Cairo University
Faculty of Engineering
Electronics and Electrical Communications Department

Professional Masters Program – Major Telecommunications

ECP 610: Multimedia Communications

Part 3: VoIP Signaling Protocols

Dr. Wagdy Anis Aziz, Adjunct Doctor
Senior Manager, Core Network Support, Mobinil
wanis@mobinil.com
+201222201073

VoIP Signaling Protocols

VoIP Signaling Protocols

1-H.323

ITU-T Recommendation H.323 Version 4 , CISCO IP Phones.

2- SIP

IETF RFC 2543 Session Initiation Protocol.

IMS , SIP IP Phones

3-MGCP /Megaco/H.248

Call-Agents (MGC) & Gateways (MG)

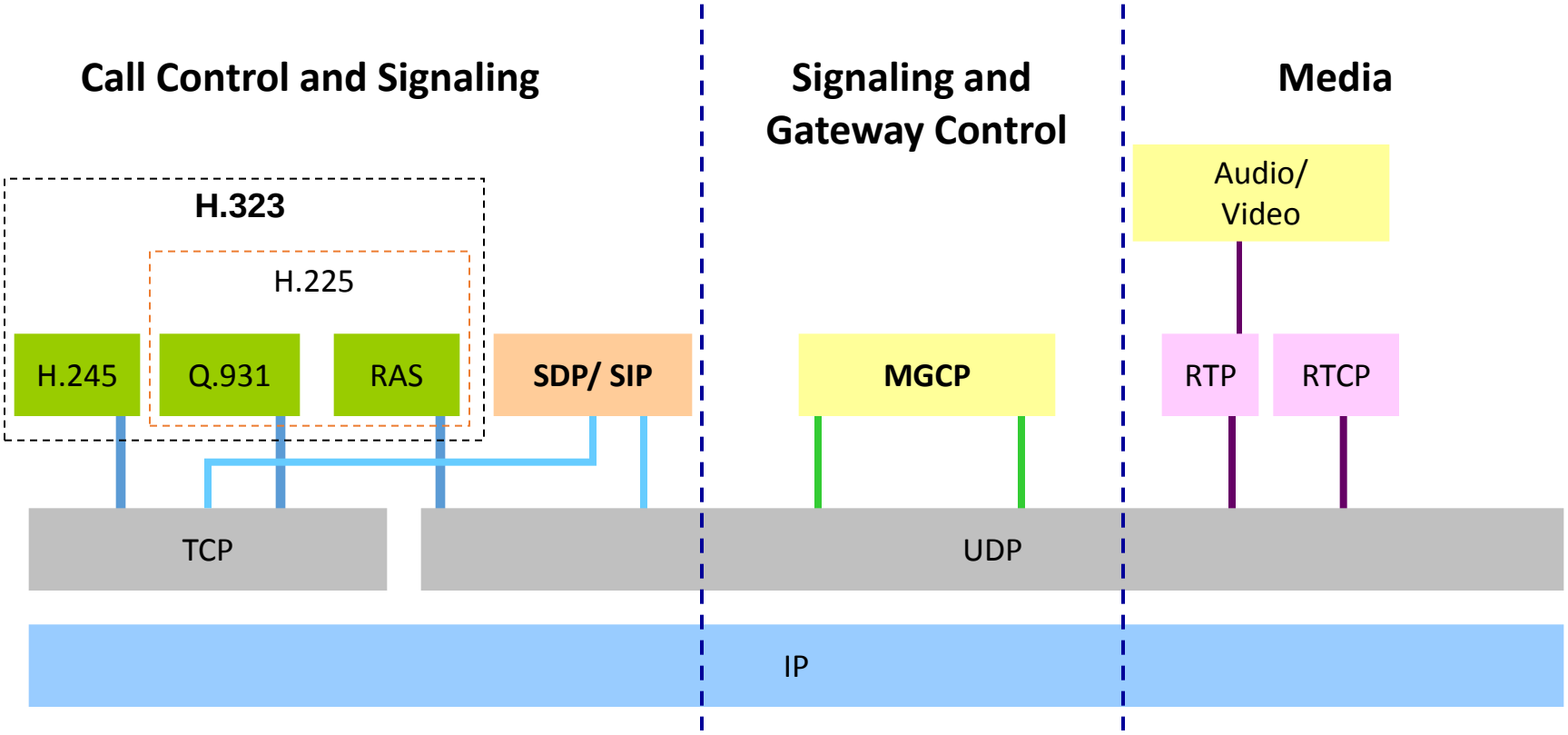
Same Protocol but different suppliers

Megaco from IETF and H.248 from ITU

Used between MGC and MGW for VoIP between two MGW under the same MGC

4- BICC used between two different MGCs to pass VoIP

SIP, H.323 and MGCP



H225: Call control signaling
H245: Control channel signaling, Media control
IP: Internet Protocol
MGCP: Media Gateway Control Protocol
Q.931: ISDN signaling
RAS: Registration, Admission, Status

RTCP: RTP Control Protocol
RTP: Real-time Transport Protocol
SDP: Session Description Protocol
SIP: Session Initiation Protocol
TCP: Transport Control Protocol
UDP: User datagram Protocol

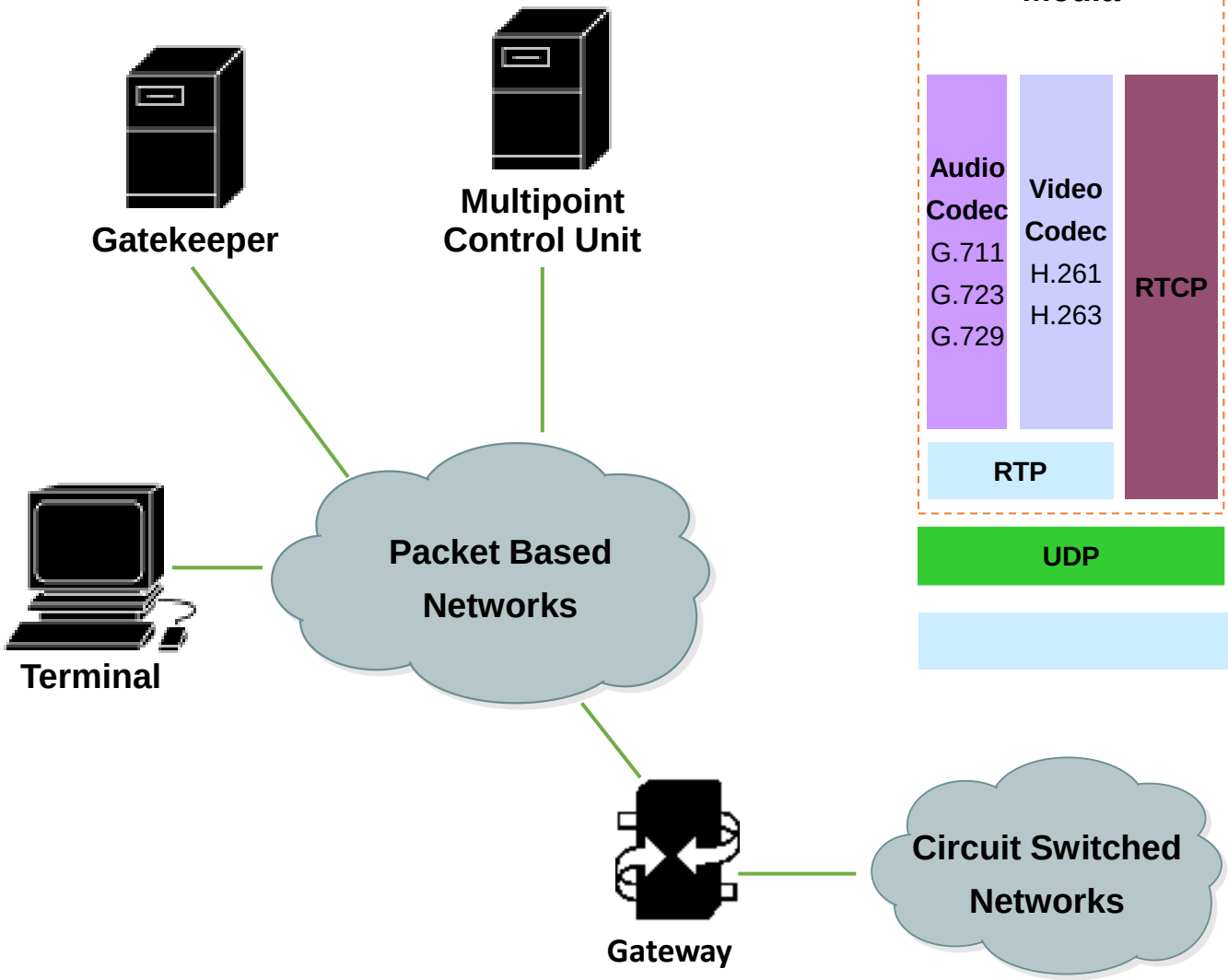
VoIP Signaling Protocols (H.323)

What is H.323 ?

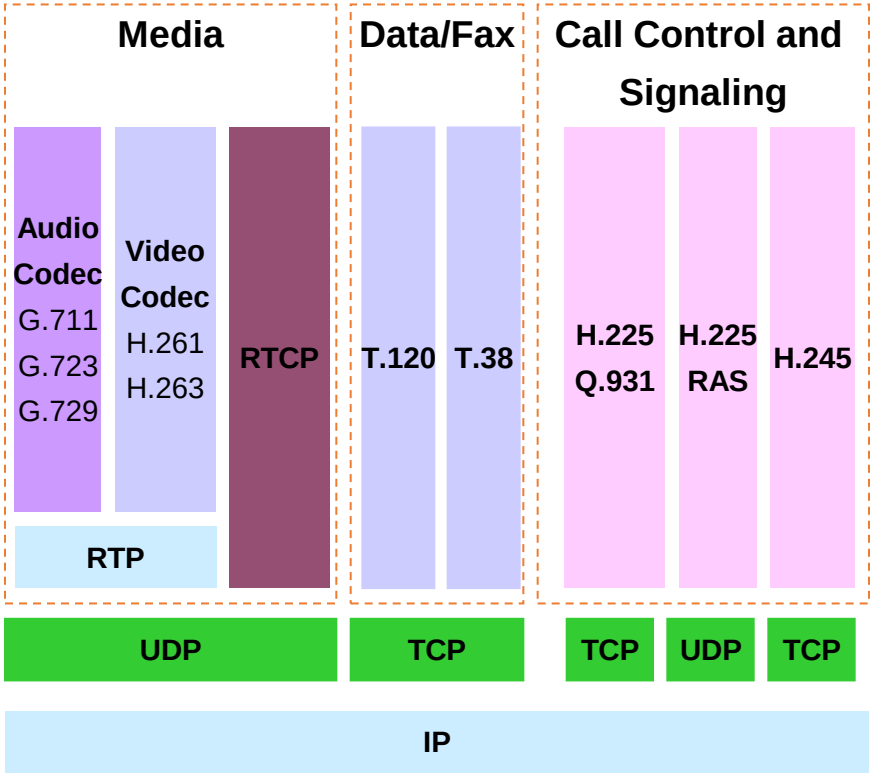
Describes terminals and other entities that provide multimedia communications services over Packet Based Networks (PBN) which may not provide a guaranteed Quality of Service. H.323 entities may provide real-time audio, video and/or data communications.

ITU-T Recommendation H.323 Version 4

H.323 Components - 1



H.323 is an "Umbrella" Specification



H.323 Components - 2

- **H.323 Terminals**
  - **IP Phone**
  - **Software** implemented on PC like NetMeeting.
- **H.323 Gateway**
  - Gateway can provide translation between entities in a packet switched network (example, IP network) and circuit switched network (example, PSTN network).
- **H.323 Gatekeepers**
  - **Address translation**.

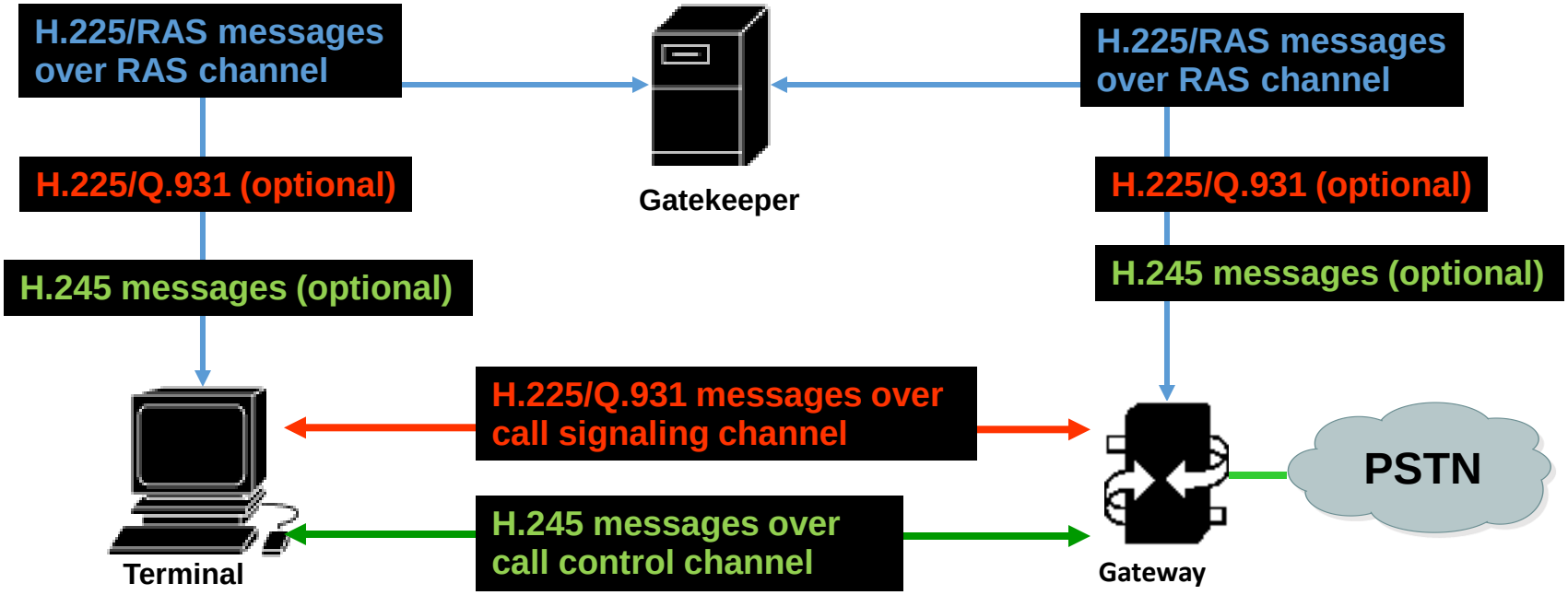
    From IP to E164
  - **Admission control.**

    To control the service required by the customer.
  - **Bandwidth control**.

    Control the number of multimedia calls to save the Bandwidth.
- **H.323 Multipoint Control Unit**

  MCU provide support for **conferences** of three or more endpoints.

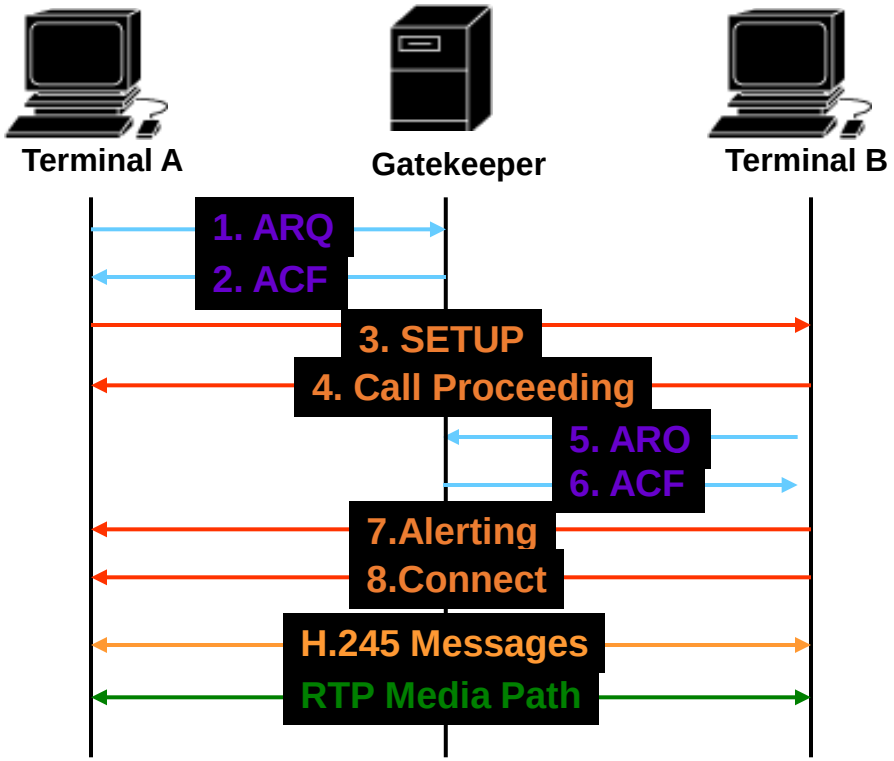
H.323 Components and Signaling



- RAS – Registration, Admission and Status Protocol used for communicating between an H.323 endpoint and a gatekeeper.
- Q.931 – A protocol for call setup and call establishment between terminals.
- H.245 – A protocol for capabilities advertisement, media channel establishment and conference control and Call Control

