



Lezione n.4 del 16/03/2015



- ❑ Le informazioni contenute negli oggetti flowspec e filter spec vengono rispettivamente utilizzate per configurare i parametri di due moduli:
 - Packet scheduler
 - Packet classifier



Packet classifier

- Ha il compito di dividere i pacchetti che giungono al nodo sulla base della QoS loro assegnata, esso in pratica identifica i vari flussi di dati cui sono riservate QoS diverse e li accoda in modo opportuno.
- Il classificatore utilizzato è di tipo multi-field (MF) cioè classifica i pacchetti entranti in base ad una combinazione di campi dell'intestazione del pacchetto IP.

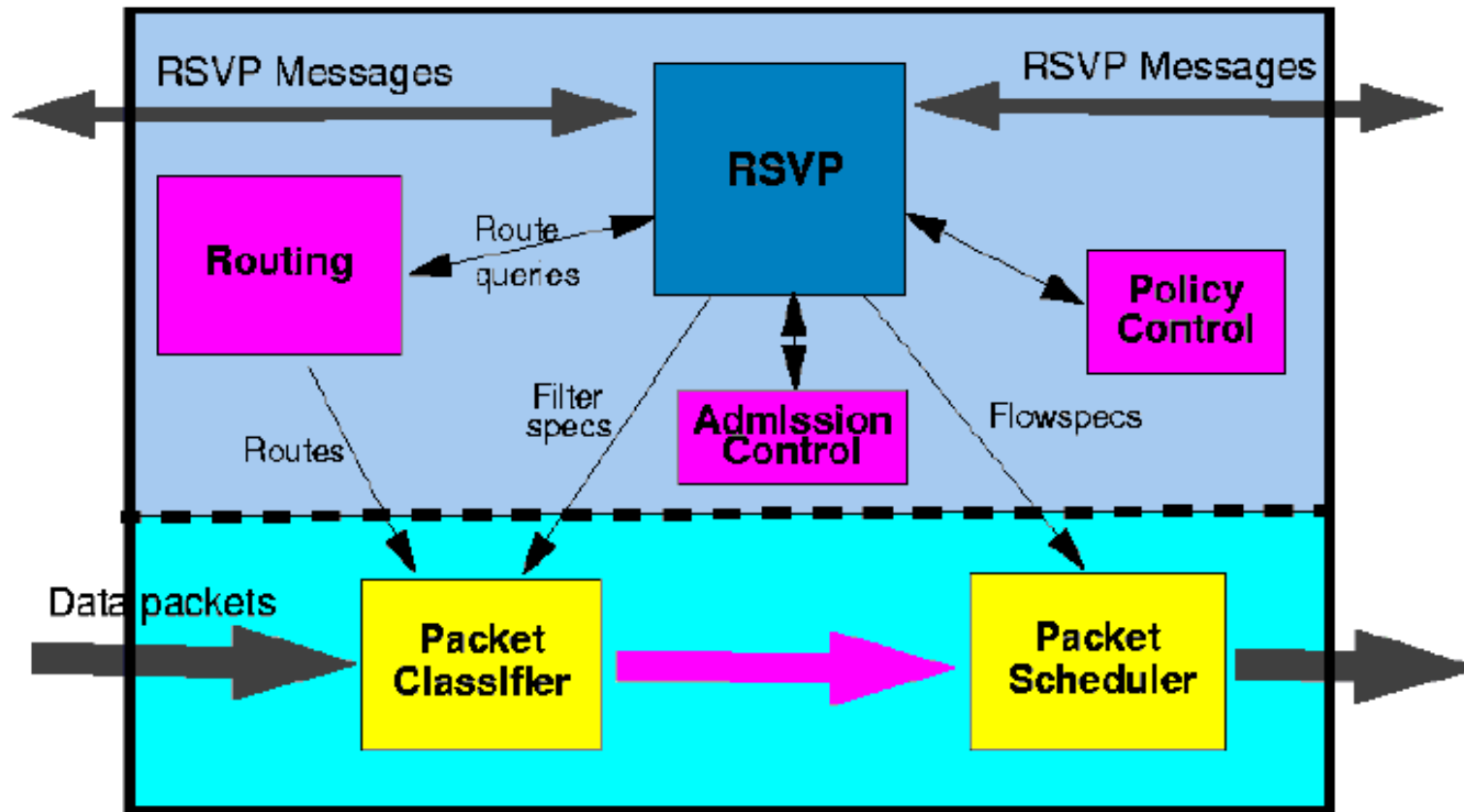


Packet scheduler

- ❑ Si occupa dell'inoltro dei pacchetti sulle interfacce uscenti dei nodi, basandosi sull'utilizzo di una o più code gestite secondo una politica prestabilita.
- ❑ Esso deve inviare i datagrammi IP sul mezzo fisico in maniera da rispettare la QoS negoziata selezionando i pacchetti delle code ed instradandoli in rete al momento opportuno.



