



VoIP (Voix sur IP)

-
- Généralités
 - Un protocole particulier : SIP
 - Quelques exemples d'IPBX :
 - Asterisk
-

La voix sur IP (1)

- **Définition :**

Le principe est de faire circuler sur Internet, grâce au protocole IP, les paquets de données correspondant à des échantillons de voix numérisée.

- **Fonctionnement :**

La conversation analogique est encodée dans un format numérique, avec compression possible, et encapsulée dans des paquets IP pour être transportée sur le réseau WAN/LAN

La VoIP consiste à **intégrer la voix et les données dans un même réseau :**

➡ ce qui permet de réaliser des économies car maintenance d'un seul réseau

La voix sur IP (2)

✦ Quelques problèmes

- ✦ Un réseau unique et en cas de panne, des conséquences plus graves
- ✦ Une fiabilité des réseaux de données moins bonne que celles des réseaux voix (➔ perte des paquets)
- ✦ Présence d'une latence dans le réseau
 - 0 à 300 ms , acceptable pour la plupart des conversations
 - 300 à 700 ms, devient pratiquement une conversation half duplex
 - > 700 ms, inutilisable
- ✦ Notion de Qos (Quality of Service)
- ✦ Sécurité des conversations échangées

Terminologie (1)

✦ Terminologie

- ✦ **VoIP: Voix sur IP (Voice over IP)**
La voix circule sur le réseau IP et aussi sur le réseau RTC
- ✦ **ToIP : Téléphonie sur IP (Telephony over IP)**
La voix ne circule que sur le réseau IP
- ✦ **IP Phone** : Un téléphone se branchant sur le réseau
téléphone analogique + adaptateur téléphonique analogique = ip phone
- ✦ **Softphone** : Un logiciel faisant office de téléphone
(nécessite la configuration de la carte son → x-lite, msn, skype, ...)
- ✦ **IPBX** : Autocommutateur logiciel travaillant avec communications IP (voire des communications RTC)

Terminologie (2)

✦ Terminologie

- ✦ **PABX** : Private Automatic Branching eXchange
nom commun donné pour un autocommutateur
- ✦ **PSTN** : Public Switched Telephone Network
nom anglais du **RTC**
- ✦ **PoE** : Power Over Ethernet (norme 802.3af)
➔ permet d'amener le courant électrique via la prise ethernet
(utilisé pour les téléphones, les points d'accès, les capteurs,...)
- ✦ **FXO/FXS** : Foreign eXchange Office / Foreign eXchange Service
Fxo : prise terminal (exemple téléphone)
Fxs : prise fournissant le courant

Comment ça marche ?

✧ VoIP n'est pas un protocole

- ✧ La voix sur IP est un nom générique regroupant un ensemble de protocoles et de méthodes permettant d'encoder, de transporter et de "router" des appels audios.

✧ Plusieurs étapes

- La voix est échantillonnée et codée sur 8 bits
- Le signal est compressé (utilisation des codecs de compression)
- Mise en relation des différentes entités
couche session : H323, SIP, ...
- Chaque paquet est envoyé sur le réseau
Problème : le réseau n'utilise que très peu de contraintes temporelles
nouveaux protocoles : RTP, RTcP
- etc ...

Codec

✱ Codec :

✱ Compresseur / Décompresseur }
Codeur / Décodeur } ?

- **G. 711** (alias PCM Pulse Code Modulation)
 - > Codage de la voix sur 8 bits, 8000 fois par seconde
 - > autrement nommé : u-law (en Amérique) et a-law (reste du monde)
- **G. 723.1** : débit de 5,3kb/s ou 6,3 kb/s
 - > compatible avec le protocole H323
 - > licence protégé par brevet
- **G. 729** : -> débit 8 kb/s , codage CS-ACELP , licence protégé par brevet
- **GSM** : débit 13 kb/s
 - > moins bon que le reste, mais gratuit

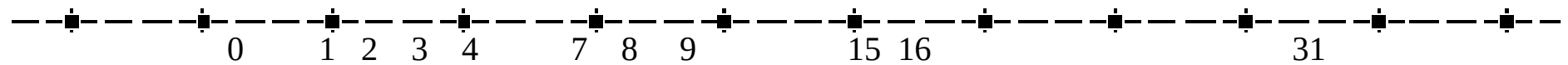
RTP

✦ RTP : Real Time Protocol

- ✦ RFC 3550
- ✦ Protocole de transfert de données en temps réel pour l'audio et la vidéo
- ✦ Au-dessus de la couche 4 (utilisation d'UDP ou de TCP)
- ✦ Permet
 - de reconstituer une base de temps
 - le séquençement des paquets
 - l'identification des contenus

RTP s'appuie sur **RTCP** (Real Time Control Protocol) pour le contrôle de flux et la QoS.