Simplified H.323 Call Setup

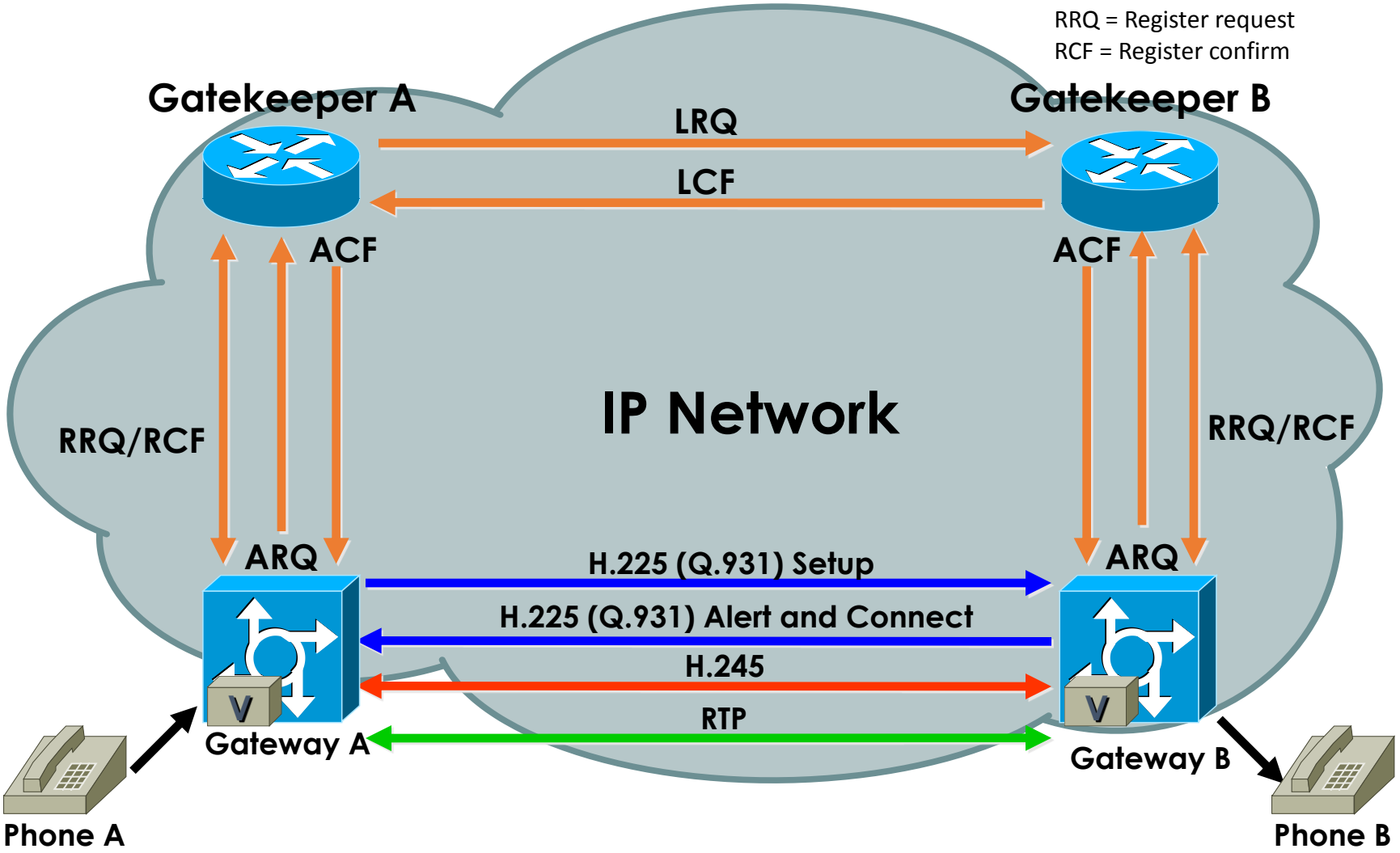
- Both endpoints have previously registered with the gatekeeper.
- Terminal A initiate the call to the gatekeeper. (RAS messages are exchanged). 1 and 2
- The gatekeeper provides information for Terminal A to contact Terminal B.
- Terminal A sends a SETUP message to Terminal B. 3
- Terminal B responds with a Call Proceeding message and also contacts the gatekeeper for permission. 4,5,6.
- Terminal B sends a Alerting and Connect message.7,8
- Terminal B and A exchange H.245 messages to determine master slave, terminal capabilities, and open logical channels.
- The two terminals establish RTP media paths.



LRQ = Location request
LCF = Location confirm
ARQ = Admission request
ACF = Admission confirm
RRQ = Register request
RCF = Register confirm

Basic H.323 Call Setup

LRQ = Location request
LCF = Location confirm
ARQ = Admission request
ACF = Admission confirm
RRQ = Register request
RCF = Register confirm



VoIP Signaling Protocols (SIP)

Contents

What's SIP?

- SIP: Session Initiation Protocol
  - Session Initiation Protocol - An application layer signaling protocol that defines initiation, modification and termination of interactive, multimedia communication sessions between users.
  - Setting up, controlling and tearing down sessions
  - IETF RFC 2543 Session Initiation Protocol
- SIP is an IETF protocol for multi-media sessions
  - Sessions including text, video, voice, etc.
- SIP is one of the Internet protocol
  - Text based like HTTP
  - Request/Reply protocol
  - Widely used for successful Internet
- Can be based on UDP/TCP/SCTP, most of the case now use UDP

Session Related Protocols

- SDP (Session Description Protocol)
  - Always is included in SIP message body
  - Session description (SDP) separated from Session management (SIP)
- RTP (Real-time Transmission Protocol)
  - Media transmission e.g. voice, video
- RTCP (Real-time Transmission Control Protocol)
  - Report and adjust the media transmission

Session Related Protocols

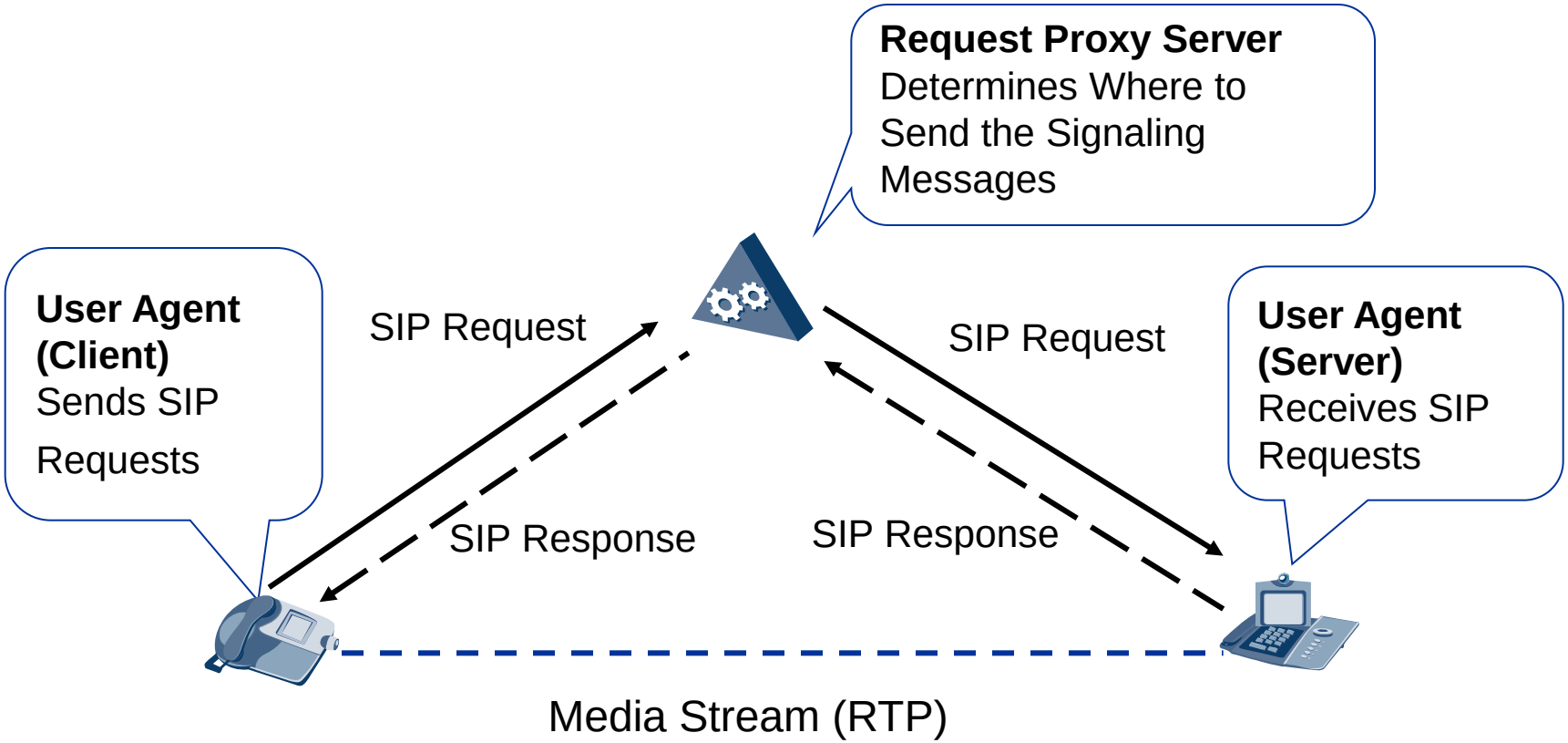
- Signaling protocol
  - Registration, user locating, and routing
  - Session establishment, modification, and release

- Media transport protocol
  - Transmission of voice and video

SIP – signaling protocol

Session description (SDP) is separated from session management (SIP).

Simplified SIP Network Architecture



SIP Network Elements

- SIP use client/server architecture

Logical SIP entities are:

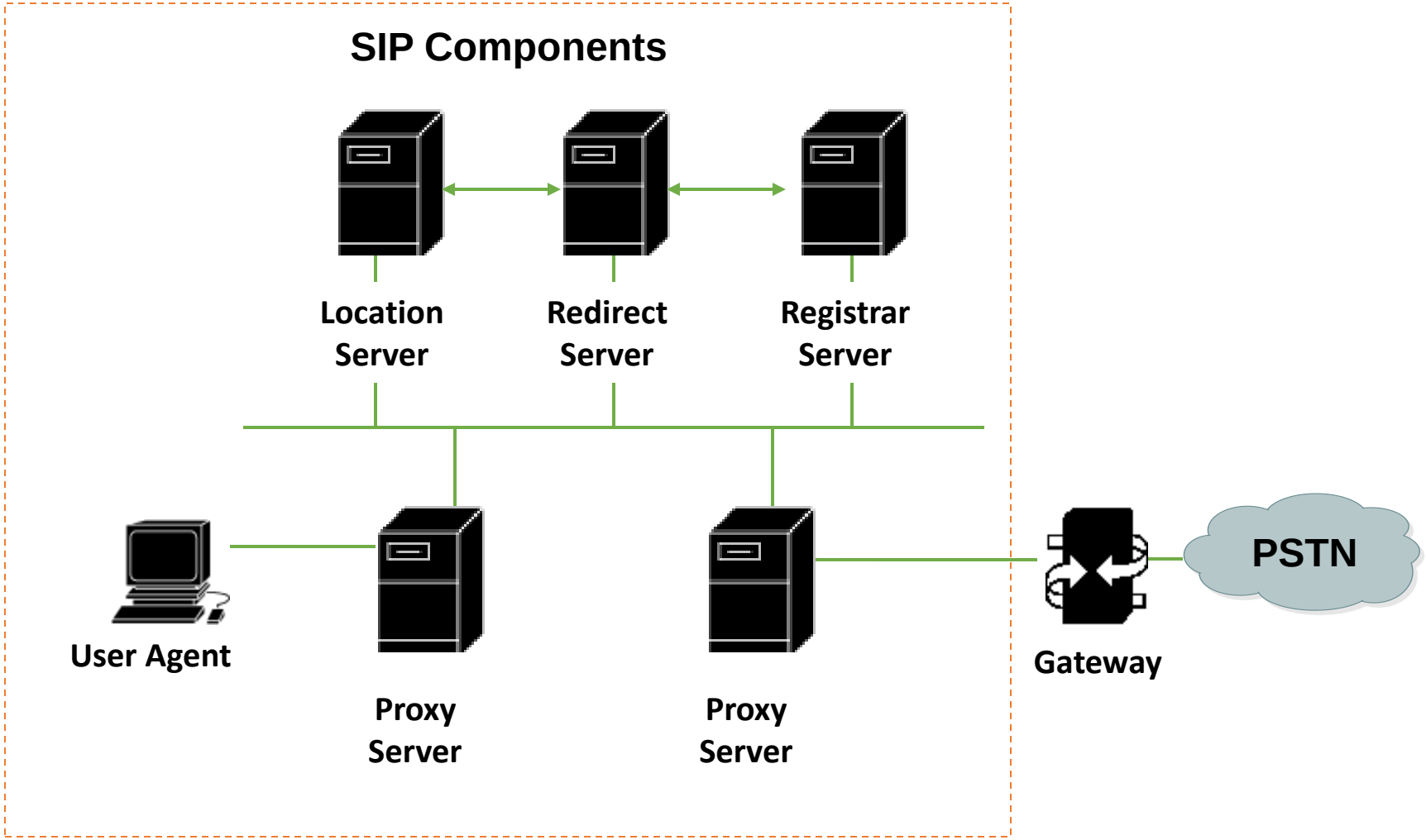
User Agents (UA)

- **User Agent Client (UAC):** Initiates SIP requests
- **User Agent Server (UAS):** Returns SIP response
- Both UAC and UAS can terminate a call

Network Servers

- **Register:** Maintains the location of SIP users, sip clients needs to update the location using register request
- **Proxy:** decides next hop and forwards request
- **Redirect:** Accepts SIP request and translates to new address

SIP Network Elements



SIP NEs definitions - 1

User Agents

- An application that initiates, receives and terminates calls.
  - User Agent Clients (UAC) – An entity that initiates a call.
  - User Agent Server (UAS) – An entity that receives a call.

  Both UAC and UAS can terminate a call.

Proxy Server

- The most important tasks are **Routing, Authentication and billing**.
- There are two types of SIP Proxy servers:
- Stateless Servers

  Simple messages forwarder, Faster.
- Stateful Servers

  Create a state and keep it until the transaction finishes.

  Used in billing and forking.

SIP NEs definitions - 2

Registrar Server

- **A server that accepts REGISTER requests.**
- **Extracts the users data (IP Address, port and user name)**
- **Stores the information in Location Database.**
- Map bob@b.com to bob@1.2.3.4:5060
- A registrar server is typically co-located with a proxy server and may offer location services.

SIP NEs definitions - 3

Location Server

A location server is used by a SIP redirect or proxy server to obtain information about a called party's possible locations.
To give you the address of the callee.

Redirect Server

- **A server that accepts a SIP request and send back a reply containing a list of current location of user.**
- Unlike a proxy server, the redirect server does not initiate its own SIP request.
- Unlike a user agent server, the redirect server does not accept or terminate calls.

SIP URI – SIP Addressing

- SIP URI : A SIP URI is in the same format as an E-mail address
  - 2 types of SIP URI
    - Address-of-record (AOR): It identifies a user, and is open to the public, for instance -- **SIP:admin@wagdy.com**
    - Full qualified domain name (FQDN) or IP address: It identifies a device, for instance – **SIP:admin@172.19.1.193**

Contents

Contents

Message Types

- SIP messages can be classified into two types:
  - Request
    - Initiates a session.
  - Response
    - Responds to a request.

SIP Messages, Requests and Responses

SIP components communicate by exchanging SIP messages:

- SIP Requests:
  - INVITE – Initiates a call by inviting user to participate in session.
  - ACK - Confirms that the client has received a final response to an INVITE request.
  - BYE - Indicates termination of the call.
  - CANCEL - Cancels a pending request.
  - REGISTER – Registers the user agent.

- SIP Responses:
  - 1xx - Provisional Responses.
  - 100 Trying and 180 Ringing
  - 2xx - Successful Responses.
  - 200 OK
  - 3xx - Redirection Responses.
  - 302 Moved Temporary
  - 4xx – Negative Final Responses.
  - 486 Busy Here
  - 5xx - Server Failure Responses.
  - 500 Server Error
  - 503 Server Unavailable.
  - 6xx - Global Failures Responses.
  - 604 Does not exist

SIP Message Type -- Request

Basic Request

- **INVITE:** to initiate a session
- **ACK**: the response of INVITE
- **CANCEL**: to cancel a session
- **BYE**: to terminate a session
- **REGISTER**: to register in a server
- **OPTIONS**: for querying servers about their capabilities

Extended Request

- **MESSAGE**: is applied to IM
- **SUBSCRIBE** : to subscribe to a notify event
- **NOTIFY**: to send a notify event
- **UPDATE**: to modify the session attributes at the establishment stage of a call
- **PUBLISH**: to distribute its event state to the status server
- **PRACK**: to indicate the reliability of a temporary response

SIP Message Type -- Response

- SIP response type:
  - **1xx:** Provisional -- request received, continuing to process the request;
  - **2xx:** Success -- the action was successfully received, understood, and accepted;
  - **3xx:** Redirection -- further action needs to be taken in order to complete the request;
  - **4xx:** Client Error -- the request contains bad syntax or cannot be fulfilled at this server;
  - **5xx:** Server Error -- the server failed to fulfill an apparently valid request;
  - **6xx:** Global Failure -- the request cannot be fulfilled at any server.