Componenti in un nodo IP





Classificazione delle applicazioni

- La struttura degli IntServ ha classificato le applicazioni nelle seguenti categorie:
 - Elastic Traffic
 - Real Time Traffic



Elastic Traffic

- ❑ E' il traffico tradizionale delle reti TCP/IP
- ❑ È indifferente alle variazioni di ritardo di transito dei pacchetti
- ❑ Applicazioni come Telnet, FTP, Web browsing rientrano in questa categoria
- ❑ Il modello di servizio best-effort è accettabile per queste applicazioni



Real Time Traffic

- ❑ Le applicazioni real-time sono le applicazioni dette playback application, cioè applicazioni in cui in fase di ricezione è necessario riprodurre i pacchetti partiti in un istante esatto detto playback point
- ❑ Questo induce la necessità di un delay bound da non superare affinché i pacchetti ricevuti siano ancora validi.
- ❑ Necessita un trattamento preferenziale rispetto alle applicazioni elastiche



- Le applicazioni Real Time si dividono in due sottocategorie:
 - Intolleranti: che scartano completamente un pacchetto scaduto
 - Tolleranti: che riescono ad adeguarsi a variazioni del ritardo compensando con una perdita di qualità
- I parametri di QoS richiesti sono:
- Throughput
 - Delay
 - Jitter
 - Perdita di pacchetti



Protocollo RSVP

- ❑ Il protocollo RSVP (ReSerVation Protocol) permette di prenotare risorse all'interno della rete internet in modo da garantire una certa QoS a determinati flussi all'interno di una data sessione.
- ❑ Una sessione è un insieme di uno o più flussi di dati individuato da una certa destinazione ed un certo protocollo di trasporto
- ❑ Per poter realizzare i propri obiettivi il protocollo RSVP usa dei messaggi di controllo incapsulati in pacchetti IP ed instradati dai router alla stessa stregua dei normali datagrammi IP.



- ❑ È stato ideato per venire incontro alle esigenze delle applicazioni multimediali
- ❑ E permettere a tali applicazioni di poter usufruire di un servizio di trasporto dati con una certa qualità garantita e negoziata precedentemente all'invio dei dati stessi
- ❑ Quando un'applicazione su un terminale IP richiede per un flusso di dati una certa QoS il protocollo RSVP si occupa di inoltrare tale richiesta a ciascun nodo della rete IP



