Journal of Theoretical and Applied Information Technology
15th February 2012. Vol. 36 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NEW BINDING CACHE MANAGEMENT POLICY FOR NEMO AND MIPV6

[1*] SAMER SAMI HASAN, [2] ROSILAH HASSAN, [3] FAISAL ELHADI ABDALLA

[1, 2, 3] NCT LAB, FTSM, University Kebangsaan Bangi, 43600, Malaysia

E-Mail: [1]hsamersami@sun1.ftsm.ukm.my, samerhasan@ieee.org, [2]rosilah@ftsm.ukm.my

ABSTRACT

Internet access requirements within the heterogeneous environments are increasing rapidly; especially in mobile environments, also the ubiquitous mobile users and services proliferated in a wide range, such as users in trains, planes and buses. Network Mobility (NEMO) basic support protocol is an extension from Mobile IPv6 (MIPv6) protocol stated by IETF to handle the mobility on a set of mobile nodes. But this NEMO suffers from a number of limitations like the handoff latency and the optimized path especially when number of mobile nodes increased; to reduce the over loaded signaling a new mechanism is proposed to solve the drop link problem inside the Correspondent node, due to the limited binding cache size that may be substantially insufficient to bind a new entries. This paper present the new mechanism named as Distributed Alternative Binding Cache mechanism (DABC), to reduce the packet loss in MIPv6 network environments and also reduce the signaling overheads resulted from deleted links. Network simulator NS-2 has been used to evaluate this mechanism and compare it with the original NEMO.

Keywords: *Network Mobility (NEMO), Mobile IPv6 (MIPv6), Distributed Alternative Binding Cache Mechanism (DABC), Binding Cache, NS-2.*

1. INTRODUCTION

Future generation of mobile networks gave arise to the demand for ubiquitous connectivity. The existing users expect to be connected to the Internet at anytime and from anywhere. The Internet Engineers Task Force (IETF) developed a protocol named Mobile IPv4 (MIP) [1] and for IPv6 communication environments MIPv6 [2] is developed to support fast and smooth connectivity to the mobile node. Nowadays, each Internet user may own more than one device while his movement, also having multiple interface that can be connected with each other as well as with other networks. Example includes set of devices on a vehicle connected to the Internet. IETF extends MIPv6 to design the NEMO bsp [3] to handle nodes mobility in aggregate way using responsible router as shown in (figure 1). In NEMO bsp there are four main entities defined as follows, Correspondents Node (CN), Mobile Router (MR), Home Agent (HA), and Mobile Network Node (MNN). Where the CN is any type of entities communicate with MNN. MR is a router that handles all movement transparently for all MNN underneath. HA is a router located usually in the home network of MNN that acts on behalf of the mobile node while a way from the home link. MNN described as a mobile node that's have the ability to move through different networks with seamless connectivity. When MNN leaves its home link and enter a new subnet, it notifies its home agent on its home link. After updating the HA with new address acquired from the foreign link which based on the foreign prefix and named as Care of Address (CoA), now the MNN are reachable through it's HA. In this case, network overheads and handoffs latency will be increased because insufficient route (i.e. Pinball Routing problem) [4]. IETF develop an optimization procedure to come up with a solution for this problem. Direct connection establish between the MNN and the CN.

This optimization procedure is protected by authentication procedure named as Return Routability Procedure (RRP) to make a verification of Home of Address (HoA), Care of Address (CoA), and Network Prefixes (NPFs) of the mobile entity if it's originally owned by the MNN before optimization started. At each time the MNN establish optimized connection with CN as new one or as result from dropping link by CN, RRP should be used. This will increase signaling overhead especially when we have a large number of MNNs in real system. Therefore, to reduce these signaling

Journal of Theoretical and Applied Information Technology
15th February 2012. Vol. 36 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

overheads with in large number of MNNs, a control management policy must be activated to minimize the number of dropped links by the CN. Each time connection established between the MNN→ CN, CN will cache the link with its addresses and needed mobility options in its limited Binding Cache (BC). Due to this finite size of BC, CN may choose to drop any entry exists in its BC if it's substantially insufficient in order to make space for a new entry. When entries are deleted from the CN, as a result packet loss will be increased. At that time the CN will pass a packet without destination option set and routed through HA of MNN to inform them that the CN needs a new Binding Update (BU) with its RRP. The new BU will lead to increase additional overhead and latency in delivering packets to the mobile node. Due to this problem we propose a new light cache switch mechanism to manage the limited cache size in the CN, also can be performed on any other caching entity in NEMO environments. This mechanism will reduce the deleted link in BC by using a new caching buffer hold by Mobile Agent (MA) named Distributed Alternative Binding Cache (DABC). This new DABC mechanism will reduce the packet loss and header overheads.

