Domain-based distributed identifier-locator mapping management in Internet-of-Things networks

Nak-Jung Choi | Moneeb Gohar | Seok-Joo Koh

1 School of Computer Science and Engineering, Kyungpook National University, Daegu, Republic of Korea
2 Department of Computer Science, Bahria University, Islamabad, Pakistan

Correspondence
Seok-Joo Koh, School of Computer Science and Engineering, Kyungpook National University, Daegu, Republic of Korea.
Email: sjkoh@knu.ac.kr

Funding information
Ministry of Education, Grant/Award Number: NRF-2017R1D1A3B03032156;
Ministry of Science and ICT, Grant/Award Number: 2015-0-00912

Summary
Recently, a lot of studies on identifier-locator mapping management for mobility control have been proposed. However, these studies did not consider the Internet-of-Things (IoT) environment. Thus, from the viewpoint of identifier-locator mapping management for IoT devices, the existing schemes are still subject to scalability problems and performance degradation. In this paper, we propose a domain-based identifier-locator mapping management for IoT devices in a distributed manner, in which an identifier of IoT device represents the domain to which it belongs. Each domain gateway maintains the mapping information of identifiers and locators based on the domain information contained in the identifier. From the numerical analysis and testbed experimentation, we see that the proposed domain-based distributed identifier-locator mapping management scheme provides better performance than the existing schemes in terms of total delay, including registration delay and data delivery delay.

1 | INTRODUCTION

Wireless networks are the biggest components of network industry where computer networking is done by using wireless connections by connecting network nodes. The mobile phones now become the basic needs, and the Internet Protocol (IP) is used when mobile devices move from one network to another. Nowadays, mobile users are not limited to one network device, and the communication method will be extended to the machine-to-machine, human-to-machine, and human-to-human communications. This major penetration introduced the idea of Internet of Things (IoT)\(^1\). Internet of Things serves in many fields, such as medical sciences, electronic engineering, and mechanical engineering. Internet of Things also plays a major role in the field of health care. With this trend, the mobility management is one of the key requirements for future IP-based mobile networks. To provide effective mobility support, a number of studies have recently been conducted on identifier-locator (ID-LOC) mapping management, using the ID-LOC separation principle. Examples of such works include the host identity protocol\(^2\) and locator-identifier separation protocol\(^3\).

Host identity protocol and locator-identifier separation protocol can be viewed as centralized mobility schemes in which a central server performs the ID-LOC mapping operation. For more effective ID-LOC mapping management, a variety of distributed mapping management (DMM) schemes have been studied\(^4-6\), which include the hash-based scheme using the distributed hash table (DHT). However, the existing hash-based scheme may still suffer from performance degradation, because it uses DHT for ID-LOC mapping resolution\(^7-9\).

In the IoT environment, various sensor devices are connected to the Internet\(^10\). These sensors require an ID for data transmission. We cannot control the IoT devices, if we do not have an ID of the sensors\(^11\). In this paper, we propose a domain-based DMM scheme, where the ID of the IoT devices represents the domain to which the mobile node belongs.
The ID-LOC mapping management is carried out in a distributed manner at each domain gateway (GW), which is based on the domain information contained in the identifier.

This paper is organized as follows. Section 2 describes the existing centralized and hash-based mapping management schemes. Section 3 presents the proposed domain-based ID-LOC mapping management scheme. In Section 4, we compare the proposed scheme and the existing schemes by performance analysis and testbed experimentation. Section 5 concludes this paper.

2 | EXISTING MAPPING MANAGEMENT SCHEMES

Existing mapping management schemes using ID-LOC separation can be classified as a centralized mapping management (CMM) or hash-based distributed mapping management (H-DMM) schemes.

2.1 | Centralized mapping management (CMM)

In CMM, both the mapping management operations are performed on a central server, such as a rendezvous or map server. Initially, the IoT devices are registered on the central mapping server (CMS) via a GW. When a correspondent IoT device or sensor wants to send a data packet, it first needs to identify the location of the target IoT device by contacting with the CMS. Then, data communication begins between the IoT devices. If the distance between CMS and IoT device increases, the total delay (TD) associated with registration may increase.

