

---

***Support de Travaux Pratiques*** Qualité de service dans le réseau IP

Niveau 2<sup>ème</sup> année Mastère Professionnel Réseaux Informatiques et  
Télécommunications

***Etablissement*** Enet'com Sfax

***Année universitaire*** 2019-2020

---

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université de Sfax  
Ecole nationale d'électronique et des télécommunications de Sfax



## Travaux Pratiques

### Qualité de service dans le réseau IP



2019 – 2020

2<sup>ème</sup> année Master Professionnel réseaux informatiques & télécommunications

#### Enseignants:

Nessrine ELLOUMI, Assistante Contractuelle à Enet'com de sfax

Kais MNIF, Maître-Assistant à Enet'com de sfax



## Sommaire

**TP 1 :** Configuration d'une politique de qualité de service avec class-map

**TP 2 :** Configuration d'une politique de qualité des services avec ACL

**TP 3 :** Qos et VoIP

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique  
Université de Sfax  
Ecole nationale d'électronique et des télécommunications de Sfax



## Qualité de service dans le réseau IP

### TP 1 : Configuration d'une politique de qualité de service avec class-map

2019 – 2020

2<sup>ème</sup> année Master Professionnel réseaux informatiques & télécommunications

#### Enseignants:

**Nessrine ELLOUMI**, Assistante Contractuelle à Enet'com de sfax

**Kais MNIF**, Maitre-Assistant à Enet'com de sfax

## ***TP 1 : Configuration d'une politique de qualité de service avec class-map***

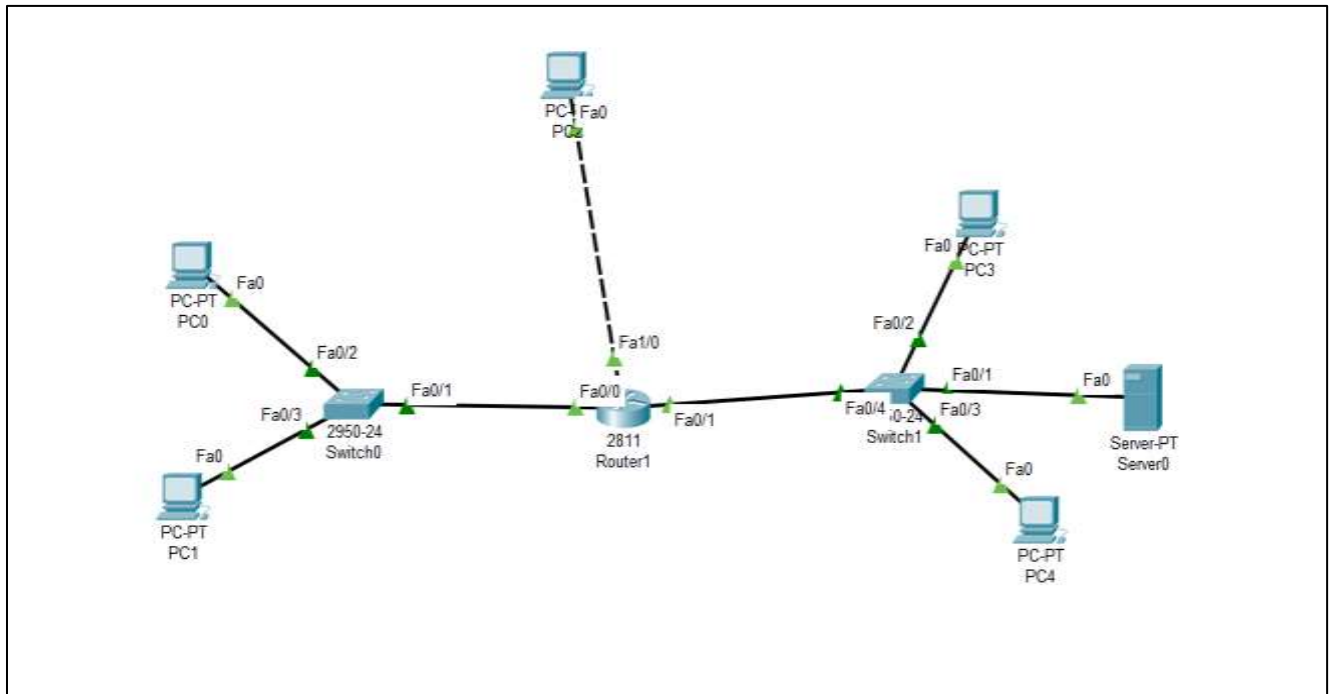
### **Objectif :**

- Mise en place d'une politique de qualité de service sur les routeurs CISCO en utilisant class-map.

### **Manipulation**

Les flux émis vers la partie droite du routeur « Router 1 » ceci est dû au trafic destiné pour le serveur FTP. Une mise en place d'une politique de qualité de service est nécessaire.

Pour favoriser le trafic entrant par l'interface fa1/0 du routeur, Il faut attribuer une priorité haute pour tous les flux entrant par l'interface fa1/0.



### **• Premier exemple : Qualité de service sur une interface**

La mise en place d'une politique de qualité de service en fonction d'une interface sur les routeurs Cisco est la suivante :

1. Déclaration d'une ou plusieurs classes de flux, en fonction des protocoles concernée par le flux.
2. Déclaration d'une politique de qualité de service dans laquelle chaque classe de flux attribuer à un niveau de priorité.
3. Application de cette politique sur une interface suivant le sens d'envoi en entrée où en sortie.

### **Étape 1 - déclaration de classe de flux**

Pour mettre en place une politique de qualité de service sur une interface qui appartienne au routeur **Enetcom** il faut tout d'abord déclarée une class-map "**priorit-interface**". Puis, associer à cette classe le flux provient de l'interface Fast Ethernet 1/0 en « input ».

```
Enetcom>enable
Enetcom#configure terminal
Enetcom(config)#class-map match-all priorit-interface
Enetcom(config-cmap)#match input-interface fastEthernet 1/0
```

Pour vérifier la déclaration de la classe "**priorit-interface**" sur le routeur **Enetcom** taper la commande "show class-map" sous le mode de configuration convenable.

```
Enetcom(config-cmap)#exit
Enetcom(config)#exit
Enetcom#
%SYS-5-CONFIG_I: Configured from console by console
```

```
Enetcom#show class-map
Class Map match-any class-default (id 0)
  Match any
Class Map match-all prio-sur-interface (id 1)
  Match input-interface FastEthernet1/0
Class Map match-all prio-sur-ftp (id 2)
  Match protocol ftp
Class Map match-all priorit-interface (id 3)
  Match input-interface FastEthernet1/0
Enetcom#
```

### **Étape 2 - Déclaration d'une politique de qualité de service (Qos)**

La définition d'une politique de Qos se fait a travers la modification du champ DSCP de qui se trouve dans l'entête du packet IP. Sur le routeur Enetcom on va modifier le champ DSCP des packet provient de l'interface Fast Ethernet 1/0 en donnant une priorité haute (7). Les valeurs de priorité de ce champ sont définies du plus fort DSCP=7 au plus faible DSCP=1

```
Enetcom#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Enetcom(config)#policy-map politique-qos
Enetcom(config-pmap)#class priorit-interface
Enetcom(config-pmap-c)#set ip dscp cs7
```

### **Étape 3 - Application de la politique de qualité de service**

Dans cette étape nous allons appliquer la politique de Qos déclaré dans l'étape 2 sur l'interface fast Ethernet 1/0 du router Enetcom.

```
Enetcom(config-pmap-c)#exit
Enetcom(config-pmap)#exit
Enetcom(config)#interface fastEthernet 1/0
```

Enetcom(config-if)#service-policy output politique-qos

Enregistrer la configuration dans le fichier startup-config.

Enetcom#copy running-config startup-config

Destination filename [startup-config]?

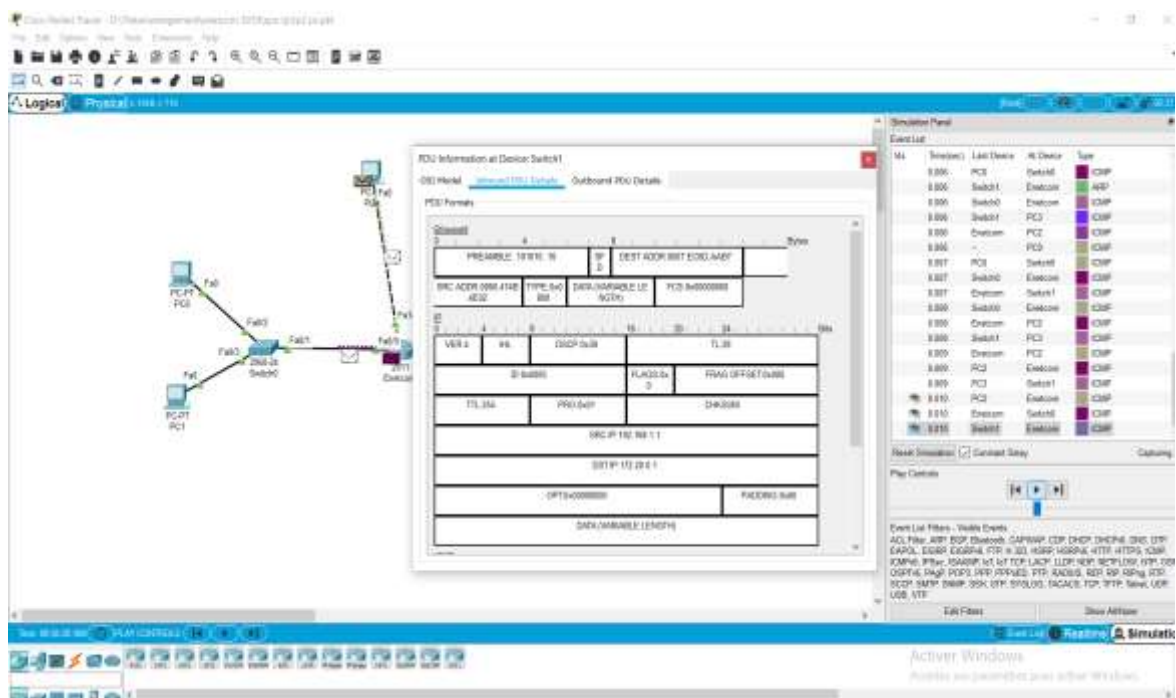
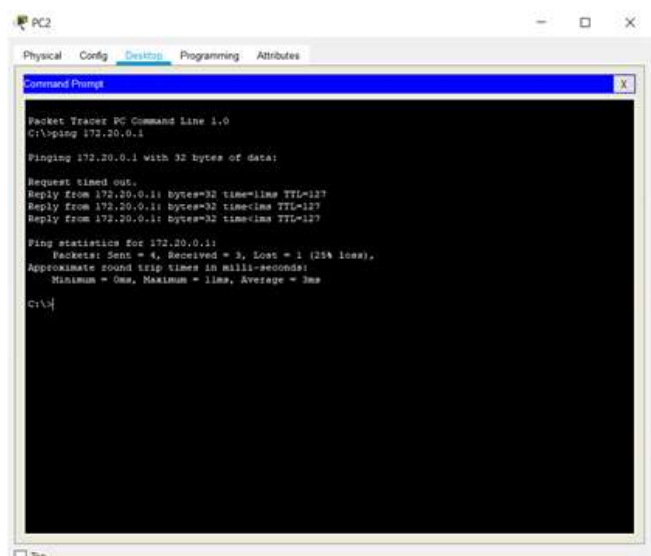
Building configuration...

