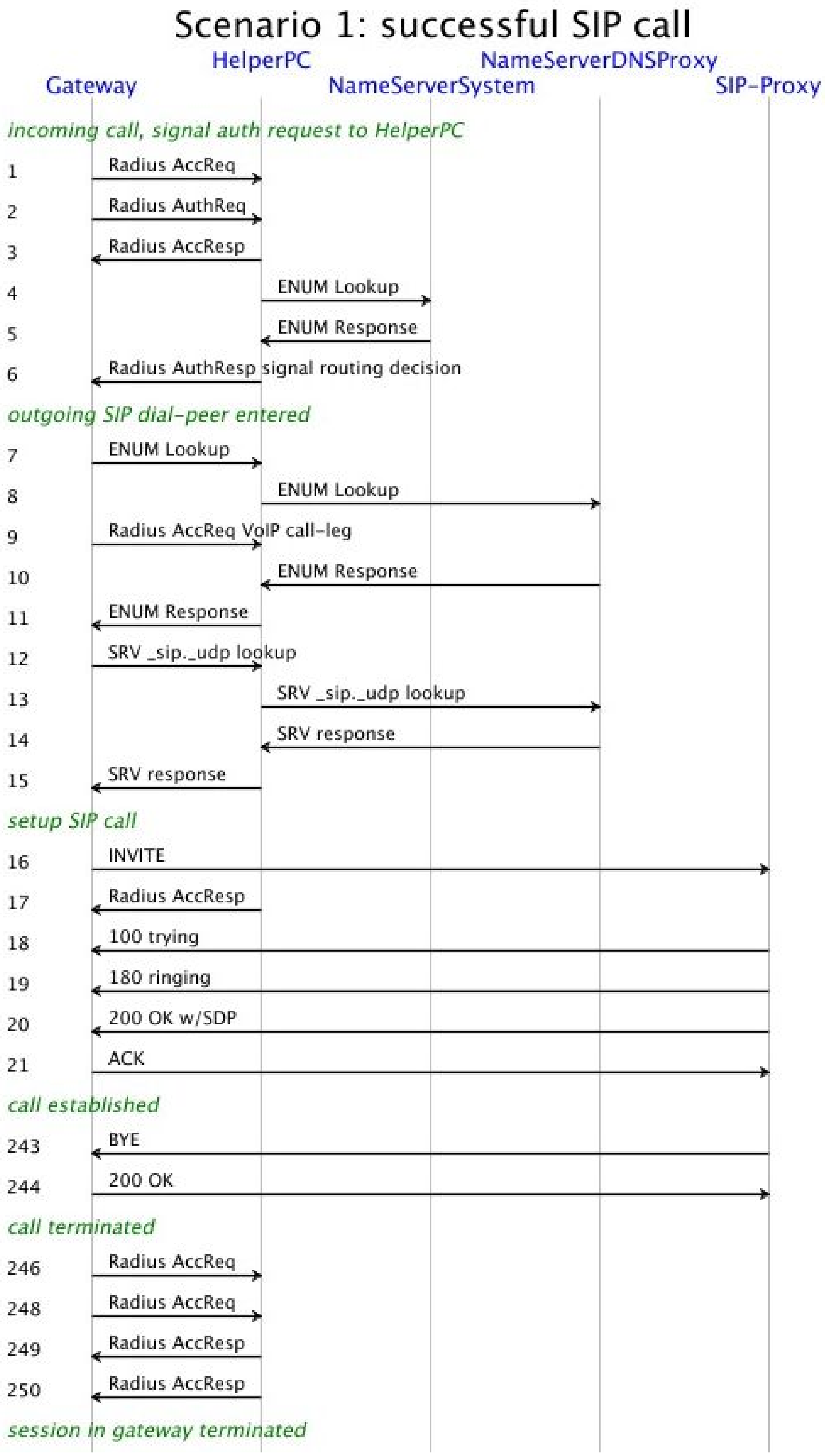
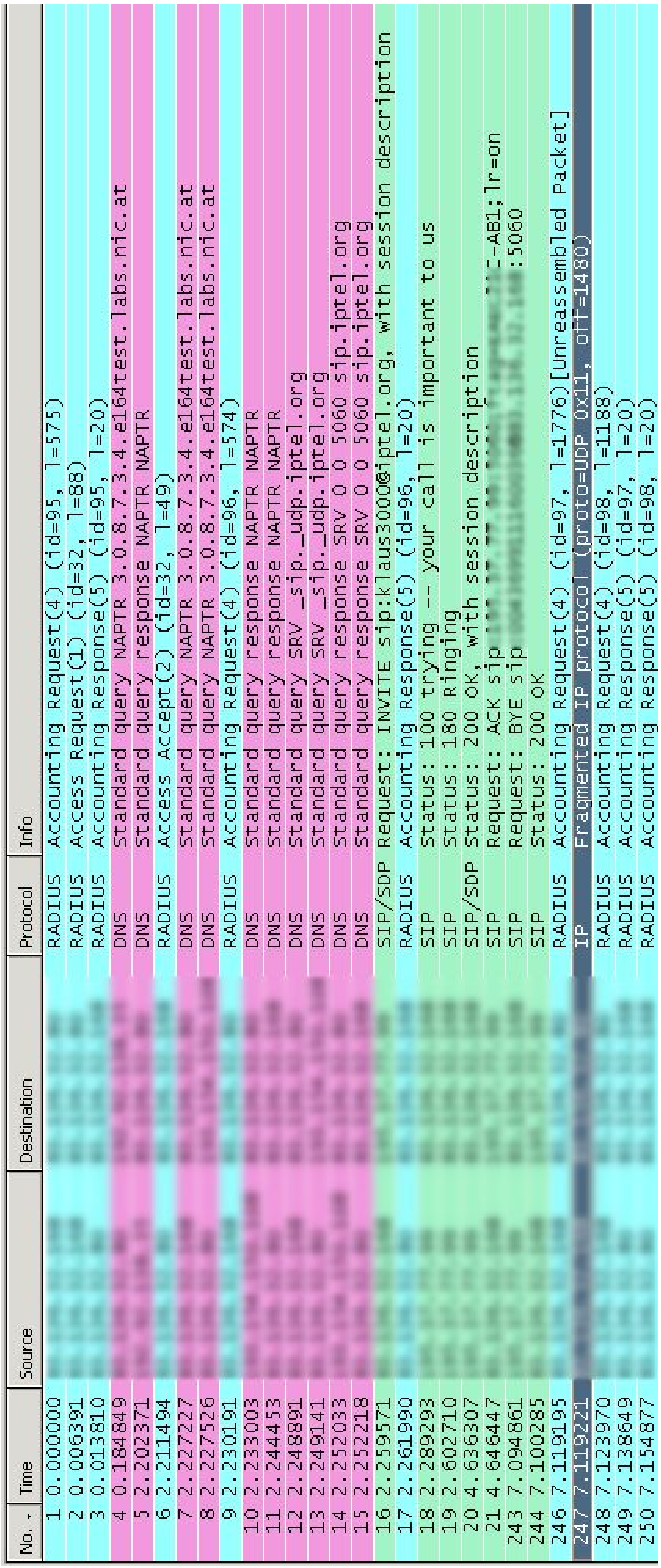


