LIP Laboratory
ENS-Lyon - France

A Session Aware Admission Control Scheme for Next Generation IP Services

Narjess Ayari, Denis Barbaron, FT R&D – LANNION, France

Laurent Lefèvre, Pascale Primet, INRIA / LIP / Université de Lyon, France

Contact: {narjess.ayari,denis.barbaron}@orange-ftgroup.com
{laurent.lefèvre,pascale.primet}@ens-lyon.fr

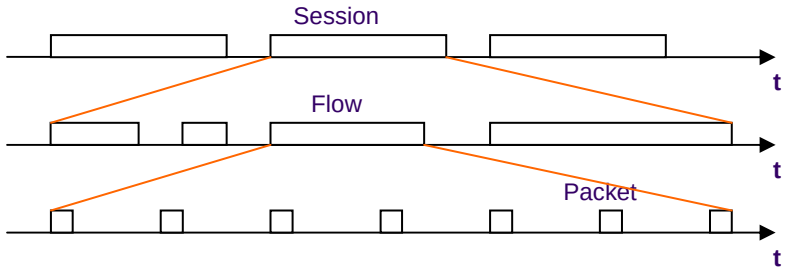
Fifth IEEE Consumer Communications & Networking Conference - CCNC 2008 - Las Vegas, Nevada, USA, January, 10th – 12th 2008.

Agenda &

- The multiple-flow based service "Dilemma"
- A fine grained scalable session identification engine
- General Architecture
- The Session Aware Admission Control Engine
- Conclusion & Future work

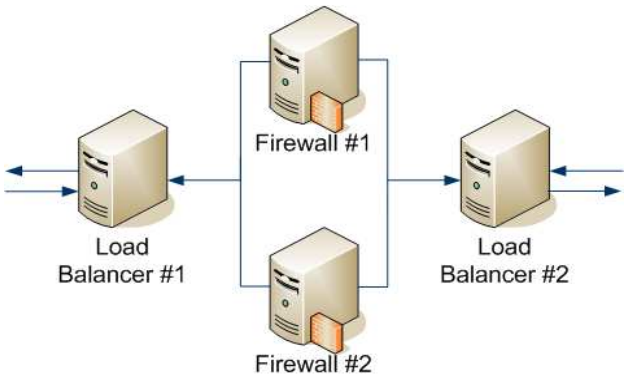
The multiple flow based service & Dilemma (1/4)

- **Some services are built upon a session model which spans over multiple flows required for**
  - The signaling
  - The data exchange and the control data exchange
- **Examples**
  - "Familiar" services
    - File transfer using FTP, etc.
  - "NGN" services
    - Media delivery (RTSP/RTCP/RTP), Voice over IP (H323/RTP/RTCP, SIP/RTP/RTCP), etc.



The multiple flow based service & Dilemma (2/4)

- **The operator's perspective focuses on**
  - Building **robust** and **scalable** IP service frameworks
    - by replacing single servers by server farms or CLUSTERS
      - *Application servers, firewalls & NAT devices.*
  - Providing a **secure** access to the operator's resources
    - by hiding critical traffic delivery/processing components behind complex NAT and firewall devices



The multiple flow based service Dilemma (3/4)