[OK]

#### Étape 4 - Vérification de la valeur du champ DSCP

Pour vérifier la modification du marquage du champ DSCP il faut entrer en mode « simulation », et envoyer un paquet ICMP à partir du poste PC2 vers PC3.

Résultat de la commande ping entre PC2 et PC3



- **Deuxième exemple : Qualité de service en fonction d'un protocole**

**Étape 1 - Déclaration d'une nouvelle classe de flux**

Pour mettre en place une politique de qualité de service en fonction un protocole applicatif sur le routeur **Enetcom** il faut tout d'abord déclarée une class-map "**priorit-ftp**". Puis, associer à cette classe le flux destiné vers le serveur FTP.

```
Enetcom#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Enetcom(config)#class-map match-all priorit-ftp
Enetcom(config-cmap)#match protocol ftp
```

Pour vérifier la déclaration d'une classe on utilise la commande "show class-map"

```
Enetcom#show class-map
Class Map match-any class-default (id 0)
  Match any
Class Map match-all prio-sur-interface (id 1)
  Match input-interface FastEthernet1/0
Class Map match-all priorit-ftp (id 2)
  Match protocol ftp
```

**Étape 2 - Application de la politique de qualité de service**

Dans cette étape nous allons appliquer la politique de Qos déclaré dans l'étape 1 sur les paquets du trafic Ftp en donnant une priorité faible (1)

```
Enetcom#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Enetcom(config)#policy-map politique-qos
Enetcom(config-pmap)#class priorit-ftp
Enetcom(config-pmap-c)#set ip dscp cs1
```

