International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.609.5

(04/2020)

SERIES X: DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

OSI networking and system aspects – Networking

Managed P2P communications: Overlay management protocol

Recommendation ITU-T X.609.5

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

PUBLIC DATA NETWORKS	
Services and facilities	X.1–X.19
Interfaces	X.20–X.49
Transmission, signalling and switching	X.50–X.89
Network aspects	X.90–X.149
Maintenance	X.150–X.179
Administrative arrangements	X.180–X.199
OPEN SYSTEMS INTERCONNECTION	
Model and notation	X.200–X.209
Service definitions	X.210–X.219
Connection-mode protocol specifications	X.220–X.229
Connectionless-mode protocol specifications	X.230–X.239
PICS proformas	X.240–X.259
Protocol Identification	X.260–X.269
Security Protocols	X.270–X.279
Layer Managed Objects	X.280–X.289
Conformance testing	X.290–X.299
INTERWORKING BETWEEN NETWORKS	
General	X.300–X.349
Satellite data transmission systems	X.350–X.369
IP-based networks	X.370–X.379
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	
Networking	**X.600–X.629**
Efficiency	X.630–X.639
Quality of service	X.640–X.649
Naming, Addressing and Registration	X.650–X.679
Abstract Syntax Notation One (ASN.1)	X.680–X.699
OSI MANAGEMENT	
Systems management framework and architecture	X.700–X.709
Management communication service and protocol	X.710–X.719
Structure of management information	X.720–X.729
Management functions and ODMA functions	X.730–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	
Commitment, concurrency and recovery	X.850–X.859
Transaction processing	X.860–X.879
Remote operations	X.880–X.889
Generic applications of ASN.1	X.890–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999
INFORMATION AND NETWORK SECURITY	X.1000–X.1099
SECURE APPLICATIONS AND SERVICES (1)	X.1100–X.1199
CYBERSPACE SECURITY	X.1200–X.1299
SECURE APPLICATIONS AND SERVICES (2)	X.1300–X.1499
CYBERSECURITY INFORMATION EXCHANGE	X.1500–X.1599
CLOUD COMPUTING SECURITY	X.1600–X.1699
QUANTUM COMMUNICATION	X.1700–X.1729
DATA SECURITY	X.1750–X.1799
5G SECURITY	X.1800–X.1819

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T X.609.5

Managed P2P communications: Overlay management protocol

Summary

Recommendation ITU-T X.609.5 specifies an overlay management protocol (OMP)that runs on the interface among entities of managed peer-to-peer (P2P) communications. The management functionalities covered in this Recommendation include overlay network management including creation, modification, and termination and peer management including membership control and information maintenance. The protocol is applicable to various services such as multimedia streaming service and content distribution service over managed P2P communications. This Recommendation provides protocol operations and message formats.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T X.609.5	2018-01-13	11	11.1002/1000/13494
2.0	ITU-T X.609.5	2020-04-29	11	11.1002/1000/14247

Keywords

Content distribution, managed P2P, multimedia streaming, overlay network management.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T X.609.5

Managed P2P communications: Overlay management protocol

1 Scope

This Recommendation specifies overlay management protocol for various services including multimedia streaming and content distribution over managed peer-to-peer infrastructure. This Recommendation describes the following details:

– overview of overlay management protocol,

– elements of protocol messages,

– protocol messages and its parameters,

– protocol behaviours including information flows.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T X.609] Recommendation ITU-T X.609 (2015), *Managed P2P communications: Functional architecture*.

[ITU-T X.609.1] Recommendation ITU-T X.609.1 (2016), *Managed P2P communications: Peer activity management protocol (PAMP)*.

[ITU-T X.609.2] Recommendation ITU-T X.609.2 (2016), *Managed P2P communications: Overlay resource control protocol (ORCP)*.

[ITU-T X.609.3] Recommendation ITU-T X.609.3 (2017), *Managed P2P communications: Multimedia streaming signalling requirements*.

[ITU-T X.609.4] Recommendation ITU-T X.609.4 (2018), *Managed P2P communications: Multimedia streaming peer protocol*.

[ITU-T X.609.6] Recommendation ITU-T X.609.6 (2018), *Managed P2P communications: Content distribution signalling requirements*.

[ITU-T X.609.7] Recommendation ITU-T X.609.7 (2018), *Managed P2P communications: Content distribution peer protocol*.

[IETF RFC 7159] IETF RFC 7159 (2014), *The JavaScript Object Notation (JSON) Data Interchange Format*.

[IETF RFC 7231] IETF RFC 7231 (2014), *Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content*.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 overlay network [b-ITU-T X.1162]: An overlay network is a virtual network that runs on top of another network. Like any other network, the overlay network comprises a set of nodes and links between them. Because the links are logical ones, they may correspond to many physical links of the underlying network.

3.1.2 peer [b-ITU-T X.1161]: Communication node on P2P network that functions simultaneously as both "client" and "server" to the other nodes on the network.

3.1.3 peer-to-peer (P2P) [b-ITU-T Y.2206]: A system is considered to be P2P if the nodes of the system share their resources in order to provide the service the system supports. The nodes in the system both provide services to other nodes and request services from other nodes.

NOTE – Peer is the node in a P2P system.

3.1.4 managed P2P [b-ISO/IEC TR 20002]: P2P with manageability features to manage the P2P-based service and P2P network by the P2P participants such as P2P service provider, ISP and peer.

3.1.5 fragment [ITU-T X.609]: A piece of the shared content.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 overlay ID: An identifier of an overlay network. Each overlay network can be identified by own overlay ID.

3.2.2 peer ID: An identifier of a peer. Each peer can be identified by own peer ID.