Transaction and Dialog

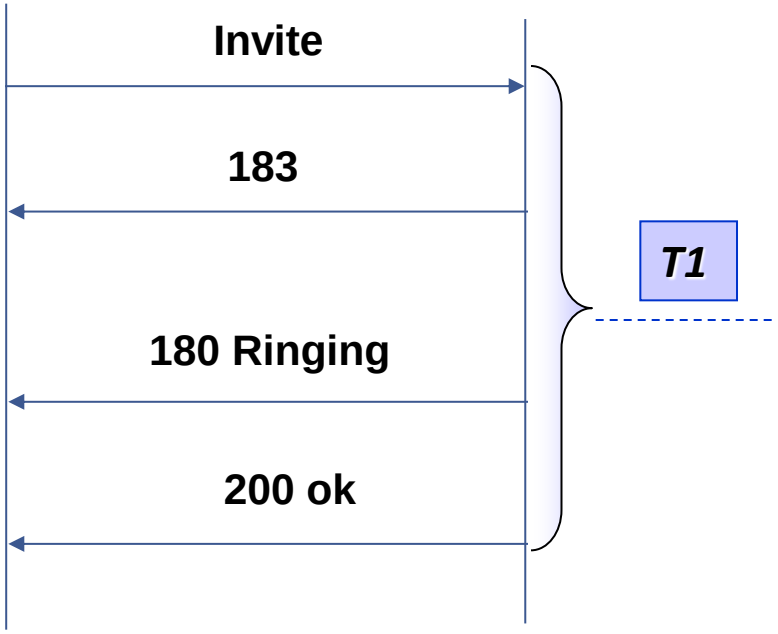
- Transaction
  - comprises all messages from the 1st request and all the response.
  - At least includes one final response (not 1xx Response)
  - Use branch field in Via header and Cseq header to identify

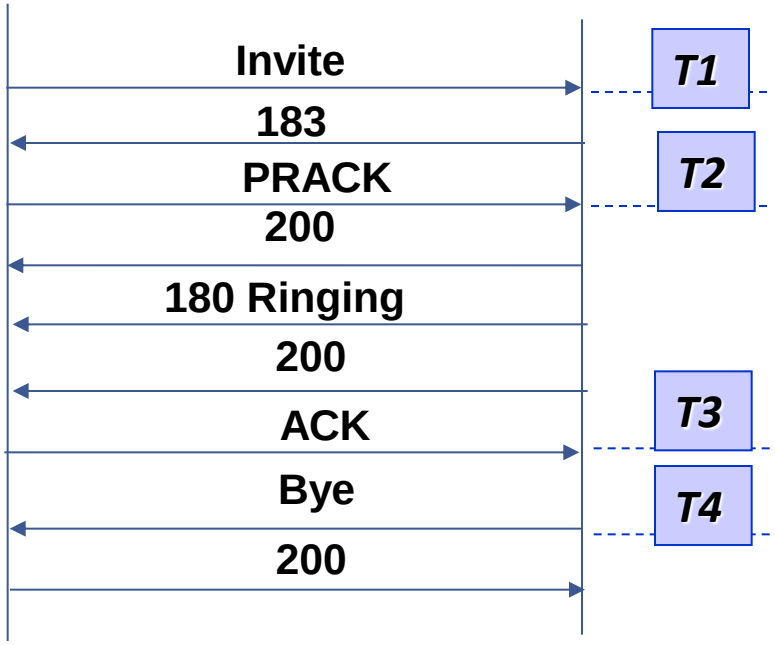
- Dialog
  - Composed with more transactions
  - INVITE is the only command to create a Dialog.
  - Identified by Call-ID, Local Tag and Remote Tag

Transaction and Dialog (*Cont.*)

- **Transaction**
  - A request and all its response

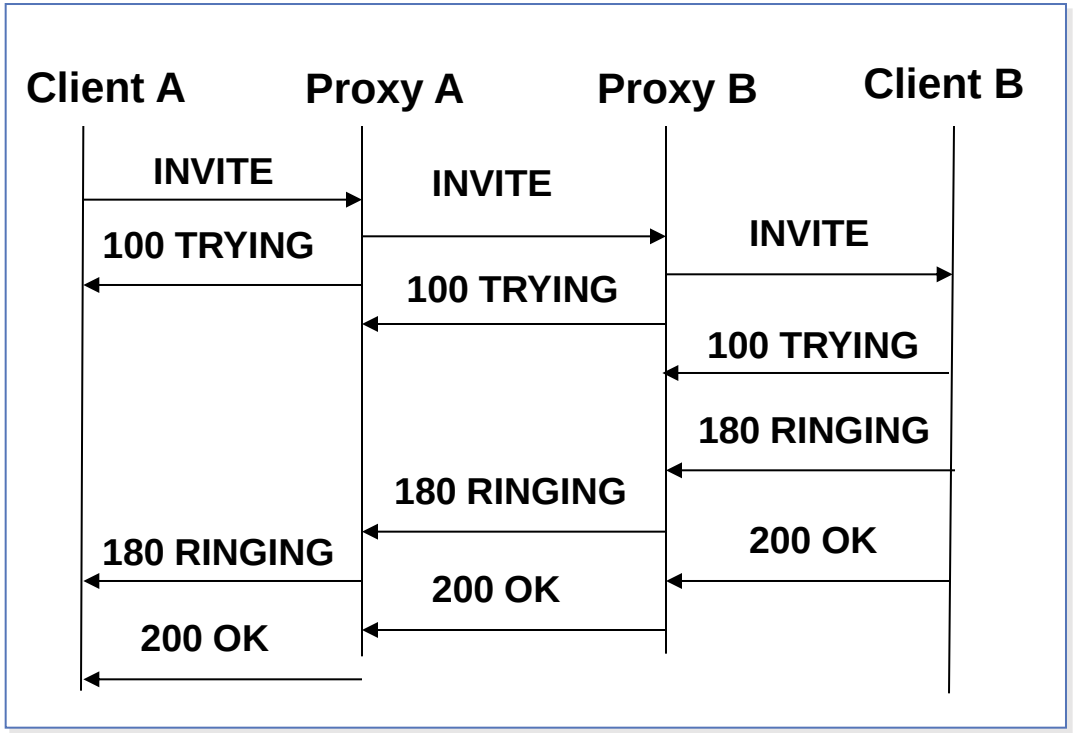
- **Dialog**
  - Consist of several transactions



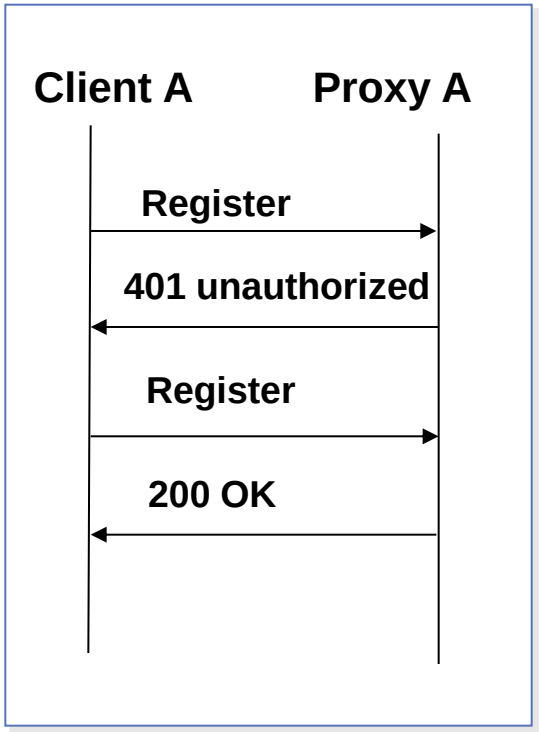


SIP Basic Flow

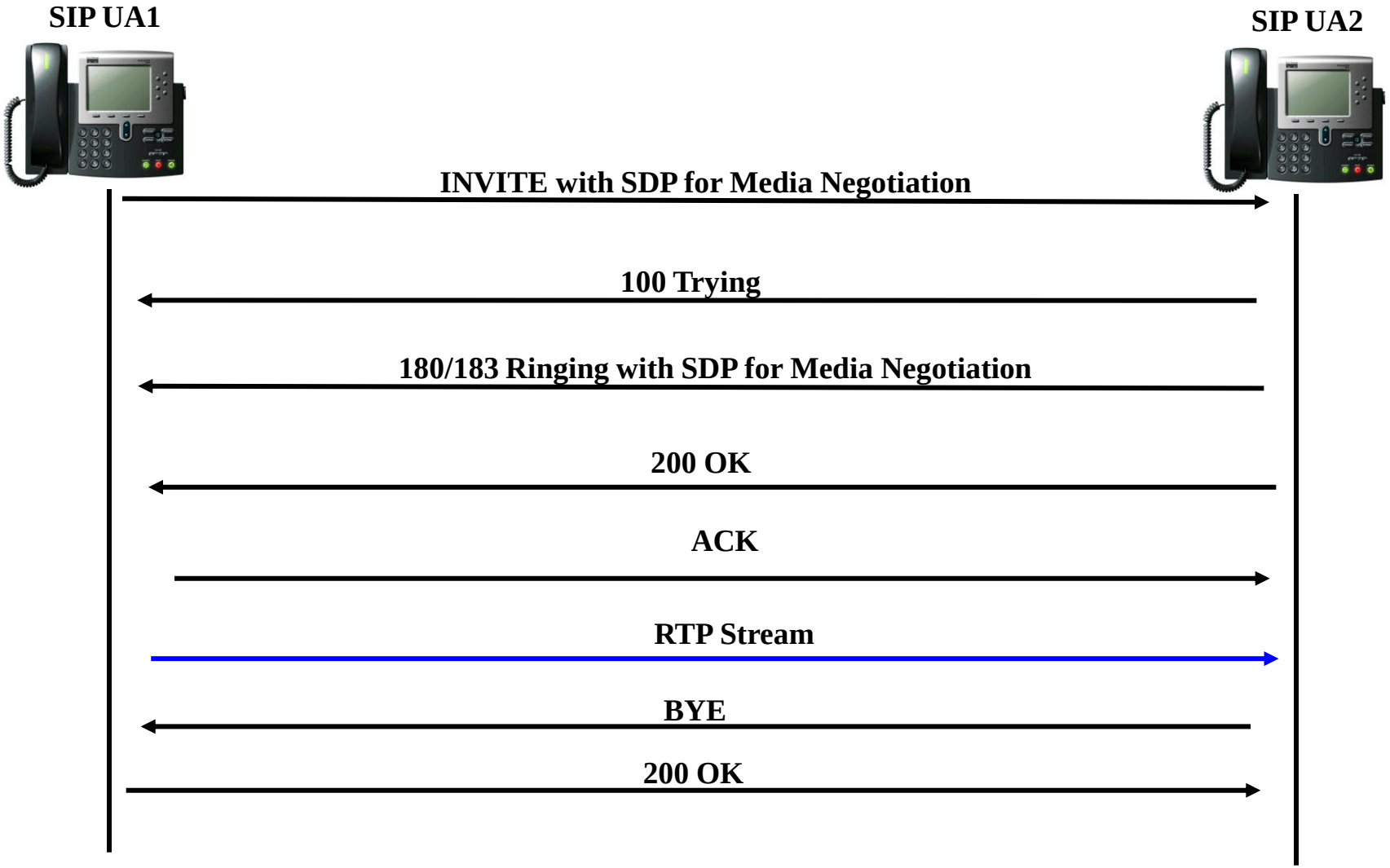
- **Session setup**



- **Registration**



SIP Basic Call Flow



Process for establishing communication

Establishing communication using SIP usually occurs in six steps:

- Registering, initiating and locating the user.
- Determine the media to use – involves delivering a description of the session that the user is invited to.
- Determine the willingness of the called party to communicate – the called party must send a response message to indicate willingness to communicate – accept or reject.
- Call setup.
- Call modification or handling – example, call transfer (optional).
- Call termination.

Contents

SIP Message Structure

- SIP is a Text-based protocol and comprise by 3 parts:
  - Request/Response-Line
  - Header
  - Body

```
⊟ Session Initiation Protocol
  ⊞ Request-Line: INVITE tel:0696395860 SIP/2
  ⊞ Message Header
  ⊞ Message body
```

```
⊟ Request-Line: INVITE tel:0696395860 SIP/2.0
    Method: INVITE
    [Resent Packet: False]
```

```
⊟ Message Header
    Via: SIP/2.0/UDP 192.168.16.57:5060;branch=z9hC
    Route: <sip:212.51.94.5:5060;lr>
  ⊞ From: <sip:+36303711408@ims.t-mobile.hu>;tag=12
  ⊞ To: <tel:0696395860>
    P-Access-Network-Info: 3GPP-UTRAN-TDD ;utran-ce
    Call-ID: 1274161795@192.168.16.57
    CSeq: 5 INVITE
    Contact: <sip:+36303711408@192.168.16.57:5060>
    Max-Forwards: 70
    User-Agent: Flexit-IMS Softphone V1.2D301SP03
    Expires: 120
```

```
⊟ Message body
  ⊟ Session Description Protocol
      Session Description Protocol Version (v): 0
    ⊞ Owner/Creator, Session Id (o): HuaweiFlexit1.0 2(
      Session Name (s): -
    ⊞ Connection Information (c): IN IP4 192.168.16.57
```

Request-line in SIP Request Message

- Request-line = **Method SP Request-URI SP SIP-Version CRLF**

Example: INVITE sip:bob@wagdy.com SIP/2.0

- Method: This specification defines six method
  - REGISTER: for registering contact information
  - INVITE: for setting up sessions
  - ACK: for setting up sessions
  - CANCEL: for cancelling a session
  - BYE: for terminating sessions
  - OPTIONS: for querying servers about their capabilities
- Request-URI: It indicates the user or service to which this request is being addressed.
- SIP-version

Response-line in SIP Response Message

- Status-line

  Example: SIP/2.0 200 OK

  - SIP-version
  - Status-code
    - 1xx
    - 2xx
    - 3xx
    - 4xx
    - 5xx
    - 6xx
  - Reason-phase

SIP Message Header

- Message-header:
  - Format: field-name: field-value

REGISTER sip:registrar.wagdy.com SIP/2.0

Via: SIP/2.0/UDP bobspc.wagdy.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
To: Bob <sip:bob@wagdy.com>
From: Bob <sip:bob@wagdy.com>;tag=456248
Call-ID: 843817637684230@998sdasdh09
CSeq: 1826 REGISTER
Contact: <sip:bob@192.0.2.4>
Expires: 7200
Content-Length: 0

SIP Message Body

- Optional, can be any protocol
- Most implementations: SDP
  - SDP: Session description protocol, convey sufficient information to calling and called party about the **user capabilities**
  - SDP includes: Media to use; Media destination; Session name and purpose; Contact information
  - SDP field have a required order

SIP Message Body - SDP Example

```
v=0
o=wagdy 868 868 IN IP4 10.216.9.200
s=Sip Call
c=IN IP4 10.216.6.108
t=0 0
m=audio 17368 RTP/AVP 8
a=rtpmap:8 PCMA/8000
```

SDP Parameter	Parameter Name	Remarks
v	Version number	v=0
o	Origin containing name	o=<user name> <session id> <version> <network type> <address type> <address>
s	session name	
c	Connection	Connection IP address for media (10.216.6.108)
t	Time	t=<start time> <stop time>
m	Media	Media format (audio); Port number(17368)
a	Attribute	Media encoding (PCM A Law); Sample rate (8000Hz)

Contents

Header Fields – Basic Headers

INVITE sip:66500002@191.169.1.110 SIP/2.0

From: <sip:44510000@191.169.1.116>;tag=1ccb6df3

To: <sip:66500002@191.169.1.110>

CSeq: 1 INVITE

Call-ID: 20973e49f7c52937fc6be224f9e52543@sx3000

Via:SIP/2.0/UDP server9.example.com;branch=z9hG4kb77ef4c23

Via: SIP/2.0/UDP 191.169.1.116:5061;branch=z9hG4bkbc427dad6

Record-Route:<sip:server9.example.com.lr>

Route:<sip:server10.example.com.lr>

Contact: <sip:44510000@191.169.1.116:5061>

Supported: 100rel,

Max-Forwards:70

User-agent: Flexit-IMS softphone V1.2D301SP03

Expires:120

Content-Length:230

Content-Type: application/sdp

SIP Header Fields

- **TO** ： the target of this request
- **FROM** ： the logical identity of the initiator of the request
- **Cseq**： command sequence No., unique in the Call-ID range
  - Consists of a sequence number and a method
  - The SIP method must be the same as that carried in the request.
- **Call-ID**: A globally unique identifier . Call-ID and tags are used to identify a dialog.
- **Via**: Identifies the route for the response. The **Via** field prevents loops in the message transfer and ensures that the request and response follow the same path.
  - The **Via** field must contain the **branch** parameter to identify the transaction.

SIP Header Fields

- **Record-Route**： It is added by a proxy in a request to force subsequent requests in the session to be routed through the proxy. It is used to create the **Route** header field in subsequent requests.
- **Route**: It is used to forcibly route a request through the listed set of proxies.
- **Contact**: It provides an address for direct communication with the user. It is present in INVITE, ACK, and REGISTER requests, success responses, call process responses, and redirection responses
- **Supported:** 100rel,: supporting 1XX responses. A terminal can return a PRACK response to improve reliability.
- **Max-Forwards**: It limits the number of hops a request can transit on the way to its destination. The error response is 483(too many hops). It is used in request messages only.