Figure 1 shows a network model for CMM. In the figure, it is assumed that IoT devices are attached to access routers (ARs). It means that networks are divided into a global Internet domain and many local domains. A central server is used in the global domain, and local domain is connected to the global domain through a GW.

Figure 2 illustrates the ID-LOC mapping management operations in CMM. In the figure, we assume that IoT devices were subscribed to different local domain. When an IoT device is connected to an AR in the local domain, the ID information of IoT device is sent to GW through the AR. Then, GW sends a map request message to central server with the ID and GW’s location. When the central server receives a map request message, it saves the ID and LOC information in the mapping table. Now, the IoT device is registered with the central sever. When a correspondent device wants to communicate with a mobile IoT device, the correspondent device will get the ID-LOC mapping information of the mobile IoT device by contacting with the central server.

Figure 3 shows the handover operations, in which IoT device moves from AR_1 to AR_2 in the GW_2 network. By handover, the IoT device transmits a handover signal to the GW_2 via AR2. The signal message will include the information on the current location (AR_2). Then, GW_2 updates the ID-LOC mapping table. After that, the data packet will be delivered to the IoT device via AR_2.

![Central Server](image.png)

**FIGURE 1** Centralized mapping management model
2.2 | Hash-based distributed mapping management (H-DMM)

In H-DMM, the mapping management operations are performed by GWs, instead of a central server, in a distributed manner. Each GW maintains the ID-LOC mapping table based on the DHT hash function with the ID of the concerned host. That is, the GW that will manage the ID-LOC mapping information of IoT devices will be determined by applying a DHT hash function to the host IDs.

When an IoT device is attached to the network, a hash algorithm is used to determine the hashed GW (HGW) that will be responsible for ID-LOC mapping management of the IoT device. When a correspondent device wants to send a data packet to the IoT device, the concerned GW looks up its DHT table to identify the HGW of the IoT device. Then, the data packet can be delivered between IoT devices via an optimized path. Depending on the used hashing mechanism and network environment, this mapping management scheme tends to induce large delay, since numerous hops between GW and HGW may be involved in the ID-LOC mapping resolution operations.

Figure 4 shows the H-DMM network model. In the figure, it is assumed that IoT devices are attached to local domains. Each IoT device has its own HGW that is determined by the used hash algorithm.

Figure 5 illustrates the H-DMM operations. When an IoT device is attached to the network, it sends a Registration REQ message (included ID information) to GW_2 through AR. If GW receives the REQ message, it finds the HGW by using the hash algorithm and sends the Map Request message to the HGW. Hashed gateway saves the ID-LOC mapping information for the IoT device, and it responds the GW_2 with Map ACK message. In turn, GW_2 responds to the IoT device with Registration ACK message.

Figure 6 shows the handover operation for H-DMM, in which IoT device moves from AR_1 to AR_2 in the GW_2 network. This handover operation is the same as that of CMM.
FIGURE 4  Hash-based distributed mapping management model

FIGURE 5  Hash-based distributed mapping management operations

FIGURE 6  Handover operation for hash-based distributed mapping management
3 | PROPOSED DOMAIN-BASED ID-LOC MAPPING MANAGEMENT

In this section, we describe the domain-based distributed ID-LOC mapping management (D-DMM) scheme, in which we assume that the ID of IoT device represents its domain information.

3.1 | ID format and network model

In the proposed scheme, we consider the 16-byte ID structure containing the home domain information of IoT device\(^{12,13}\). Figure 7 shows an example of ID format for IoT device.

We first consider the 2-byte prefix space. In this paper, we will simply use the code 2002 as the prefix code. The home domain identifier (6-byte) shows the home domain information of the IoT device: for example, 8412:3320:8080. The home domain typically represents the Internet service provider. The subscriber ID (8-byte) represents the unique value of the device within the home domain. The subscriber ID may be allocated by the home domain administrator. For example, we can consider the ID format of 2002::aaaa: aaaa: aaaa: aaaa or 2002::bbbb: bbbb: bbbb: bbbb for communication\(^{14}\).