3.2.3 peer list: A list of peer IDs which identifies the peers participating in an overlay network.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

CS Cache Server

JSON Javascript Object Notation

MP2P Managed Peer-to-Peer

OMS Overlay Management Server

OMP Overlay Management Protocol

PAMS Peer Activity Management Server

MP2P Managed Peer-to-Peer

P2P Peer-to-Peer

REST Representational State Transfer

RS Relay Server

UMS User Management Server

UNIS Underlying Network Information Server

5 Conventions

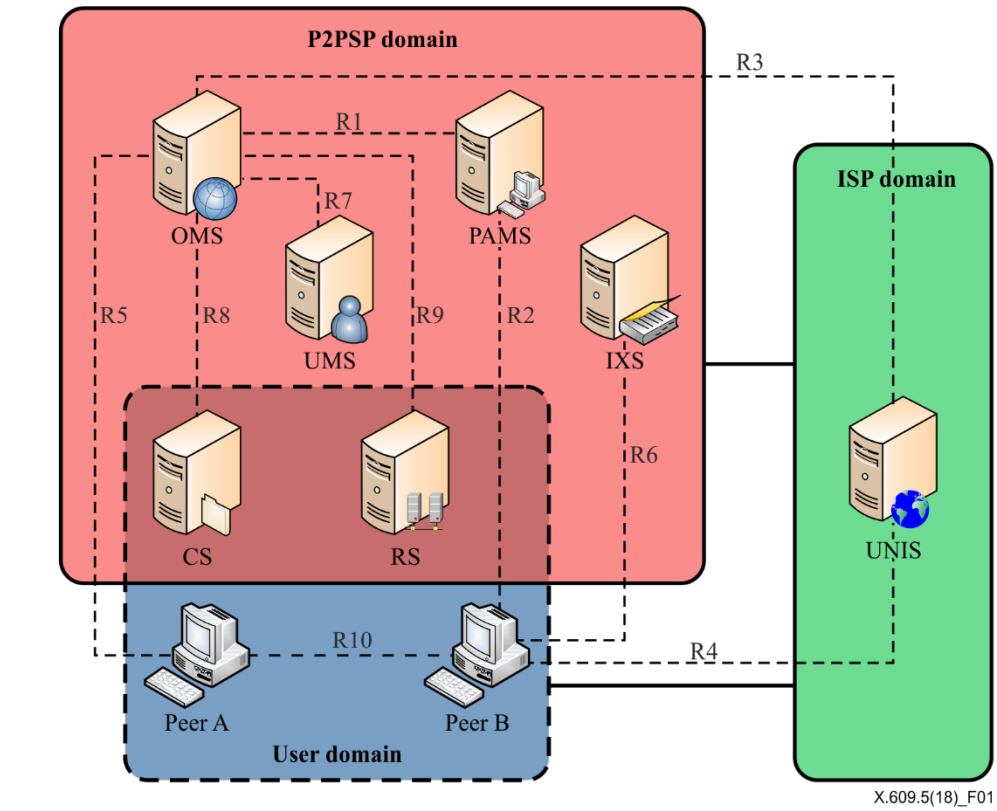
Resource elements in clause 8.1 are encoded in JavaScript object notation (JSON) [IETF RFC 7159], and the grammar used in representing objects defined in this Recommendation is as follows:

– "STRING", "BOOLEAN", and "NUMBER" types are used to indicate string, boolean and number, respectively;

– An array of collective values are enclosed in brackets "[ ]" with value separated by commas ",";
– Selective options are separated by a vertical bar " | ".

6 Overview

The framework of managed peer-to-peer (MP2P) communications is defined in [b-ISO/IEC TR 20002], and the details of the entities and the reference points among the entities are defined in [ITU-T X.609].



UMS: user profile management server
OMS: overlay management server
PAMS: peer activity management server
RS: relay server
CS: cache server
IXS: index server
UNIS: underlying network information server

Figure 1 – Framework and reference points of MP2P [ITU-T X.609]

Figure 1 shows the framework and reference points of MP2P communications. From the management point of view, the overlay management server (OMS) manages the information of established overlay networks and controls resources such as cache server (CS) and relay server (RS). A peer interacts with OMS to join a specific overlay network. The peer also interacts with OMS when it leaves the joined overlay network. This Recommendation defines the overlay management protocol (OMP) running over reference point R5, which is used for the management related to overlay network. OMS can interact with other entities such as peer activity management server (PAMS), underlying network information server (UNIS), and user management server (UMS) to form a well-organized overlay network, but those interactions are not part of this Recommendation.

The basic service flow for overlay network management in MP2P communication is shown in Figure 2.

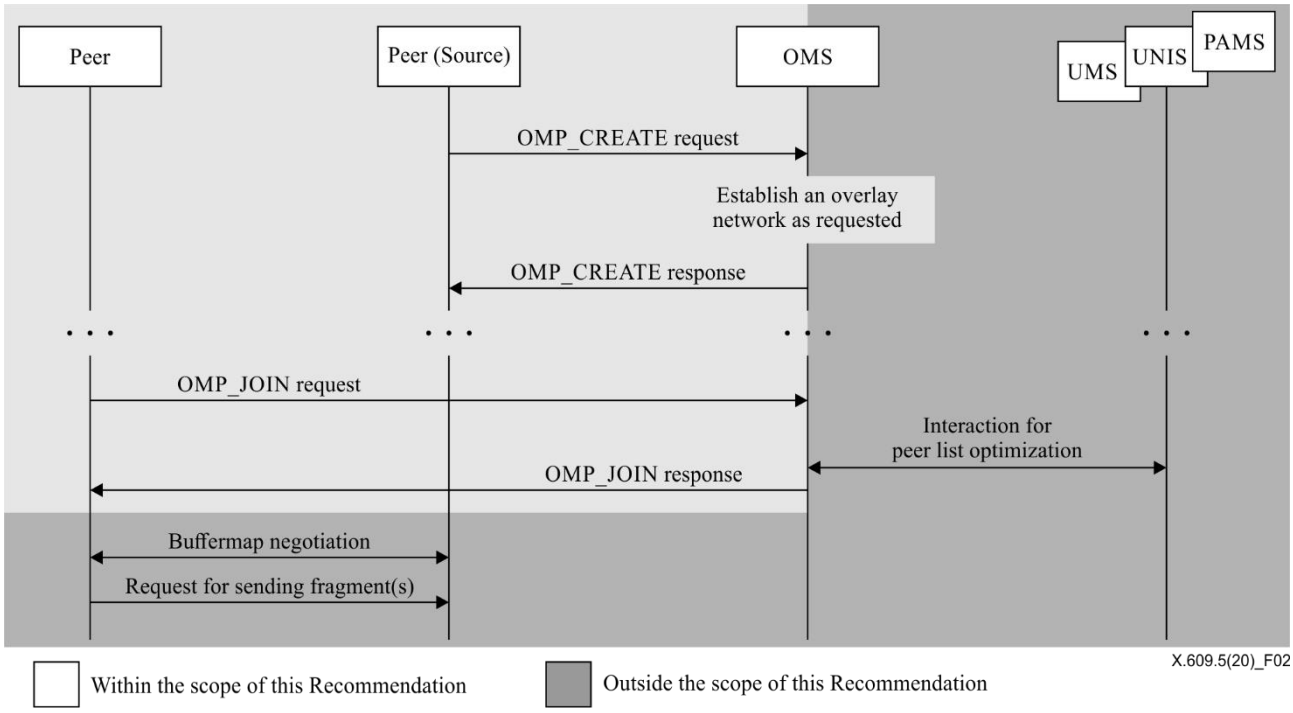


Figure 2 – Basic service flow of overlay network management

A peer requests OMS to create a new overlay network. If the request is valid, OMS establishes a new overlay network as requested and responds with the information of the established overlay network such as the identifier of the overlay network. After an overlay network is created, peers can join the overlay network by interacting with OMS. A peer requests OMS to join a specific overlay network. As a response, OMS provides a peer list that contains contact information of peers participating in the overlay network. The peer list can be generated by OMS itself, or OMS can interact with other servers such as UMS, UNIS and PAMS to generate the optimized peer list. Upon receiving the response including peer list, the peer contacts other peers on the list and conducts rest processes for receiving various types of content such as multimedia streams or files.