Enregistrer la configuration dans le fichier startup startup-config

```
Enetcom#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

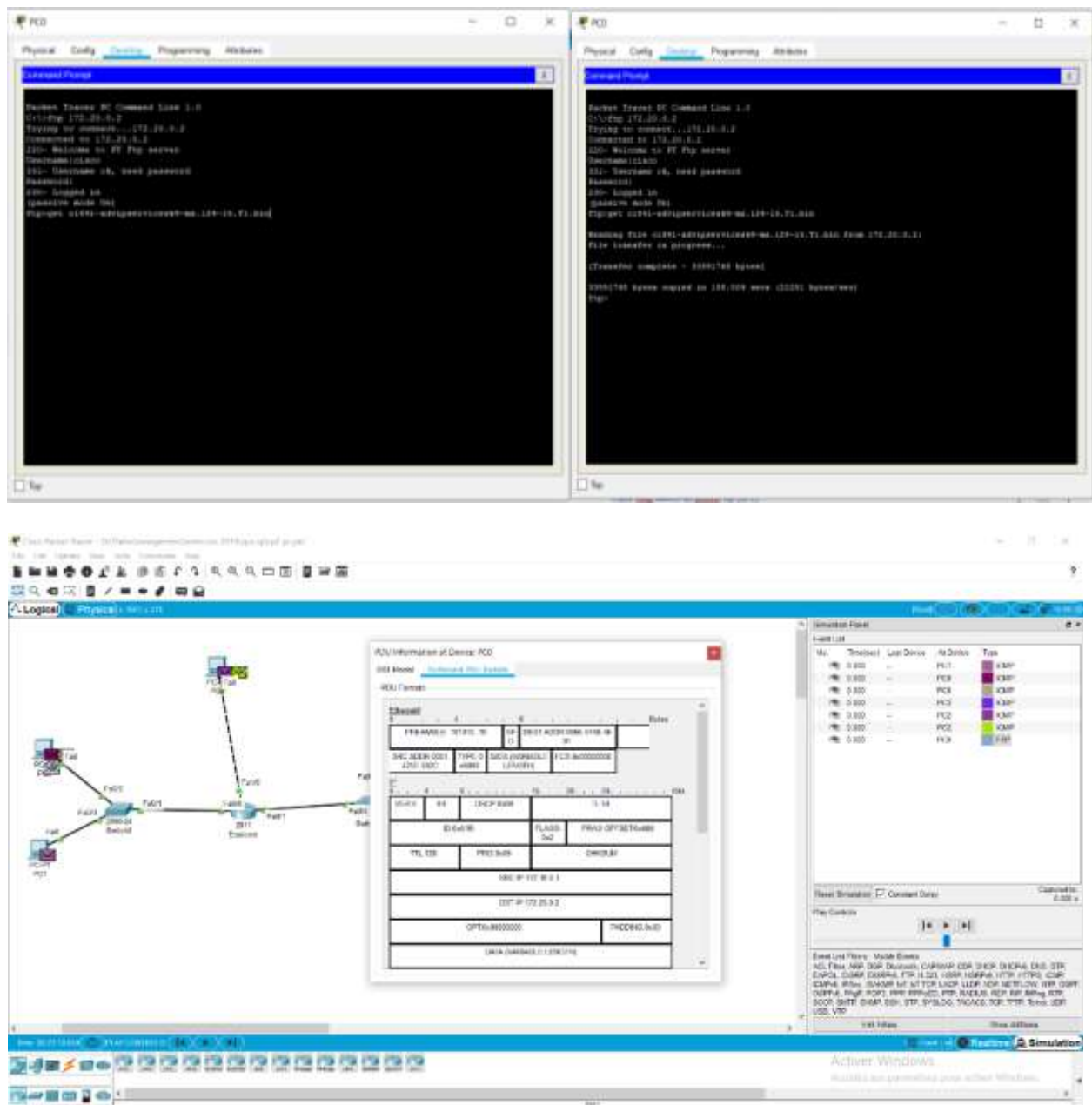
**Étape 3 – Vérification du marquage DSCP**

A partir du **PC0** ouvrir l'invite de commande et connecter au serveur **Ftp** pour télécharger le fichier de mise à jour du system d'exploitation CISCO.

Pour se connecter utilisé :

**Username : cisco et password : cisco (Par défaut)**





- ⇒ Après l'application de la politique de qualité de service sur le routeur Enetcom on remarque que la valeur du champ DSCP est changer suivant la valeur de la priorité donnée par la politique « politique-qos »

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique  
Université de Sfax  
Ecole nationale d'électronique et des télécommunications de Sfax



## Qualité de service dans le réseau IP

### TP 2 Configuration d'une politique de qualité de service avec ACL

2019 – 2020

2<sup>ème</sup> année Master Professionnel réseaux informatiques & télécommunications

#### **Enseignants:**

**Nessrine ELLOUMI**, Assistante Contractuelle à Enet'com de sfax

**Kais MNIF**, Maitre-Assistant à Enet'com de sfax

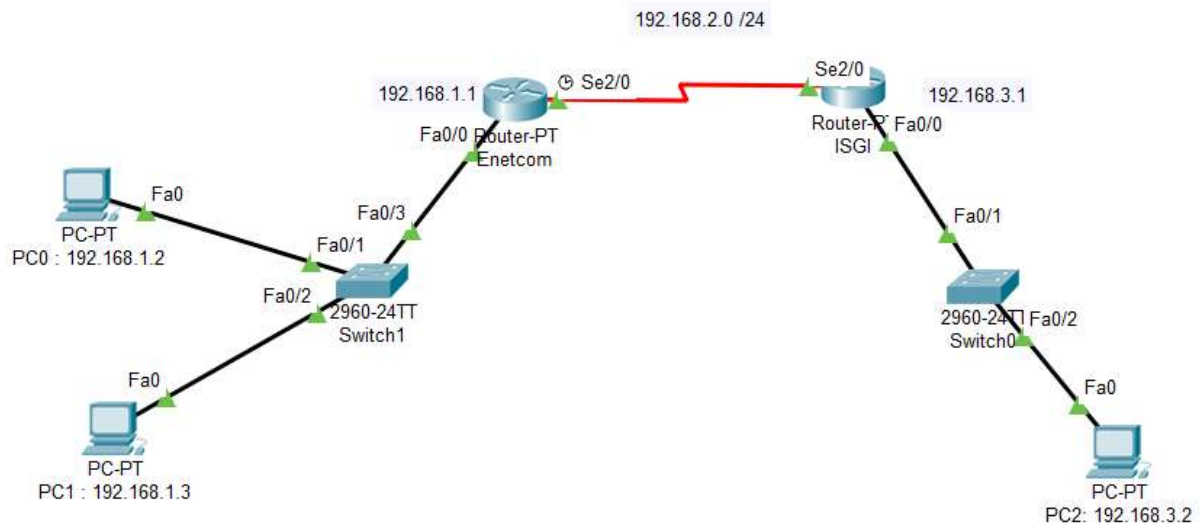
## TP 2 : Configuration d'une politique de qualité de service avec ACL

### Objectif :

- Mise en place d'une politique de qualité de service sur les routeurs CISCO.
- Configuration de file d'attente sur les routeurs CISCO.

### Manipulation :

Simuler le schéma suivant en utilisant « Packet Tracer ».

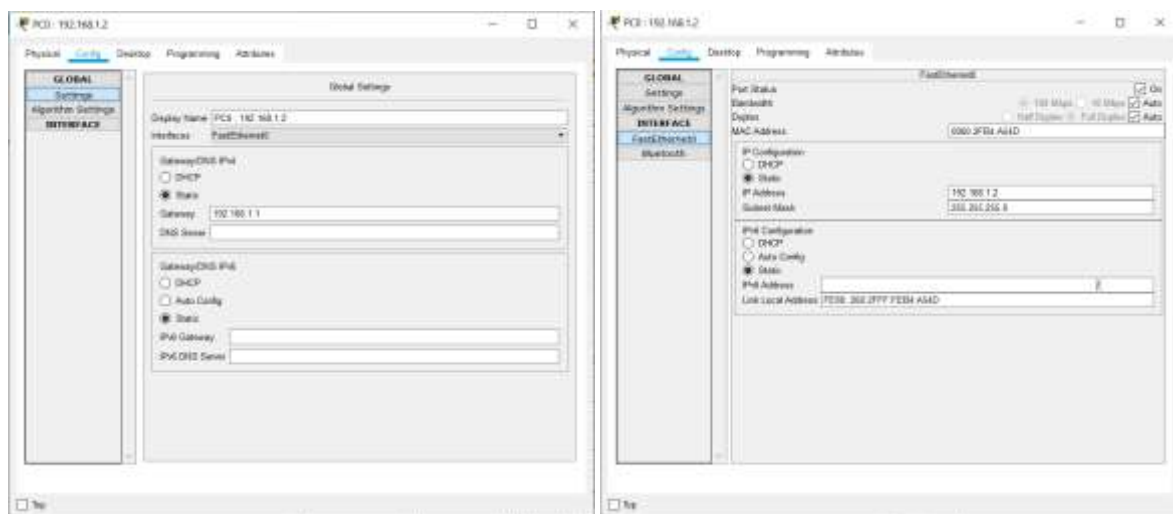


### 1 : Configuration IP des interfaces Ethernet

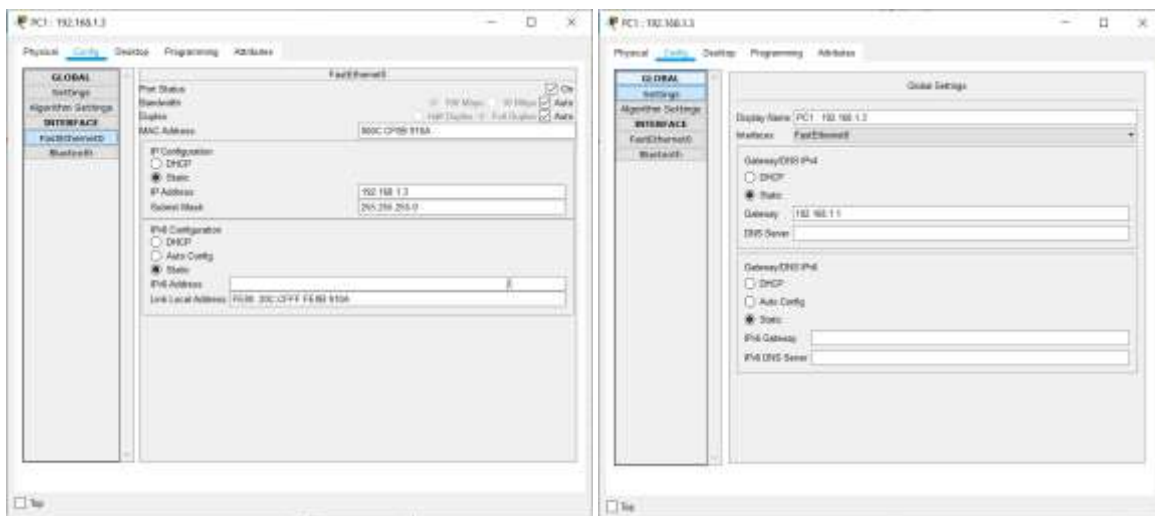
Configurez les informations IP sur les trois ordinateurs du réseau.

Interface	Adresse IP
Fast Ethernet PC0	192.168.1.2 /24
Fast Ethernet PC1	192.168.1.3 /24
Fast Ethernet PC2	192.168.3.2 /24

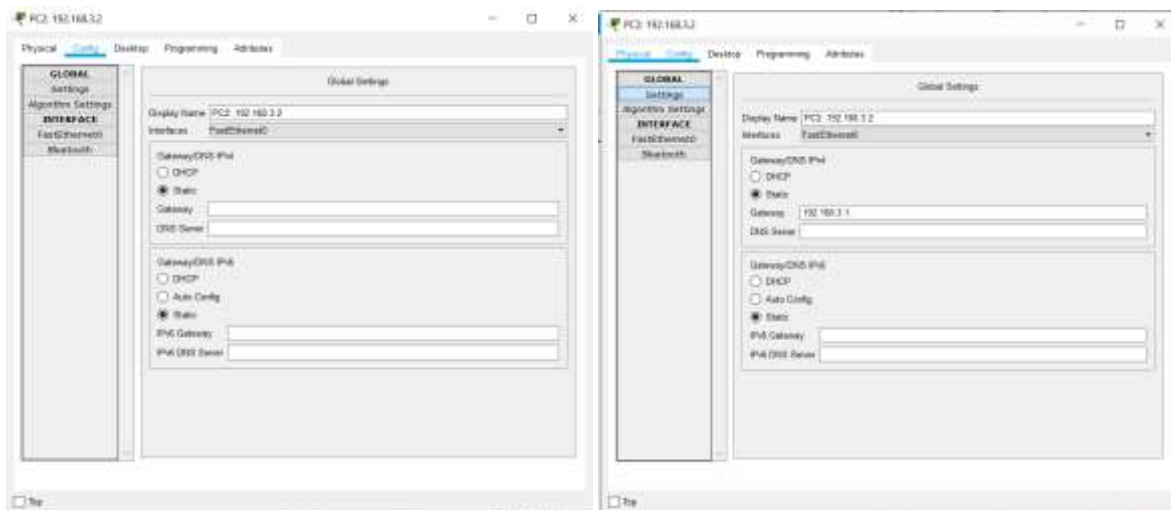
#### ➤ PC 0



## ➤ PC 1



## ➤ PC 2

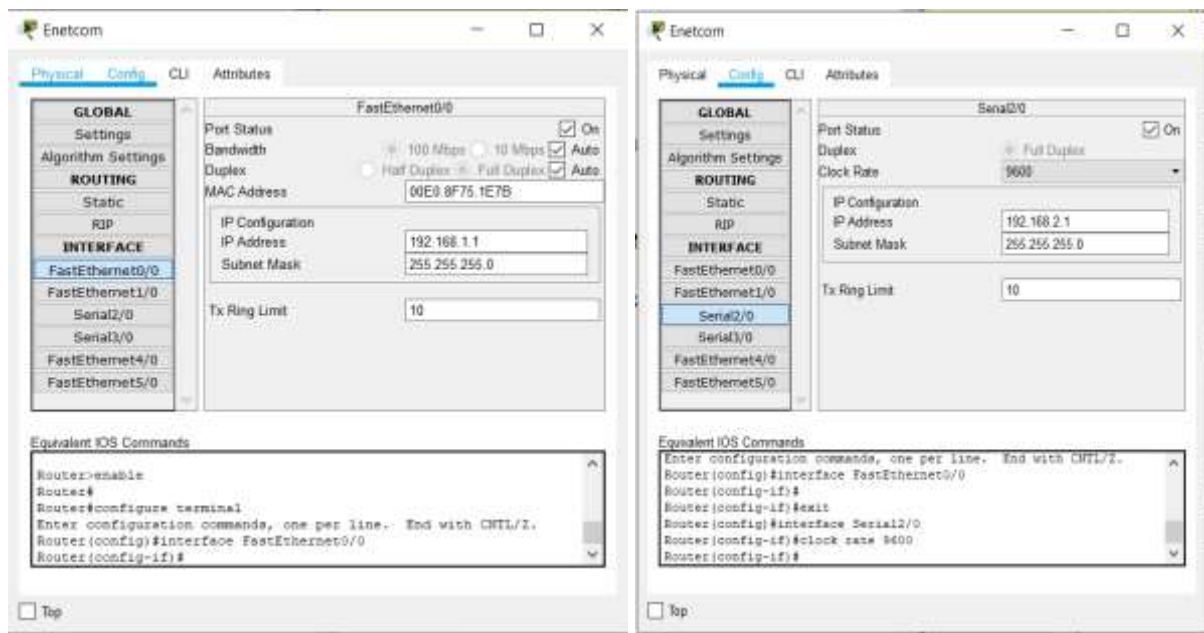
**2 : Configuration des interfaces Ethernet et Série des deux routeurs.**

Configurez les informations IP sur les interfaces des routeur Enetcom et ISGI.

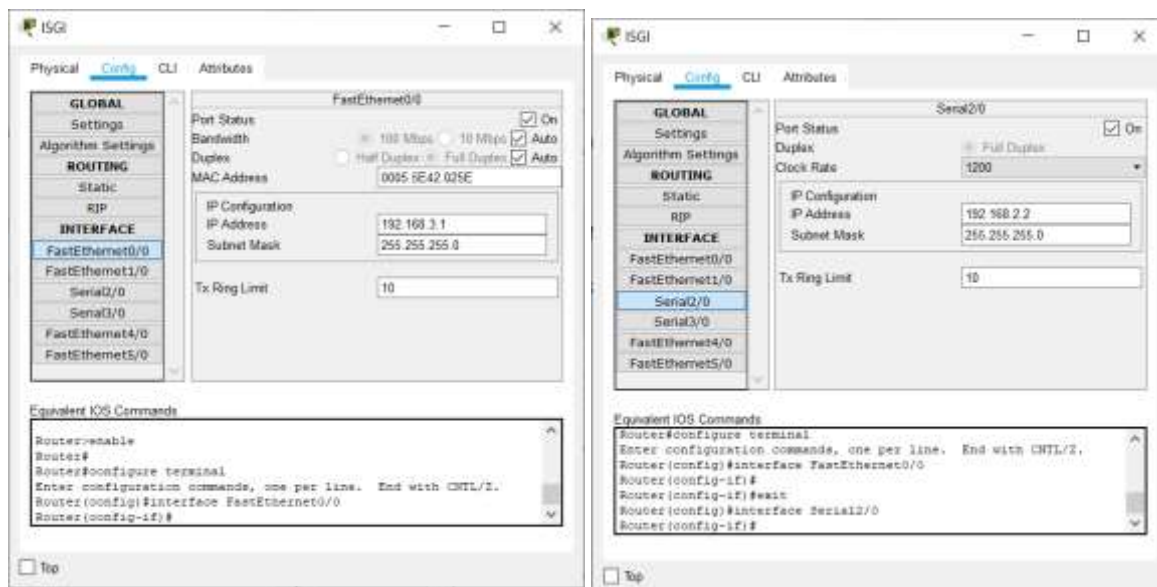
Configurez les interfaces séries des deux routeurs R1 et R2. Utilisez une liaison DCE "**clock rate 9600**" pour faire la synchronisation entre les deux routeurs.

Interface	Adresse IP
Fast Ethernet Enetcom	192.168.1.1 /24
Fast Ethernet ISGI	192.168.13.1 /24
Serial 2/0 Enetcom	192.168.2.1 /24
Serial 2/0 ISGI	192.168.2.2/24

### ➤ Routeur Enetcom



### ➤ Routeur ISGI



Enregistrez la configuration des deux routeur Enetcom et ISGI dans la NVRAM des routeurs.