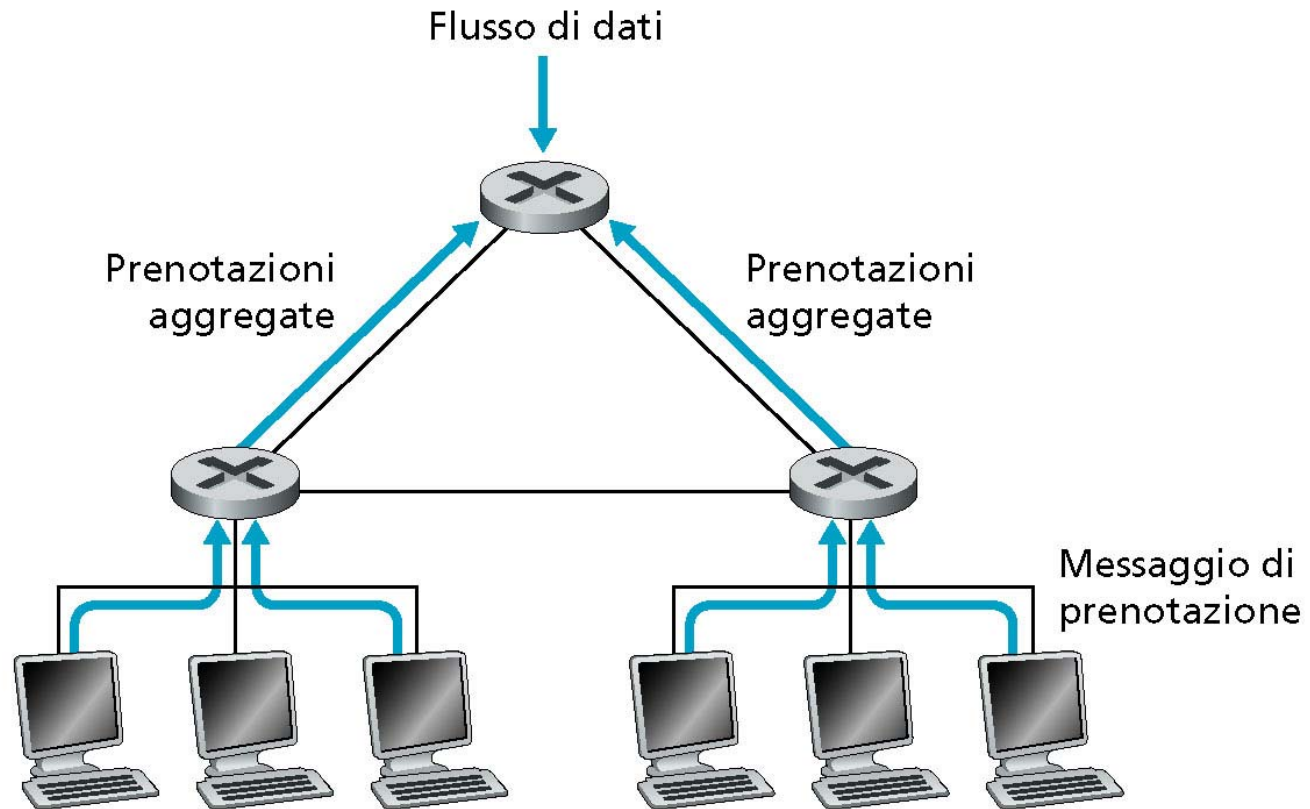
- **Il protocollo introduce quindi un meccanismo di discriminazione tra i vari flussi di dati in quanto consente di riservare a determinate applicazioni un trattamento privilegiato**



Caratteristiche del protocollo

Le caratteristiche salienti del protocollo sono:

- È un protocollo di controllo che si poggia sullo strato di rete IP
- È in grado di prenotare risorse all'interno della rete sia per trasmissioni unicast che multicast
- È orientato al ricevitore, ossia è l'host ricevente che, sulla base della QoS con cui desidera ricevere i dati, decide la quantità di risorse da prenotare
- Si adatta in maniera veloce e robusta ai cambiamenti dei percorsi





Messaggi fondamentali di RSVP

- I messaggi fondamentali utilizzati dal protocollo RSVP sono due:
 - PATH
 - RESV



Messaggio di Path

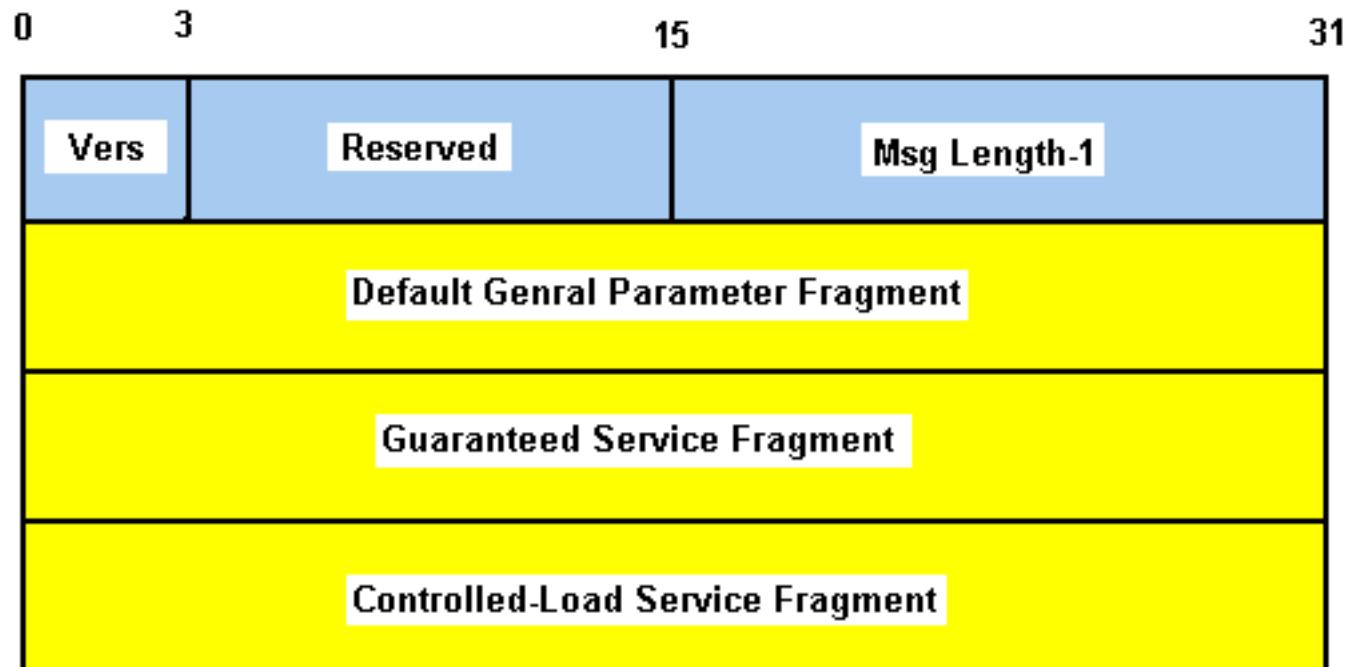
- ❑ Quando un host sorgente vuole utilizzare il protocollo RSVP, per consentire che il proprio flusso di dati venga ricevuto con Qos, invia un messaggio di Path che ha come indirizzo di destinazione l'indirizzo IP del ricevitore al quale vuole trasmettere.
- ❑ Questo messaggio viene instradato come un normale messaggio IP, ma ogni router che lo elabora deve immagazzinare alcune informazioni, dette Path State:
 - Il router deve tenere nota dell'indirizzo IP del nodo precedente, così che i messaggi di prenotazione RESV possa essere instradato correttamente
 - Il path trasporta inoltre le informazioni necessarie per l'attuazione del meccanismo di prenotazione denominato One Pass With Advertising (OPWA).