As depicted in Figure 2, this Recommendation covers the interactions between peer and OMS, which runs over reference point R5 shown in Figure 1. The interactions can be classified into two categories. The first category is overlay network management. Overlay network management is about creation, modification, and termination of an overlay network. The second category is peer management. Peer management is about joining and leaving a specific overlay network. In addition, the Recommendation includes the interaction for query which is within the scope of both categories. For instance, a peer can query the information of a specific overlay network or the peer list of a specific overlay network.

7 Protocol operation

7.1 Overlay network management

7.1.1 Creation of overlay network

A peer can request OMS to create a new overlay network. The request may specify parameters such as peer ID of the peer that sent the request or a list of peer IDs for closed group communication. Upon receiving the request, OMS checks whether the request is valid. If the request is valid, OMS creates an overlay network as requested. When OMS creates the overlay network, it should also generate a unique overlay ID for the overlay network since each overlay network is distinguished by own overlay ID. After completing creation, OMS responds with the overlay ID. This process is depicted in Figure 3. This Recommendation does not specify how the overlay ID is generated.

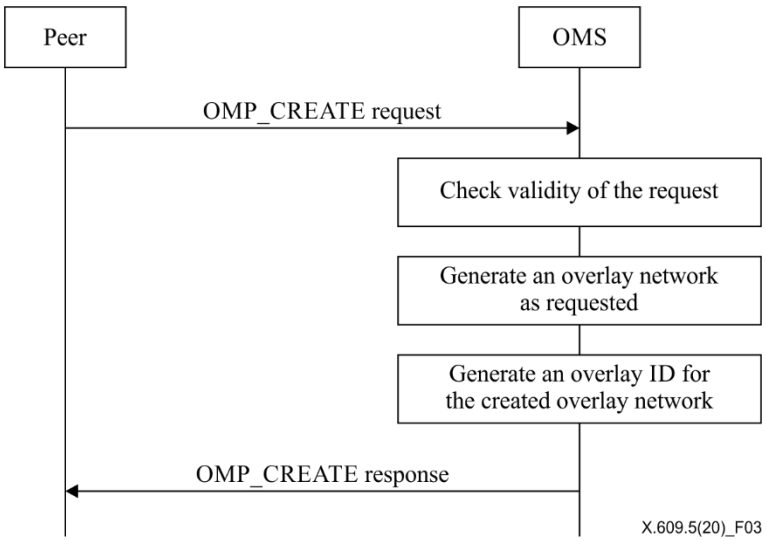


Figure 3 – Operations for overlay network creation

7.1.2 Modification of overlay network

A peer can request OMS to modify information of the overlay network. To prevent unauthorized modification, OMS accepts the request only if the request is issued by the peer that requested OMS to create the overlay network. Any parameter configured when the overlay network is created can be modified, and new parameters can be newly configured by the modification operations. OMS checks validity of the request upon receiving the request and performs modifications as requested. When OMS completes the requested modification, it responds that modification has been successful. This process is depicted in Figure 4.

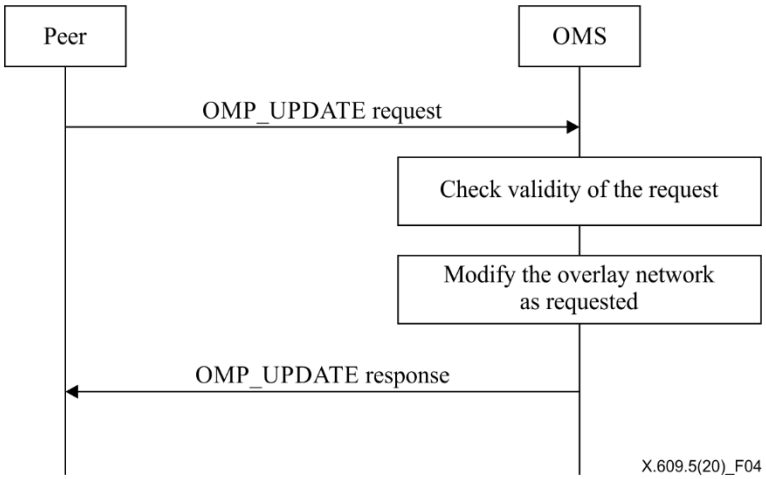


Figure 4 – Operations for overlay network modification

7.1.3 Termination of overlay network

A peer can request OMS to terminate the overlay network. To prevent unauthorized termination, OMS accepts the request only if the request is issued by the peer that requested OMS to create the overlay network. After checking validity of the request, OMS removes all data related to the overlay network from itself. When OMS completes the requested termination, it responds that termination has been successful. This process is depicted in Figure 5.

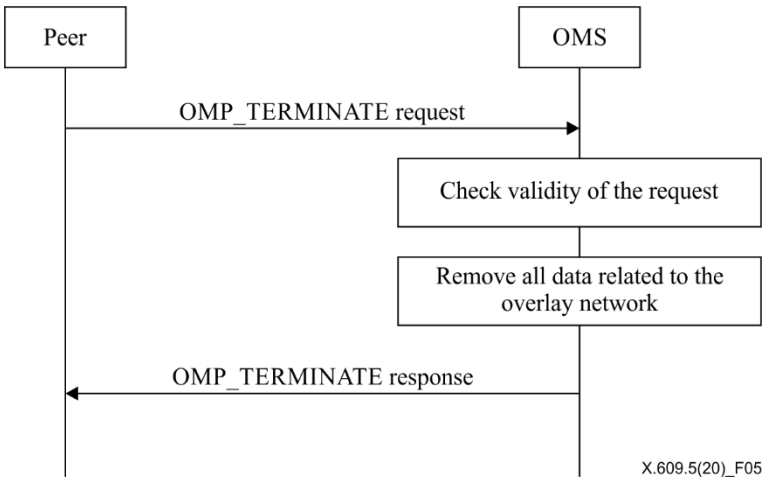


Figure 5 – Operations for overlay network termination

7.1.4 Query of overlay network

A peer can send OMS a query about the status of overlay networks. When the query includes a specific overlay ID, OMS responds with the status information of the corresponding overlay network. If the query does not specify any overlay network, OMS responds with the information of all the overlay networks that it manages. The query may also specify the peer ID of a specific peer. In such case, OMS responds with the information about the overlay network that the specified peer had created. This process is depicted in Figure 6.