It is noted in D-DMM that the mapping information between a domain ID and the location of the associated domain GW should be shared among all GWs in the network. Such domain ID-location mapping information may be shared by GWs through manual configurations among domain administrators or a routing protocol such as BGP. The detail of how to share such mapping information is for further study.

Figure 8 shows the network model of the proposed D-DMM scheme. It is assumed that all home domain gateways (H-DGWs) are interconnected. When an IoT device is attached to the network, the network can easily identify the H-DGW of the device from the device ID, since the device ID contains the home domain ID.

Before going into the detailed description of the D-DMM procedures, let us compare the existing and proposed mapping management schemes in the architectural perspective. Centralized mapping management uses a centralized server for ID-LOC mapping management. In H-DMM, The ID-LOC mapping management is performed in a distributed manner by using a hash algorithm. In the proposed D-DMM, the 16-byte host ID is assigned by the Internet service provider, and the ID-LOC mapping management is performed based on the domain ID in a distributed manner.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prefix(2)</td>
<td>Home Domain Identifier(6)</td>
<td>Subscriber ID(8)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FIGURE 7  ID format for D-DMM

FIGURE 8  Domain-based distributed mapping management model
3.2 | D-DMM procedures

Figure 9 illustrates the D-DMM operations when the IoT device is staying within its home domain (nonroaming case). In the figure, we assume that IoT device and correspondent device are staying in different local domains. When IoT device is attached to the network, it sends a Registration REQ message (including the device ID information) to GW (H-DGW) through AR. When GW receives the REQ message, it will realize that it is the concerned H-DGW from the ID, since the device ID includes its home domain information. Thus, it will update the ID-LOC mapping table with the information of ID and AR.

Now, let us consider the data transmission between a correspondent device and an IoT device. A data packet is delivered from the correspondent device to the GW. Then, GW will identify the H-DGW of the IoT device and perform the LOC query operation with the H-DGW. Then, the data packet is forwarded to the IoT device through H-DGW and AR.

Figure 10 illustrates the D-DMM operations in the roaming case. We assume that the IoT device is attached to a visiting domain (V-DGW). When the IoT device is attached to the network, it sends a Registration REQ message to V-DGW through AR. Because V-DGW is not the H-DGW, it will send a Map Request message to the H-DGW. Then, H-DGW updates the associated ID-LOC mapping table and responds to the V-DGW with the Map ACK message. In turn, V-DGW will respond to the IoT device with the Registration ACK message. Note that the data transmission operation is the same with that of the non-roaming case.

Figure 11 shows the handover operations for D-DMM, in which IoT device moves from AR_1 to AR_2 in the home domain. By handover, the IoT device transmits a handover signal to the H-DGW, with the current location of AR_2 and its ID information. Then, H-DGW updates its ID-LOC mapping table. After that, the data packets will be delivered to the IoT device via AR_2.

3.3 | Protocol stack

Figure 12 shows the protocol stack for D-DMM. If IoT devices are in the same domain, they will use just the local locator. However, if they are located in different domains, they should use the global locator. In this case, GW needs to perform the LOC translation between global LOC and local LOC.

4 | PERFORMANCE ANALYSIS

In this section, we analyze the TDs associated with registration and data delivery (DD). For this purpose, we define the parameters used for the numerical analysis in Table 1, by referring to 17.

We define \( T_{a-b}(S) \) as the transmission delay of a message with size \( S \) sent from \( a \) to \( b \), which is expressed as follows:

\[
T_{a-b}(S) = \frac{S}{B + L} \alpha.
\]
FIGURE 10  D-DMM operations in the roaming case

FIGURE 11  Handover operations for D-DMM

FIGURE 12  Protocol stack of D-DMM
In the equation, $\alpha$ represents the hop count between node a and b. In the analysis, the TD of each candidate scheme consists of the registration delay (RD) and DD; that is, $TD = RD + DD$.