En-tête RTP (1)



V=2	P	X	CC	M	PT	N° séquence
Marqueur temporel						
Identificateur de Source de Synchronisation (SSRC)						
Identificateur de Source Contributive (CSRC)						

.....

v = version (actuellement 2)

P : padding → existence de bourrage à la fin du paquet ? 0=non

X : extension de l'entête : 0 = non, 1 = oui

CC : nombre de CSRC (généralement 0)

M : Marquage → à 1, après un silence (reprise de parole)

Pt : Payload = type de codec utilisé ex : G711 a-law =8, u-law=0, G729=18

N° séquence : permet de repérer le paquet (augmente de 1 à chaque fois)

Marqueur temporel : permet la gestion des tampons et de la gigue du réseau, augmente de 1

SSRC : Nombre aléatoire désignant la source du paquet RTP (ce nombre ne change pas)

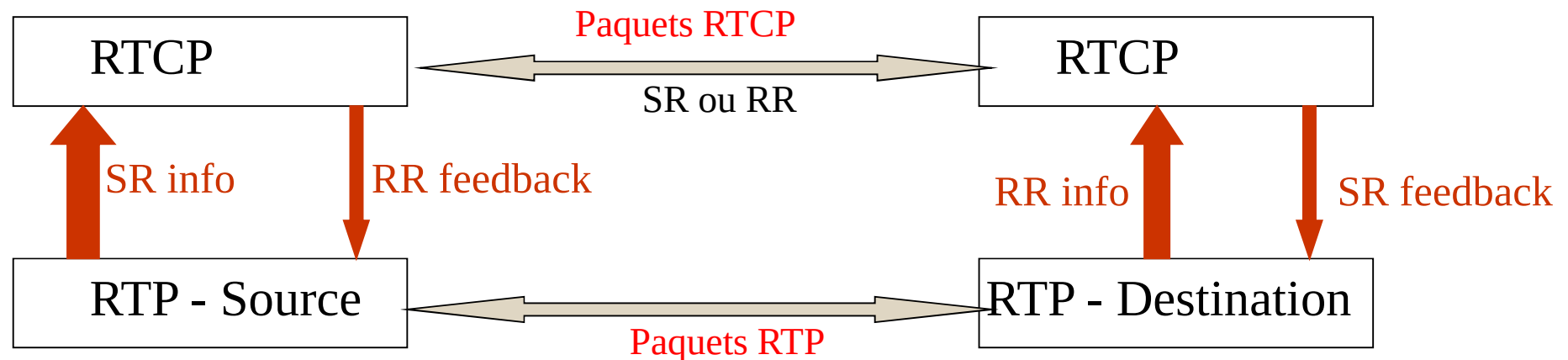
En-tête RTP (2)

✧ Calcul

- ✧ entête d'une trame RTP : $IP(20) + UDP(8) + RTP(12) = 40$ octets
- ✧ La longueur d'un paquet RTP est $\leq$ au MTU du réseau utilisé
 $MTU(IP) = 1500 \rightarrow$ taille des données = 1460 octets
- Emission **d'une vidéo** : 30 trames/s, 720*480 résolution, 3 octets/point
- total = 248 832 000 b/s
- utilisation d'un codec, compression par 100 donc 2 488 320 b/s
- nb de paquets : $2\,488\,320 / (1460 * 8) = 214$ paquets/s
soit 2 567 072 b/s avec entête
- donc débit doit être supérieur à 2,5 Mb/s

RTCP (1)

- ✧ RTCP permet d'assurer une QoS à RTP
(trame de contrôle de RTP)
- ✧ En général, RTCP utilise le port juste au-dessus de RTP
exemple : RTP : 6066 <-> 12567 RTCP: 60767 <-> 12568
- ✧ 2 sortes de paquets :
 - ◆ SR : Sender Report
 - ◆ RR : Receiver Report



RTCP (2)

✧ 3 sections pour une SR, 1 ou 2 sections pour un RR

✧ Section 1 (SR et RR)

- ✧ Version : 2 bits
- ✧ P : Padding sur 1 bit
- ✧ RC (5 bits) : nombre de blocs d'information dans le rapport (0, 1 ou plus)
- ✧ PT (8bits) : Payload type : 200= SR, 201 = RR, 202 = description
- ✧ longueur (16 bits) : mots de 32 bits
- ✧ SSRC (32 bits) : identificateur de la source (idem que celui au niveau RTP)

✧ Section 2 (SR)

- ✧ Horloges NTP : (64 bits) : valeur d'horloge quelconque, **mais isochrone pour la session**
- ✧ Marqueur temporel (32 bits) : identique à la source RTP
(permet d'avoir une relation temporelle entre ce marqueur temporel logique et NTP (temps physique))
- ✧ Nb de paquets envoyés depuis le début de la conversation (32 bits)
- ✧ Nb d'octets envoyés depuis le début de la conversation (32 bits)

RTCP (3)

✦ Section 3 (SR et RR) Bloc d'information (cf section 1)

- ◆ SSRC (32 bits) : identificateur de la source
- ◆ taux de perte (8 bits) : nb perdus / nb attendus
- ◆ nb cumulé de paquets perdus (24 bits)
- ◆ plus grand N° séquence reçu (32 bits)
- ◆ valeur de la gigue (32 bits)
- ◆ dernier marqueur temporel reçu (32 bits)
- ◆ délai depuis le dernier marqueur temporel reçu (32 bits)

Plusieurs sections 3 possibles

Pour gagner du temps, une trame RTCP peut encapsuler plusieurs paquets RTCP (exemple un SR et un description)

SIP (1)

✦ SIP (Session Initiation Protocol)

- ✦ Concurrent direct de H 323, mais au départ plus simple
- ✦ créé par l'IETF, RFC 3261
 - mais des extensions -> RFC 3262 à 3265
 - d'autres protocoles :
 - ♦ SDP Session Description Protocol (RFC 2327)
 - ♦ RSVP : utilisé pour la qualité de service
 - ♦ RTP/RTCP : pour faire transiter les données