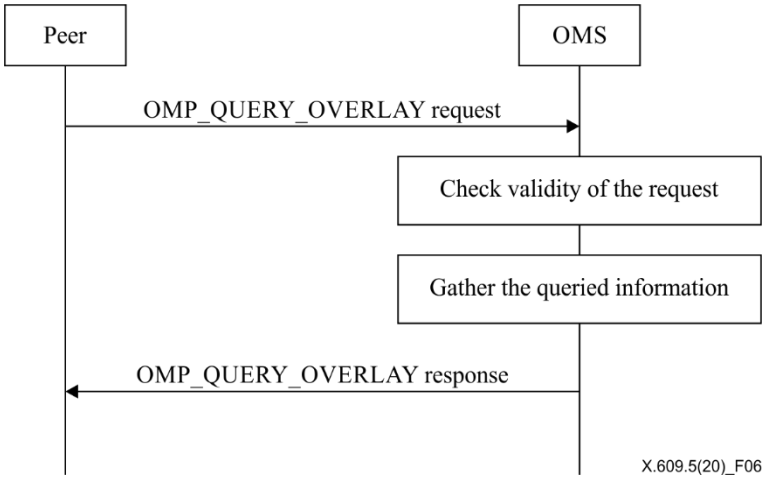


Figure 6 – Operations for overlay network query

7.2 Peer management

7.2.1 Overlay network join

To join a specific overlay network, a peer sends OMS a request to join the target overlay network. If the request is valid, OMS responds with a peer list that contains the contact information and peer ID of peers already participating in the overlay network. The request from the peer may include network information of the requesting peer so that OMS can manage the peer list of each overlay network. In addition, the request may additionally include an authentication key if authentication is needed to join the overlay network. This process is depicted in Figure 7.

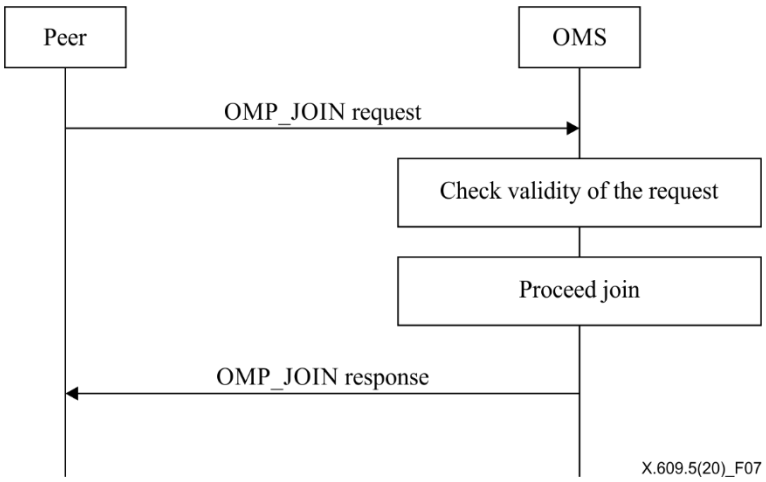


Figure 7 – Operations for overlay network joining

7.2.2 Peer information update

After joining an overlay network, a peer can send a request for updating its information. The request can be considered as renewing the subscription so that OMS can keep the status of the requesting peer alive. As a response, OMS sends the latest version of the peer list. Then the requesting peer can communicate with peers on the peer list. This process is depicted in Figure 8.

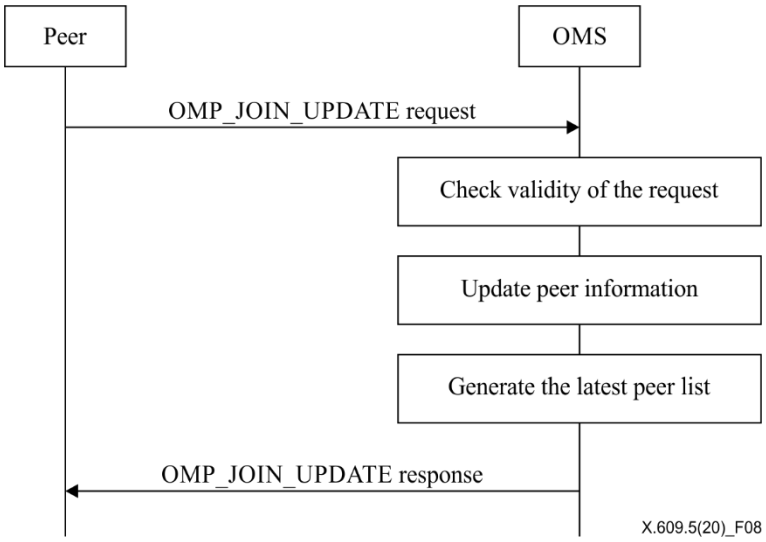


Figure 8 – Operations for peer information update

7.2.3 Overlay network leave

A peer can leave the joined overlay network anytime without the need to notify OMS. However, the peer may notify OMS of its leave for a graceful exit. The graceful exit helps OMS manage overlay networks precisely. OMS considers that a specific peer leaves if a specific peer does not send its peer information update request within a predefined period of time. This process is depicted in Figure 9.

