Internet Engineering Task Force
Internet Draft
draft-hamid-seamoby-ct-reqs-01.txt
Expires: September 2001

Hamid Syed
Gary Kenward
Nortel Networks

March, 2001

General Requirements for a Context Transfer Framework

Status of this Memo

Copyright Notice

Abstract

This document captures the set of general requirements for context transfer. These requirements are provided for the replication and synchronization of the context associated with a mobile node's traffic between access routers.

1 Introduction

In networks where hosts are mobile, the success of real-time sensitive services like VoIP telephony, video, etc. depends heavily on the ability of the network to support seamless handover. Ideally, seamless means that the handoff will not introduce any degradation in the quality of the service provided to the user. At the very least, the user should not perceive any degradation in service quality during handoff.

The service quality offered at an access router is embodied in the context of the support provided to the IP traffic. The ability of a new access router to support the same service quality after handoff is determined by the router's built-in capabilities, by the availability of the necessary router resources, by the availability of unused bandwidth on the links that the traffic must traverse to and from the router, and, by the timely available of the service support context at the router.

The support context referred to here is comprised of the information necessary to support the all the committed service features, such as AAA, header compression, Differentiated Services, Integrated Services, policy enforcement, etc. [2]. This context is initially established when the service is set-up between the mobile node and the network, and changes over time as the components supporting the service features change state.

In order for this context to be available at a new access router after handoff, it must be replicated from the access router currently supporting the mobile modes traffic. The replicated context must represent the most recent support state, if the service is not to be interrupted or degraded. Thus, when the mobile node's traffic arrives at a new access router, the replicated context must be synchronized with the context at the previous access router just prior to the handoff. For seamless reactive context transfer, the time scale of this synchronization is roughly on the order of the allowable incremental delay for forwarding the next packet. For proactive context transfer, the synchronization latency is on the order of the average packet inter-arrival time for the mobile node's traffic.

This document captures the requirements this context transfer in Support of seamless mobility.

2 Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [1].

3 Terminology

The terminology and the definitions used in the document are for the most part taken from [2]. This document defines additional terminology needed to explain the requirements for the transfer of context. This section presents the new general definitions.

3.1 Coverage Area (CA)

The coverage area for a given AR is defined in terms of the access points (APs) that are connected to that AR. Each AP forwards traffic between AR and a given MN.

For the purpose of describing Context Transfer, there is no need to assume a given cardinality between ARs and APs. Thus, an AP may be connected to multiple ARs, and an AR may be connected to multiple APs. Each AP must be connected to at least one AR, and each AR must be connected to at least one AP.

3.2 Forwarding Path Handover Scenarios

3.2.1 Break-before-make

Break-before-make is a term used to describe a discontinuity in connectivity between MN and the access network during a handoff. With a break-before-make handoff, the forwarding of an MN's traffic along the current path is discontinued before forwarding of that traffic is initated along the new path through the network. The old connection for the traffic flow is "broken" before the new connection is "made".

For example: an MN moves between the CAs of two ARs. In a break-before-make scenario, the MN's traffic through the old AR, and old AP, is stoppe before being redirected through the new AR/AP pair.

In break-before-make, there is exists some interval where the MN's traffic cannot be forwarded. The extent of this interval, and the impact on the IP packets (additional packet drops or buffering delay) is dependent upon the details of the break-before-make alogorithm.

This definition of break-before-make is independent of the method used for or the timing of the context transfer. The context transfer may still be "proactive" or "reactive" (c.f. below).

3.2.2 Make-before-break

Make-before-break is a term used to describe the continuity of connectivity between MN and the access network during a handoff. With a make-before-break handoff, the MN's traffic flow is established along the new path through the network, before the old path is released. The new connection for the traffic flow is "made" before the new connection is "broken".

For example, an MN moves between the CAs of two ARs. In a make-before-break scenario, the MN's traffic will be forwarded to the new AR, and the new AP, while the old AR/AP pair continues to forward traffic. In make-before-break, there is exists some interval where the MN's traffic traverses both paths. Whether these two flows contain duplicated packets is dependent upon the details of the make-before-break alogorithm.

This definition of make-before-break is independent of the method used for or the timing of the context transfer. The context transfer may still be "proactive" or "reactive" (c.f. below).

3.3 Context Transfer Scenarios

3.3.1 Mobile Arrival-Departure Event (MADE)

The MADE is an notification delivered to an AR when the MN enters its CA. Reception of a MADE indicates that connectivity exists between the AR and the MN through at least one AP.

3.3.2 Reactive Context Transfer

The context information required to completely supporting an IP micro-flow is replicated to the access router at the instant when a packet from that micro-flow arrives at the new access router.

A reactive context transfer can be performed for a make-before-break or for a break-before-make handoff.

3.3.3 Proactive Context Transfer

The context information required to completely support an IP micro-flow is replicated to the access router(s), that detect the presence of MN in its coverage area, in advance of the first packet arrival to one or any of the ARs.

A proactive context transfer can be performed for a make-before-break or for a break-before-make handoff.

4 General Requirements for a Context Transfer Framework

This section captures the general requirements for context transfer. The general requirements cover two functional areas.

- Distributed framework approach
- Context transfer mechanism

4.1 Distributed Framework Approach

An MN may have connectivity to the access network through more than one access points (AP) at one time. The determination of which APs are able to communicate with an MN is dependent entirely on the link characteristics and the layer 2 protocols and services.

The APs able to communicate with an MN may be linked with one or more ARs. In the scenario where two or more ARs are candidates for fowarding an MN's traffic, the context for the MN's active micro-flows must be replicated at every AR.

- The framework MUST support one-to-many context transfer.