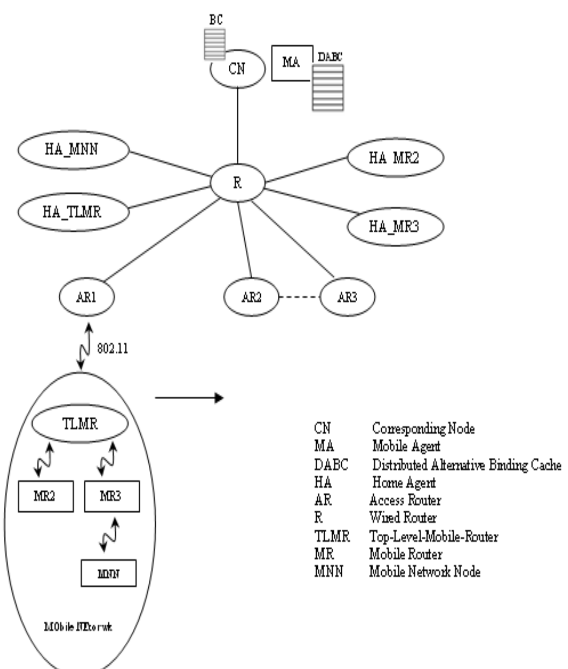

Fig.1. NEMO Architecture shows network entities that holds Binding Cache

The rest of this paper is organized as follows; in section 2, the current state of arts is described. In section 3, the new mechanism is presented. Section 4, evaluate the presented work. Finally in section 5, the conclusion of this presented work.

2. STATE OF THE ART

In the current state of art for mobility management in MIPv6 and NEMO_bsp, Imre, S., Schulcz, R., Csegedi, Cs. and Vajda, Sz., from Budapest University of technology and economics [5], studies the simulation environment of OMNeT++ within macro mobility domain in IPv6 network environments. And during there simulation test they study the effect of limited capacity of Binding Cache (BC) on the rate of packets sent via triangle routing.

Chun-Shian T., at 2011 [6], a novel mobility management scheme for network layer in NEMO environments is proposed to eliminate the pinball routing for Vehicle based on Hierarchal MIPv6 (HMIPv6), where a specific binding cache (SBC) is used during the design of the scheme to reduce signaling overheads.

3. THE NEW PROPOSED MECHANISM WITH DABC

This part proposes a new management mechanism named as Distributed Alternative Binding cache (DABC) mechanism. DABC can be located in NEMO, HA, and especially in CN. In this presented work we just use the DABC in CN, to reduce the problem of limited cache size in CN, also to eliminate the rate of registration signaling. In the proposed architecture, the CN has two binding cache. The first cache is the original Binding Cache (BC) with a finite size located inside the CN entity. And the second caching entry is the alternative binding cache generated by new Mobile Agent (MA) where this MA can be located inside the CN itself or any other entity plugged into the CN to come up from the challenge of updating the CN. Each time when the CN receive a new entry from its egress interface through Binding Update (BU) message it will be behaves as shown in the pseudo code illustrated bellow:

```
Timer initialized;
T //Threshold time sets (500 ms) to check
IF BC empty and DABC Flag ON;
if (Timer <= T) {
      if! (BU_message_lifetime == 0){
            if (lookup_entry == entry) {
                  Update_entry ();
            }//if step 5
            else
            {
```

Journal of Theoretical and Applied Information Technology
15th February 2012. Vol. 36 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

```
10                    if (BC_full) {
11                    Lifetime_LRU_entry ();
//choose the least lifetime to be inserted
into DABC and deleted from BC;
12                    send_entry_to_DABC ();
13                    DABC_flag ON
14                    Timer 0;
15                    Insert_entry_BC ();
16                    }//if step 10
17                   else
18                    {
19                      Insert_entry_BC ();
20                      }//if step 18
21              }//if step 9
22        }//if step 4
23        else
24        {
25        delete_entry_BC ();
26        if (DABC_flag == ON) {
27                     Pick_first_entry ();
28                     delete_entry_DABC ();
29                   if (DABC.nextelement ()
        == null) DABC_flag OFF;
30                  send_entry_to_BC ();
31                 Timer 0;
32        }//if step 26
33        else
34        {
35              Timer 0;
36              return to step 3
37        }//if step 34
38}//if step 24
39}//if step 3
```

As shown from above, DABC cache entry is used to replace an entry from original Binding Cache (BC) if it's substantially insufficient to bind more entries. Where the DABC structure is dynamic buffer controlled by Mobile Agent (MA) and synchronized with original BC. When a CN lookup for entries they use the new modified lookup procedure if the DABC_flag set ON. The modified procedure will check both the BC and DABC to process these entries. In DABC mechanism the expiration of entries life time was considered as shown in (figure 2). Where in the case of the BC is not full or some of its entries expired duo to the expiration of the life time, and at the same time the DABC having entries, the CN will depend on threshold time to replace the entries from the DABC cache to the empty original BC. This mechanism works as a simple method to reduce the number of dropped links between CN and mobile entities.