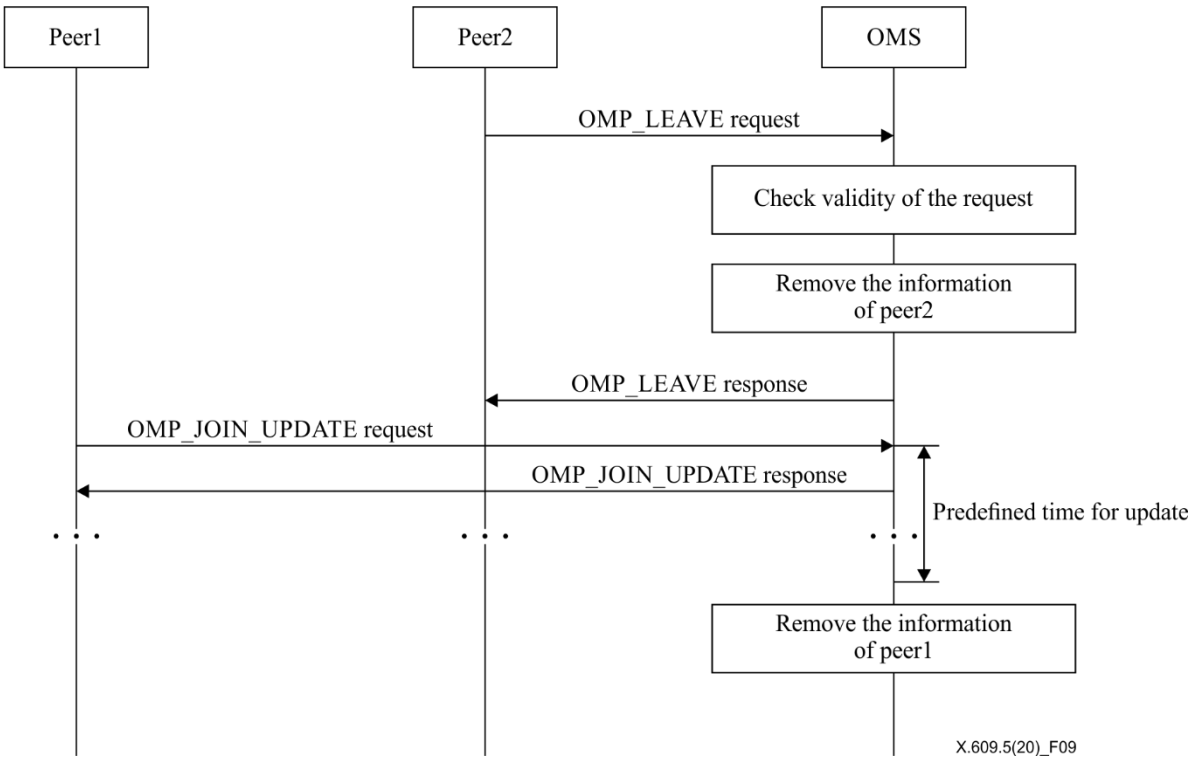


Figure 9 – Operations for overlay network leave

7.2.4 Peer list query

A peer can request OMS to send peer list of a specific overlay network. The request may include conditions of querying so that OMS can find the corresponding peers and respond with them. The condition includes list of fragments or range of fragments. Then, OMS responds with the information about the corresponding peers which possesses the fragments specified in the request. This process is depicted in Figure 10.

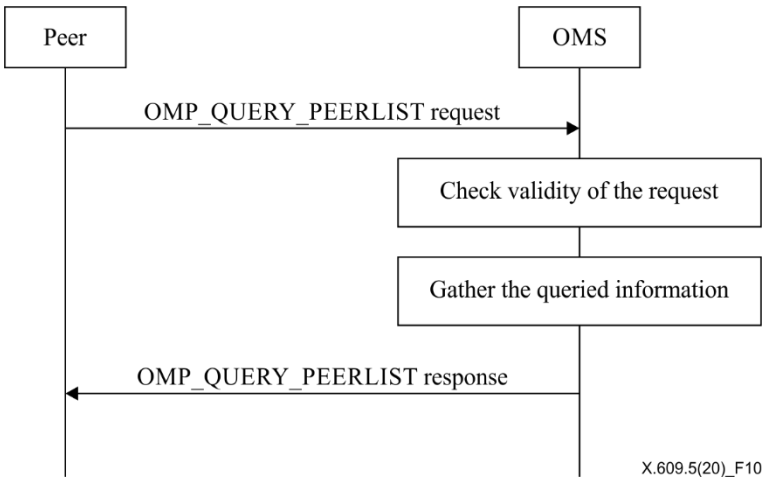


Figure 10 – Operations for peer list query

8 Messages

8.1 Resource element type

8.1.1 Peer information element

The peer information element provides the information about peer. The peer information element is defined as follows:

– Object {
– STRING peer_id;
– network net_info;
– } peer_info

The description of the attributes is as follows:

– *peer_id* is an identifier of a peer;
– *net_info* is an object of network element.

8.1.2 Network element

The network element provides the information regarding network address. The network element is defined as follows:

– Object {
– STRING ip-address;
– INTEGER port;
– BOOLEAN public;
– } network

The description of the attributes is as follows:

– *ip-address* is an IP address of a peer;
– *port* is a port number of a peer;
– *public* indicates whether ip-address is a public IP address. The value is set to TRUE, if ip-address is a public address.

8.1.3 Peer list element

The peer list element provides the information about peers and fragments. The peer list element is defined as follows:

– Object {
– peer_info [peer_info];
– } peer_list

The description of the attributes is as follows:

– *peer_info* is an array of peer_info objects.

8.1.4 Overlay network information element

The overlay network information element provides identification information of an overlay network.

The overlay network information element is defined as follows:

– Object {
– STRING version;
– STRING overlay-network-id;

– STRING index-url;
– STRING owner-id;
– INTEGER expires;
– pam_conf_info pam_conf;
– auth_info auth;
– overlay_status status;
– peer_list peer_list
– } overlay_network_information;

The description of the attributes is as follows:

– *version* indicates the version of the overlay network identified by *overlay-network-id*;
– *overlay-network-id* is an identifier of overlay network;
– *index-url* is an URL address of index server;
– *owner-id* is an identifier of peer which created the overlay network;
– *expires* indicates expiration time. Peers belonging to the overlay network is required to update their information within the expiration time;
– *pam_conf* is an object of pam_conf_info element;
– *auth* is an object of auth_info element;
– *status* is an object of overlay_status element;
– *peer_list* is an object of peer_list element.

8.1.5 Overlay network list element

The overlay network list element provides the information about overlay networks. The overlay network list element is defined as follows:

– Object {
– STRING [overlay_network_id];
– } overlay_network_list

The description of the attributes is as follows:

– *overlay_network_id* is an array of identifiers of each overlay network. The array contains ordered list of overlay network identifiers.

8.1.6 Fragment list element

The fragment list element provides the information about fragments. The fragment list element is defined as follows:

– Object {
– NUMBER num_of_fragment;
– NUMBER fragment_size;
– NUMBER [fragment];
– } fragment_list

The description of the attributes is as follows:

– *num_of_fragment* indicates the total number of fragments organizing the content shared in an overlay network;
– *fragment_size* indicates the size of fragment in kilobytes;

– *fragment* is an array of identifiers of each fragment. fragment_list element can include list of fragment IDs.