To achieve seamless handover, the introduction of additional packet

delays and drops must be avoided. Context transfer will require
some exchange of information and since the context needs to be

established before an AR can provide the appropriate forwarding treatment, it is necessary to initiate the transfer well before forwarding is required to begin.

- The framework MUST support proactive context transfer.

The unpredictability of some channels, and the vagaries of layer 2 handoff mechanism ensure that a proactive approach may not always be possible. There will be situations where there is no warning, and an AR requires the context needed to forward traffic immediately.

- The framework SHOULD support reactive context transfer.

There are various alternative approaches to context transfer, some of which were reviewed in [2]. The main distinction between these alternatives begins with the choice of the functional entity or entities that orchestrate the context transfer (e.g. MN driven versus network driven, centralized versus distributed.

A single entity or centralized approach to context transfer will likely suffer from scalability difficulties as the number MN's or the rate of handovers increases. Moreover, the most current context information will only be available at the access router(s) actively supporting an MN's flows. Thus, a centralized approach will first require retrieving context from an AR before distributing it to other ARs.

- The framework MUST support a distributed transfer approach in which the access routers are responsible for transferring context.

The actual context associated with an MN reflects the service parameters that were agreed upon between the MN and the access network when each microflow was established, and the state variables for the service facilities supporting each microflow. Various protocols participate in setting up the service support for a given micro-flow, and many may require state be maintained for the duration of the session. A few examples of context types are captured in [2].

It is likely that more than one network entity will be involved in updating the context due to the interaction of the various protocols with different network services. The the most relevant instantiation of the context, however, is that which is local to the AR and maintained for the purpose of suppor supporting a microflow. A context transfer approach that uses the active AR as the source of the context, and delivers the context directly to the new AR would be the most efficient. The number of entities involved in the context transfer would simply the number of ARs requiring the context for a particular microflow. In addition, by implication, the number of protocol exchanges would be less, as the number of communicating entities is limited to those same ARs.

4.2 Context Transfer Protocol

The context transfer protocol is the mechanism for transporting context information from one AR to another AR. The outcome of a context transfer will be an up-to-date replication of the configuration and state information from the source AR at the new AR.

- The context transfer protocol MUST provide 100% reliable transfer of the context information. 100% reliable information transfer means no loss of information and no induced errors.

- The context transfer protocol MUST deliver the context without duplication or re-ordering of the information.

- The context transfer protocol MUST transfer the context fast enough for the information to be meaningful at the receiving AR.

The context at the AR actually supporting traffic from the MN will change over time. In addition to the progression of the various state information, the MN may initiate new microflow(s) or discontinue existing microflows. The timing of these changes in context is on the order of the intervals between packet arrivals in the MN's traffic flow.

- The context transfer protocol MUST provide method for synchronizing context information when it changes.

- The synchronization of context MUST preserve the integrity, and thus the meaning, of the context at each AR who has received the context.

As a corollary, any signaling exchanges required by the context transfer protocol will introduce additional delay. Protocols such as TCP [4] and COPS [5] require signalling exchanges, or "handshakes" between the communicating entities at various stages of the protocol session.

- The context transfer meachanism SHOULD minimize signaling overhead when performing an actual context transfer.

The time taken to replicate context depends greatly upon the number of packet exchanges required to complete a transfer of the context information. In many situations, such as with a reactive break-before-make scenario, the context transfer delay becomes a critical factor in determining whether the service is disrupted or not.

- The context transfer MUST complete with minimum number of protocol exchanges between the source AR and the rest of the ARs.

- The context transfer protocol MUST sustain the security of context information.

Similarly, if the context transfer protocol delivers the information in a form that requires significant processing at the AR before the context is useable - for example, if the information has to be re-ordered, then significant delay may be introduced in establishing the replicated context.

- The context transfer protocol MUST minimize any processing at the ARs.

A seamless handover of an MN's active sessions requires that there be at least one AR capable of supporting the MN's traffic. In order for the handoff to be targetted to the ARs capable of supporting the MN's traffic, each AR must be able to return the admission status of the context transfer.

- The context transfer protocol MUST provide for feedback from each candidate AR of the admission status for each context transfer attempt.

- The context transfer protocol MUST interwork with the micro-mobility mechanism [3].

In a situation where a single AR is not available to support the whole context associated with an MN's traffic, a mechanism could be provided to negotiate the handover of each of the active sessions to different ARs.

Similarly, when complete support for a particular micro-flow is not possible at any AR, it may be perferred that a degraded service be negotiated over dropping the micro-flow at the time of handoff.

- The context transfer protocol MAY provide a mechanism for negotiating partial context transfer.

- Any mechanism for partial context transfer MUST interwork with the micro-mobility mechanism [3].

5 References

[1] S. Bradner, "keywords for use in RFCs to Indicate Requirement Levels", RFC2119 (BCP), IETF, March 1997.

[2] The seamoby CT design team, "Context transfer: problem statement", draft-ietf-seamoby-context-transfer-problem-stat-00.txt.

[3] The seamoby MM design team, "Micro-mobility: problem statement", draft-ietf-seamoby-mm-problem-00.txt.

[4] "Transmission Control Protocol", RFC 793, September 1981.

[5] D.Durham et. al, "The COPS (Common open Policy Services) protocol", RFC2748, January 2000.

6 Acknowledgments

The contents of this draft are a result of the discussions within the Nortel Networks Advanced Wireless Network Technology Lab and we would like to thank all the members who contributed in these discussions.

7 Author's Address

Syed, Hamid
100-Constellation Crescent
Nepean, Ontario. K2G 6J8 Phone: 1-613-763-6553
Canada Email: hmsyed@nortelnetworks.com

Kenward, Gary
100-Constellation Crescent
Nepean, Ontario. K2G 6J8 Phone: 1-613-765-1437
Canada Email: gkenward@nortelnetworks.com

8 Full Copyright Statement