### 4.1 Numerical analysis

In CMM, when an IoT device is attached to a new AR, the AR performs the registration with CMS, which updates its mapping table. The data packet is first delivered to GW, and then forwarded to the IoT device. The ID-LOC mapping table lookup delay is set as $\beta = \log_2 (\text{number of nodes})$. Therefore, the TD of CMM is calculated as follows.

$$RD_{CMM} = (T_{\text{IoT-AR}} + T_{\text{AR-GW}} + T_{\text{GW-CMS}})^2$$ \hspace{1cm} (2)$$

$$DD_{CMM} = T_{\text{IoT-AR}} + T_{\text{AR-GW}} + T_{\text{GW-GW}} + \beta + T_{\text{GW-AR}} + T_{\text{AR-IoT}}$$ \hspace{1cm} (3)$$

$$TD_{CMM} = RD_{CMM} + DD_{CMM}$$ \hspace{1cm} (4)$$

In H-DMM, AR performs the registration with the distributed GW. The GW finds the HGW to manage the ID-LOC mapping information of the IoT device, and the HGW updates its mapping table. When a data packet is delivered to the GW of the IoT device, the GW determines the LOC of the IoT device by contacting its HGW, and then the data packet is delivered via this path. Thus, the TD of H-DMM is obtained as follows.

$$RD_{H-DMM} = (T_{\text{IoT-AR}} + T_{\text{AR-GW}} + T_{\text{GW-HGW}})^2$$ \hspace{1cm} (5)$$

$$DD_{H-DMM} = T_{\text{IoT-AR}} + T_{\text{AR-GW}} + T_{\text{GW-HMG}} + \beta + T_{\text{HMG-GW}} + T_{\text{GW-GW}} + T_{\text{GW-AR}} + T_{\text{AR-IoT}}$$ \hspace{1cm} (6)$$

$$TD_{H-DMM} = RD_{H-DMM} + DD_{H-DMM}$$ \hspace{1cm} (7)$$

In D-DMM, when an IoT device is attached to a new AR, the AR performs the registration with H-DGW, possibly via the V-DGW, and the H-DGW updates its mapping table. When a data packet is delivered to the DGW of the corresponding IoT device, the DGW determines the LOC of the IoT device by contacting its H-DGW, and the data packet is delivered via the optimal path.

If the IoT device starts at the home domain (non-roaming case), we get the following RD and DD:

$$RD_{D-DMM} = (T_{\text{IoT-AR}} + T_{\text{AR-H-DGW}})^2$$ \hspace{1cm} (8)$$

### Table 1 Parameters used for numerical analysis

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{a,b}$</td>
<td>Packet transmission time between nodes a and b</td>
</tr>
<tr>
<td>$S_C$</td>
<td>Size of control packets (bytes)</td>
</tr>
<tr>
<td>$S_d$</td>
<td>Size of data packets (bytes)</td>
</tr>
<tr>
<td>$B_w$</td>
<td>Wired link bandwidth (Mbps)</td>
</tr>
<tr>
<td>$B_{wl}$</td>
<td>Wireless link bandwidth (Mbps)</td>
</tr>
<tr>
<td>$L_w$</td>
<td>Wired link delay (s)</td>
</tr>
<tr>
<td>$L_{wl}$</td>
<td>Wireless link delay (s)</td>
</tr>
<tr>
<td>$N_{node}$</td>
<td>Number of nodes</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Hop count between gateways</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Time to look up table (s)</td>
</tr>
</tbody>
</table>
\[ DD_{D-MM} = T_{IoT-AR} + T_{AR-H-DGW} + \beta + T_{H-DGW-DGW} + T_{DGW-AR} + T_{AR-IoT}. \] (9)