Figure 5: Packet flow for a SIP call



No.	Time	Source	Destination	Protocol	Info
1	0.000000	[illegible]	[illegible]	RADIUS	Accounting Request(4) (id=95, l=575)
2	0.006391	[illegible]	[illegible]	RADIUS	Access Request(1) (id=32, l=88)
3	0.013810	[illegible]	[illegible]	RADIUS	Accounting Response(5) (id=95, l=20)
4	0.184849	[illegible]	[illegible]	DNS	Standard query NAPTR 3.0.8.7.3.4.e164test.labs.nic.at
5	2.202371	[illegible]	[illegible]	DNS	Standard query response NAPTR NAPTR
6	2.211494	[illegible]	[illegible]	RADIUS	Access Accept(2) (id=32, l=49)
7	2.227227	[illegible]	[illegible]	DNS	Standard query NAPTR 3.0.8.7.3.4.e164test.labs.nic.at
8	2.227526	[illegible]	[illegible]	DNS	Standard query NAPTR 3.0.8.7.3.4.e164test.labs.nic.at
9	2.230191	[illegible]	[illegible]	RADIUS	Accounting Request(4) (id=96, l=574)
10	2.233003	[illegible]	[illegible]	DNS	Standard query response NAPTR NAPTR
11	2.244453	[illegible]	[illegible]	DNS	Standard query response NAPTR NAPTR
12	2.248891	[illegible]	[illegible]	DNS	Standard query SRV _sip._udp.iptel.org
13	2.249141	[illegible]	[illegible]	DNS	Standard query SRV _sip._udp.iptel.org
14	2.252033	[illegible]	[illegible]	DNS	Standard query response SRV 0 0 5060 sip.iptel.org
15	2.252218	[illegible]	[illegible]	DNS	Standard query response SRV 0 0 5060 sip.iptel.org
16	2.259571	[illegible]	[illegible]	SIP/SDP	Request: INVITE sip:klaus3000@iptel.org, with session description
17	2.261990	[illegible]	[illegible]	RADIUS	Accounting Response(5) (id=96, l=20)
18	2.289293	[illegible]	[illegible]	SIP	Status: 100 trying -- your call is important to us
19	2.602710	[illegible]	[illegible]	SIP	Status: 180 Ringing
20	4.636307	[illegible]	[illegible]	SIP/SDP	Status: 200 OK, with session description
21	4.646447	[illegible]	[illegible]	SIP	Request: ACK sip [illegible]-AB1;lr=on
243	7.094861	[illegible]	[illegible]	SIP	Request: BYE sip [illegible]:5060
244	7.100285	[illegible]	[illegible]	SIP	Status: 200 OK
246	7.119195	[illegible]	[illegible]	RADIUS	Accounting Request(4) (id=97, l=1776)[Unreassembled Packet]
247	7.119221	[illegible]	[illegible]	IP	Fragmented IP protocol (proto=UDP 0x11, off=1480)
248	7.123970	[illegible]	[illegible]	RADIUS	Accounting Request(4) (id=98, l=1188)
249	7.138649	[illegible]	[illegible]	RADIUS	Accounting Response(5) (id=97, l=20)
250	7.154877	[illegible]	[illegible]	RADIUS	Accounting Response(5) (id=98, l=20)

