Migrating Optimization Scheme for CDN based on PMIP-DMM

Shen YAN

2013/7/31
Background

- PMIPv6 based DMM
- CDN-Like service
- Handover between IDCs
- Routing Optimization
Distributed Mobility Management

- Push the anchor to the very edge of network
- Forwarding and Signaling separated
- Centralized Management (PMIP)

- You can find more information on [http://tools.ietf.org/wg/dmm/](http://tools.ietf.org/wg/dmm/)
Why that? When DMM+CDN

• DMM makes Anchor moved

• Target CHANGED
  – Can be changed to better one
Keys

- Make it support mobility
- Redirect when roaming from IDC#1 to IDC#2 (e.g. Flow#3 → Flow#4)
Solution

<table>
<thead>
<tr>
<th>MN</th>
<th>vIP</th>
<th>IP_MAG</th>
<th>IP_Mirror</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>vIP1</td>
<td>G21</td>
<td>M2</td>
</tr>
<tr>
<td>Ux</td>
<td>vIPx</td>
<td>Gxx</td>
<td>Mx</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MN</th>
<th>vIP</th>
<th>IP_NAT</th>
<th>IP_Mirror</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>vIP1</td>
<td>N2</td>
<td>M2</td>
</tr>
<tr>
<td>Ux</td>
<td>vIPx</td>
<td>Nxx</td>
<td>Mx</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MN</th>
<th>vIP</th>
<th>IP_NAT</th>
<th>IP_Mirror</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>vIP1</td>
<td>N1</td>
<td>M1</td>
</tr>
<tr>
<td>Ux</td>
<td>vIPx</td>
<td>Nxx</td>
<td>Mx</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MN</th>
<th>vIP</th>
<th>IP_NAT</th>
<th>IP_Mirror</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>vIP1</td>
<td>N1</td>
<td>M1</td>
</tr>
<tr>
<td>Ux</td>
<td>vIPx</td>
<td>Nxx</td>
<td>Mx</td>
</tr>
</tbody>
</table>

CDN Server (GSLB)

CDN Mirror #2

CDN Mirror #1

MAG#21

MAG#12

MAG#11

LMA

User #1

User #1

User #1

User #1

MN vIP IP_NAT IP_Mirror
U1 vIP1 N2 M2
Ux vIPx Nxx Mx
-- -- -- --

MN vIP IP_NAT IP_Mirror
U1 vIP1 N1 M1
Ux vIPx Nxx Mx
-- -- -- --

MN vIP IP_NAT IP_Mirror
U1 vIP1 N1 M1
Ux vIPx Nxx Mx
-- -- -- --
• Session Initiation

1. Roaming and register
2. Request
3. Send the info of CDN Mirror #1, NAT#1, vIP and User#1 to LMA
4. Configure the table in MAG#11 and the ACK back
5. Ack with the IP of MAG#11
6. Configure the table in NAT#1 and the ACK back
7. Return the vIP
8. Request to vIP
9. Encapsulate the packet and forward to NAT#1
10. De-encapsulate and translate vIP to IP of Mirror#1 and forward to Mirror#1
11. Send the data back
12. Translate IP of Mirror#1 to vIP and encapsulate the packet and forward to MAG#11
13. De-encapsulate and forward
• Handover between MAGs

1. Roaming and register
2. DMM handover ensure the continuity of data
3. request the entry of User#1
4. Reply entry
5. Configure the table in MAG#12 and the ACK back
6. Notice the change of User#1 with the IP of MAG#12
7. Configure NAT#1 and the ACK back
8. NAT#1 redirects the tunnel to MAG#12
Signaling Flow 3/3

• Handover between Mirrors

1. Notice the change of User#1 with the IP of MAG#21
2. Request the entry of User#1
3. Reply the entry of User#1
4. Configure NAT#2 and the ACK back
5. Request the data and state of User#1
6. Reply
7. Configure the data transmission and the ACK back
8. Notice the change of User#1 with the IP of NAT#2
9. Configure MAG#21 and the ACK back
10. ACK
11. Delete the entry on NAT#1 and shut down CDN Mirror#2
12. Tunnel is redirected and transmission keeps continuity
Dilemma

• How to *wake up* Mirror#2?
  – UDP: Seems easy
  – TCP: three-way handshake ONLY
  – APP: URL

• Solution:
  – Modify TCP
  – Cheat
  – Agent style
Migratory TCP (M-TCP) 1/2

• F. Sultan, K. Srinivasan, and L. Iftode, "Transport layer support for highly-available network services." p. 182.


Figure 1. Migration mechanism in M-TCP. Connection $C_{id}$, initially established by client $C$ with server $S_1$, migrates to alternate server $S_2$. 
Migratory TCP (M-TCP) 2/2

Migratory TCP (version YS)
Demo

- http://niclab.bupt.edu.cn/%E6%88%90%E6%9E%9C%E5%B1%95%E7%A4%BA/sdn-and-dmm/

- Welcome to http://niclab.bupt.edu.cn/
• Thanks~