If the IoT device starts at a visiting domain (roaming case), we have the following RD and DD:

\[ RD_{D-MM} = (T_{IoT-AR} + T_{AR-DGW} + T_{DGW-H-DGW}) \star 2. \] (10)

\[ DD_{D-MM} = T_{IoT-AR} + T_{AR-DGW} + T_{DGW-H-DGW} + \beta + T_{H-DGW-DGW} + T_{DGW-AR} + T_{AR-IoT}. \] (11)

The TD of D-DMM is obtained as follows.

\[ TD_{D-MM} = RD_{D-MM} + DD_{D-MM}. \] (12)

### 4.2 Numerical results and discussion

Based on the analysis provided thus far, we compare the performances of the existing and proposed schemes. The default parameter values for numerical analysis are configured, as shown in Table 2, by referring to 17. All numerical results are obtained based on the analysis of Section 4.1.

Figure 13 provides the comparison of TDs of the candidate schemes for different hop counts between the CMS and GW, or between GWs in the network. From the figure, we can see that the CMM scheme provides the best performance for a small hop count (e.g., 1 or 2). However, as the hop count increases, the proposed D-DMM (non-roaming) scheme shows the best performance. This is because the proposed scheme can reduce the ID-LOC mapping processing delay by using the domain-based identifier. On the other hand, D-DMM (roaming) gives the same performance with H-DMM.

Figure 14 illustrates the impact of wired link delays on the TD. The 2 D-DMM schemes show better performance than the CMM and H-DMM schemes, and the nonroaming D-DMM scheme gives the best performance. Note that the performance gaps between the existing and proposed schemes get larger, as the link delay increases.

Figure 15 shows the impact of wired link bandwidth on the TD. We can again observe that the 2 D-DMM schemes provide superior performance to the CMM and H-DMM schemes, and the nonroaming D-DMM scheme gives the best performance.

Figure 16 displays the impact of the number of devices on the TD. We can also see that the proposed D-DMM scheme provides the best performance. Note that the number of devices gives no effect on TD.

Now, we analyze the performance for relatively large networks, in which the number of GWs is set to 10 000 000 and the number of nodes is set to 100 000 000 000. Figure 17 shows the TDs by handover for CMM, H-DMM, and D-DMM. We can see that CMM shows the worst performance, because CMM uses a central server. On the other hand, H-DMM and D-DMM use the distributed mapping servers, and thus, they provide relatively better performances than CMM.

### TABLE 2 Parameter values for analysis

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>( S_C )</td>
<td>200 bytes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( S_d )</td>
<td>10 000 bytes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( B_w )</td>
<td>12 500 Mbps</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( B_{wl} )</td>
<td>1 250 Mbps</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( L_{wl} )</td>
<td>0.008 s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( N_{node} )</td>
<td>100 000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( L_w )</td>
<td>0.004 s</td>
<td>0.001 s</td>
<td>0.01 s</td>
</tr>
<tr>
<td>( N_{GW} )</td>
<td>10</td>
<td>10</td>
<td>1 000</td>
</tr>
<tr>
<td>( \alpha_{(CMM)} )</td>
<td>5, 10</td>
<td></td>
<td>10</td>
</tr>
</tbody>
</table>
For applicability validation of the proposed D-DMM scheme, we perform the testbed experimentation. For this, we configured the testbed network, as shown in Figure 18. The locator is implemented by using the IPv4 address. The device ID is implemented by using the IPv6 address, which operates as 3.5 layer over IPv4.

**FIGURE 13** Impact of hop count on total delay

**FIGURE 14** Impact of link delay on total delay

**FIGURE 15** Impact of bandwidth on total delay

### 4.3 Testbed experimentation

For applicability validation of the proposed D-DMM scheme, we perform the testbed experimentation. For this, we configured the testbed network, as shown in Figure 18. The locator is implemented by using the IPv4 address. The device ID is implemented by using the IPv6 address, which operates as 3.5 layer over IPv4.
The D-DMM operations at GW can be summarized in Figure 19, which were implemented by using the Click Router software. When a packet arrives at the GW, it checks whether it is a control or data packet. If it is a control packet, the GW identifies the home GW for the IoT device, based on the device ID. Note that the home GW of IoT device can be identified from the domain ID. In case that the packet is a data packet, if the GW already knows the LOC of the