- ❑ *OPWA* si riferisce al modello di prenotazione nel caso in cui la sorgente include l'oggetto *ADSPEC* nel suo messaggio di *Path* per abilitare il ricevente a determinare il servizio end-to-end che risulterà da una data richiesta di prenotazione.
- ❑ L'*ADSPEC* è un oggetto opzionale che la sorgente può includere nel suo messaggio di *Path* per avvertire il ricevente circa le caratteristiche della comunicazione end-to-end. L'informazione può essere usata dal ricevente per determinare il livello di prenotazione richiesto per raggiungere il livello di *QoS* end-to-end desiderato.



Formato dell'oggetto ADSPEC





- ❑ L'*ADSPEC* consiste di un messaggio di intestazione, un frammento che contiene i parametri generali di default denominato *Default General Parameters fragment*, ed almeno uno dei frammenti relativi alla classe di servizio che può essere selezionata dall'applicazione ricevente, e cioè almeno uno tra *Guaranteed Services fragment* e *Controlled-Load Services fragment*. L'omissione di uno dei due frammenti relativi ai servizi è un'indicazione al ricevitore che in servizio omesso non è disponibile

- ❑ Il *Default General Parameters fragment* include i seguenti campi, che sono aggiornati ad ogni router RSVP lungo il percorso per far pervenire i valori al ricevente:
 - ❑ Minimum Path Latency (somma delle latenze individuali dei link).
 - ❑ Path bandwidth (minimo delle bande dei link individuali lungo il percorso).
 - ❑ Global break bit
 - ❑ Integrated Services hop count
 - ❑ PathMTU – unità di trasmissione massima