```
Enetcom#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

```
ISGI#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

### 3 : Configuration du routage entre les réseaux

Pour établir la connexion entre les réseaux il faut configurer un routage en se basant sur les protocoles de routage dynamiques. Dans cette partie nous allons utiliser le protocole RIP pour effectuer le routage dynamique entre les différentes parties du schéma.

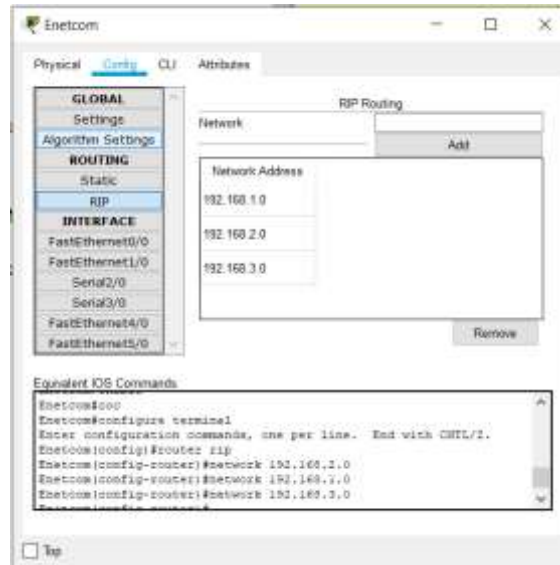
#### ➤ Routeur Enetcom

```
Enetcom(config)#router rip
```

```
Enetcom(config-router)#network 192.168.2.0
```

```
Enetcom(config-router)#network 192.168.1.0
```

```
Enetcom(config-router)#network 192.168.3.0
```



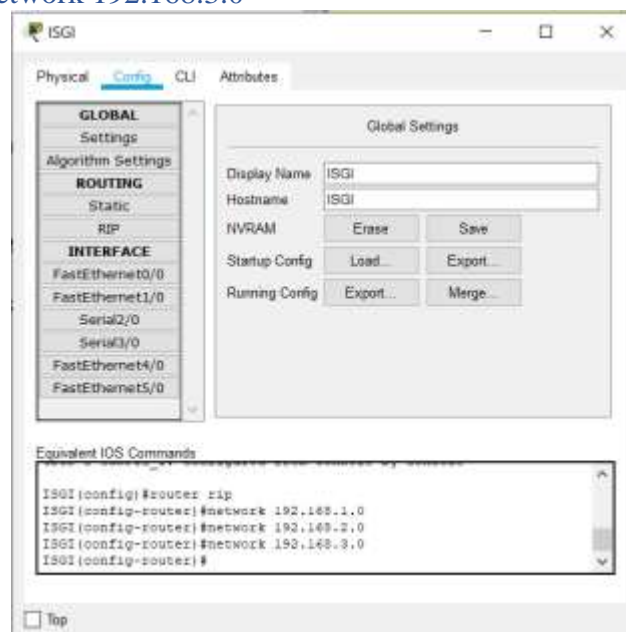
#### ➤ Routeur ISGI

```
ISGI(config)#router rip
```

```
ISGI(config-router)#network 192.168.1.0
```

```
ISGI(config-router)#network 192.168.2.0
```

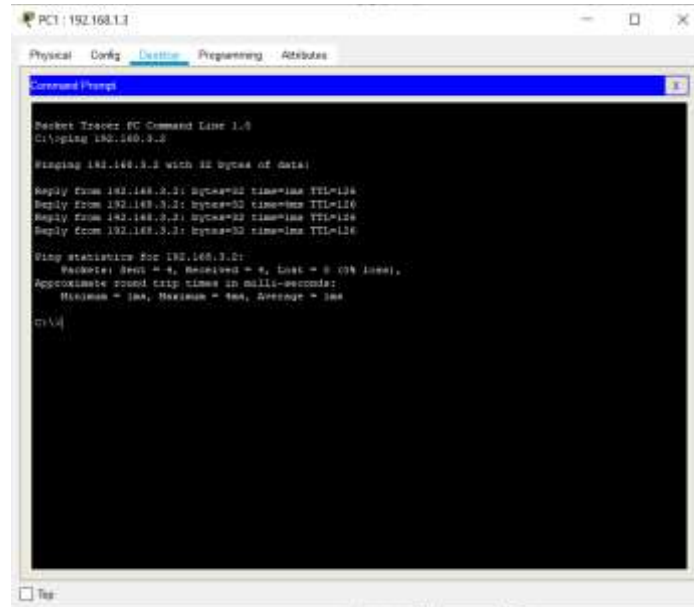
```
ISGI(config-router)#network 192.168.3.0
```



#### 4 : Vérification de la connectivité entre les équipements.

Exécuter la commande Ping @IP pour vérifier le bon fonctionnement du réseau entre tous les équipements.

Résultat de la commande Ping entre PC1 et PC2



### Filtrage de flux

Pour mettre en place une qualité de service il faut identifier les flux qu'on veut filtrer. Sur l'IOS d'un routeur CISCO il existe deux techniques de filtrage qui sont les ACL (**Access Control List**) et les **class-map**.

#### ❖ Configuration d'une ACL :

La configuration d'une ACL sur un routeur se fait à travers la commande « access-list ».

Il existe deux types d'ACL : ACL standard (numéro entre [1-99]) et ACL étendu (numéro entre [100-199])

L'action de filtrage dans une ACL peut être Accepter « permit » ou bien Interdit « deny » Une Configurer une ACL sur le routeur **Enetcom** qui permet de filtrer les paquets TCP provient de tous les @IP à destination du PC2 192.168.3.2, sur le port 80.

```
Enetcom(config)#access-list 101 permit tcp 0.0.0.0 255.255.255.255 192.168.3.2
255.255.255.0 eq 80
```

## Configuration d'une politique de qualité de service

### ❖ Application 1 Configuration d'une politique de Qos :

Configurer une politique de Qos le routeur **Enetcom** afin de filtrer le trafic **HTTP**.

#### ✓ **Etape 1 : Création de la classe de qualité de service**

Créer une class-map « **ma\_classe** » pour filter tous les paquets HTTP.

```
Enetcom>enable
Enetcom#configure terminal
Enetcom(config)#class-map match-all ma_classe
Enetcom(config-cmap)#match protocol Http
```

#### ✓ **Etape 2 : Création d'une politique de Qos**

Créer une politique de qualité de service « **policy** ». Appliquée cette politique sur les paquets HTTP avec une priorité moyenne (5) .

```
Enetcom#configure terminal
Enetcom(config)#policy-map policy
Enetcom(config-pmap-c)#set ip dscp cs5
```

#### ✓ **Etape 3 : Application de la politique de Qos a une interface**

Appliquer la politique de Qos « **policy** » à l'interface serial 2/0 en entrée sur le routeur **Enetcom**.

```
Enetcom(config)#interface serial 2/0
Enetcom(config-if)#service-policy input policy
```

Vérifier la configuration de la politique de Qos.

```
Enetcom#show policy-map interface serial 2/0
Serial2/0
```

Service-policy input: policy

```
Class-map: ma_classe (match-all)
  0 packets, 0 bytes
  5 minute offered rate 0 bps, drop rate 0 bps
  Match: protocol http
  QoS Set
    dscp cs5
    Packets marked 0
```

```
Class-map: class-default (match-any)
  22 packets, 886 bytes
  5 minute offered rate 32 bps, drop rate 0 bps
  Match: any
```

#### ✓ **Etape 4 : Réserve 40 % de la bande passante au trafic HTTP.**



Modifier la politique de Qos pour réservé 40% de la bande passante au trafic http.

```
Enetcom#configure terminal
Enetcom(config)#policy-map policy
Enetcom(config-pmap)#class ma_classe
Enetcom(config-pmap-c)#bandwidth percent 40
```

### ❖ Application 2 : Configuration de file d'attente sur les routeurs CISCO

Configurer une file d'attente à faible latence sur le routeur **Enetcom**.

#### ✓ **Etape 1 : Créer une class-map :**

Créer une class-map appelée « **FI\_Faible** » pour filtrer le trafic du protocole SSH - Secure Shell provient du PC0 vers PC2. Créer une politique « **policy\_Enetcom** » pour réservé 20% de la bande passante au trafic SSH avec un file d'attente à faible latence.

```
Enetcom#configure terminal
Enetcom(config)#access-list 105 permit tcp host 192.168.1.2 host 192.168.3.2 eq 22
Enetcom(config)#class-map FI_Faible
Enetcom(config-cmap)#match access-group 105
Enetcom(config-cmap)#policy-map policy_Enetcom
Enetcom(config-pmap)#class FI_Faible
Enetcom(config-pmap-c)#priority 50
Enetcom(config-pmap-c)#bandwidth percent 20
Enetcom(config-pmap-c)#fair-queue
```

#### ✓ **Etape 2 : Appliquer la politique « policy\_Enetcom » sur les paquets sortant de l'interface Fast Ethernet 0/0 :**

```
Enetcom(config)#interface FastEthernet 0/0
Enetcom(config-if)#service-policy output policy_Enetcom
```

Enregistrer les modifications dans le fichier startup-config.

```
Enetcom#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

### ❖ Application3 : Configurer une file d'attente pondérée sur les routeurs CISCO.

#### ✓ **Etape 1 : Créer deux class-map sur le routeur ISGI :**

- Pour la première class-map « **classe1\_ISGI** », la ACL 101 permet d'autorisé le trafic Telnet provenant du PC0 vers PC2. Alloué 40% de la bande passante a cette classe.
- Pour la deuxième class-map, « **classe2\_ISGI** », la ACL 102 permet d'autorisé le trafic HTTPS provenant du PC0 vers PC2. Alloué 20% de la bande passante a cette classe.

```
ISGI>enable
ISGI#configure terminal
ISGI(config)#access-list 101 permit tcp host 192.168.1.2 host 192.168.3.2 eq 23
ISGI(config)#access-list 102 permit tcp host 192.168.1.2 host 192.168.3.2 eq 443
ISGI(config)#class-map classe1_ISGI
ISGI(config-cmap)#match access-group 101
ISGI(config-cmap)#exit
ISGI(config)#class-map classe2_ISGI
ISGI(config-cmap)#match access-group 102
ISGI(config-cmap)#exit
ISGI(config)#policy-map policy
ISGI(config-pmap)#class classe1_ISGI
ISGI(config-pmap-c)#bandwidth percent 40
ISGI(config-pmap-c)#que
ISGI(config-pmap-c)#queue-limit 30
ISGI(config-pmap-c)#exit
ISGI(config-pmap)#class classe2_ISGI
ISGI(config-pmap-c)#bandwidth percent 20
```

- ✓ **Etape 2 : appliquer la première class-map sur les paquets sortant de l'interface Fast Ethernet 0/0.**

```
ISGI(config)#interface fastEthernet 0/0
ISGI(config-if)#service output policy
```