Attention, SIP ne prend pas en charge le transfert des données

- ✦ Protocole tout IP
- ✦ Requête/ réponse très proche de HTTP ou SMTP
- ✦ Développé pour prendre en charge l'ouverture d'une session, puis de fournir une description du type de session demandée

Les entités SIP (1)

✦ User Agent

Equipement terminal pour recevoir ou émettre

- ✦ UAC : User Agent Client
-> initie les requêtes SIP
- ✦ UAS : User Agent Server
-> réponse à la requête

SIP était à l'origine un protocole orienté point à point

→ Pour contacter UAS, besoin de connaître son @IP
impossible à généraliser

✦ Registrar ou Registration Server

Enregistre dans une base de données les correspondances entre les adresses SIP (URI SIP **sip:utilisateur@domaine.com**) et IP

- ✦ Un UAC s'enregistre toujours dans un registrar

Les entités SIP (2)

✦ Proxy SIP

- ✦ Prend en charge le routage des appels
- ✦ Permet de trouver l'adresse du destinataire (peut utiliser des requêtes DNS)
- ✦ Registrar et Proxy SIP sont en général très liés

✦ Redirect Server

- ✦ permet de rediriger une adresse SIP vers une autre adresse SIP

✦ Location Server

- ✦ interroge les registars pour trouver l'adresse IP

Les messages SIP

✱ SIP est un protocole client/serveur (comme http)

- ✱ le client envoie une requête et attend une réponse
- ✱ le chemin parcouru par les paquets est arbitraire (sur UDP)
- ✱ les requêtes et les réponses ont la même structure :
 - une ligne indiquant le statut
 - l'entête du message
 - une ligne vide
 - corps du message décrit en SDP

Request-Line: **INVITE** sip:toto@172.16.65.212:58816;rinstance=757a49061e5673ed SIP/2.0

Message Header

Via: SIP/2.0/UDP 172.16.65.166:5060;branch=z9hG4bK0979a28d;rport

From: "Cisco phone 1" <sip:6000@172.16.65.166>;tag=as4e34c443

To: <sip:toto@172.16.65.212:58816;rinstance=757a49061e5673ed>

Contact: <sip:6000@172.16.65.166>

Call-ID: 625c5bca5e34fb082e6b93506ad68ab8@172.16.65.166

CSeq: 102 INVITE

Content-Type: application/sdp

Content-Length: 265

Message body

Session Description Protocol

Session Description Protocol Version (v): 0

....

SDP

SDP Exemple 1

v=0
o=- 2 2 IN IP4 172.16.66.212
s=Counter Path X-lite
t=0 0
c=IN IP4 172.16.65.212
m=audio 49170 RTP/AVP 0 3

SDP Exemple 2

v=0
o=picard 124333 67895 IN IP4
uunet.com
s=Engage!
t=0 0
c=IN IP4 101.234.2.1
m=audio 3456 RTP/AVP 0 107 109 8 98 ...

	Champ	Description	
	Version	v=0	
	Origine	o=<username> <session id> <version> <network type> <address type> <address>	
	Nom de session	s=<session name>	
	Times	t=<start time> <stop time>	
	Connexion info	c=<network type> <address type> <connection address>	
	Media	m=<media> <port> <transport> <media format list>	

code pour le G.711

Les méthodes (1)

- ✦ Les méthodes sont spécifiées dans les premiers octets de chaque requête SIP, spécifiant ainsi le contenu du message
- ✦ Il y a deux types de requêtes :
 - ✦ celle initialisant le dialogue telle que INVITE ou SUBSCRIBE
 - ✦ celle à l'intérieur d'un dialogue et pouvant modifier la session actuelle, telle que BYE ou NOTIFY
- ✦ **INVITE**
 - ✦ initialise la session, "invite" un client à parler
 - ✦ paramètres de la session se trouve dans le corps de la requête
(Media Description, Media Attribute,...)

Les méthodes (2)

✧ ACK

- ✧ confirme l'établissement d'une session (retour de INVITE)

✧ BYE

- ✧ termine une session

✧ CANCEL

- ✧ annule une transaction (exemple : INVITE)

✧ OPTIONS

- ✧ permet de récupérer des informations sur le serveur

✧ REGISTER

- ✧ permet de s'enregistrer auprès d'un registrar

✧ SUBSCRIBE

- ✧ pour s'informer sur un agent, comme dans une messagerie instantanée

✧ NOTIFY

- ✧ permet d'envoyer un événement dans la "messagerie instantanée"

✧ etc ...

Les réponses

---■---■---■---■---■---■---■---■---■---■---■---
☀ Comme pour HTTP, les réponses sont des codes sur 3 chiffres