Header Fields – From & To

- TO: the target of this request
- FROM: the logical identity of the initiator of the request, possibly the user's address-of-record

Example:

```
INVITE sip:bob@wagdy.com SIP/2.0
        To: Bob <sip:bob@wagdy.com>
        From: Alice <sip:alice@atlanta.com>;tag=1928301774

                SIP/2.0 180 Ringing
                To: Bob <sip:bob@wagdy.com>;tag=a6c85cf
                From: Alice <sip:alice@atlanta.com>;tag=1928301774
```

Header Fields– From & To (*Cont.*)

- Request URI & From & To

REGISTER

```
REGISTER sip:registrar.wagdy.com SIP/2.0

To: Bob <sip:bob@wagdy.com>
From: Bob<sip:bob@wagdy.com>;tag=456248
```

destination

subscriber

INVITE

```
INVITE sip:ice@wagdy.com SIP/2.0
To:Ice <sip:ice@wagdy.com>

From: Bob<sip:bob@wagdy.com>;tag=456248
```

callee

caller

Header Fields– Contact

- Contact
  - Provide an address for direct communication with the user.
  - Present in INVITE, ACK, and REGISTER requests, success responses, call process responses, and redirection responses

Example:

```
INVITE sip:bob@wagdy.com SIP/2.0
Contact: <sip:44510000@191.169.1.116:5061>
```

Header Fields– Call-ID & Cseq

- Call-ID: a globally unique identifier
  - Call-ID and tags are used to identify a dialog.
- Cseq: as a way to identify and order transactions
  - Consists of a sequence number and a method

```
Example:
  INVITE sip:bob@wagdy.com SIP/2.0
            Call-ID: a84b4c76e66710
            CSeq: 314159 INVITE

                                 SIP/2.0 180 Ringing
                                 Call-ID: a84b4c76e66710
                                 CSeq: 314159 INVITE
```

Header Fields– Record-Route and Route

- **Record-Route**
  - Added by a proxy in a request to force subsequent requests in the session to be routed through the proxy
  - Is used to create the **Route** header field in subsequent requests
- **Route**
  - Is used to forcibly route a request through the listed set of proxies

Example:
INVITE sip:bob@wagdy.com SIP/2.
Record-Route:<scscf.wagdy.com.lr>
Record-Route:<pcscf.wagdy.com.lr>

Example:
INVITE sip:bob@wagdy.com SIP/2.
Route:<scscf.wagdy.com.lr>

Header Fields - Via

- Via:
  - Identifies the route for the response
  - MUST contain a branch parameter: to identify the transaction

Example:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP server10.wagdy.com;    branch=z9hG4bKnashds8;
Via: SIP/2.0/UDP bigbox3.site3.atlanta.com;
branch=z9hG4bK77ef4c2312983.1;
Via: SIP/2.0/UDP pc33.atlanta.com;
branch=z9hG4bK776asdhds ;received=192.0.2.1
```

Header Fields - Supported

- Supported
  - **100rel** extension provides an appropriate mechanism for the reliable transportation of the 100 response
  - The acknowledgement request method for a provisional response in **100rel** is **PRACK**
  - 100rel extension can be realized through the **Supported** header fields

Example:

```
INVITE sip:bob@wagdy.com SIP/2.0
Supported: 100rel,
```

Header Fields - Content-length/Content-Type

- Content-length
  - Indicates the size of the message body
  - If a message does not contain a message body, the value of the Content-Length header field must be set to 0.

Example:

```
INVITE sip:bob@wagdy.com SIP/2.0
Content-Length: 142
```

- Content-type
  - Indicates the media type of the message body sent to the recipient

Example:

```
INVITE sip:bob@wagdy.com SIP/2.0
Content-Type: application/sdp
```

Header Fields - Expires

- Expires
  - Gives the relative time after which the message (or content) expires

For the REGISTER message
If the value of Expires fields is 0, means this is the **DE-REGISTER** request

REGISTER sip:registrar.wagdy.com SIP/2.0
To: Bob <sip:bob@wagdy.com>
From: Bob<sip:bob@wagdy.com>;tag=456248
Expires: 0

Header Fields - Max-Forwards & User-Agent

- Max-Forwards: to limit the number of hops a request can transit on the way to its destination
  - The error response is 483 (too many hops).
  - In request message

Example:

```
INVITE sip:bob@wagdy.com SIP/2.0
Max-Forwards: 70
```

- User-agent
  - contains information about the UAC originating the request

Example:

```
INVITE sip:bob@wagdy.com SIP/2.0
User-agent: Flexit-IMS softphone V1.2D301
```

Summary

- SIP is used to establish, modify and terminate a multimedia conference, such as conference call over Internet. SIP can be used to initiate sessions as well as inviting members to sessions that have been advertised and established by other means.
- SIP request messages: SIP messages sent by a client to the server on the basis of designated operation for activation, which include such messages as INVITE, PRACK, BYE, CANCEL, UPDATE, etc.
- SIP Response Messages: Used to display the status of the requests sent by clients to the server, including the 1xx, 2xx, 3xx, 4xx, 5xx and 6xx responses and ACK.
- The functions of each command in SIP call process should be mastered as the key points.

SIP Life Examples

FUZZING Test – Introduction

- The FUZZING tests are in total extracted out of the "PROTOS Test-Suite: c07-sip" of the University of OULU
- The main focus of this test suite is how the implementation of a User Agent (UA) or a SIP Proxy Server handle INVITE requests
- This test suite focuses the robustness and security testing of the Implementation Under Test (IUT) receiving malformed and INVITE requests
- The test suite tests SIP via UDP only.
- To provoke failures the sent INVITE Messages contains one or more of the exceptional elements

FUZZING Test – Failure criteria

1. A device undergoes a fatal failure and stops functioning normally.
2. A process or a device crashes or hangs and needs to be restarted manually.
3. A process or a device crashes and restarts automatically.
4. A process consumes almost all CPU and/or memory resources for an exceptionally long or indefinite time.

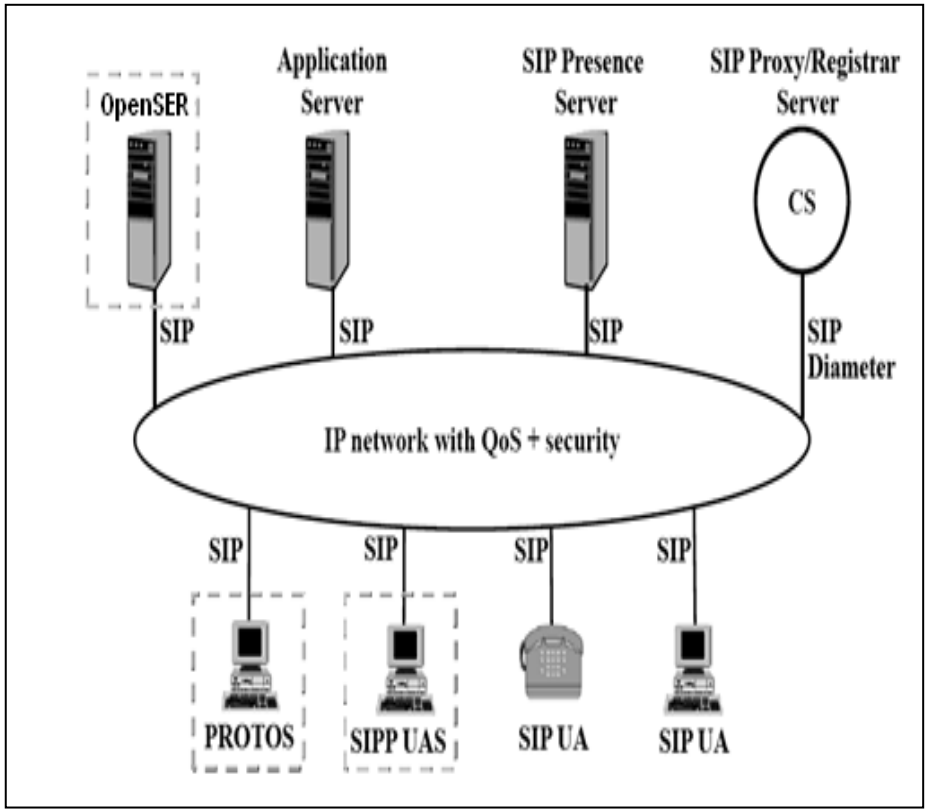
The test fails if one of the above mentioned failures occurs due to receiving an INVITE containing one or more of the exceptional elements.

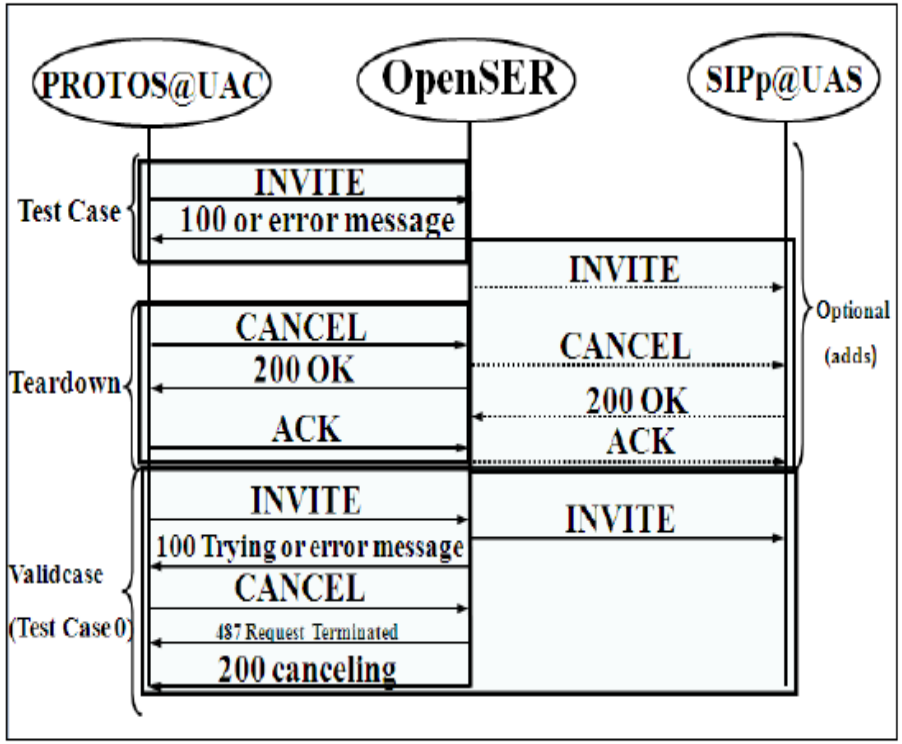
FUZZING Tests –Exceptional Element Categories

Name	Description
empty	Omitted (empty) element content
ipv4-ascii	Exceptional IPv4 addresses in ascii
overflow-general	'a' (0x61) character overflows up to 128k
overflow-slash	Overflows of '/' up to 128 kbytes
overflow-colon	Overflows of ':' up to 128 kbytes
overflow-space	Overflows of ' ' up to 128 kbytes
overflow-null	Overflows of 0x61 and nulls (0x00) mixed
overflow-leftbracket	Overflows of '<' up to 128k
overflow-rightbracket	Overflows of '>' up to 128k
overflow-at	Overflows of '@' up to 128k
overflow-equal	Overflows of '=' up to 128k
fmtstring	Format strings (eg. %s%s%s or %.4097d)
utf-8	Malformed UTF-8 sequences
integer-ascii	Pos/Neg ASCII encoded integers
ansi-escape	Malformed ANSI escape sequences
sip-version	Malformed "SIP/2.0"
content-type	Malformed "application/sdp"
sip-URI	Malformed SIP-URI
sip-tag	Malformed tags
crlf	Arrangements of CR (0x0d) and LF (0x0a)

An exceptional element is a piece of data designed to provoke undesired behavior of the test subject. A single test-case contains one or more exceptional elements. An exceptional element can violate the protocol specification, but often it is legal or in the hazy region between legal and illegal constructs. In a nutshell, an exceptional element is an input that might not have been considered properly when implementing the software.

FUZZING Test Scenario



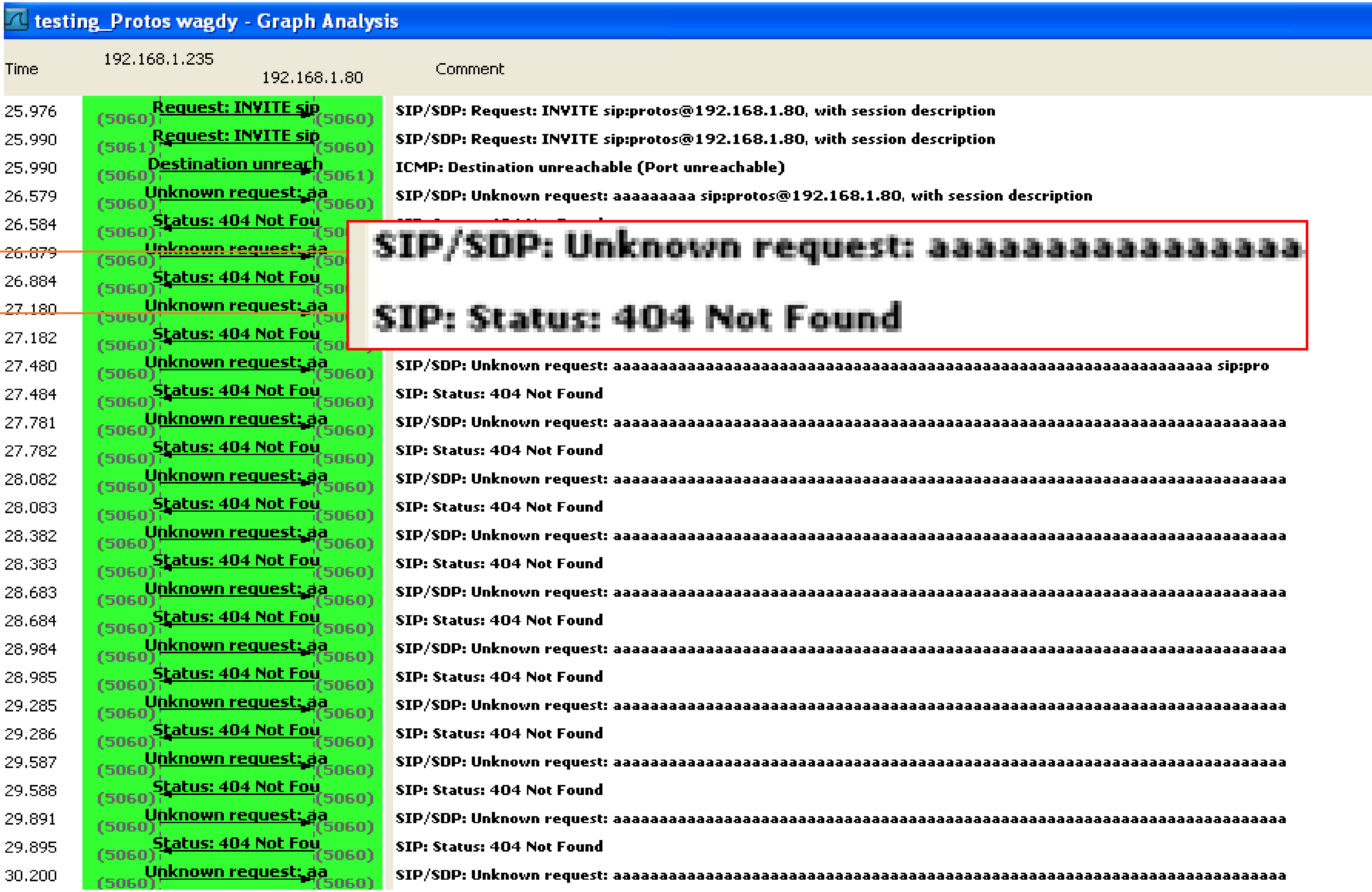


FUZZING Test Results

```
SIP Request:

aaaaaaaa sip:protos@192.168.1.80 SIP/2.0
Via: SIP/2.0/UDP rx-desktop:5060;branch=z9hG4bK00003000003
From: 3 <sip:user@rx-desktop>;tag=3
To: Receiver <sip:protos@192.168.1.80>
Call-ID: 3@rx-desktop
CSeq: 1 INVITE
Contact: 3 <sip:user@rx-desktop>
Expires: 1200
Max-Forwards: 70
Content-Type: application/sdp
Content-Length: 120

v=0
o=3 3 3 IN IP4 rx-desktop
s=Session SDP
c=IN IP4 127.0.1.1
t=0 0
m=audio 9876 RTP/AVP 0
a=rtpmap:0 PCMU/8000

SIP Response:

SIP/2.0 404 Not Found
Via: SIP/2.0/UDP rx-desktop:5060;branch=z9hG4bK00003000003;received=192.168.1.235
From: 3 <sip:user@rx-desktop>;tag=3
To: Receiver <sip:protos@192.168.1.80>;tag=fa997f81440371de71ab448ebdb9af56-6192
Call-ID: 3@rx-desktop
CSeq: 1 INVITE
Server: OpenSER (1.3.2-notls (i386/linux))
Content-Length: 0
```

PROTOS test trace for OpenSER Overflow - general, 'a' (0x61) character overflows up to 128k

FUZZING Test Results

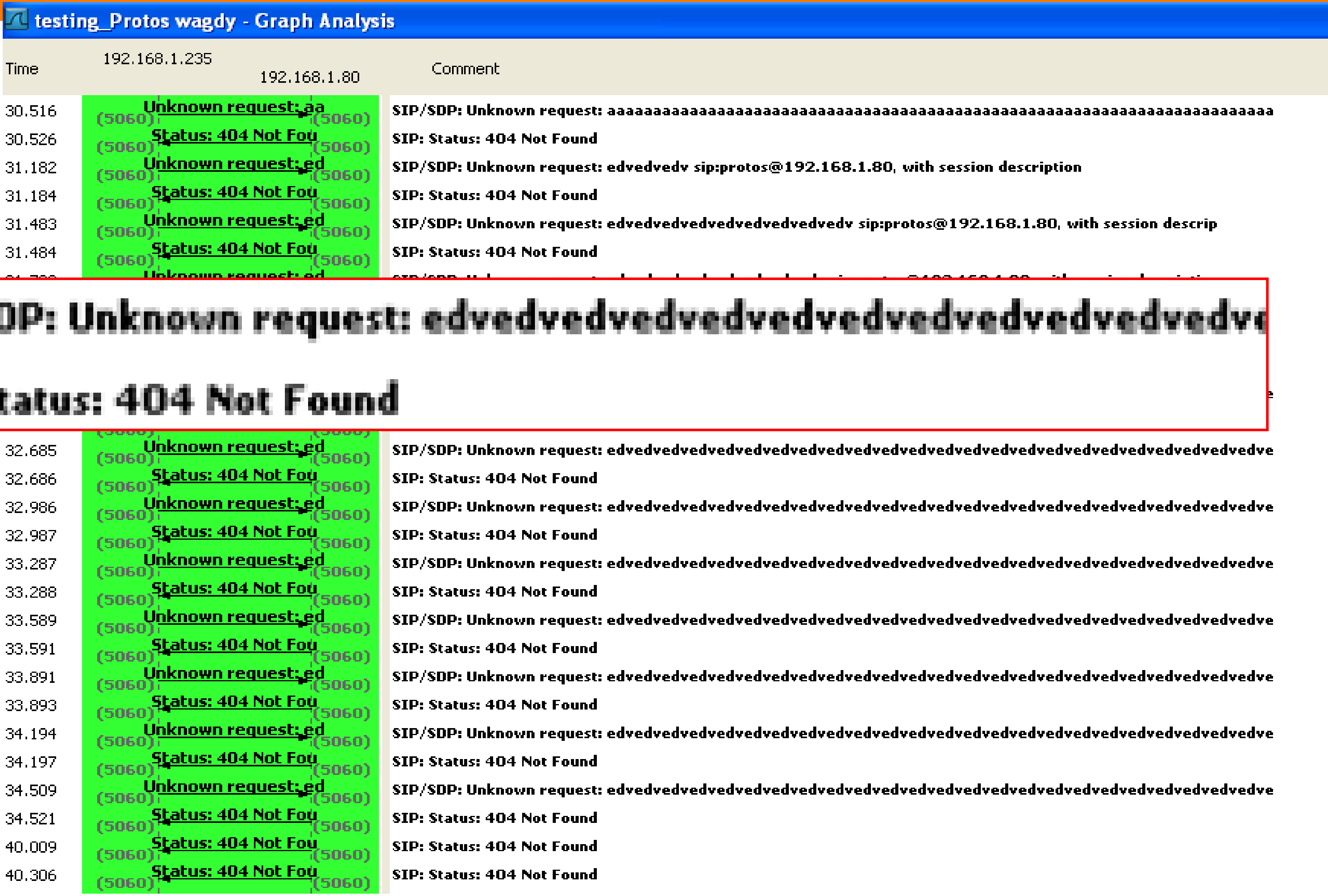