Messaggio di RESV

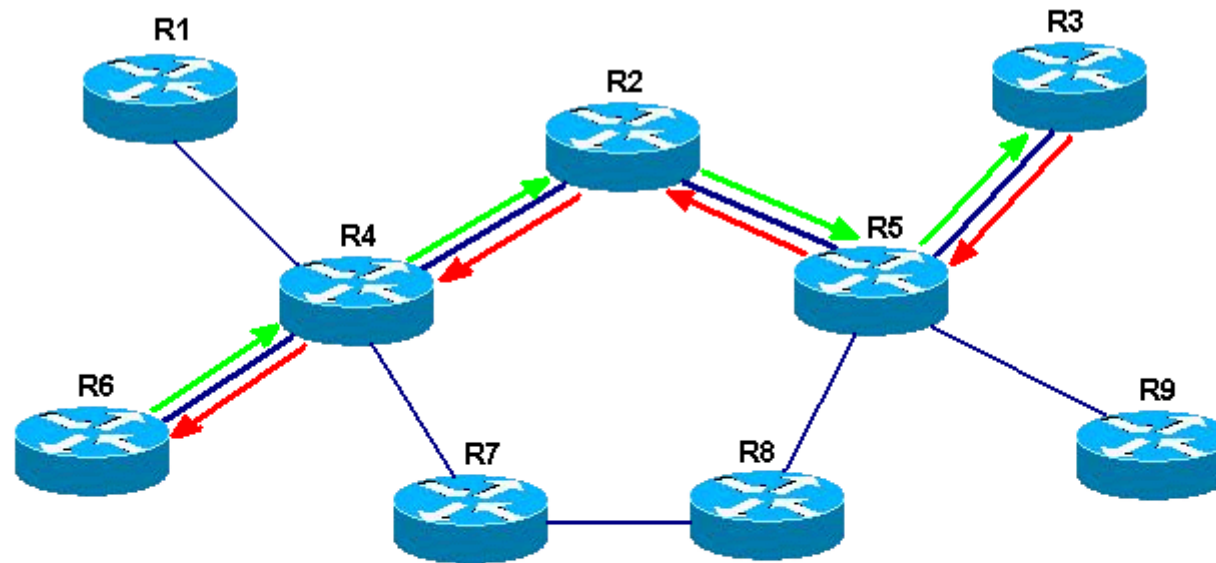
- ❑ Il messaggio Resv viene utilizzato dagli host riceventi per inoltrare le richieste di prenotazione delle risorse ai nodi che si trovano sul cammino verso la sorgente del flusso di dati oggetto della richiesta.
- ❑ Tale messaggio deve seguire "a ritroso" il cammino seguito dai dati dalla sorgente verso il ricevitore, per cui essi sono instradati hop-by-hop e recano come indirizzo di destinazione l'indirizzo del prossimo nodo presente sul cammino verso la sorgente.



- ❑ Ad ogni nodo la richiesta di prenotazione è passata ai moduli admission control e policy control
- ❑ Se uno dei due moduli non accetta la richiesta, nessuna risorsa viene allocata e viene inviato un messaggio di errore verso l'host ricevente che aveva originata la richiesta.
- ❑ Se invece la richiesta è accettata vengono intraprese due azioni:



- Attraverso le informazioni contenute nel flow descriptor vengono configurati i parametri del packet classifier e del packet scheduler ed allocate le risorse in loco
 - La richiesta viene ulteriormente propagata verso gli host sorgenti
-
- L'insieme delle informazioni necessarie in un nodo a tenere allocate determinate risorse per una certa sessione identifica il cosiddetto Resv State.



	Setup: Path (R6,R4,R2,R5,R3)
	Reply: Resv



Meccanismo di refresh

- ❑ Le risorse che vengono allocate in un nodo per servire un certo flusso di dati in seguito ad un messaggio di prenotazione Resv non sono mantenute indefinitamente, ma dopo un certo periodo di tempo (cleanup timeout) vengono liberate
- ❑ Affinché un certo host possa continuare a ricevere dati con la QoS precedentemente richiesta deve inviare allo scadere di un certo refresh timer dei messaggi di refresh in modo che le risorse per esso allocate rimangano tali.



Altri tipi di messaggi

- ❑ Esistono altri tipi di messaggi che vengono utilizzati dal protocollo di prenotazione RSVP:
 - Messaggi di Teardown: sono utilizzati per liberare delle risorse che non servono più:
 - PathTear: si propaga verso tutti gli host riceventi un dato flusso a partire dal suo punto di origine provocando in ogni nodo la rimozione del Path State, ossia il rilascio delle risorse allocate
 - ResvTear: si propaga dal suo punto di inizio verso le sorgenti di un dato flusso e provoca in ogni nodo che attraversa la rimozione delle risorse allocate per il filter spec in esso contenute.
- ❑ Una richiesta di rilascio può essere effettuata o da un'applicazione su un host o da un router a seguito dello scadere del cleanup timeout.



Formato dei messaggi RSVP

- ❑ I messaggi RSVP sono incapsulati in normali pacchetti IP con valore di protocolID pari a 46. L'intestazione comune è costituita da 8 byte.