◆ 1xx : Informational response

- 100 : trying
- 180 : ringing

◆ 2xx : succès

- 200 : OK

◆ 3xx : Redirection response

- 301 : moved permanently

◆ 4xx : erreur client

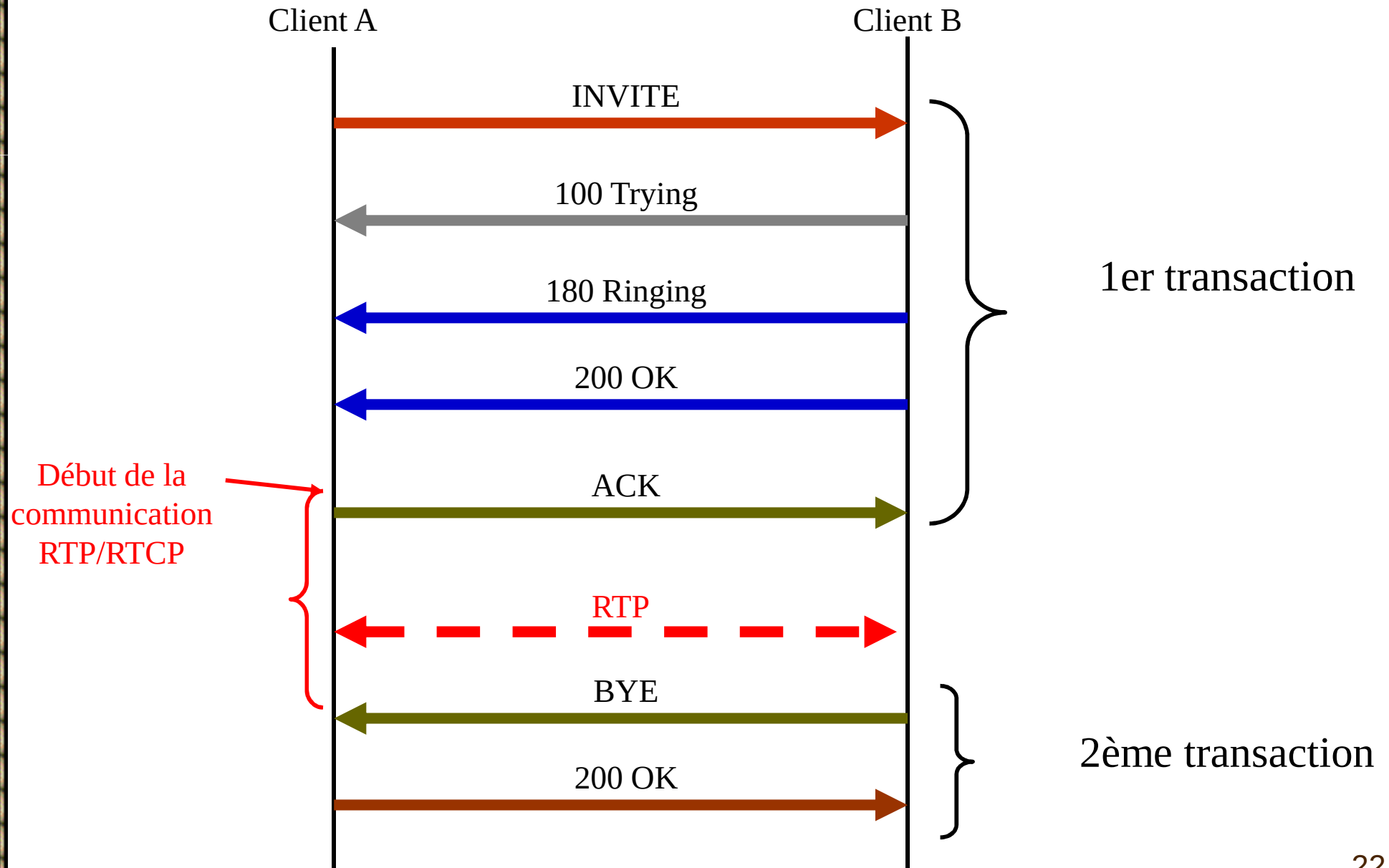
- 400 : Bad Request
- 404 : Not found
- 408 : Request Timeout