```
testing_Protos wagdy - Graph Analysis

Time      192.168.1.235                          Comment
                      192.168.1.80
25.976  (5060) Request: INVITE sip (5060)    SIP/SDP: Request: INVITE sip:protos@192.168.1.80, with session description
25.990  (5061) Request: INVITE sip (5060)    SIP/SDP: Request: INVITE sip:protos@192.168.1.80, with session description
25.990  (5060) Destination unreach (5061)    ICMP: Destination unreachable (Port unreachable)
26.579  (5060) Unknown request: aa (5060)    SIP/SDP: Unknown request: aaaaaaaaa sip:protos@192.168.1.80, with session description
26.584  (5060) Status: 404 Not Fou (50
26.879  (5060) Unknown request: aa (50
26.884  (5060) Status: 404 Not Fou (50
27.180  (5060) Unknown request: aa (50
27.182  (5060) Status: 404 Not Fou (50
27.480  (5060) Unknown request: aa (5060)    SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa sip:pro
27.484  (5060) Status: 404 Not Fou (5060)    SIP: Status: 404 Not Found
27.781  (5060) Unknown request: aa (5060)    SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
27.782  (5060) Status: 404 Not Fou (5060)    SIP: Status: 404 Not Found
28.082  (5060) Unknown request: aa (5060)    SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
28.083  (5060) Status: 404 Not Fou (5060)    SIP: Status: 404 Not Found
28.382  (5060) Unknown request: aa (5060)    SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
28.383  (5060) Status: 404 Not Fou (5060)    SIP: Status: 404 Not Found
28.683  (5060) Unknown request: aa (5060)    SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
28.684  (5060) Status: 404 Not Fou (5060)    SIP: Status: 404 Not Found
28.984  (5060) Unknown request: aa (5060)    SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
28.985  (5060) Status: 404 Not Fou (5060)    SIP: Status: 404 Not Found
29.285  (5060) Unknown request: aa (5060)    SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
29.286  (5060) Status: 404 Not Fou (5060)    SIP: Status: 404 Not Found
29.587  (5060) Unknown request: aa (5060)    SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
29.588  (5060) Status: 404 Not Fou (5060)    SIP: Status: 404 Not Found
29.891  (5060) Unknown request: aa (5060)    SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
29.895  (5060) Status: 404 Not Fou (5060)    SIP: Status: 404 Not Found
30.200  (5060) Unknown request: aa (5060)    SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```

```
SIP/SDP: Unknown request: aaaaaaaaaaaaaaaa
SIP: Status: 404 Not Found
```

PROTOS test trace for OpenSER Overflow - general, 'a' (0x61) character overflows up to 128k

FUZZING Test Results



Time	192.168.1.235		192.168.1.80	Comment
30.516	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
30.526	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
31.182	(5060)	Unknown request: ed	(5060)	SIP/SDP: Unknown request: edvedvedv sip:protos@192.168.1.80, with session description
31.184	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
31.483	(5060)	Unknown request: ed	(5060)	SIP/SDP: Unknown request: edvedvedvedvedvedvedvedvedv sip:protos@192.168.1.80, with session descrip
31.484	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
32.685	(5060)	Unknown request: ed	(5060)	SIP/SDP: Unknown request: edvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedve
32.686	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
32.986	(5060)	Unknown request: ed	(5060)	SIP/SDP: Unknown request: edvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedve
32.987	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
33.287	(5060)	Unknown request: ed	(5060)	SIP/SDP: Unknown request: edvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedve
33.288	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
33.589	(5060)	Unknown request: ed	(5060)	SIP/SDP: Unknown request: edvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedve
33.591	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
33.891	(5060)	Unknown request: ed	(5060)	SIP/SDP: Unknown request: edvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedve
33.893	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
34.194	(5060)	Unknown request: ed	(5060)	SIP/SDP: Unknown request: edvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedve
34.197	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
34.509	(5060)	Unknown request: ed	(5060)	SIP/SDP: Unknown request: edvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedvedve
34.521	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
40.009	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
40.306	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found

SIP/SDP: Unknown request: edvedvedvedvedvedvedvedvedvedvedvedvedvedve

SIP: Status: 404 Not Found

PROTOS test trace for OpenSER Overflow- general, 'edv' (0x61) character overflows up to 128k

FUZZING Test Results

testing_Protos wagdy - Graph Analysis

Time	192.168.1.235	192.168.1.80	Comment
50.303	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
50.608	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
50.904	(5060) Unknown request: %	(5060)	SIP/SDP: Unknown request: % sip:protos@192.168.1.80, with session description
50.906	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
51.205	(5060) Unknown request: %s	(5060)	SIP/SDP: Unknown request: %s%x%n sip:protos@192.168.1.80, with session description
51.206	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
51.506	(5060) Unknown request: %.	(5060)	SIP/SDP: Unknown request: %.127d sip:protos@192.168.1.80, with session description
51.506	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
51.806	(5060) Unknown request: %.	(5060)	SIP/SDP: Unknown request: %.555d sip:protos@192.168.1.80, with session description
51.807	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
52.107	(5060) Unknown request: %.	(5060)	SIP/SDP: Unknown request: %.999d sip:protos@192.168.1.80, with session description
52.108	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
52.408	(5060) Unknown request: %.	(5060)	SIP/SDP: Unknown request: %.1270d sip:protos@192.168.1.80, with session description
52.408	(5060) Status: 404 Not Fou	(506	
52.708	(5060) Unknown request: %.	(506	SIP/SDP: Unknown request: %.4097d sip:protos@192.168.1.80, with session description
52.709	(5060) Status: 404 Not Fou	(506	SIP: Status: 404 Not Found
53.009	(5060) Unknown request: %.	(506	
53.010	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
53.309	(5060) Unknown request: %.	(5060)	SIP/SDP: Unknown request: %.12700d sip:protos@192.168.1.80, with session description
53.310	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
53.610	(5060) Unknown request: %.	(5060)	SIP/SDP: Unknown request: %.127000d sip:protos@192.168.1.80, with session description
53.611	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
53.911	(5060) Unknown request: %n	(5060)	SIP/SDP: Unknown request: %n%n%n%n%n%n%n%n%n%n%n%n%n%n%x%x%x%x%x%x%x%x%x%x%x%x%x%x sip:protos@192.168.1.80,
53.912	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
54.211	(5060) Unknown request: %n	(5060)	SIP/SDP: Unknown request: %n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%
54.212	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
54.513	(5060) Unknown request: %n	(5060)	SIP/SDP: Unknown request: %n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%
54.515	(5060) Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found

PROTOS test trace for OpenSER Format strings (eg. %s%s%s or %.4097d)

PROTOS Test Results – Life Sample

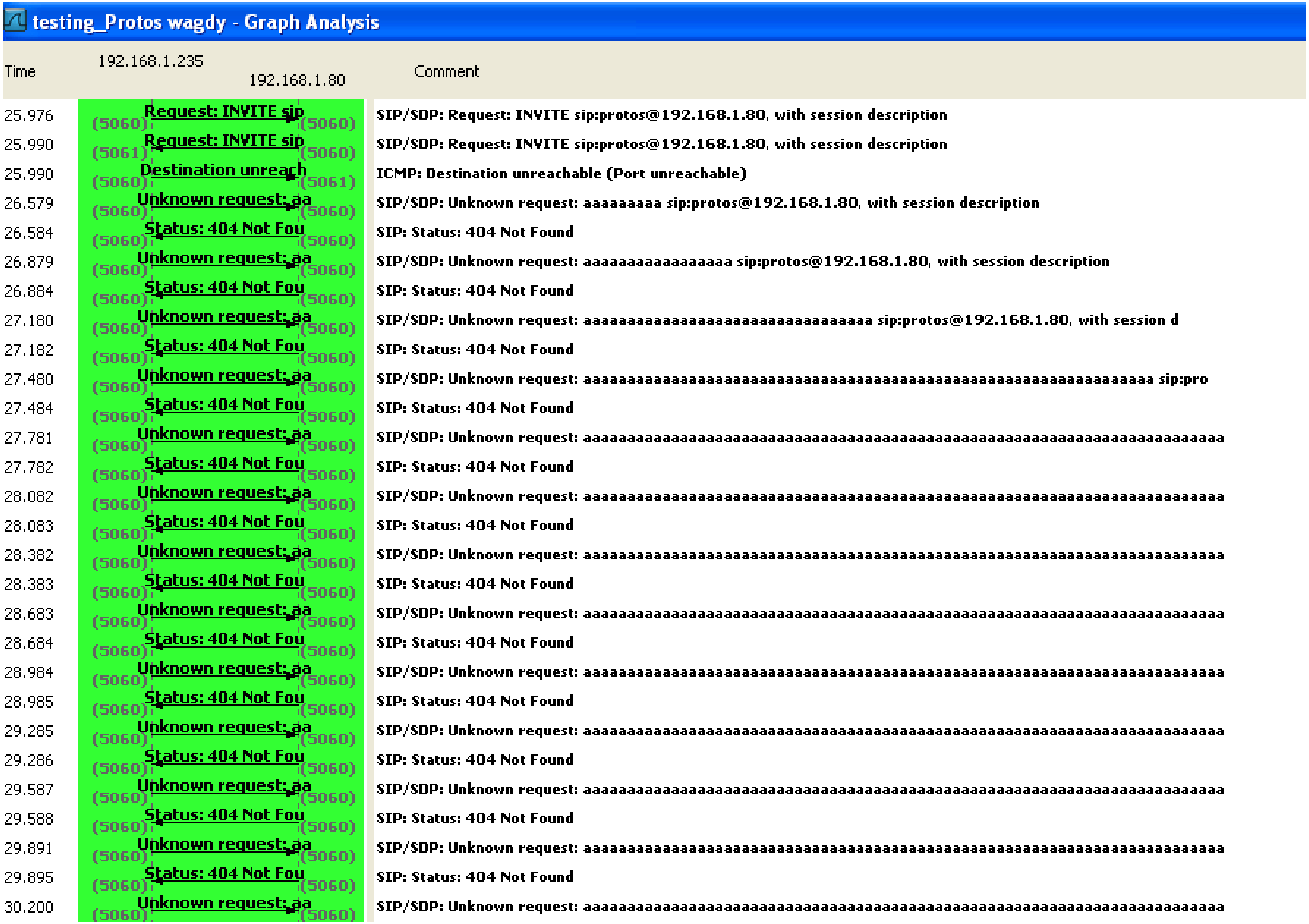
testing_Protos wagdy - Wireshark

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter: sip Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
15	25.976179	192.168.1.235	192.168.1.80	SIP/SDP	Request: INVITE sip:protos@192.168.1.80, with session descriptio
16	25.989600	192.168.1.80	192.168.1.235	SIP/SDP	Request: INVITE sip:protos@192.168.1.80, with session descriptio
17	25.989649	192.168.1.235	192.168.1.80	ICMP	Destination unreachable (Port unreachable)
19	26.578607	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaa sip:protos@192.168.1.80, with session
20	26.584060	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
21	26.879184	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaaaaaaaaaa sip:protos@192.168.1.80, with
22	26.883974	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
23	27.179784	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa sip:protos@19
24	27.182245	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
25	27.480393	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
26	27.484351	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
27	27.781004	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
28	27.782242	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
29	28.081615	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
30	28.082611	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
31	28.382276	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
32	28.383442	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
33	28.682991	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
34	28.684085	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
36	28.983870	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
37	28.985324	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
39	29.284952	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
40	29.286311	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
44	29.586516	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
45	29.588323	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
57	29.891061	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
58	29.894866	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found

FUZZING Test Results – Life Sample



testing_Protos wagdy - Graph Analysis

Time	192.168.1.235		192.168.1.80	Comment
25.976	(5060)	Request: INVITE sip	(5060)	SIP/SDP: Request: INVITE sip:protos@192.168.1.80, with session description
25.990	(5061)	Request: INVITE sip	(5060)	SIP/SDP: Request: INVITE sip:protos@192.168.1.80, with session description
25.990	(5060)	Destination unreach	(5061)	ICMP: Destination unreachable (Port unreachable)
26.579	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaa sip:protos@192.168.1.80, with session description
26.584	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
26.879	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaa sip:protos@192.168.1.80, with session description
26.884	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
27.180	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa sip:protos@192.168.1.80, with session d
27.182	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
27.480	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa sip:pro
27.484	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
27.781	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
27.782	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
28.082	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
28.083	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
28.382	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
28.383	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
28.683	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
28.684	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
28.984	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
28.985	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
29.285	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
29.286	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
29.587	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
29.588	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
29.891	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
29.895	(5060)	Status: 404 Not Fou	(5060)	SIP: Status: 404 Not Found
30.200	(5060)	Unknown request: aa	(5060)	SIP/SDP: Unknown request: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

FUZZING Test Results – Life Sample

```
SIP Request:

aaaaaaaa sip:protos@192.168.1.80 SIP/2.0
Via: SIP/2.0/UDP rx-desktop:5060;branch=z9hG4bK00003000003
From: 3 <sip:user@rx-desktop>;tag=3
To: Receiver <sip:protos@192.168.1.80>
Call-ID: 3@rx-desktop
CSeq: 1 INVITE
Contact: 3 <sip:user@rx-desktop>
Expires: 1200
Max-Forwards: 70
Content-Type: application/sdp
Content-Length: 120

v=0
o=3 3 3 IN IP4 rx-desktop
s=Session SDP
c=IN IP4 127.0.1.1
t=0 0
m=audio 9876 RTP/AVP 0
a=rtpmap:0 PCMU/8000

SIP Response:

SIP/2.0 404 Not Found
Via: SIP/2.0/UDP rx-desktop:5060;branch=z9hG4bK00003000003;received=192.168.1.235
From: 3 <sip:user@rx-desktop>;tag=3
To: Receiver <sip:protos@192.168.1.80>;tag=fa997f8144037lde7lab448ebdb9af56-6192
Call-ID: 3@rx-desktop
CSeq: 1 INVITE
Server: OpenSER (1.3.2-notls (i386/linux))
Content-Length: 0
```

PROTOS test trace for OpenSER Overflow - general, 'a' (0x61) character overflows up to 128k

FUZZING Test Results – Life Sample

testing_Protos wagdy - Wireshark

File Edit View Go Capture Analyze Statistics Telephony Tools Help

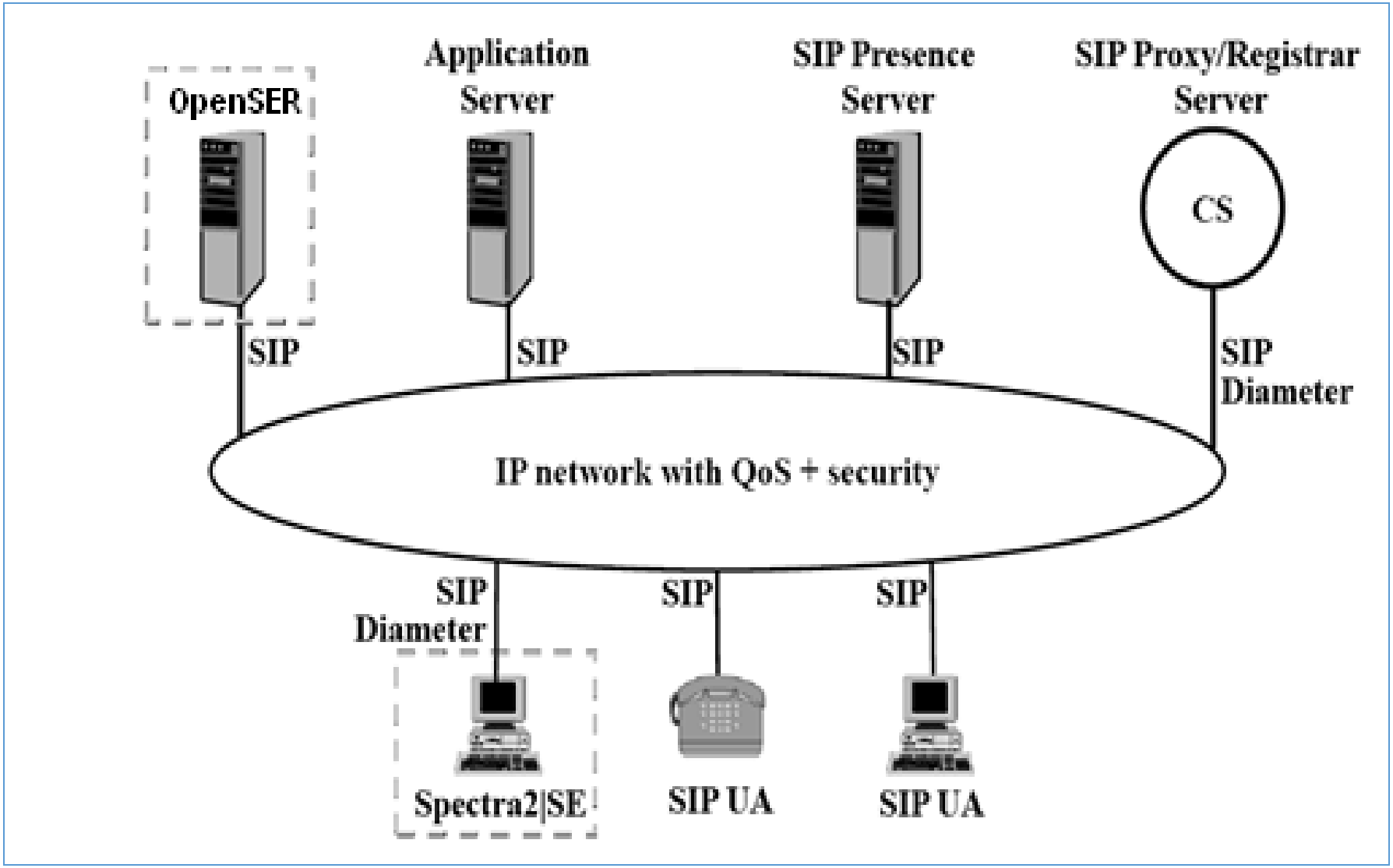
Filter: sip Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
649	50.302799	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
650	50.608059	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
651	50.904447	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: % sip:protos@192.168.1.80, with session descrip
652	50.905558	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
653	51.205094	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %s%x%n sip:protos@192.168.1.80, with session de
654	51.206149	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
655	51.505697	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %.127d sip:protos@192.168.1.80, with session de
656	51.506470	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
657	51.806293	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %.555d sip:protos@192.168.1.80, with session de
658	51.807160	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
659	52.106886	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %.999d sip:protos@192.168.1.80, with session de
660	52.107798	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
661	52.407513	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %.1270d sip:protos@192.168.1.80, with session d
662	52.408452	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
663	52.708125	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %.4097d sip:protos@192.168.1.80, with session d
664	52.709008	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
665	53.008754	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %.9999d sip:protos@192.168.1.80, with session d
666	53.009655	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
667	53.309395	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %.12700d sip:protos@192.168.1.80, with session
668	53.310423	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
669	53.610018	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %.127000d sip:protos@192.168.1.80, with session
670	53.610792	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
671	53.910645	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %n%n%n%n%n%n%n%n%n%n%n%x%x%x%x%x%x%x%x%x%x%x%x%
672	53.911550	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
673	54.211386	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%
674	54.212405	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found
678	54.513096	192.168.1.235	192.168.1.80	SIP/SDP	Unknown request: %n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%n%
679	54.514573	192.168.1.80	192.168.1.235	SIP	Status: 404 Not Found