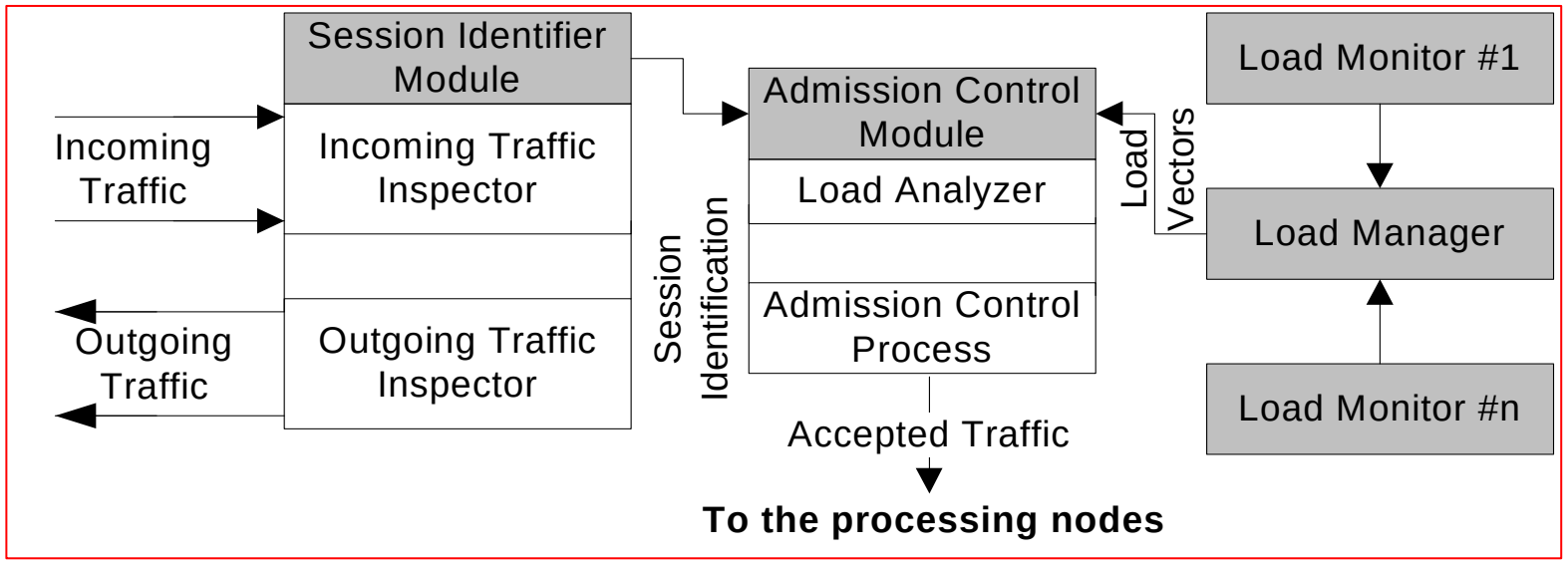
- **Flow aware processing at the front end is not enough to provide multiple flow based services with efficient processing in terms of**
  - Building of scalable frameworks by means of
    - Load balancing
      - How to achieve an improved throughput while associating the traffic pertaining to a given session to the same processing server?
    - Admission control
      - How to achieve an improved useful throughput?

The multiple flow based service Dilemma (4/4)

- **Flow aware processing at the front end is not enough to provide multiple-flow based services with efficient processing in terms of**
  - Traversal of intelligent NAT and firewall devices
    - How to scale firewalls while avoiding the interruption of the legitimate sessions?
      - The filtering "granularity" is the session rather than the flow
        - *Involved flows may use "non standard" ports, etc.*
  - Building of redundancy based highly available frameworks
    - How to avoid the interruption of the ongoing sessions during failover?

Proposed Architecture &

- **Aims to prevent the overload of the cluster resources while maximizing the operator's profitability**
  - Maximize the useful throughput in terms of completed sessions per unit of time