	0	4	8	16	24	31
RSVP Header	Version	Flags	Message Type	RSVP Checksum		
	Send TTL		Reserved	RSVP Length		
Obj Head	Object Length			Class-Num	C-Type	
RSVP Object 1	Object Data					
	...					
Obj Head	Object Length			Class-Num	C-Type	
RSVP Object n	Object Data					



PDU RSVP

- ❑ Vers: identifica la versione del protocollo in uso (attualmente la 1)
- ❑ Flag: non sono stati ancora definiti
- ❑ Message type: permette di specificare il tipo di messaggio RSVP con cui si ha a che fare (Path, Resv, PathTear, etc.)
- ❑ SendTTL: contiene il valore corrispondente al campo TTL (Time to Live) dell'intestazione IP
- ❑ RSVP Checksum: contiene un codice di controllo per la correzione di errore
- ❑ RSVP Length: esprime la lunghezza totale in byte del messaggio, includendo sia intestazione comune che gli oggetti variabili



L'uso di RSVP con i Servizi Integrati

- La prima funzione nel modello dei servizi integrati è fornire servizi di QoS tramite una differenziazione del traffico diviso in classi. Il modello dei servizi integrati prevede due classi di servizio:
 - Controlled Load Services (CLS) e
 - Guaranteed Services (GS).



Controlled Load Services

- ❑ La classe Controlled Load nasce per fornire un servizio in precedenza definito Better than Best Effort
- ❑ L'idea è quella di offrire ai pacchetti di questa classe lo stesso servizio che si otterrebbe con la modalità di invio best-effort, ma in condizione di **rete scarica**, ovvero assenza di congestione, garantendo un **basso ritardo di accodamento** ed una **bassa probabilità di dropping** in seguito ad overflow dei buffer di trasmissione.



CLS

- ❑ In altre parole la sessione potrebbe considerare che una percentuale molto alta dei suoi pacchetti passerà con successo attraverso i router senza essere scartata e con ritardi di coda molto piccoli
- ❑ Cosa importante è che questa classe **non specifica cosa costituisce una percentuale molto alta di pacchetti né quale qualità del servizio approssima quella di un elemento di rete scarico.**



Guaranteed Services

- ❑ L'unica classe che permette l'ottenimento di garanzie quantificabili è la classe Guaranteed
- ❑ Essa ha lo scopo di fornire un bound rigoroso e calcolabile a priori sul massimo ritardo di accodamento dei pacchetti.
- ❑ In generale durante una trasmissione il ritardo incontrato dal pacchetto è costituito da due parti:
 - Un ritardo fisso, dipendente dal percorso: il **ritardo di propagazione**
 - Un ritardo variabile, dovuto al tempo che il pacchetto trascorre nei buffer nei nodi all'interno della rete, detto **ritardo di accodamento**.



GS

- ❑ Lo scopo della classe Guaranteed è quello di permettere la limitazione del ritardo variabile, l'unico ad essere in qualche modo controllabile dall'applicazione.
- ❑ L'informazione necessarie alle richieste di ciascuna classe di servizio, sono contenute all'interno della Flowspec che il ricevente invia verso l'host sorgente, nel messaggio Resv.
- ❑ La Flowspec, a sua volta, contiene due gruppi di informazione:
 - Tspec: La traffic Specification, ossia l'insieme dei parametri atti a caratterizzare i traffico
 - Rspec: La Request Specification, ossia l'insieme dei parametri con cui viene realizzata una richiesta esplicita di banda, per soddisfare le esigenze in termini di ritardo.



Tspec

□ La Tspec è costituita da cinque parametri:

- Token rate: r
- Token bucket size: b
- Peak rate: p
- Maximum datagram size: M
- Minimum policies unit: m

Il parametro M indica la dimensione massima dei datagrammi inviati.

Il parametro m serve ai fini della policy, per la quale i pacchetti inviati di dimensione inferiore a m verranno considerati pari ad m , per evitare problemi dovuti all'overhead sui messaggi.



Rspec

- Altri due parametri contenuti nell'Rspec sono utilizzati dalla classe Guaranteed:
 - Rate Richiesta: R
 - Slack: S con $S \geq 0$;
- Il parametro R serve ad effettuare una richiesta in termini di banda
- Il termine di correzione S indica **la differenza tra il ritardo desiderato e il ritardo ottenuto tramite l'assegnazione della banda R** ed è utilizzato dagli elementi di rete per ridurre l'assegnazione di banda al flusso.



CLS vs GS

- ❑ La classe Controlled Load fornisce una garanzia di tipo qualitativo: per essa è previsto l'invio della sola T_{spec} , che viene utilizzata per effettuare l'admission control, senza alcuna esplicita richiesta di banda.
- ❑ La classe Guaranteed è l'unica ad usare il parametro R_{spec} . Essa fornisce garanzie quantificabili: **ha lo scopo di fornire un bound matematicamente calcolabile sul massimo ritardo di accodamento.**