Testing with Spectra2|SE

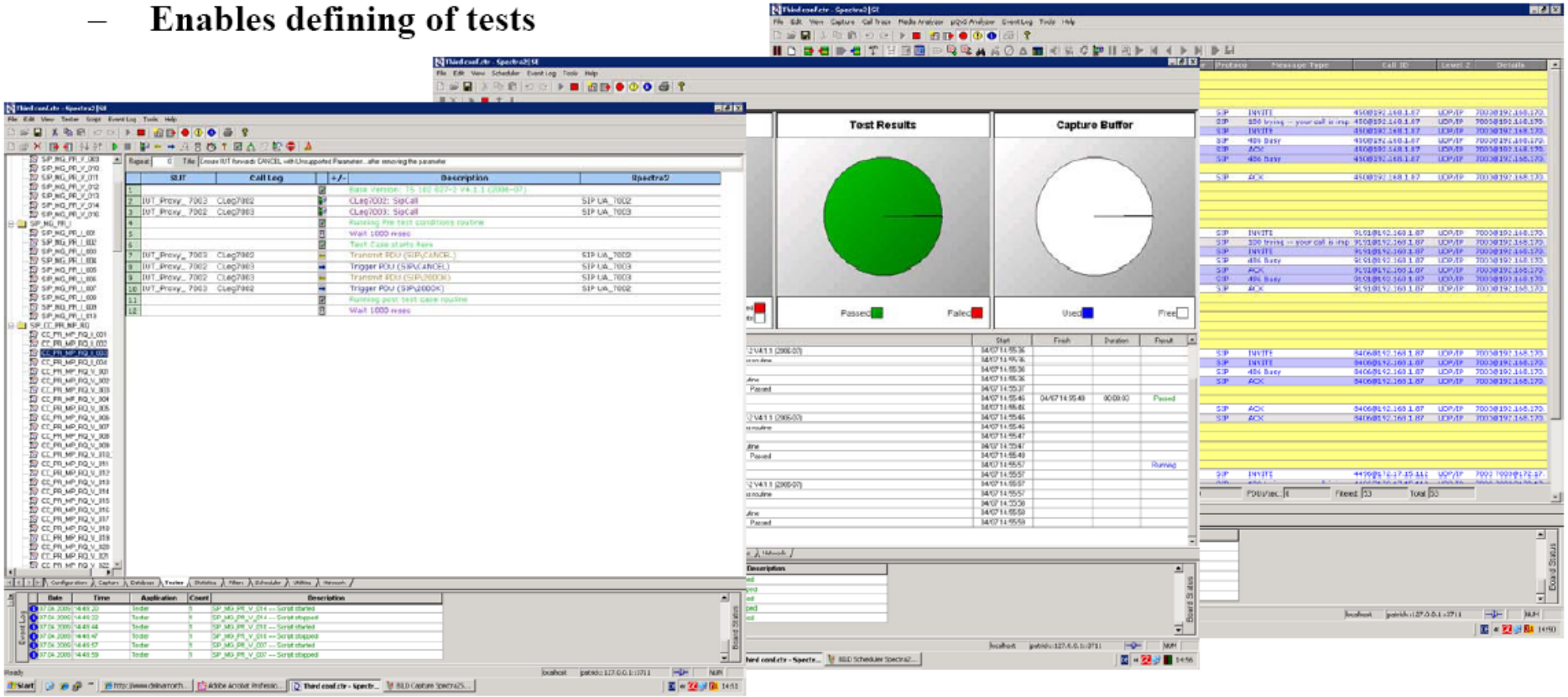
- The Spectra 2|SE software testing solution is a Windows PC based software application that provides users with the capability to test IMS and VoIP networks from their laptops or desktops.
- Spectra 2|SE is an industry leading easy to learn and easy to use scripting interface.Spectra2|SE delivers pre installed possibilities and Enables defining of tests.
- **Testing with the Spectra2|SE standard comply using ETSI TS 102 027-2 V6.1.1 or RFC 3261.**
- This test was experimentally studied with the OpenSER running on Intel Pentium 4 CPU with 3.20 GHz and 1 GB RAM.
- The operating system is a Debian/Linux with Kernel version 2.6.18-6-686.
- The Spectra 2|SE PC and the OpenSER PC are connected on a 100 Mb/s Ethernet LAN network.
- The test suite tests SIP via UDP only.

Testing with Spectra2|SE



Testing withSpectra2|SE

- Testing with the Spectra2|SE standard comply using e.g. ETSI TS 102 027-2 V4.1.1 (2006-07) or RFC 4475
- Spectra2|SE is a industry leading easy-to-learn, easy-to-use scripting interface
- Spectra2|SE delivers pre installed test suites including adequate monitoring possibilities
- Enables defining of tests



Testing withSpectra2|SE

Third conf.ctr - Spectra2|SE

File Edit View Scheduler Event Log Tools Help

Scheduler
- Batch Folders
  - SIP Proxy Total
    - Test Suite SIP Proxy
- Test Suites
  - Test Suite SIP Proxy T...
    - SIP_MG_PR_V_001
    - SIP_MG_PR_V_002
    - SIP_MG_PR_V_003
    - SIP_MG_PR_V_004
    - SIP_MG_PR_V_005
    - SIP_MG_PR_V_006
    - SIP_MG_PR_V_007
    - SIP_MG_PR_V_008
    - SIP_MG_PR_V_009
    - SIP_MG_PR_V_010
    - SIP_MG_PR_V_011
    - SIP_MG_PR_V_012
    - SIP_MG_PR_V_013
    - SIP_MG_PR_V_014
    - SIP_MG_PR_V_016
    - SIP_MG_PR_I_001
    - SIP_MG_PR_I_002
    - SIP_MG_PR_I_003
    - SIP_MG_PR_I_004
    - SIP_MG_PR_I_005
    - SIP_MG_PR_I_006
    - SIP_MG_PR_I_007
    - SIP_MG_PR_I_008
    - SIP_MG_PR_I_009
    - SIP_MG_PR_I_010
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_C
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...
    - CC_PR_MP_RQ_V_...

Test Progress

Passed, Failed, Cancelled, Incomplete

Test Results

Passed, Failed

Capture Buffer

Used, Free

	Start	Finish	Duration	Result
CC_PR_TR_SE_TI_008	08/11 11:14:08	08/11 11:14:11	00:00:03	Failed
CC_PR_TR_SE_TI_008: -- Test Started	08/11 11:14:08			
CC_PR_TR_SE_TI_008: Base Version: TS 102 027-2 V4.1.1 (2006-07)	08/11 11:14:08			
CC_PR_TR_SE_TI_008: Running Pre test conditions routine	08/11 11:14:08			
CC_PR_TR_SE_TI_008: Test Case starts here	08/11 11:14:09			
CC_PR_TR_SE_TI_008: IUT Invite Proceeding State	08/11 11:14:09			
CC_PR_TR_SE_TI_008: Trigger failed at line 15	08/11 11:14:10			
CC_PR_TR_SE_TI_008: -- Test Finished : Result = Failed	08/11 11:14:10			
CC_PR_TR_SE_TI_009	08/11 11:14:16	08/11 11:14:50	00:00:34	Failed
CC_PR_TR_SE_TI_009: -- Test Started	08/11 11:14:16			
CC_PR_TR_SE_TI_009: Base Version: TS 102 027-2 V4.1.1 (2006-07)	08/11 11:14:16			
CC_PR_TR_SE_TI_009: Running Pre test conditions routine	08/11 11:14:16			
CC_PR_TR_SE_TI_009: Test Case starts here	08/11 11:14:17			
CC_PR_TR_SE_TI_009: IUT Invite Proceeding State	08/11 11:14:17			
CC_PR_TR_SE_TI_009: Call UP	08/11 11:14:17			
CC_PR_TR_SE_TI_009: Trigger failed at line 25	08/11 11:14:49			
CC_PR_TR_SE_TI_009: -- Test Finished : Result = Failed	08/11 11:14:49			
Capture Saved to C:\SIP Proxy Total_08112009 111455.cap	08/11 11:14:55			
Scheduler Stopped	08/11 11:14:56			

Configuration | Capture | Database | Tester | Statistics | Filters | Scheduler | Utilities | Network

Event Log

Date	Time	Application	Count	Description
11.08.2009	11:14:02	Tester	1	CC_PR_TR_SE_TI_007 --- Script stopped
11.08.2009	11:14:08	Tester	1	CC_PR_TR_SE_TI_008 --- Script started
11.08.2009	11:14:10	Tester	1	CC_PR_TR_SE_TI_008 --- Script stopped
11.08.2009	11:14:16	Tester	1	CC_PR_TR_SE_TI_009 --- Script started
11.08.2009	11:14:49	Tester	1	CC_PR_TR_SE_TI_009 --- Script stopped
11.08.2009	11:14:55	Capture	1	Capture file C:\SIP Proxy Total_08112009 111455.cap saved

Board Status

Ready | localhost | patrick::127.0.0.1::1140 | NUM

Start | Adobe Acrobat Professio... | Third conf.ctr - Spectr... | (Untitled) - Wireshark | 11:16

Spectra2|SE Results – Life Sample

TPId: IP_CC_PR_MP_RQ_V_032
Status: Mandatory
Ref: RFC 3261 [2] sections 16.3, item 3 and 16.10.
Purpose: Ensure that the IUT on receipt of a CANCEL request that does not correspond to an existing context including a Max-Forwards header set to 0, sends a Too many hops (483 Too many hops) request failure response.

```
SIP Request: CANCEL
sip:7003@192.168.1.80 SIP/2.0
Via: SIP/2.0/UDP192.168.1.87:5060;branch=z9hG4bK3776328-bdcc3b69
Max-Forwards: 0
From: sip:7002@192.168.1.87
To: sip:7003@192.168.1.80
Call-ID: 22969@192.168.1.87
CSeq: 0 CANCEL
Content-Length: 0

SIP Response:
SIP/2.0 483 Too Many Hops
Via:SIP/2.0/UDP192.168.1.87:5060;branch=z9hG4bK3776328-bdcc3b69
From: sip:7002@192.168.1.87
To:sip:7003@192.168.1.80;tag=e42c76af37d5792c84fff6077ad77fa8.7a49
Call-ID: 22969@192.168.1.87
CSeq: 0 CANCEL
Server:OpenSER (1.3.2-notls (i386/linux))
Content-Length: 0
```

VoIP Signaling Protocols (MGCP,MEGACO,H.248)

What is MGCP ?

Media Gateway Control Protocol .

A protocol for controlling telephony gateways from external call control elements called media gateway controllers or call agents.

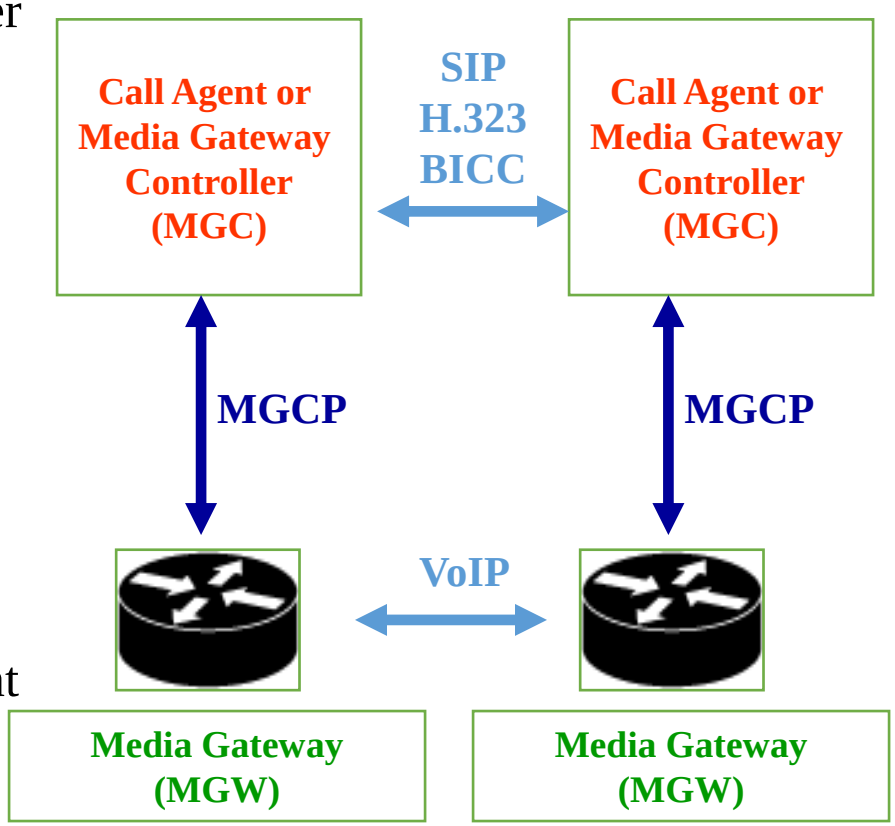
IETF RFC 2705 Media Gateway Control Protocol

Characteristics of MGCP

- A master/slave protocol.
- Assumes limited intelligence at the edge (endpoints) and intelligence at the core (call agent).
- Used between call agents and media gateways.
- Differs from SIP and H.323 which are peer-to-peer protocols.
- Interoperates with SIP and H.323.

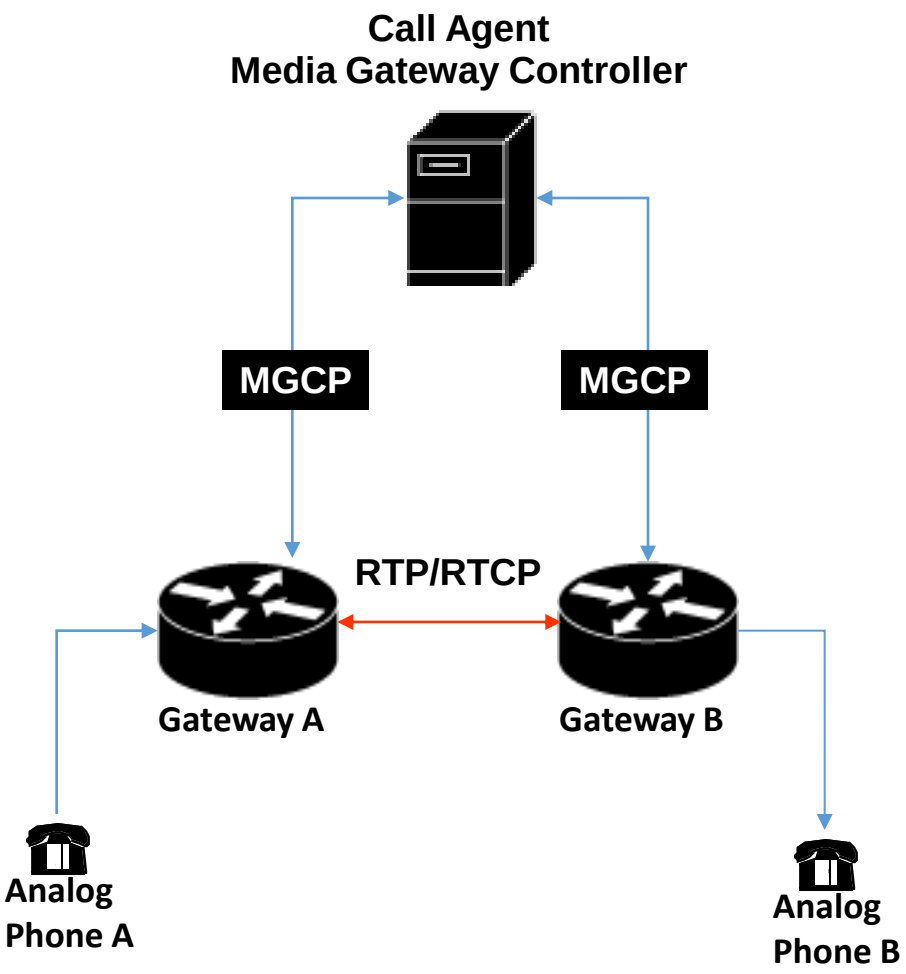
MGCP Components

- Call agent or media gateway controller
  - Provides call signaling, control and processing intelligence to the gateway.
  - Sends and receives commands to/from the gateway.
- Gateway
  - Provides translations between circuit switched networks and packet switched networks.
  - Sends notification to the call agent about endpoint events.
  - Execute commands from the call agents.