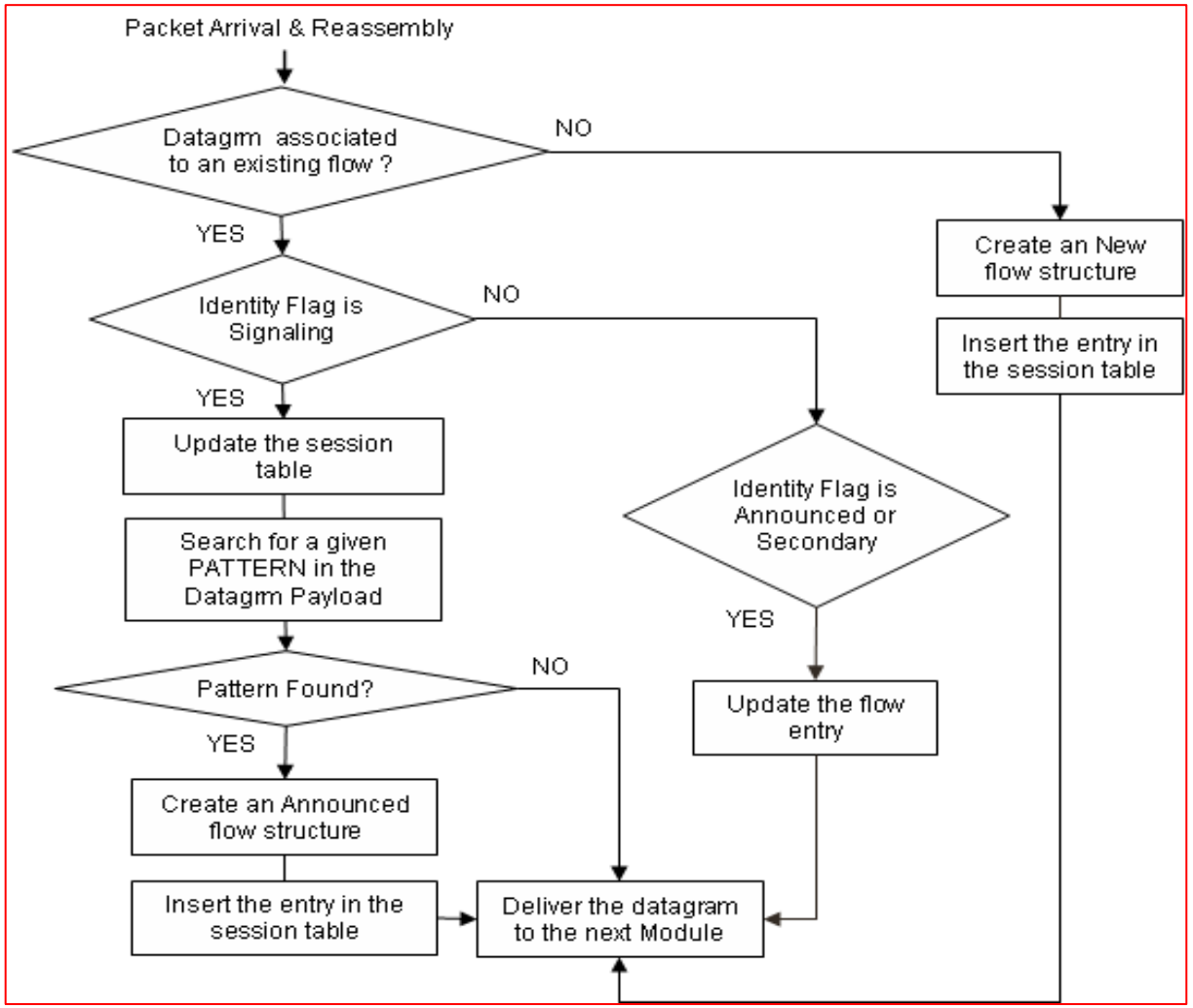
A scalable Session Identification Engine(1/4)

- **Session awareness**
  - Achieved by means of the explicit identification of the flows pertaining to a single session
- **We assume a typical multiple-flow based session model**
  - Data flow identifiers and Ctrl Data flow identifiers are negotiated over the main signalling flow
- **The session identification engine is built as a stateful engine**
  - Inspects the payload of the messages exchanged over the signalling flows
  - Maintains an in-memory session table updated either by
    - Adding new entries
    - Updating the existing ones

A scalable Session Identification Engine (2/4)

- A session is identified by the set of the associated transport level flows used for
  - The signaling
  - The data exchange
- A flow state is maintained within the session table
  - Identity flag
    - Tells whether the handled flow is a signaling flow or an expected flow
  - Status flag
    - Marks new flows, already established flows and inactive flows
  - Timestamp & Timeout
    - Used to detect the inactivity of a flow