Figure 6: IP packet trace of the SIP call

1. A call arrives at the ISDN interface at the Cisco GW (the whole E.164 number will be signaled to the GW as extension of its ISDN connection)
2. The gateway activates an IVR application in the incoming dial peer. This application sends a radius authentication request to the radius server which includes the calling and called number (packet 2).
3. The radius server forwards the authentication request to a Perl script.
4. The Perl script performs an ENUM lookup (packet 4 and 5) for the called number. The result of this lookup (sip, h323 or fax) will be signaled back to the gateway in the radius response (packet 6).
5. The IVR application evaluates the radius response:
   - sip: prefix the called number with '1'
   - h.323: prefix the called number with '2'
   - fax: activate fax
   - others: signal corresponding ISDN cause code back to the PSTN

   Then, the application finishes and the gateway chooses the corresponding outgoing dial peer.
6. The outgoing dial-peer performs an ENUM lookup for the called destination. Before this lookup, the first digit of the called number will be stripped.
7. The gateway sends the DNS request (packet 7) to the configured DNS server. This is the DNS proxy.
8. The DNS proxy forwards the DNS request to the configured name server (packet 8). The DNS responses (packet 10) will be parsed for NAPTR records and the service fields will be rewritten to “Cisco style”. Then the response will be forwarded to the gateway (packet 11).
9. The ENUM lookup picks one of the supported NAPTRs from the DNS response and establishes the VoIP call:
   - resolve the IP address of the next SIP hop (SIP proxy or SIP UA) (packets 12-15)
   - SIP signaling (packets 16, 18-21, 243 and 244)
10. Every time a call-leg is initialized or destroyed, an accounting request will be sent to the radius server (packets 1, 9, 246 and 248), which will be acknowledged by the Radius server (packets 2, 17, 249 and 250).
11. The radius server uses a Radiator hook to rewrite the accounting request to fetch the Cisco proprietary accounting parameters. Then, the accounting attributes will be written into the postgresql database.

3.1.2 H.323

The Radius and DNS signaling is identical to the SIP call, except the SIP signaling is replaced by the H.323 signaling.

3.1.3 Fax

Step 1-5 are identical to the SIP call scenario. The following steps are:

6. The mmoip dial peer will be activated. This dial peer activates the fax_vfc_onramp

application at the DSPs which connects to the incoming call.

7. The fax application waits for fax signaling. If an incoming fax will be detected, the fax will be received.
8. The fax will be sent as email with a tiff attachment to the configured SMTP server.
9. The SMTP server is configured to deliver the fax-email to the local user *ifax*.
10. A `.procmailrc` file in the home directory of the ifax user forwards the email to the sendfax Perl script **and** stores the received fax-email in the spool directory of the ifax user.
11. The script retrieves the attachment from the original fax-email and creates a new email with pre-defined subject, mail body and the retrieved attachment. The new email will be sent to the ifax:mailto address retrieved by an ENUM lookup.
12. Every time a call-leg is initialized or destroyed, an accounting request will be sent to the radius server, which will be acknowledged by the Radius server.
13. The radius server uses a Radiator hook to rewrite the accounting request to fetch the Cisco proprietary accounting parameters. Then, the accounting attributes will be written into the postgresql database.

3.2 Cisco Gateway

The gateway (AS5300 platform) supports ISDN, SIP, H.323 and fax signaling. The gateway supports ENUM lookups according to the obsolete RFC 2916, which causes interoperability problems:

Cisco supported service types[1]:

```
sip+E2U
h323+E2U
```

ENUM records according to RFC 3761:

```
E2U+type:subtype+type:subtype+...
```

e.g. a SIP service type for voice telephony as recommended by ETSI TS 102 172 V1.1.1 (2003-03)

```
E2U+voice:sip
```

Therefore, "new" NAPTR records won't be accepted by the GW although it would be able perform calls to the particular URI. To resolve this problem, the DNS query is done via a DNS proxy which rewrites the service field in the answers to the "old" style. The ENUM resolver in the gateway is case insensitive for the "service type" field ("SIP+E2U" will also be accepted).

If the Cisco GW is configured with 2 identical outgoing dial-peers – one configured for SIP and one for H.323 – the GW performs ENUM lookup on both stacks and chooses the one with an ENUM entry found. If both stacks reveal a valid destination, the preference can be set in the dial-peer options.