Enregistrer les modifications dans le fichier startup-config

```
ISGI#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

Ministère de l'Enseignement Supérieure et de la Recherche Scientifique  
Université de Sfax  
Ecole nationale d'électronique et des télécommunications de Sfax



## Qualité de service dans le réseau IP

### TP 3 Qos et VoIP

2019 – 2020

2<sup>ème</sup> année Master Professionnel réseaux informatiques & télécommunications

#### Enseignants:

**Nessrine ELLOUMI**, Assistante Contractuelle à Enet'com de sfax

**Kais MNIF**, Maitre-Assistant à Enet'com de sfax

## TP 3 : Qos et VoIP

### Objectifs :

- Prise en main du logiciel WireShark
- Visualiser et capturer les trames, les paquets de différents protocoles réseau
- Analyse de la qualité de service du trafic VoIP sur le réseau.

Afin d'analyser la qualité de service du trafic VoIP on a utilisé un serveur **Asterisk-sur fedora** installer et configurer et deux client **X-lite**.

Dans ce TP nous allons utiliser **Wireshark** pour capturer et analyser le trafic VoIP.



Version 3.6.1 (v3.6.1-0-ga0a473c7c1ba)

Wireshark est un analyseur de paquets réseau qui permet de présenter les données des paquets capturées d'une manière détaillée. Il est l'un des meilleurs analyseurs de trafic réseau. Wireshark est un outil qui aide l'administrateur réseau d'examiner ce qui se passe à l'intérieur de la carte réseau. Wireshark est un logiciel open source et gratuit.

Wireshark est utilisé par :

- ✓ Les administrateurs réseau pour résoudre les problèmes de réseau
- ✓ Les ingénieurs en sécurité réseau pour examiner les problèmes de sécurité
- ✓ Les ingénieurs d'assurance qualité pour vérifier les applications réseau
- ✓ Les développeurs pour déboguer les implémentations de protocole
- ✓ Les gens pour apprendre le fonctionnement interne des protocoles réseaux

Wireshark offre de nombreuses fonctionnalités:

- ✓ Disponible pour UNIX et Windows.
- ✓ Capturez des données de paquets en direct à partir d'une interface réseau.
- ✓ Ouvrez les fichiers contenant des données de paquets capturées avec tcpdump/WinDump, Wireshark et de nombreux autres programmes de capture de paquets.
- ✓ Importez des paquets à partir de fichiers texte contenant des vidages hexadécimaux de données de paquets.
- ✓ Affichez les paquets avec des informations de protocole très détaillées.
- ✓ Enregistrer les données de paquets capturées.
- ✓ Exportez les paquets dans un certain nombre de formats de fichiers de capture.
- ✓ Filtrez les paquets selon des critères personnalisés.
- ✓ Recherchez des paquets sur de nombreux critères.
- ✓ Coloriser l'affichage des paquets en fonction des filtres.
- ✓ Créer diverses statistiques.

## Manipulation :

Pour analyser le trafic VoIP nous avons installé l'analyseur de trafic Wireshark sur une station X-lite. Puis nous avons démarré une session SIP entre les deux clients. Enfin nous avons enregistré le trafic (avec Wireshark) pour une session de trois minutes.

Les messages SIP échangés entre les différentes entités SIP sont enregistrés dans un fichier nommé « **trafic voix ip.pcap** ».

Pour visualiser le trafic capturé, ouvrir le fichier « **trafic voix ip.pcap** » enregistrés sous **D:/trafic voix ip.pcap** dans wireshark.

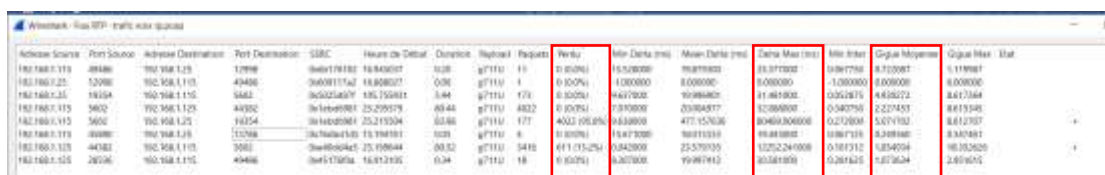
1. Déterminer pour cette session :

- ✓ Nombre de paquets transmis.

Le nombre de paquet transmis pour cette session est : 17820 paquets

- ✓ Déterminer les paramètres de QoS

Les paramètres de la Qos de cette session sont : taux de perte, délai max, gigue moyenne.



Adresse Source	Port Source	Adresse Destination	Port Destination	SSRC	Heure de Début	Durée	Protocole	Paquets	Perte	Min Delta (ms)	Mean Delta (ms)	Delta Max (ms)	Min Jitter	Jitter Moyenne	Jitter Max	Stat
192.168.0.119	8000	192.168.0.125	12000	0x00000001	18.840000	0.00	UDP	1	0.00%	1.000000	0.000000	1.000000	0.000000	0.000000	1.000000	
192.168.0.125	12000	192.168.0.119	8000	0x00000002	18.840000	0.00	UDP	1	0.00%	1.000000	0.000000	1.000000	0.000000	0.000000	1.000000	
192.168.0.119	8000	192.168.0.125	12000	0x00000003	18.840000	0.00	UDP	1	0.00%	1.000000	0.000000	1.000000	0.000000	0.000000	1.000000	
192.168.0.125	12000	192.168.0.119	8000	0x00000004	18.840000	0.00	UDP	1	0.00%	1.000000	0.000000	1.000000	0.000000	0.000000	1.000000	
192.168.0.119	8000	192.168.0.125	12000	0x00000005	18.840000	0.00	UDP	1	0.00%	1.000000	0.000000	1.000000	0.000000	0.000000	1.000000	
192.168.0.125	12000	192.168.0.119	8000	0x00000006	18.840000	0.00	UDP	1	0.00%	1.000000	0.000000	1.000000	0.000000	0.000000	1.000000	
192.168.0.119	8000	192.168.0.125	12000	0x00000007	18.840000	0.00	UDP	1	0.00%	1.000000	0.000000	1.000000	0.000000	0.000000	1.000000	
192.168.0.125	12000	192.168.0.119	8000	0x00000008	18.840000	0.00	UDP	1	0.00%	1.000000	0.000000	1.000000	0.000000	0.000000	1.000000	
192.168.0.119	8000	192.168.0.125	12000	0x00000009	18.840000	0.00	UDP	1	0.00%	1.000000	0.000000	1.000000	0.000000	0.000000	1.000000	
192.168.0.125	12000	192.168.0.119	8000	0x0000000A	18.840000	0.00	UDP	1	0.00%	1.000000	0.000000	1.000000	0.000000	0.000000	1.000000	

- ✓ Le protocole **SIP** utilise quel protocole de la couche transport? Quel est le port utilisé par le serveur.

Le protocole Session Initiation Protocol (SIP) est supporté par UDP ou TCP sur le port 5060 et/ou 5061.

No.	Time	Source	Destination	Protocol	Length	Info
443	4.284403	192.168.1.25	192.168.1.25	SIP/SDP	359	Request: INVITE sip:115@192.168.1.25

SIP/2.0  
 192.168.1.25  
 192.168.1.25  
 INVITE  
 359  
 Request: ACK sip:115@192.168.1.25

- ✓ Le nombre total des messages **SIP** échangés.

Le nombre des message SIP échangés est : 38

- ✓ Préciser les requêtes et les réponses.

- Les requêtes :

No.	Time	Source	Destination	Protocol	Length	Info
441	4.282403	192.168.1.115	192.168.1.25	SIP/SDP	1045	Request: INVITE sip:115@192.168.1.25

- Les réponses :

No.	Time	Source	Destination	Protocol	Length	Info
432	4.362910	192.168.1.25	192.168.1.110	TCP	60	607 Proxy Authentication Required
433	4.388093	192.168.1.25	192.168.1.110	TCP	60	607 Proxy
434	4.390321	192.168.1.25	192.168.1.110	TCP	60	607 Proxy
435	5.004085	192.168.1.25	192.168.1.110	TCP	60	607 Proxy Authentication Required
436	5.006249	192.168.1.25	192.168.1.110	TCP	60	607 Proxy
437	10.119081	192.168.1.25	192.168.1.110	HTTP/1.0	10	200 OK (text/css)
438	10.129147	192.168.1.110	192.168.1.25	HTTP	41	200 OK (text/css)
439	10.367013	192.168.1.25	192.168.1.110	TCP	60	607 Proxy Authentication Required
440	10.3684317	192.168.1.25	192.168.1.110	TCP	60	607 Proxy
441	10.369055	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
442	10.373638	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
443	10.374150	192.168.1.25	192.168.1.110	TCP	60	607 Proxy