Simplified Call Flow

- When Phone A goes offhook Gateway A sends a signal to the call agent.
- Gateway A generates dial tone and collects the dialed digits.
- The digits are forwarded to the call agent.
- The call agent determines how to route the call.
- The call agent sends commands to Gateway B.
- Gateway B rings phone B.
- The call agent sends commands to both gateways to establish RTP/RTCP sessions.



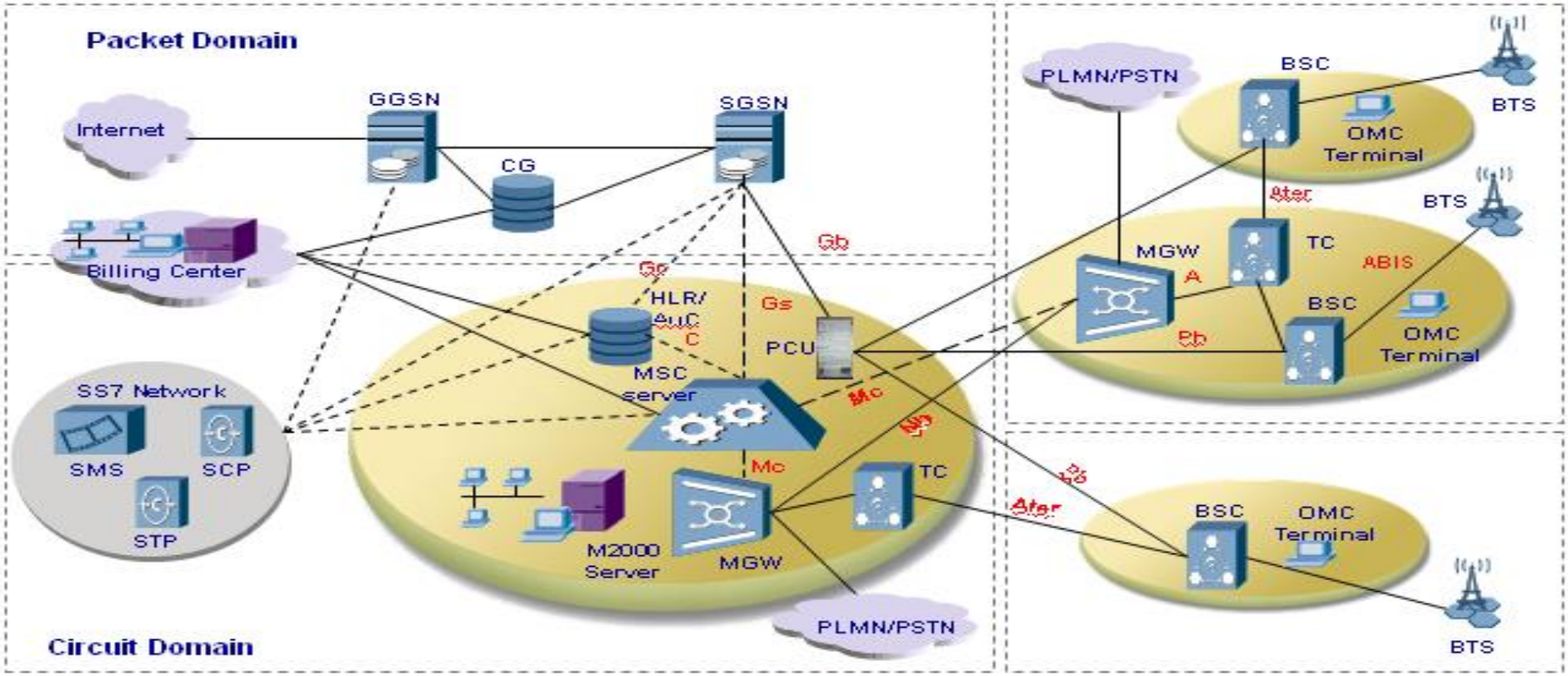
What is MEGACO ?

A protocol that is evolving from MGCP and developed jointly by ITU and IETF:

Megaco - IETF.

H.248 - ITU.

MEGACO Network



VoIP Signaling Protocols (BICC)

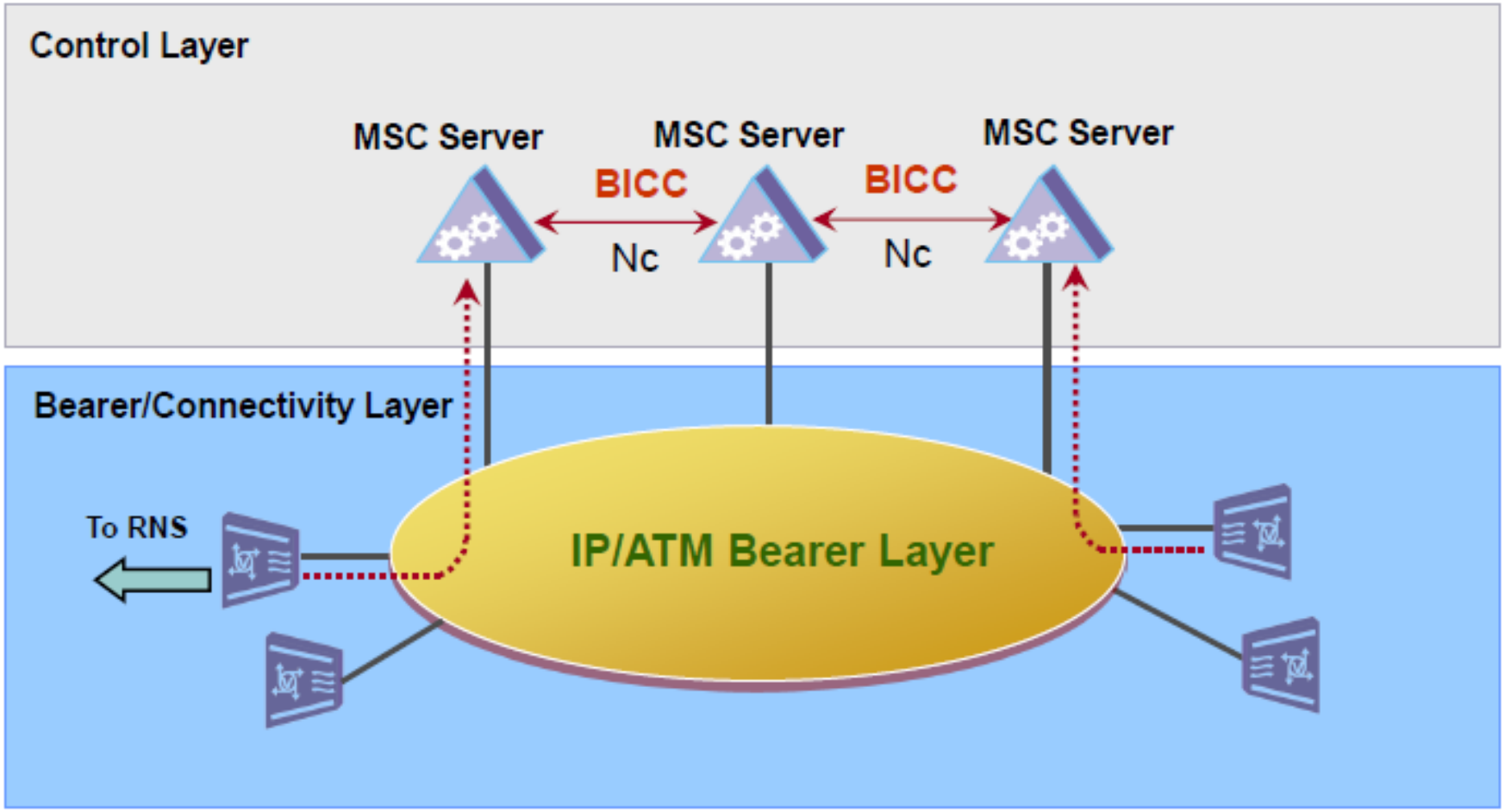
Content

1. Basic Knowledge of BICC
2. Application Transport Mechanism Principle
3. Main Call Control Flow Introduction

Content

BICC in Soft-Switch Core Network

BICC in Soft-switch Core Network



Introduction to BICC Protocol

Introduction of BICC Protocol

- BICC protocol is the Bearer Independent Call Control (BICC) protocol used in backbone networks, which include various data networks (ATM or IP network), BICC protocol can implement full PLMN/PSTN/ISDN services.
- BICC is a protocol which is characterized by the separation between call control and bearer. It does not control media resource directly， but control these resources by standard bearer control protocol(H.248 protocol)

Introduction to BICC Protocol

Introduction of BICC Protocol

- The BICC protocol is an adaptation of the ISUP protocol definition, but it is not peer-to-peer compatible with ISUP (see ITU-T Q.1912.1).
- BICC protocol can be transported by the transport layer protocols of MTP3/MPT3B/M3UA/SCTP.

Introduction to BICC Protocol

Introduction of BICC Protocol

- BICC protocol is defined by ITU-T :
  - Q.1902 serials: defining BICC mechanism , message, parameters and flow.
  - Q.2150: definition of STC, BICC protocol use the Signaling Transport Converter layer to transport signaling message, so the BICC protocol is independent of under layer transport protocol.
  - Q.765: definition of BICC Application Transport Mechanism

Main Services Supported by BICC

Main Services Supported by BICC

- **Main function and service supported by BICC includes:**
  - **Basic service:**
    - Speech / 3.1kHz audio：audio service
    - Fax：
    - 64kb/s unrestricted：unstructured data service
    - Tones and announcements：
  - **Supplementary service (for example CFW)**
  - **Additional function (for example Number Portability)**
- **Functions and services supported by BICC protocol are plentiful. Please refer Q.1902-1 for detail.**

BICC Protocol Versions

BICC Protocol Version: CS1 and CS2

- CS：Capability Set，is release version of BICC protocol.
- BICC CS1：
  - 2000 ITU-T Q.1901series definition. It is characterized by the non-separation between the call control physical entities and the bearer physical entities, which uses ATM bearer for supporting N-ISDN.
- BICC CS2：
  - 2000 /2001 ITU-T Q.1902 series definition. It is characterized by the separation between the call control physical entities and the bearer physical entities, It supports IP bearer and ATM bearer, and support tunneling transport mechanism.

Differences Between BICC and ISUP

Differences Between BICC and ISUP

- The differences between BICC and ISUP are:
  - New message in BICC : Application Transport Message (APM)
  - BICC no longer uses BLO message and BLA message which are used in ISUP.
  - CIC concept of BICC is different from ISUP.

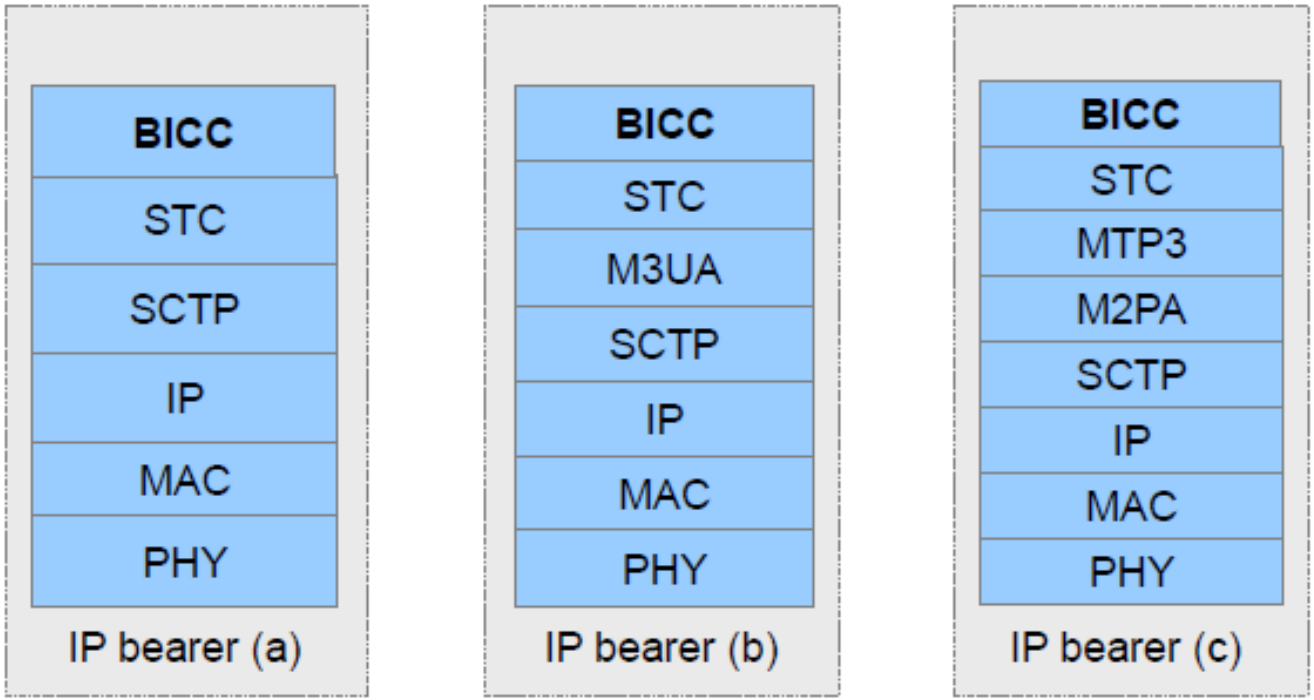
Content

BICC Protocol Stacks

STC: Signaling Transport Converter used by Bearer Independent Call Control (BICC)

BICC Protocol Stacks: IP Based

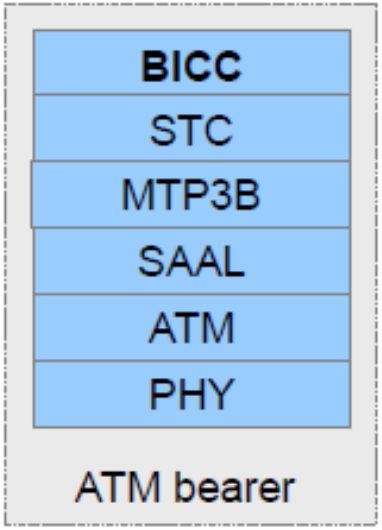
- **Different protocol stacks of IP bearer**

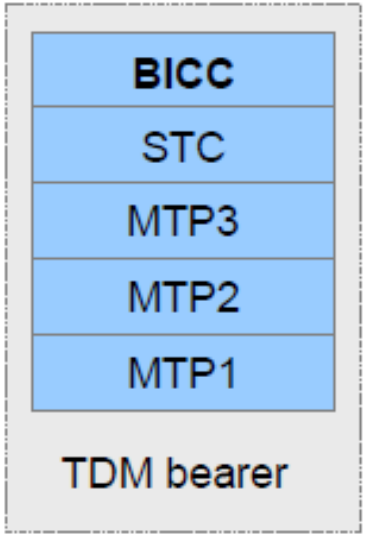


BICC Protocol Stacks

STC: Signaling Transport Converter used by Bearer Independent Call Control (BICC)

BICC Protocol Stacks: ATM and TDM Based





Content

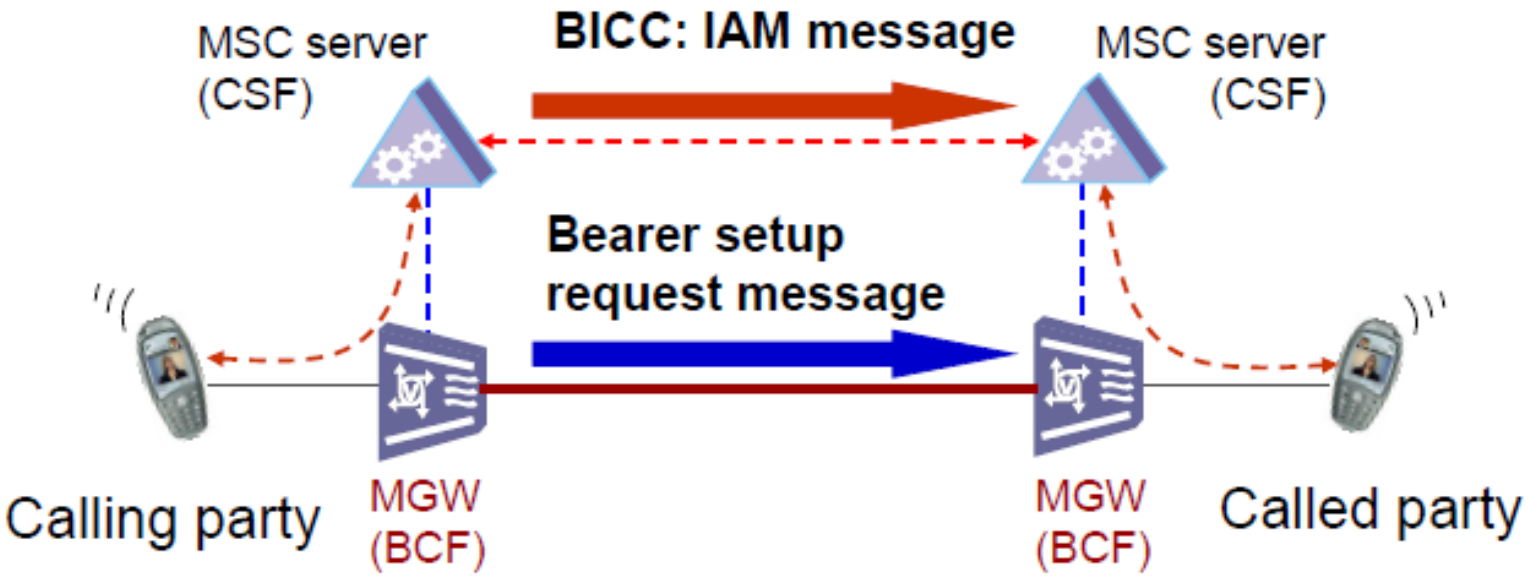
New Concepts of BICC

New Concepts of BICC

- The following is the main and new concepts of BICC :
  - Call Instance Code (CIC): to identify signaling relation between peer BICC entities.
  - Bearer establish direction: forward or backward.
  - Codec negotiation: negotiate Codec between network entities through IAM, APM messages.
  - BICC tunneling: support IP bearer in CS2. IP bearer control protocol is transported by BICC tunneling.