If the GW should also support onramp fax (receiving fax calls from the PSTN and deliver it by email), the decision if an incoming call is a voice or fax call has to made before matching an outgoing dial-peer. Therefore, an IVR application in the incoming dial-peer has to make this decision. The decision is based on an ENUM lookup – therefore, the built in ENUM lookup can't be

1 Cisco ignores everything behind the E2U pattern. Therefore service fields like `sip+E2U:voice` and `h323+E2Uabcdef` will also be accepted by the GW.

used for this decision as it can be activated only in the outgoing dial-peer. Therefore, the ENUM lookup is done by an external Perl script. A simple way to give control to an external application is via a Radius authentication request. The Radius response includes a Cisco proprietary Radius attribute which indicates the protocol the GW should use, i. e. *sip*, *h323* or *fax*. This attribute is queried by the IVR script and used to select the proper outgoing dial-peer.

Format of the radius authentication request (sent from the gateway to the radius server):

```
User-Name = "436991234567"
User-Password = "XXXXXXXXXXX"
Calling-Station-Id = "436991234567"
Called-Station-Id = "437801"
Service-Type = Login-User
NAS-IP-Address = 10.0.0.168
```

Format of the radius authentication response (sent from the gateway to the radius server):

```
Service-Type = Login-User
cisco-avpair = "h323-ivr-in=sip"
```

Table 4 shows the possible return values in the authentication response and their meaning.

aaa_avpair h323-ivr-in value	***action***
sip	prefix “1” to destination number to activate SIP dial-peer
h323	prefix “2” to destination number to activate H.323 dial-peer
fax	*set callInfo(fax) true* to activate the mmoip dial-peer
none	no supported service was found, signal ISDN cause code 3 - no route to destination
dnserror	there was an error while trying to resolve the domain (e. g. a DNS timeout), signal ISDN cause code 41 - temporary failure
nodomain	this domain is not delegated, signal ISDN cause code 1 - unallocated (unassigned) number
destblocked	not implemented yet (the called number is on the blacklist and the call is not allowed, signal ISDN cause code 41 (XXX is there a better response code?)
srcblocked	the calling number is on the blacklist and the call is not allowed, signal ISDN cause code 41 (XXX is there a better response code?)

Table 4Radius authentication response return values

Once the GW has chosen an outgoing dial-peer, it performs an ENUM lookup to retrieve the URI for the particular protocol. Thus, there are two ENUM lookups for each incoming call:

- in the call routing script (activated by the Radius authentication request) to decide between sip, h233 and fax
- in the GW in the outgoing dial-peer to retrieve the destination URL in case of sip or h323
- in the ifax application to retrieve the mailto: URL in case of fax

3.3 Radius Server

The GG uses Radius for the authentication (which performs an ENUM lookup) and for storing the accounting data into a database. The authentication task performs an ENUM lookup which involves DNS queries. Therefore, the authentication requests may last several seconds. Therefore, the radius server will fork for each authentication requests. As this conflicts with the authentication task, which wants to write into the database reusing existing TCP connections (fork will close the file handles and network connections), the GG uses two instances of the radius server – one for the accounting task and one for the authentication task.

3.3.1 Software

Radiator (http://www.open.com.au/) is used as radius server for authentication and accounting (call logging). The Radiator software can be downloaded as architecture independent rpm package. As the Helper PC uses Debian as operating system, the rpm package has to be converted in a Debian package (.deb). This is done using the "alien" tool (if alien is not installed, install it with "apt-get install alien").

3.3.2 Radius Authentication Server and Perl Call Routing Script

The Radius server receives the authentication requests from the GW and pass the details (called number) to the call routing script. The call routing script performs an ENUM lookup and decides which protocol should be used by the GW. Currently there is no authentication, but this can be added based on blacklists for caller and callee number and appropriate announcement messages at the GW, e. g. "You are not allowed to use this service!" or "This number is blocked!".

Note: Playing back announcement messages into the PSTN with connecting the call is not supported by the UNI (user network interface), therefore it is not implemented.

3.3.2.1 Configuration

The configuration file (`/etc/radiator/radius-auth.cfg`), the authentication script (`/etc/radiator/auth/auth-enum.pl`) and the startup script (`/etc/init.d/radiator-auth`) have to be installed. The files can be found in the directory `radiator+scripts`.

3.3.2.2 Maintenance

The radius server logs requests and responses into the logfile `/var/log/radiator-auth/logfile`. The detail level of the logging can be configured in `/etc/radiator/radius-auth.cfg`.

The radius authentication server can be started/stopped by:
`/etc/init.d/radiator-auth start|stop|restart`

3.3.3 Radius Accounting Server

The GW produces *start*, *update* (every 1 minute) and *stop* records for each call-leg. Each call-leg is identified by the *acctsessionid* which is unique until a reboot of the GW. A call is identified by the *h323-incoming-conf-id*. Table 5 shows typical call scenarios and the corresponding number of call-legs.

call scenario	*number of call-legs*
standard SIP/H.323 call	2
fax call	1
call to invalid number	1
call with connection problems and retries and the outgoing call	>2

Table 5: Call scenarios and the corresponding number of call legs

A call typically consists of two call-legs.

The radius server receives the accounting request and starts a pre-authentication hook which rewrites the radius request from “Cisco style” into a style suitable for further processing in the radius server. Then, the radius server retrieves the attributes from the request, creates an SQL queries and inserts the data into the database using a Perl DBI database connection. Unsuccessful inserts into the database will be stored in SQL format in the file `/var/radiator/missedaccounting`.