444	10.375491	192.168.1.25	192.168.1.110	TCP	60	607 Proxy Authentication Required
445	10.376935	192.168.1.25	192.168.1.110	TCP	60	607 Proxy
446	11.016047	192.168.1.25	192.168.1.110	TCP	60	607 Proxy
447	11.020713	192.168.1.25	192.168.1.110	TCP	60	607 Proxy
448	12.003499	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
449	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
450	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
451	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
452	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
453	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
454	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
455	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
456	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
457	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
458	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
459	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
460	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
461	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
462	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
463	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
464	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
465	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
466	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
467	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
468	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
469	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
470	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
471	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
472	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
473	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
474	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
475	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
476	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
477	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
478	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
479	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
480	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
481	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
482	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
483	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
484	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
485	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
486	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
487	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
488	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
489	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
490	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
491	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
492	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
493	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
494	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
495	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
496	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
497	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
498	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
499	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
500	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
501	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
502	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
503	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
504	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
505	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
506	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
507	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
508	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
509	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
510	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
511	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
512	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
513	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
514	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
515	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
516	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
517	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
518	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
519	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
520	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
521	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
522	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
523	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
524	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
525	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
526	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
527	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
528	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
529	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
530	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
531	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
532	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
533	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
534	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
535	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
536	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
537	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
538	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
539	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
540	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
541	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
542	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
543	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
544	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
545	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
546	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
547	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
548	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
549	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
550	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
551	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
552	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
553	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
554	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
555	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
556	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
557	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
558	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