◆ 5xx : server failure

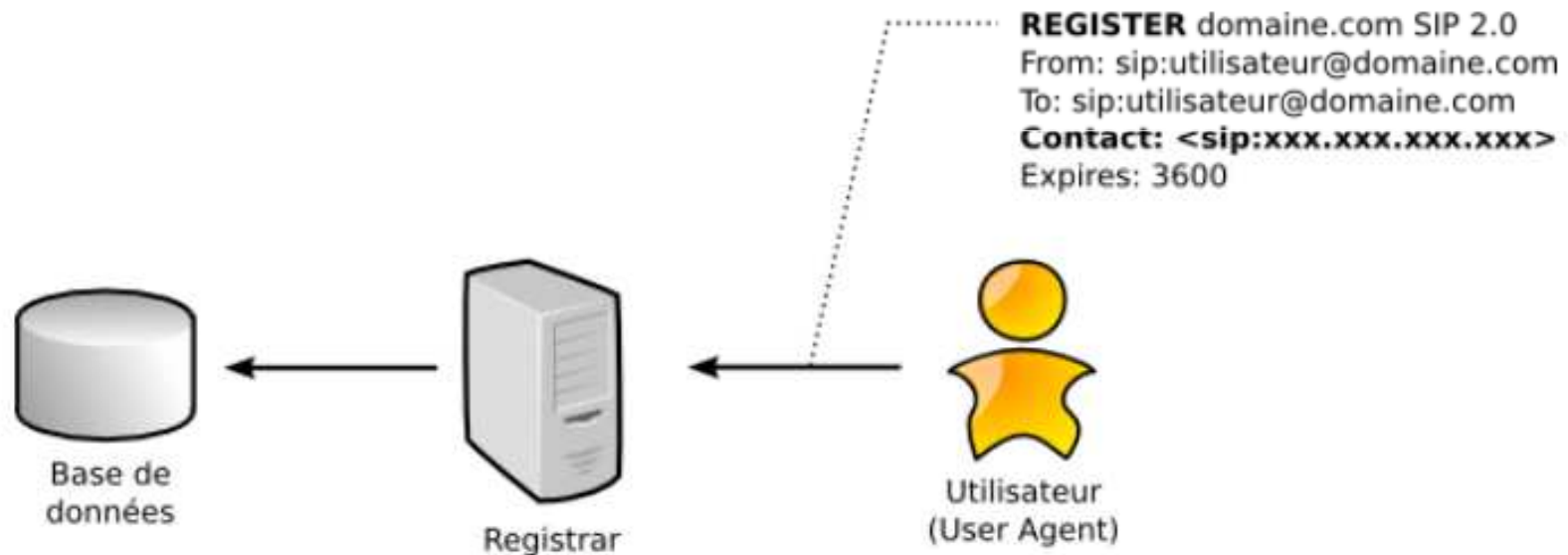
- 500 : internal error

◆ 6xx : Global failures

Exemple de dialogue (1)

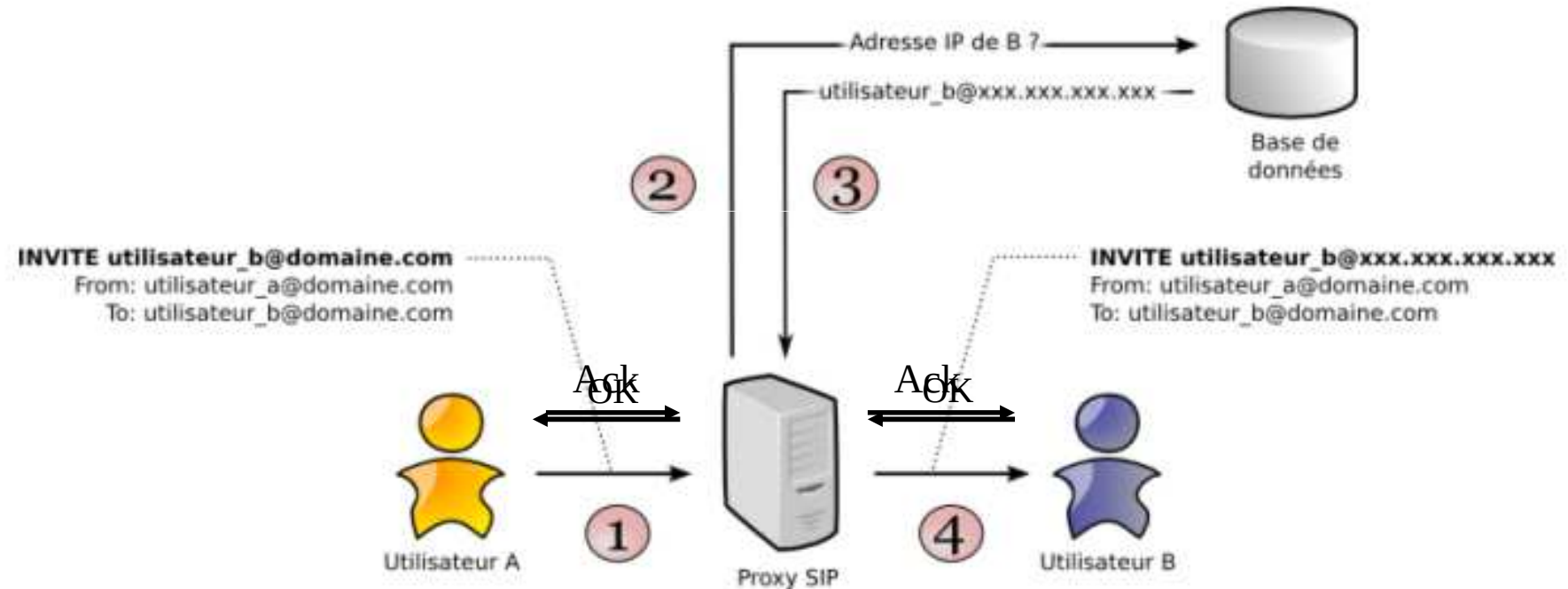


Exemple de dialogue (2)



Pour certain softphone, l'utilisation d'un registrar est obligatoire
(exemple : x-lite,...)