3.3.3.1 Configuration

The configuration file (`/etc/radiator/radius.cfg`), the hooks (`/etc/radiator/hooks`) and the startup script (`/etc/init.d/radiator`) have to be installed. The files can be found in the directory `radiator+scripts`.
The accounting data are written into a postgresql database. Therefore, the database server, the database, the username and the password have to be configured in `/etc/radiator/radius.cfg`. The format is:

```
DBSource        dbi:Pg:dbname=NameOfTheDatabase;host=serverName
DBUsername      userName
DBAuth          userPassword
```

3.4 local DNS server

The call setup delays heavily depends on the performance of the DNS lookups (ENUM lookups, SRV records lookup, A record lookups). Therefore, if the GG uses a public DNS server, performance problems on this server will lead to problems within the GG. Thus, the GG includes a local DNS server, which performs recursive, caching-only lookups. This will speed up DNS lookups and call setups.

3.5 Database System

The database stores all the accounting information. For debugging it is important to match start, stop and keep-alive records. This must be done by the application which analyzes the accounting (CDR = call detail records) data. The GG uses the standard postgresql DB within Debian Woody. Installation is done by:

To **verify the accounting**, you have to login to the database and query the accounting table:

```
helper1:/etc/radiator# psql ggacc gguser
Password:
...
ggacc=> select * from accounting;
```

```
ggacc=> select * from fax;
.............
```

As all the components of the GG can be run on a dedicated PC, the communication should be configured to use the public IP address of the database server, not the loopback address.

3.6 Fax System

The fax system consists of several components whose configuration options must be synchronized. The fax delivery will be performed as task of the local *ifax* user.

3.6.1 Cisco Gateway

The following line configures the SMTP server to which the gateway should send the emails, the email subject, the filename of the TIFF attachment, and the sender email address (username@hostname).

```
fax interface-type fax-mail
mta send server 10.0.0.232 port 25
mta send server 10.0.0.233 port 25
mta send subject Fax message received
mta send filename incoming_fax date
mta send mail-from hostname ggpc1
mta send mail-from username $s$
```

The MMOIP dial-peer configuration includes the To: email address of the email. The domain configured here (in this case 'helper') must also be configured in the exim configuration on the helper PC.

```
dial-peer voice 14 mmoip
   preference 3
   application fax_on_vfc_onramp_app out-bound
   destination-pattern 43780T
   information-type fax
   session target mailto:$d$@faxrelay
```

3.6.2 Exim SMTP Server

The mail server must be configured to accept the emails from the gateway and deliver them to the local user *ifax*.

3.6.3 Procmail

Procmail will be used for the local delivery and forwarding to the perl script, which re-formats the emails and performs the ENUM lookup.

3.6.4 sendfax Perl Script

This perl script performs the following tasks:

- retrieve the complete original ('old') email sent by the Cisco GW from procmail via stdin

- retrieve from the old email:
  - the called phone number
  - the calling phone number
  - the 'Reply-To' phone number configured in the originating fax device
  - the tiff attachment
- create a new email with
  - calling phone number in the subject
  - calling, called und reply-to number in the body of the email
  - a description about the generic gateway service in the body of the email
- ifax:mailto ENUM lookup for the called phone number
- send the new email to the resolved email address
- database logging into the table 'fax' of the 'ggacc' database
  - when entering the script, a CDR with status 'received' will be inserted in the DB
  - after successful delivery to the SMTP server, this CDR is updated to the status 'sent'. Note: this means, that the local SMTP server has accepted the email for delivery, but does not mean that the email received the destination mail box.

3.6.4.1 Installation and Configuration

Installation: Copy the sendfax script in the home directory of the ifax user. The exact destination of the script must be configured in the .procmailrc file. Furthermore, the access rights must be configured to allow the script to be executed (# chmod a+x sendmail.pl). The database user and the IP address of the DB (fax logging) must be configured inside the sendfax.pl script.

3.7 TFTP Server

The TFPT server is necessary to upload/download configurations, applications or IOS versions to/from the Cisco gateway.

3.7.1 Installation and Configuration

Installation via apt-get:

```
#apt-get install tftpd
```

A root directory for the tftp server must be created and configured:

```
# mkdir /var/spool/tftpboot
```

The tftp server will be started by the inet daemon. Thus, the configuration must be done in:

```
/etc/inetd.conf:
tftp  dgram  udp  wait  nobody  /usr/sbin/tcpd  /usr/sbin/in.tftpd  /var/spool/tftpboot
```

4 Hardware and Software Requirements

Physically, the generic gateway consists of two devices with the following requirements:

4.1 Cisco AS5300 Gateway

- at least one PRA interface and one Fast Ethernet interface
- at least one voice feature card (vfc) module installed (DSPs for VoIP).
- Cisco IOS version: IOS (tm) 5300 Software (C5300-JS-M), Version 12.2(13)T3, RELEASE SOFTWARE (fc2)
  Note: The gateway is tested and verified with this version. Newer versions should work too. For a detailed software version description refer to the appendix.
- Fax onramp application "fax_on_vfc_onramp_ap" (Note: Installation can be tested with "show call application voice summary")
- Firewall configuration:
  - The gateway must be allowed to receive/send UDP and TCP packets from/to any IP address and port on local ports for SIP and H.323.
  - The gateway must be allowed to receive/send UDP packets from/to any IP address and port on local ports 16384-32767 for RTP/RTCP.
  - The gateway must be allowed to receive/send DNS packets (port 53).
  - The gateway must be allowed to contact the helper PC for the following protocols:
    - Radius incoming and outgoing
    - DNS incoming and outgoing
    - SMTP outgoing
    - Telnet incoming (configuration)
    - tftp outgoing (maintenance of IOS and call applications)
    - (SIP/H.323/RTP in case of VoIP test applications on the helper PC incoming and outgoing)

4.2 Helper PC

- Standard x86 compatible server hardware (hardware must be supported by Debian woody 3.0r2 Linux distribution)
- Operating system: Debian woody 3.0r2 Linux
- Software:
  - Radiator (or any other radius server which supports execution of perl scripts and database accounting)
  - Perl (included in Debian distribution)
  - Perl enum.pm module from http://jprs.co.jp/enum/software/ENUM.pm
  - postgresql database (for storing accounting data, included in Debian)
  - exim SMTP server (for delivery of fax emails, included in Debian)
  - procmail (for email handling, included in Debian)
  - dnsproxy to rewrite "old-style" NAPTRs.
  - tftpd server (included in Debian)
- Firewall configuration – the following services must be opened:
  - DNS
  - HTTP (outgoing for OS updates, incoming for call analysis from dedicated IP addresses)
  - SMTP outgoing (sending fax-emails)
  - SSH incoming and outgoing (remote maintenance and debugging)
  - The helper PC must be allowed to communicate with the gateway with the

following protocols:

- Radius
- DNS
- SMTP incoming (receiving fax-emails from gateway
- Telnet (configuration)
- tftp incoming (maintenance of gateway)
- (SIP/H.323/RTP in case of local "test numbers" incoming and outgoing)

5 Limitations

- Cisco DNS only supports UDP; UDP only can store up to 7 NAPTRs in one packet, ore more in case of fragmented packets. **Note:** Cisco DNS requests does not indicate support for fragmented packets (EDNS0), but it accepts fragmented packets :-)
- SIP uri without transport parameter should be resolved for _sip._udp and _sip._tcp. Cisco only tries _udp. TCP is only used when a transport=tcp parameter is present.
- H323: VAD and FastStart must be disabled to work with Netmeeting: supports 323v3 complete and some v4 things. **Note:** Some H.323 terminals does not show the calling line number.
-

6 Global Configuration changes

6.1 Nameserver

If the you want to change the nameservers used by the GG, you have to change this in several locations. The perl scripts performing ENUM lookups uses the local configured nameservers. That is, `/etc/radiator/auth/auth-enum.pl` and `/home/ifax/sendfax.pl` use the nameservers configured in `/etc/resolv.conf`. All this components should use the local DNS server as name server to avoid delays caused by slow public name servers.

The Cisco gateway must be configured to use the DNS proxy as nameserver. The DNS proxy uses the nameserver configured in dnsproxy.conf.

Note: The nameserver for the DNS proxy must be recursive nameservers!

6.2 ENUM tree

The used ENUM tree must be configured in every part of the GG which performs ENUM lookups. These are:

- `/etc/radiator/auth/auth-enum.pl` in the configuration part of the script
- `/home/ifax/sendfax.pl` in the configuration part of the script
- the Cisco configuration:

```
!
voice enum-match-table 2
!
```

-

6.3 IP Addresses

IP addresses changes of the gateway or the helper PCs have dramatic effects on the whole system. Therefore, IP addresses shouldn't be changed without good reasons.

If IP addresses are changed, the firewall rules of the LAN firewall must be adopted and also the local IP-tables Linux firewall rules must be adopted (if availably).

As every service of the GG uses a dedicated IP address, all services must be adopted to “bind” to the new IP addresses. These service are:

- Radius Authentication Server
- Radius Accounting Server
- Postgresql Server
- DNS Proxy
- Bind9 DNS Server
- exim SMTP Server
- standard services: ssh, tftp, ...

7 High Availability Concept

ENUM lookups, Radius authentication and mail delivery can be done on any helper PC. The Cisco gateway will detect failures of the external services ans will switch over to another server if multiple servers are configured. The fail over is separate for each server server, i. e. if the Radius service on one server fails, the Radius client in the Cisco GW will switch over to another server, but the other services still uses the original server.

If the Radius accounting switches over to another server, the CDRs will be split into 2 databases, which have to be merged before analyzing the CDRs. The Cisco GW allows multiple configuration of DNS servers, SMTP MTA servers and Radius servers with automatic fail-over.

Figure 7 shows a high availability (HA) concept for the generic gateway. In this scenario, there are 2 “Helper Server” for a single PSTN-IP gateway.

It is possible to use several Cisco GWs and several Helper PCs – any arbitrary setup is supported.