**FIGURE 16** Impact of number of devices on total delay

**FIGURE 17** Total delays by handover in large networks

**FIGURE 18** Testbed configuration for experimentation of D-DMM

The D-DMM operations at GW can be summarized in Figure 19, which were implemented by using the Click Router software. When a packet arrives at the GW, it checks whether it is a control or data packet. If it is a control packet, the GW identifies the home GW for the IoT device, based on the device ID. Note that the home GW of IoT device can be identified from the domain ID. In case that the packet is a data packet, if the GW already knows the LOC of the
corresponding IoT device, it will forward the packet directly. Otherwise, GW should first get the LOC of the IoT device by contacting with the home GW of the IoT device.

5 | CONCLUSION

In this paper, we proposed a domain-based ID-LOC mapping management for IoT network environment. In the proposed scheme, the identifier contains the domain information of IoT device for distributed mapping control. From numerical analysis, we can see that the proposed domain-based scheme provides better performance than the existing centralized and H-DMM schemes in registration and DD delays. This implies that the domain-based ID-LOC mapping management could be effectively used for mobility management in the IoT network environments with numerous devices and sensors.

For further study, the proposed D-DMM scheme needs to be tested and evaluated in a variety of network conditions, since one DMM solution is not always advantageous in all network situations. To enhance the applicability of DMM into real-world network, we also need to examine the combination of CMM, H-DMM, and D-DMM schemes for intradomain mobility and interdomain mobility scenarios.

ACKNOWLEDGEMENTS

This research was supported by the Ministry of Science and ICT under the National Program for Excellence in SW (2015-0-00912) supervised by the IITP, and also by the Ministry of Education under the Basic Science Research program (NRF-2017R1D1A3B03032156) supervised by the NRF.

ORCID

Seok-Joo Koh http://orcid.org/0000-0003-3429-2040

REFERENCES


Nak-Jung Choi received the B.S. and M.S. degrees in the School of Computer Science and Engineering from Kyungpook National University in 2011 and 2013, respectively. He has been as a doctoral student with the School of Computer Science and Engineering from Kyungpook National University. His current research interests include IP mobility, Internet of Things, and Future Internet.

Moneeb Gohar received the BS degree in Computer Science from University of Peshawar, Pakistan, and the MS degree in Technology Management from Institute of Management Sciences, Pakistan, in 2006 and 2009, respectively. He also received the PhD degree from the School of Computer Science and Engineering in the Kyungpook National University, Korea, in 2012. From September 2012 to September 2014, he worked as a postdoctoral researcher in Kyungpook National University, Korea. From September 2014 to September 2016, he worked as a foreign assistant professor with the Department of Information and Communication Engineering in the Yeungnam University. He has been as a senior assistant professor with the Department of Computer Science in the Bahria University, Islamabad, Pakistan, since September 2016. His current research interests include network layer protocols, wireless communication, mobile multicasting, wireless sensors networks, TRILL, and Internet mobility.

Seok-Joo Koh received the BS and MS degrees in Management Science from Korea Advanced Institute of Science and Technology in 1992 and 1994, respectively. He also received PhD degree in Industrial Engineering from Korea Advanced Institute of Science and Technology in 1998. From August 1998 to February 2004, he worked for Protocol Engineering Center in Electronics and Telecommunications Research Institute. He has been as a professor with the school of Computer Science and Engineering in the Kyungpook National University since March 2004. His current research interests include mobility management in Internet of Things, IP mobility, multicasting, and SCTP. He has so far participated in the international standardization as an editor in ITU-T SG13 and ISO/IEC JTC1/SC6.

How to cite this article: Choi N-J, Gohar M, Koh S-J. Domain-based distributed identifier-locator mapping management in Internet-of-Things networks. Int J Network Mgmt. 2018;e2035. https://doi.org/10.1002/nem.2035