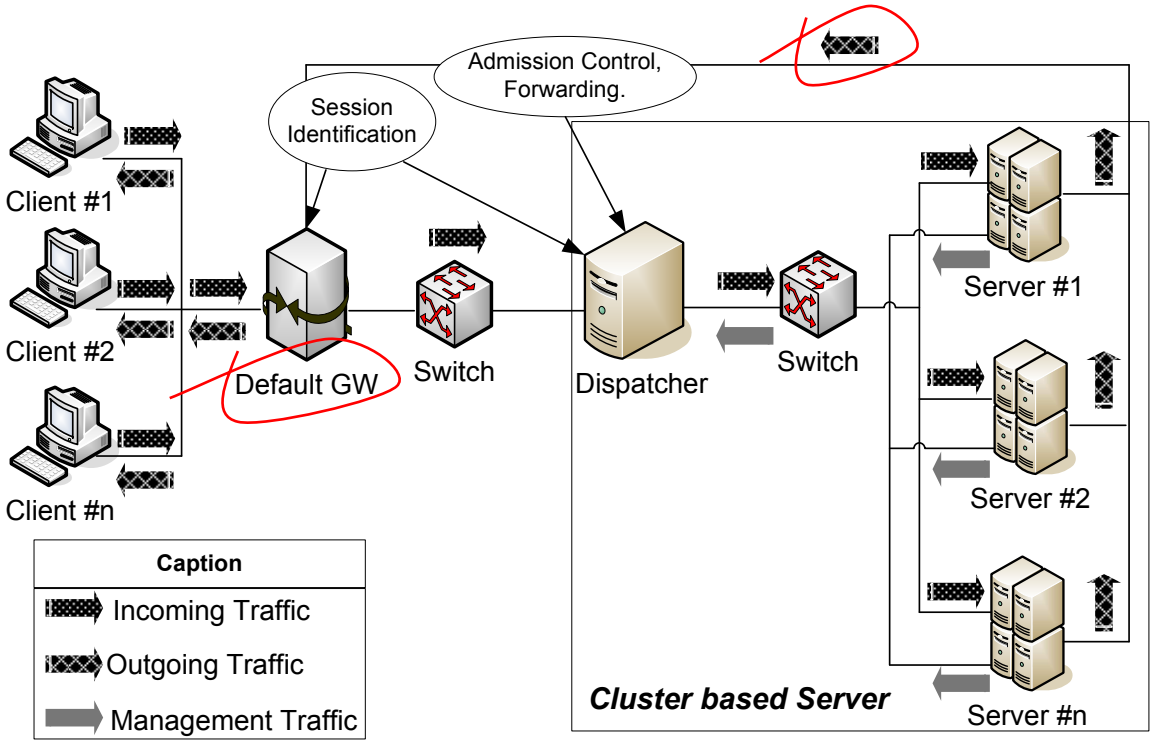
A scalable Session Identification Engine (3/4)



A scalable Session Identification & Engine (4/4)

Scalability matter

- The entry point to the cluster bottleneck issue!
  - One-way or two-way?
    - *The outgoing traffic bypasses the entry point to the cluster*



The Session Aware Admission Control & Engine *(1/6)*

- **Grant the same priority to the datagrams pertaining to the same session**
- **Under heavy load ➔ Apply a probabilistic dropping of the offered network traffic**
  - Datagrams pertaining to the already established sessions are granted a **higher priority** than those holding requests for the establishment of new sessions

The Session Aware Admission Control & Engine *(2/6)*

- **Two equations specify the dropping probability of a given offered datagram in time**
  - **(1)** A measurement based estimation of the cluster global load computed at time *Ii*

$$l_c^i := \varphi\left(l_j^i\right) = \alpha * \overline{l_j^i} + (1-\alpha) * \sigma\left(l_j^i\right)^2 \Big/ 1 \le j \le N$$

