


CATS Requirements

draft-ietf-cats-ps-usecases-req-00

K. Yao, China Mobile

D. Trossen  Huawei

M. Boucadair  Orange

Luis M. Contreras  Telefonica(Presenter)

H. Shi, Y. Li, Huawei

S. Zhang, China Unicom

What requirements have been identified so far?

- Computing-Aware Traffic Steering (CATS) is about the process of selecting service instances for directing traffic to them based on observed metrics for both computing and networking
- The following requirements emerge from that:
 - Dynamic and effective selection among multiple service instances
 - Agreement on Metric Representation
 - Moderated Metric Distribution
 - Flexible Usage of Metrics
 - Session and Service Continuity
 - Preserve Communication Confidentiality
 - Security

Requirements' details (1/4)

- **Dynamic and effective selection among multiple service instances**
 - Rationale: different service instances will be located in multiple computing sites
 - R1: MUST provide a **discovery and resolving** methodology for the **mapping of a service identifier to a specific address**.
 - R2: MUST provide a **mapping methods** for further **quickly selecting** the service instance.
- **Agreement on Metric Representation**
 - Rationale: Agree on a common representation of the metrics of relevance
 - R3: MUST **agree on** using **metrics** that are **oriented towards compute** capabilities and resources and their representation among service elements in the participating edges.
 - R4: MUST **include network metrics**

Requirements' details (2/4)

- **Moderated Metric Distribution**

- Rationale: find a scalable way of distributing (ofently changing) metrics
- R5: MUST provide **mechanisms to distribute the metrics**
- R6: MUST realize means for **rate control for distributing** of metrics

- **Flexible Usage of Metrics**

- Rationale: usage both the network and computing metrics
- R7: a **computing semantic model** SHOULD be defined for the mapping selection.
- R8: there MUST exist **flexibility** in term of **metrics definition and utilization** for the selection of service instance.
- R9: MUST set up **metric information** that can be **understood by CATS components**.
- R10: MUST **use network and computing metrics** in a flexible way that **includes a default action** for the interoperation of **network nodes which** may or may not **support the specific metrics**.

Requirements' details (3/4)

- **Session and Service Continuity**

- Rationale: possibility of changing the service instance along the service lifetime
- R11: **session as well as service continuity** MUST be maintained
- R12: MUST maintain "**instance affinity**" which MAY span one or more service requests, i.e., all the packets from the same application-level flow MUST go to the same service instance unless the original service instance is unreachable
- R13: MUST **avoid keeping fine runtime-state granularity** in network nodes for providing session and service continuity.
- R14: MUST **provide mechanisms to minimize client side states** in order to achieve the instance affinity.
- R15: SHOULD support the **UE and service instance mobility**.

Requirements' details (4/4)

- **Preserve Communication Confidentiality**

- Rationale: avoid the leakage of computing domain and application privacy
- R16: MUST preserve the **confidentiality of the communication relation between user and service** provider by minimizing the exposure of user-relevant information according to user needs.

- **Security Considerations**

- Rationale: Secure design of CATS
- R17: **service data** MUST be **protected** from interception.
- R18: the nature of **user's activities** **SHOULD be hidden** as much as possible.
- R19: **secure advertisements** are REQUIRED to prevent rogue nodes from participating in the network.

Outstanding Comments

- Some requirements are prescriptive, for example:
 - ruling out out-of-band methods with a “low latency” justification which does not clearly state why out-of-band is incompatible with low latency.
 - the “mapping of a service identifier to a specific address”.
- There are some vague expressions, for example:
 - “quickly” selecting an instance is not quantifiable so it is difficult to satisfy.
 - metrics flexibility is risky and could open up significant interoperability issues.
- Metric collection is also an important aspect that should be considered.
- Discussions on how session and service continuity are applied to different cases should be presented. Likely, service affinity is not required in all use cases.
- Use cases are too high level to derive specific requirements.
- Security considerations are too high level, and security requirements should be moved to section 5, security considerations should talk about the security issues brought in by the mechanism should talk about these security issues brought in by the mechanism
- Format issues.

Next Steps

- Address these valuable comments in the next revised version.
- Welcome more discussions and comments.