8.1.7 Fragment range element

The fragment range element provides the information about range of fragments. The fragment range element is defined as follows:

– Object {
– NUMBER start_fragment_id;
– NUMBER end_fragment_id;
– } fragment_range

The description of the attributes is as follows:

– *start_fragment_id* is an identifier of the first fragment in the range of fragments;
– *end_fragment_id* is an identifier of the last fragment in the range of fragments.

8.1.8 Peer activity management configuration information element

The peer activity management (PAM) configuration information element provides the configuration information about PAM. The PAM configuration information element is defined as follows:

– Object {
– BOOLEAN pam_enabled*;*
– STRING pams_url*;*
– INTERGER report_interval*;*
– } pam_conf_info

The description of the attributes is as follows:

– *pam_enabled* indicates whether PAM function is enabled. If the value is set to *true*, PAM function is enabled;
– *pams_url* is a URL of PAMS. pam_url is used to indicate the location of PAMS;
– *report_interval* is periodic interval in second for reporting dynamic status by peer.

8.1.9 Authentication information element

The authentication information element provides the information related to authentication required for an overlay network. The authentication information element is defined as follows:

– Object {
– STRING closed*;*
– STRING auth-key*;*
– STRING [user_id];
– } auth_info

The description of the attributes is as follows:

– *closed* indicates whether an overlay network is closed group. The value is set to YES, if the overlay network is a closed group. In addition, user-id array should be specified in order to list members of the group. The value is set to NO, if the overlay network is not a closed group. The value can be set to AUTH if the overlay network is a closed group requiring a specific authentication key for joining the group;
– *auth-key* is an authentication key. This attribute is set only if attribute *closed* is set to AUTH;
– *user-id* is an array of identifiers of each peer. The array contains list of peer identifiers so that the listed peers can join the overlay network.

8.1.10 Overlay status element

The overlay status element provides the status of an overlay network. The overlay status element is defined as follows:

– Object {
– INTEGER num-of-seed*;*
– INTEGER num-of-leech*;*
– STRING time-of-start;
– STRING time-of-last-activity;
– } overlay_status

The description of the attributes is as follows:

– *num-of-seed* is the number of seeds in an overlay network;
– *num-of-leech* is the number of leeches in an overlay network;
– *time-of-start* indicates the time when an overlay network is created;
– *time-of-last-activity* indicates the time when the latest report from a peer is received.

8.2 Message format

This clause provides the format of messages for the operations explained in clause 7. All operations have request and response messages. For extensibility, OMP uses representational state transfer (REST)-ful architecture [b-REST].

8.2.1 OMP_CREATE

OMP_CREATE is initiated by a peer to create a new overlay network into OMS. On the basis of the request for OMP_CREATE, OMS can create an overlay network.

8.2.1.1 Request

The request message format for OMP_CREATE is shown in Table 1.

Table 1 – Request message format for OMP_CREATE

Method	POST
URI	**http://{OMS_ADDRESS}[a]/overlay_networks/**
Body	overlay_network_information (refer to clause 8.1.4)
[a] {OMS_ADDRESS} refers to the FQDN address of OMS	

An example HTTP request message for OMP_CREATE is as follows:

```
POST /overlay_networks/ HTTP/1.1
Host: www.exampleoms.com
Content-Length: 122
Content-Type: application/json
Accept: application/json
{
   "overlay_network_information" : {
          "version" : 1,
          "owner-id" : "8djdhd",
          "expires" : 5,
```

```
            "pam_conf" : {
                 "pam_enabled" : "TRUE"
            },
            "auth" : {
                  "closed" : "YES",
                  "auth-key" : "9i8u7y",
                  "user-id" : [7y6t5r, 9i8u7y52]
            }
      }
   }
```

8.2.1.2 Response

The response to an OMP_CREATE request uses a response code to indicate the result. Table 2 lists the response codes and semantics for OMP_CREATE. This Recommendation follows [IETF RFC 7231] for other response codes.

Table 2 – Response codes for OMP_CREATE

Response code and semantics		Body
200	OK The request is accepted and creation is done.	overlay_network_information (refer to clause 8.1.4)
401	Unauthorized The request requires user authentication. A peer may repeat the request with a suitable Authorization in HTTP header.	N/A

An example HTTP response message for OMP_CREATE is as follows:

```
HTTP/1.1 200 OK
Content-Length: 118
Content-Type: application/json
Connection: Closed
{
   "overlay_network_information" : {
          "version" : 1,
       "overlay_network_id" : "12ekd4kd8",
   "index-url" : "http://www.exampleidx.com/ 12ekd4kd8",
          "owner-id" : "8djdhd",
          "expires" : 5,
          "pam_conf" : {
               "pam_enabled" : "TRUE",
               "pams_url" : "http://www.examplepams.com/",
               "report_interval" : 3
           }
```

```
    }
}
```

8.2.2 OMP_UPDATE

OMP_UPDATE is initiated by a peer to update the information of a specific overlay network. OMS allows the request only if the corresponding overlay network has been created by the requesting peer.

8.2.2.1 Request

The request message format for OMP_UPDATE is shown in Table 3.

Table 3 – Request message format for OMP_UPDATE

Method	PUT
URI	**http://{OMS_ADDRESS}[a]/overlay_networks/{NID}[b]**
Body	overlay_network_information (refer to clause 8.1.4)
[a] {OMS_ADDRESS} refers to the FQDN address of OMS. [b] {NID} refers to the ID of overlay network.	

An example HTTP request message for OMP_UPDATE is as follows:

```
PUT /overlay_networks/12ekd4kd8/ HTTP/1.1
Host: www.exampleoms.com
Content-Length: 122
Content-Type: application/json
Accept: application/json
{

"overlay_network_information" : {
          "version" : 2,
   "index-url" : "http://www.exampleixs.com/12ekd4kd8",
          "owner-id" : "8djdhd",
          "expires" : 5,
          "auth" : {
                "closed" : "NO"
          }
   }
}
```

8.2.2.2 Response

The response to an OMP_UPDATE request uses a response code to indicate the result. Table 4 lists the response codes and semantics for OMP_UPDATE. This Recommendation follows [IETF RFC 7231] for other response codes.