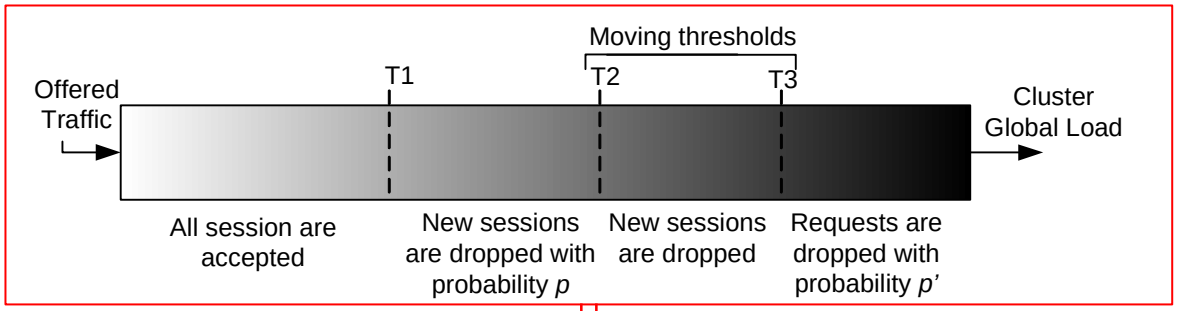
where:

* $\underline{N}$ is the number of the cluster servers, *I0 = 0* and *Ii+1 = Ii +t*,
* l_j^i is the mean load of the cluster nodes,
* $\sigma\left(l_j^i\right)^2$ is the load variance of the cluster nodes,
* α is a smoothing factor within [0,1].

  - **(2)** A drop probability ***p*** of the offered datagrams

$$p = p\left(l^i, l_c^i\right)$$

The Session Aware Admission Control & Engine *(3/6)*



$$p=\begin{cases}0 & \text{, if } \max(l^i, l_c^i) < T_1 \\ 1-f(\max(l^i, l_c^i)),\ f(x)=\left|\dfrac{x-T_1}{T_2-T_1}\right| & \text{, if \{new session\} and } T_1 \le \max(l^i, l_c^i) < T_2 \\ 1 & \text{, if \{new session\} and } T_2 \le \max(l^i, l_c^i) < T_3 \\ \max(p^u, 1-p^u),\ p^u = 1-g(\max(l^i, l_c^i)),\ g(x)=\left|\dfrac{x-T_3}{C-T_3}\right| & \text{, otherwise} \end{cases}$$

$$\begin{cases} T_2 = T_2 - \Delta\left(T_2, \max(l^i, l_c^i)\right) \\ T_3 = T_3 * \Delta\left(T_3, \max(l^i, l_c^i)\right) \end{cases}$$

The Session Aware Admission Control & Engine *(4/6)*

- **Stability versus Responsiveness?**
  - Sudden bursts of load due to short lived session are not true overload situations
    - We need sensitivity to the characteristics of the load sustained within the cluster through a short term cluster load estimation
      - *A simple forward linear regression model is applied to a T time period history load matrix*
      - *It provides the global cluster load value to be considered in equation (2)*
        - ** At the entry point to the cluster*
        - ** Inside the cluster*

The Session Aware Admission Control & Engine *(5/6)*

- **An estimation of the load of a given node *j* at *Ii+1* is calculated as follows:**

$$\hat{l}_j^{i+1} = \phi\left(l_j^k\right) \pm err_i = \sum_{k=i*T}^{k=i*T+T-1} \alpha_j^k * l_j^k \pm err_i,\ j:0..N$$

where:

- $\phi(x)$ applies a simple forward linear regression model,
- err_i is a periodically updated error used to regulate the accuracy of the prediction model. It is computed as the normalized step between an estimated value and its effective measure as follows.

$$err_i = \left|\Delta\left(\hat{l}_j^i, l_j^i\right) \Big/ l_j^i\right|$$

The Session Aware Admission Control & Engine *(6/6)*

- The error is used as a damp coefficient measuring the step between the stable and the responsive admission control decisions as follows.

$$\breve{l}_c^i := (1 - err_i) * \hat{l}_c^{i+1} + err_i * l_c^i$$

- l_c^i and l^i are substituted by $\breve{l}_c^i$ and $\breve{l}^i$ to provide adaptive and stable decisions.
- In particular, a value of err_i set to 1 defines an exclusively measurement based session aware admission control policy.

Conclusion &

- We discussed means to provide fine grained scalable session aware intelligence required for an operator to improve the QoS of services provided to end clients
  - Useful throughput in a cluster of servers
  - High availability of the ongoing sessions
- Stable and responsive admission control approach


Thanks
Any Questions?