559	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
560	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
561	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
562	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
563	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
564	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
565	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
566	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
567	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
568	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
569	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
570	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
571	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
572	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
573	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
574	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
575	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
576	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
577	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
578	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
579	12.010400	192.168.1.110	192.168.1.25	TCP	60	607 Proxy
580	12.010400	192.168.1.110				

- ✓ Visualiser les messages échangés entre le client et le serveur.

- Exemple de requête

Wireshark packet capture showing SIP messages between 192.168.1.115 and 192.168.1.25. The selected packet is a INVITE request (1210 bytes) with details expanded below.

No.	Time	Source	Destination	Protocol	Length	Info
441	4.282483	192.168.1.115	192.168.1.25	SIP/SDP	1845	Request: INVITE sip:115@192.168.1.25
442	4.292999	192.168.1.25	192.168.1.115	SIP	599	Status: 407 Proxy Authentication Required
443	4.303885	192.168.1.115	192.168.1.25	SDP	350	Request: ACK sip:115@192.168.1.25
444	4.304651	192.168.1.115	192.168.1.25	SIP/SDP	1210	Request: INVITE sip:115@192.168.1.25
445	4.304999	192.168.1.25	192.168.1.115	SIP	510	Status: 100 Trying
456	4.308109	192.168.1.25	192.168.1.115	SIP	512	Status: 180 Ringing
1554	15.053846	192.168.1.115	192.168.1.25	SIP/SDP	1857	Request: INVITE sip:115@192.168.1.25
1555	15.054405	192.168.1.25	192.168.1.115	SIP	611	Status: 407 Proxy Authentication Required
1556	15.055313	192.168.1.115	192.168.1.25	SIP	371	Request: ACK sip:115@192.168.1.25
1557	15.055908	192.168.1.115	192.168.1.25	SIP/SDP	1222	Request: INVITE sip:115@192.168.1.25
1558	15.056529	192.168.1.25	192.168.1.115	SIP	528	Status: 100 Trying
1560	15.159881	192.168.1.25	192.168.1.115	SIP/SDP	884	Status: 200 OK (INVITE)
1587	15.266563	192.168.1.115	192.168.1.25	SIP	642	Request: ACK sip:115@192.168.1.25
1588	15.268935	192.168.1.25	192.168.1.115	SIP	466	Request: BYE sip:115@192.168.1.115:49788
1600	15.370107	192.168.1.115	192.168.1.25	SIP	419	Status: 200 OK (BYE)
1769	16.087013	192.168.1.25	192.168.1.115	SIP/SDP	872	Status: 200 OK (INVITE)

Frame 444: 1210 bytes on wire (9680 bits), 1210 bytes captured (9680 bits) on Ethernet II, Src: SURECOM\_b0:25:c8 (00:02:44:b0:25:c8), Dst: SURECOM\_b0:25:c1 (00:02:44:b0:25:c1)

Internet Protocol Version 4, Src: 192.168.1.115, Dst: 192.168.1.25

User Datagram Protocol, Src Port: 49788, Dst Port: 5060

Source Port: 49788  
Destination Port: 5060  
Length: 1176  
Checksum: 0x7515 [unverified]  
[Checksum Status: Unverified]  
[Stream Index: 4]  
[Timestamps]

UDP payload (1168 bytes)

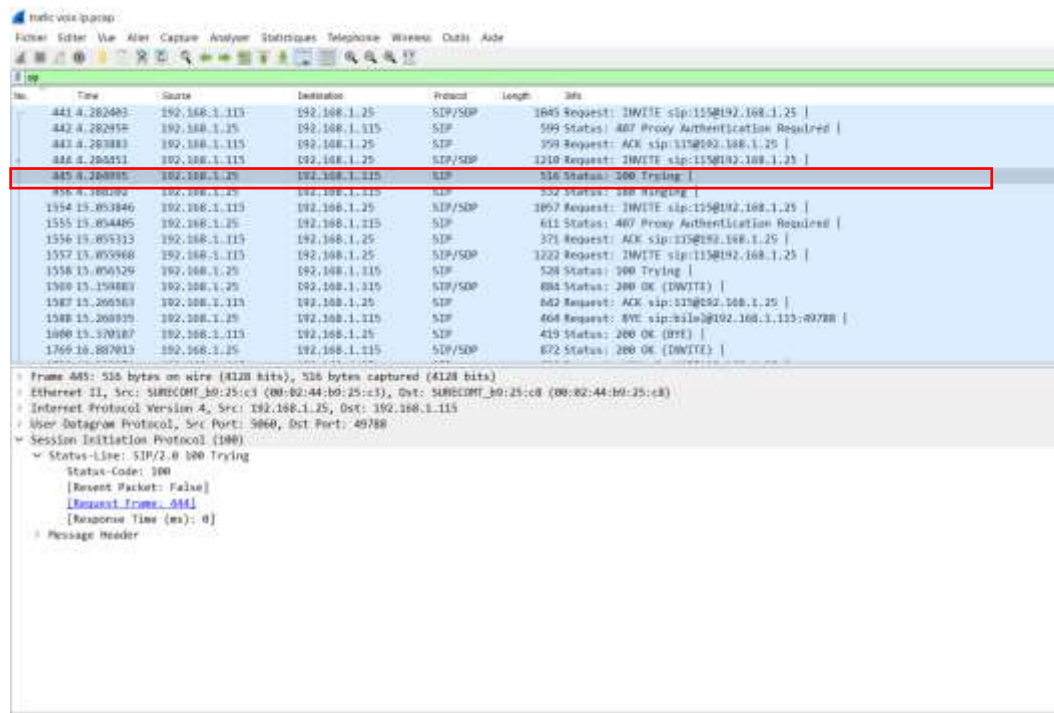
Session Initiation Protocol (INVITE)

Request-Line: INVITE sip:115@192.168.1.25 SIP/2.0  
Method: INVITE  
Request-URI: sip:115@192.168.1.25  
[Revised Packet: False]

Message Header  
Message Body



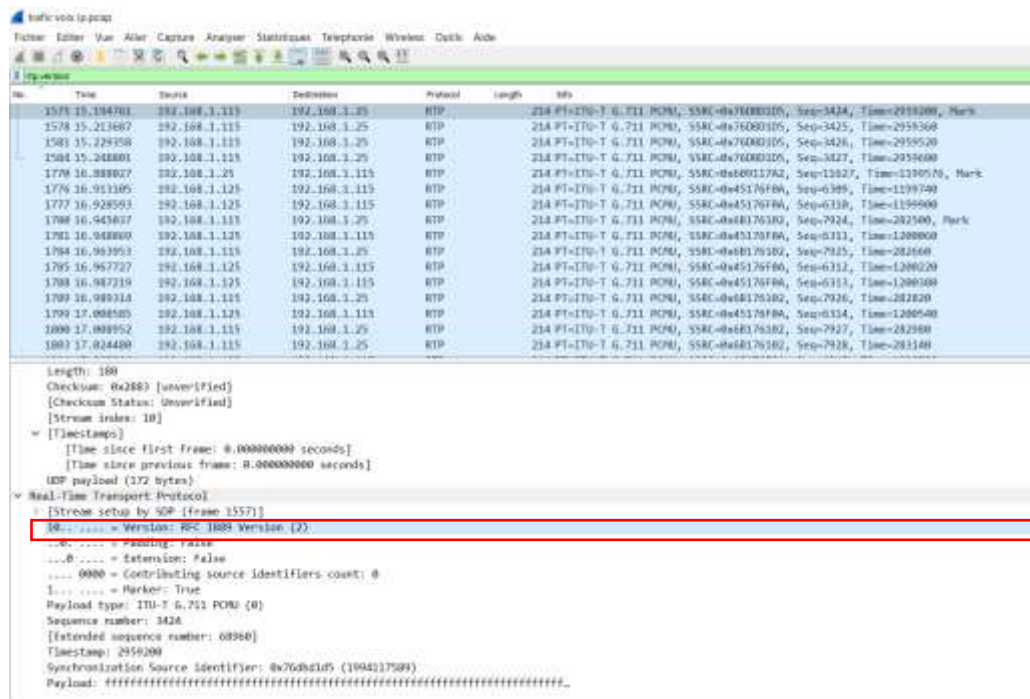
- Exemple de réponse :



## 2. Sélectionner un paquet **RTP** :

- ✓ Déterminer la version.

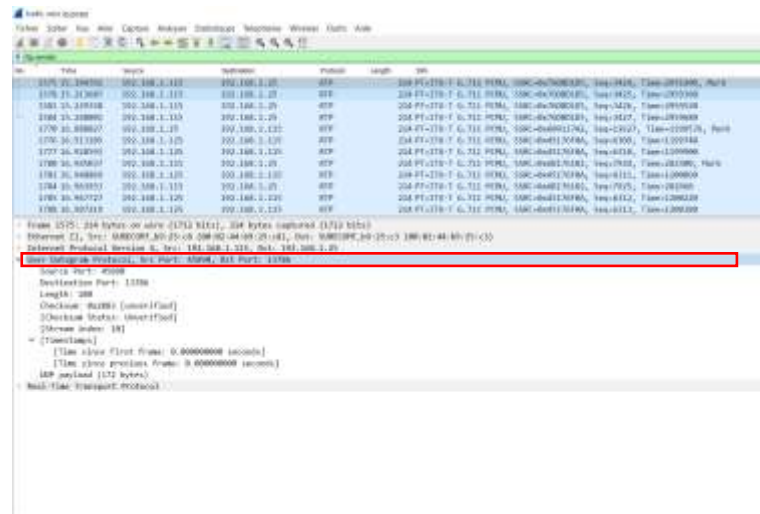
La version du protocole RTP utilisée est : RTP Version (2)





- ✓ Quel est le protocole de transport utilisé par RTP.

Le protocole RTP est supporté par UDP.



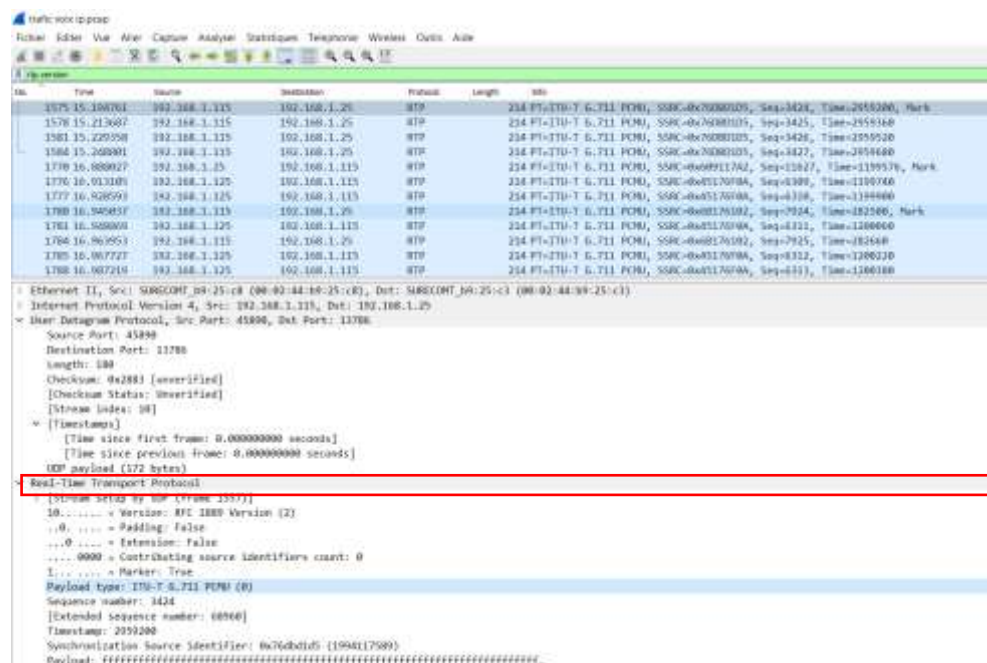
- ✓ Déterminer la taille d'un paquet Real-Time Transport Protocol (RTP).

La taille d'un paquet Real-Time Transport Protocol (RTP) 214 octets.

- ✓ Déterminer le type du payload, expliquer.

Le type payload est : ITU-T G.711 PCM (0)

Ce payload indique que les données transporter sont de type voix IP. Le codeur audio G.711 est utilisé en téléphonie pour fournir un son de qualité interurbaine à 64 kbit/s.



- ✓ Quel est le rôle du champ Sequence Number pour deux paquets successives.

Le champ Sequence Number permet d'organiser les paquets reçus et détecter les paquets perdus

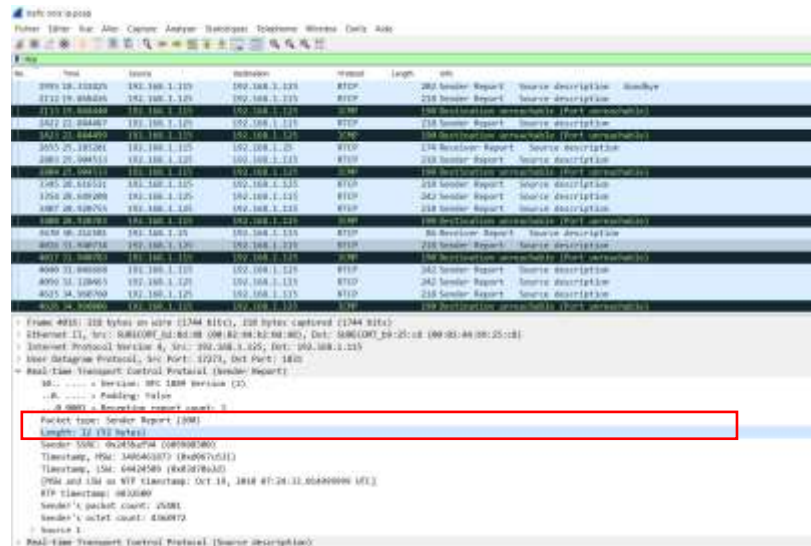
## 3. Sélectionner un paquet RTCP (SR) :

- ✓ Déterminer la taille de ce paquet.

Le taille de ce paquet RTCP (SR) est : 12 octet

- ✓ Quel est son type, donner sa valeur.

Ce paquet est de type: Sender Report, sa valeur est : 200



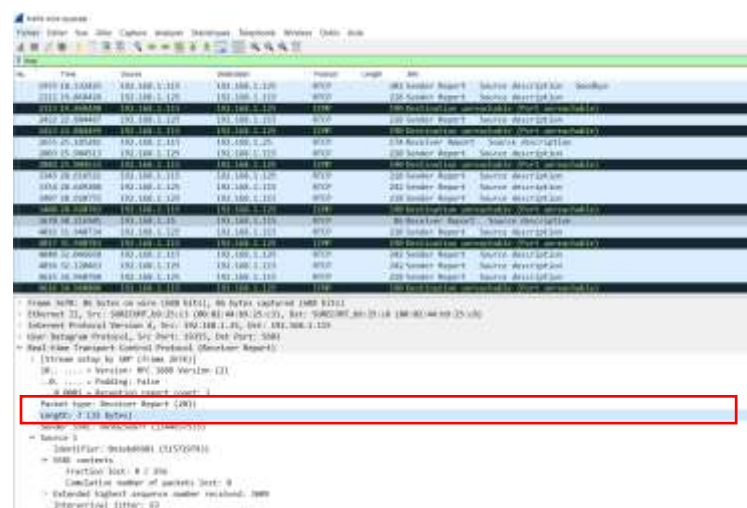
## 4. Sélectionner un paquet RTCP (RR) :

- ✓ Déterminer la taille de ce paquet.

Le taille de ce paquet RTCP (RR) est : 7 octet.

- ✓ Quel est son type, donner sa valeur

Ce paquet est de type: Receiver Report, sa valeur est : 201



## 5. Est-ce que les paquets RTCP sont envoyés de façon périodique?

Oui, le protocole RTCP est basé sur l'envoi périodique de paquets de contrôle. RTP permet d'envoyer des informations sur les participants d'une session et sur la qualité de service.

## 6. Sélectionnez le paquet RTP (n°1584) :

- ✓ Déterminer la source qui a généré ce paquet

L'adresse source qui a généré ce paquet est : 192.168.1.115

- ✓ Déterminer la taille de ce paquet (en octets)

Le taille du paquet RTP est : 214 octets

- ✓ Déterminer son numéro de séquence

Le numéro de séquence de ce paquet est: 3427

- ✓ Déterminer son timestamp (TS).

Le Timestamp de ce paquet est: 2959680

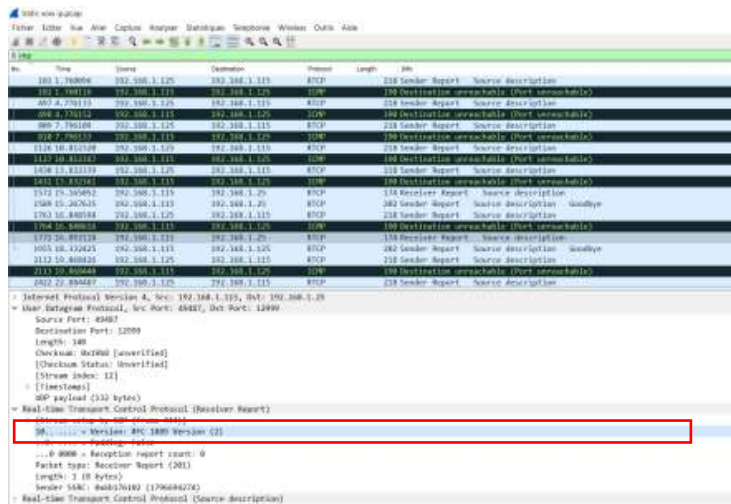
- ✓ Déterminer la valeur du champ SSRC

La valeur du champ SSRC est : SSRC=0x76DBD1D5

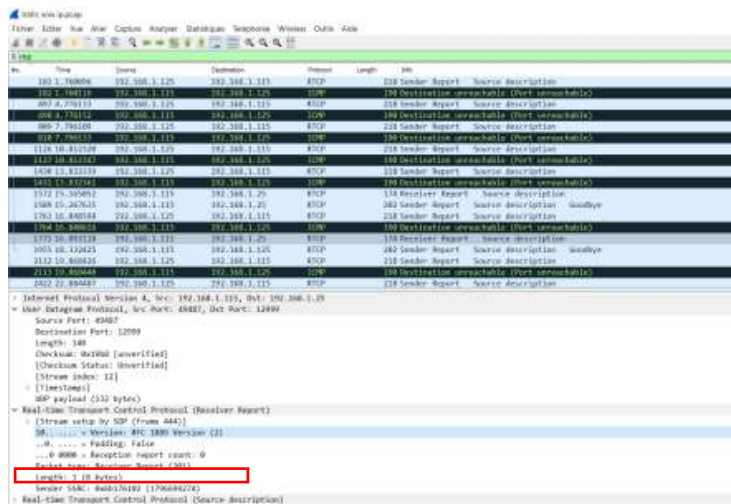
No.	Time	Source	Destination	Protocol	Length	Info
1575	15.194761	192.168.1.115	192.168.1.25	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x76DBD1D5, Seq=3424, Time=2959280, Mark
1576	15.213667	192.168.1.115	192.168.1.25	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x76DBD1D5, Seq=3425, Time=2959360
1581	15.229258	192.168.1.115	192.168.1.25	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x76DBD1D5, Seq=3426, Time=2959520
1584	15.248881	192.168.1.115	192.168.1.25	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x76DBD1D5, Seq=3427, Time=2959680
1770	16.888827	192.168.1.25	192.168.1.115	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x00000000, Seq=13627, Time=1199576, Mark
1776	16.913185	192.168.1.125	192.168.1.115	RTP	214	PT=ITU-T G.711 PCMU, SSRC=0x05176F0A, Seq=6389, Time=1199748