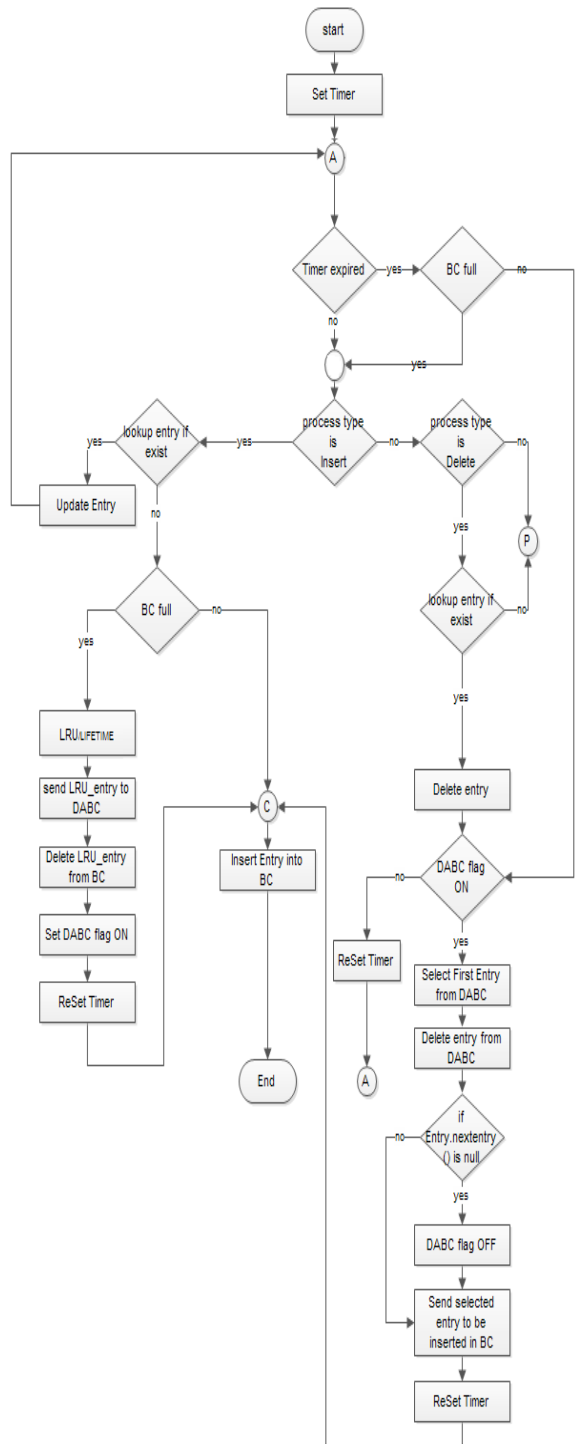

Fig.2. The Flow Diagram of DABC mechanism

Journal of Theoretical and Applied Information Technology
15th February 2012. Vol. 36 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. EVALUATION

4.1. SYSTEM ANALYSIS USING QUALITATIVE METRICS

In the following, DABC is evaluated using the following metrics:

1. Degree of Deploybility: Deploybility is the new entities added to the base MIPv6 infrastructure. In DABC no additional entities added to the infrastructure, therefore it's easy to be deployed.
2. Transparency: Where the MNN still hidden to the CN by using the NEMO features during its movement with an optimized link.
3. Signaling: DABC will eliminate signaling that could be happen when CN drop mobile node link from its BC. If drops happened Re-registration must be applied if needed, this will leads to non scalable system in real environments.
4. End-to-End delay: the MIPv6 or NEMO_bsp traditional caching mechanism will increase the end-to-end delay, when CN drops an entry from its BC and the mobile node sends its packet designated to this CN, the last one will ignore the packets and send packets through the home agent of mobile node to inform them that the mobile node entry was dropped. While in DABC this problem is solved by preventing drop of mobile nodes link, to reduce the tunneling delay and continue using optimized link till expiration time.
5. Packet loss: in DABC packet loss will be eliminated. When MN entity dropped from CN they still expect that its have an entry in CN, they will sends packets distinated to the CN and it will be dropped by them causing packets loss.