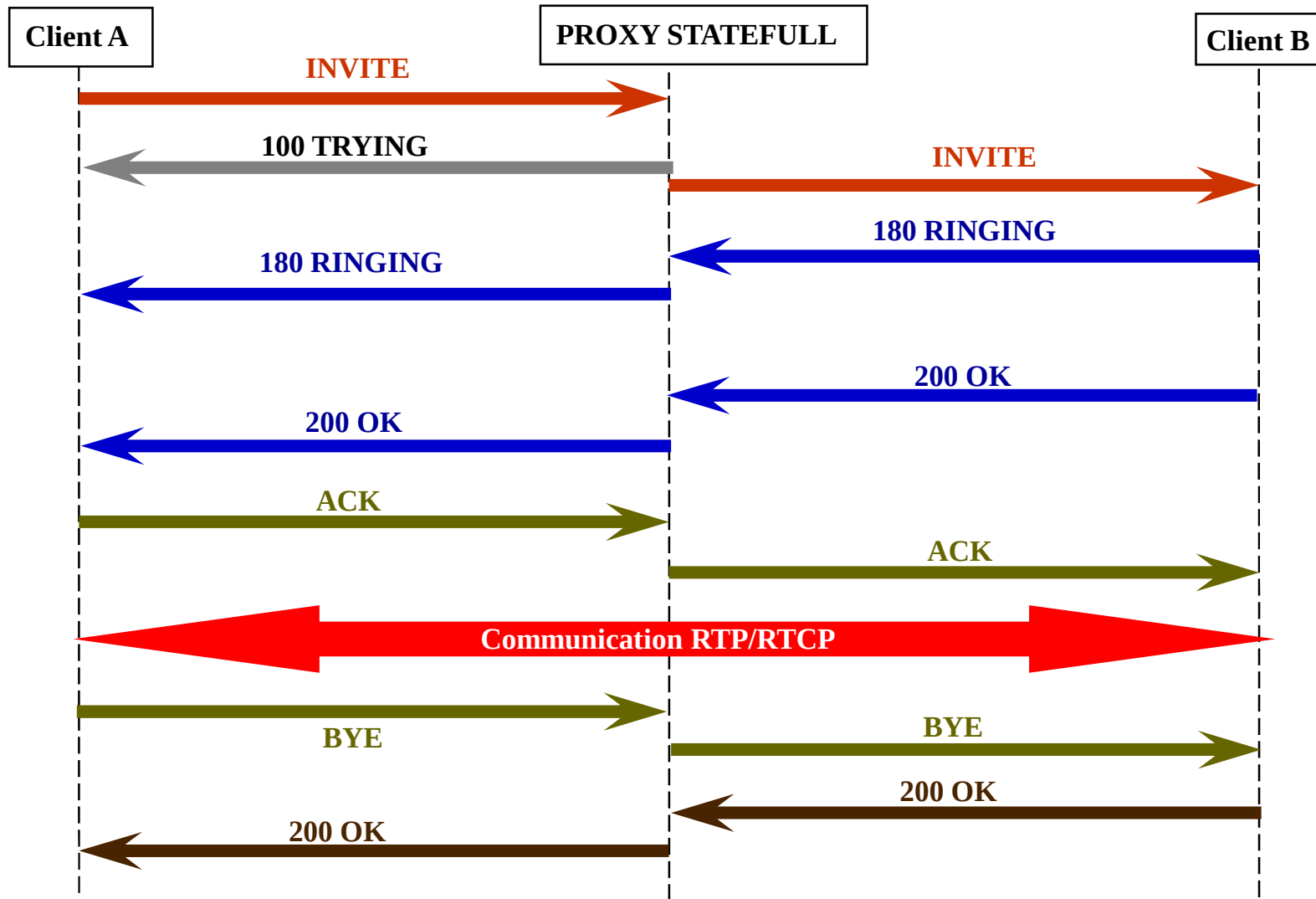
Exemple de dialogue (3)



Après ces différents échanges, mise en place de la communication en RTP

- utilisation d'un proxy sans état (stateless)
➡ communication ne passe plus par lui
- utilisation d'un proxy avec état (statefull)
➡ proxy sert de passerelle
mise en place d'autres fonctionnalités...

Exemple de dialogue (3 bis)



Problèmes SIP

✦ Pare-feu et NAT

- ✦ SIP fonctionne en UDP sur le port 5060

→ simple pour un pare-feu

MAIS session RTP sur un port quelconque....

- ✦ Problème d'adressage pour le NAT

- présence de l'adresse non routable dans SDP

➡ impossible de faire un paquet retour

•Solution :

- STUN (RFC 3489) : (Simple Traversal of UDP through NATs)
utilisation d'un serveur auxiliaire STUN
- ICE (Interactive Connectivity Establishment)
- TURN (Traversal Using Relay NAT)
- IPv6 (sans NAT)

Divers

✧ SIP et QoS

- ✧ SIP ne s'occupe pas du tout de la QoS
(SIP = protocole de signalisation)
- ✧ Rajout de mot-clé dans SIP pour intégrer la QoS
(exemple : COMEN)
- ✧ Groupe de travail sur la QoS
 - définition de : diffserv (differentiate service), RSVP de l'INTSERV...

Asterisk (1)

✧ **Asterisk** est un PABX applicatif open source permettant d'interconnecter en temps réel des réseaux de voix sur IP et des réseaux de téléphonies classiques via des cartes d'interface téléphonique.

✧ Protocoles implémentés:

- ◆ IAX™ (Inter-Asterisk Exchange)
- ◆ H.323
- ◆ SIP (Session Initiation Protocol)
- ◆ MGCP (Media Gateway Control Protocol)
- ◆ SCCP (Cisco® Skinny®)

✧ Codecs:

- ◆ ADPCM
- ◆ G.711 (A-Law & μ -Law)
- ◆ G.723.1 (pass through)
- ◆ G.726
- ◆ G.729 (through purchase of commercial license through Digium)
- ◆ GSM
- ◆ iLBC
- ◆ Linear
- ◆ LPC-10
- ◆ Speex

Asterisk (2)

✧ Interconnexion avec le RTC

- ✧ mise en place de cartes : BRI, PRI, FXO/FXS
 - cartes développés par la société DIGIUM (entre autres)
 - configuration par les fichiers : zaptel.conf , zapata.conf ou dahdi

✧ Configuration

- ✧ fichiers :
 - sip.conf
 - ✧ pour définir les utilisateurs SIP
 - extensions.conf
 - ✧ pour définir le plan de numérotation (dialplan)
 - fichiers annexes
 - ✧ voicemail.conf, sccp.conf, modules.conf ...