## 7. Sélectionner le paquet RTCP (n°1771) :

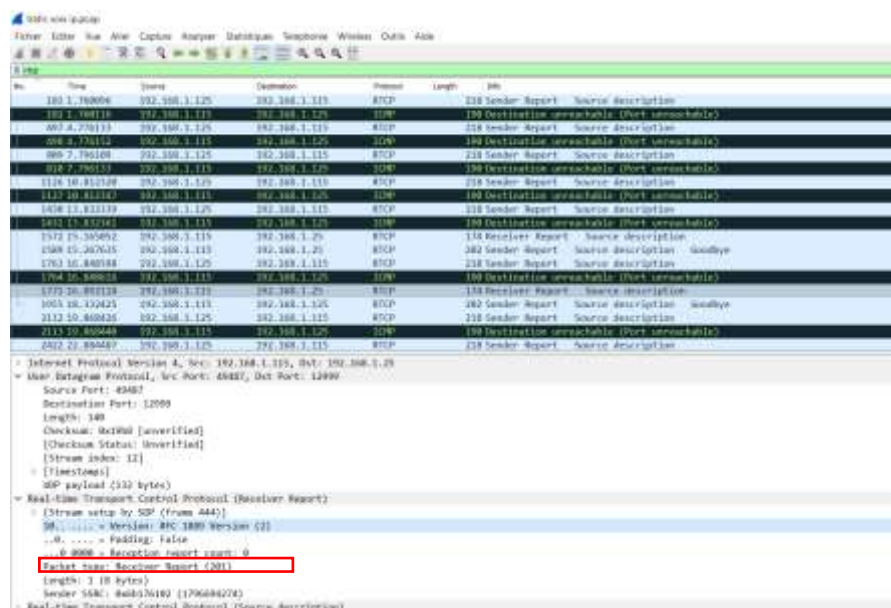
- ✓ Déterminer la version du protocole RTCP



✓ Déterminer la taille (en octets)

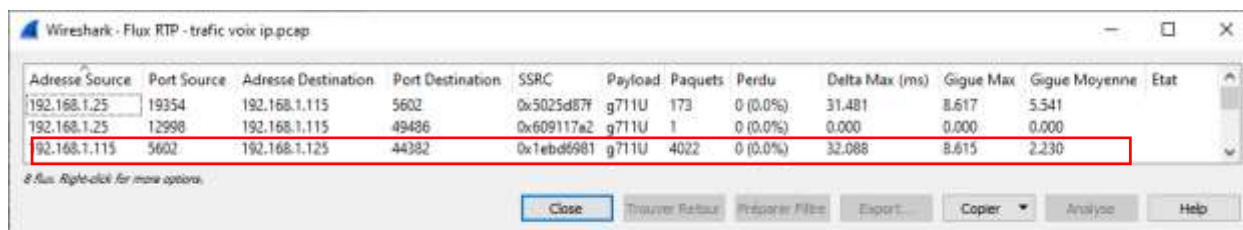


✓ Son type, expliquer



Pour voir les statistiques sur les paramètres de QoS comme le taux de pertes, le délai max et la valeur de la gigue (max et moyenne) pour une session VoIP avec Wireshark

Téléphonie/RTP/Flux RTP/

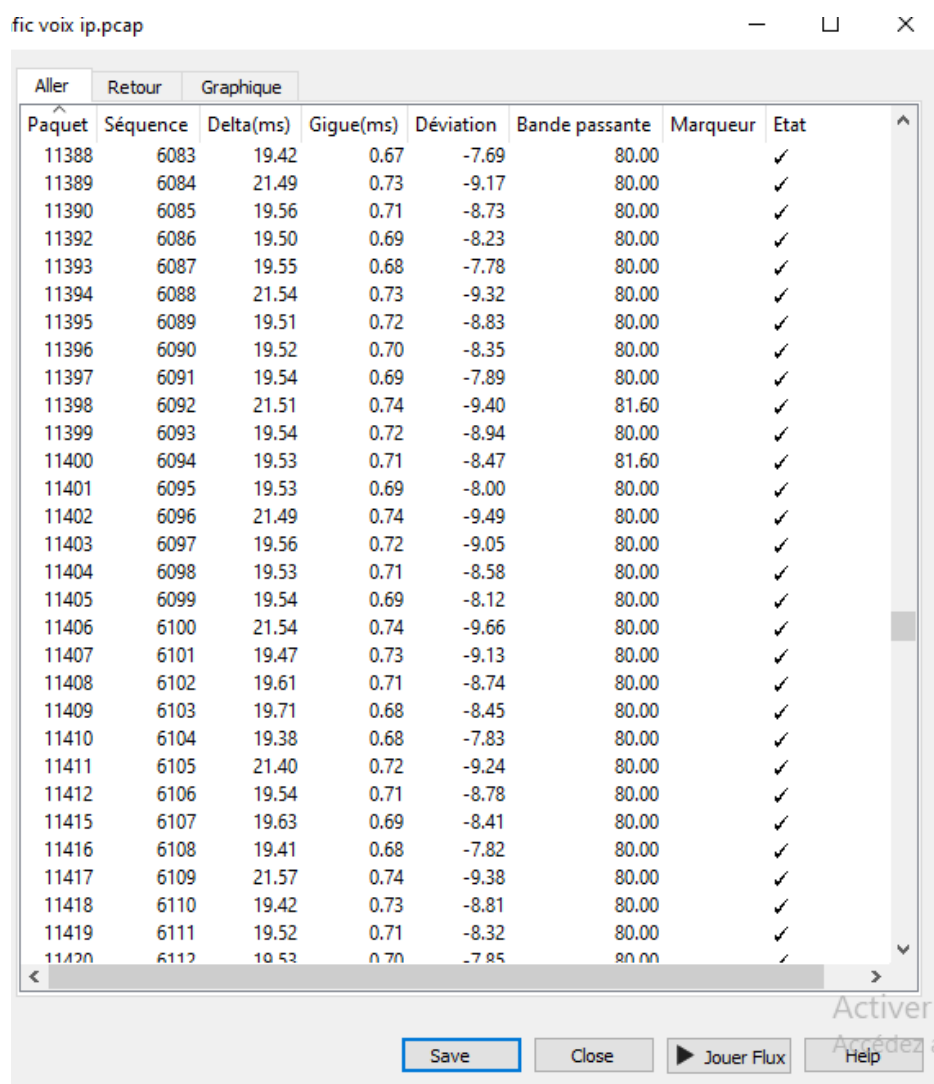


Adresse Source	Port Source	Adresse Destination	Port Destination	SSRC	Payload	Paquets	Perdu	Delta Max (ms)	Gigue Max	Gigue Moyenne	Etat
192.168.1.25	19354	192.168.1.115	5602	0x5025d87f	g711u	173	0 (0.0%)	31.481	8.617	5.541	
192.168.1.25	12998	192.168.1.115	49486	0x609117e2	g711u	1	0 (0.0%)	0.000	0.000	0.000	
192.168.1.115	5602	192.168.1.125	44382	0x1ebd6981	g711u	4022	0 (0.0%)	32.088	8.615	2.230	

Pour plus de détails et pour chaque paquet RTP, on peut visualiser les paramètres suivants :

- Le délai en ms
- La gigue en ms
- La variation de délai  $D(i, i+1)$
- La bande passante

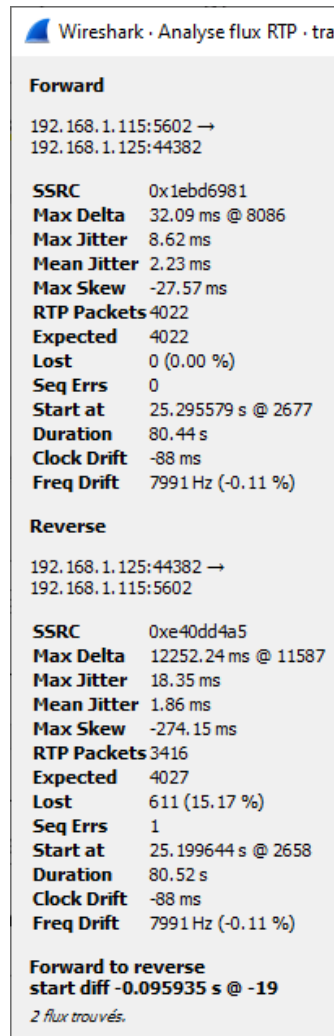
On sélectionne un paquet RTP puis Téléphonie/RTP/Analyse de Flux/



Paquet	Séquence	Delta(ms)	Gigue(ms)	Déviaton	Bande passante	Marqueur	Etat
11388	6083	19.42	0.67	-7.69	80.00		✓
11389	6084	21.49	0.73	-9.17	80.00		✓
11390	6085	19.56	0.71	-8.73	80.00		✓
11392	6086	19.50	0.69	-8.23	80.00		✓
11393	6087	19.55	0.68	-7.78	80.00		✓
11394	6088	21.54	0.73	-9.32	80.00		✓
11395	6089	19.51	0.72	-8.83	80.00		✓
11396	6090	19.52	0.70	-8.35	80.00		✓
11397	6091	19.54	0.69	-7.89	80.00		✓
11398	6092	21.51	0.74	-9.40	81.60		✓
11399	6093	19.54	0.72	-8.94	80.00		✓
11400	6094	19.53	0.71	-8.47	81.60		✓
11401	6095	19.53	0.69	-8.00	80.00		✓
11402	6096	21.49	0.74	-9.49	80.00		✓
11403	6097	19.56	0.72	-9.05	80.00		✓
11404	6098	19.53	0.71	-8.58	80.00		✓
11405	6099	19.54	0.69	-8.12	80.00		✓
11406	6100	21.54	0.74	-9.66	80.00		✓
11407	6101	19.47	0.73	-9.13	80.00		✓
11408	6102	19.61	0.71	-8.74	80.00		✓
11409	6103	19.71	0.68	-8.45	80.00		✓
11410	6104	19.38	0.68	-7.83	80.00		✓
11411	6105	21.40	0.72	-9.24	80.00		✓
11412	6106	19.54	0.71	-8.78	80.00		✓
11415	6107	19.63	0.69	-8.41	80.00		✓
11416	6108	19.41	0.68	-7.82	80.00		✓
11417	6109	21.57	0.74	-9.38	80.00		✓
11418	6110	19.42	0.73	-8.81	80.00		✓
11419	6111	19.52	0.71	-8.32	80.00		✓
11420	6112	19.53	0.70	-7.85	80.00		✓

On peut aussi visualiser ces paramètres pour cette session et dans chaque direction

On sélectionne un paquet RTP puis Téléphonie/RTP/Analyse de Flux/



Wireshark · Analyser flux RTP · tra

**Forward**

192.168.1.115:5602 →  
192.168.1.125:44382

**SSRC** 0x1ebd6981  
**Max Delta** 32.09 ms @ 8086  
**Max Jitter** 8.62 ms  
**Mean Jitter** 2.23 ms  
**Max Skew** -27.57 ms  
**RTP Packets** 4022  
**Expected** 4022  
**Lost** 0 (0.00 %)  
**Seq Errs** 0  
**Start at** 25.295579 s @ 2677  
**Duration** 80.44 s  
**Clock Drift** -88 ms  
**Freq Drift** 7991 Hz (-0.11 %)

**Reverse**

192.168.1.125:44382 →  
192.168.1.115:5602

**SSRC** 0xe40dd4a5  
**Max Delta** 12252.24 ms @ 11587  
**Max Jitter** 18.35 ms  
**Mean Jitter** 1.86 ms  
**Max Skew** -274.15 ms  
**RTP Packets** 3416  
**Expected** 4027  
**Lost** 611 (15.17 %)  
**Seq Errs** 1  
**Start at** 25.199644 s @ 2658  
**Duration** 80.52 s  
**Clock Drift** -88 ms  
**Freq Drift** 7991 Hz (-0.11 %)

**Forward to reverse**  
**start diff** -0.095935 s @ -19

2 flux trouvés.

Le protocole RTCP est utilisé pour envoyer des rapports (SR : Sender Report et RR : Receiver Report) sur les paramètres de la qualité de transmission

En sélectionnant un paquet RTCP (SR ou RR), on peut visualiser les paramètres comme :

- Fraction lost
- Cumulative number of
- Interarrival Jitter
- Delay since last SR
- Etc.



14390	92.317132	192.168.1.115	192.168.1.125	RTCP	242	Sender Report	Sou
14455	92.633973	192.168.1.125	192.168.1.115	RTCP	242	Sender Report	Sou

< >

> Frame 14390: 242 bytes on wire (1936 bits), 242 bytes captured (1936 bits)  
 > Ethernet II, Src: SURECOMT\_b9:25:c8 (00:02:44:b9:25:c8), Dst: SURECOMT\_b2:0d:48 (00:02:44:b2:0d:48)  
 > Internet Protocol Version 4, Src: 192.168.1.115, Dst: 192.168.1.125  
 > User Datagram Protocol, Src Port: 5603, Dst Port: 44383  
 > Real-time Transport Control Protocol (Sender Report)

> [Stream setup by SDP (frame 2676)]  
 10.. .... = Version: RFC 1889 Version (2)  
 ..0. .... = Padding: False  
 ...0 0010 = Reception report count: 2  
 Packet type: Sender Report (200)  
 Length: 18 (76 bytes)  
 Sender SSRC: 0x1ebd6981 (515729793)  
 Timestamp, MSW: 3496465339 (0xd067d3bb)  
 Timestamp, LSW: 871878361 (0x33f7ced9)  
 [MSW and LSW as NTP timestamp: Oct 19, 2010 08:22:19.202999999 UTC]  
 RTP timestamp: 2754840  
 Sender's packet count: 3355  
 Sender's octet count: 577060

Source 1

Identifier: 0x5025d87f (1344657535)

SSRC contents

Extended highest sequence number received: 0  
 Sequence number cycles count: 0  
 Highest sequence number received: 0  
 Interarrival jitter: 0  
 Last SR timestamp: 0 (0x00000000)  
 Delay since last SR timestamp: 0 (0 milliseconds)

Source 2

Identifier: 0xe40dd4a5 (382611653)

SSRC contents

Extended highest sequence number received: 10206  
 Sequence number cycles count: 0  
 Highest sequence number received: 10206  
 Interarrival jitter: 15  
 Last SR timestamp: 3328849870 (0xc66a37ce)  
 Delay since last SR timestamp: 206962 (3157 milliseconds)

> Real-time Transport Control Protocol (Source description)

13965	90.232620	192.168.1.125	192.168.1.115	RTCP	86	Receiver Report	Sou
14104	02.300001	192.168.1.115	192.168.1.125	RTCP	218	Sender Report	Sou

< >

> Frame 13965: 86 bytes on wire (688 bits), 86 bytes captured (688 bits)  
 > Ethernet II, Src: SURECOMT\_b9:25:c3 (00:02:44:b9:25:c3), Dst: SURECOMT\_b9:25:c8 (00:02:44:b9:25:c8)  
 > Internet Protocol Version 4, Src: 192.168.1.125, Dst: 192.168.1.115  
 > User Datagram Protocol, Src Port: 19355, Dst Port: 5603  
 > Real-time Transport Control Protocol (Receiver Report)

> [Stream setup by SDP (frame 2676)]  
 10.. .... = Version: RFC 1889 Version (2)  
 ..0. .... = Padding: False  
 ...0 0001 = Reception report count: 1  
 Packet type: Receiver Report (201)  
 Length: 7 (32 bytes)  
 Sender SSRC: 0x5025d87f (1344657535)

Source 1

Identifier: 0x1ebd6981 (515729793)

SSRC contents

Fraction lost: 0 / 256  
 Cumulative number of packets lost: 0

Extended highest sequence number received: 3609  
 Sequence number cycles count: 0  
 Highest sequence number received: 3609  
 Interarrival jitter: 63  
 Last SR timestamp: 0 (0x00000000)  
 Delay since last SR timestamp: 4294237225 (65524860 milliseconds)

> Real-time Transport Control Protocol (Source description)