Table 4 – Response codes for OMP_UPDATE

Response code and semantics		Body
200	OK The request is accepted and creation is done.	N/A
401	Unauthorized The request requires user authentication. A peer may repeat the request with a suitable Authorization in HTTP header.	N/A

8.2.3 OMP_TERMINATE

OMP_TERMINATE is initiated by a peer to terminate a specific overlay network. OMS allows termination, only if the requesting peer created the overlay network.

8.2.3.1 Request

The request message format for OMP_TERMINATE is shown in Table 5.

Table 5 – Request message format for OMP_TERMINATE

Method	DELETE
URL	**`http://{OMS_ADDRESS}`**[a]**`overlay_networks/{NID}`**[b]**`/`**
Body	N/A
[a] {OMS_ADDRESS} refers to the FQDN address of OMS. [b] {NID} refers to the ID of overlay network to be deleted.	

8.2.3.2 Response

The response to an OMP_TERMINATE request uses a response code to indicate the result. Table 6 lists response codes and semantics for OMP _TERMINATE. This Recommendation follows [IETF RFC 7231] for other response codes.

Table 6 – Response codes for OMP_TERMINATE

Response code and semantics		Body
200	OK The request is accepted and deregistration is done.	N/A
401	Unauthorized The request requires user authentication. OMS may repeat the request with a suitable Authorization in HTTP header.	N/A
404	Not Found The request is denied because there is no responding overlay network with the requested identifier.	N/A

8.2.4 OMP_QUERY_OVERLAY

OMP_QUERY_OVERLAY is initiated by a peer to query the status of a specific overlay network.

8.2.4.1 Request

The request message format for OMP_QUERY_OVERLAY is shown in Table 7.

Table 7 – Request message format for OMP_QUERY_OVERLAY

Method	GET
URI	**http://{OMS_ADDRESS}[a]/overlay_networks/{NID}[b]**
Body	N/A
[a] {OMS_ADDRESS} refers to the FQDN address of OMS. [b] {NID} refers to the ID of overlay network.	

8.2.4.2 Response

The response to an OMP_QUERY_OVERLAY request uses a response code to indicate the result. Table 8 lists response codes and semantics for OMP_QUERY_OVERLAY. This Recommendation follows [IETF RFC 7231] for other response codes.

Table 8 – Response codes for OMP_QUERY_OVERLAY

Response code and semantics		Body
200	OK The request is succeeded and this response contains peer list.	overlay_network_information (refer to clause 8.1.2)
401	Unauthorized The request requires user authentication. Peer may repeat the request with a suitable Authorization in HTTP header	N/A
404	Not Found The request is denied because there is no responding peer with the requested identifier.	N/A

An example HTTP response message for OMP_QUERY_OVERLAY is as follows:

```
HTTP/1.1 200 OK
Content-Length: 255
Content-Type: application/json
{
"overlay_network_information" : {
          "version" : 2,
          "owner-id" : "8djdhd",
          "expires" : 5,
          "auth" : {
                "closed" : "NO"
          }
          "overlay_status" : {
                "num-of-seed" : 3,
                "num-of-leech" : 10,
                "time-of-start" : "1d10m55s",
```

```
                    "time-of-last-activity" : "1s"
                }
    }
```

8.2.5 OMP_JOIN

OMP_JOIN is initiated by a peer to join a specific overlay network. OMS responds with the peer list containing the peers already participating in the overlay network.

8.2.5.1 Request

The request message format for OMP_JOIN is shown in Table 9.

Table 9 – Request message format for OMP_JOIN

Method	POST
URI	**http://{OMS_ADDRESS}[a]/overlay_networks/{NID}[b]/peer/**
Body	peer_information (refer to clause 8.1.1)
[a] {OMS_ADDRESS} refers to the FQDN address of OMS. [b] {NID} refers to the ID of overlay network.	

An example HTTP request message for OMP_JOIN is as follows:

```
POST /overlay_networks/12ekd4kd8/peer/ HTTP/1.1
Host: www.exampleoms.com
Content-Length: 117
Content-Type: application/json
Accept: application/json
{
   "peer_information" : {
      "peer_id" : "8djdhd",
      "net_info" : {
                "ip-address" : "123.1.2.3",
                "port" : 5241,
                "public" : "YES"
         }
   }
}
```

8.2.5.2 Response

The response to an OMP_JOIN request uses a response code to indicate the result. Table 10 lists response codes and semantics for OMP_JOIN. This Recommendation follows [IETF RFC 7231] for other response codes.

Table 10 – Response codes for OMP_JOIN

Response code and semantics		Body
200	OK The request is accepted and registration is done.	overlay_network_information (refer to clause 8.1.4)

Table 10 – Response codes for OMP_JOIN

Response code and semantics		Body
404	Not Found The request is denied because there is no responding overlay network with the requested identifier.	N/A
409	Conflict The request is denied because peer with the same identifier is already joined.	N/A

An example HTTP response message for OMP_JOIN is as follows:

```
HTTP/1.1 200 OK
Content-Length: 118
Content-Type: application/json
{
  "overlay_network_information" : {
          "version" : 2,
          "expires" : 5,
          "pam_conf" : {
               "pam_enabled" : "TRUE",
               "pams_url" : "http://www.examplepams.com/",
               "report_interval" : 3
           },
           "peer_list" : {
               "peers" :  ["peerd", "peerb", "peerc"]
           }
  }
}
```

8.2.6 OMP _JOIN_UPDATE

OMP _JOIN_UPDATE is initiated by a peer to renew its subscription regarding specific overlay networks. A peer periodically requests OMS to renew its subscription by sending an OMP _JOIN_UPDATE request.

8.2.6.1 Request

The request message format for OMP_JOIN_UPDATE is shown in Table 11.

Table 11 – Request message format for OMP_JOIN_UPDATE

Method	PUT
URI	**http://{OMS_ADDRESS}**[a]**/overlay_networks/{NID}**[b]**/peer/{PID}**[c]
Body	peer_information (refer to clause 8.1.1), auth_info (refer to clause 8.1.9)
[a] {OMS_ADDRESS} refers to the FQDN address of OMS. [b] {NID} refers to the ID of overlay network. [c] {PID} refers to the ID of peer to be updated.	

An example HTTP request message for OMP_JOIN_UPDATE is as follows:

```
PUT /overlay_networks/12ekd4kd8/peer/8djdhd HTTP/1.1
Host: www.exampleaoms.com
Content-Length: 117
Content-Type: application/json
Accept: application/json
{
   "peer_information" : {
      "peer_id" : "8djdhd"
   },
   "auth_info" : {
         "auth-key" : "78ue3ee2"
    }
}
```

8.2.6.2 Response

The response to an OMP_JOIN_UPDATE request uses a response code to indicate the result. Table 12 lists response codes and semantics for OMP_JOIN_UPDATE. This Recommendation follows [IETF RFC 7231] for other response codes.