Assez simple à mettre en œuvre

Asterisk (3)

✱ Les différentes fonctionnalités

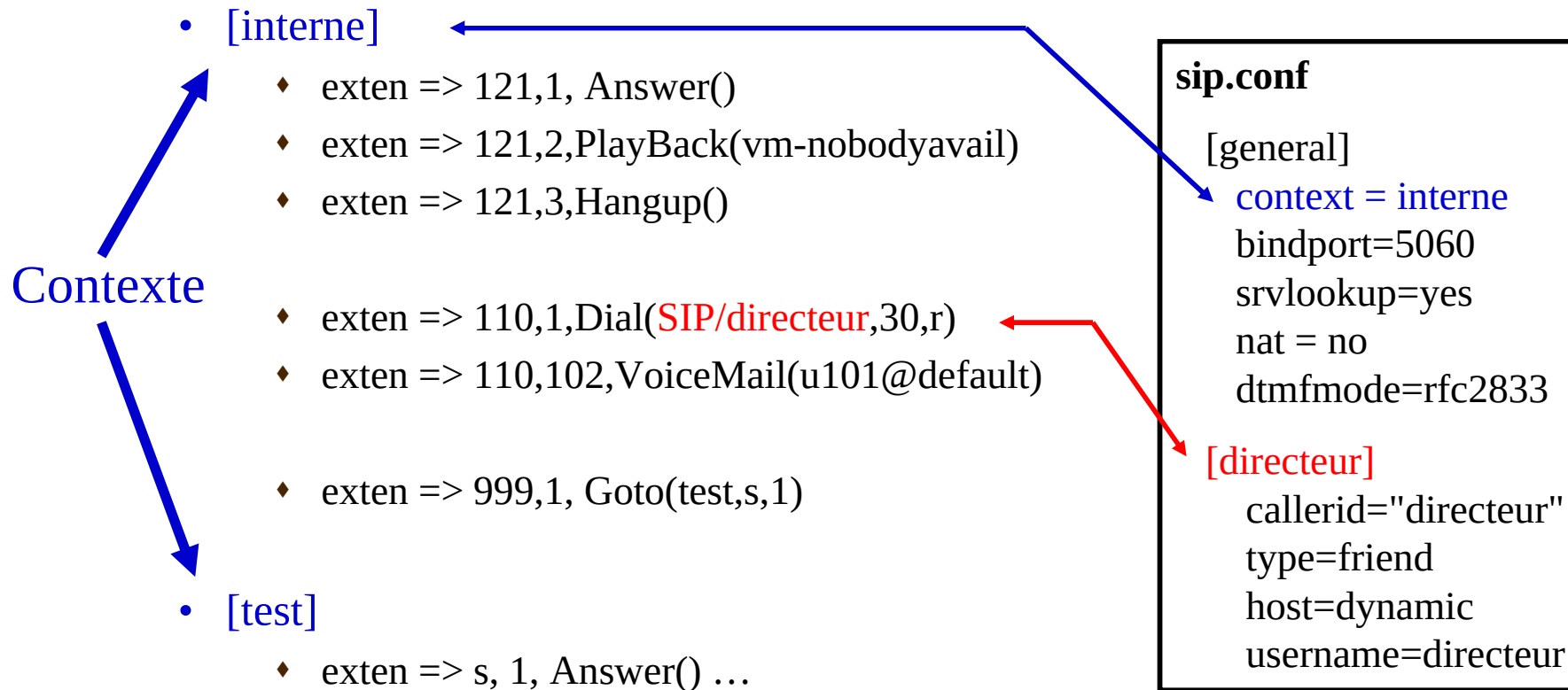
- ◆ Asterisk fonctionne principalement grâce au **plan de numérotation**
 - ◆ à configurer dans le fichier extensions.conf
 - ◆ chaque action à une priorité et utilise une application
- ◆ Quelques applications possibles
 - Answer: Répond lors d'un appel (lorsqu'un canal sonne)
 - BackGround: Joue un fichier audio tout en acceptant la composition de chiffres
 - Busy: Indique au canal de sonner occupé (normal busy)
 - Congestion: Indique l'encombrement sur le canal (fast busy)
 - Dial: Tente de connecter des canaux entre eux
 - Directory: Fournit un répertoire d'extensions à composer
 - MeetMe: Place la personne qui appelle dans une conférence en ligne
 - MP3Player: Joue un fichier MP3 au sein du flux audio
 - MusicOnHold: Joue indéfiniment la musique d'attente
 - Read : Capture les chiffres que l'appelant compose au clavier dans une variable
 - VoiceMail: Connecte la personne à la messagerie
 - VoiceMailMain: Entre dans le système de boîte vocale
 -

Asterisk (4)

✧ Les extensions (dans le fichier extensions.conf)

✧ exten => nom, priorité, application(arguments)

✧ exemple





ANNEXE

Error Codes

Note: Informational text is for the user and any number of 1xx messages can be sent e.g

100 Message received
100 Looking up number
100 Found carrier "AT&T"
100 Dialling number

Note:

Many codes are same as HTTP
SIP specific codes start x80

Informational

100 Trying
180 Ringing
181 Call Is Being Forwarded
182 Queued
183 Session Progress

Success

200 OK
202 Accepted

Redirection

300 Multiple Choices
301 Moved Permanently
302 Moved Temporarily
303 See Other
305 Use Proxy
380 Alternative Service

Client error

400 Bad Request
401 Unauthorized
402 Payment Required
403 Forbidden
404 Not Found
405 Method Not Allowed
406 Not Acceptable
407 Proxy Authentication
Required
408 Request Timeout
409 Conflict
410 Gone
411 Length Required
413 Request Entity Too Large
414 Request-URI Too Large
415 Unsupported Media Type
420 Bad Extension
480 Temporarily not available
481 Call Leg/Transaction Does
Not Exist
482 Loop Detected
483 Too Many Hops