4.2.SIMULATION TEST RESULT AND DISCUSSION

The simulation environments are used to implement the proposed DABC and it will give the answer of how this mechanism behaves. The simulation environments are used under simulator NS-2 [7] and over the MobiWan extension for NEMO [8] to implement the DABC under limited binding cache size assumption. The version of NS-2 simulator that is used to test this work is NS-2.28 under windows machine (hardware: Pentium-IV 2.00 GHz CPU; operating system: Windows XP, Cygwin; Software: Ns-2.28, eclipse; and Programming language: C++, Otcl, awk). Network environments that are used in this script have a total area (1600x800) and with different number of mobile nodes distributed through the network area. These mobile nodes are connected to the Mobile Routers (MRs). Also the CN is connected to the wired Router (R) with address (0.0.0). The DABC cache list is attached directly to the CN through the Mobile Agent (MA) which is created inside the CN.

After running the old extension of NEMO with traditional caching mechanism and the new extension with the new proposed mechanism, two trace files are generated and by comparing these two files we have got the following results through using the awk files. As shown in (figure 3) the impact of binding cache size on the packet loss ratio for both NEMO bsp with traditional caching and the NEMO with DABC is illustrated. Where the new DABC mechanism reduces the packet loss compared with the original extension, and Binding cache size not mainly affect the DABC mechanism within fixed number of mobile nodes.

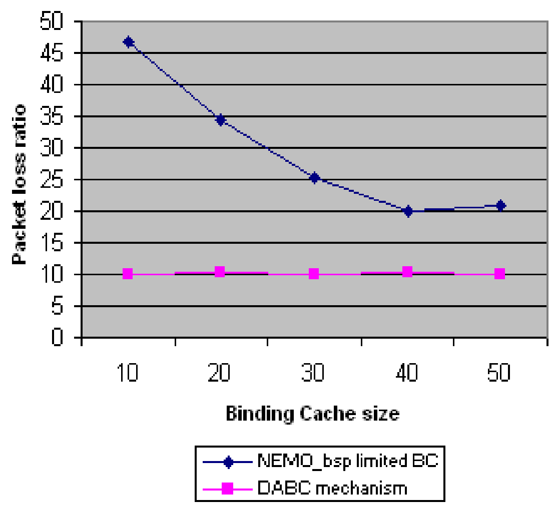

Fig.3.Compare packet loss ratio with variant BC size

5. CONCLUSION

In this paper we have proposed an efficient management control mechanism which is along with features of MIPv6 to handle the limited

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

caching size. These limitations will increase the dropping links between the corresponding node and mobile nodes. In this proposed work the mobile node will be registered by the corresponding node although if it's not sophisticated enough to handle binding new entries by using this DABC mechanism. This management mechanism is based on the size of binding cache along with the number of mobile entities. By simulation results, we have shown that, the handoff delay and packet loss are reduced in our scheme, when compared with the standard NEMO based handoff. As a future work, a new prioritizing mechanism depends on QoS can be used to improve the selections of which link entry will be replaced with the DABC.

REFERENCES:

[1] C. Perkins, (2002). "IP mobility support for IPv4". RFC 3220.

[2] Johnson, D., Perkins, C. and J. Arkko,(2004). "Mobility Support in IPv6". RFC 3775.

[3] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, (2005). "NEtwork MObility (NEMO) basic support protocol". RFC 3963.

[4] Hassan, S.S. and Hassan, R. 2011. IPv6 network mobility route optimization survey. Am. J. Applied Sci., 8: 579-583.DOI: 10.3844/ajassp.2011.579.583, URL: http://www.thescipub.com/abstract/10.3844/ajassp.2011.579.583.

[5] Imre, S., Schulcz, R., Csegedi, Cs. and Vajda, Sz., (2002). IPv6 Macro Mobility Simulation Using OMNeT++ Environment", International Seminar on Telecommunication Networks and Telegraphic Theory, organized by IEEE. 2002.

[6] Chun-Shian Tsai, (2011). "An Enhancement of Binding Update for Vehicular Hmipv6 in Wireless Network Mobility Environment". Pioneer journal computer science and Engineering Technology PJCSET.

[7] surfed at 2011, http://www.arcst.whu.edu.cn/web_kongrs/nemo_sim.htm .

[8] MobiWan: "Ns-2 extensions to study mobility in wide-area IPv6 Networks". Available at http://www.inrialpes.fr/planete/manobiw/. 2002.