Note: As the DNS proxy only supports forwarding to single DNS server, the DNS proxies must be configured with different nameservers to avoid a single point of failure.

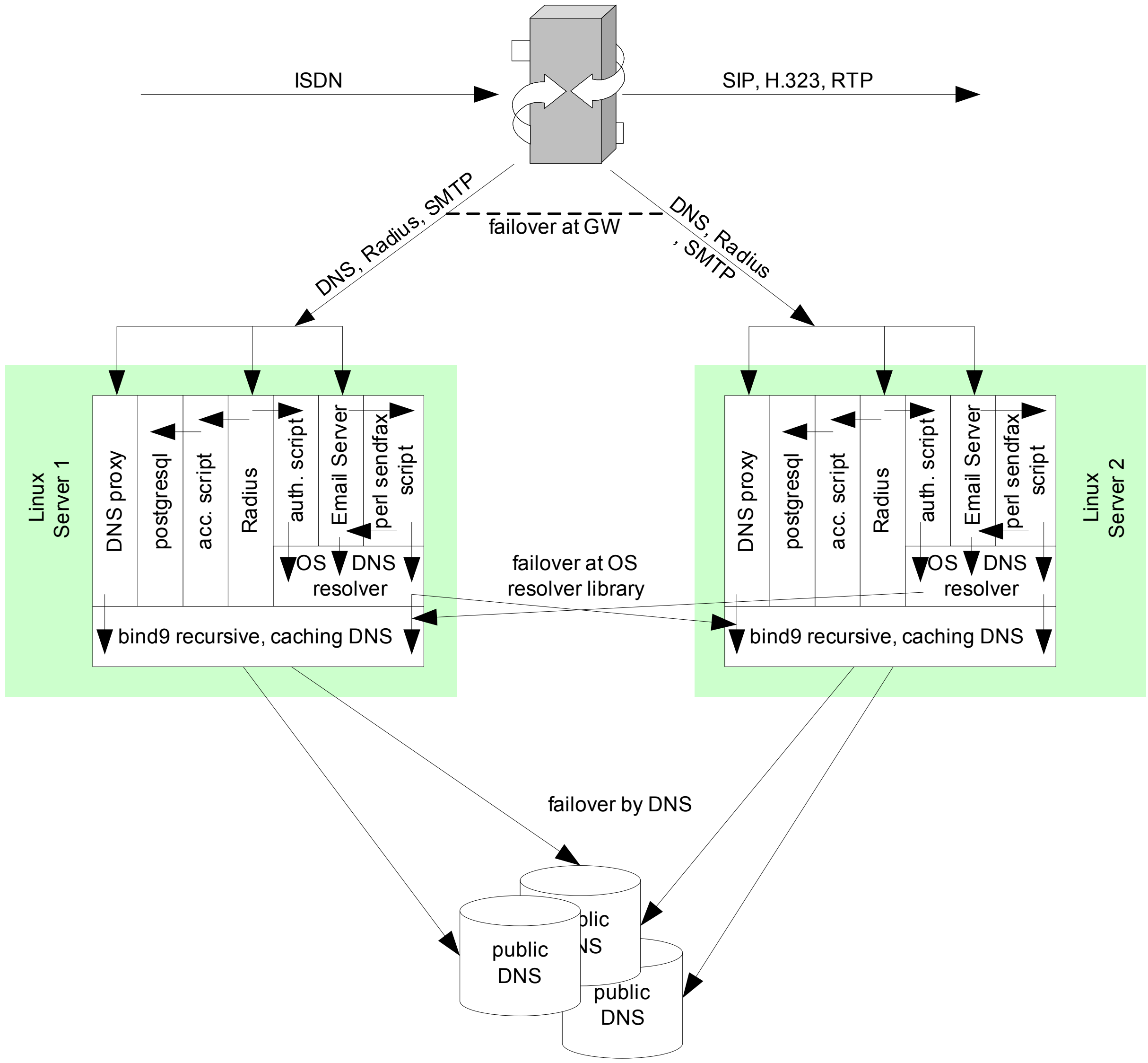


Figure 7Failover with single gateway and doubled Helper Server

8 Appendix

8.1 Cisco Gateway Version

output of the “show version” command:

Cisco Internetwork Operating System Software
IOS (tm) 5300 Software (C5300-JS-M), Version 12.3(10), RELEASE SOFTWARE (fc3)
Copyright (c) 1986-2004 by cisco Systems, Inc.
Compiled Tue 17-Aug-04 00:53 by kellythw
Image text-base: 0x60008AEC, data-base: 0x618A8000