Forward Bearer Setup

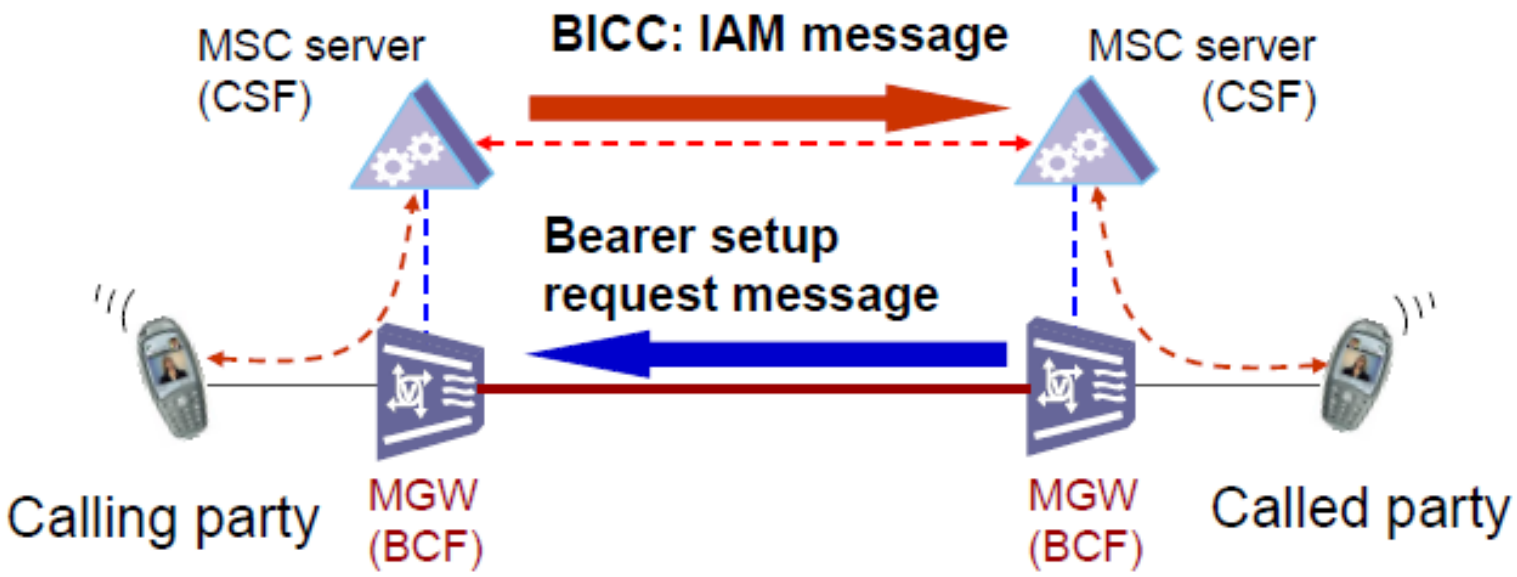
Forward Bearer Setup



- **The direction of bearer setup request message(ALCAP or IPBCP message) is the same as IAM message.**

Backward Bearer Setup

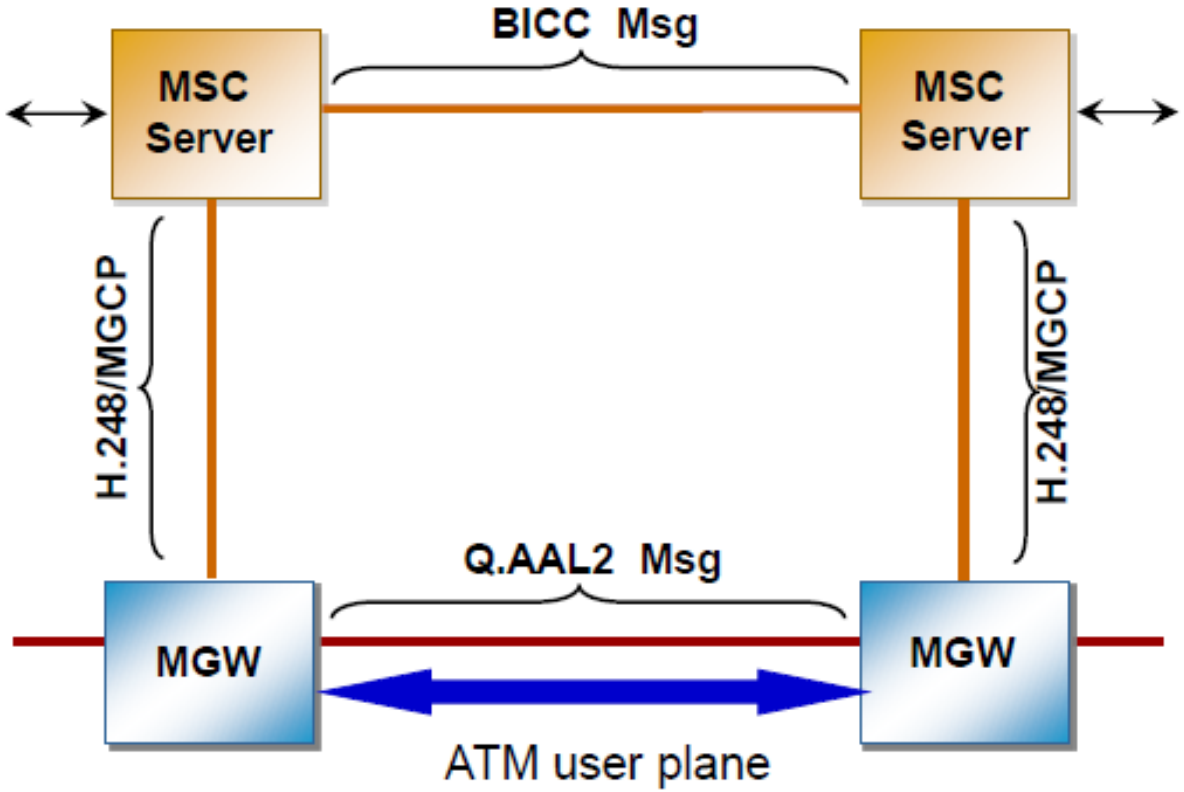
Backward Bearer Setup



- **The direction of bearer setup request message(ALCAP or IPBCP message) is opposite to IAM message.**

ATM Bearer Mode

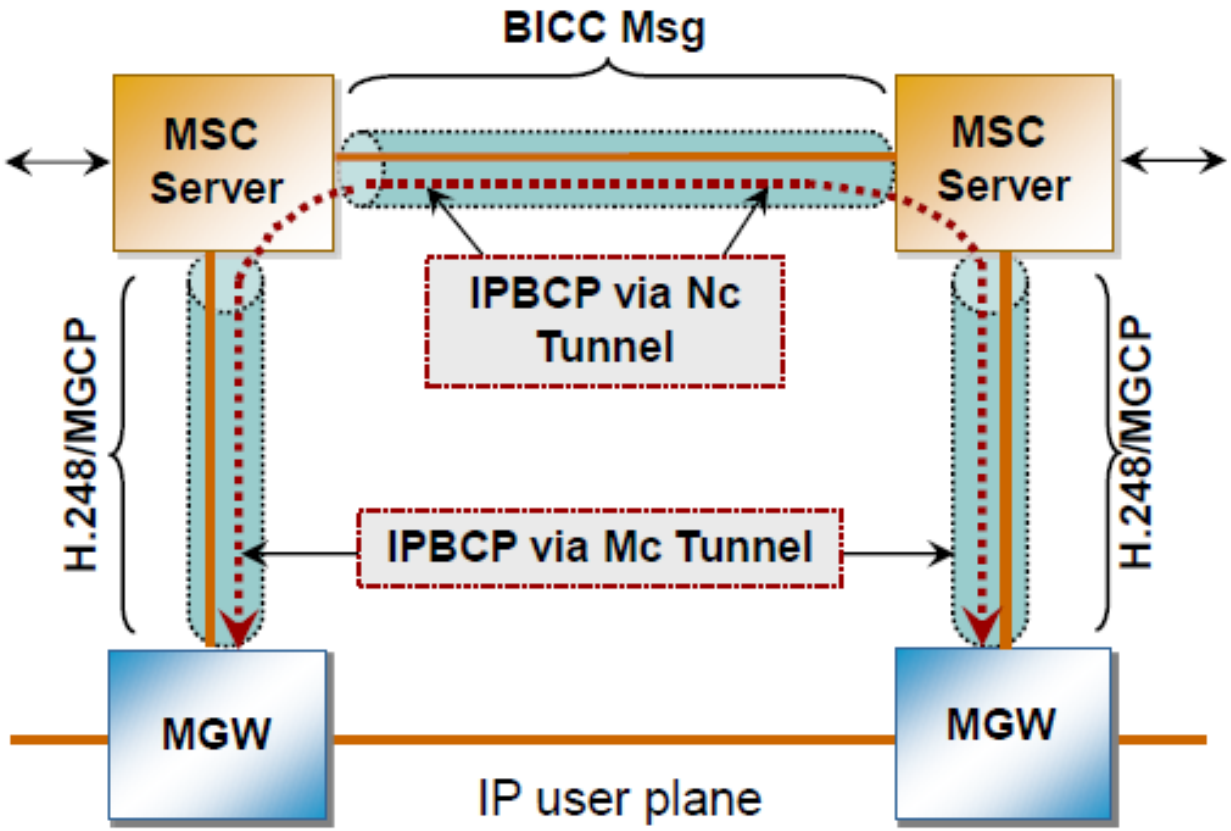
ATM Bearer Mode: No Tunnel



- ATM bearer setup messages(ALCAP/Q.AAL2/Q.2630) are transported directly between MGWs.

IP Bearer Mode

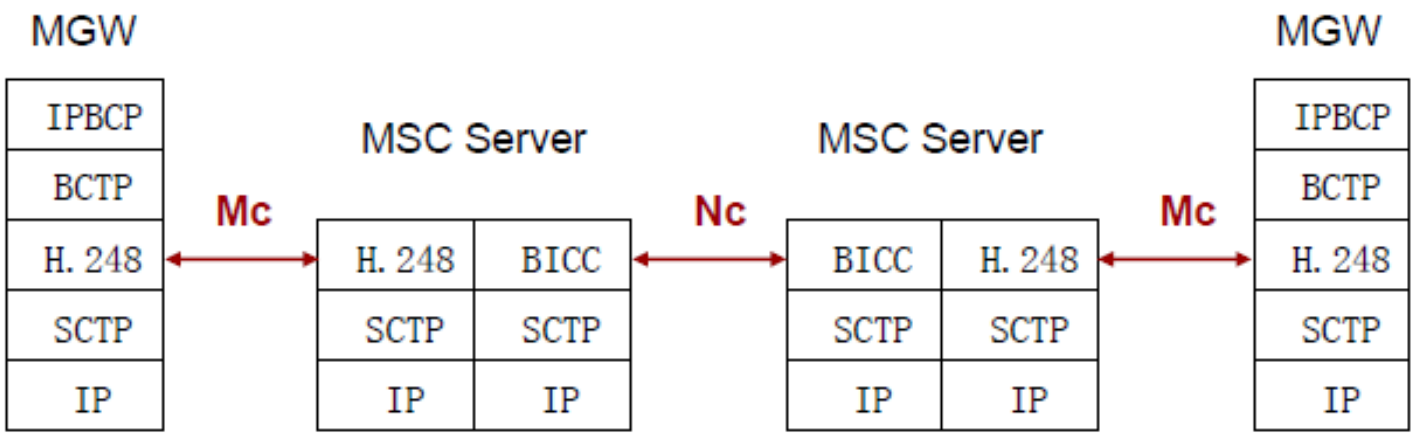
IP Bearer Mode: Using Tunnel



- IP bearer setup messages(IPBCP/Q.1970) are transported between the MGWs via the Mc and Nc interface tunnel.

The Tunneling Transport Mode of IPBCP

The Tunneling Transport Mode of IPBCP

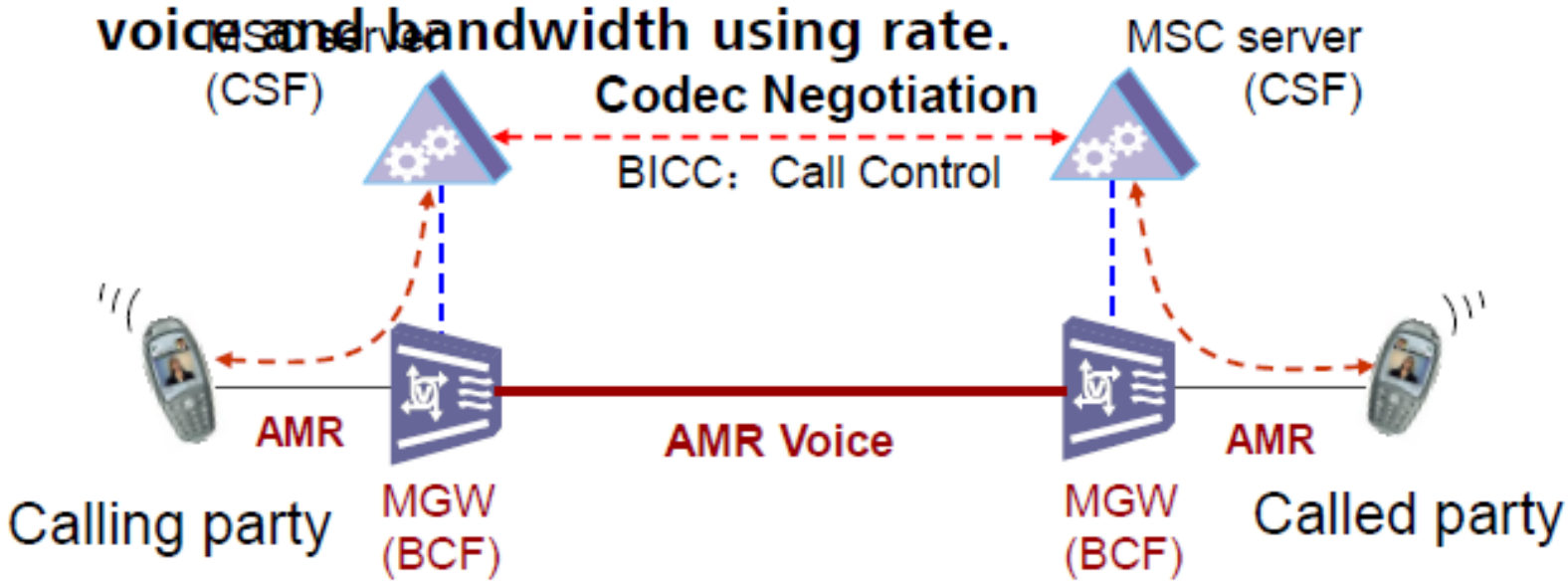


- IPBCP is encapsulated in BCTP and transferred through Mc interface and Nc interface.
- MSC Server transparent transfer the IPBCP+BCTP message.
- MGW originate and terminate IPBCP package.

Codec Negotiation Function

Codec Negotiation Function

- By codec negotiation, the network can reduce unnecessary coding and decoding process, thus saving the TC resource and improving the QOS of voice and bandwidth using rate.



Content

BICC Message Structure

BICC Message Structure

CIC
Message type code
Mandatory fixed part
Mandatory variable part
Optional part

- CIC: call instance code. It is used to identify the inter-office calling relation belonging to a call.
- Message type code: Message type ,CS2 has 38 messages totally. Common messages include IAM、APM、ACM、ANM、REL、RLC etc.

Codec List

Codec List

- Codec List: consist of some Codec IE

8	7	6	5	4	3	2	1
Single Codec information element							
Single Codec information element							
Single Codec information element							

- Single Codec: detail Codec format ,consist of standard organization IE and detail Codec IE .

	MSB 8	7	6	5	4	3	2	1 LSB
1	Organization Identifier							
2 n	Codec Information							

Codec Information

Codec Information

- **Codec defined by ITU-T :**
  - G.711 64 kbit/s A-law
  - G.711 64 kbit/s μ-law
  - G.711 56 kbit/s A-law
  - G.711 56 kbit/s μ-law
  - G.722 (SB-ADPCM)
  - G.723.1
  - G.723.1 Annex A (silence suppression)
  - G.726 (ADPCM)
  - G.727 (Embedded ADPCM)
  - G.728
  - G.729 (CS-ACELP)
  - G.729 Annex B (silence suppression)
- **Codec extended by 3GPP: AMR and AMR2.**

Codec Information Trace

Codec Information Trace

- **AMR Codec information**

```
single codec content len
  SINGLE CODEC CONTENT
    compatibility information
    organization identifier: etsi (2)
    codec type etsi: umts adaptive multi rate2 (6)
    codec information multi rate
      stMultiRateACS
        rate475: 0x1 (1)
        rate515: 0x1 (1)
        rate590: 0x1 (1)
        rate670: 0x1 (1)
        rate740: 0x1 (1)
        rate795: 0x1 (1)
        rate102: 0x1 (1)
        rate122: 0x1 (1)
      stMultiRateSCS
        rate475: 0x1 (1)
        rate515: 0x1 (1)
        rate590: 0x1 (1)
        rate670: 0x1 (1)
        rate740: 0x1 (1)
        rate795: 0x1 (1)
        rate102: 0x1 (1)
        rate122: 0x1 (1)
      stMultiRateOmMACS
        bit3Macs: 0x0 (0)
        bit1OM: 0x1 (1)
        spare: 0x0 (0)
```

Thank You