Table 12 – Response codes for OMP_JOIN_UPDATE

Response code and semantics		Body
200	OK The request is accepted and registration is done.	overlay_network_information (refer to clause 8.1.4)
404	Not Found The request is denied because there is no responding peer with the requested identifier.	N/A

An example HTTP response message for OMP_JOIN_UPDATE is as follows:

```
HTTP/1.1 200 OK
Content-Length: 118
Content-Type: application/json
{
  "overlay_network_information" : {
          "expires" : 2
   }
}
```

8.2.7 OMP_LEAVE

OMP_LEAVE is initiated by a peer to leave a specific overlay network.

8.2.7.1 Request

The request message format for OMP_LEAVE is shown in Table 13.

Table 13 – Request message format for OMP_LEAVE

Method	DELETE
URI	**http://{OMS_ADDRESS}[a]/overlay_networks/{NID}[b]/peer/{PID}[c]**
Body	N/A
[a] {OMS_ADDRESS} refers to the FQDN address of OMS. [b] {NID} refers to the ID of overlay network. [c] {PID} refers to the ID of leaving peer.	

8.2.7.2 Response

The response to an OMP_LEAVE request uses a response code to indicate the result. Table 14 lists response codes and semantics for OMP_LEAVE. This Recommendation follows [IETF RFC 7231] for other response codes.

Table 14 – Response codes for OMP_LEAVE

Response code and semantics		**Body**
200	OK The request is accepted and registration is done.	N/A
401	Unauthorized The request requires user authentication. Peer may repeat the request with a suitable Authorization in HTTP header	N/A
404	Not Found The request is denied because there is no responding peer with the requested identifier.	N/A

8.2.8 OMP_QUERY_PEER

OMP_QUERY_PEER is initiated by a peer to query the status of a specific peer joining a specific overlay network.

8.2.8.1 Request

The request message format for OMP_QUERY_PEER is shown in Table 15.

Table 15 – Request message format for OMP_QUERY_PEER

Method	GET
URI	**http://{OMS_ADDRESS}[a]/overlay_networks/{NID}[b]/peer/{PID}[c]**
Body	N/A
[a] {OMS_ADDRESS} refers to the FQDN address of OMS. [b] {NID} refers to the ID of overlay network. [c] {PID} refers to the ID of leaving peer.	

8.2.8.2 Response

The response to an OMP_QUERY_PEER request uses a response code to indicate the result. Table 16 lists response codes and semantics for OMP_QUERY_PEER. This Recommendation follows [IETF RFC 7231] for other response codes.

Table 16 – Response codes for OMP_QUERY_PEER

Response code and semantics		Body
200	OK The request is succeeded and this response contains peer list.	peer_information (refers to clause 8.1.1)
401	Unauthorized The request requires user authentication. Peer may repeat the request with a suitable Authorization in HTTP header	N/A
404	Not Found The request is denied because there is no responding peer with the requested identifier.	N/A

An example HTTP response message for OMP_QUERY_PEER is as follows:

```
HTTP/1.1 200 OK
Content-Length: 145
Content-Type: application/json
{
  "peer_information” : {
      "peer_id” : “8djdhd”,
      “net_ifo” : {
         “ip-address” : “123.1.2.3”,
         “port” : 5241,
         “public” : “TRUE”
    }
}
```

8.2.9 OMP_QUERY_PEERLIST

OMP_QUERY_PEERLIST is initiated by a peer to request OMS to send a peer list of a specific overlay network. The body of the request message may include a query condition so that peers possessing a certain range of fragment can be queried.

8.2.9.1 Request

The request message format for OMP_QUERY_PEERLIST is shown in Table 17.

Table 17 – Request message format for OMP_QUERY_PEERLIST

Method	GET
URI	**http://{OMS_ADDRESS}[a]/overlay_networks/{NID}[b]/peer/**
Body	fragment_list (refer to clause 8.1.6), fragment_range (refer to clause 8.1.7)
[a] {OMS_ADDRESS} refers to the FQDN address of OMS. [b] {NID} refers to the ID of overlay network.	

An example HTTP request message for OMP_QUERY_PEERLIST is as follows:

```
GET /overlay_networks/12ekd4kd8/peer/8djdhd HTTP/1.1
Host: www.exampleaoms.com
Content-Length: 73
Content-Type: application/json
Accept: application/json
{
  "fragment_range":
  {
    "start_fragment_id":0,
    "end_fragment_id":99
  }
}
```

8.2.9.2 Response

The response to an OMP_ QUERY_PEERLIST request uses a response code to indicate the result. Table 18 lists response codes and semantics for OMP_QUERY_PEERLIST. This Recommendation follows [IETF RFC 7231] for other response codes.

Table 18 – Response codes for OMP_QUERY_PEERLIST

Response code and semantics		Body
200	OK The request is succeeded and this response contains peer list.	peer_list (refers to clause 8.1.3)
401	Unauthorized The request requires user authentication. Peer may repeat the request with a suitable Authorization in HTTP header	N/A
404	Not Found The request is denied because there is no responding peer with the requested identifier.	N/A

An example HTTP response message for OMP_QUERY_PEERLIST is as follows:

```
HTTP/1.1 200 OK
Content-Length: 255
Content-Type: application/json
{
  "peer_list" : {
    "peers" :  [“peerd”, “peerb”, “peerc”]
   }
  "fragment_list" : {
    "fragment" : [100, 102]
  },
  "fragment_range":
  {
    "start_fragment_id":0,
    "end_fragment_id":99
   }
}
```

Bibliography

[b-ITU-T X.1161] Recommendation ITU-T X.1161 (2008), *Framework for secure peer-to-peer communications*.

[b-ITU-T X.1162] Recommendation ITU-T X.1162 (2008), *Security architecture and operations for peer-to-peer networks.*

[b-ITU-T Y.2206] Recommendation ITU-T Y.2206 (2010), *Requirements for distributed service networking capabilities*.

[b-REST] Fielding, R. (2000), *Architectural Styles and the Design of Network-based Software Architectures*, Doctoral Dissertation, University of California, Irvine, September.

[b-ISO/IEC TR 20002] ISO/IEC TR 20002 (2012), *Information technology – Telecommunications and information exchange between systems – Managed P2P: Framework.*

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	**Data networks, open system communications and security**
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2020