ROM: System Bootstrap, Version 12.0(2)XD1, EARLY DEPLOYMENT RELEASE SOFTWARE (fc1)
BOOTLDR: 5300 Software (C5300-BOOT-M), Version 12.0(4)T1, RELEASE SOFTWARE (fc1)

backup5300 uptime is 4 minutes
System returned to ROM by reload at 14:29:29 GMT Tue Sep 21 2004
System restarted at 14:30:19 GMT Tue Sep 21 2004
System image file is "flash:c5300-js-mz.123-10.bin"

cisco AS5300 (R4K) processor (revision A.32) with 131072K/16384K bytes of memory.
Processor board ID 19550053
R4700 CPU at 150MHz, Implementation 33, Rev 1.0, 512KB L2 Cache
Channelized E1, Version 1.0.
Bridging software.
X.25 software, Version 3.0.0.
SuperLAT software (copyright 1990 by Meridian Technology Corp).
TN3270 Emulation software.
Primary Rate ISDN software, Version 1.1.
Backplane revision 2
Manufacture Cookie Info:
 EEPROM Type 0x0001, EEPROM Version 0x01, Board ID 0x30,
 Board Hardware Version 3.1, Item Number 800-2544-03,
 Board Revision D0, Serial Number 19550053,
 PLD/ISP Version 0.0, Manufacture Date 14-Apr-2000.
1 Ethernet/IEEE 802.3 interface(s)
1 FastEthernet/IEEE 802.3 interface(s)
35 Serial network interface(s)
4 Channelized E1/PRI port(s)
30 DSP(s), 60 Voice resource(s)
128K bytes of non-volatile configuration memory.
16384K bytes of processor board System flash (Read/Write)
8192K bytes of processor board Boot flash (Read/Write)

Configuration register is 0x2102

8.2 GG related documents

8.2.1 ENUM and DNS

RFC 2915	The Naming Authority Pointer (NAPTR) DNS Resource Record. M. Mealling, R. Daniel. September 2000. (Obsoleted by RFC3401, RFC3402, RFC3403, RFC3404) (Updates RFC2168)
RFC 2916	E.164 number and DNS. P. Faltstrom. September 2000. (Obsoleted by RFC3761)
RFC 3401	Dynamic Delegation Discovery System (DDDS) Part One: The Comprehensive DDDS. M. Mealling. October 2002. (Obsoletes RFC2915, RFC2168) (Updates RFC2276)
RFC 3402	Dynamic Delegation Discovery System (DDDS) Part Two: The Algorithm. M. Mealling. October 2002. (Obsoletes RFC2915, RFC2168)
RFC 3403	Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database. M. Mealling. October 2002. (Obsoletes RFC2915, RFC2168)
RFC 3404	Dynamic Delegation Discovery System (DDDS) Part Four: The Uniform Resource Identifiers (URI). M. Mealling. October 2002. (Obsoletes RFC2915, RFC2168)
RFC 3761	The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM). P. Faltstrom, M. Mealling. April 2004. (Obsoletes RFC2916)

RFC 3762	Telephone Number Mapping (ENUM) Service Registration for H.323. O. Levin. April 2004.
RFC 3764	enumservice registration for Session Initiation Protocol (SIP) Addresses-of-Record. J. Peterson. April 2004.
ETSI TS 102 172	Services and Protocols for Advanced Networks (SPAN); Minimum requirements for interoperability of ENUM implementations

8.2.2 DTMF

RFC 2833	RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals. H. Schulzrinne, S. Petrack. May 2000
Cisco Documentation	http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t11/fth3dtmf.htm

8.2.3 SIP

RFC 3261	SIP: Session Initiation Protocol. J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler. June 2002. (Obsoletes RFC2543) (Updated by RFC3265, RFC3853)
RFC 3263	Session Initiation Protocol (SIP): Locating SIP Servers. J. Rosenberg, H. Schulzrinne. June 2002. (Obsoletes RFC2543)
RFC 3264	An Offer/Answer Model with Session Description Protocol (SDP). J. Rosenberg, H. Schulzrinne. June 2002. (Obsoletes RFC2543)

8.2.4 RTP

RFC 3550	RTP: A Transport Protocol for Real-Time Applications. H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson. July 2003. (Obsoletes RFC1889)
RFC 3551	RTP Profile for Audio and Video Conferences with Minimal Control. H. Schulzrinne, S. Casner. July 2003. (Obsoletes RFC1890)

8.2.5 Radius

RFC 2865	Remote Authentication Dial In User Service (RADIUS). C. Rigney, S. Willens, A. Rubens, W. Simpson. June 2000. (Obsoletes RFC2138) (Updated by RFC2868, RFC3575)
RFC 2866	RADIUS Accounting. C. Rigney. June 2000. (Obsoletes RFC2139) (Updated by RFC2867)

RFC 3575	IANA Considerations for RADIUS (Remote Authentication Dial In User Service). B. Aboba. July 2003. (Updates RFC2865)
Cisco Documentation	http://www.cisco.com/univercd/cc/td/doc/product/access/acs_serv/vapp_dev/vsaig3.htm

8.2.6 SMTP

RFC 2821	Simple Mail Transfer Protocol. J. Klensin, Ed.. April 2001. (Obsoletes RFC0821, RFC0974, RFC1869)

8.3 Future Work

Following are some features which might be interesting for an enhanced version of the generic gateway (version 2):

- blacklist for incoming calls
- T.38 fax
- call transfer
- multiple number blocks, private ENUM trees (43 720 ...)
- hunt groups (if an ENUM destination fail, try another destination – not DDDS conform)
- debugging and support tool (web based, asterisk based...)