484 Address Incomplete
485 Ambiguous
486 Busy Here
487 Request Cancelled
488 Not Acceptable Here

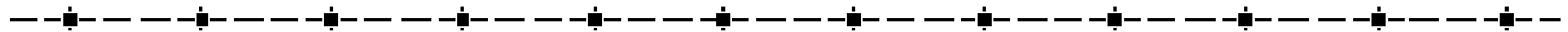
Server error

500 Internal Server Error
501 Not Implemented
502 Bad Gateway
503 Service Unavailable
504 Gateway Time-out
505 SIP Version not
supported

Global failure

600 Busy Everywhere
603 Decline
604 Does not exist
anywhere
606 Not Acceptable

SIP Headers



Examples of Headers Used in Requests and Responses

HEADER

FUNCTION

Call-ID

-Used to uniquely identify a call between two user agents

Contact

-Used to convey URL of original resource requested or request originator

CSeq

-Command Sequence identifies out of sequence requests & retransmissions

From

-Identifies originator of request

To

-Indicates recipient of request

Subject

-Optional header indicating subject of media session

Content-Length

-Number of octets in the message body

Content-Type

-Indicates Internet media type. If not present application/SDP is assumed

User Agent

-Provides additional information about the user agent e.g. manufacturer

Server

-Provides additional information about the User Agent Server

Via

-Records the route taken by a request and used to route response

Record-Route

-Used to force all requests between UAs to be routed through a Proxy

Route

-Forces routing through a path extracted from a Record-Route header

Authorization

-Carries credentials of user agent to a server

Encryption

-Used to specify the portion of a SIP message that has been encrypted

Hide

-Requests next hop proxy to encrypt the Via headers

Priority

-Allow the user agent to set the priority of a request: e.g. urgent, emergency

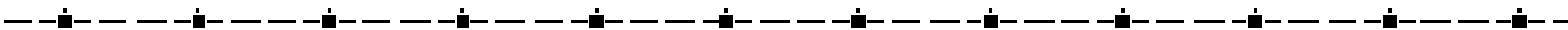
Supported

-List one more options implemented in a user agent or server

Unsupported

-Indicates features that are not supported by the server

SIP abbreviated headers



Short Name	Long Name
c	Content type
e	Content encoding
f	From
i	Call ID
m	Contact (moved)
L	Content length
s	subject
t	To
v	Via

Example:

```
INVITE sip:56655665@149.112.230.64 SIP/2.0
v:SIP/2.0/UDP 149.112.230.198
i:254182843@149.112.230.198
CSeq:48001 INVITE
m:<sip:566255662@149.112.230.198>
User-Agent:3Com ICD 0.10.1.84.7
f:<sip:56625662@149.112.230.64>;tag=4086-16212
t:<sip:56655665@149.112.230.64>
L:105
c:application/sdp

v=0
o=username 0 0 IN IP4 149.112.230.198
s=
c=IN IP4 149.112.230.198
t=0 0
m=audio 4766 RTP/AVP 0
```

REGISTER Request

Mandatory headers

Call-ID

Content-Length

CSeq

From

To

Via

Optional headers

(in examples)

Contact

Expires

User Agent

Usage

Used by user agent to notify SIP network of current IP address and URLs for calls

Examples:

```
REGISTER sip:10.27.5.2 SIP/2.0
Via: SIP/2.0/UDP 10.27.2.3
From: <sip:10.27.2.3>
To: <sip:10.27.2.3>
Call-ID: ac22-11d5-8e63-00c0490ea233@10.27.2.3
CSeq: 3 REGISTER
Contact: <sip:10.27.2.3>
Expires: 30
Content-Length: 0
```

```
REGISTER sip:10.27.2.2 SIP/2.0
Via: SIP/2.0/UDP 10.27.5.9
Call-ID: 5479041946@10.27.5.9
CSeq: 2103 REGISTER
Expires: 3600
Contact: <sip:8213333@10.27.5.9>
User-Agent: 3Com ICD 0.10.1.84.14
From: <sip:8213333@10.27.2.2>;tag=7419-23010
To: <sip:8213333@10.27.2.2>
Content-Length: 0
```

SIP Methods - INVITE

Mandatory headers

Call-ID

Content-Length

CSeq

From

To

Via

Contact

Optional message body
usually SDP

Usage

Used to establish media sessions
between user agents

Example:

```
INVITE sip:8224444@10.27.2.2 SIP/2.0
Via:SIP/2.0/UDP 10.27.5.9
Call-ID:2075716627@10.27.5.9
CSeq:24892 INVITE
Contact:<sip:8223333@10.27.5.9>
User-Agent:3Com ICD 0.10.1.84.14
From: <sip:8223333@10.27.2.2>;tag=17515-10113
To: <sip:8224444@10.27.2.2>
Content-length: 93
Content-type: application/sdp
```

```
v=0
o=username 0 0 IN IP4 10.27.5.9
s=
c=IN IP4 10.27.5.9
t=0 0
m=audio 4248 RTP/AVP 0
```